

Learning Deep Neural Networks: Necessity and Scope of Prior Knowledge, Raw Data, and Labels



Yuki M. ASANO

Balliol College

University of Oxford

Supervisor: Professor Andrea Vedaldi
Work submitted for the D.Phil degree

1st July 2021

Abstract

The recent rise in machine learning has been largely made possible by novel algorithms, such as convolutional neural networks and large-scale labelled datasets. Yet obtaining labelled datasets is expensive, does not scale well, and should not be necessary for learning *general* representations of data such as images or videos. Thus, by shifting the training of neural networks to not require labelled data, it is possible to obtain more robust, diverse and generalizable neural networks that can scale to the vast quantities of readily available unlabelled data. Learning transferable representations from raw data can thus drastically reduce the cost of solving new tasks and improve performance in many settings where supervisory signals are scarce. The field of self-supervised learning has therefore become an increasingly popular framework for learning without labels and in this thesis I provide several works that shed light on the workings of self-supervised learning, and that contributed to, and shaped the state of the art in this field. This thesis covers works that: i) analyze the role of image transformations; ii) develop methods for self-supervised clustering in the image and video domain; iii) develop novel representation learning methods for video-audio and video-text data and; iv) propose a new dataset better suited to self-supervised pretraining than the current standard. Put together, these works investigate and further the state of self-supervised learning from multiple dimensions and its insights continue to shape the future of deep learning without labels.

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfillment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Yuki M. Asano, June 2021.

Acknowledgements

I am indebted to the consistent support and genius guidance of my supervisor Andrea Vedaldi. Without his intuition, rigor and creativity for developing new ideas, this thesis would not have been possible. Similarly I am extremely fortunate and grateful for working with Christian Rupprecht who has helped me navigate this field. I also thank Andrew Zisserman for his support and advice and Doyne Farmer for being a true mentor.

I thank also my collaborators Mandela Patrick, with whom I have worked on many of the topics in this thesis and without whom research would have not been as fun and productive, and Triantafyllos Afouras with whom I should have started working a lot earlier. I also thank Liliane Momeni, Andrew Brown, Samuel Albanie and Gül Varol and my other VGG members for their support and camaraderie.

I thank the EPSRC AIMS CDT, and the Qualcomm Innovation Fellowship for funding, and within the CDT in particular Wendy Poole whose tireless support has been the backbone of many CDT students, including myself.

I thank my friends and collaborators outside of computer vision: Gesa for not only being a genius in thinking about and working on sustainability, but also for being a friend and continous inspiration. Maria for always being there. Šimon and Clint for keeping my knowledge in mathematics and my running legs fresh. I also thank all my other friends that I have met in Oxford, Munich, Tokyo, Santa Fe, Sicily and Tallinn that continuously make me see things from a new perspective and show me the world is full of amazing people.

Finally, I wish to thank my family. To my parents, who have always supported and encouraged me, be it with packages of cookies or warm words. To my brother who taught me Python and constantly inspires me to keep going. To my late grandpa who introduced me to the world of nature and engineering. Finally to Eva, I am immensely thankful for you being my pillar of support, my hiking/climbing/gaming/travelling/food partner, advisor, fan, therapist and biggest source of happiness.

Contents

1	Introduction and Overview	7
1.1	Motivation	7
1.2	Key Ideas and Literature Review	10
1.2.1	Self-supervised Learning	10
1.2.2	Multi-modal Learning	17
1.3	Publication List	19
2	Conclusion	22
2.1	Summaries and Impact	22
2.2	Open Challenges	27
3	A critical analysis of self-supervision, or what we can learn from a single image	29
4	Self-labelling via simultaneous clustering and representation learning	47
5	Labelling unlabelled videos from scratch with multi-modal self-supervision	70

Contents

6	Support-set bottlenecks for video-text representation learning	87
7	Multi-modal Self-Supervision from Generalized Data Transformations	106
8	Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning	124
9	Self-supervised object detection from audio-visual correspondence	141
10	PASS: An ImageNet replacement for self-supervised pretraining without humans	160
	Bibliography	187
	Appendices	

1

Introduction and Overview

In this chapter, we give an overview on the research questions tackled in this thesis along with an overview of the existing literature.

1.1 Motivation

Since the first successful training of a deep neural network on a large scale dataset of over a million images in 2012 [Krizhevsky et al., 2012], their performance, applications, and our understanding of them has kept increasing. While this advance is also due to the utilisation of faster computation via Graphical Processing Units (GPUs), a key ingredient was the major advances in algorithms, most notably the use of convolutional neural networks (CNNs) [LeCun et al., 1998]. Importantly, it has been found that these models, once trained, can *transfer* to other tasks and domains relatively well [Zeiler and Fergus, 2014; Yosinski et al., 2014]. This means that a model trained for *e.g.* image classification can be re-used and easily finetuned on a different task, such as object detection or segmentation, while requiring less data than otherwise necessary. The encoding that these networks learn, formally described as *representations*, are therefore of general nature, and extract widely useful properties of the input data.

1. Introduction and Overview

However, the performance of the CNNs—and more recently architectures such as the vision transformer [Dosovitskiy et al., 2021]—greatly depends on the amount of training data supplied. While ever larger and richer datasets such as ImageNet [Deng et al., 2009b], COCO [Lin et al., 2014], Conceptual Captions [Sharma et al., 2018] or OpenImages [Kuznetsova et al., 2018] have been developed and have improved the performance of these algorithms, deep learning progress has relied on having *human generated labels* for these datasets which indicate what is in a picture or even which pixel belongs to what object category. This process is slow and expensive, making labelling new datasets a bottleneck for the whole computer vision community. Therefore, not relying on labelled data is crucial in several aspects, it allows for larger scale, better accessibility, fundamental insights, and avoidance of ambiguity.

Scale. Data, such as images and videos are abundant on the web. Combined with the fact that current methods positively scale with the amount of training data supplied [Hestness et al., 2017; Goyal et al., 2021] this means that large improvements can be expected when the need for labeled data is abolished.

Accessibility. From satellite image analysis [Ma et al., 2019] to medical imaging [Azizi et al., 2021], computer vision models are finding a wide set of application areas. Especially since stronger models require less training data for achieving a given performance level [Azizi et al., 2021; Chen et al., 2020b; Goyal et al., 2021; Caron et al., 2020], the development of better models thanks to the larger scale pretraining will allow even more researchers and practitioners to use this powerful set of tools for their own research questions. That means that even problems that have little training data, such as fine-grained rare plant species identification [Tan et al., 2019], can start to benefit from these technologies.

Fundamental research. Humans develop a working visual system without an adult pointing to every object every second and pointing out the name of it. Despite the large differences in artificial neural networks and the biological vision systems [Sinz

1. Introduction and Overview

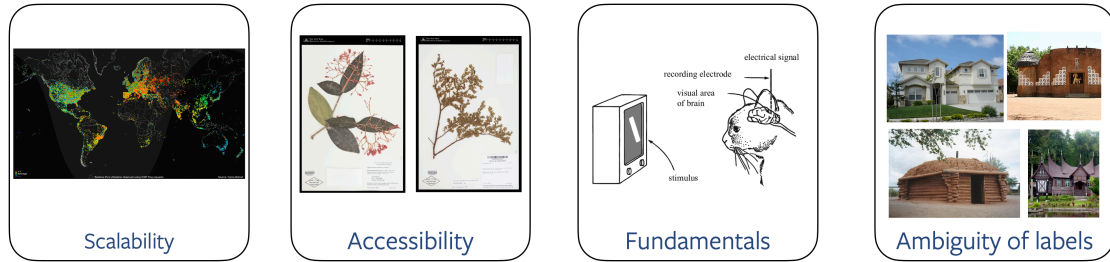


Figure 1.1: Advantages of using self-supervised learning.

[et al., 2019](#)] – for which great insights have been found by investigating cats’ neuron activations [[Hubel and Wiesel, 1962](#)] – exploring just *how far* neural networks can be optimized without labels is an important research question. While learning with labels was the starting point for the deep learning revolution, it by no means is truly a necessary component of it. This is even more the case for learning from multiple modalities, such as when presented with videos, where for example audio is additionally available.

Avoid ambiguity of labels. In the case of labeled data, not only are labels such as “house” or “boat” highly ambiguous by what they mean, they are also heavily dependent on the culture and socio-economic background of the person conducting the dataset annotations [[Paullada et al., 2020](#)]. This means using labels will necessarily incorporate some kind of human bias in the data. While this might not be crucial when the task is simply determining whether an image is a dog or a cat, it is problematic when these biases are infused into the model in the pretraining step, as pretrained models are widely used as a starting point and insidious biases can easily go unchecked, such as was the case for an face upsampling work PULSE [[Menon et al., 2020](#)]. Of course learning without labels does not eliminate bias [[Steed and Caliskan, 2021](#)] and for example geographic bias might still exist [[Shankar et al., 2017](#)].

For these reasons, deep learning methods that can train successfully without using labels are crucial for the field of computer vision, and especially representation learning, to continue its fast progress in a safe and scalable fashion. It is thus, that in this thesis we ask the question: *What is the scope and necessity of prior knowledge, data and labels in learning deep neural networks?*

1. Introduction and Overview

Such methods, that can train a neural network from a random initialisation to a state in which it is useful for other tasks – without the need of labeled data – are categorized under the umbrella of “self-supervised learning” (SSL) methods. These SSL methods work on the raw data, such as images and videos and train a model such as a CNN using a “pretext task” that can be constructed on the inputs alone and does not require knowledge of labels. One special subset of these methods are multi-modal representation learning methods. These leverage the association of different modalities within raw data that comes for free, such as the audio-stream in a video, or the speech in text form from the audio. Both single modal and multi-modal representation learning have matured rapidly and, while they are very related, in the following we provide an overview of the key motivations and related works for both topics separately, making connections where appropriate.

1.2 Key Ideas and Literature Review

1.2.1 Self-supervised Learning

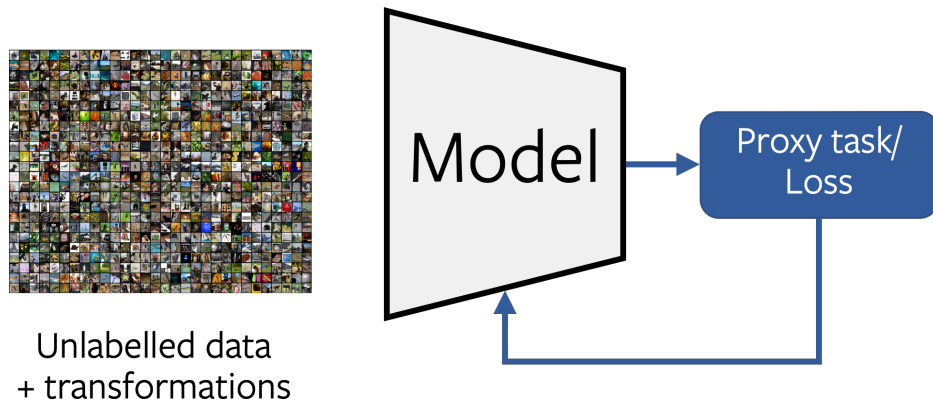


Figure 1.2: Basic framework for training with self-supervised methods.

The key idea of learning without labels is to construct a task which can provide a learning signal, despite not using annotations. This learning signal is obtained by the construction of “pretext tasks”, and therefore the performance on this task is not suited for judging if the neural network has learned meaningful representations of the visual

1. Introduction and Overview

world. Instead, performance evaluations are conducted in a standardized way on other datasets, referred to as “downstream tasks”, for which labels are used.

Performance evaluations. Generally, there are two main quantitative ways of evaluating the performance of a trained neural network [Goyal et al., 2019]: by keeping the weights of the model frozen or by further fine-tuning the weights. For the former, the trained model is frozen and typically only a weak model, such as a linear classifier¹, is trained on some labelled dataset. For image models, datasets such as ImageNet LSVRC-12 [Deng et al., 2009a] (as shown in Fig. 1.3), Places205 [Zhou et al., 2017] or iNaturalist [Van Horn et al., 2018] are used. For video datasets, evaluation is conducted on the UCF-101 [Soomro et al., 2012] and HMDB-51 [Kuehne et al., 2011] and includes also nearest neighbor based class retrieval. It is also possible to evaluate without further training a linear layer: the individual images’ representations are saved and either clustering [Sariyildiz et al., 2020; Zheltonozhskii et al., 2020] with k-means, or k-nearest neighbor (kNN) evaluations [Caron et al., 2021; Chen et al., 2021b] are performed. Both for kNN and linear evaluation, accuracy values are typically reported. For clustering evaluations, generalized correlation metrics such as the normalized mutual information (NMI) or the adjusted Rand-Index (aRI) are reported, both of which inform about the alignment between the ground-truth set of human labels and the unsupervisedly determined ones.

In the case of fine-tuning, the pretrained model is further optimized on some dataset such as Pascal VOC [Everingham et al., 2010] or COCO [Lin et al., 2014] for tasks such as object detection and segmentation, or using only a few labelled samples further trained in a “data-efficient” manner, *e.g.* using 1% of images in ImageNet [Caron et al., 2020; Chen et al., 2020b; Gidaris et al., 2021]. For videos, fine-tuning for action classification is the most common evaluation, and typically conducted on UCF-101 [Soomro et al., 2012], HMDB-51 [Kuehne et al., 2011] and Kinetics-400 [Kay et al., 2017]. For evaluating

¹The reason for utilizing a linear classifier is because it is relatively weak model (effectively a projection into the label space). Thus, a higher performance means that the features obtained from pretraining are well aligned and generalizeable to the downstream task.

1. Introduction and Overview

audio representations, datasets such as ESC-50 [Piczak, 2015] and DCASE [Stowell et al., 2015] are used. For multi-modal datasets, cross-modal retrieval is also often used for example on datasets such as MSR-VTT [Xu et al., 2016], MSVD [Venugopalan et al., 2015], VATEX [Wang et al., 2019b] and typically performance at various neighborhood levels is reported. While fine-tuning pretrained networks has been criticised [He et al., 2019] due to unclear gains in *final* performance, under common settings, such as the same number of iterations, pretraining yields significant gains.

In the following, we first provide an overview on image-based self-supervised learning methods and then describe the relevant literature on learning from multi-modal data.

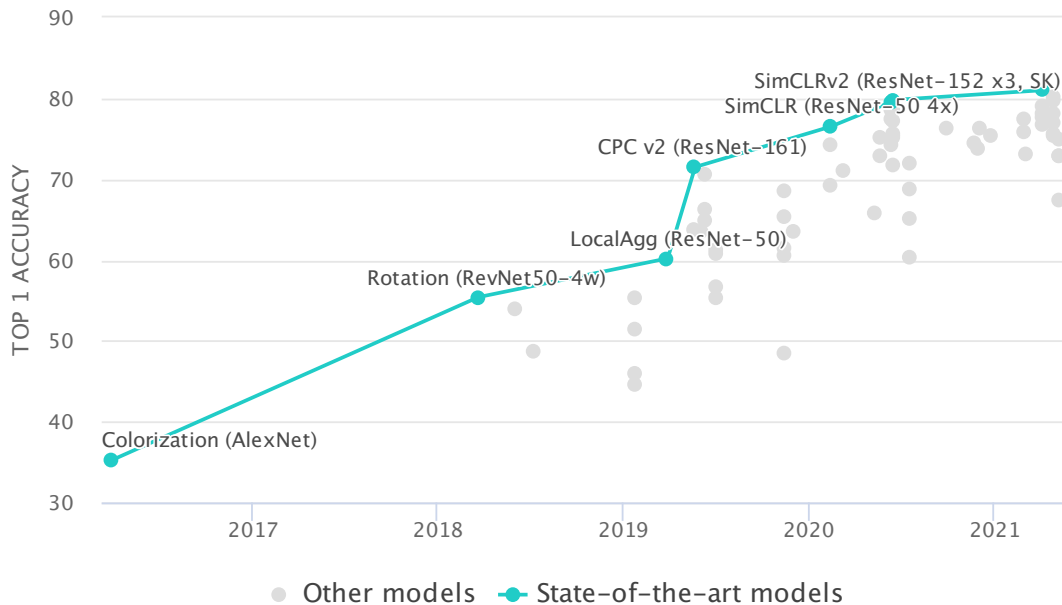


Figure 1.3: Increase in performance on a benchmark of self-supervised methods over the recent years. Even when keeping the architecture constant (such as a ResNet-50), performance has increased drastically. Plot taken from [Papers with Code, 2021].

Image-based

Context. One of the earliest self-supervised methods was developed with a pretext task of predicting the context within a single image. This was framed as classifying the relative position of two images patches [Doersch et al., 2015]. This method was further developed and framed as a task akin to letting the network solve 3×3 jigsaw

1. Introduction and Overview

puzzles of image patches [Noroozi and Favaro, 2016]. Further adjustments improved the performance, such as increasing the difficulty of the task [Kim et al., 2018] and correcting for shortcuts the network might take such as learning chromatic aberration in photographs [Mundhenk et al., 2018]. Similarly, removing a part of the image and letting the network inpaint the details has been posed as another task in [Pathak et al., 2016]. Most recently, the publication of BEiT [Bao et al., 2021] has shown that vision transformers are well-suited to pretraining supervisedly using an objective similar to masked language modelling, a common pretraining method for natural language models. Here, random patches are simply masked and the model along with a decoder is trained using a generative objective.

Transformations. In [Zhang et al., 2016; Larsson et al., 2017, 2016], a neural network is used to colorize an image from its grey-scale version. This exploits the invariance of the semantic content to color and poses a difficult task that needs to learn to extract information from a larger context. An extension to this method is developed in [Zhang et al., 2017]. Because the colorization task can only extract features from the grey-scale image, it does not learn information from colors. Hence the authors let the network predict a subset of input channels given another subset of channels (*e.g.* predicting the luminosity from the blue–yellow component and the green–red component in the ‘Lab’ color space). The method introduced in [Gidaris et al., 2018] uses image transformations to explicitly exploit an aspect of the training data: most photographs are taken upright. This photographer’s bias is used to construct a proxy task in which an image is randomly rotated by a multiple of 90° and the network is tasked with classifying the original orientation. The authors argue that for this task low-level features such as edges are not enough and that the network has to instead learn meaningful feature extractors. Another method uses transformations such as subdividing the image and rescaling it to force the network to learn equivariance of visual primitives, such as eyes [Noroozi et al., 2017]. Another take on using transformations is to force the network to predict which transformation has been applied to an image [Zhang et al., 2019; Lin et al., 2019]. Again, the authors argue that this requires high-level reasoning

1. Introduction and Overview

and this method shows how there are multiple gains still achievable from exploiting data transformations. This is indeed a similar message of our work [Asano et al., 2020b] presented in Chapter 3, where we investigate the effect of data augmentations in self-supervised learning by training on only one source image. This work is also related to studies of babies early visual inputs in which it has been found that very little diversity exists, but that instead very few objects or persons are within the field of view of the baby for long periods of time [Orhan et al., 2020]

Generative Adversarial Approaches. With the development of Generative Adversarial Networks (GANs) in [Goodfellow et al., 2014], its architecture of using two networks that improve each other’s performance has been taken to self-supervised learning first in [Radford et al., 2015]. Soon, this architecture was improved for feature representation learning by including an additional encoder in this setup (yielding three networks to be trained) in the Bi-directional GAN [Donahue et al., 2017; Dumoulin et al., 2016]. This has been further scaled to achieve stronger performances using ResNet architectures alongside more powerful generative networks [Donahue and Simonyan, 2019]. Another use of GANs is in [Jenni and Favaro, 2018], where the network is trained to spot high-level artifacts in images generated by another neural network.

Noise-Contrastive. While contrastive learning has been first introduced in [Hadsell et al., 2006; Chopra et al., 2005], one of the first usages in self-supervised representation learning is [Wu et al., 2018a]. This work, where a pretext task of discriminating each image instance from other images is first introduced, has also been motivated by the work of Dosovitskiy et al. [2016], in which pseudolabels are created by applying a large set of transformations to each image and the network is trained to ignore these transformations. This work has also recently been further scaled in [Liu et al., 2021] to 1 Million classes. Noise-contrastive learning has quickly taken center stage in self-supervised learning with the publication of CPC [Hénaff et al., 2019; Oord et al., 2018] in which the task is framed as an auto-regressive task, [Misra and van der Maaten, 2020] where contrastive learning is combined with further pretext task invariances,

1. Introduction and Overview

and MoCo [He et al., 2020], in which a slow, exponential-moving average network is used to replace the memory bank in [Wu et al., 2018a]. Following this, works such as SimCLR [Chen et al., 2020a], CMC [Tian et al., 2020a], MoCo-v2 [Chen et al., 2020c], SimCLR-v2 [Chen et al., 2020b] have further pushed the state-of-the-art in this field and also sparked multiple improvements upon this basic structure: Works such as [Kalantidis et al., 2020] introduce learning from synthetic hard negatives while [Hu et al., 2020] construct hard negatives synthetically; [Xie et al., 2021b] bring the contrastive loss to the localised feature-map; [Cai et al., 2020] generate multiple views to better approximate invariances; [Li et al., 2021] combines contrastive learning with clustering and [Ayush et al., 2020] does this with geo-location discrimination; [Shen et al., 2020] introduces the usage of augmentations such as cut-mix and mix-up to MoCo-style methods; [Zhu et al., 2020] shows how noise-contrastive learning can be conducted even with as little as 16 negatives by adding a margin parameter and changing the temperature and learning rate; and [Wei et al., 2021] add an additional consistency loss to alleviate the impact of false negatives. Building on SimCLR, [Dwivedi et al., 2021] further use nearest neighbors in embeddings space sampled from a queue, to enforce better more class-oriented invariances.

There are also multiple works that analyze the workings of contrastive learning [Wang and Isola, 2020], the role of the different views [Tian et al., 2020b], as well as comparative studies of the different self-supervised methods [Xiao et al., 2021; Gansbeke et al., 2021] and their pretraining datasets [Kotar et al., 2021].

Non-contrastive agreement between views. Building on the key insight from contrastive learning that augmentation invariance provides a strong supervisory signal, BYOL [Grill et al., 2020] first introduced the notion of learning solely by agreement between views. Works such as [Zbontar et al., 2021; Bardes et al., 2021; Ermolov et al., 2020] similarly learn this invariance by explicitly enforcing feature-level cross-correlations to be low and [Chen and He, 2021] simplifies learning from two shared encoders by introducing simple stop-gradients instead of momentum updates to keep

1. Introduction and Overview

training stable. Building on BYOL, [Koohpayegani et al., 2021] leverages using multiple nearest neighbors in embedding space, effectively implicitly clustering the representations in an online fashion.

Clustering. In [Bojanowski and Joulin, 2017], a neural network is trained to learn an embedding on a unit-sphere that optimizes random clusters. In [Caron et al., 2018], a simple alternating procedure was developed to train a model by switching between training a model on pseudolabels and assigning pseudolabels based on k-means on the features of the whole data set. By incorporating transformations, this allows the network to become invariant to them and works by extracting and amplifying the initial feature extractor (a randomly initialized neural network). An online version of this was proposed in [Zhan et al., 2020]. We build on this work by noticing that the k-means objective and the network training objective (cross-entropy) are not equal and develop a novel loss that is shared between clustering and representation learning in [Asano et al., 2020c]. This loss is efficiently optimized using optimal-transport [Cuturi, 2013] for obtaining highly semantic clusters. This Sinkhorn-Knopp optimal transport based clustering is also used in one of the state-of-the-art pretraining methods SwAV [Caron et al., 2020], in which online clustering is done and the network is tasked with predicting cluster assignments across different augmentations. More recently, clustering on the localised feature-maps has been proposed in [Gidaris et al., 2020] and further improved to work in an online fashion in [Gidaris et al., 2021].

Combinations. There are few papers that show that combining multiple tasks yields superior performance. In [Doersch and Zisserman, 2017], colorization [Larsson et al., 2017], relative position [Doersch et al., 2015], exemplar [Dosovitskiy et al., 2016] and motion segmentation [Pathak et al., 2016] is combined. In [Kim et al., 2018], jigsaw [Noroozi and Favaro, 2016] is combined with colorization [Larsson et al., 2017]. In [Feng et al., 2019], rotation prediction [Gidaris et al., 2018] is combined with instance retrieval [Wu et al., 2018b]. There are also works which combine contrastive or cross-view agreement with some clustering notion [Xie et al., 2021a; Li et al., 2021].

1. Introduction and Overview

Video-based. Some of the tasks that use the space dimension of images have been extended to the space-time dimensions of videos by crafting equivalent tasks. These include jigsaw puzzles [Kim et al., 2019], and predicting rotations [Jing and Tian, 2018] or future frames [Han et al., 2019]. Other tasks leverage the temporal dimension of videos to learn representations by predicting shuffled frames [Misra et al., 2016], the direction of time [Wei et al., 2018], motion [Wang et al., 2019a], clip and sequence order [Xu et al., 2019; Lee et al., 2017], and playback speed [Benaim et al., 2020; Cho et al., 2020; Fernando et al., 2017]. More recently, works that leverage noise-contrastive learning have also been extended to the video domain [Han et al., 2019, 2020b,a]. Besides the raw RGB videos, video data often contains multiple associated modalities such as audio, visual flow (computed from raw videos), or text captions obtained from the audio. And indeed, it has been consistently found that models that leverage multiple modalities can be leveraged to further improve visual representation learning. In the following, we provide an overview on multi-modal self-supervised learning frameworks.

1.2.2 Multi-modal Learning

Learning from multiple modalities not only contains more raw information and provides alternate views for more robust learning, but can also be motivated from human learning. Humans indeed learn by leveraging the redundancy of the different sensory experiences [Barlow, 1989; Edelman, 1987]. Since humans experience multiple signals coming from the vision, sound, movement and touch, leveraging these signals together allows for more robust processing [Edelman, 1987] as well as for learning from co-occurrences across modalities [de Sa, 1994].

Videos, unlike images, are a rich source of a variety of modalities such as speech, audio, and optical flow, and their correlation can be used as a supervisory signal. While this idea has been proposed even before deep learning has become mainstream [de Sa, 1994], only recently, it has been used to successfully learn effective representations. We start with methods that learn from audio-visual data. Works that plainly leverage the natural correspondence [Owens et al., 2016; Aytar et al., 2016; Arandjelović and

1. Introduction and Overview

Zisserman, 2017; Alwassel et al., 2020a] for learning are among the first in this domain. Building on this, audio-visual synchronization [Chung and Zisserman, 2016; Korbar et al., 2018; Owens and Efros, 2018] has further driven the development of methods that learn to leverage the information contained in aligned video-audio clips. In a similar fashion, cross-modal contrastive methods such as CMC [Tian et al., 2020a] and GDT [Patrick et al., 2020], presented in Chapter 7, can leverage this signal but in addition can learn arbitrary invariances and variances introduced by transformations. Similar to GDT, but utilizing the idea of using better contrastive pairs are [Morgado et al., 2020, 2021]. In Chapter 8, we provide orthogonal contributions by tailoring the architecture and augmentations to videos in a noise-contrastive audio-visual setting and achieving state-of-the-art performances [Patrick et al., 2021a]. Work that extends BYOL and MoCo to the video domain include [Recasens et al., 2021], which uses two different temporal horizons for computing views, as well as [Feichtenhofer et al., 2021] which show that augmentations for video BYOL and MoCo play a crucial role in obtaining high performances. Works that use clustering for learning representations include XDC [Alwassel et al., 2020b], which generalized DeepCluster [Caron et al., 2018] to the audio-visual domain by predicting cluster assignment across modalities; as well as [Piergiovanni et al., 2020], which uses clustering for meta-learning a loss function; and [Asano et al., 2020a], which we present in Chapter 5 and which provides a generalization of [Asano et al., 2020c] to the non-uniform, and multi-modal case.

Moving beyond audio-visual data, video-text data has also been popular for training representations, especially since the release of the HowTo100M dataset [Miech et al., 2019]. This dataset of 100M clips from 1.2M videos is a large-scale instructional video dataset, and has allowed many papers to leverage speech as a weak supervisory signal to train video representations. Research on learning video-text representations for the specific purpose of cross-modal retrieval, has applications such as search engines (text-to-video) or captioning (video-to-text). In [Sun et al., 2019a; Miech et al., 2020; Rouditchenko et al., 2020; Li and Wang, 2020; Bain et al., 2021], training on video-text datasets is done by training cross-modally using contrastive learning. Models such as [Sun et al., 2019b; Nagrani et al., 2020; Luo et al., 2020; Zhu and Yang, 2020;

1. Introduction and Overview

[Gabeur et al., 2020](#)] instead jointly model visual and textual inputs in a combined model to model interactions. While all these approaches rely on a contrastive objective, VidTranslate [\[Korbar et al., 2020\]](#) shows that a generative objective can also be used to learn joint video-text representations. In contrast to [\[Korbar et al., 2020\]](#), in [\[Patrick et al., 2021b\]](#) presented in Chapter 6, we show that combining contrastive *and* generative objectives to pretrain video-text representations on large-scale data such as HowTo100M is very effective. The generative objective thus mitigates the strictness of the instance discrimination task of the contrastive objective, and thus yields similar benefits to works that change the raw noise-contrastive objective introduced above.

Most recently, there are also works which go beyond learning from just two modalities, either via distillation [\[Piergiovanni et al., 2020\]](#), or via explicit modelling of multiple embedding spaces [\[Alayrac et al., 2020; Akbari et al., 2021\]](#).

Finally, multi-modal learning has also been leveraged for applications. These include lip reading [\[Chung and Zisserman, 2016; Chung et al., 2017; Afouras et al., 2018c\]](#), audio-visual source separation and localization [\[Arandjelović and Zisserman, 2018; Zhao et al., 2018; Afouras et al., 2018b; Zhao et al., 2019; Harwath et al., 2018; Hu et al., 2019\]](#), speech recognition [\[Potamianos et al., 2003; Afouras et al., 2018a\]](#), efficient inference [\[Gao et al., 2020; Korbar et al., 2019\]](#), egocentric action recognition [\[Kazakos et al., 2019\]](#) and navigation [\[Chen et al., 2019\]](#). Most recently, multi-modal self-supervised learning has been leveraged for learning a generic object detector for images in [\[Afouras et al., 2021\]](#) and presented in Chapter 9.

1.3 Publication List

Chapters 3 to 10 present papers accepted (Chapters 3 to 6), or currently under review (Chapters 7 to 10) at major conferences and are left in their original or preprint formatting. I am a first/co-first author in each of these papers and statements of authorship are provided in Appendix Chapter 10. The publications list is as follows:

1. Introduction and Overview

1. Chapter 3: *A critical analysis of self-supervision, or what we can learn from a single image*, presented at ICLR 2020.
2. Chapter 4: *Self-labelling via simultaneous clustering and representation learning*, presented at ICLR 2020 as a spotlight.
3. Chapter 5: *Labelling unlabelled videos from scratch with multi-modal self-supervision*, presented at NeurIPS 2020.
4. Chapter 6: *Support-set bottlenecks for video-text representation learning*, presented at ICLR 2021 as a spotlight.
5. Chapter 7: *Multi-modal Self-Supervision from Generalized Data Transformations*, presented at ICCV 2021.
6. Chapter 8: *Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning*, presented at ICCV 2021.
7. Chapter 9: *Self-supervised object detection from audio-visual correspondence*, arXiv preprint.
8. Chapter 10: *PASS: An ImageNet replacement for self-supervised pretraining without humans*, presented at NeurIPS Datasets 2021.

In addition, I have written these following works during the time of my D.Phil but exclude them due to their lower relevance to the main topic of this thesis (underlined my name):

1. *Emergent inequality and endogenous dynamics in a simple behavioral macroeconomic model*.

YM. Asano, J. Kolb, J. Heitzig, JD. Farmer.

Proceedings of the National Academy of Sciences (PNAS). 2021.

1. *Introduction and Overview*

2. *Memes in the Wild: Assessing the Generalizability of the Hateful Memes Challenge Dataset.*

H. Kirk, Y. Jun, P. Rauba, G. Wachtel, R. Li, X. Bai, N. Broestl, M. Doff-Sotta, A. Shtedritski, YM. Asano.

ACL 2021 Workshop on Online Abuse and Harms.

3. Privacy-preserving Object Detection.

P. He, C. Griffin, K Kacprzyk, A. Joosen, M. Collyer, A. Shtedritski, YM. Asano
ICLR 2021 Synthetic Data Generation Workshop.

4. *How True is GPT-2? An Empirical Analysis of Intersectional Occupational Biases.*

H. Kirk, Y. Jun, H. Iqbal, E. Benussi, F. Volpin, F. Dreyer, A. Shtedritski, YM. Asano.

arXiv 2102.04130 [under review]

5. *Rising adoption and retention of meat-free diets in online recipe data.*

YM. Asano, G. Biermann.

Nature Sustainability. 2019.

2

Conclusion

In this thesis, we address the problem of how we can learn the weights of deep neural networks from raw data, and *how far* leveraging transformations, as well as the multi-modal nature of signals as prior knowledge can take us. In the following, we present summaries and research impact of the included papers and current open questions.

2.1 Summaries and Impact

As shown in Fig. 1.2, self-supervised learning methods contain two parts: a proxy task, and data transformations. While many papers propose innovations in the former, few study the effect and role on the latter.

A critical analysis of self-supervision, or what we can learn from a single image (Chapter 3) [published at ICLR 2020] is an investigation into the role of image augmentations in self-supervised learning. In order to isolate the effect of the image augmentations, we investigate three, previously state of the art, self-supervised methods with regards to their performance in extreme scenarios such as using only one image as a pretraining dataset. Surprisingly, we find that all three methods can learn meaningful representations and their early layers perform on par with those trained on ImageNet.

2. Conclusion

While for deeper layers, a performance gap arises, the gain of switching from zero images (random initialization) to one image remains large. This analysis highlighted that self-supervised learning methods’ essential ingredient is not necessarily massive amounts of data, but the usage of data augmentation strategies. This can be understood as infusing some prior knowledge (such as invariance against image noise or rotations) into the network, and motivated further research into augmentations.

As such, subsequent works have continuously improved the augmentations used for self-supervised learning, such as SimCLR [Chen et al., 2020a], by introduced Gaussian Blurring, CPC [Hénaff et al., 2019], by the usage of AutoAugment [Cubuk et al., 2019], SwAV [Caron et al., 2020], by introducing MultiCrop and BYOL [Grill et al., 2020], by introducing the usage of Solarization. As the first work to self-supervisedly train on images lacking actual photographic datasets, this work has inspired investigations into learning from fractal images [Kataoka et al., 2021] and learning solely from GAN-generated or structured noise images [Baradad et al., 2021].

The next work, **Self-labelling via simultaneous clustering and representation learning** (Chapter 4) [spotlight at ICLR 2020] develops a novel method for training a neural network in a self-supervised manner using clustering that outperforms the previous methods such as the ones investigated in the previous paper. By building the clustering loss from two first principles of a) using the same loss function as supervised training (cross-entropy) and b) requiring clusters to be of the same size, this paper illuminates how clustering can be solved using optimal transport. This can substantially improve performance due to using a single loss function for both clustering and representation learning, and therefore does not suffer from alternating objectives. This method yields state-of-the-art results on downstream tasks and, as a by-product of representation learning, creates a set of pseudo-labels for a dataset. These pseudo-labels, in turn are highly semantic and are close to the ground-truth human annotations. This work has found numerous applications, including clustering of video datasets (Chapter 5, Chapter 9), as well as being the core component in a current state-of-the-art self-supervised learning method, SwAV [Caron et al., 2020].

2. Conclusion

In addition, the labels created by this method have been investigated with regards to human interpretability [Laina et al., 2020].

While the previous paper has shown that self-supervised clustering of images can be solved efficiently, in the case of multi-modal data, such as video-audio this is non-trivial, as clustering across modalities yields two sets of disparate clusters, while concatenating allows the stronger modality to simply dominate.

Labelling unlabelled videos from scratch with multi-modal self-supervision

[published at NeurIPS'20] solves this problem and proposes a general self-supervised clustering method for multi-modal data. The proposed method solves this issue of multiple modalities by instead viewing each one as an augmentation (or “view”) of a common, implicit semantic category and we obtain consistent clustering of video datasets that group videos of similar content successfully together. For clustering, we build on the SeLa framework (Chapter 4) from above and further extend it: Firstly, as multi-modal learning requires two sets of networks (in this case: one for audio, one for video), this makes clustering both jointly non-trivial. Second, we develop a method that allows for clustering into arbitrary distributions, such as Gaussian or Zipf distributions, better suited to video data in the wild. Finally, we provide the first benchmark in the domain of clustering videos without any supervision and find that, contrary to the prevailing dogma, strong representation learning performance does not imply strong clustering performance, e.g. a strong MIL-NCE pretrained encoder’s clusters generated from k-means are inferior to the clusters obtained using this method.

This method forms the base clustering module in our self-supervised object detection method of Chapter 9 and has inspired more clustering works in the video domain [Xu et al., 2021] as well as works that utilize additional modalities or information [Chen et al., 2021a; Brown et al., 2021]. While this paper shows that clustering multi-modal data needs to be researched on its own, learning strong video and audio representations is still highly useful for a number of other tasks, such as human action prediction and sound classification. The following three works are contributions to this topic in the video-audio and video-text domain.

2. Conclusion

Multi-modal Self-Supervision from Generalized Data Transformations [published at ICCV'21] introduces a novel framework for multi-modal representation learning. In particular, we focus on noise-contrastive learning for the audio-visual domain and show how various older self-supervised tasks can be posed as simply learning variances (as opposed to invariances) to special video transformations, such as shifting the timing between the audio and the video. Using this insight, we find novel combinations that yield state-of-the-art performance in video and audio representation learning benchmarks.

This work is also being actively investigated (and potentially used) as a recommendation backbone for Instagram's Reels, as described in these public blogposts [[Facebook AI Research](#), 2020, 2021].

Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning [published at ICCV'21] pushes the state-of-the-art in video representation learning further. In this work, we propose to better utilise both the (latent) time and space dimensions. While the usage of multiple crops in the spatial domain has been shown to be highly useful in the image domain for learning strong representations [[Caron et al., 2020](#)], a similar approach for videos, due to their time-dimension is computationally prohibitive. In this work we thus first propose a novel scheme for cropping in space and time in the embedding space after the last convolutional layer. This makes the generation of augmentations computationally much more efficient, while still yielding gains of similar magnitude. Motivated by this better usage of the latent spatial dimension, we also propose the usage of a shallow transformer for aggregating the latent time dimension of the representations. This allows for gains in performances and usage of longer clips, that together achieve state-of-the-art performances in multiple sets of downstream tasks.

Support-set bottlenecks for video-text representation learning [spotlight at ICLR'21], is a novel method for learning video-text representations for the purpose of video-to-text and text-to-video retrieval. This work extends the insights gained from recent contrastive self-supervised learning, such as MoCo [[He et al., 2020](#)] or

2. Conclusion

SimCLR [Chen et al., 2020a] to learn stronger video-text encoders. Motivated by the fact that vanilla noise-contrastive learning treats all other samples as different classes and therefore will necessarily include many false negatives, we investigate how we can encourage the network to learn common concepts shared *between* the samples. We find that adding a generative captioning task that forces the learning of commonalities between videos, we achieve significant gains over the vanilla baseline and obtain state-of-the-art video-to-audio and text-to-video retrieval performance for common video-text benchmarking datasets. Due to the task’s generative nature, the text-output also allows for investigating how the model is “seeing” data and yields better interpretability than plain contrastive methods. While the above three works have focused on multi-modal representation learning, the next work shows that combining various parts of self-supervised learning can even allow us to solve tasks previously reserved for with supervised domain. One such task is object detection, one of the hallmark tasks in the supervised image domain.

Self-supervised object detection from audio-visual correspondence [under review], combines noise-contrastive and clustering based self-supervised learning to detect objects in images. Crucially, and in stark difference to previous works such as [Hu et al., 2019; Owens et al., 2016], our method generates bounding boxes alongside class-predictions instead of solely outputting heatmaps. Furthermore, our methods allows for training an off-the-shelf detector such as faster-RCNN, our method can also conduct inference on input data without associated sound, i.e. detect objects on normal images. This method thus shows how important knowledge obtained from audio-visual associations can be distilled into image models and leveraged for image-based object detection.

PASS: An ImageNet replacement for self-supervised pretraining without humans [published at NeurIPS Datasets’21], is a return to the fundamental image-based representation learning domain with a focus on the training data and its ethical dimension. More specifically, we focus on generating a pretraining dataset that respect privacy regulations such as General Data Protection Regulation (GDPR) and reduces

2. Conclusion

biases associated to human depictions. In this paper we propose a novel dataset named “PASS”, which contains random images without humans, or human parts. Firstly, this dataset allows for pretraining representations without infringing on the privacy of people while simultaneously reducing the extent of biases present in images, as it does not contain people and further does not utilize a search engine (which can yield stereotypical and skewed results [Hanna et al., 2020]) in its creation. Importantly, we show that self-supervised representations trained on this dataset perform on-par with those pretrained on ImageNet on a large set of downstream tasks. Lastly, being random and therefore not object-oriented or class-balanced like ImageNet, this dataset approximates the use case of self-supervised learning methods much clearer and should aid in developing novel methods that scale better.

2.2 Open Challenges

Since 2018, self-supervised learning has matured rapidly, yet, it still faces two major challenges to become a truly foundational pillar for deep learning.

The first is the lack of self-supervised learning methods that can leverage learning different modalities (such as images, audio and text) *from* different datasets, as most datasets do not contain more than two modalities. Without this, the space of potential learning signals will stay limited and self-supervised learning progress potentially stall. At present, there are only three works ([Alayrac et al., 2020; Akbari et al., 2021; Piergiovanni et al., 2020]) that learn from more than three modalities, but even then require those modalities to be included in a single dataset. To address this challenge, the effective scale at which self-supervised learning can operate needs to be increased: being able to learn from multiple kinds of inputs such as images, video, audio, and text *simultaneously* will unlock new capabilities and datasets and yield a new paradigm of learning from any data – a long-term goal of self-supervised learning as a field. Combining multiple modalities with the perception of time (such as is

2. Conclusion

present in video) is also an attractive goal as it is more closely aligned to how humans learn and understand concepts.

The second challenge is the narrow scope of current self-supervised learning methods, focusing mostly on image or video *classification*. Thus novel, self-supervised methods have lots of space to catch up with their supervised counterparts and move beyond representation learning. So far, these methods have only been applied in a limited fashion: tasks such as audio-visual synchronization [Korbar et al., 2018; Owens and Efros, 2018], tracking [Lai et al., 2020], clustering [Alwassel et al., 2020b; Asano et al., 2020a; Chen et al., 2021a; Brown et al., 2021] and most recently object detection [Afouras et al., 2021] are only a small subset of what can be learned without labels and will surely keep expanding and improving. Finally, it is important to note that the development of multi-modal methods requires substantial amounts of compute requirements, which can be limiting for researchers not in industry. For continued progress, it is therefore important to keep this field as open as possible (e.g. by keeping default training sets small) to prevent the field from splitting and methods from being developed twice.

3

A critical analysis of self-supervision, or
what we can learn from a single image

**This work was presented at the International Conference on
Learning Representations (ICLR) 2020.**

A CRITICAL ANALYSIS OF SELF-SUPERVISION, OR WHAT WE CAN LEARN FROM A SINGLE IMAGE

Yuki M. Asano

Christian Rupprecht

Andrea Vedaldi

Visual Geometry Group

University of Oxford

{yuki, chrisr, vedaldi}@robots.ox.ac.uk

ABSTRACT

We look critically at popular self-supervision techniques for learning deep convolutional neural networks without manual labels. We show that three different and representative methods, BiGAN, RotNet and DeepCluster, can learn the first few layers of a convolutional network *from a single image* as well as using millions of images and manual labels, provided that strong data augmentation is used. However, for deeper layers the gap with manual supervision cannot be closed even if millions of unlabelled images are used for training. We conclude that: (1) the weights of the early layers of deep networks contain limited information about the statistics of natural images, that (2) such low-level statistics can be learned through self-supervision just as well as through strong supervision, and that (3) the low-level statistics can be captured via synthetic transformations instead of using a large image dataset.

1 INTRODUCTION

Despite tremendous progress in supervised learning, learning without external supervision remains difficult. Self-supervision has recently emerged as one of the most promising approaches to address this limitation. Self-supervision builds on the fact that convolutional neural networks (CNNs) transfer well between tasks (Shin et al., 2016; Oquab et al., 2014; Girshick, 2015; Huh et al., 2016). The idea then is to pre-train networks via *pretext tasks* that do not require expensive manual annotations and can be automatically generated from the data itself. Once pre-trained, networks can be applied to a target task by using only a modest amount of labelled data.

Early successes in self-supervision have encouraged authors to develop a large variety of pretext tasks, from colorization to rotation estimation and image autoencoding. Recent papers have shown performance competitive with supervised learning by learning complex neural networks on very large image datasets. Nevertheless, for a given model complexity, pre-training by using an off-the-shelf annotated image datasets such as ImageNet remains much more efficient.

In this paper, we aim to *investigate the effectiveness of current self-supervised approaches* by characterizing how much information they can extract from a given dataset of images. Since deep networks learn a *hierarchy* of representations, we further break down this investigation on a per-layer basis. We are motivated by the fact that the first few layers of most networks extract low-level information (Yosinski et al., 2014), and thus learning them may not require the high-level semantic information captured by manual labels.

Concretely, in this paper we answer the following simple question: “*is self-supervision able to exploit the information contained in a large number of images in order to learn different parts of a neural network?*”

We contribute two key findings. First, we show that *as little as a single image* is sufficient, when combined with self-supervision and data augmentation, to learn the first few layers of standard deep networks as well as using millions of images and full supervision (Figure 1). Hence, while self-supervised learning works well for these layers, this may be due more to the limited complexity of such features than the strength of the supervisory technique. This also confirms the intuition that early layers in a convolutional network amounts to low-level feature extractors, analogous to early

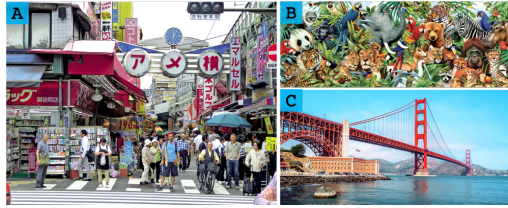
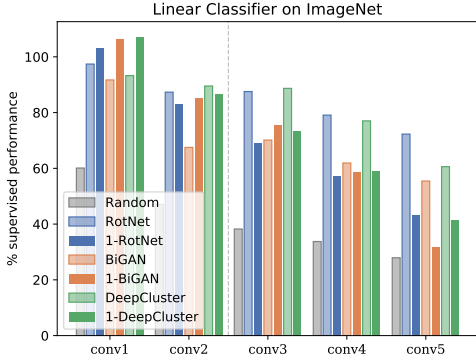


Figure 1: **Single-image self-supervision.** We show that several self-supervision methods can be used to train the first few layers of a deep neural networks using a *single training image*, such as this Image A, B or even C (above), provided that sufficient data augmentation is used.

learned and hand-crafted features for visual recognition (Olshausen & Field, 1997; Lowe, 2004; Dalal & Triggs, 2005). Finally, it demonstrates the importance of image transformations in learning such low-level features as opposed to image diversity.¹

Our second finding is about the deeper layers of the network. For these, self-supervision remains inferior to strong supervision even if millions of images are used for training. Our finding is that this is unlikely to change with the addition of more data. In particular, we show that training these layers with self-supervision and a single image already achieves as much as two thirds of the performance that can be achieved by using a million different images.

We show that these conclusions hold true for three different self-supervised methods, BiGAN (Donahue et al., 2017), RotNet (Gidaris et al., 2018) and DeepCluster (Caron et al., 2018), which are representative of the spectrum of techniques that are currently popular. We find that performance as a function of the amount of data is dependent on the method, but all three methods can indeed leverage a single image to learn the first few layers of a deep network almost “perfectly”.

Overall, while our results *do not* improve self-supervision *per-se*, they help to characterize the *limitations of current methods* and to better focus on the important open challenges.

2 RELATED WORK

Our paper relates to three broad areas of research: (a) self-supervised/unsupervised learning, (b) learning from a single sample, and (c) designing/learning low-level feature extractors. We discuss closely related work for each.

Self-supervised learning: A wide variety of proxy tasks, requiring no manual annotations, have been proposed for the self-training of deep convolutional neural networks. These methods use various cues and tasks namely, in-painting (Pathak et al., 2016), patch context and jigsaw puzzles (Doersch et al., 2015; Noroozi & Favaro, 2016; Noroozi et al., 2018; Mundhenk et al., 2017), clustering (Caron et al., 2018), noise-as-targets (Bojanowski & Joulin, 2017), colorization (Zhang et al., 2016; Larsson et al., 2017), generation (Jenni & Favaro, 2018; Ren & Lee, 2018; Donahue et al., 2017), geometry (Dosovitskiy et al., 2016; Gidaris et al., 2018) and counting (Noroozi et al., 2017). The idea is that the pretext task can be constructed automatically and easily on images alone. Thus, methods often modify information in the images and require the network to recover them. In-painting or colorization techniques fall in this category. However these methods have the downside that the features are learned on modified images which potentially harms the generalization to unmodified ones. For example, colorization uses a gray scale image as input, thus the network cannot learn to extract color information, which can be important for other tasks.

Slightly less related are methods that use additional information to learn features. Here, often temporal information is used in the form of videos. Typical pretext tasks are based on temporal-context (Misra et al., 2016; Wei et al., 2018; Lee et al., 2017; Sermanet et al., 2018), spatio-temporal

¹Example applications that only rely on low-level feature extractors include template matching (Kat et al., 2018; Talmi et al., 2017) and style transfer (Gatys et al., 2016; Johnson et al., 2016), which currently rely on pre-training with millions of images.

cues (Isola et al., 2015; Gao et al., 2016; Wang et al., 2017), foreground-background segmentation via video segmentation (Pathak et al., 2017), optical-flow (Gan et al., 2018; Mahendran et al., 2018), future-frame synthesis (Srivastava et al., 2015), audio prediction from video (de Sa, 1994; Owens et al., 2016), audio-video alignment (Arandjelović & Zisserman, 2017), ego-motion estimation (Jayaraman & Grauman, 2015), slow feature analysis with higher order temporal coherence (Jayaraman & Grauman, 2016), transformation between frames (Agrawal et al., 2015) and patch tracking in videos (Wang & Gupta, 2015). Since we are interested in learning features from as little data as one image, we cannot make use of methods that rely on video input.

Our contribution inspects three unsupervised feature learning methods that use very different means of extracting information from the data: BiGAN (Donahue et al., 2017) utilizes a generative adversarial task, RotNet (Gidaris et al., 2018) exploits the photographic bias in the dataset and DeepCluster (Caron et al., 2018) learns stable feature representations under a number of image transformations by proxy labels obtained from clustering. These are described in more detail in the Methods section.

Learning from a single sample: In some applications of computer vision, the bold idea of learning from a single sample comes out of necessity. For general object tracking, methods such as max margin correlation filters (Rodriguez et al., 2013) learn robust tracking templates from a single sample of the patch. A single image can also be used to learn and interpolate multi-scale textures with a GAN framework (Rott Shaham et al., 2019). Single sample learning was pursued by the semi-parametric exemplar SVM model (Malisiewicz et al., 2011). They learn one SVM per positive sample separating it from all negative patches mined from the background. While only one sample is used for the positive set, the negative set consists of thousands of images and is a necessary component of their method. The negative space was approximated by a multi-dimensional Gaussian by the Exemplar LDA (Hariharan et al., 2012). These SVMs, one per positive sample, are pooled together using a max aggregation. We differ from both of these approaches in that we do not use a large collection of negative images to train our model. Instead we restrict ourselves to a single or a few images with a systematic augmentation strategy.

Classical learned and hand-crafted low-level feature extractors: Learning and hand-crafting features pre-dates modern deep learning approaches and self-supervision techniques. For example the classical work of (Olshausen & Field, 1997) shows that edge-like filters can be learned via sparse coding of just 10 natural scene images. SIFT (Lowe, 2004) and HOG (Dalal & Triggs, 2005) have been used extensively before the advent of convolutional neural networks and, in many ways, they resemble the first layers of these networks. The scatter transform of Bruna & Mallat (2013); Oyallon et al. (2017) is an handcrafted design that aims at replacing at least the first few layers of a deep network. While these results show that effective low-level features can be handcrafted, this is insufficient to clarify the power and limitation of self-supervision in deep networks. For instance, it is not obvious whether deep networks can learn better low level features than these, how many images may be required to learn them, and how effective self-supervision may be in doing so. For instance, as we also show in the experiments, replacing low-level layers in a convolutional networks with handcrafted features such as Oyallon et al. (2017) may still decrease the overall performance of the model. Furthermore, this says little about deeper layers, which we also investigate.

In this work we show that current deep learning methods learn slightly better low-level representations than hand crafted features such as the scattering transform. Additionally, these representations can be learned from one single image with augmentations and without supervision. The results show how current self-supervised learning approaches that use one million images yield only relatively small gains when compared to what can be achieved from one image and augmentations, and motivates a renewed focus on augmentations and incorporating prior knowledge into feature extractors.

3 METHODS

We discuss first our data and data augmentation strategy (section 3.1) and then we summarize the three different methods for unsupervised feature learning used in the experiments (section 3.2).

3.1 DATA

Our goal is to understand the performance of representation learning methods as a function of the image data used to train them. To make comparisons as fair as possible, we develop a protocol where only the nature of the training data is changed, but all other parameters remain fixed.

In order to do so, given a baseline method trained on d source images, we replace those with another set of d images. Of these, now only $N \ll d$ are *source* images (i.e. i.i.d. samples), while the remaining $d - N$ are augmentations of the source ones. Thus, the amount of information in the training data is controlled by N and we can generate a continuum of datasets that vary from one extreme, utilizing a single source image $N = 1$, to the other extreme, using all $N = d$ original training set images. For example, if the baseline method is trained on ImageNet, then $d = 1,281,167$. When $N = 1$, it means that we train the method using a single source image and generate the remaining 1,281,166 images via augmentation. Other baselines use CIFAR-10/100 images, so in those cases $d = 50,000$ instead.

The data augmentation protocol, is an extreme version of augmentations already employed by most deep learning protocols. Each method we test, in fact, already performs some data augmentation internally. Thus, when the method is applied on our augmented data, this can be equivalently thought of as incrementing these “native” augmentations by concatenating them with our own.

Choice of augmentations. Next, we describe how the N source images are expanded to additional $d - N$ images so that the models can be trained on exactly d images, independent from the choice of N . The idea is to use an aggressive form of data augmentation involving cropping, scaling, rotation, contrast changes, and adding noise. These transformations are representative of invariances that one may wish to incorporate in the features. Augmentation can be seen as imposing a prior on how we expect the manifold of natural images to look like. When training with very few images, these priors become more important since the model cannot extract them directly from data.

Given a source image of size $H \times W$, we first extract a certain number of random patches of size (w, h) , where $w \leq W$ and $h \leq H$ satisfy the additional constraints $\beta \leq \frac{wh}{WH}$ and $\gamma \leq \frac{h}{w} \leq \gamma^{-1}$. Thus, the smallest size of the crops is limited to be at least βWH and at most the whole image. Additionally, changes to the aspect ratio are limited by γ . In practice we use $\beta = 10^{-3}$ and $\gamma = \frac{3}{4}$.

Second, good features should not change much by small image rotations, so images are rotated (before cropping to avoid border artifacts) by $\alpha \in (-35, 35)$ degrees. Due to symmetry in image statistics, images are also flipped left-to-right with 50% probability.

Illumination changes are common in natural images, we thus expect image features to be robust to color and contrast changes. Thus, we employ a set of linear transformations in RGB space to model this variability in real data. Additionally, the color/intensity of single pixels should not affect the feature representation, as this does not change the contents of the image. To this end, color jitter with additive brightness, contrast and saturation are sampled from three uniform distributions in $(0.6, 1.4)$ and hue noise from $(-0.1, 0.1)$ is applied to the image patches. Finally, the cropped and transformed patches are scaled to the color range $(-1, 1)$ and then rescaled to full $S \times S$ resolution to be supplied to each representation learning method, using bilinear interpolation. This formulation ensures that the patches are created in the target resolution S , independent from the size and aspect ratio W, H of the source image.

Real samples. The images used for the $N = 1$ and $N = 10$ experiments are shown in Figure 1 and the appendix respectively (this is *all* the training data used in such experiments). For the special case of using a single training image, i.e. $N = 1$, we have chosen one photographic (2560×1920) and one drawn image (600×225), which we call *Image A* and *Image B*, respectively. The two images were manually selected as they contain rich texture and are diverse, but their choice was not optimized for performance. We test only two images due to the cost of running a full set of experiments (each image is expanded up to 1.2M times for training some of the models, as explained above). However, this is sufficient to prove our main points. We also test another (1165×585) *Image C* to ablate the “crowdedness” of an image, as this latter contains large areas covering no objects. While resolution matters to some extent as a bigger image contains more pixels, the information within is still far more correlated, and thus more redundant than sampling several smaller images. In particular, the resolution difference in Image A and B appears to be negligible in our experiments. For CIFAR-10, where $S = 32$ we only use Image B due to the resolution difference. In direct comparison, Image B

is the size of about 132 CIFAR images which is still much less than $d = 50,000$. For $N > 1$, we select the source images randomly from each method’s training set.

3.2 REPRESENTATION LEARNING METHODS

Generative models. Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) learn to generate images using an adversarial objective: a generator network maps noise samples to image samples, approximating a target image distribution and a discriminator network is tasked with distinguishing generated and real samples. Generator and discriminator are pitched one against the other and learned together; when an equilibrium is reached, the generator produces images indistinguishable (at least from the viewpoint of the discriminator) from real ones.

Bidirectional Generative Adversarial Networks (BiGAN) (Donahue et al., 2017; Dumoulin et al., 2016) are an extension of GANs designed to learn a useful image representation as an approximate inverse of the generator through joint inference on an encoding and the image. This method’s native augmentation uses random crops and random horizontal flips to learn features from $S = 128$ sized images. As opposed to the other two methods discussed below it employs leaky ReLU nonlinearities as is typical in GAN discriminators.

Rotation. Most image datasets contain pictures that are ‘upright’ as this is how humans prefer to take and look at them. This photographer bias can be understood as a form of implicit data labelling. RotNet (Gidaris et al., 2018) exploits this by tasking a network with predicting the upright direction of a picture after applying to it a random rotation multiple of 90 degrees (in practice this is formulated as a 4-way classification problem). The authors reason that the concept of ‘upright’ requires learning high level concepts in the image and hence this method is not vulnerable to exploiting low-level visual information, encouraging the network to learn more abstract features. In our experiments, we test this hypothesis by learning from impoverished datasets that may lack the photographer bias. The native augmentations that RotNet uses on the $S = 256$ inputs only comprise horizontal flips and non-scaled random crops to 224×224 .

Clustering. DeepCluster (Caron et al., 2018) is a recent state-of-the-art unsupervised representation learning method. This approach alternates k -means clustering to produce pseudo-labels for the data and feature learning to fit the representation to these labels. The authors attribute the success of the method to the prior knowledge ingrained in the structure of the convolutional neural network (Ulyanov et al., 2018).

The method alternates between a clustering step, in which k -means is applied on the PCA-reduced features with $k = 10^4$, and a learning step, in which the network is trained to predict the cluster ID for each image under a set of augmentations (random resized crops with $\beta = 0.08$, $\gamma = \frac{3}{4}$ and horizontal flips) that constitute its native augmentations used on top of the $S = 256$ input images.

4 EXPERIMENTS

We evaluate the representation learning methods on ImageNet and CIFAR-10/100 using linear probes (Section 4.1). After ablating various choices of transformations in our augmentation protocol (Section 4.2), we move to the core question of the paper: whether a large dataset is beneficial to unsupervised learning, especially for learning early convolutional features (Section 4.3).

4.1 LINEAR PROBES AND BASELINE ARCHITECTURE

In order to quantify if a neural network has learned useful feature representations, we follow the standard approach of using linear probes (Zhang et al., 2017). This amounts to solving a difficult task such as ImageNet classification by training a linear classifier on top of pre-trained feature representations, which are kept fixed. Linear classifiers heavily rely on the quality of the representation since their discriminative power is low.

We apply linear probes to all intermediate convolutional layers of networks and train on the ImageNet LSVRC-12 (Deng et al., 2009) and CIFAR-10/100 (Krizhevsky, 2009) datasets, which are the standard benchmarks for evaluation in self-supervised learning. Our base encoder architecture is AlexNet (Krizhevsky et al., 2012) with BatchNorm, since this is a good representative model and is most often used in other unsupervised learning work for the purpose of benchmarking. This model

	CIFAR-10			
	conv1	conv2	conv3	conv4
(a) Fully sup.	66.5	70.1	72.4	75.9
(b) Random feat.	57.8	55.5	54.2	47.3
(c) No aug.	57.9	56.2	54.2	47.8
(d) Jitter	58.9	58.0	57.0	49.8
(e) Rotation	61.4	58.8	56.1	47.5
(f) Scale	67.9	69.3	67.9	59.1
(g) Rot. & jitter	64.9	63.6	61.0	53.4
(h) Rot. & scale	67.6	69.9	68.0	60.7
(i) Jitter & scale	68.1	71.3	69.5	62.4
(j) All	68.1	72.3	70.8	63.5

Table 1: **Ablating data augmentation using MonoGAN (left)**. Training a linear classifier on the features extracted at different depths of the network for CIFAR-10.

Table 2: **ImageNet LSVRC-12 linear probing evaluation (below)**. A linear classifier is trained on the (downsampled) activations of each layer in the pretrained model. We report classification accuracy averaged over 10 crops. The $\dagger$ indicated that numbers are taken from (Zhang et al., 2017).

			ILSVRC-12				
Method, Reference		#images	conv1	conv2	conv3	conv4	conv5
(a)	Full-supervision [†]	1,281,167	19.3	36.3	44.2	48.3	50.5
(b)	(Oyallon et al., 2017): Scattering	0	-	18.9	-	-	-
(c)	Random [†]	0	11.6	17.1	16.9	16.3	14.1
(d)	(Krähenbühl et al., 2016): <i>k</i> -means [†]	≈ 160	17.5	23.0	24.5	23.2	20.6
(e)	(Donahue et al., 2017): BiGAN [†]	1,281,167	17.7	24.5	31.0	29.9	28.0
(f)	mono, Image A	1	20.4	30.9	33.4	28.4	16.0
(g)	mono, Image B	1	20.5	30.4	31.6	27.0	16.8
(h)	deka	10	16.2	16.5	16.5	13.1	7.5
(i)	kilo	1,000	16.1	17.7	18.3	17.6	13.5
(j)	(Gidaris et al., 2018): RotNet	1,281,167	18.8	31.7	38.7	38.2	36.5
(k)	mono, Image A	1	19.9	30.2	30.6	27.6	21.9
(l)	mono, Image B	1	17.8	27.6	27.9	25.4	20.2
(m)	deka	10	19.6	30.7	32.6	28.9	22.6
(n)	kilo	1,000	21.0	33.5	36.5	34.0	29.4
(o)	(Caron et al., 2018): DeepCluster	1,281,167	18.0	32.5	39.2	37.2	30.6
(p)	mono, Image A	1	20.7	31.5	32.5	28.5	21.0
(q)	mono, Image B	1	19.7	30.1	31.6	28.5	20.4
(r)	mono, Image C	1	18.9	29.2	31.5	28.9	23.5
(s)	deka	10	18.5	29.0	31.1	28.2	21.9
(t)	kilo	1,000	19.5	29.8	33.0	31.7	26.8

Table 3: **CIFAR-10/100**. Accuracy of linear classifiers on different network layers.

Model	Dataset	CIFAR-10				CIFAR-100			
		conv1	conv2	conv3	conv4	conv1	conv2	conv3	conv4
Fully supervised		66.5	70.1	72.4	75.9	38.7	43.6	44.4	46.5
Random		57.8	55.5	54.2	47.3	30.9	29.8	28.6	24.1
RotNet		64.4	65.6	65.6	59.1	36.0	35.9	34.2	25.8
GAN (CIFAR-10)		67.7	73.0	72.5	69.2	39.6	46.0	45.1	39.9
GAN (CIFAR-100)		-	-	-	-	38.1	42.2	44.0	46.6
MonoGAN		68.1	72.3	70.8	63.5	39.9	46.9	44.5	38.8

has five convolutional blocks (each comprising a linear convolution later followed by ReLU and optionally max pooling). We insert the probes right after the ReLU layer in each block, and denote these entry points `conv1` to `conv5`. Applying the linear probes at each convolutional layer allows studying the quality of the representation learned at different depths of the network.

Details. While linear probes are conceptually straightforward, there are several technical details that affect the final accuracy by a few percentage points. Unfortunately, prior work has used several slightly different setups, so that comparing results of different publications must be done with caution. To make matters more difficult, not all papers released evaluation source code. We prove this standardized testing code here².

²<https://github.com/yukimasano/linear-probes>

In our implementation, we follow the original proposal (Zhang et al., 2017) in pooling each representation to a vector with 9600, 9216, 9600, 9600, 9216 dimensions for conv1-5 using adaptive max-pooling, and absorb the batch normalization weights into the preceding convolutions. For evaluation on ImageNet we follow RotNet to train linear probes: images are resized such that the shorter edge has a length of 256 pixels, random crops of 224×224 are computed and flipped horizontally with 50% probability. Learning lasts for 36 epochs and the learning rate schedule starts from 0.01 and is divided by five at epochs 5, 15 and 25. The top-1 accuracy of the linear classifier is then measured on the ImageNet validation subset. This uses DeepCluster’s protocol, extracting 10 crops for each validation image (four at the corners and one at the center along with their horizontal flips) and averaging the prediction scores before the accuracy is computed. For CIFAR-10/100 data, we follow the same learning rate schedule and for both training and evaluation we do not reduce the dimensionality of the representations and keep the images’ original size of 32×32 .

4.2 EFFECT OF AUGMENTATIONS

In order to better understand which image transformations are important to learn a good feature representations, we analyze the impact of augmentation settings. For speed, these experiments are conducted using the CIFAR-10 images ($d = 50,000$ in the training set) and with the smaller source Image B and a GAN using the Wasserstein GAN formulation with gradient penalty (Gulrajani et al., 2017). The encoder is a smaller AlexNet-like CNN consisting of four convolutional layers (kernel sizes: 7, 5, 3, 3; strides: 3, 2, 2, 1) followed by a single fully connected layer as the discriminator. Given that the GAN is trained on a single image (w/ augmentations), we call this setting *MonoGAN*.

Table 1 reports all 2^3 combinations of the three main augmentations (scale, rotation, and jitter) and a randomly initialized network baseline (see Table 1 (b)) using the linear probes protocol discussed above. Without data augmentation the model only achieves marginally better performance than the random network (which also achieves a non-negligible level of performance (Ulyanov et al., 2017; Caron et al., 2018)). This is understandable since the dataset literally consists of a single training image cloned d times. Color jitter and rotation slightly improve the performance of all probes by 1-2% points, but random rescaling adds at least ten points at every depth (see Table 1 (f,h,i)) and is the most important single augmentation. A similar conclusion can be drawn when two augmentations are combined, although there are diminishing returns as more augmentations are combined. Overall, we find all three types of augmentations are of importance when training in the ultra-low data setting.

4.3 BENCHMARK EVALUATION

We analyze how performance varies as a function N , the number of actual samples that are used to generate the augmented datasets, and compare it to the gold-standard setup (in terms of choice of training data) defined in the papers that introduced each method. The evaluation is again based on linear probes (Section 4.1).

Mono is enough. From Table 2 we make the following observations. Training with just a single source image (f,g,k,l,p,q) is much better than random initialization (c) for all layers. Notably, these models also outperform Gabor-like filters from Scattering networks (Bruna & Mallat, 2013), which are hand crafted image features, replacing the first two convolutional layers as in (Oyallon et al., 2017). Using the same protocol as in the paper, this only achieves an accuracy of 18.9% compared to (p)’s conv2 > 30%.

More importantly, when comparing within pretext task, even with one image we are able to improve the quality of conv1-conv3 features compared to full (unsupervised) ImageNet training for GAN based self-supervision (e-i). For the other methods (j-n, o-s) we reach and also surpass the performance for the first layer and are within 1.5% points for the second. Given that the best unsupervised performance for conv2 is 32.5, our method using a single source Image A (Table 2, p) is remarkably close with 31.5.

Image contents. While we surpass the GAN based approach of (Donahue et al., 2017) for both single source images, we find more nuanced results for the other two methods: For RotNet, as expected, the photographic bias cannot be extracted from a single image. Thus its performance is low with little training data and increases together with the number of images (Table 2, j-n). When comparing Image A and B trained networks for RotNet, we find that the photograph yields better performance than the hand drawn animal image. This indicates that the method can extract rotation

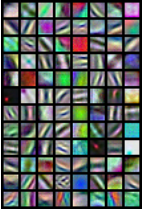
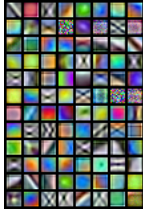
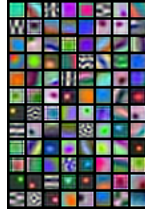
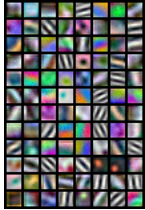
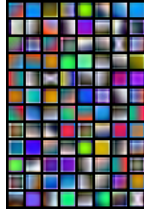
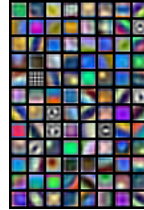
Method, Image A			Method, Image B		
BiGAN	RotNet	DeepCluster	BiGAN	RotNet	DeepCluster
					
20.4	19.9	20.7	20.5	17.8	19.7

Figure 2: **conv1 filters trained using a single image.** The 96 learned ($3 \times 11 \times 11$) filters for the first layer of AlexNet are shown for each single training image and method along with their linear classifier performance. For visualization, each filter is normalized to be in the range of $(-1, 1)$.

information from low level image features such as patches which is at first counter intuitive. Considering that the hand-drawn image does not work well, we can assume that lighting and shadows even in small patches can indeed give important cues on the up direction which can be learned even from a single (real) image. DeepCluster shows poor performance in `conv1` which we can improve upon in the single image setting (Table 2, o-r). Naturally, the image content matters: a trivial image without any image gradient (e.g. picture of a white wall) would not provide enough signal for any method. To better understand this issue, we also train DeepCluster on the much less cluttered Image C to analyze how much the image influences our claims. We find that even though this image contains large parts of sky and sea, the performance is only slightly lower than that of Image A. This finding indicates that the augmentations can even compensate for large untextured areas and the exact choice of image is not critical.

More than one image. While BiGAN fails to converge for $N \in \{10, 1000\}$, most likely due to issues in learning from a distribution which is neither whole images nor only patches, we find that both RotNet and DeepCluster improve their performance in deeper layers when increasing the number of training images. However, for `conv1` and `conv2`, a single image is enough. In deeper layers, DeepCluster seems to require large amounts of source images to yield the reported results as the deka- and kilo- variants start improving over the single image case (Table 2, o-t). This need for data also explains the gap between the two input images which have different resolutions. Summarizing Table 2, we can conclude that learning `conv1`, `conv2` and for the most part `conv3` (33.4 vs. 39.4) on over 1M images does not yield a significant performance increase over using one single training image — a highly unexpected result.

Generalization. In Table 3, we show the results of training linear classifiers for the CIFAR-10 dataset and compare against various baselines. We find that the GAN trained on the smaller Image B outperforms all other methods including the fully-supervised trained one for the first convolutional layer. We also outperform the same architecture trained on the full CIFAR-10 training set using RotNet, which might be due to the fact that either CIFAR images do not contain much information about the orientation of the picture or because they do not contain as many objects as in ImageNet. While the GAN trained on the whole dataset outperforms the MonoGAN on the deeper layers, the gap stays very small until the last layer. These findings are also reflected in the experiments on the CIFAR-100 dataset shown in Table 3. We find that our method obtains the best performance for the first two layers, even against the fully supervised version. The gap between our mono variant and the other methods increases again with deeper layers, hinting to the fact that we cannot learn very high level concepts in deeper layers from just one single image. These results corroborate the finding that our method allows learning very generalizable early features that are not domain dependent.

4.4 QUALITATIVE ANALYSIS

Visual comparison of weights. In Figure 2, we compare the learned filters of all first-layer convolutions of an AlexNet trained with the different methods and a single image. First, we find that the filters closely resemble those obtained via supervised training: Gabor-like edge detectors and various color blobs. Second, we find that the look is not easily predictive of its performance, e.g.

while generatively learned filters (BiGAN) show many edge detectors, its linear probes performance is about the same as that of DeepCluster which seems to learn many somewhat redundant point features. However, we also find that some edge detectors are required, as we can confirm from RotNet and DeepCluster trained on Image B, which yield less crisp filters and worse performances.

Table 4: **Finetuning experiments** The pre-trained model’s first two convolutions are left frozen (or replaced by the Scattering transform) and the network is retrained using ImageNet LSVRC-12 training set.

	Top-1
Full sup.	59.4
Random	42.6
Scattering	49.2
BiGAN, A	51.4
RotNet, A	49.5
DeepCluster A	52.5

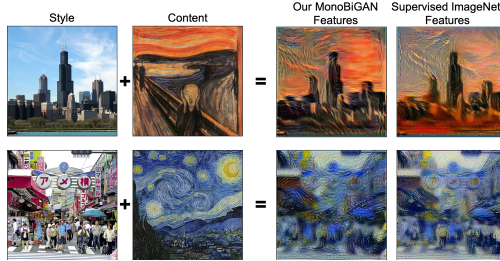


Figure 3: **Style transfer with single-image pre-training.** We show two style transfer results using the Image A trained BiGAN and the ImageNet pretrained AlexNet.

Fine-tuning instead of freezing. In Tab. 4, we show the results of retraining a network with the first two convolutional filters, or the scattering transform from (Oyallon et al., 2017), left frozen. We observe that our single image trained DeepCluster and BiGAN models achieve performances closes to the supervised benchmark. Notably, the scattering transform as a replacement for conv1-2 performs slightly worse than the analyzed single image methods. We also show in the appendix the results of retraining a network initialized with the first two convolutional layers obtained from a single image and subsequently linearly probing the model. The results are shown in Appendix Tab. 5 and we find that we can recover the performance of fully-supervised networks, i.e. the first two convolutional filters trained from just a single image generalize well and do not get stuck in an image specific minimum.

Neural style transfer. Lastly, we show how our features trained on only a single image can be used for other applications. In Figure 3 we show two basic style transfers using the method of (Gatys et al., 2016) from an official PyTorch tutorial³. Image content and style are separated and the style is transferred from the source to target image using all CNN features, not just the shallow layers. We visually compare the results of using our features and from full ImageNet supervision. We find almost no visual differences in the stylized images and can conclude that our early features are equally powerful as fully supervised ones for this task.

5 CONCLUSIONS

We have made the surprising observation that we can learn good and generalizable features through self-supervision from one single source image, provided that sufficient data augmentation is used. Our results complement recent works (Mahajan et al., 2018; Goyal et al., 2019) that have investigated self-supervision in the very large data regime. Our main conclusion is that these methods succeed perfectly in capturing the simplest image statistics, but that for deeper layers a gap exist with strong supervision which is compensated only in limited manner by using large datasets. This novel finding motivates a renewed focus on the role of augmentations in self-supervised learning and critical rethinking of how to better leverage the available data.

ACKNOWLEDGEMENTS.

We thank Aravindh Mahendran for fruitful discussions. Yuki Asano gratefully acknowledges support from the EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems (EP/L015897/1). The work is supported by ERC IDIU-638009.

³https://pytorch.org/tutorials/advanced/neural_style_tutorial.html

REFERENCES

- Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proc. ICCV*, pp. 37–45. IEEE, 2015. 3
- R. Arandjelović and A. Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017. 3
- Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proc. ICML*, pp. 517–526. PMLR, 2017. 2
- Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013. 3, 7
- M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *Proc. ECCV*, 2018. 2, 3, 5, 6, 7, 14
- N. Dalal and B Triggs. Histogram of Oriented Gradients for Human Detection. In *Proc. CVPR*, volume 2, pp. 886–893, 2005. 2, 3
- Virginia R de Sa. Learning classification with unlabeled data. In *NIPS*, pp. 112–119, 1994. 3
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 5
- Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, pp. 1422–1430, 2015. 2
- Jeff Donahue, Philipp Krhenbhl, and Trevor Darrell. Adversarial feature learning. *Proc. ICLR*, 2017. 2, 3, 5, 6, 7
- A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE PAMI*, 38(9): 1734–1747, Sept 2016. ISSN 0162-8828. doi: 10.1109/TPAMI.2015.2496141. 2
- V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016. 5
- D. Erhan, Y. Bengio, A. Courville, and P. Vincent. Visualizing higher-layer features of a deep network. Technical Report 1341, University of Montreal, Jun 2009. 14
- Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J Guibas. Geometry guided convolutional neural networks for self-supervised video representation learning. In *Proc. CVPR*, 2018. 3
- Rouhan Gao, Dinesh Jayaraman, and Kristen Grauman. Object-centric representation learning from unlabeled videos. In *Proc. ACCV*, 2016. 3
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proc. CVPR*, pp. 2414–2423, 2016. 2, 9
- R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. 15
- Spyros Gidaris, Praveen Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. ICLR*, 2018. 2, 3, 5, 6
- R. B. Girshick. Fast R-CNN. In *Proc. ICCV*, 2015. 1
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pp. 2672–2680, 2014. 5
- Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. *arXiv preprint arXiv:1905.01235*, 2019. 9

- Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *NIPS*, pp. 5767–5777, 2017. 7
- B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *Proc. ECCV*, 2012. 3
- M. Huh, P. Agrawal, and A. A. Efros. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016. 1
- Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. In *Proc. ICLR*, 2015. 3
- Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *Proc. ICCV*, 2015. 3
- Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *Proc. CVPR*, 2016. 3
- Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Proc. CVPR*, 2018. 2
- J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proc. ECCV*, pp. 694–711, 2016. 2
- R. Kat, R. Jevnisek, and S. Avidan. Matching pixels using co-occurrence statistics. In *Proc. ICCV*, pp. 1751–1759, 2018. 2
- Philipp Krähenbühl, Carl Doersch, Jeff Donahue, and Trevor Darrell. Data-dependent initializations of convolutional neural networks. *Proc. ICLR*, 2016. 6
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pp. 1106–1114, 2012. 5
- Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 5
- Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proc. CVPR*, 2017. 2
- Hsin-Ying Lee, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequence. In *Proc. ICCV*, 2017. 2
- D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2, 3
- Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 181–196, 2018. 9
- A. Mahendran, J. Thewlis, and A. Vedaldi. Cross pixel optical-flow similarity for self-supervised learning. In *Proc. ACCV*, 2018. 3
- T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-SVMs for object detection and beyond. In *Proc. ICCV*, 2011. 3
- Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *Proc. ECCV*, 2016. 2
- T Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. In *Proc. CVPR*, 2017. 2
- Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, pp. 69–84. Springer, 2016. 2

- Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proc. ICCV*, 2017. 2
- Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Proc. CVPR*, 2018. 2
- Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 3(3):e10, 2018. 14
- B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997. 2, 3
- M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In *Proc. CVPR*, 2014. 1
- Andrew Owens, Phillip Isola, Josh H. McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. Visually indicated sounds. In *Proc. CVPR*, pp. 2405–2413, 2016. 3
- Edouard Oyallon, Eugene Belilovsky, and Sergey Zagoruyko. Scaling the scattering transform: Deep hybrid networks. pp. 5618–5627, 2017. 3, 6, 7, 9
- Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proc. CVPR*, pp. 2536–2544, 2016. 2
- Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proc. CVPR*, 2017. 3
- Zhongzheng Ren and Yong Jae Lee. Cross-domain self-supervised multi-task feature learning using synthetic imagery. In *Proc. CVPR*, 2018. 2
- A. Rodriguez, V. Naresh Boddeti, BVK V. Kumar, and A. Mahalanobis. Maximum margin correlation filter: A new approach for localization and classification. *IEEE Transactions on Image Processing*, 22(2):631–643, 2013. 3
- Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. Singan: Learning a generative model from a single natural image. In *Computer Vision (ICCV), IEEE International Conference on*, 2019. 3
- Pierre Sermanet et al. Time-contrastive networks: Self-supervised learning from video. In *Proc. Intl. Conf. on Robotics and Automation*, 2018. 2
- H. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE Trans. on Medical Imaging*, 35(5):1285–1298, May 2016. ISSN 0278-0062. 1
- N. Srivastava, E. Mansimov, and R. Salakhudinov. Unsupervised learning of video representations using lstms. In *Proc. ICML*, 2015. 3
- I. Talmi, R. Mechrez, and L. Zelnik-Manor. Template matching with deformable diversity similarity. In *Proc. CVPR*, pp. 175–183, 2017. 2
- D. Ulyanov, A. Vedaldi, and V. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proc. CVPR*, 2017. 7
- D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Proc. CVPR*, 2018. 5
- X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In *Proc. ICCV*, pp. 2794–2802, 2015. 3
- Xiaolong Wang, Kaiming He, and Abhinav Gupta. Transitive invariance for self-supervised visual representation learning. In *Proc. ICCV*, 2017. 3
- D. Wei, J. Lim, A. Zisserman, and W. T. Freeman. Learning and using the arrow of time. In *Proc. CVPR*, 2018. 2

- J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *NIPS*, pp. 3320–3328, 2014. [1](#)
- M. D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proc. ECCV*, 2014. [14](#)
- Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. ECCV*, pp. 649–666. Springer, 2016. [2](#)
- Richard Zhang, Phillip Isola, and Alexei A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proc. CVPR*, 2017. [5](#), [6](#), [7](#), [14](#)

A APPENDIX

A.1 IMAGENET TRAINING IMAGES



Figure 4: ImageNet images for the $N = 10$ experiments.

The images used for the $N = 10$ experiments are shown in fig. 4.

A.2 VISUAL COMPARISON OF FILTERS

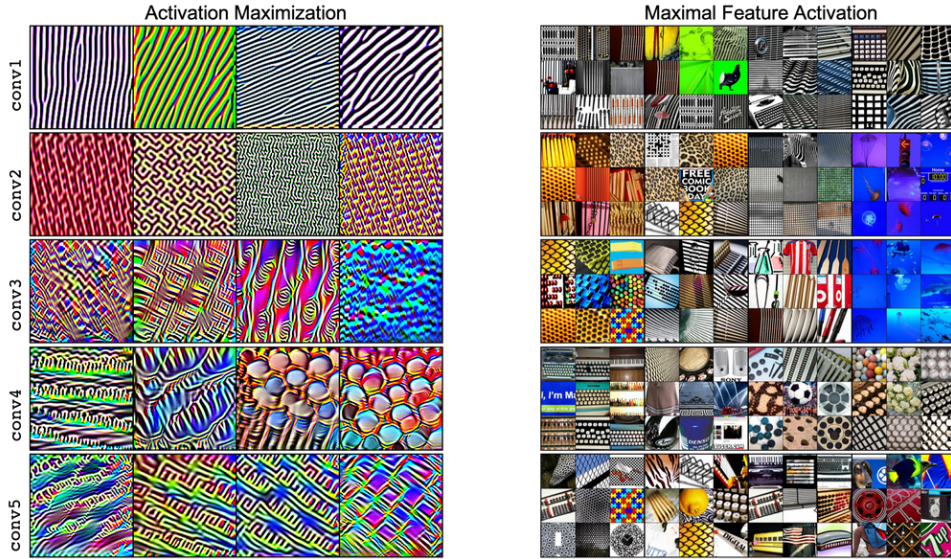


Figure 5: **Filter visualization.** We show activation maximization (left) and retrieval of top 9 activated images from the training set of ImageNet (right) for four random non-cherrypicked target filters. From top to bottom: conv1–5 of the BiGAN trained on a single image A. The filter visualization is obtained by learning a (regularized) input image that maximizes the response to the target filter using the library Lucid (Olah et al., 2018).

In order to understand what deeper neurons are responding to in our model, we visualize random neurons via activation maximization (Erhan et al., 2009; Zeiler & Fergus, 2014) in each layer. Additionally, we retrieve the top-9 images in the ImageNet training set that activate each neuron most in Figure 5. Since the mono networks are not trained on the ImageNet dataset, it can be used here for visualization. From the first convolutional layer we find typical neurons strongly reacting to oriented edges. In layers 2-4 we find patterns such as grids (conv2:3), and textures such as leopard skin (conv2:2) and round grid cover (conv4:4). Confirming our hypothesis that the neural network is only extracting patterns and not semantic information, we do not find any neurons particularly specialized to certain objects even in higher levels as for example dog faces or similar which can be found in supervised networks. This finding aligns with the observations of other unsupervised methods (Caron et al., 2018; Zhang et al., 2017). As most neurons extract simple patterns and

Table 5: **Finetuning experiments** Models are initialized using `conv1` and `conv2` from various single image trained models and the *whole* network is fine-tuned using ImageNet LSVRC-12 training set. Accuracy is averaged over 10 crops.

	c1	c2	c3	c4	c5
Full sup.	19.3	36.3	44.2	48.3	50.5
BiGAN, A	22.5	37.6	44.2	47.6	48.3
RotNet, A	22.0	38.2	44.8	49.2	51.8
DeepCluster, A	21.8	35.9	43.6	48.8	50.4

textures, the surprising effectiveness of training a network using a single image can be explained by the recent finding that even CNNs trained on ImageNet rely on texture (as opposed to shape) information to classify (Geirhos et al., 2019).

A.3 RETRAINING FROM SINGLE IMAGE INITIALIZATION

In Table 5, we initialize AlexNet models using the first two convolutional filters learned from a single image and retrain them using ImageNet. We find that the networks recover their performance fully and the first filters do not make the network stuck in a bad local minimum despite having been trained on a single image from a different distribution. The difference from the BiGAN to the full supervision model is likely due to it using a smaller input resolution (112 instead of 224), as the BiGAN’s output resolution is limited.

A.4 LINEAR PROBES ON IMAGENET

We show two plots of the ImageNet linear probes results (Table 2 of the paper) in fig. 6. On the left we plot performance per layer in absolute scale. Naturally the performance of the supervised model improves with depth, while all unsupervised models degrade after `conv3`. From the relative plot on the right, it becomes clear that with our training scheme, we can even slightly surpass supervised performance on `conv1` presumably since our model is trained with sometimes very small patches, thus receiving an emphasis on learning good low level filters. The gap between all self-supervised methods and the supervised baseline increases with depth, due to the fact that the supervised model is trained for this specific task, whereas the self-supervised models learn from a surrogate task without labels.

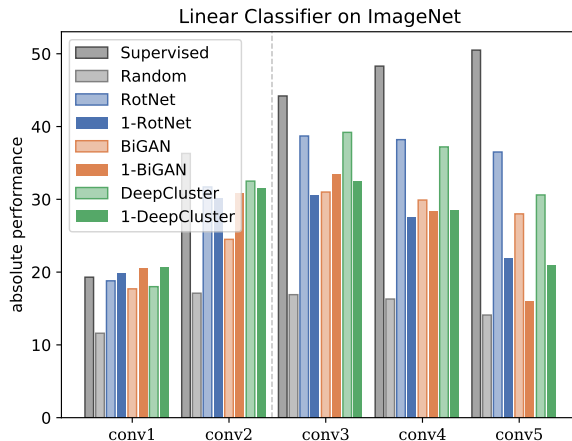


Figure 6: **Linear Classifiers on ImageNet.** Classification accuracies of linear classifiers trained on the representations from Table 2 are shown in absolute scale.

A.5 EXAMPLE AUGMENTED TRAINING DATA

In figs. 7 to 10 we show example patches generated by our augmentation strategy for the datasets with different N . Even though the images and patches are very different in color and shape distribu-

tion, our model learns weights that perform similarly in the linear probes benchmark (see Table 2 in the paper).

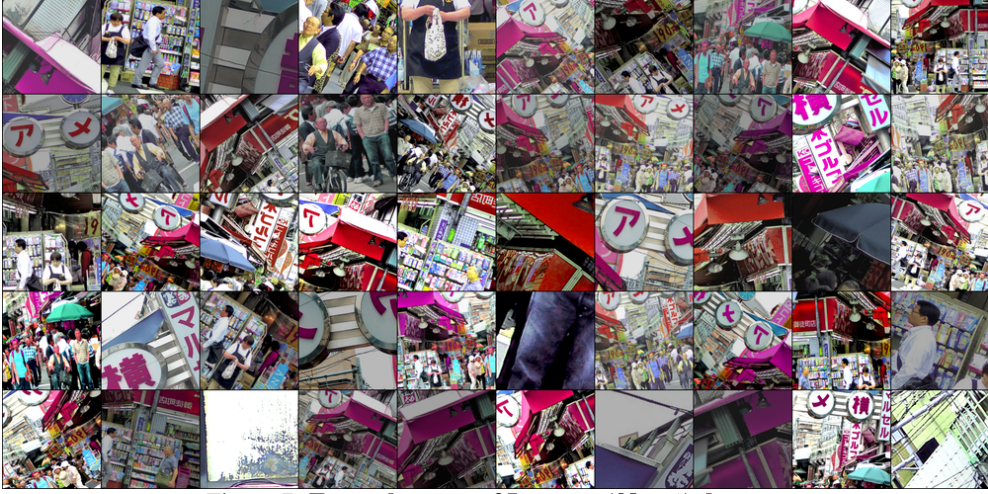


Figure 7: Example crops of Image A ($N = 1$) dataset.

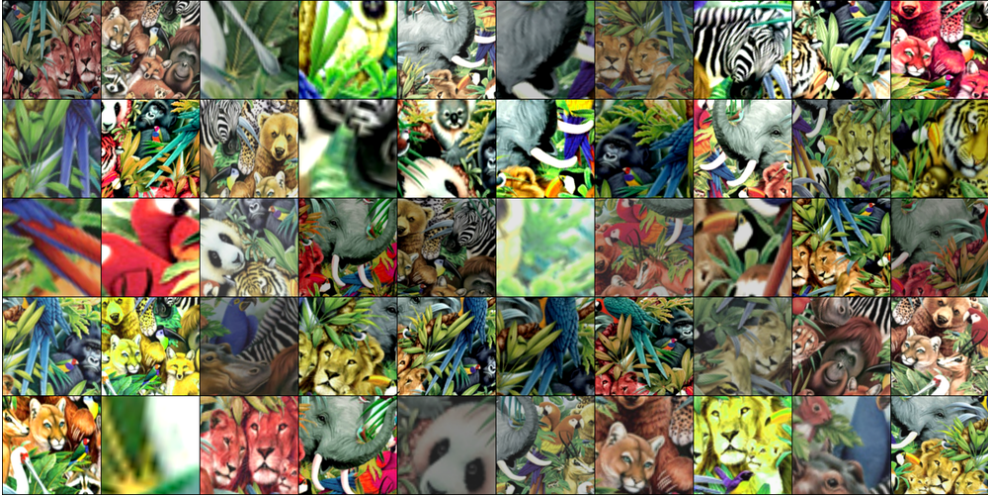


Figure 8: **Example crops of Image B ($N = 1$) dataset.** 50 samples were selected randomly.

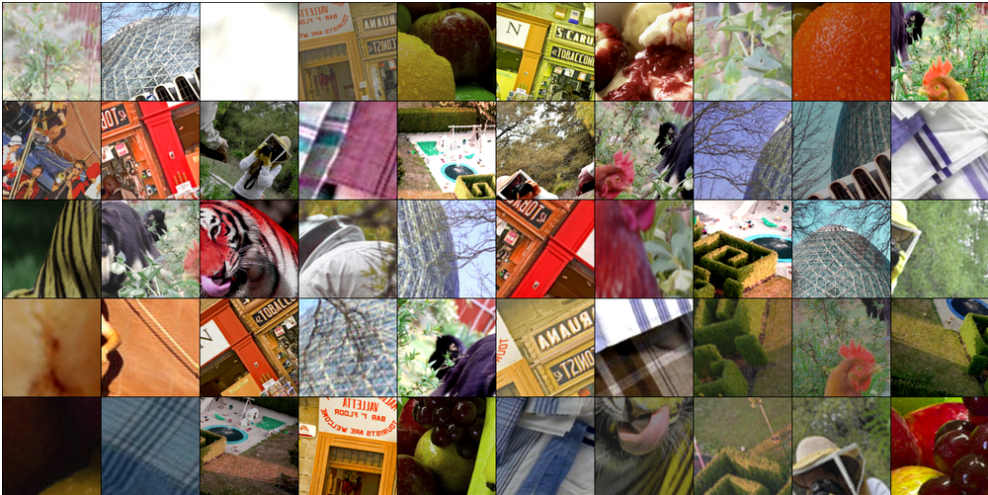


Figure 9: **Example crops of deka ($N = 10$) dataset.** 50 samples were selected randomly.



Figure 10: **Example crops of kilo ($N = 1000$) dataset.** 50 samples were selected randomly.

4

Self-labelling via simultaneous clustering and representation learning

This work was presented at the International Conference on Learning Representations (ICLR) 2020 as a spotlight.

SELF-LABELLING VIA SIMULTANEOUS CLUSTERING AND REPRESENTATION LEARNING

Yuki M. Asano

Christian Rupprecht

Andrea Vedaldi

Visual Geometry Group
University of Oxford

{yuki, chrisr, vedaldi}@robots.ox.ac.uk

ABSTRACT

Combining clustering and representation learning is one of the most promising approaches for unsupervised learning of deep neural networks. However, doing so naively leads to ill posed learning problems with degenerate solutions. In this paper, we propose a novel and principled learning formulation that addresses these issues. The method is obtained by maximizing the information between labels and input data indices. We show that this criterion extends standard cross-entropy minimization to an optimal transport problem, which we solve efficiently for millions of input images and thousands of labels using a fast variant of the Sinkhorn-Knopp algorithm. The resulting method is able to self-label visual data so as to train highly competitive image representations without manual labels. Our method achieves state of the art representation learning performance for AlexNet and ResNet-50 on SVHN, CIFAR-10, CIFAR-100 and ImageNet and yields the first self-supervised AlexNet that outperforms the supervised Pascal VOC detection baseline. Code and models are available¹.

1 INTRODUCTION

Learning from unlabelled data can dramatically reduce the cost of deploying machine learning algorithms to new applications, thus amplifying their impact in the real world. Self-supervision is an increasingly popular framework for learning without labels. The idea is to define pretext learning tasks that can be constructed from raw data alone, but that still result in neural networks that transfer well to useful applications.

Much of the research in self-supervision has focused on designing new pretext tasks. However, given supervised data such as ImageNet (Deng et al., 2009), the standard classification objective of minimizing the cross-entropy loss still results in better or at least as good pre-training than any of such methods (for a given amount of data and for a given model complexity). This suggests that the task of classification is sufficient for pre-training networks, provided that suitable data labels are available. In this paper, we thus focus on the problem of obtaining the labels automatically by designing a self-labelling algorithm.

Learning a deep neural network together while discovering the data labels can be viewed as *simultaneous clustering and representation learning*. The latter can be approached by combining cross-entropy minimization with an off-the-shelf clustering algorithm such as K -means. This is precisely the approach adopted by the recent DeepCluster method (Caron et al., 2018), which achieves excellent results in unsupervised representation learning. However, combining representation learning, which is a discriminative task, with clustering is not at all trivial. In particular, we show that the combination of cross-entropy minimization and K -means as adopted by DeepCluster cannot be described as the optimization of an overall learning objective; instead, there exist degenerate solutions that the algorithm avoids via particular implementation choices.

In order to address this technical shortcoming, in this paper, we contribute a new principled formulation for simultaneous clustering and representation learning. The starting point is to minimize a single loss, the cross-entropy loss, for learning the deep network *and* for estimating the data labels. This

¹<https://github.com/yukimasano/self-label>

is often done in semi-supervised learning and multiple instance learning. However, when applied naively to the unsupervised case, it immediately leads to a degenerate solution where all data points are mapped to the same cluster.

We solve this issue by adding the constraint that the labels must induce an equipartition of the data, which we show maximizes the information between data indices and labels. We also show that the resulting label assignment problem is the same as optimal transport, and can therefore be solved in polynomial time by linear programming. However, since we want to scale the algorithm to millions of data points and thousands of labels, standard transport solvers are inadequate. Thus, we also propose to use a fast version of the Sinkhorn-Knopp algorithm for finding an approximate solution to the transport problem efficiently at scale, using fast matrix-vector algebra.

Compared to methods such as DeepCluster, the new formulation is more principled and allows to more easily demonstrate properties of the method such as convergence. Most importantly, via extensive experimentation, we show that our new approach leads to significantly superior results than DeepCluster, achieving the new state of the art for representation learning approaches. In fact, the method’s performance surpasses others that use a single type of supervisory signal for self-supervision, and is on par or better than very recent contributions as well (Tian et al., 2019; He et al., 2019; Misra & van der Maaten, 2019; Oord et al., 2018).

2 RELATED WORK

Our paper relates to two broad areas of research: (a) self-supervised representation learning, and (b) more specifically, training a deep neural network using pseudo-labels, i.e. the assignment of a label to each image. We discuss closely related works for each.

Self-supervised learning: A wide variety of methods that do not require manual annotations have been proposed for the self-training of deep convolutional neural networks. These methods use various cues and proxy tasks namely, in-painting (Pathak et al., 2016), patch context and jigsaw puzzles (Doersch et al., 2015; Noroozi & Favaro, 2016; Noroozi et al., 2018; Mundhenk et al., 2017), clustering (Caron et al., 2018; Huang et al., 2019; Zhuang et al., 2019; Bautista et al., 2016), noise-as-targets (Bojanowski & Joulin, 2017), colorization (Zhang et al., 2016; Larsson et al., 2017), generation (Jenni & Favaro, 2018; Ren & Lee, 2018; Donahue et al., 2017; Donahue & Simonyan, 2019), geometry (Dosovitskiy et al., 2016), predicting transformations (Gidaris et al., 2018; Zhang et al., 2019) and counting (Noroozi et al., 2017). Most recently, contrastive methods have shown great performance gains, (Oord et al., 2018; Hénaff et al., 2019; Tian et al., 2019; He et al., 2019) by leveraging augmentation and adequate losses. In (Feng et al., 2019), predicting rotation (Gidaris et al., 2018) is combined with instance retrieval (Wu et al., 2018) and multiple tasks are combined in (Doersch & Zisserman, 2017).

Pseudo-labels for images: In the self-supervised domain, we find a spectrum of methods that either give each data point a unique label (Wu et al., 2018; Dosovitskiy et al., 2016) or train on a flexible number of labels with K -means (Caron et al., 2018), with mutual information (Ji et al., 2018) or with noise (Bojanowski & Joulin, 2017). In (Noroozi et al., 2018) a large network is trained with a pretext task and a smaller network is trained via knowledge transfer of the clustered data. Finally, (Bach & Harchaoui, 2008; Vo et al., 2019) use convex relaxations to regularized affine-transformation invariant linear clustering, but can not scale to larger datasets.

Our contribution is a simple method that combines a novel pseudo-label extraction procedure from raw data alone and the training of a deep neural network using a standard cross-entropy loss.

3 METHOD

We will first derive our self-labelling method, then interpret the method as optimizing labels and targets of a cross-entropy loss and finally analyze similarities and differences with other clustering-based methods.

3.1 SELF-LABELLING

Neural network pre-training is often achieved via a supervised data classification task. Formally, consider a deep neural network $\mathbf{x} = \Phi(I)$ mapping data I (e.g. images) to feature vectors $\mathbf{x} \in \mathbb{R}^D$. The model is trained using a dataset (e.g. ImageNet) of N data points $I_1, \dots, I_N$ with corresponding

labels $y_1, \dots, y_N \in \{1, \dots, K\}$, drawn from a space of K possible labels. The representation is followed by a *classification head* $h : \mathbb{R}^D \rightarrow \mathbb{R}^K$, usually consisting of a single linear layer, converting the feature vector into a vector of class scores. The class scores are mapped to class probabilities via the softmax operator:

$$p(y = \cdot | \mathbf{x}_i) = \text{softmax}(h \circ \Phi(\mathbf{x}_i)).$$

The model and head parameters are learned by minimizing the average cross-entropy loss

$$E(p|y_1, \dots, y_N) = -\frac{1}{N} \sum_{i=1}^N \log p(y_i | \mathbf{x}_i). \quad (1)$$

Training with objective (1) requires a labelled dataset. When labels are unavailable, we require a *self-labelling* mechanism to assign the labels automatically.

In semi-supervised learning, self-labelling is often achieved by jointly optimizing (1) with respect to the model $h \circ \Phi$ and the labels $y_1, \dots, y_N$. This can work if at least part of the labels are known, thus constraining the optimization. However, in the fully unsupervised case, it leads to a degenerate solution: eq. (1) is trivially minimized by assigning all data points to a single (arbitrary) label.

To address this issue, we first rewrite eq. (1) by encoding the labels as posterior distributions $q(y|\mathbf{x}_i)$:

$$E(p, q) = -\frac{1}{N} \sum_{i=1}^N \sum_{y=1}^K q(y|\mathbf{x}_i) \log p(y|\mathbf{x}_i). \quad (2)$$

If we set the posterior distributions $q(y|\mathbf{x}_i) = \delta(y - y_i)$ to be deterministic, the formulations in eqs. (1) and (2) are equivalent, in the sense that $E(p, q) = E(p|y_1, \dots, y_N)$. In this case, optimizing q is the same as reassigning the labels, which leads to the degeneracy. To avoid this, we add the constraint that the label assignments must partition the data in equally-sized subsets. Formally, the learning objective² is thus:

$$\min_{p, q} E(p, q) \quad \text{subject to} \quad \forall y : q(y|\mathbf{x}_i) \in \{0, 1\} \text{ and } \sum_{i=1}^N q(y|\mathbf{x}_i) = \frac{N}{K}. \quad (3)$$

The constraints mean that each data point $\mathbf{x}_i$ is assigned to exactly one label and that, overall, the N data points are split uniformly among the K classes.

The objective in eq. (3) is combinatorial in q and thus may appear very difficult to optimize. However, this is an instance of the *optimal transport problem*, which can be solved relatively efficiently. In order to see this more clearly, let $P_{yi} = p(y|\mathbf{x}_i) \frac{1}{N}$ be the $K \times N$ matrix of joint probabilities estimated by the model. Likewise, let $Q_{yi} = q(y|\mathbf{x}_i) \frac{1}{N}$ be $K \times N$ matrix of assigned joint probabilities. Using the notation of (Cuturi, 2013), we relax matrix Q to be an element of the *transportation polytope*

$$U(r, c) := \{Q \in \mathbb{R}_+^{K \times N} \mid Q\mathbb{1} = r, Q^\top \mathbb{1} = c\}. \quad (4)$$

Here $\mathbb{1}$ are vectors of all ones of the appropriate dimensions, so that r and c are the marginal projections of matrix Q onto its rows and columns, respectively. In our case, we require Q to be a matrix of conditional probability distributions that split the data uniformly, which is captured by:

$$r = \frac{1}{K} \cdot \mathbb{1}, \quad c = \frac{1}{N} \cdot \mathbb{1}.$$

With this notation, we can rewrite the objective function in eq. (3), up to a constant shift, as

$$E(p, q) + \log N = \langle Q, -\log P \rangle, \quad (5)$$

where $\langle \cdot \rangle$ is the Frobenius dot-product between two matrices and $\log$ is applied element-wise. Hence optimizing eq. (3) with respect to the assignments Q is equivalent to solving the problem:

$$\min_{Q \in U(r, c)} \langle Q, -\log P \rangle. \quad (6)$$

²We assume for simplicity that K divides N exactly, but the formulation is easily extended to any $N \geq K$ by setting the constraints to either $\lfloor N/K \rfloor$ or $\lfloor N/K \rfloor + 1$, in order to assure that there is a feasible solution.

This is a *linear program*, and can thus be solved in polynomial time. Furthermore, solving this problem always leads to an integral solution despite having relaxed Q to the continuous polytope $U(r, c)$, guaranteeing the exact equivalence to the original problem.

In practice, however, the resulting linear program is large, involving millions of data points and thousands of classes. Traditional algorithms to solve the transport problem scale badly to instances of this size. We address this issue by adopting a fast version (Cuturi, 2013) of the *Sinkhorn-Knopp algorithm*. This amounts to introducing a regularization term

$$\min_{Q \in U(r, c)} \langle Q, -\log P \rangle + \frac{1}{\lambda} \text{KL}(Q \| rc^\top), \quad (7)$$

where KL is the Kullback-Leibler divergence and $rc^\top$ can be interpreted as a $K \times N$ probability matrix. The advantage of this regularization term is that the minimizer of eq. (7) can be written as:

$$Q = \text{diag}(\alpha) P^\lambda \text{diag}(\beta) \quad (8)$$

where exponentiation is meant element-wise and α and β are two vectors of scaling coefficients chosen so that the resulting matrix Q is also a probability matrix (see (Cuturi, 2013) for a derivation). The vectors α and β can be obtained, as shown below, via a simple matrix scaling iteration.

For very large λ , optimizing eq. (7) is of course equivalent to optimizing eq. (6), but even for moderate values of λ the two objectives tend to have approximately the same optimizer (Cuturi, 2013). Choosing λ trades off convergence speed with closeness to the original transport problem. In our case, using a fixed λ is appropriate as we are ultimately interested in the final clustering and representation learning results, rather than in solving the transport problem exactly.

Our final algorithm’s core can be described as follows. We learn a model $h \circ \Phi$ and a label assignment matrix Q by solving the optimization problem eq. (6) with respect to both Q , which is a probability matrix, and the model $h \circ \Phi$, which determines the predictions $P_{yi} = \text{softmax}_y(h \circ \Phi(x_i))$. We do so by alternating the following two steps:

Step 1: representation learning. Given the current label assignments Q , the model is updated by minimizing eq. (6) with respect to (the parameters of) $h \circ \Phi$. This is the same as training the model using the common cross-entropy loss for classification.

Step 2: self-labelling. Given the current model $h \circ \Phi$, we compute the log probabilities P . Then, we find Q using eq. (8) by iterating the updates (Cuturi, 2013)

$$\forall y : \alpha_y \leftarrow [P^\lambda \beta]_y^{-1} \quad \forall i : \beta_i \leftarrow [\alpha^\top P^\lambda]_i^{-1}.$$

Each update involves a single matrix-vector multiplication with complexity $\mathcal{O}(NK)$, so it is relatively quick even for millions of data points and thousands of labels and so the cost of this method scales linearly with the number of images N . In practice, convergence is reached within 2 minutes on ImageNet when computed on a GPU. Also, note that the parameters α and β can be retained between steps, thus allowing a warm start of Step 2.

3.2 INTERPRETATION

As shown above, the formulation in eq. (2) uses scaled versions of the probabilities. We can interpret these by treating the data *index* i as a random variable with uniform distribution $p(i) = 1/N$ and by rewriting the posteriors $p(y|x_i) = p(y|i)$ and $q(y|x_i) = q(y|i)$ as conditional distributions with respect to the data index i instead of the feature vector x_i . With these changes, we can rewrite eq. (5) as

$$E(p, q) + \log N = - \sum_{i=1}^N \sum_{y=1}^K q(y, i) \log p(y, i) = H(q, p), \quad (9)$$

which is the cross-entropy between the joint label-index distributions $q(y, i)$ and $p(y, i)$. The minimum of this quantity w.r.t. q is obtained when $p = q$, in which case $E(q, q) + \log N$ reduces to the entropy $H_q(y, i)$ of the random variables y and i . Additionally, since we assumed that $q(i) = 1/N$, the marginal entropy $H_q(i) = \log N$ is constant and, due to the equipartition condition $q(y) = 1/K$, $H_q(y) = \log K$ is *also* constant. Subtracting these two constants from the entropy yields:

$$\min_p E(p, q) + \log N = E(q, q) + \log N = H_q(y, i) = H_q(y) + H_q(i) - I_q(y, i) = \text{const.} - I_q(y, i).$$

Thus we see that minimizing $E(p, q)$ is the same as maximizing the *mutual information* between the label y and the data index i .

In our formulation, the maximization above is carried out under the equipartition constraint. We can instead relax this constraint and directly maximize the information $I(y, i)$. However, by rewriting information as the difference $I(y, i) = H(y) - H(y|i)$, we see that the optimal solution is given by $H(y|i) = 0$, which states each data point i is associated to only one label deterministically, and by $H(y) = \ln K$, which is another way of stating the equipartition condition.

In other words, our learning formulation can be interpreted as maximizing the information between data indices and labels while explicitly enforcing the equipartition condition, which is implied by maximizing the information in any case. Compared to minimizing the entropy alone, maximizing information avoids degenerate solutions as the latter carry no mutual information between labels y and indices i . Similar considerations can be found in (Ji et al., 2018).

3.3 RELATION TO SIMULTANEOUS REPRESENTATION LEARNING AND CLUSTERING

In the discussion above, self-labelling amounts to assigning discrete labels to data and can thus be interpreted as clustering. Most of the traditional clustering approaches are *generative*. For example, K -means takes a dataset $\mathbf{x}_1, \dots, \mathbf{x}_N$ of vectors and partitions it into K classes in order to minimize the reconstruction error

$$E(\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, y_1, \dots, y_N) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \boldsymbol{\mu}_{y_i}\|^2 \quad (10)$$

where $y_i \in \{1, \dots, K\}$ are the data-to-cluster assignments and $\boldsymbol{\mu}_y$ are means approximating the vectors in the corresponding clusters. The K -means energy can thus be interpreted as the average data reconstruction error.

It is natural to ask whether a clustering method such as K -means, which is based on approximating the input data, could be combined with representation learning, which uses a discriminative objective. In this setting, the feature vectors $\mathbf{x} = \Phi(I)$ are extracted by the neural network Φ from the input data I . Unfortunately, optimizing a loss such as eq. (10) with respect to the clustering and representation parameters is meaningless: in fact, the obvious solution is to let the representation send all the data points to the same constant feature vector and setting all the means to coincide with it, in which case the K -means reconstruction error is zero (and thus minimal).

Nevertheless, DeepCluster (Caron et al., 2018) *does* successfully combine K -means with representation learning. DeepCluster can be related to our approach as follows. Step 1 of the algorithm, namely representation learning via cross-entropy minimization, is exactly the same. Step 2, namely self-labelling, differs: where we solve an optimal transport problem to obtain the pseudo-labels, they do so by running K -means on the feature vectors extracted by the neural network.

DeepCluster does have an obvious degenerate solution: we can assign all data points to the same label and learn a constant representation, achieving simultaneously a minimum of the cross-entropy loss in Step 1 and of the K -means loss in Step 2. The reason why DeepCluster avoids this pitfall is due to the particular interaction between the two steps. First, during Step 2, the features $\mathbf{x}_i$ are fixed so K -means cannot pull them together. Instead, the means spread to cover the features as they are, resulting in a balanced partitioning. Second, during the classification step, the cluster assignments y_i are fixed, and optimizing the features $\mathbf{x}_i$ with respect to the cross-entropy loss tends to separate them. Lastly, the method in (Caron et al., 2018) also uses other heuristics such as sampling the training data inversely to their associated clusters' size, leading to further regularization.

However, a downside of DeepCluster is that it does not have a single, well-defined objective to optimize, which means that it is difficult to characterize its convergence properties. By contrast, in our formulation, both Step 1 and Step 2 optimize *the same objective*, with the advantage that convergence to a (local) optimum is guaranteed.

3.4 AUGMENTING SELF-LABELLING VIA DATA TRANSFORMATIONS

Methods such as DeepCluster extend the training data via augmentations. In vision problems, this amounts to (heavily) distorting and cropping the input images at random. Augmentations are applied so that the neural network is encouraged to learn a labelling function which is *transformation invariant*.

In practice, this is crucial to learn good clusters and representations, so we adopt it here. This is achieved by setting $P_{y_i} = \mathbb{E}_t[\log \text{softmax}_y h \circ \Phi(tx_i)]$ where the transformations t are sampled at random. In practice, in Step 1 (representation learning), this is implemented via the application of the random transformations to data batches during optimization via SGD, which corresponds to the usual data augmentation scheme for deep neural networks. As noted in (YM. et al., 2020), and as can be noted by an analysis of recent publications (Hénaff et al., 2019; Tian et al., 2019; Misra & van der Maaten, 2019), augmentation is critical for good performance.

3.5 MULTIPLE SIMULTANEOUS SELF-LABELINGS

Intuitively, the same data can often be clustered in many equally good ways. For example, visual objects can be clustered by color, size, typology, viewpoint, and many other attributes. Since our main objective is to use clustering to learn a good data representation Φ , we consider a multi-task setting in which the same representation is shared among several different clustering tasks, which can potentially capture different and complementary clustering axis.

In our formulation, this is easily achieved by considering *multiple heads* (Ji et al., 2018) $h_1, \dots, h_T$, one for each of T clustering tasks (which may also have a different number of labels). Then, we optimize a sum of objective functions of the type eq. (6), one for each task, while sharing the parameters of the feature extractor Φ among them.

4 EXPERIMENTS

In this section, we evaluate the quality of the representations learned by our *Self Labelling* (SeLa) technique. We first test variants of our method, including ablating its components, in order to find an optimal configuration. Then, we compare our results to the state of the art in self-supervised representation learning, where we find that our method is the best among clustering-based techniques and overall state-of-the-art or at least highly competitive in many benchmarks. In the appendix, we also show qualitatively that the labels identified by our algorithm are usually meaningful and group visually similar concepts in the same clusters, often even capturing whole ImageNet classes.

4.1 SETUP

Linear probes. In order to quantify if a neural network has learned useful feature representations, we follow the standard approach of using *linear probes* (Zhang et al., 2017). This amounts to solving a difficult task, such as ImageNet classification, by training a linear classifier on top of a pre-trained feature representation, which is kept fixed. Linear classifiers heavily rely on the quality of the representation since their discriminative power is low. We apply linear probes to all intermediate convolutional blocks of representative networks. While linear probes are conceptually straightforward, there are several technical details that can affect the final accuracy, so we follow the standard protocol further outlined in the Appendix.

Data. For training data we consider ImageNet LSVRC-12 (Deng et al., 2009) and other smaller scale datasets. We also test our features by transferring them to MIT Places (Zhou et al., 2014). All of these are standard benchmarks for evaluation in self-supervised learning.

Architectures. Our base encoder architecture is AlexNet (Krizhevsky et al., 2012), since this is the most frequently used in other self-supervised learning works for the purpose of benchmarking. We inject the probes right after the ReLU layer in each of the five blocks, and denote these entry points `conv1` to `conv5`. Furthermore, since the `conv1` and `conv2` can be learned effectively from data augmentations alone (YM. et al., 2020), we focus the analysis on the deeper layers `conv2` to `conv5` which are more sensitive to the quality of the learning algorithm. In addition to AlexNet, we also test ResNet-50 (He et al., 2016) models. Further experimental details are given in the Appendix.

4.2 OPTIMAL CONFIGURATION AND ABLATIONS

In tables 1 and 5, we first validate various modelling and configuration choices. Two key hyper-parameters are the number of clusters K and the number of clustering heads T , which we denote in the experiments below with the shorthand “SeLa[$K \times T$]”. We run SeLa by alternating steps 1 and 2 as described in section 3.1. Step 1 amounts to standard CE training, which we run for a fixed number of epochs. Step 2 can be interleaved at any point in the optimization; to amortize its cost, we run it

Table 1: **Ablation: number of self-labelling steps.**

Method	#opt.	c3	c4	c5
SeLa [3k × 1]	0	20.8	18.3	13.4
SeLa [3k × 1]	40	42.7	43.4	39.2
SeLa [3k × 1]	80	43.0	44.7	40.9
SeLa [3k × 1]	160	42.4	44.6	40.7

Table 2: **Number of clusters K .**

Method	c3	c4	c5
SeLa [1k × 1]	40.1	42.1	38.8
SeLa [3k × 1]	43.0	44.7	40.9
SeLa [5k × 1]	42.5	43.9	40.2
SeLa [10k × 1]	42.2	43.8	39.7

Table 3: **Ablation: number of heads T .** (c4 for AlexNet)

Method	Architecture	Top-1
SeLa [3k × 1]	AlexNet	44.7
SeLa [3k × 10]	AlexNet	46.7
SeLa [3k × 1]	ResNet-50	51.8
SeLa [3k × 10]	ResNet-50	61.5

Table 4: **Different architectures.**

Method	Architecture	Top-1
SeLa [3k × 1]	AlexNet (small)	41.3
SeLa [3k × 1]	AlexNet	44.7
SeLa [3k × 1]	ResNet-50	51.8

Table 5: **Label transfer.**

Method	Source (Top-1)	Target (Top-1)
SeLa [3k × 10]	AlexNet (46.7)	AlexNet (46.5)
SeLa [3k × 1]	ResNet-50 (51.8)	AlexNet (45.0)
SeLa [3k × 10]	ResNet-50 (61.5)	AlexNet (48.4)

at most once per epoch, and usually less, with a schedule described and validated below. For these experiments, we train the representation and the linear probes on ImageNet.

Number of clusters K . Table 2, compares different values for K : moving from 1k to 3k improves the results, but larger numbers decrease the quality slightly.

Ablation: number of heads T . Table 3 shows that increasing the number of heads from $T = 1$ to $T = 10$ yields a large performance gain: +2% for AlexNet and +10% for ResNet. The latter more expressive model appears to benefit more from a more diverse training signal.

Ablation: number of self-labelling iterations. First, in table 1, we show that self-labelling (step 2) is essential for good performance, as opposed to only relying on the initial random label assignments and the data augmentations. For this, we vary the number of times the self-labelling algorithm (step 2) is run during training (#opts), from zero to once per step 1 epoch. We see that self-labelling is essential, with the best value around 80 (for 160 step 1 epochs in total).

Architectures. Table 4 compares a smaller variant of AlexNet which uses (64, 192) filters in its first two convolutional layers (Krizhevsky, 2014), to the standard variant with (96, 256) (Krizhevsky et al., 2012), all the way to a ResNet-50. SeLa works well in all cases, for large models such as ResNet but also smaller ones such as AlexNet, for which methods such as BigBiGAN (Donahue & Simonyan, 2019) or CPC (Hénaff et al., 2019) are unsuitable.

4.3 LABEL TRANSFER

An appealing property of SeLa is that the label it assigns to the images can be used to train another model from scratch, using standard supervised training. For instance, table 5 shows that, given the labels assigned by applying SeLa to AlexNet, we can re-train AlexNet from scratch using a shorter 90-epochs schedule with achieving the same final accuracy. This shows that the quality of the learned representation depends only the final label assignment, not on the fact that the representation is learned jointly with the labels. More interestingly, we can transfer labels between different architectures. For example, the labels obtained by applying SeLa [3k × 1] and SeLa [3k × 10] to ResNet-50 can be used to train a better AlexNet model than applying SeLa to the latter directly. For this reason, we publish on our website the self-labels for the ImageNet dataset in addition to the code and trained models.

4.4 SMALL-SCALE DATASETS

Here, we evaluate our method on relatively simple and small datasets, namely CIFAR-10/100 (Krizhevsky et al., 2009) and SVHN (Netzer et al., 2011). For this, we follow the experimental and evaluation protocol from the current state of the art in self-supervised learning in these datasets, AND (Huang et al., 2019). In table 6, we compare our method with the settings [128 × 10] for CIFAR-10, [512 × 10] for CIFAR-100 and [128 × 1] for SVHN to other published methods; details on the evaluation method are provided in the appendix. We observe that our proposed method outperforms the best previous method by 5.8% for CIFAR-10, by 9.5% for CIFAR-100 and by 0.8% for SVHN when training a linear classifier on top of the frozen network. The relatively minor gains on SVHN can be explained by the fact that the gap between the supervised

Table 6: Nearest Neighbour and linear classification evaluation on small datasets using AlexNet. Results of previous methods are taken from (Huang et al., 2019).

Method	Dataset		
	CIFAR-10	CIFAR-100	SVHN
Classifier/Feature	Linear	Classifier / conv5	
<i>Supervised</i>	91.8	71.0	96.1
Counting	50.9	18.2	63.4
DeepCluster	77.9	41.9	92.0
Instance	70.1	39.4	89.3
AND	77.6	47.9	93.7
SL	83.4	57.4	94.5
Classifier/Feature	Weighted kNN / FC		
<i>Supervised</i>	91.9	69.7	96.5
Counting	41.7	15.9	43.4
DeepCluster	62.3	22.7	84.9
Instance	60.3	32.7	79.8
AND	74.8	41.5	90.9
SL	77.6	44.2	92.8

Table 7: **PASCAL VOC finetuning.** VOC07-Classification %mAP, VOC07-Detection %mAP and VOC12-Segmentation %mIU. * denotes a larger AlexNet variant.

Method	PASCAL VOC Task			
	Cls. fc6-8	all	Det. all	Seg. all
ImageNet labels	78.9	79.9	59.1	48.0
Random	—	53.3	43.4	—
Random Rescaled	—	56.6	45.6	32.6
BiGAN	52.3	60.1	46.9	35.2
Context*	55.1	65.3	51.1	—
Context 2	—	69.6	55.8	41.4
CC+VGG	—	72.5	56.5	42.6
RotNet	70.9	73.0	54.4	39.1
DeepCluster*	72.0	73.4	55.4	45.1
RotNet+retrieval*	72.5	74.7	<u>58.0</u>	45.9
SeLa* [3k × 10]	73.1	75.3	55.9	43.7
SeLa* [3k × 10] [−]	74.4	75.9	57.8	44.7
SeLa* [3k × 10] [−] +Rot	75.6	77.2	59.2	<u>45.7</u>

Table 8: Nearest Neighbour and linear classification evaluation using imbalanced CIFAR-10 training data. We evaluate on the normal CIFAR-10 test set and on CIFAR-100 to analyze the transferability of the features. Difference to the supervised baseline in parentheses. See section 4.5 for details.

Training data/Method	kNN		Linear/conv5	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
<i>CIFAR-10, full</i>				
Supervised	92.1	24.0	90.2	54.2
ours (<i>K</i> -means) [128 × 1]	64.7 (−17.4)	19.3 (−4.7)	77.5 (−12.7)	45.6 (−8.8)
ours (SK) [128 × 1]	72.9 (−9.2)	28.9 (+4.9)	79.8 (−9.4)	49.4 (−4.8)
<i>CIFAR-10, light imbalance</i>				
Supervised	92.0	24.0	90.4	53.6
ours (<i>K</i> -means) [128 × 1]	64.2 (−17.8)	18.1 (−5.9)	77.0 (−13.4)	44.8 (−8.8)
ours (SK) [128 × 1]	71.7 (−10.3)	28.2 (+4.2)	79.5 (−10.9)	48.6 (−5.0)
<i>CIFAR-10, heavy imbalance</i>				
Supervised	86.7	22.6	86.8	51.4
ours (<i>K</i> -means) [128 × 1]	60.7 (−16.0)	17.8 (−4.8)	75.2 (−11.6)	44.3 (−7.1)
ours (SK) [128 × 1]	67.6 (−9.1)	26.7 (+3.9)	77.2 (−9.6)	47.5 (−2.9)

baseline and the self-supervised results is already very small ($< 3\%$). We also evaluate our method using weighted kNN using an embedding of size 128. We find that the proposed method consistently outperforms the previous state of the art by around 2% across these datasets, even though AND is based on explicitly learning local neighbourhoods.

4.5 IMBALANCED DATA EXPERIMENTS

In order to understand if our equipartition regularization is affected by the underlying class distribution of a dataset, we perform multiple ablation experiments on artificially imbalanced datasets in table 8. We consider three training datasets based on CIFAR-10. The first is the original dataset with 5000 images for each class (*full* in table 8). Second, we remove 50% of the images of one class (truck) while the rest remains untouched (*light imbalance*) and finally we remove 10% of one class, 20% of the second class and so on (*heavy imbalance*). On each of the three datasets we compare the performance of our method “ours (SK)” with a baseline that replaces our Sinkhorn-Knopp optimization with *K*-means clustering “ours (*K*-means)”. We also compare the performance to training a network under full supervision. The evaluation follows Huang et al. (2019) and is based on linear probing and kNN classification — both on CIFAR-10 and, to understand feature generalization, on CIFAR-100.

Table 9: **Linear probing evaluation – AlexNet.** A linear classifier is trained on the (downsampled) activations of each layer in the pretrained model. We **bold** the best result in each layer and underline the second best. The best layer is highlighted in **blue**. * denotes a larger AlexNet variant. $-$ refers to AlexNets trained with self-label transfer from a corresponding ResNet-50. "+Rot" refers to retraining using labels and an additional RotNet loss, "+ more aug." includes further augmentation during retraining. See Table A.2 in the Appendix for a full version of this table and details.

Method	ILSVRC-12					Places				
	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
ImageNet supervised, (Zhang et al., 2017)	19.3	36.3	44.2	48.3	50.5	22.7	34.8	38.4	39.4	38.7
Places supervised, (Zhang et al., 2017)	-	-	-	-	-	22.1	35.1	40.2	43.3	44.6
Random, (Zhang et al., 2017)	11.6	17.1	16.9	16.3	14.1	15.7	20.3	19.8	19.1	17.5
Inpainting, (Pathak et al., 2016)	14.1	20.7	21.0	19.8	15.5	18.2	23.2	23.4	21.9	18.4
BiGAN, (Donahue et al., 2017)	17.7	24.5	31.0	29.9	28.0	22.0	28.7	31.8	31.3	29.7
Instance retrieval, (Wu et al., 2018)	16.8	26.5	31.8	34.1	35.6	18.8	24.3	31.9	34.5	33.6
RotNet, (Gidaris et al., 2018)	18.8	31.7	38.7	38.2	36.5	21.5	31.0	35.1	34.6	33.7
AND*, (Huang et al., 2019)	15.6	27.0	35.9	39.7	37.9	-	-	-	-	-
CMC*, (Tian et al., 2019)	18.4	33.5	38.1	40.4	42.6	-	-	-	-	-
AET*, (Zhang et al., 2019)	19.3	35.4	44.0	43.6	42.4	22.1	32.9	37.1	36.2	34.7
RotNet+retrieval*, (Feng et al., 2019)	20.8	35.2	41.8	44.3	44.4	24.0	33.8	37.5	39.3	38.9
SeLa [3k $\times$ 10]*	20.3	32.2	38.6	41.4	39.6	24.5	31.9	36.7	38.0	37.0
SeLa [3k $\times$ 10] $^-$ +Rot*	20.6	32.3	40.4	43.1	42.3	24.0	31.7	37.1	39.0	37.6
SeLa [3k $\times$ 10] $^-$ +Rot*+more aug.	19.2	32.6	40.8	44.4	44.7	21.1	30.4	36.5	37.9	37.3
ImageNet supervised*	21.6	37.2	46.9	52.9	54.4	22.6	33.2	39.0	41.3	39.7
DeepCluster*, (Caron et al., 2018)	13.4	32.3	41.0	39.6	38.2	23.8	32.8	37.3	36.0	31.0
Local Agg. *, (Zhuang et al., 2019)	18.7	32.7	38.1	42.3	42.4	18.7	32.7	38.2	40.3	39.5
RotNet+retrieval*, (Feng et al., 2019)	22.2	38.2	45.7	48.7	48.3	25.5	36.0	40.1	42.2	41.3
SeLa [3k $\times$ 10]*	22.5	37.4	44.7	47.1	44.1	26.7	34.9	39.9	41.8	39.7
SeLa [3k $\times$ 10] $^-$ +Rot*	22.8	37.8	46.7	49.7	48.4	26.8	35.5	41.0	43.0	41.3
SeLa [3k $\times$ 10] $^-$ +Rot*+more aug.	21.9	37.1	46.0	50.0	50.0	23.4	33.0	39.4	41.4	39.9

To our surprise, we find that our method generalizes better to CIFAR-100 than the supervised baseline during kNN evaluation, potentially due to overfitting when training with labels. We also find that using SK optimization for obtaining pseudo-labels is always better than K -means on all metrics and datasets. When comparing the imbalance settings, we find that under the *light imbalance* scenario, the methods' performances are ranked the same and no method is strongly affected by the imbalance. Under the *heavy imbalance* scenario, all methods drop in performance. However, compared to full data and light imbalance, the gap between supervised and self-supervised even decreases slightly for both K -means and our method, indicating stronger robustness of self-supervised methods compared to a supervised one.

In conclusion, our proposed method does not rely on the data to contain the same number of images for every class and outperforms a K -means baseline even in very strong imbalance settings. This confirms the intuition that the equipartitioning constraint acts as a regularizer and does not exploit the class distribution of the dataset.

4.6 LARGE SCALE BENCHMARKS

To compare to the state of the art and concurrent work, we evaluate several architectures using linear probes on public benchmark datasets.

AlexNet. The main benchmark for feature learning methods is linear probing of an AlexNet trained on ImageNet. In table 9 we compare the performance across layers also on the Places dataset. We find that across both datasets our method outperforms DeepCluster and local Aggregation at every layer. From our ablation studies in tables 1-5 we also note that even our single head variant [3k $\times$ 1] outperforms both methods. Given that our method provides labels for a dataset that can be used for retraining a network quickly, we find that we can improve upon this initial performance. And by adopting a hybrid approach, similar to (Feng et al., 2019), of training an AlexNet with 10 heads and one additional head for computing the RotNet loss, we find further improvement. This result (SeLa

Table 10: **Linear evaluation - ResNet.** A linear layer is trained on top of the global average pooled features of ResNets. All evaluations use a single centred crop. We have separated much larger architectures such as RevNet-50 $\times$ 4 and ResNet-161. Methods in brackets use a augmentation policy learned from supervised training and methods with * are not explicit about which further augmentations they use. See Table A.3 in the Appendix for a full version of this table.

Method	Architecture	Top-1	Top-5
Supervised, (Donahue & Simonyan, 2019)	ResNet-50	76.3	93.1
Jigsaw, (Kolesnikov et al., 2019)	ResNet-50	38.4	—
Rotation, (Kolesnikov et al., 2019)	ResNet-50	43.8	—
CPC, (Oord et al., 2018)	ResNet-101	48.7	73.6
BigBiGAN, (Donahue & Simonyan, 2019)	ResNet-50	55.4	77.4
LocalAggregation, (Zhuang et al., 2019)	ResNet-50	60.2	—
Efficient CPC v2.1, (Hénaff et al., 2019)	ResNet-50	(63.8)	(85.3)
CMC, (Tian et al., 2019)	ResNet-50	(64.1)	(85.4)
MoCo, (He et al., 2019)	ResNet-50	60.6	—
PIRL, (Misra & van der Maaten, 2019)*	ResNet-50	63.6	—
SeLa [3k $\times$ 10]	ResNet-50	<u>61.5</u>	84.0
<i>other architectures</i>			
MoCo, (He et al., 2019)	RevNet-50 $\times$ 4	<u>68.6</u>	—
Efficient CPC v2.1, (Hénaff et al., 2019)	ResNet-161	71.5	90.1

[3k $\times$ 10][−]+Rot) achieves state of the art in unsupervised representation learning for AlexNet, with a gap of 1.3% to the previous best performance on ImageNet and surpasses the ImageNet supervised baseline transferred to Places by 1.7%.

ResNet. Training better models than AlexNets is not yet standardized in the feature learning community. In Table 10 we compare a ResNet-50 trained with our method to other works. With top-1 accuracy of 61.5, we outperform than all other methods including Local Aggregation, CPCv1 and MoCo that use the same level of data augmentation. We even outperform larger architectures such as BigBiGAN’s RevNet-50 $\times$ 4 and reach close to the performance of models using AutoAugment-style transformations.

4.7 FINE-TUNING: CLASSIFICATION, OBJECT DETECTION AND SEMANTIC SEGMENTATION

Finally, since pre-training is usually aimed at improving down-stream tasks, we evaluate the quality of the learned features by fine-tuning the model for three distinct tasks on the PASCAL VOC benchmark. In Table 7 we compare results with regard to multi-label classification, object detection and semantic segmentation on PASCAL VOC (Everingham et al., 2010).

As in the linear probe experiments, we find our method better than the current state of the art in detection and classification with both fine-tuning only the last fully connected layers and when fine-tuning the whole network (“all”). Notably, our fine-tuned AlexNet outperforms its supervised ImageNet baseline on the VOC detection task. Also for segmentation the method is very close (0.2%) to the best performing method. This shows that our trained network does not only learn useful feature representations but is also able to perform well when fine-tuned on actual down-stream tasks.

5 CONCLUSION

We present a self-supervised feature learning method that is based on clustering. In contrast to other methods, ours optimizes the same objective during feature learning and during clustering. This becomes possible through a weak assumption that the number of samples should be equal across clusters. This constraint is explicitly encoded in the label assignment step and can be solved for efficiently using a modified Sinkhorn-Knopp algorithm. Our method outperforms all other feature learning approaches and achieves SOTA on SVHN, CIFAR-10/100 and ImageNet for AlexNet and ResNet-50. By virtue of the method, the resulting self-labels can be used to quickly learn features for new architectures using simple cross-entropy training.

ACKNOWLEDGMENTS

Yuki Asano gratefully acknowledges support from the EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems (EP/L015897/1). We are also grateful to ERC IDIU-638009, AWS Machine Learning Research Awards (MLRA) and the use of the University of Oxford Advanced Research Computing (ARC).

REFERENCES

- Francis R. Bach and Zaïd Harchaoui. Diffrac: a discriminative and flexible framework for clustering. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis (eds.), *Advances in Neural Information Processing Systems 20*, pp. 49–56. Curran Associates, Inc., 2008. 2
- Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019. 18
- Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Björn Ommer. Cliqecnn: Deep unsupervised exemplar learning. In *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*, pp. 3846–3854, 2016. 2
- Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proc. ICML*, pp. 517–526. PMLR, 2017. 2
- M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *Proc. ECCV*, 2018. 1, 2, 5, 9, 14, 15, 16, 17
- Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pp. 2292–2300, 2013. 3, 4, 14
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 1, 6
- Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proc. ICCV*, 2017. 2, 18
- Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, pp. 1422–1430, 2015. 2, 17
- Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning, 2019. 2, 7, 10, 18
- Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *Proc. ICLR*, 2017. 2, 9, 17
- A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE PAMI*, 38(9):1734–1747, Sept 2016. ISSN 0162-8828. doi: 10.1109/TPAMI.2015.2496141. 2
- Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010. 10
- Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10364–10374, 2019. 2, 9, 17
- Spyros Gidaris, Praveen Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. ICLR*, 2018. 2, 9, 17
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016. 6
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2019. 2, 10, 18

- Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272v2*, 2019. 2, 6, 7, 10, 18
- Jiabo Huang, Q Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning by neighbourhood discovery. In *Proceedings of the International Conference on machine learning (ICML)*, 2019. 2, 7, 8, 9, 14, 17
- Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Proc. CVPR*, 2018. 2, 17
- Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information distillation for unsupervised image segmentation and clustering. *arXiv preprint arXiv:1807.06653*, 2018. 2, 5, 6
- Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. *arXiv preprint arXiv:1901.09005*, 2019. 10, 18
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pp. 1106–1114, 2012. 6, 7
- Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014. 7
- Alex Krizhevsky et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 7
- Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proc. CVPR*, 2017. 2
- Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations, 2019. 2, 6, 10, 18
- T Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. In *Proc. CVPR*, 2017. 2
- T. Nathan Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. pp. 9339–9348, 2018. 17
- Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Ng. Reading digits in natural images with unsupervised feature learning. *NIPS*, 01 2011. 7
- Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, pp. 69–84. Springer, 2016. 2, 17
- Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proc. ICCV*, 2017. 2, 17
- Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Proc. CVPR*, 2018. 2, 17
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 10, 18
- Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proc. CVPR*, pp. 2536–2544, 2016. 2, 9, 17
- Zhongzheng Ren and Yong Jae Lee. Cross-domain self-supervised multi-task feature learning using synthetic imagery. In *Proc. CVPR*, 2018. 2
- Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding, 2019. 2, 6, 9, 10, 17, 18
- Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11(Oct):2837–2854, 2010. 14

- Huy V Vo, Francis Bach, Minsu Cho, Kai Han, Yann LeCun, Patrick Pérez, and Jean Ponce. Unsupervised image matching and object discovery as optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8287–8296, 2019. 2
- Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018. 2, 9, 14, 17
- Asano YM., Rupprecht C., and Vedaldi A. A critical analysis of self-supervision, or what we can learn from a single image. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=Blesx6EYvr>. 6
- Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2547–2555, 2019. 2, 9, 17
- Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. ECCV*, pp. 649–666. Springer, 2016. 2, 17
- Richard Zhang, Phillip Isola, and Alexei A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proc. CVPR*, 2017. 6, 9, 14, 17
- Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pp. 487–495, 2014. 6
- Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6002–6012, 2019. 2, 9, 10, 17, 18

A APPENDIX

A.1 IMPLEMENTATION DETAILS

Learning Details Unless otherwise noted, we train all our self-supervised models with SGD and initial learning rate 0.05 for 400 epochs with two learning rate drops where we divide the rate by ten at 150 and 300 and 350 epochs. We spread our pseudo-label optimizations throughout the whole training process in a logarithmic distribution. We optimize the label assignment at $t_i = \left(\frac{i}{M-1}\right)^2$, $i \in \{1, \dots, M\}$, where M is the user-defined number of optimizations and t_i is expressed as a fraction of total training epochs. For the Sinkhorn-Knopp optimization we set $\lambda = 25$ as in (Cuturi, 2013). We use standard data augmentations during training that consist of randomly resized crops, horizontal flipping and adding noise, as in (Wu et al., 2018).

Quantitative Evaluation – Technical Details. Unfortunately, prior work has used several slightly different setups, so that comparing results between different publications must be done with caution.

In our ImageNet implementation, we follow the original proposal (Zhang et al., 2017) in pooling each representation to a vector with 9600, 9216, 9600, 9216 dimensions for conv1–5 using adaptive max-pooling, and absorb the batch normalization weights into the preceding convolutions. For evaluation on ImageNet we follow RotNet to train linear probes: images are resized such that the shorter edge has a length of 256 pixels, random crops of 224×224 are computed and flipped horizontally with 50% probability. Learning lasts for 36 epochs and the learning rate schedule starts from 0.01 and is divided by five at epochs 5, 15 and 25. The top-1 accuracy of the linear classifier is then measured on the ImageNet validation subset by optionally extracting 10 crops for each validation image (four at the corners and one at the center along with their horizontal flips) and averaging the prediction scores before the accuracy is computed or just taking the a centred crop. For CIFAR-10/100 and SVHN we train AlexNet architectures on the resized images with batchsize 128, learning rate 0.03 and also the same image augmentations (random resized crops, color jitter and random grayscale) as is used in prior work (Huang et al., 2019). We use the same linear probing protocol as for our ImageNet experiments but without using 10 crops. For the weighted kNN experiments we use $k = 50$, $\sigma = 0.1$ and we use an embedding of size 128 as done in previous works.

In Table 9, when retraining an AlexNet using ResNet generated labels, we can apply heavier augmentation strategies as the labels are kept constant. Hence for the experiments denoted by "+ more aug.", in addition to the usual augmentations, we further randomly apply one of equalize, autocontrast and sharpening. We find that this raises the performance for ImageNet but lowers the performance on Places by a small amount, hence illuminating the need to always also report performance on both datasets.

A.2 FURTHER DETAILS

NMI over time In fig. A.1 we find that most learning takes place in the early epochs, and we reach a final NMI value of around 66%. Similarly, we find that due to the updating of the pseudo-labels at regular intervals and our data augmentation, the pseudo-label accuracies keep continuously rising without overfitting to these labels.

Clustering metrics In table A.1, we report standard clustering metrics (see (Vinh et al., 2010) for detailed definitions) of our trained models with regards to the ImageNet validation set ground-truth labels. These metrics include chance-corrected metrics which are the adjusted normalized mutual information (NMI) and the adjusted Rand-Index, as well as the default NMI, also reported in DeepCluster (Caron et al., 2018).

Conv1 filters In fig. A.3 we show the first convolutional filters of two of our trained models. We can find the typical Gabor-like edge detectors as well as color blobs and dot-detectors.

Entropy over time In fig. A.4, we show how the distribution of entropy with regards to the true ImageNet labels changes with training time. We find that while at first, all 3000 pseudo-labels contain random real ImageNet labels, yielding high entropy of around $6 \approx \ln(400) = \ln(1.2 \cdot 10^6 / 3000)$. Towards the end of training we arrive at a broad spectrum of entropies with some as low as

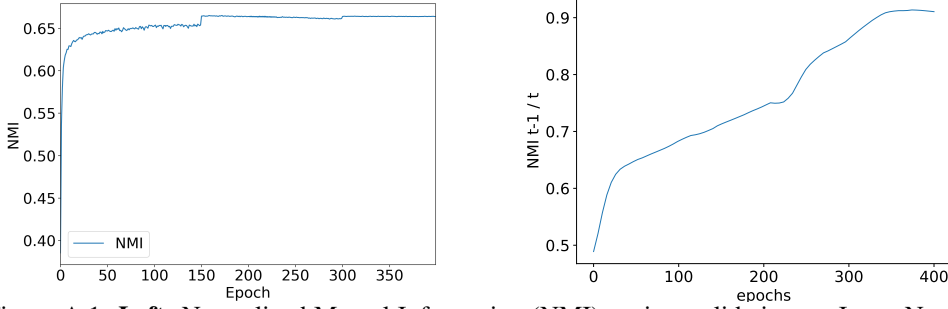


Figure A.1: **Left:** Normalized Mutual Information (NMI) against validation set ImageNet labels. This measure is not used for training but indicates how good a clustering is. **Right:** Similarities of consecutive labellings using NMI. Both plots use the $[10k \times 1]$ AlexNet for comparability with the DeepCluster paper (Caron et al., 2018).

Table A.1: Clustering metrics that compare with ground-truth labels of the ImageNet validation set (with 1-crop). For reference, we provide the best Top-1 error on ImageNet linear probing (as reported in the main part).*: for the multi-head variants, we simply use predictions of a randomly picked, single head.

Variant	Metrics			Top-1 Acc.
	NMI	adjusted NMI	adjusted Rand-Index	
SeLa $[1k \times 1]$ AlexNet	50.5%	12.2%	2.7%	42.1%
SeLa $[3k \times 1]$ AlexNet	59.1%	9.8%	2.5%	44.7%
SeLa $[5k \times 1]$ AlexNet	66.2%	7.4%	1.8%	43.9%
SeLa $[10k \times 1]$ AlexNet	66.4%	4.7%	1.0%	43.8%
SeLa $[3k \times 1]$ ResNet-50	60.0%	13.5%	3.8%	51.8%
SeLa $[3k \times 10]^*$ ResNet-50	66.3%	26.4%	10.3%	61.5%

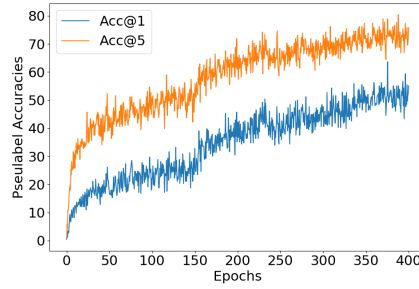


Figure A.2: Pseudo-label accuracies for the training data versus training time for the $[10k \times 1]$ AlexNet.

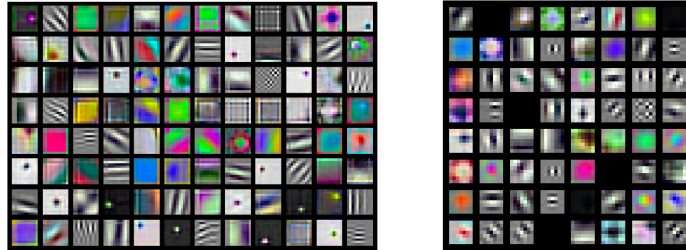


Figure A.3: Visualization of the first convolutional layers of our $[3k \times 10]$ AlexNet (left) and the $[1k \times 1]$ ResNet-50 (right). The filters are scaled to lie between (0,1) for visualization.

$0.07 \approx \ln(1.07)$ (see Fig. A.5 and A.6 for low entropy label visualizations) and the mean around $4.2 \approx \ln(66)$ (see Fig. A.7 and A.8 for randomly chosen labels' visualizations).

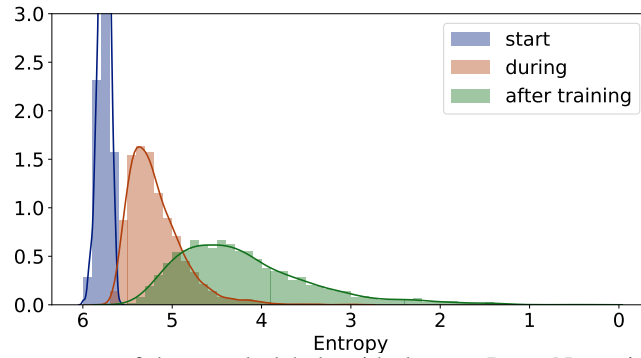


Figure A.4: Cross-entropy of the pseudo-labels with the true ImageNet training set labels. This measure is not used for training but indicates how good a clustering is. This plot uses the $[10k \times 1]$ AlexNet to compare to the equivalent plot in (Caron et al., 2018).

A.3 COMPLETE TABLES

In the following, we report the unabridged tables with all related work.

Table A.2: **Linear probing evaluation – AlexNet.** A linear classifier is trained on the (downsampled) activations of each layer in the pretrained model. We **bold** the best result in each layer and underline the second best. The best layer is highlighted in **blue**. * denotes a larger AlexNet variant. – refers to AlexNets trained with self-label transfer from a corresponding ResNet-50. "+Rot" refers to retraining using labels and an additional RotNet loss, "+ more aug." includes further augmentation during retraining.

Method	ILSVRC-12					Places				
	c1	c2	c3	c4	c5	c1	c2	c3	c4	c5
ImageNet supervised, (Zhang et al., 2017)	19.3	36.3	44.2	48.3	50.5	22.7	34.8	38.4	39.4	38.7
Places supervised, (Zhang et al., 2017)	-	-	-	-	-	22.1	35.1	40.2	43.3	44.6
Random, (Zhang et al., 2017)	11.6	17.1	16.9	16.3	14.1	15.7	20.3	19.8	19.1	17.5
Random*	15.6	16.8	17.4	15.6	10.6	16.5	17.6	18.6	18.1	16.3
Inpainting, (Pathak et al., 2016)	14.1	20.7	21.0	19.8	15.5	18.2	23.2	23.4	21.9	18.4
BiGAN, (Donahue et al., 2017)	17.7	24.5	31.0	29.9	28.0	22.0	28.7	31.8	31.3	29.7
Context*, (Doersch et al., 2015)	16.2	23.3	30.2	31.7	29.6	19.7	26.7	31.9	32.7	30.9
Colorization, (Zhang et al., 2016)	13.1	24.8	31.0	32.6	31.8	16.0	25.7	29.6	30.3	29.7
Jigsaw, (Noroozi & Favaro, 2016)	18.2	28.8	34.0	33.9	27.1	23.0	31.9	35.0	34.2	29.3
Counting, (Noroozi et al., 2017)	18.0	30.6	34.3	32.5	25.7	23.3	33.9	36.3	34.7	29.6
SplitBrain, (Zhang et al., 2017)	17.7	29.3	35.4	35.2	32.8	21.3	30.7	34.0	34.1	32.5
Instance retrieval, (Wu et al., 2018)	16.8	26.5	31.8	34.1	35.6	18.8	24.3	31.9	34.5	33.6
CC+VGG-, (Noroozi et al., 2018)	19.2	32.0	37.3	37.1	34.6	22.9	34.2	37.5	37.1	34.4
Context 2 (Mundhenk et al., 2018)	19.6	31.8	37.6	37.8	33.7	23.7	34.2	37.2	37.2	34.9
RotNet, (Gidaris et al., 2018)	18.8	31.7	38.7	38.2	36.5	21.5	31.0	35.1	34.6	33.7
Artifacts, (Jenni & Favaro, 2018)	19.5	33.3	37.9	38.9	34.9	23.3	34.3	36.9	37.3	34.4
AND*, (Huang et al., 2019)	15.6	27.0	35.9	39.7	37.9	-	-	-	-	-
CMC*, (Tian et al., 2019)	18.4	33.5	38.1	40.4	42.6	-	-	-	-	-
AET*, (Zhang et al., 2019)	19.3	35.4	44.0	43.6	42.4	22.1	32.9	37.1	36.2	34.7
RotNet+retrieval*, (Feng et al., 2019)	20.8	35.2	41.8	44.3	44.4	24.0	33.8	37.5	39.3	38.9
SeLa [3k × 10]*	20.3	32.2	38.6	41.4	39.6	24.5	31.9	36.7	38.0	37.0
SeLa [3k × 10] ⁻ +Rot*	20.6	32.3	40.4	43.1	42.3	24.0	31.7	37.1	39.0	37.6
SeLa [3k × 10] ⁻ +Rot*+more aug.	19.2	32.6	40.8	44.4	44.7	21.1	30.4	36.5	37.9	37.3
ImageNet supervised*	21.6	37.2	46.9	52.9	54.4	22.6	33.2	39.0	41.3	39.7
Random*	17.6	20.3	20.6	17.8	11.0	19.2	20.7	21.8	21.3	19.0
DeepCluster (RGB)*, (Caron et al., 2018)	18.0	32.5	39.2	37.2	30.6	-	-	-	-	-
DeepCluster*, (Caron et al., 2018)	13.4	32.3	41.0	39.6	38.2	23.8	32.8	37.3	36.0	31.0
Local Agg. *, (Zhuang et al., 2019)	18.7	32.7	38.1	42.3	42.4	18.7	32.7	38.2	40.3	39.5
RotNet+retrieval*, (Feng et al., 2019)	22.2	38.2	45.7	48.7	48.3	25.5	36.0	40.1	42.2	41.3
SeLa [3k × 10]*	22.5	37.4	44.7	47.1	44.1	26.7	34.9	39.9	41.8	39.7
SeLa [3k × 10] ⁻ +Rot*	22.8	37.8	46.7	49.7	48.4	26.8	35.5	41.0	43.0	41.3
SeLa [3k × 10] ⁻ +Rot*+more aug.	21.9	37.1	46.0	50.0	50.0	23.4	33.0	39.4	41.4	39.9

Table A.3: **Linear evaluation - ResNet.** A linear layer is trained on top of the global average pooled features of ResNets. All evaluations use a single centred crop. We have separated much larger architectures such as RevNet-50 $\times$ 4 and ResNet-161. Methods in brackets use a augmentation policy learned from supervised training and methods with * are not explicit about which further augmentations they use.

Method	Architecture	Top-1	Top-5
Supervised, (Donahue & Simonyan, 2019)	ResNet-50	76.3	93.1
Supervised, (Donahue & Simonyan, 2019)	ResNet-101	77.8	93.8
Jigsaw, (Kolesnikov et al., 2019)	ResNet-50	38.4	—
RelPathLoc, (Kolesnikov et al., 2019)	ResNet-50	42.2	—
Exemplar, (Kolesnikov et al., 2019)	ResNet-50	43.0	—
Rotation, (Kolesnikov et al., 2019)	ResNet-50	43.8	—
Multi-task, (Doersch & Zisserman, 2017)	ResNet-101	—	69.3
CPC, (Oord et al., 2018)	ResNet-101	48.7	73.6
BigBiGAN, (Donahue & Simonyan, 2019)	ResNet-50	55.4	77.4
LocalAggregation, (Zhuang et al., 2019)	ResNet-50	60.2	—
Efficient CPC v2.1, (Hénaff et al., 2019)	ResNet-50	(63.8)	(85.3)
CMC, (Tian et al., 2019)	ResNet-50	(64.1)	(85.4)
MoCo, (He et al., 2019)	ResNet-50	60.6	—
PIRL, (Misra & van der Maaten, 2019)*	ResNet-50	63.6	—
SeLa [3k $\times$ 10]	ResNet-50	<u>61.5</u>	84.0
<i>other architectures</i>			
Rotation, (Kolesnikov et al., 2019)	RevNet-50 $\times$ 4	53.7	—
BigBiGAN, (Donahue & Simonyan, 2019)	RevNet-50 $\times$ 4	60.8	81.4
AMDIM, (Bachman et al., 2019)	Custom-103	(67.4)	(81.8)
CMC, (Tian et al., 2019)	RevNet-50 $\times$ 4	68.4	88.2
MoCo, (He et al., 2019)	RevNet-50 $\times$ 4	<u>68.6</u>	—
Efficient CPC v2.1, (Hénaff et al., 2019)	ResNet-161	71.5	90.1

A.4 LOW ENTROPY PSEUDOCASSES



Figure A.5: Here we show a random sample of images associated to the lowest entropy pseudoclasses. The entropy is given by true image labels which are also shown as a frame around each picture with a random color. This visualization uses ResNet-50 [3k × 1]. The entropy varies from 0.07 — 0.83



Figure A.6: Visualization of pseudoclasses on the validation set. Here we show random samples of validation set images associated to the lowest entropy pseudoclasses of training set. For further details, see Figure A.5. Classes with less than 9 images are sampled with repetition.

A.5 RANDOM PSEUDOCLASSES

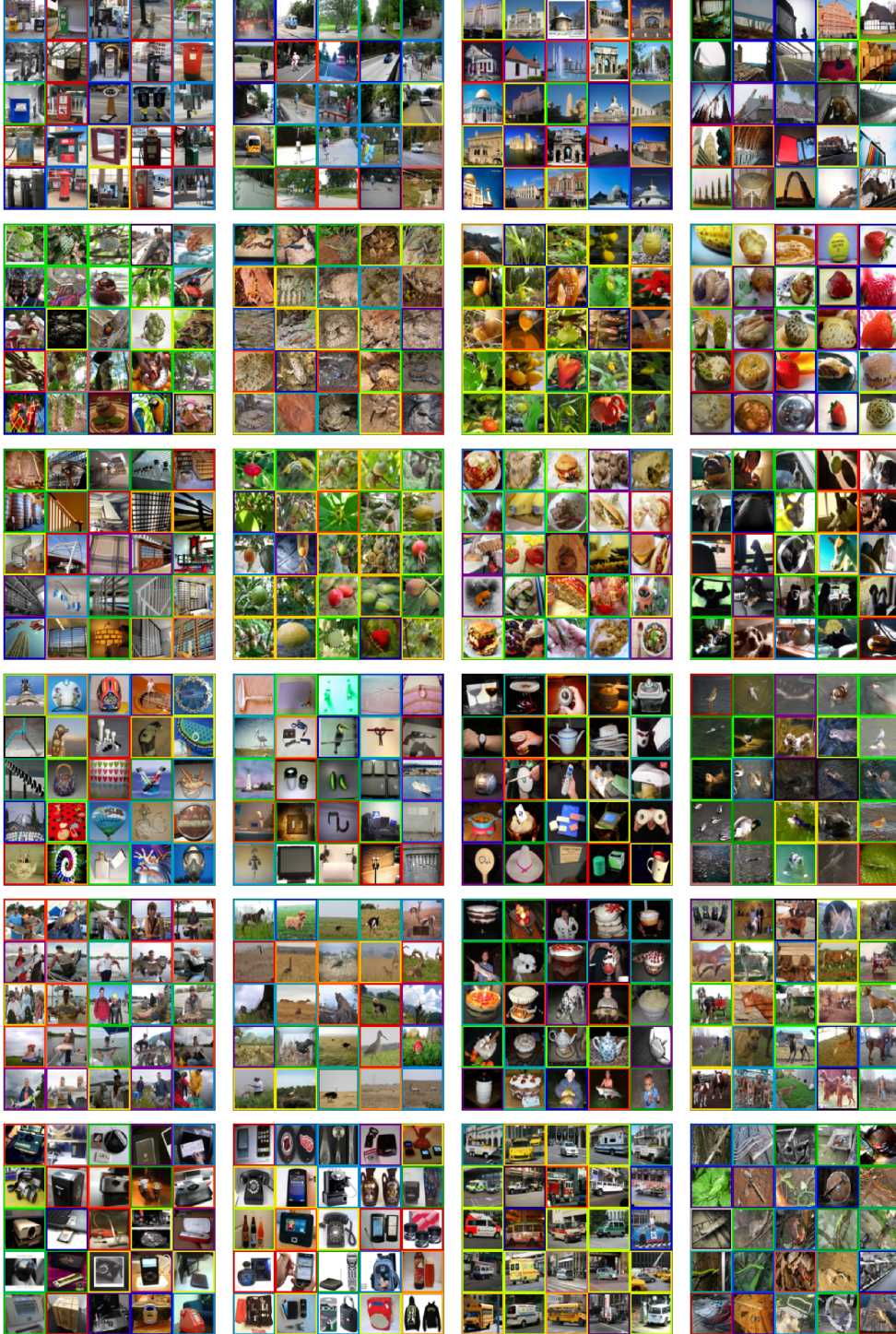


Figure A.7: Here we show a random sample of Imagenet training set images associated to the random pseudoclasses. The entropy is given by true image labels which are also shown as a frame around each picture with a random color. This visualization uses ResNet-50 $[3k \times 1]$.

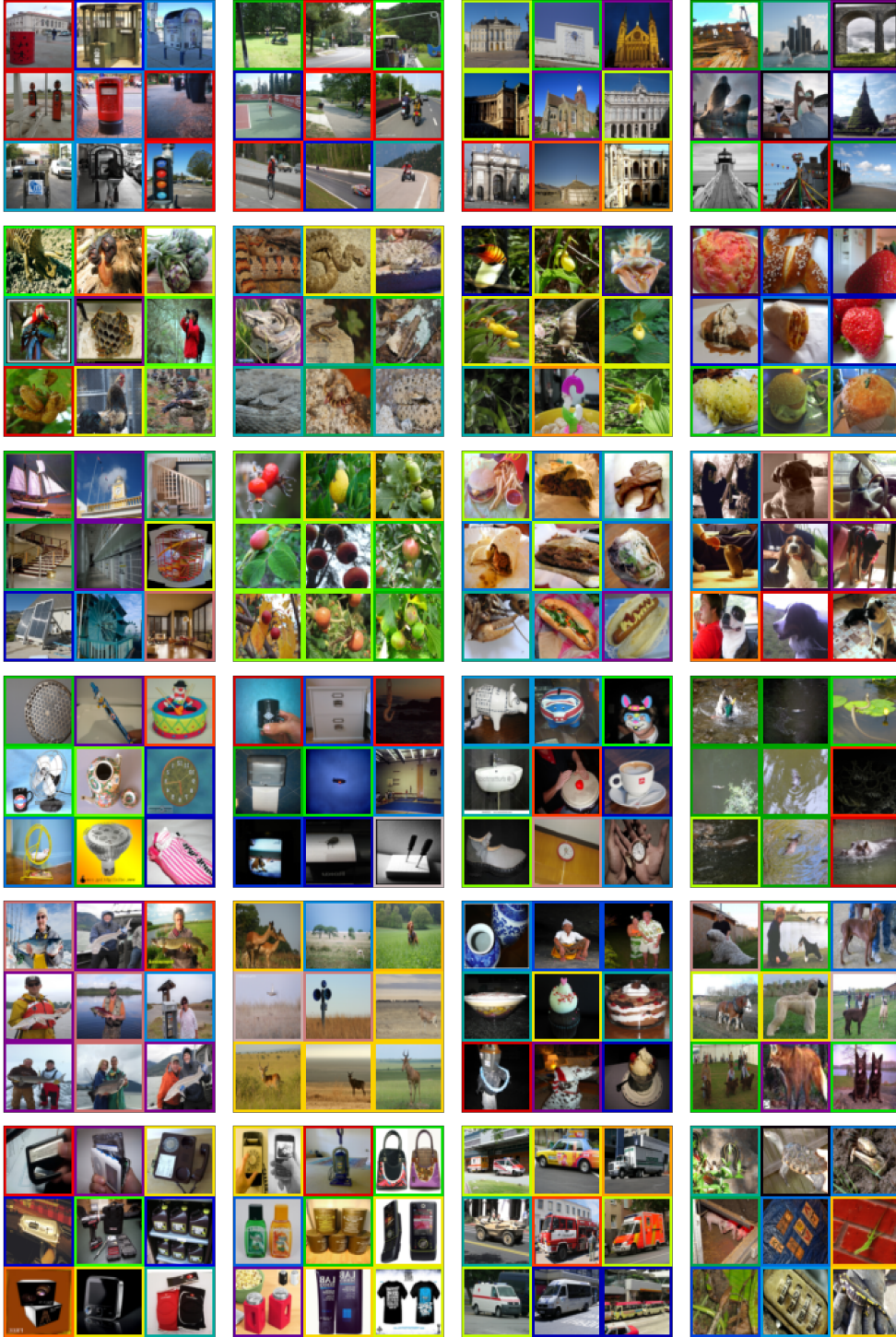


Figure A.8: Here we show a random sample of validation set images associated to random pseudo-classes. The entropy is given by true image labels which are also shown as a frame around each picture with a random color. This visualization uses ResNet-50 [3k × 1]. Classes with less than 9 images are sampled with repetition.

5

Labelling unlabelled videos from scratch with multi-modal self-supervision

This work was presented at the conference on Neural Information Processing Systems (NeurIPS) 2020.

Labelling unlabelled videos from scratch with multi-modal self-supervision

Yuki M. Asano^{1*} Mandela Patrick^{1,2*} Christian Rupprecht¹ Andrea Vedaldi^{1,2}

¹ Visual Geometry Group, University of Oxford

yuki@robots.ox.ac.uk

² Facebook AI Research
mandelapatrik@fb.com

Abstract

A large part of the current success of deep learning lies in the effectiveness of data – more precisely: labelled data. Yet, labelling a dataset with human annotation continues to carry high costs, especially for videos. While in the image domain, recent methods have allowed to generate meaningful (pseudo-) labels for unlabelled datasets without supervision, this development is missing for the video domain where learning feature representations is the current focus. In this work, we a) show that unsupervised labelling of a video dataset does not come for free from strong feature encoders and b) propose a novel clustering method that allows pseudo-labelling of a video dataset without any human annotations, by leveraging the natural correspondence between the audio and visual modalities. An extensive analysis shows that the resulting clusters have high semantic overlap to ground truth human labels. We further introduce the first benchmarking results on unsupervised labelling of common video datasets Kinetics, Kinetics-Sound, VGG-Sound and AVE².

1 Introduction

One of the key tasks in machine learning is to convert continuous perceptual data such as images and videos into a symbolic representation, assigning discrete labels to it. This task is generally formulated as *clustering* [31]. For images, recent contributions such as [6, 13, 37, 72] have obtained good results by combining clustering and representation learning. However, progress has been more limited for videos, which pose unique challenges and opportunities. Compared to images, videos are much more expensive to annotate; at the same time, they contain more information, including a temporal dimension and two modalities, aural and visual, which can be exploited for better clustering. In this paper, we are thus interested in developing methods to *cluster video datasets without manual supervision*, potentially reducing the cost and amount of manual labelling required for video data.

Just as for most tasks in machine learning, clustering can be greatly facilitated by extracting a suitable *representation* of the data. However, representations are usually learned by means of manually supplied labels, which we wish to avoid. Inspired by [79], we note that a solution is to consider one of the recent state-of-the-art self-supervised representation learning methods and apply an off-the-shelf clustering algorithm *post-hoc*. With this, we show that we can obtain very strong baselines for clustering videos.

*Joint first authors

²Code will be made available at <https://github.com/facebookresearch/selavi>

Still, this begs the question of whether even better performance could be obtained by *simultaneously learning to cluster and represent* video data. Our main contribution is to answer this question affirmatively and thus to show that *good clusters do not come for free from good representations*.

In order to do so, we consider the recent method SeLa [6], which learns clusters and representations for still images by solving an optimal transport problem, and substantially improve it to work with multi-modal data. We do this in three ways. First, we relax the assumption made in [6] that clusters are equally probable; this is not the case for semantic video labels, which tend to have a highly-skewed distribution [1, 29, 41], and extend the algorithm accordingly. Second, we account for the multi-modal nature of video data, by formulating the extraction of audio and visual information from a video as a form of data augmentation, thus learning a clustering function which is invariant to such augmentations. For this to work well, we also propose a new initialization scheme that synchronizes the different modalities before clustering begins. This encourages clusters to be more abstract and thus ‘semantic’ and learns a redundant clustering function which can be computed robustly from either modality (this is useful when a modality is unreliable, because of noise or compression). Third, since clustering is inherently ambiguous, we propose to learn multiple clustering functions in parallel, while keeping them orthogonal, in order to cover a wider space of valid solutions.

With these technical improvements, our method for Self-Labeling Videos (SeLaVi) substantially outperforms the post-hoc approach [79], SeLa [6] applied to video frames, as well as a recent multi-modal clustering-based representation learning method, XDC [2]. We evaluate our method by testing how well the automatically learned clusters match manually annotated labels in four different video datasets: VGG-Sound [17], AVE [68], Kinetics [41] and Kinetics-Sound [3]. We show that our proposed model results in substantially better clustering performance than alternatives. For example, our method can perfectly group 32% of the videos in the VGG-Sound dataset and 55% in the AVE dataset without using any labels during training. Furthermore, we show that, while some clusters do not align with the ground truth classes, they are generally semantically meaningful (e.g. they contain similar background music) and provide an interactive cluster visualization³.

In a nutshell, our key contributions are: **(i)** establishing video clustering benchmark results on four datasets for which labels need to be obtained in an unsupervised manner; **(ii)** developing and assessing several strong clustering baselines using state-of-the-art methods for video representation learning, and **(iii)** developing a new algorithm tailored to clustering multi-modal data resulting in state-of-the-art highly semantic labels.

2 Related work

Unsupervised labelling for images. Early approaches to clustering images include agglomerative clustering [9] and partially ordered sets of hand-crafted features [10], while more recent methods combine feature learning with clustering. First, there are methods which propose to implicitly learn a clustering function by maximizing mutual information between the image and nuisance transformations [35, 37]. Second, there are methods which use explicit clustering combined with representation learning [6, 13, 14, 16, 48, 77, 80]. Lastly, there are methods which build on strong feature representations and, at a second stage, utilize these to obtain clusters [47, 72, 79].

Representation learning from videos. There is a growing literature on representation learning from videos. Many of these methods are *uni-modal*, leveraging works from the image domain [5, 8, 18, 26, 27, 56, 57, 69, 75, 81], such as predicting rotations [39] and 3D jigsaw puzzles [42]. Other works leverage temporal information explicitly and predict future features [30], the order of frames [46, 53] and clips [78], the direction of time [74] or the framerate [11, 19]. However, videos usually contain multiple modalities, such as audio, speech and optical flow. *Multi-modal* learning, originally proposed by de Sa [22], has seen a resurgence with the goal of learning strong feature representations that can be used for finetuning on downstream tasks. Most works leverage audio-visual semantic correspondence [3, 7, 59, 60] or the synchronized timing of content [43, 58] between the audio and visual streams. Some works use this information to obtain within-clip sound localisation [4, 34, 58, 63, 64, 82] as well as audio-separation [15, 25]. Other methods use a modality distillation framework to learn video encoders from other modalities [59, 61]. In [61], a loss function is meta-learned by computing common self-supervised losses and distilling these and clustering is

³<https://www.robots.ox.ac.uk/~vgg/research/selavi>



Figure 1: **Our model** views modalities as different *augmentations* and produces a multi-modal clustering of video datasets from scratch that can closely match human annotated labels.

used as an evaluation metric for meta-learning. New methods have started to learn even stronger representations by using ASR generated text from videos as another modality [49, 52, 55, 66, 67].

Clustering videos. Perhaps the simplest way of combining representation learning and clustering in videos is to apply *post-hoc* a clustering algorithm after pretraining a representation. In Cluster-Fit [79], the authors show that running a simple k -means algorithm on the features from a pretrained network on the pretraining dataset yields small but consistent gains for representation learning when these clusters are used as labels and the networks are retrained. While in [79], the authors found the optimal number of clusters to consistently be at least one order of magnitude higher than the number of ground-truth labels, we investigate applying this method on various pretrained methods as baselines for our task of labelling an unlabelled video dataset. Specifically, we apply k -means on state-of-the-art single modality models such as DPC [30] and MIL-NCE [52], as well the multi-modal model XDC [2], which itself uses k -means on the audio and visual streams to learn representations. However, they do this as a pretext task for representation learning and obtain separate clusters for audio and video. In contrast, our goal is multi-modally labelling an unlabelled dataset, and we find that our method works significantly better at this task.

3 Method

Given a dataset $D = \{\mathbf{x}_i\}_{i \in \{1, \dots, N\}}$ of multi-modal data $\mathbf{x}_i$, our goal is to learn a labelling function $y(\mathbf{x}) \in \{1, \dots, K\}$ without access to any ground-truth label annotations. There are two requirements that the labelling function must satisfy. First, the labels should capture, as well as possible, the *semantic content* of the data, in the sense of reproducing the labels that a human annotator would intuitively associate to the videos. As part of this, we wish to account for the fact that semantic classes are not all equally probable, and tend instead to follow a Zipf distribution [1, 41]. We then evaluate the quality of the discovered labels by matching them to the ones provided by human annotators, using datasets where ground-truth labels are known. The second requirement is that the labelling method should not overly rely on a single modality. Instead, we wish to treat each modality *as equally informative* for clustering. In this way, we can learn a more robust clustering function, which can work from either modality. Furthermore, correlating of modalities has been shown to be a proxy to learn better abstractions [4, 43, 58, 60].

While our method can work with any number of data modalities (vision, audio, depth, textual transcripts, ...), we illustrate it under the assumption of video data $\mathbf{x} = (a, v)$, comprising an audio stream a and a visual stream v . The following two sections describe our method in detail and show how it meets our requirements.

3.1 Non-degenerate clustering via optimal transport

In this section, we briefly summarize the formulation of [6] to interpret clustering as an optimal transport problem. SeLa [6] is a method that learns representations via clustering images. The labelling function can be expressed as the composition $y(\Psi(\mathbf{x}))$, where $z = \Psi(\mathbf{x})$ is a data representation (i.e. a feature extractor implemented by a deep neural network), and $y(z) \in \{1, \dots, K\}$ operates on top of the features rather than the raw data.

Any traditional clustering algorithm, such as k -means or Gaussian mixture models, defines an energy function $E(y)$ that, minimized, gives the best data clustering function y . When the representation is accounted for, the energy $E(y, \Psi)$ is a function of both y and Ψ , and we may be naïvely tempted to

optimize over both. However, this is well known to yield unbalanced solutions, which necessitates ad-hoc techniques such as non-uniform sampling or re-initialization of unused clusters [13, 14]. Theoretically, in fact, for most choices of E , the energy is trivially minimized by the representation Ψ that maps all data to a constant.

Asano et al. [6] address this issue by constraining the marginal probability distributions of the clusters to be uniform, and show that this reduces to an optimal transport problem. The algorithm then reduces to alternating the fast Sinkhorn-Knopp algorithm [21] for clustering, and standard neural network training for representation learning. To do this, one introduces the cross-entropy loss $E(q, p)$, between the labels given as one-hot vectors in q (i.e. $q(y(\mathbf{x})) = 1 \forall \mathbf{x}$) and the softmax outputs p of a network Ψ :

$$E(p, q) = -\frac{1}{N} \sum_{i=1}^N \sum_{y=1}^K q(y|\mathbf{x}_i) \log p(y|\mathbf{x}_i), \quad p(y|\mathbf{x}_i) = \text{softmax } \Psi(\mathbf{x}_i), \quad (1)$$

where K is the number of clusters. This energy is optimized under the constraint that the marginal cluster probability $\sum_{i=1}^N \frac{1}{N} p(y|\mathbf{x}_i) = \frac{1}{K}$ is constant (meaning all clusters are a-priori equally likely). Note that minimizing E with respect to p is the same as training the deep network Ψ using the standard cross-entropy loss.

Next, we show that minimizing $E(p, q)$ w.r.t. the label assignments q results in an optimal transport problem. Let $P_{yi} = p(y|\mathbf{x}_i) \frac{1}{N}$ be the $K \times N$ matrix of joint probabilities estimated by the model and $Q_{yi} = q(y|\mathbf{x}_i) \frac{1}{N}$ be $K \times N$ matrix of assigned joint probabilities. Matrix Q is relaxed to be an element of a transportation polytope

$$U(r, c) := \{Q \in \mathbb{R}_+^{K \times N} \mid Q\mathbb{1} = r, Q^\top \mathbb{1} = c\}, \quad r = \mathbb{1}/K, \quad c = \mathbb{1}/N. \quad (2)$$

where $\mathbb{1}$ are vectors of ones, and r and c the marginal projections of matrix Q onto its clusters and data indices, respectively. Finally, optimizing $E(P, Q)$ w.r.t. to $Q \in U(r, c)$ is a linear optimal transport problem, for which [21] provides a fast, matrix-vector multiplication based solution.

3.2 Clustering with arbitrary prior distributions

A shortcoming of the algorithm just described is the assumption that all clusters are equally probable. This avoids converging to degenerate cases but is too constraining in practice since real datasets follow highly skewed distributions [1, 41], and even in datasets that are collected to be uniform, they are not completely so [17, 41, 68]. Furthermore, knowledge of the data distribution, for example long-tailedness, can be used as additional information (e.g. as in [61] for meta-learning) that can improve the clustering by allocating the right number of data points to each cluster. Next, we describe a mechanism to change this distribution arbitrarily.

In the algorithm above, changing the label prior amounts to choosing a different cluster marginal r in the polytope $U(r, c)$. The difficulty is that r is only known up to an arbitrary permutation of the clusters, as we do not know a-priori which clusters are more frequent and which ones less so. To understand how this issue can be addressed, we need to explicitly write out the energy optimised by the Sinkhorn-Knopp (SK) algorithm [21] to solve the optimal transport problem. This energy is:

$$\min_{Q \in U(r, c)} \langle Q, -\log P \rangle + \frac{1}{\lambda} \text{KL}(Q \| r c^\top), \quad (3)$$

where λ is a fixed parameter. Let $r' = Rr$ where R is a permutation matrix matching clusters to marginals. We then seek to optimize the same quantity w.r.t. R , obtaining the optimal permutation as $R^* = \text{argmin}_R E(R)$ where

$$E(R) = \langle Q, -\log P \rangle + \frac{1}{\lambda} \text{KL}(Q \| R r c^\top) = \text{const} + \sum_y -q(y) [R \log r]_y. \quad (4)$$

While there is a combinatorial number of permutation matrices, we show that minimizing Eq. (4) can be done by first sorting classes y in order of increasing $q(y)$, so that $y > y' \Rightarrow q(y) > q(y')$, and

then finding the permutation that R that also sorts $[R \log r]_y$ in increasing order.⁴ We conclude that R cannot be optimal unless it sorts all pairs. After this step, the SK algorithm can be applied using the optimal permutation R^* , without any significant cost (as solving for R is equivalent to sorting $\mathcal{O}(K \log K)$ with $K \ll N$). The advantage is that it allows to choose any marginal distribution, even highly unbalanced ones which are likely to be a better match for real world image and video classes than a uniform distribution.

3.3 Multi-modal single labelling

Next, we tackle our second requirement of extracting as much information as possible from multi-modal data. In principle, all we require to use the clustering formulation Eq. (1) with multi-modal data $\mathbf{x} = (a, v)$ is to design a corresponding multi-modal representation $\Psi(\mathbf{x}) = \Psi(a, v)$. However, we argue for *multi-modal single labelling* instead. By this, we mean that we wish to cluster data one modality at a time, but in a way that is modality agnostic. Formally, we introduce *modality splicing transformations* [60] $t_a(\mathbf{x}) = a$ and $t_v(\mathbf{x}) = v$ and use these as *data augmentations*. Recall that augmentations are random transformations t such as rotating an image or distorting an audio track that one believes should leave the label/cluster invariant. We thus require our activations used for clustering to be an average over augmentations by replacing matrix $\log P$ with

$$[\log P]_{yi} = \mathbb{E}_t[\log \text{softmax}_y \Psi(t\mathbf{x}_i)]. \quad (6)$$

If we consider splicing as part of the augmentations, we can learn clusters that are invariant to standard augmentations *as well as* the choice of modality. In practice, to account for modality splicing, we define and learn a pair $\Psi = (\Psi_a, \Psi_v)$ of representations, one per modality, resulting in the same clusters ($\Psi_a(t_a(\mathbf{x})) \approx \Psi_v(t_v(\mathbf{x}))$). This is illustrated in Figure 1.

Initialization and alignment. Since networks Ψ_a and Ψ_v are randomly initialized, at the beginning of training their output layers are *de-synchronized*. This means that there is no reason to believe that $\Psi_a(t_a(\mathbf{x})) \approx \Psi_v(t_v(\mathbf{x}))$ simply because the *order* of the labels in the two networks is arbitrary. Nevertheless, in many self-supervised learning formulations, one exploits the fact that even randomly initialized networks capture a useful data prior [71], which is useful to bootstrap learning.

In order to enjoy a similar benefit in our formulation, we propose to *synchronise* the two output layers of Ψ_a and Ψ_b before training the model. Formally, we wish to find the permutation matrix R that, applied to the last layer of one of the two encoders maximizes the agreement with the other (still leaving all the weights to their initial random values). For this, let W_a and W_v be the last layer weight matrices of the two networks,⁵ such as $\Psi_a(a) = W_a \bar{\Psi}_a(a)$ and $\Psi_v(v) = W_v \bar{\Psi}_v(v)$. We find the optimal permutation R by solving the optimisation problem:

$$\min_R \sum_{i=1}^N |\text{softmax}(RW_a \bar{\Psi}_a(t_a(\mathbf{x}_i))) - \text{softmax}(W_v \bar{\Psi}_v(t_v(\mathbf{x}_i)))|, \quad (7)$$

In order to compare softmax distributions, we choose $|\cdot|$ as the 1-norm, similar to [33]. We optimize Eq. (7) with a greedy algorithm: starting with a feasible solution and switching random pairs when they reduce the cost function [20], as these are quick to compute and we do not require the global minimum. Further details are given in Appendix A.3. With this permutation, the weight matrix of the last layer of one network can be resorted to match the other.

Decorrelated clustering heads. Conceptually, there is no single ‘correct’ way of clustering a dataset: for example, we may cluster videos of animals by their species, or whether they are taken indoor or outdoor. In order to alleviate this potential issue, inspired by [6, 37], we simply learn

⁴To see why this is optimal, and ignoring ties for simplicity, let R be any permutation and construct a permutation $\bar{R}$ by applying R and then by further swapping two labels $y > y'$. We can relate the energy of R and $\bar{R}$ as:

$$\begin{aligned} E(R) &= E(\bar{R}) + q(y)[\bar{R} \log r]_y + q(y')[\bar{R} \log r]_{y'} - q(y)[\bar{R} \log r]_{y'} - q(y')[\bar{R} \log r]_y \\ &= E(\bar{R}) + (q(y) - q(y'))([\bar{R} \log r]_y - [\bar{R} \log r]_{y'}). \end{aligned} \quad (5)$$

Since the first factor is positive by assumption, this equation shows that the modified permutation $\bar{R}$ has a lower energy than R if, and only if, $[\bar{R} \log r]_y > [\bar{R} \log r]_{y'}$, which means that $\bar{R}$ sorts the pair in increasing order.

⁵We assume that the linear layer biases are incorporated in the weight matrices.

multiple labelling functions y , using multiple classification heads for the network. We improve this scheme as follows. In each round of clustering, we generate two random augmentations of the data. Then, the applications of SK to half of the heads (at random) see the first version, and the other half the second version, thus increasing the variance of the resulting clusters. This increases the cost of the algorithm by only a small amount — as more time is used for training instead of clustering.

4 Experiments

The experiments are divided into three parts. First, in Section 4.1, we analyze the need for using both modalities when clustering and investigate the effect of our individual technical contributions via ablations and comparison to other approaches. Second, in Section 4.2, we demonstrate how our method achieves its stated goal of labelling a video dataset without human supervision. Third, in Section 4.3, we show that a side effect of our method is to learn effective audio-visual representations that can be used for downstream tasks *e.g.* video action retrieval, establishing a new state of the art.

Datasets. While the goal and target application of this work is to group unlabelled video datasets, for analysis purposes only, we use datasets that contain human annotated labels. The datasets range from small- to large-scale: The first is the recently released **VGG-Sound** [17], which contains around 200k videos obtained in the wild from YouTube with low labelling noise and covering 309 categories of general classes. The second dataset is **Kinetics-400** [41], which contains around 230k videos covering 400 human action categories. Third, we test our results on **Kinetics-Sound** proposed in [3], formed by filtering the Kinetics dataset to 34 classes that are potentially manifested visually and audibly, leading to 22k videos. Lastly, we use the small-scale **AVE Dataset** [68], originally proposed for audio-visual event localization and containing only around 4k videos. Among these, only VGG-Sound and Kinetics-400 are large enough for learning strong representations from scratch. We therefore train on these datasets and unsupervisedly finetune the VGG-Sound model on Kinetics-Sound and AVE.

Training details. Our visual encoder is a R(2+1)D-18 [70] network and our audio encoder is a ResNet [32] with 9 layers. For optimization, we use SGD for 200 epochs with weight decay of 10^{-5} and momentum of 0.9, further implementation details are provided in Appendix A.2.

Table 1: **Architectures and pretraining datasets.** We use state-of-the-art representation learning methods and combine pretrained representations with k -means as baselines in the Tables 5a to 5d.

Method	Input shape	Architecture	Pretrain dataset
Supervised	$32 \times 3 \times 112 \times 112$	R(2+1)D-18	Kinetics-400
DPC [30]	$40 \times 3 \times 224 \times 224$	R3D-34	Kinetics-400
MIL-NCE [52]	$32 \times 3 \times 224 \times 224$	S3D	HowTo100M
XDC [2]	$32 \times 3 \times 224 \times 224$	R(2+1)D-18	Kinetics-400

Baselines. To compare our method on this novel task of clustering these datasets, we obtained various pretrained video representations (DPC [30], MIL-NCE [52] and XDC [2]), both supervised⁶ and self-supervised (see Table 1 for details). For comparison, following [79], we run k -means on the global-average-pooled features, setting k to the same number of clusters as our method to ensure a fair comparison. For the k -means algorithm, we use the GPU accelerated version from the Faiss library [40].

Evaluation. We adopt standard metrics from the self-supervised and unsupervised learning literature: the *normalized mutual information* (NMI), the *adjusted rand index* (ARI) and *accuracy* (Acc) after matching of the self-supervised labels to the ground truth ones (for this we use the Kuhn–Munkres/Hungarian algorithm [45]). We also report the *mean entropy* and the *mean maximal purity* per cluster, defined in Appendix A.4, to analyze the qualities of the clusters. For comparability and interpretability, we evaluate the results using the ground truth number of clusters – which usually is unknown – but we find our results to be stable w.r.t. other number of clusters (see Appendix).

⁶The R(2+1)D-18 model from PyTorch trained on Kinetics-400 [41] from <https://github.com/pytorch/vision/blob/master/torchvision/models/video/resnet.py>.

Table 2: **The value of multi-modal understanding** is observed for obtaining a strong set of labels for VGG-Sound. Our method combines both modalities effectively to yield accuracies beyond a single modality and other methods.

Method			NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$
Random	✗	✓	10.2	4.0	2.2	4.9	3.5
Supervised	✗	✓	46.5	15.6	24.3	2.9	30.8
DPC	✗	✓	15.4	0.7	3.2	4.7	4.9
MIL-NCE	✗	✓	48.5	12.5	22.0	2.6	32.9
XDC	✗	✓	16.7	1.0	3.9	4.5	6.4
	✓	✗	14.0	0.8	2.9	4.6	4.4
	✓	✓	18.1	1.2	4.5	4.41	7.4
SeLaVi	✗	✓	52.8	19.7	30.1	2.6	35.6
	✓	✗	47.5	15.2	26.5	2.8	32.9
	✓	✓	56.7	22.5	32.3	2.4	38.0

Table 3: **Ablation** of multi-modality, Modality Alignment and Gaussian marginals. Decorrelated Heads. Models are evaluated at 100 epochs on the VGG-Sound dataset.

Method				MA?	G.?	DH?	NMI	ft.-Acc
(a) SeLa	✓	✗	✗	–	–	–	20.1	5.9
(b) Concat	✗	✓	✗	–	✗	✗	24.7	7.6
(c) SeLaVi	✗	✓	✗	✗	✗	✗	51.9	27.5
(d) SeLaVi	✗	✓	✗	✓	✓	✓	52.6	27.8
(e) SeLaVi	✗	✓	✓	✓	✗	✓	52.7	28.0
(f) SeLaVi	✗	✓	✓	✓	✓	✗	52.4	27.3
(g) SeLaVi	✗	✓	✓	✓	✓	✓	53.2	28.7

4.1 Technical Analysis

Multi-modality. In order to shed light on the nature of labelling a multi-modal dataset, we provide a detailed study of the use and combination of modalities in Table 2. While visual-only methods such as the Kinetics-400 supervisedly pretrained model, MIL-NCE or DPC cannot work when only audio-stream () is provided, we show the results for XDC and our method when a single or both modalities are provided. In particular, we find that even when only the visual-stream () is present at test-time, our method (57% NMI) already outperforms methods solely developed for representation learning, even surpassing the 100M videos with transcripts trained MIL-NCE (49% NMI). When only the audio-stream is used, our method’s performance drops only slightly, congruent with the balanced informativeness of both modalities in our method. Finally, when both modalities are used, we find that our method profits from both, such that the combined performance is significantly higher than the maximal performance of each single modality alone.

Degraded modality. In Fig. 2, we analyze how well our method fares when the quality of one modality is reduced. For this, we compress the video-stream by down and upsampling of the video resolution by factors 1, 4, 8 and 16 (details are provided in Appendix A.5). Even though our method has not been trained with compressed videos, we find that its performance degrades more gracefully than the baselines indicating it has learned to rely on both modalities.

Ablation. Next, we ablate the key parameters of our method and show how they each contribute to the overall clustering quality in Table 3. First, we show a baseline model in Table 3(a), when naïvely applying the publically available source-code for SeLa [6] on video frames () this yields a NMI of 20%. Compared to this frame-only method, in row (b), we find the results for concatenating the features of both modalities (followed by a single clustering head) to only lead to a small improvement upon the frame-only method with a NMI of 25%. Our method is shown in row (c), where we

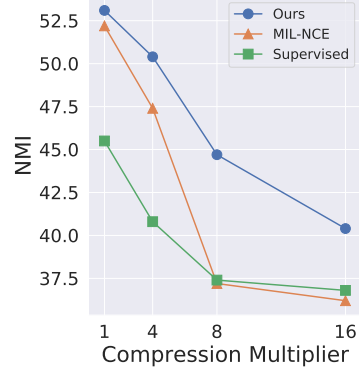


Figure 2: **Effective use of multi-modality** is found for our method when the visual input is compressed and decompressed.

Table 4: **Retrieval** via various number of nearest neighbors.

	Recall @	HMDB			UCF		
		1	5	20	1	5	20
3D-Puzzle [42]	–	–	–	–	19.7	28.5	40.0
OPN [46]	–	–	–	–	19.9	28.7	40.6
ST Order [12]	–	–	–	–	25.7	36.2	49.2
ClipOrder [78]	7.6	22.9	48.8	–	14.1	30.3	51.1
SpeedNet [11]	–	–	–	–	13.0	28.1	49.5
VCP [51]	7.6	24.4	53.6	–	18.6	33.6	53.5
VSP [19]	10.3	26.6	54.6	–	24.6	41.9	76.9
SeLaVi	24.8	47.6	75.5	52.0	68.6	84.5	–

Table 5: **Unsupervised labelling of datasets.** We compare labels from our method to labels that are obtained with k -means on the representations from a supervised and various unsupervised methods on four datasets.

(a) VGG-Sound.						(b) AVE.					
Method	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$	Method	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$
Random	10.2	4.0	2.2	4.9	3.5	Random	9.2	1.3	9.3	2.9	12.6
Supervised	46.5	15.6	24.3	2.9	30.8	Supervised	58.4	34.8	50.5	1.1	60.6
DPC	15.4	0.7	3.2	4.7	4.9	DPC	18.4	5.0	15.1	2.7	17.5
XDC	18.1	1.2	4.5	4.41	7.4	XDC	17.1	6.0	16.4	2.6	19.1
MIL-NCE	48.5	12.5	22.0	2.6	32.9	MIL-NCE	56.3	30.3	42.6	1.2	57.1
SeLaVi	56.7	22.5	32.3	2.4	38.0	SeLaVi	64.4	43.0	54.8	1.2	58.6

(c) Kinetics.						(d) Kinetics-Sound.					
Method	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$	Method	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$
Random	11.1	0.2	1.8	5.1	3.3	Random	2.8	0.5	5.9	3.3	8.3
Supervised	70.5	43.4	54.9	1.6	62.2	Supervised	81.7	66.3	75.0	0.5	85.4
DPC	16.1	0.6	2.7	4.9	3.9	DPC	8.8	2.2	9.6	3.1	13.6
XDC	17.2	0.8	3.4	4.7	6.2	XDC	7.5	1.9	9.4	3.1	13.6
MIL-NCE	48.9	12.5	23.5	2.7	33.7	MIL-NCE	47.5	24.0	37.8	1.5	51.0
SeLaVi	24.9	2.5	6.6	4.4	8.7	SeLaVi	50.2	31.4	43.2	1.7	48.1

find a substantial improvement with a NMI of 52%, i.e. a relative gain of more than 100%. While part of the gain comes from multi-modality, especially compared to row (b), the largest gain comes from the ability of our method in exploiting the natural correspondence provided in the multi-modal data. Finally, by ablating the technical improvements in rows (d)-(f) we find the strongest gain to be coming decorrelated heads, followed by the audio-visual modality alignment (MA) procedure, and that each improvement indeed benefits the model. To analyze the gains obtained by using multiple heads, we have also computed the average NMI between all pairs of heads as $(77.8 \pm 4\%)$. This means that the different heads do learn fairly different clusterings (as NMI takes permutations into account) whilst being at a similar distance to the ‘ground-truth’ $(53.1 \pm 0.1\%)$.

4.2 Unsupervised labelling audio-visual data

Table 5 shows the quality of the labels obtained automatically by our algorithm. We find that for the datasets VGG-Sound, Kinetics-Sound, and AVE, our method achieves state-of-the-art clustering performance with high accuracies of 32%, 43%, 55%, even surpassing the one of the strongest video feature encoder at present, the manually-supervised R(2+1)D-18 network. This result echoes the findings in the image domain [72] where plain k -means on representations is found to be less effective compared to learning clusters. For Kinetics-400, we find that the clusters obtained from our method are not well aligned to the human annotated labels. This difference can be explained by the fact that Kinetics is strongly focused on visual (human) actions and thus the audio is given almost no weighting in the annotation. We encourage exploring our interactive material, where our method finds clusters grouped by similar background music, wind or screaming crowds. We stress that such a grouping is *ipso facto* not wrong, only not aligned to this set of ground truth labels.

4.3 Labelling helps representation learning

Finally, we show how the visual feature representations unsupervisedly obtained from our method perform on downstream tasks. While not the goal of this paper, we test our representation on a standardized video action retrieval task in Table 4 and also provide results on video action classification in Table A.3, and refer to the Appendix for implementation details. We find that in obtaining strong labels, our method simultaneously learns robust, visual representations that can be used for other tasks without any finetuning and significantly improve the state of the art by over 100% for Recall @1 on UCF-101 and HMDB-51.

5 Conclusion

In this work, we have established strong baselines for the problem of unsupervised labelling of several popular video datasets; introduced a simultaneous clustering and representation learning approach for multi-modal data that outperforms all other methods on these benchmarks; and analysed the importance of multi-modality for this task in detail. We have further found that strong representations are not a sufficient criterion for obtaining good clustering results, yet, the strongest feature representations remain those obtained by supervised, *i.e.* well-clustered training. We thus expect the field of multi-modal clustering to be rapidly adopted by the research community who can build upon the presented method and baselines.

Broader Impact

We propose a method for clustering videos automatically. As such, we see two main areas of potential broader impact on the community and society as a whole.

Few-label harmful content detection. Our method clusters a video dataset into multiple sets of similar videos, as evidenced by the audio- and visual-stream and produces consistent, homogenous groupings. In practice, unsupervised clustering is especially useful for reducing the amount of data that human annotators have to label, since for highly consistent clusters only a single label needs to be manually obtained which can be propagated to the rest of the videos in the cluster. Using such an approach for the purpose of detecting harmful online content is especially promising. In addition, label-propagation might further lead to a beneficial reduction of type I errors (saying a video is safe when it is not). Furthermore, the multi-modality of our method allows it to potentially detect harmful content that is only manifested in one modality such as static background videos of harmful audio. Multi-modal harmful content detection has also been a subject of a recent data challenge that emphasizes insufficiency of using a single modality⁷. Lastly, the generality of our method allows it to also scale beyond these two modalities and in the future also include textual transcripts. Given the importance of this topic, it is also important to acknowledge, while less of a direct consequence, potential biases that can be carried by the dataset. Indeed models trained using our method will inherit the biases present in the dataset, which could be known but also unknown, potentially leading to propagation of biases without a clear way to analyze them, such as via labels. However, given the numerous pitfalls and failures when deploying computer vision systems to the real world, we believe that the positive impact of foundational research on public datasets, such as is presented in this paper, far outweighs these risks lying further downstream.

Overestimating clustering quality. The main benefit of our approach is to reduce the cost of grouping large collections of video data in a ‘meaningful’ way. It is difficult to think of an application where such a capability would lead directly to misuse. In part, this is due to the fact that better clustering results can generally be obtained by using some manual labels, so even where clustering videos could be misused, this probably would not be the method of choice. Perhaps the most direct risk is that a user of the algorithm might overestimate its capabilities. Clustering images is sometimes done in critical applications (e.g. medical science [36, 38, 50]). Our method clusters data based on basic statistical properties and the inductive prior of convolutional neural networks, without being able to tap into the deep understanding that a human expert would have of such domain expertise. Hence, the clusters determined by our method may not necessarily match the clusters an expert would make in a particular domain. Further, as the method is unsupervised, it may learn to exploit biases present in the data that might not be desired by the users. While we believe it has potential to be broadly applied after being finetuned to a specific domain, at present, our method is a better fit for applications such as indexing personal video collections where clustering ‘errors’ can be tolerated.

Acknowledgments and Disclosure of Funding

We are grateful for support from the Rhodes Trust (M.P.), Qualcomm Innovation Fellowship (Y.A.) and EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems

⁷Hateful memes challenge: <https://www.drivendata.org/competitions/64/hateful-memes/>

[EP/L015897/1] (M.P. and Y.A.). M.P. funding was received under his Oxford affiliation. C.R. is supported by ERC IDIU-638009. We thank Weidi Xie and Xu Ji from VGG for fruitful discussions.

References

- [1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.
- [2] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *arXiv preprint arXiv:1911.12667*, 2019.
- [3] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017.
- [4] Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.
- [5] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *ICLR*, 2020.
- [6] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.
- [7] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.
- [8] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views, 2019.
- [9] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Bjorn Ommer. Cliqecnn: Deep unsupervised exemplar learning. In *NeurIPS*, pages 3846–3854, 2016.
- [10] Miguel A Bautista, Artsiom Sanakoyeu, and Bjorn Ommer. Deep unsupervised similarity learning using partially ordered sets. In *CVPR*, pages 7130–7139, 2017.
- [11] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos, 2020.
- [12] Uta Buchler, Biagio Brattoli, and Bjorn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 770–786, 2018.
- [13] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- [14] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, 2019.
- [15] Anna Llagostera Casanovas, Gianluca Monaci, Pierre Vanderghenst, and Rémi Gribonval. Blind audiovisual source separation based on sparse redundant representations. *IEEE Transactions on Multimedia*, 12(5):358–371, 2010.
- [16] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep adaptive image clustering. In *ICCV*, pages 5879–5887, 2017.
- [17] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. *ICASSP*, May 2020.
- [18] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.
- [19] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatiotemporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.
- [20] Nicos Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group, 1976.
- [21] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, pages 2292–2300, 2013.
- [22] Virginia R. de Sa. Learning classification with unlabeled data. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *NeurIPS*, pages 112–119. Morgan-Kaufmann, 1994.
- [23] Thomas Feo and Mauricio Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6:109–133, 03 1995. doi: 10.1007/BF01096763.
- [24] Paola Festa and Mauricio GC Resende. An annotated bibliography of grasp. *Operations Research Letters*, 8:67–71, 2004.
- [25] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *ECCV*, pages 35–53, 2018.

- [26] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.
- [27] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words, 2020.
- [28] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyröla, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.
- [29] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, pages 6047–6056, 2018.
- [30] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV*, 2019.
- [31] John A Hartigan. Direct clustering of a data matrix. *Journal of the american statistical association*, 67(337):123–129, 1972.
- [32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [33] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- [34] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *CVPR*, pages 9248–9257, 2019.
- [35] Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. Learning discrete representations via information maximizing self-augmented training. In *ICML*, pages 1558–1567, 2017.
- [36] Dimitris K Iakovidis, Spiros V Georgakopoulos, Michael Vasilakakis, Anastasios Koulaouzidis, and Vassilis P Plagianakos. Detecting and locating gastrointestinal anomalies using deep learning and iterative cluster unification. *IEEE transactions on medical imaging*, 37(10):2196–2210, 2018.
- [37] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.
- [38] Yizhang Jiang, Kaifa Zhao, Kaijian Xia, Jing Xue, Leyuan Zhou, Yang Ding, and Pengjiang Qian. A novel distributed multitask fuzzy clustering algorithm for automatic mr brain image segmentation. *Journal of medical systems*, 43(5):118, 2019.
- [39] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.
- [40] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.
- [41] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.
- [42] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.
- [43] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.
- [44] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.
- [45] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955.
- [46] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.
- [47] Juho Lee, Yoonho Lee, and Yee Whye Teh. Deep amortized clustering. *arXiv preprint arXiv:1909.13433*, 2019.
- [48] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.
- [49] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination, 2020.
- [50] Zhuoling Li, Minghui Dong, Shiping Wen, Xiang Hu, Pan Zhou, and Zhigang Zeng. Clu-cnns: Object detection for medical images. *Neurocomputing*, 350:53–59, 2019.

- [51] Dezhao Luo, Chang Liu, Yu Zhou, Dongbao Yang, Can Ma, Qixiang Ye, and Weiping Wang. Video cloze procedure for self-supervised spatio-temporal learning. In *AAAI*, 2020.
- [52] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos, 2019.
- [53] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.
- [54] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement, 2020.
- [55] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.
- [56] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.
- [57] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, 2017.
- [58] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multi-sensory features. In *ECCV*, 2018.
- [59] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.
- [60] Mandela Patrick, Yuki M. Asano, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations, 2020.
- [61] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning, 2020.
- [62] Mauricio GC Resende and Celso C Ribeiro. Greedy randomized adaptive search procedures: Advances and extensions. In *Handbook of metaheuristics*, pages 169–220. Springer, 2019.
- [63] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *ICASSP*, 2019.
- [64] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. *CVPR*, Jun 2018.
- [65] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.
- [66] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019.
- [67] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7464–7473, 2019.
- [68] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018.
- [69] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.
- [70] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.
- [71] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *arXiv preprint arXiv:1711.10925*, 2017.
- [72] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision (ECCV)*, 2020.
- [73] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, 2019.
- [74] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.
- [75] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, June 2018.
- [76] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.
- [77] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016.
- [78] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.

- [79] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Cluster-Fit: Improving Generalization of Visual Representations. In *CVPR*, 2020.
- [80] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *CVPR*, pages 5147–5156, 2016.
- [81] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.
- [82] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, pages 1735–1744, 2019.

A Appendix

A.1 Pretrained model details

Here we provide additional information about the pretrained models we have used in this work.

Table A.1: **Details for audio encoder.** Architectural and pretraining details for XDC’s audio encoder used for benchmarking.

Method	Input shape	Architecture	Pretrain dataset
XDC	$40 \times 1 \times 100$	Resnet-18	Kinetics-400

A.2 Implementation details

We train our method using the Sinkhorn-Knopp parameter $\lambda = 20$, an inverse quadratic clustering schedule with 100 clustering operations and 10 heads which we adopt from [6]. For evaluation, we report results for head 0 to compare against the ground-truth, as we found no significant difference in performance between heads. For the Gaussian distribution, we take the marginals to be from $\mathcal{N}(1, 0.1) * N/K$. For the clustering-heads, we use two-layer MLP-heads as in [8, 18]. The video inputs are 30 frame long clips sampled consecutively from 30fps videos and are resized such that the shorter side is 128 and during training a random crop of size 112 is extracted, no color-jittering is applied. Random horizontal flipping is applied to the video frames with probability 0.5, and then the channels of the video frames are Z-normalized using mean and standard deviation statistics computed across the dataset. The audio is processed as a $1 \times 257 \times 199$ image, by taking the log-mel bank features with 257 filters and 199 time-frames and for training, random volume jittering between 90% and 110% is applied to raw waveform, similar to [54]. For evaluation, a center-crop is taken instead for the video inputs and audio volume is not jittered. We use a mini-batch size of 16 on each of our 64 GPUs giving an effective batch size of 1024 for distributed training for 200 epochs. The initial learning rate is set to 0.01 which we linearly scale with the number of GPUs, after following a gradual warm-up schedule for the first 10 epochs [28]. For training on Kinetics-Sound and AVE, we initialize our model with a VGG-Sound pretrained backbone due to the small training set sizes ($N = 22k$ and $N = 3328$). The clustering heads are re-initialized randomly. This ensures a more fair comparison as XDC, DPC and the supervised model are pretrained on Kinetics-400 with $N = 230k$ and MIL-NCE on HowTo100M with $N = 100M$ videos. We train on VGG-Sound for 200 epochs, which takes around 2 days.

A.3 Pair-based optimization for AV-Alignment

For aligning the visual and audio encoder, we use a greedy switching algorithm that starts from a feasible initial solution [23, 24, 62]. In particular, we consider 50000 potential pair switches with 5 randomized restarts and take the final permutation that yields the lowest cost.

A.4 Evaluation metrics details

The **normalized mutual information** (NMI) is calculated by the formula

$$\text{NMI} = \frac{\text{MI}(U, V)}{0.5H(U) + 0.5H(V)}, \quad (8)$$

where the Mutual information MI is given by $\text{MI}(U, V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i, j) \log \left(\frac{P(i, j)}{P(i)P(j)} \right)$, and H is the standard entropy, with $H(U) = -\sum_{i=1}^{|U|} P(i) \log(P(i))$. The NMI ranges from 0 (no mutual information) to 100%, which implies perfect correlation.

The rand index (RI) is given by $\text{RI} = \frac{a+b}{C}$, where a, b are the number of pairs of elements that are in the same/different set in the ground truth labelling and in the same/different set in the predicted

clustering and C is the total number of such pairs. The **adjusted Rand index** (ARI) corrects for random assignments and is given by

$$\text{ARI} = \frac{\text{RI} - E[\text{RI}]}{\max(\text{RI}) - E[\text{RI}]}, \quad (9)$$

where the expected RI of a random label assignment is subtracted in both nominator and denominator. Due to the subtraction, the ARI varies from -1 to 1 with a value close to 0 implying random correlation and a value of 1 implying identical agreement.

The **mean entropy per cluster** is given by

$$\langle H \rangle = \frac{1}{K} \sum_{k \in K} H(p(y|\hat{y}_k = k)), \quad (10)$$

where $\hat{y}$ are unsupervisedly obtained clusters and $p(y|\hat{y}_k = k)$ is the distribution of ground-truth clusters for cluster k . Hence, the optimal number of this metric is 0 and a chance assignment yields $\langle H \rangle = -\log 1/K$.

Further, as we wish to understand the semantic purity compared to the ground truth labels of each cluster, so we additionally report the **mean maximal purity per cluster**,

$$\langle p_{\max} \rangle = \frac{1}{K} \sum_{k \in K} \max(p(y|\hat{y}_k = k)), \quad (11)$$

which ranges from $\langle p_{\max} \rangle = 1/K$ (chance level) to perfect matching at $\langle p_{\max} \rangle = 100\%$.

A.5 Single modality degradation experiment details

We use the default input-sizes for each model, i.e. 112 for ours and the supervised model, 224 for MIL-NCE. Compression is implemented by nearest-neighbor downsampling and subsequently nearest-neighbor upsampling for speed. For this experiment only, we evaluate the performance on the smaller validation sets.

A.6 Further ablations

In Table A.2, we provide the results for varying the number of clusters K in our algorithm. We find that even when moving from the ground-truth number of classes ($K = 309$), to lower numbers ($K = 256$) or higher estimates ($K = 619, 1024$) our results remain stable with the NMI staying almost constant. While the ARI does drop for larger K , we also observe an increase in the purity of the clusters for a larger number of clusters from $\langle p_{\max} \rangle = 38.0$ for $K = 309$ to $\langle p_{\max} \rangle = 42.7$ for $K = 619$, which can be particularly useful when dividing the dataset into clusters and subsequently only obtaining human annotations for few examples per cluster.

Table A.2: **Varying K** in our method degrades performances only slightly, showing that our method is robust to various estimations of the ground-truth number of classes. Results on VGG-Sound.

Method	K	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{\max} \rangle$
SeLaVi	309	56.7	22.5	32.3	2.4	38.0
SeLaVi	256	56.8	24.3	34.2	2.4	36.9
SeLaVi	619	56.9	16.8	23.0	2.2	42.7
SeLaVi	1024	55.1	16.3	9.6	2.1	42.2

A.7 Retrieval downstream task implementation details

We follow [78] in our evaluation protocol and use split 1 of UCF101 and HMDB-51. We uniformly sample 10 clips per video, and average the max-pooled features after the last residual block for each clip per video. We then utilize the averaged features from the validation set to query the videos in the training set. The cosine distance of representations between the query clip and all clips in the

training set are computed and when the class of a test clip appears in the classes of k nearest training clips, it is considered to be correctly retrieved. $R@k$ refers to the retrieval performance using k nearest neighbors.

A.8 Visual classification downstream task

Table A.3: **Representation learning downstream evaluation.** Self-supervised and fully-supervised trained methods on UCF101 and HMDB51 benchmarks. We follow the standard protocol and report the average top-1 accuracy over the official splits and show results for finetuning the whole network. Methods with [†] indicate the additional use of video titles and ASR generated text as supervision. Methods with * use ASR generated text.

Method	Architecture	Pretrain Dataset	Top-1 Acc%	
			UCF	HMDB
Full supervision [2]	R(2+1)D-18	ImageNet	82.8	46.7
Full supervision [2]	R(2+1)D-18	Kinetics-400	93.1	63.6
Weak supervision, CPD [49] [†]	3D-Resnet50	Kinetics-400	88.7	57.7
MotionPred [73]	C3D	Kinetics-400	61.2	33.4
RotNet3D [39]	3D-ResNet18	Kinetics-600	62.9	33.7
ST-Puzzle [42]	3D-ResNet18	Kinetics-400	65.8	33.7
ClipOrder [78]	R(2+1)D-18	Kinetics-400	72.4	30.9
DPC [30]	3D-ResNet34	Kinetics-400	75.7	35.7
CBT [66]	S3D	Kinetics-600	79.5	44.6
Multisensory [58]	3D-ResNet18	Kinetics-400	82.1	-
XDC [2]	R(2+1)D-18	Kinetics-400	84.2	47.1
AVTS [43]	MC3-18	Kinetics-400	85.8	56.9
AV Sync+RotNet [76]	AVSlowFast	Kinetics-400	87.0	54.6
GDT [60]	R(2+1)D-18	Kinetics-400	<u>88.7</u>	<u>57.8</u>
SeLaVi	R(2+1)D-18	Kinetics-400	83.1	47.1
SeLaVi	R(2+1)D-18	VGG-Sound	87.7	53.1

In Table A.3 we show the performance of our method on two common visual-only video feature representation benchmarks, UCF-101 [65] and HMDB-51 [44]. Note that, as is the standard in this evaluation, we use our visual encoder as initialization and fine-tune the whole network on the target down-stream task. In particular, we follow the finetuning schedule of the one of the current state-of-the-art methods [60]. We find that we achieve competitive performance when trained on VGG-Sound, even surpassing XDC, despite our method using only a spatial resolution of 112×112 and not 224×224 .

6

Support-set bottlenecks for video-text representation learning

This work was presented at the International Conference on Learning Representations (ICLR) 2021 as a spotlight.

SUPPORT-SET BOTTLENECKS FOR VIDEO-TEXT REPRESENTATION LEARNING

Mandela Patrick*, Po-Yao Huang*, Florian Metze & Andrea Vedaldi

Facebook AI

{mandelapatt, berniehuang, fmetze, vedaldi}@fb.com

Alexander Hauptmann

Language Technologies Institute

Carnegie Mellon University

alex@cs.cmu.edu

Yuki M. Asano* & João Henriques

Visual Geometry Group

University of Oxford

{yuki, joao}@robots.ox.ac.uk

ABSTRACT

The dominant paradigm for learning video-text representations – noise contrastive learning – increases the similarity of the representations of pairs of samples that are known to be related, such as text and video from the same sample, and pushes away the representations of all other pairs. We posit that this last behaviour is too strict, enforcing dissimilar representations even for samples that are semantically-related – for example, visually similar videos or ones that share the same depicted action. In this paper, we propose a novel method that alleviates this by leveraging a generative model to naturally push these related samples together: each sample’s caption must be reconstructed as a weighted combination of other support samples’ visual representations. This simple idea ensures that representations are not overly-specialized to individual samples, are reusable across the dataset, and results in representations that explicitly encode semantics shared between samples, unlike noise contrastive learning. Our proposed method outperforms others by a large margin on MSR-VTT, VATEX, ActivityNet, and MSVD for video-to-text and text-to-video retrieval.

1 INTRODUCTION

Noise contrastive learning (Gutmann & Hyvärinen, 2010) is emerging as one of the best approaches to learn data representations both for supervised (Khosla et al., 2020) and unsupervised regimes (Chen et al., 2020c). The idea is to learn a representation that discriminates any two data samples while being invariant to certain data transformations. For example, one might learn a representation that identifies a specific image up to arbitrary rotations (Misra & van der Maaten, 2020). In a multi-modal setting, the transformations can separate different modalities, for example, by extracting the audio and visual signals from a video. The resulting noise contrastive representation associates audio and visual signals that come from the same source video, differentiating others (Patrick et al., 2020).

The noise contrastive approach is motivated by the fact that the transformations that are applied to the data samples leave their ‘meaning’ unchanged. For example, rotating an image does not change the fact that it contains a cat or not (Gidaris et al., 2018). However, in most cases, we expect to find many data samples that share the same content without being necessarily related by simple transformations (e.g. think of any two images of cats). Existing noise contrastive formulations are unaware of these relationships and still try to assign different representations to these samples (Wu et al., 2018), despite the fact that they are semantically equivalent. If the representation is learned for a downstream task such as semantic video retrieval, this might degrade performance.

This suggests that there might be other learning signals that could complement and improve pure contrastive formulations. In this paper, we explore this idea in the case of learning from two modali-

*Joint first authors.

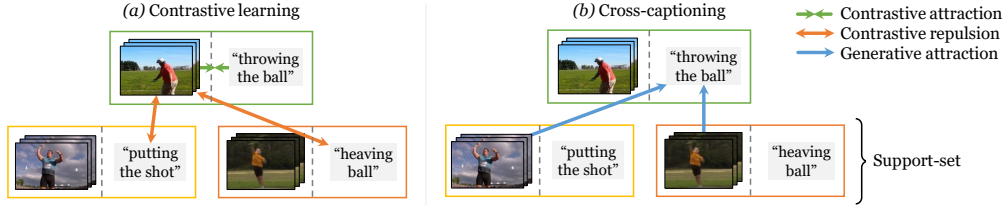


Fig. 1: **Cross-modal discrimination and cross-captioning.** Our model learns from two complementary losses: (a) Cross-modal contrastive learning learns strong joint video-text embeddings, but every other sample is considered a negative, pushing away even semantically related captions (orange arrows). (b) We introduce a generative task of cross-captioning, which alleviates this by learning to reconstruct a sample’s text representation as a weighted combination of a support-set, composed of video representations from other samples.

ties: videos and text, in the form of video transcripts or captions. Given a state-of-the-art contrastive formulation that learns from these two modalities, we investigate complementary pretext objectives to improve it. First, we consider the (*instance*) *captioning* task, namely mapping a video to the corresponding text, casting this as a conditional stochastic text generation problem. We show that this brings only a modest benefit.

We observe that the captioning task is highly sample-specific, as the goal is to produce a caption which describes a specific video and not any other video, and thus it suffers from the same disadvantages (discouraging concept sharing among samples) as contrastive learning. Thus, we propose to address this issue by switching to a different text generation task. The idea is to modify the text generator to take as input a learnable mixture of a support-set of videos, which we call *cross-instance captioning*. The mixture weights are generated by comparing the learned video representations to captions’ representations in an online way over the batch. The limited set of support samples acts as a bottleneck that encourages extraction of shared semantics. In this manner, the embeddings can associate videos that share similar captions even if the contrastive loss tries to push them apart.

We show that, when the captioning task is added in this manner, it brings a sensible improvement to already very strong video representation learning results, further improving our own state-of-the-art baseline by a significant margin.

2 RELATED WORKS

Learning data representations from unlabelled data has been a long standing goal of machine learning. These approaches are called “self-supervised learning” because the learning signals, termed pretext tasks, are obtained from the data itself. In the image and video domain, pretext tasks include colorization (Zhang et al., 2016), rotation (Gidaris et al., 2018), or clustering (Asano et al., 2020a;b; Caron et al., 2018; Ji et al., 2018), while in the natural language domain, masked language modeling (Devlin et al., 2019), and next word prediction (Mikolov et al., 2013; Pennington et al., 2014) are extremely popular. These pretext tasks can be broadly classified into two classes: generative and discriminative.

Discriminative approaches learn representations by differentiating input samples, using objectives such as the contrastive loss (Gutmann & Hyvärinen, 2010; Hadsell et al., 2006). Discriminative approaches have proven to be particularly successful for image (Chen et al., 2020c; He et al., 2020; Misra & van der Maaten, 2020; Wu et al., 2018) and video (Han et al., 2019; Morgado et al., 2020; Patrick et al., 2020) representation learning. Generative approaches, on the other hand, try to reconstruct its input. GANs (Donahue & Simonyan, 2019; Goodfellow et al., 2014; Radford et al., 2015), autoencoders (Hinton & Salakhutdinov, 2006) and sequence-to-sequence models (Sutskever et al., 2014) are popular generative models. In this work, we show the importance of combining both discriminative and generative objectives to learn effective video-text representations.

The success of representation learning has also been due to advances in model architectures, such as the Transformer (Vaswani et al., 2017). BERT (Devlin et al., 2019) demonstrated that a transformer architecture pretrained on large-scale textual data can learn transferable text representations that can

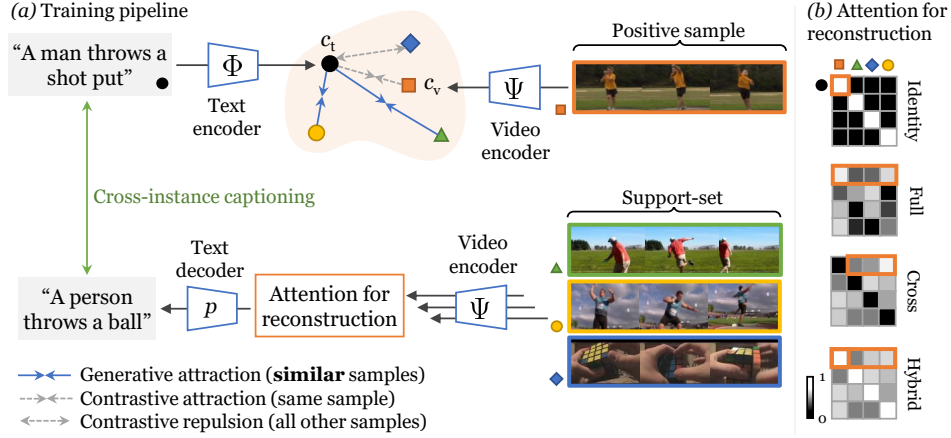


Fig. 2: **(a)** Our cross-modal framework with the discriminative (contrastive) objective and the generative objective. The model learns to associate video-text pairs in a common embedding space with text and video encoders (top). Meanwhile, the text must also be reconstructed as a weighted combination of video embeddings from a support-set (bottom), selected via attention, which enforces representation sharing between different samples. **(b)** Weights matrices (attention maps) used in each cross-captioning objective (see section 3.1.2).

be fine-tuned on a variety of downstream tasks. Subsequent works (Clark et al., 2020; Lewis et al., 2020a;b; Radford et al., 2019; Raffel et al., 2019) have improved upon the transformer architecture or training objective to learn even better representations. Inspired by the success of transformers in the NLP domain, several works have leveraged transformers to learn transferable image (Chen et al., 2020a; Desai & Johnson, 2020; Sariyildiz et al., 2020) or multi-modal image-text representations (Chen et al., 2019; Li et al., 2020a; 2019; Lu et al., 2019; Su et al., 2019; Tan & Bansal, 2019). In this work, we leverage the transformer architecture to better encode and represent text and video.

Large-scale training data has enabled the more effective pretraining of image (Sun et al., 2017; Yalniz et al., 2019), video (Ghadiyaram et al., 2019; Thomee et al., 2016) and textual representations (Raffel et al., 2019). The release of the HowTo100M dataset (Miech et al., 2019), a large-scale instructional video dataset, has spurred significant interest in leveraging large-scale pretraining to improve video-text representations for tasks such as video question-answering (Lei et al., 2018), text-video retrieval (Liu et al., 2019) and video captioning (Zhou et al., 2018b) on smaller datasets such as YouCookII (Zhou et al., 2018a), MSVD (Venugopalan et al., 2015a), MSR-VTT (Xu et al., 2016), LSMDC (Rohrbach et al., 2017), DiDeMo (Hendricks et al., 2018) and ActivityNet (Krishna et al., 2017). Although semantically rich and diverse, instructional videos from the web are super noisy and therefore a few approaches have been proposed to combat this. A few works (Luo et al., 2020; Sun et al., 2019a;b; Zhu & Yang, 2020) extend the BERT model to accept both visual and textual tokens to learn high-level semantic video-text representations. Other works have leveraged the contrastive loss (Miech et al., 2020) and show that using the raw audio (Alayrac et al., 2020; Rouditchenko et al., 2020) and other modalities (Gabeur et al., 2020) can be used to better align and improve video-text representations. While all these approaches rely on a contrastive objective, VidTranslate (Korbar et al., 2020) shows that a generative objective can also be used to learn joint video-text representations. In contrast to Korbar et al. (2020), we show that combining contrastive and generative objectives to pre-train video-text representations on large-scale data such as HowTo100M is very effective. The generative objective serves as regularizer to mitigate the strictness of the instance discrimination task of the contrastive objective, showing benefits similar to approaches such as clustering (Caron et al., 2020; Li et al., 2020b) and feature mixing (Kalantidis et al., 2020) which have been applied in the image domain.

3 METHOD

We consider the problem of learning multimodal representations from a corpus $\mathcal{C}$ of video-text pairs (v, t) , where v is a video and t is its corresponding text (caption or transcription). Our goal is to

learn a pair of representation maps $c_v = \Psi(v)$ and $c_t = \Phi(t)$, with outputs in a d -dimensional embedding space $c_v, c_t \in \mathbb{R}^d$, where semantically similar instances are close to each other.

3.1 OBJECTIVE FOR LEARNING MULTIMODAL REPRESENTATIONS

We consider two learning objectives, also illustrated in Figure 1. The first is the contrastive objective, pushing embeddings c_t and c_v to be close if text t and video v come from the same sample and pushing them apart otherwise. This assumes that every sample is its own class and does not benefit from modelling similarities *across* instances. The second objective is generative captioning. In its most basic variant, it maximizes the probability of generating the text t given the corresponding video v . However, we suggest that variants that explicitly promote concept sharing between instances will result in better downstream performance, in tasks such as video retrieval. These variants, illustrated in Figure 2, have in common that the caption t is reconstructed from a learned weighted combination over *other* videos $\hat{v}$. This is a form of attention (Bahdanau et al., 2014) which encourages the network to learn about which videos share similar semantics, compensating for the contrastive loss and grouping them implicitly.

In the following, we denote with $\mathcal{B} \subset \mathcal{C}$ a *batch* of multi-modal samples, i.e. a finite collection of video-text pairs $(t, v) \in \mathcal{C}$. For simplicity, we denote the batch as $\mathcal{B} = \{(t^i, v^i)\}_{i=1}^B$.

3.1.1 CONTRASTIVE OBJECTIVE

To define the contrastive objective, let $s(a, b) = \frac{a^\top b}{\|a\| \|b\|}$ be the similarity measure between vectors a and b . Following Faghri et al. (2018), we adopt the hinge-based triplet ranking loss with hard negative mining:

$$\mathcal{L}^{\text{contrast}} = \frac{1}{B} \sum_{i=1}^B \left[\max_j [\alpha - s(c_t^i, c_v^i) + s(c_t^i, c_v^j)]_+ + \max_j [\alpha - s(c_t^i, c_v^i) + s(c_t^j, c_v^i)]_+ \right], \quad (1)$$

where α is the correlation margin between positive and negative pairs and $[\cdot]_+ = \max\{0, \cdot\}$ is the hinge function. In our experiments, we set $\alpha = 0.2$.

3.1.2 CROSS-CAPTIONING OBJECTIVES

In the conventional captioning, the decoder seeks to optimize the negative log-likelihood of a text sequence t given its corresponding video v :

$$\mathcal{L}^{\text{caption}} = -\frac{1}{B} \sum_{i=1}^B \log p(t^i | e_v^i). \quad (2)$$

Here, the log-likelihood is obtained via auto-regressive decoding (Vaswani et al., 2017) from an intermediate video embedding $e_v^i = \Phi'(v^i)$. For the cross-captioning objective, we modify this loss to condition the generation process on a weighted average of the embeddings of the *other* videos in the batch, which we call the *support-set*. The weights themselves, which can be interpreted as a batch-wise attention, are obtained as a softmax distribution with temperature T over batch indices based on the video embeddings, as follows:

$$\mathcal{L}^{\text{cross-captioning}} = -\frac{1}{B} \sum_{i=1}^B \log p(t^i | \bar{e}_v^i), \quad \bar{e}_v^i = \sum_{j \in \mathcal{S}_i} \frac{\exp \langle c_t^i, c_v^j \rangle / T}{\sum_{k \in \mathcal{S}_i} \exp \langle c_t^i, c_v^k \rangle / T} \cdot e_v^j. \quad (3)$$

By default, the summation in the softmax is conducted over a support set $\mathcal{S}_i$ containing all indices except i . In the experiments, we consider the following attention types for reconstruction. **Identity captioning** ($\mathcal{S}_i = \{i\}$) generates the caption from the corresponding video and reduces to the standard captioning objective, eq. (2). **Full support** ($\mathcal{S}_i = \{1, \dots, B\}$) considers all videos as possible candidates for captioning. **Hybrid captioning** sets the weights in eq. (3) as the average of the weights for identity captioning and full support. **Cross-captioning** ($\mathcal{S}_i = \{j \neq i\}$) considers all *but* the video that one wishes to caption. This variant forces the network to extract all information required for captioning from other videos in the batch. Figure 2 compares graphically these attention mechanisms.

Considering both discriminative and generative objectives for learning multimodal representations, our full objective is $\mathcal{L} = \mathcal{L}^{\text{contrast}} + \lambda \mathcal{L}^{\text{cross-captioning}}$, where λ balances two objectives. We set $\lambda = 10$ to ensure similar magnitudes for both losses in our experiments. In the training phase, we use Adam (Kingma & Ba, 2015) to minimize our loss. At inference time, we directly use $\Phi(t)$ and $\Psi(v)$ to encode video and text representations for retrieval.

3.2 MODEL ARCHITECTURE

We now discuss the details of the encoders and decoder components in our architecture, illustrated in fig. 2. For the *text decoder* $p(t|e_v)$ in eq. (2) and (3), we use a pre-trained T-5 decoder (Raffel et al., 2019).

For the *video representation* $c_v = \Psi(v) = \Psi''(\Psi'(v))$, we use a video encoder $e_v = \Psi'(v)$ followed by a multi-layer transformer pooling head $c_v = \Psi''(e_v)$. The encoder $\Psi'(v)$ concatenates the output of pretrained ResNet-152 (He et al., 2016) and R(2+1)D-34 (Tran et al., 2018) networks applied to individual video frames, resulting in a code $e_v = [e_{v1} \cdots e_{vM}]$ where M is the maximum duration of a video clip. For the pooling head $c_v = \Psi''(e_v)$, we consider a transformer architecture to attend to important context and summarize it into a fixed-length representation c_v . For this, we follow MMT (Gabeur et al., 2020), but with two important differences. First, while MMT uses 7 expert features that results in $7\times$ the sequence length, we only use a transformer to attend to early-fused motion and appearance features as the video representation, thus significantly reducing the sequence length and computational cost. Second, instead of stacking 6 transformer layers to encode the visual stream as in MMT, we only use a shallow two-layer transformer architecture with additional pre-encoders, further increasing model efficiency. As temporal 1D-convolutional neural networks (CNNs) (LeCun et al., 1998) were shown to effectively capture temporal dependencies in videos (Dong et al., 2019), we integrate CNNs into our transformer pooling heads to better capture video temporal signals. In more detail, we compute $c_v = \Psi''(e_v)$ by chaining two transformer layers, each of the type:

$$\psi(e) = \text{BN}(\text{FFN}(e_{\text{attn}}) + e_{\text{attn}}), \quad e_{\text{attn}} = \text{BN}(\text{MHA}(f(e)) + f(e)). \quad (4)$$

Here f is a pre-encoder that refines the video representation; we found empirically that a 1D CNN works well for this purpose. Then, we apply multi-head self-attention (MHA) (Vaswani et al., 2017) followed by a feed-forward network (FFN) with batch normalization (BN) (Ioffe & Szegedy, 2015). The architecture maps the input sequence e_v to a new ‘contextualized’ sequence of representation vectors; we take the first one as c_v .

The text representation decomposes in the same way as $c_t = \Phi(t) = \Phi''(\Phi'(t))$. The text encoder $e_t = \Phi'(t)$ uses a pretrained T-5 network resulting in a code $e_t = [e_{t1} \cdots e_{tN}]$, where N is the maximum length of a sentence. The pooling head $c_t = \Phi''(e_t)$ follows the same design as the video case, but f is set to a recurrent neural network (RNN) instead of a CNN. Please refer to the appendix for details.

In practice, for computational reasons, we use eq. (3) to finetune the parameters of all networks except the video encoder $\Psi'(v)$, which is fixed.

4 EXPERIMENTS

We validate empirically the ability of our method to learn better representations for the downstream tasks of text-to-video and video-to-text retrieval. First, in sec. 4.2 we ablate various model components on the MSR-VTT dataset. Then, in sec. 4.3 we show that our best model significantly outperforms state-of-the-art retrieval systems on three datasets, MSR-VTT, ActivityNet and VATEX. Finally, in sec. 4.4 we analyse qualitatively the effect of the attention mechanism used during training.

4.1 EXPERIMENTAL SETUP

Datasets. **HowTo100M** (Miech et al., 2019) is a large-scale instructional video collection of 1.2 million YouTube videos, along with automatic speech recognition transcripts. We use this dataset for our pre-training experiments. **MSR-VTT** (Xu et al., 2016) contains 10,000 videos, where each

Table 2: **Model Architecture and Training Details Ablation.** Text→Video retrieval performance on MSR-VTT. Recall@1, 5, and Median Recall are shown.(a) **Video Encoder.** Stronger features and combination improves performance.

Feature source	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
R-152	20.8	46.2	6.0
R(2+1)D-34	23.7	53.2	4.0
R(2+1)D-34 + R-152	27.2	55.2	3.0

(c) **Text Encoder.** Stronger encoding of text improves retrieval.

Text Encoder	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
W2V (GloVe)	22.1	49.8	6.0
T5-Small	24.5	51.2	3.0
T5-Base	27.2	55.2	3.0

(b) **Feature Aggregation.** Learning temporal attention yields strong gains over pooling.

Temporal reduction	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
Max	21.8	49.5	8.0
Mean	22.5	51.3	6.0
Multi-Head Attn	27.2	55.2	3.0

(d) **Text Decoder.** Stronger decoding of text improves retrieval.

Text Encoder	Text Decoder	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
T5-Base	T5-Small	26.2	54.2	3.0
T5-Base	T5-Base	27.2	55.2	3.0

(e) **Contrastive Loss.** Inter-modal Triplet loss yields the best performance.

Contrastive	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
InfoNCE (inter+intra)	10.7	28.5	15.0
InfoNCE (inter)	10.8	29.0	14.5
Triplet (inter+intra)	26.8	56.2	3.0
Triplet (inter)	27.2	55.2	3.0

(f) **Support-set Size.** Retrieval degrades when reconstructing from too small and too large sets.

Size	Batch-size							Memory bank	
	8	16	32	64	128	256	512	2k	8k
R@1/5	18.5/45.6	20.7/49.9	25.2/54.6	27.2/55.2	28.0/56.1	26.9/55.0	25.3/53.5	26.8/54.7	26.2/52.7

video is annotated with 20 descriptions. We report results on the 1k-A split (9,000 training, 1,000 testing) as in Liu et al. (2019). **VATEX** (Wang et al., 2019) is a multilingual (Chinese and English) video-text dataset with 34,911 videos. We use the official training split with 25,991 videos and report on the validation split as in HGR (Chen et al., 2020b). The **ActivityNet Caption** (Krishna et al., 2017) dataset consists of densely annotated temporal segments of 20K YouTube videos. We use the 10K training split to train from scratch/ finetune the model and report the performance on the 5K ‘val1’ split. The **MSVD** (Chen & Dolan, 2011) dataset consists of 80K English descriptions for 1,970 videos from YouTube, with each video associated with around 40 sentences each. We use the standard split of 1,200, 100, and 670 videos for training, validation, and testing (Liu et al., 2019; Venugopalan et al., 2015b; Xu et al., 2015).

Evaluation Metrics. To measure the text-to-video and video-to-text retrieval performance, we choose Recall at K ($R@K$) and Median Rank (MedR), which are common metrics in information retrieval.

4.2 ABLATIONS

In Tab. 2, we first only ablate the cross-modal retrieval part of our network architecture, while the generative objectives are analysed in Tab. 1.

Video Encoder. In Tab. 2a, we show the effect of the choice of visual input features. We find that for text-to-video retrieval at Recall at 1 and 5 ($R@1, R@5$), features obtained from a video R(2+1)D-34 ResNet achieve 2.9% and 7.0% higher performance compared to only image-frame based features from a ResNet-152. A further 3.5% and 2.0% can be gained by concatenating both features, yielding the strongest MdR of 3.0%.

Table 1: **Effect of learning objectives.** Text→Video retrieval on MSR-VTT.

	$R@1 \uparrow$	$R@5 \uparrow$	$MdR \downarrow$
None	25.9	53.0	4.0
Identity	26.4	51.9	4.0
Full	25.8	53.9	3.0
Hybrid	26.0	54.8	3.0
Cross	27.2	55.2	3.0

Feature Aggregation. While the features from both video and image-based visual encoders have reduced spatial extent after a fully-connected layer, the temporal dimension can be reduced in various ways. In Tab. 2b, we find that our multi-head, parameterized attention reduction yields strong gains over the mean- or max-pooling baselines of over 4% for $R@1$. This shows that learning attention over the temporal dimension of fixed feature sets can give strong gains even without fine-tuning the encoder.

Text Encoder. In Tab. 2c, we find decent gains of 2.7% and 0.4% for $R@1,5$ for using T5-base, instead of T5-small. We do not use the T5-Large model, as in Korbar et al. (2020), due to the prohibitively large relative model size increase of +220%.

Text Decoder. In Tab. 2d, we find that using a larger text decoder gives a 1% increase in performance when using the cross-captioning objective.

Contrastive Loss. To validate the choice of a triplet loss in eq. (1), in Tab. 2e, we compare the results of the InfoNCE contrastive loss (Oord et al., 2018) with a triplet loss, with both the intra and inter-intra modality variants. We find that InfoNCE (Oord et al., 2018) loss does not work well in our case, likely due to the difficulty in tuning this loss to have the right combination of temperature and batch-size.

Support-Set Size. Lastly, in Tab. 2f, we show the effect of the size of the support set used for cross-instance captioning. We find that our reconstruction loss indeed acts as a bottleneck, with both smaller and very large sizes degrading the performance.

Captioning Objective. In Tab. 1, we show the effect of the different variants of our learning objective eq. (3). First, we find that the naive addition of a reconstruction objective (“Identity”) does not improve the contrastive-only baseline (“None”) much. Considering reconstruction from other videos improves the performance more. In particular, the “Hybrid” variant, which combines “Identity” and “Full” (sec. 3.1.2) improves Recall at 1 and 5 from 25.9% and 53.0% to 26.0% and 54.8%, respectively. However, the best result by far (27.2/55.2%) is obtained forcing captions to be reconstructed only from *other* videos, via our cross-instance attention mechanism (“Cross”). This variant cannot use information contained in a video to generate the corresponding caption and thus entirely relies on the model to discover meaningful relationship between different videos. This newly-proposed scheme seems to have the most beneficial effect for semantic retrieval.

Table 3: **Retrieval performance on the MSR-VTT dataset.** Models in the second group are additionally pretrained on HowTo100M.

	Text → Video				Video → Text			
	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$
Random Baseline	0.1	0.5	1.0	500.0	0.1	0.5	1.0	500.0
JSFusion (Yu et al., 2018)	10.2	31.2	43.2	13.0	—	—	—	—
HT100M (Miech et al., 2019)	12.1	35.0	48.0	12.0	—	—	—	—
JPoSE (Wray et al., 2019)	14.3	38.1	53.0	9.0	16.4	41.3	54.4	8.7
CE (Liu et al., 2019)	20.9	48.8	62.4	6.0	20.6	50.3	64.0	5.3
MMT (Gabeur et al., 2020)	24.6	54.0	67.1	4.0	24.4	56.0	67.8	4.0
Ours	27.4	56.3	67.7	3.0	26.6	55.1	67.5	3.0
VidTranslate (Korbar et al., 2020)	14.7	—	52.8	—	—	—	—	—
HT100M (Miech et al., 2019)	14.9	40.2	52.8	9.0	16.8	41.7	55.1	8.0
NoiseEstimation (Amrani et al., 2020)	17.4	41.6	53.6	8.0	—	—	—	—
UniVL (Luo et al., 2020)	21.2	49.6	63.1	6.0	—	—	—	—
AVLnet (Rouditchenko et al., 2020)	27.1	55.6	66.6	4.0	28.5	54.6	65.2	4.0
MMT (Gabeur et al., 2020)	26.6	57.1	69.6	4.0	27.0	57.5	69.7	3.7
Ours-pretrained	30.1	58.5	<u>69.3</u>	3.0	28.5	58.6	71.6	3.0

4.3 COMPARISON TO STATE-OF-THE-ART

In this section, we compare the results of our method to other recent text-to-video and video-to-text retrieval approaches on various datasets. In Tab. 3 to 5, we show the results of our model applied to text-to-video and video-to-text retrieval on MSR-VTT, VATEX, ActivityNet and MSVD with and

Table 4: Retrieval performance on the VATEX dataset

	Text $\rightarrow$ Video				Video $\rightarrow$ Text			
	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$
Random Baseline	0.2	0.7	1.05	2000.5	0.02	0.1	1.02	2100.5
VSE (Kiros et al., 2014)	28.0	64.3	76.9	3.0	—	—	—	—
VSE++ (Faghri et al., 2018)	33.7	70.1	81.0	2.0	—	—	—	—
Dual (Dong et al., 2019)	31.1	67.4	78.9	3.0	—	—	—	—
HGR (Chen et al., 2020b)	35.1	73.5	83.5	2.0	—	—	—	—
Ours	44.6	81.8	89.5	1.0	58.1	83.8	90.9	1.0
Ours-pretrained	45.9	82.4	90.4	1.0	61.2	85.2	91.8	1.0

Table 5: Retrieval performance on ActivityNet

	Text $\rightarrow$ Video				Video $\rightarrow$ Text			
	$R@1\uparrow$	$R@5\uparrow$	$R@50\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@50\uparrow$	$MdR\downarrow$
Random Baseline	0.02	0.1	1.02	2458	0.02	0.1	1.02	2458
FSE(Zhang et al., 2018)	18.2	44.8	89.1	7.0	16.7	43.1	88.4	7.0
CE (Liu et al., 2019)	18.2	47.7	91.4	6.0	17.7	46.6	90.9	6.0
HSE (Zhang et al., 2018)	20.5	49.3	—	—	18.7	48.1	—	—
MMT (Gabeur et al., 2020)	22.7	54.2	93.2	5.0	22.9	54.8	93.1	4.3
Ours	26.8	58.1	93.5	3.0	25.5	57.3	93.5	3.0
MMT-pretrained (Gabeur et al., 2020)	28.7	61.4	94.5	3.3	28.9	61.1	94.3	4.0
Ours-pretrained	29.2	61.6	94.7	3.0	<u>28.7</u>	<u>60.8</u>	94.8	2.0

Table 6: Retrieval performance on the MSVD dataset

	Text $\rightarrow$ Video				Video $\rightarrow$ Text			
	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$
VSE (Kiros et al., 2014)	12.3	30.1	42.3	14.0	—	—	—	—
VSE++ (Faghri et al., 2018)	15.4	39.6	53.0	9.0	—	—	—	—
Multi. Cues (Mithun et al., 2018)	20.3	47.8	61.1	6.0	—	—	—	—
CE (Liu et al., 2019)	19.8	49.0	63.8	6.0	—	—	—	—
Ours	23.0	52.8	65.8	5.0	27.3	50.7	60.8	5.0
Ours-pretrained	28.4	60.0	72.9	4.0	34.7	59.9	70.0	3.0

without pre-training on HowTo100M. Without pre-training, our method outperforms all others in all metrics and datasets. In particular, for the VATEX dataset, our retrieval performance at recall at 1 and 5 is 45.9% and 82.4%, exceeding recent state-of-the-art methods (Chen et al., 2020b) by a margin of 9%. For ActivityNet, our model outperforms MMT by a margin of 4% at recall at 1. With pre-training on HowTo100M, our performance further increases across the board. Notably, unlike MMT which uses 7 features, our model uses only 2 features and achieves state-of-the-art in most metrics.

4.4 ANALYSIS

In order to better understand the effect of our learning objective, we visualize the soft attention of our best-performing cross-instance reconstruction model in fig. 3. As we can see in the top-left square, which shows the pairwise attention between all pairs of videos in the batch, it is highly focused, with the model mostly attending one or two other instances in the batch.



Fig. 3: **Support-set attention map.** Attention scores of all pairs in a batch (top-left square) and a subset of rows/columns (other squares) on VTT.

For the first video’s caption reconstruction (second row), we find that the model solely attends to another musical performance video that is in the batch, ignoring the others. For the second video (third row), the model focuses on another sample that shows the sea but differs in most other aspects since there are no semantically-equivalent clips in the batch. The third video shares a similar scenario. These examples show that the bottleneck is effective at forcing the model to avoid memorising the video-caption association of each clip in isolation, and attempt to match other clips more broadly, since an exact (or very close) match is not guaranteed.

5 CONCLUSION

In this work, we studied classic contrastive learning methods such as the triplet loss to learn video-text representations for cross-model retrieval. We suggested that the contrastive approach might pull apart videos and captions even when they are semantically equivalent, which can hinder downstream retrieval performance. To mitigate this effect, we propose to consider a captioning pretext task as an additional learning objective. In particular, we show that cross-instance captioning can encourage the representation to pull together videos that share a similar caption, and are thus likely to be equivalent for retrieval. Leveraging these ideas, our model achieves state-of-the-art performance on the text-to-video and video-to-text retrieval tasks, on three datasets.

While we demonstrated these ideas in the specific case of text-to-video retrieval, they can in principle generalize to any setting that utilizes a contrastive loss, including self-supervised learning, provided that it is possible to learn reasonable conditional generators of a modality or data stream given another.

ACKNOWLEDGEMENTS

We are grateful for support from the Rhodes Trust (M.P.), the Royal Academy of Engineering (DFR05420, J.H), Facebook (M.P. and P.H.), EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems [EP/L015897/1] (M.P. and Y.A.) and the Qualcomm Innovation Fellowship (Y.A.). This work is also supported by the DARPA grant funded under the GAILA program (award HR00111990063) (P.H.).

REFERENCES

- Jean-Baptiste Alayrac, A. Recasens, Ros  lia G. Schneider, R. Arandjelovi  , Jason Ramapuram, J. Fauw, Lucas Smaira, S. Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *ArXiv*, abs/2006.16228, 2020.
- Humam Alwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.
- Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. *arXiv preprint arXiv:2003.03186*, 2020.
- Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *NeurIPS*, 2020a.
- Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020b.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- M. Caron, I. Misra, J. Mairal, Priya Goyal, P. Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.
- Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.
- David L Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 190–200. Association for Computational Linguistics, 2011.
- Mark Chen, Alec Radford, Rewon Child, Jeff Wu, and Heewoo Jun. Generative pretraining from pixels. In *ICML*, 2020a.
- Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. 2020b.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020c.
- Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*, 2019.
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators, 2020.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pp. 248–255, 2009.
- Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pp. 376–380, 2014.
- Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. *arXiv preprint arXiv:2006.06666*, 2020.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, 2019.
- Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *NeurIPS*, 2019.
- Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. Dual encoding for zero-example video retrieval. In *CVPR*, 2019.
- Haodong Duan, Yue Zhao, Yuanjun Xiong, Wentao Liu, and Dahua Lin. Omni-sourced webly-supervised learning for video recognition. In *ECCV*, 2020.
- Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. In *BMVC*, 2018.
- Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.
- Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.

- Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.
- Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010.
- Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
- Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV Workshops*, 2019.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020.
- Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with temporal language. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.
- Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.
- Jingyi Hou, Xinxiao Wu, Wentian Zhao, Jiebo Luo, and Yunde Jia. Joint syntax representation learning and visual cue translation for video captioning. In *ICCV*, 2019.
- Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.
- Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *NeurIPS*, 2020.
- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.
- Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation, 2020.
- Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *CVPR*, 2017.
- H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.
- Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *EMNLP*, pp. 1369–1379, 2018.
- Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. Pre-training via paraphrasing, 2020a.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, 2020b.
- Gen Li, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*, 2020a.
- Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations, 2020b.
- Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language, 2019.
- Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. Jointly localizing and describing events for dense video captioning. In *CVPR*, 2018.
- Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

- Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *BMVC*, 2019.
- Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019.
- Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation, 2020.
- Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.
- Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
- Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pp. 19–27, 2018.
- Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. *arXiv preprint arXiv:2004.12943*, 2020.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.
- Mandela Patrick, Yuki M. Asano, Polina Kuznetsova, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations, 2020.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.
- AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020.
- Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pp. 5533–5541, 2017.
- Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.
- Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Chris Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *International Journal of Computer Vision*, 2017.
- Andrew Rouditchenko, Angie Boggust, David Harwath, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Rogerio Feris, Brian Kingsbury, Michael Picheny, Antonio Torralba, et al. Avl-net: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*, 2020.
- Mert Bulent Sariyildiz, Julien Perez, and Diane Larlus. Learning visual representations with caption annotations, 2020.
- Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.
- Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vi-bert: Pre-training of generic visual-linguistic representations, 2019.

- Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. 2017.
- Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer, 2019a.
- Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019b.
- Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NeurIPS*, pp. 3104–3112, 2014.
- Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019.
- Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.
- Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.
- Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.
- Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence – video to text. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015a.
- Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence – video to text. In *ICCV*, 2015b.
- Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018.
- Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4581–4591, 2019.
- Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *ICCV*, 2019.
- Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.
- Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 305–321, 2018.
- Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.
- Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. Citeseer, 2015.
- I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification, 2019.
- Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018.
- Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, pp. 374–390, 2018.
- Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.
- Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. Object relational graph with teacher-recommended learning for video captioning. In *CVPR*, 2020.
- Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. 2018a.
- Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *CVPR*, 2018b.
- Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.

6 APPENDIX

The appendix is organized as follows: First, we provide more details about our model. Then we introduce the datasets and the experimental setup. Finally, we provide additional qualitative and quantitative experimental results for video-text retrieval and captioning.

6.1 MODEL DETAILS

Implementation details and hyper parameters. For our text encoder, we use the T5-base model pre-trained on the “Colossal Clean Crawled Corpus” (C4) (Raffel et al., 2019). We use its corresponding text tokenizer and encode a sentence into a sequence of 1024 dimensional vectors.

For our visual encoder, our model utilizes only the motion and the appearance features. For the motion feature, we use a 34-layer, R(2+1)-D (Tran et al., 2018) model pre-trained on IG65M (Ghadiyaram et al., 2019) and apply a spatial-temporal average pooling over the last convolutional layer, resulting in a 512-dimensional vector. For the appearance feature, we use the 2048-dimension flattened pool-5 layer of the standard ResNet152 (He et al., 2016) pre-trained on Imagenet (Deng et al., 2009). We extract features at a rate of 1 feature per second and simply concatenate the two features, resulting in a 2560-dimension visual input stream. Noteworthy, instead of using 9 and 7 different types of visual features as in CE (Liu et al., 2019) and MMT (Gabeur et al., 2020), we use only the above 2 features and achieve on par or superior performance. Also, with early fusion, our model does not suffer from additional computation required for the extended sequence length in MMT. For the text decoder, we use the T5-base model decoder, also pre-trained on C4.

As illustrated in Fig. 4, our transformer pooling head is composed of a pre-encoder, a multi-head self-attention, and a FFN layer. For pre-encoders, we use a one-layer MLP with a d -dimensional output for mapping video features into the common embedding space. We use 1024-dimension bi-directional GRU as the text pre-encoder. For the 1D-CNN prior, we use kernels with size $[2, 3, 4, 6]$ as the visual and text pre-encoders. We set the embedding dimension to 1024 and use 4 attention heads in the transformer pooling layers. The hidden dimension of FFN is 2048.

Training and Inference time. Pre-training on 1.2 million HowTo100M videos takes around 160 GPU hours (NVIDIA V100) for 20 epochs. We speed up the pre-training process by distributing the workload over 8 GPUs. We use 1 GPU for the fine-tuning or training from scratch experiments. For the MSR-VTT 1k-A split, it takes 12 GPU hours to train our full model on 180K video-text pairs for 20 epochs. For Vatex, it takes 32 GPU hours to train on 260K video-text pairs for 30 epochs. For ActivityNet, it takes 2.5 GPU hours to train on 10K video-text pairs for 28 epochs.

For inference, the encoding speed is around 250-300 video/sec and 200-250 text query/sec. The overall text-to-video search speed on 5,000 video-text pairs (5,000 text queries over 5,000 videos) is 30-34 seconds including encoding. The speed of text-to-video retrieval is similar to video-to-text retrieval.

6.2 EXPERIMENT DETAILS

The margin α of the max-margin loss is 0.2, and the temperature T is set to 0.1 as used in SimCLR Chen et al. (2020c). We use the Adam (Kingma & Ba, 2015) optimizer with a initial learning rate $5 \cdot 10^{-5}$ and clip gradients greater than 0.2 during the training phase. Dropout rate is 0.3 for all datasets besides ActivityNet (0.0).

As the average video/text lengths and videos available are quite different across datasets, we adjust our training scheme accordingly. When training on MSR-VTT, ActivityNet and Vatex, batch-size is set to 64. For MSR-VTT training, we sample and truncate videos to 32 seconds, text to 100 tokens and train for 20 epochs. For Vatex, videos are at most 64 seconds and we train for 30 epochs. For ActivityNet training, videos are at most 512 seconds and 256 tokens for the text part. We train for 28 epochs on ActivityNet. For fine-tuning HowTo100M pre-trained model, we reduce training epochs into quarters.

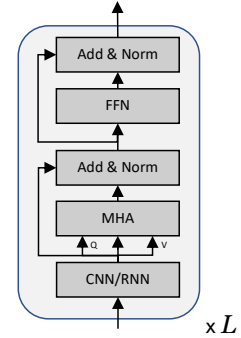


Fig. 4: Transformer pooling head.

6.3 DATASET DETAILS

HowTo100M (Miech et al., 2019) is a large-scale instructional video collection of 1.2 million Youtube videos, along with automatic speech recognition transcripts. There are more than 100 million clips (ASR segments) defined in HowTo100M. We use this dataset for pretraining.

MSR-VTT (Xu et al., 2016) contains 10,000 videos, where each video is annotated with 20 descriptions. For retrieval experiments and ablation studies, we follow the training protocol and defined in Gabeur et al. (2020); Liu et al. (2019); Miech et al. (2019) and evaluate on text-to-video and video-to-text search tasks on the 1k-A testing split with 1,000 video or text candidates defined by Yu et al. (2018). For captioning task, we evaluate on the standard testing split with 2,990 videos.

VATEX (Wang et al., 2019) is a multilingual (Chinese and English) video-text dataset with 34,911 videos. We use the official split with 25,991 videos for training. As the testing annotations are private in VATEX, we follow the protocol in Chen et al. (2020b) to split the validation set equally (1,500 validation and 1,500 testing videos) for model selection and testing. For each video, 10 English and 10 Chinese descriptions are available, and we only use the English annotations.

ActivityNet Dense Caption dataset consists densely annotated temporal segments of 20K YouTube videos. Following Gabeur et al. (2020); Zhang et al. (2018), we concatenate descriptions of segments in a video to construct “video-paragraph” for retrieval and captioning. We use the 10K training split to train from scratch/ finetune the model and report the performance on the 5K ‘val1’ split.

MSVD dataset consists of 80K English descriptions for 1,970 videos from YouTube, with each video associated with around 40 sentences each. We use the standard split of 1200, 100, and 670 videos for training, validation, and testing (Liu et al., 2019; Venugopalan et al., 2015b; Xu et al., 2015).

6.4 VIDEO CAPTIONING EXPERIMENTS

To measure captioning/text generation performance, we report BLEU4 (Papineni et al., 2002), METEOR (Denkowski & Lavie, 2014), Rogue-L (Lin, 2004) and CIDEr (Vedantam et al., 2015) metrics. We report results on the MSR-VTT, VATEX and ActivityNet datasets.

Table 7: Captioning performance on the MSR-VTT dataset

	Captioning			
	BLUE4	METEOR	Rogue-L	CIDEr
VidTranslate (Korbar et al., 2020)	41.7	28.5	—	—
POS+VCT (Hou et al., 2019)	42.3	29.7	62.8	49.1
ORG (Zhang et al., 2020)	43.6	28.8	62.1	50.9
Ours, MSR-VTT only	39.7	28.3	60.5	46.5
Ours, HT100M + MSR-VTT	38.9	28.2	59.8	48.6

Table 8: Captioning performance on the VATEX dataset

	Captioning			
	Blue@4	METEOR	Rogue-L	CIDEr
Shared Enc-Dec (Wang et al., 2019)	28.4	21.7	47.0	45.1
ORG (Zhang et al., 2020)	32.1	22.2	48.9	49.7
Ours, VATEX only	32.8	24.4	49.1	51.2
Ours, HT100M + VateX	32.5	24.1	48.9	50.5

6.5 ZERO-SHOT RETRIEVAL EXPERIMENTS

We also evaluate our model in the zero-shot setting on MSR-VTT, VateX, ActivityNet and MSVD, after pre-training on HT100M. While we are able to get reasonable results on MSR-VTT and MSVD, our results are not great on VateX and Activity-Net due to significant domain gap.

Table 9: Captioning performance on the ActivityNet dataset

	Captioning			
	Blue@4	METEOR	Rogue-L	CIDEr
DENSE (Krishna et al., 2017)	1.6	8.9	—	—
DVC-D-A (Li et al., 2018)	1.7	9.3	—	—
Bi-LSTM+TempoAttn (Zhou et al., 2018b)	2.1	10.0	—	—
Masked Transformer (Zhou et al., 2018b)	2.8	11.1	—	—
Ours, ActivityNet only	1.5	6.9	17.8	3.2
Ours, HT100M + ActivityNet	1.4	6.9	17.5	3.1

Table 10: Zero-shot Retrieval performance on VATEX, MSR-VTT, MSVD and ActivityNet.

	Video → Text				Text → Video			
	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$
Zero-Shot								
ActivityNet	0.06	0.2	0.5	1907.0	0.0	0.2	0.3	2238.0
VATEX	0.07	0.4	0.7	682.0	0.07	0.4	0.9	697
MSVD	8.9	26.0	37.9	18.0	21.4	46.2	57.7	6.0
MSR-VTT	8.7	23.0	31.1	31.0	12.7	27.5	36.2	24.0

6.6 ACTION RECOGNITION

Lastly, we evaluate our model on the video action recognition task on HMDB-51 (Kuehne et al., 2011) and UCF-101 (Soomro et al., 2012). For this, we use the R(2+1)D-34 (pretrained on IG65M) model as well as a ResNet-152 model (pretrained on Imagenet), as in our method. We extract a feature per second per video by concatenating the features from each model (2560-D), and obtain an average representation per video using either average pooling (2560-D) or our proposed transformer pooling head (1024-D) pre-trained on HT100M using cross-captioning objective. We then train a linear classifier for 1500 epochs for HMDB-51 (500 for UCF-101) on these features using Adam (Kingma & Ba, 2015) optimizer with learning rate of $1e^{-4}$ and weight decay $1e^{-4}$ with early stopping. We also drop the learning rate by 10 at epochs 200, 400 for UCF-101 and 1000, 1200 for HMDB-51. In Table 11, we show the results of training only a linear-layer on features extracted from our fixed backbone with or without a learned transformer-pooling head. We find that our transformer temporal pooling head provides significant benefits over the baseline of simply average pooling the features, demonstrating the effectiveness of building contextualized representations using our proposed transformer. In particular, we see improvements of over 7% on HMDB-51 and 34% on UCF-101 by replacing average pooling with our transformer pooling head to aggregate features. We observe that naive average pooling performs significantly worse than our transformer pooling under evaluation protocol. This is likely because 1) the average pooling collapses temporal information, making the linear layer based classification difficult 2) compared to the transformer pooling, it does not benefit from large-scale pretraining on a wide variety of action videos of HT100M. We further compare very favorably to the current state-of-the-art approaches. In particular, we outperform all other approaches, both supervised and self-supervised, except the recently introduced Omni (Duan et al., 2020) which was finetuned on both UCF-101 and HMDB-51, while we only trained a linear classifier on extracted features. However, it should be noted that it is very difficult to fairly compare all these different approaches because they may use different modalities (images, RGB video, optical flow, audio, ASR outputs), pretraining datasets (Kinetics-400, HT100M, IG65M, Imagenet), architectures (S3D, I3D, R(2+1)D, R3D), pre-training (supervised, self-supervised) and downstream training (frozen, finetuned) strategies.

6.7 STATISTICAL SIGNIFICANCE

In Table 12, we show the results of finetuning our pretrained model for 3 times on the VATEX dataset. We find that the variance is quite low and our model consistently beats the state of the art.

Table 11: **Action recognition.** Results of training only a linear-layer, on features extracted from our fixed backbone with or without a learned transformer-pooling head. We compare to the state-of-art supervised and self-supervised pretraining methods on the HMDB-51 and UCF-101 action recognition task, for different downstream training protocols (“FT?” stands for finetuned). We report average Top-1 accuracy across all 3 folds. Dataset abbreviations: AudioSet, HMDB51, HowTo100M, Interact65M, IMagenet-1000, Kinetics400, Omnisource Images + Videos, Sports1M, UCF101, YouTube8M. Other abbreviations: Video modality, Flow modality, Image modality, Audio modality, Transformer pooling, Average pooling

Method	Mod	Dataset	Model	FT?	H51	U101
<i>Self-Supervised Pre-training</i>						
MIL-NCE (Miech et al., 2020)	V,T	HM	S3D-G	✗	53.1	82.7
MIL-NCE (Miech et al., 2020)	V,T	HM	S3D-G	✓	61.0	91.3
MMV (Alayrac et al., 2020)	V,T,A	HM+AS	TSM-50x2	✗	67.1	91.8
ELo (Piergiovanni et al., 2020)	V,F,A	YT8M	R(2+1)D-50x3	✓	67.4	93.8
XDC (Alwassel et al., 2020)	V,A	IG65M	R(2+1)D-18	✓	68.9	95.5
GDT (Patrick et al., 2020)	V,A	IG65M	R(2+1)D-18	✓	72.8	95.2
MMV (Alayrac et al., 2020)	V,T,A	HM+AS	TSM-50x2	✓	75.0	95.2
<i>Supervised Pre-training</i>						
P3D (Qiu et al., 2017)	V,I	S1M+IM	P3D	✓	—	88.6
TSN (Wang et al., 2018)	V,I	IM	TSN	✓	69.4	94.2
I3D (Carreira & Zisserman, 2017)	V,I	K400+IM	I3D	✓	74.8	95.6
R(2+1)D (Tran et al., 2018)	V	K400	R(2+1)D-34	✓	74.5	96.8
S3D-G (Xie et al., 2018)	V,I	K400+IM	S3D-G	✓	75.9	96.8
I3D (Carreira & Zisserman, 2017)	V,I	K400+IM	I3D	✓	77.1	96.7
R(2+1)D (Tran et al., 2018)	V	K400	R(2+1)D-34	✓	76.4	95.5
R(2+1)D (Tran et al., 2018)	V,F	K400	R(2+1)D-34x2	✓	78.7	97.3
Omni (Duan et al., 2020)	V,I	K400+OS	Slow-8x8-R101	✓	79.0	97.3
I3D (Carreira & Zisserman, 2017)	V,F,I	K400+IM	I3Dx2	✓	80.7	98.0
Omni (Duan et al., 2020)	V,F,I	K400+OS	Slow-8x8-R101x2	✓	83.8	98.6
Ours (Avg-pooling)	V,I	IG65M+IM	R(2+1)D-34+R152	✗	73.7	64.3
Ours (T-pooling)	V,I	HM+IG65M+IM	R(2+1)D-34+R152	✗	<u>81.3</u>	<u>98.0</u>

Table 12: **Retrieval performance on the VATEX dataset**

	Text → Video				Video → Text			
	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$	$R@1\uparrow$	$R@5\uparrow$	$R@10\uparrow$	$MdR\downarrow$
Random Baseline	0.2	0.7	1.05	2000.5	0.02	0.1	1.02	2100.5
VSE (Kiros et al., 2014)	28.0	64.3	76.9	3.0	—	—	—	—
VSE++ (Faghri et al., 2018)	33.7	70.1	81.0	2.0	—	—	—	—
Dual (Dong et al., 2019)	31.1	67.4	78.9	3.0	—	—	—	—
HGR (Chen et al., 2020b)	35.1	73.5	83.5	2.0	—	—	—	—
Ours	44.9± 0.2	82.1± 0.2	89.7± 0.2	1.0	58.4± 0.1	84.4± 0.2	91.0± 0.3	1.0

6.8 ADDITIONAL QUALITATIVE RESULTS

We provide additional qualitative text-to-video retrieval results on MSR-VTT, VATEX, ActivityNet in Fig. 5. Given a text query, in most cases, our model successfully retrieves the correct videos marked in green.



Fig. 5: Examples of top-3 Text→Video retrieval results and similarities on the MSR-VTT, VATEX, and ActivityNet testing set. Only one correct video (colored in green) for each text query on the top.

7

Multi-modal Self-Supervision from Generalized Data Transformations

**This work will be presented at the International Conference
on Computer Vision (ICCV) 2021.**

On Compositions of Transformations in Contrastive Self-Supervised Learning

Mandela Patrick^{1,2*} Yuki M. Asano^{2*} Polina Kuznetsova¹ Ruth Fong²
 João F. Henriques² Geoffrey Zweig¹ Andrea Vedaldi^{1,2}

¹ Facebook AI Research

mandelapatt@facebook.com

² Visual Geometry Group, University of Oxford

yuki@robots.ox.ac.uk

Abstract

In the image domain, excellent representations can be learned by inducing invariance to content-preserving transformations via noise contrastive learning. In this paper, we generalize contrastive learning to a wider set of transformations, and their compositions, for which either invariance or distinctiveness is sought. We show that it is not immediately obvious how existing methods such as SimCLR can be extended to do so. Instead, we introduce a number of formal requirements that all contrastive formulations must satisfy, and propose a practical construction which satisfies these requirements. In order to maximise the reach of this analysis, we express all components of noise contrastive formulations as the choice of certain generalized transformations of the data (GDTs), including data sampling. We then consider videos as an example of data in which a large variety of transformations are applicable, accounting for the extra modalities – for which we analyze audio and text – and the dimension of time. We find that being invariant to certain transformations and distinctive to others is critical to learning effective video representations, improving the state-of-the-art for multiple benchmarks by a large margin, and even surpassing supervised pretraining. Code and pretrained models are available¹.

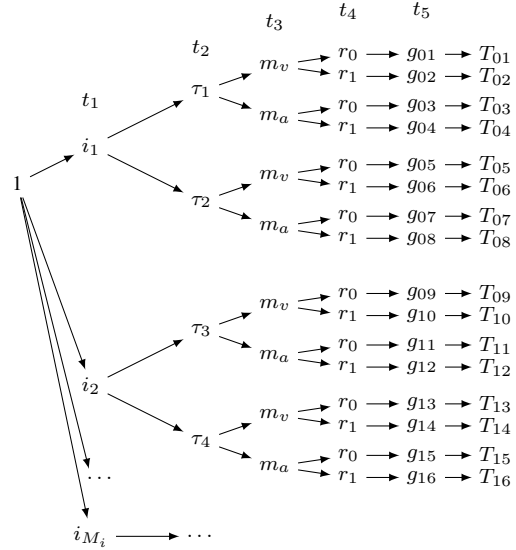


Fig. 1: Hierarchical sampling process of generalized data transformations (GDTs). Shown here are the five transformations analyzed for the audio-visual training case and their compositions: data-sampling (t_1), time-shift (t_2), modality splicing (t_3), time-reversal (t_4), and augmentation transformations, g (t_5) to learn video representations via noise contrastive learning.

1. Introduction

Works such as MoCo [31], SimCLR [13], SwAV [12] and BYOL [25] have shown that it is possible to pre-train state-of-the-art image representations without the use of any manually-provided labels. Furthermore, many of these approaches use variants of noise contrastive learning [26, 27].

Their idea is to learn a representation that is *invariant* to transformations that leave the meaning of an image unchanged (e.g. geometric distortion or cropping) and *distinctive* to changes that are likely to alter its meaning (e.g. replacing an image with another chosen at random).

These prior works have also shown that the choice of transformations is of primary importance for performance [12, 13]. This is not just a matter of selecting a

*Joint first authors

¹<https://github.com/facebookresearch/GDT>

certain type of transformation, but also to specify how different transformations should be composed, and how these compositions should be sampled to form batches for training the model. So far, these choices have been mostly driven by intuition, with little formal understanding of why certain choices may be preferable, and how these choices can be generalized.

In this work, we answer some of these questions via a formal analysis of *composable transformations in contrastive learning*. Our analysis shows how invariance and distinctiveness to individual transformations can be meaningfully incorporated in the same learning formulation. It also provides some principles to guide the construction of the training batches. We interpret existing sampling schemes, such as the one in SimCLR, as special cases with certain potential advantages and disadvantages. We do so by showing how these constructions can be extended systematically to any composition of invariant and distinctive transformations.

Furthermore, we demonstrate the utility of our analysis by exploring contrastive methods for learning representations of video data. Compared to images, videos contain a time dimension and multiple modalities, which have been shown to provide effective learning cues; for instance [60] leverages multiple modalities, and [15, 41] the time dimension. We show how these effects can be incorporated in a uniform manner in contrastive learning by considering a suitable class of *generalized data transformations* (GDTs). GDTs capture standard augmentations, as well as temporal transformations, modality slicing and data sampling. The advantages of using GDTs is that they allow us to base the entire design of the learning formulation (e.g., how to write a coherent learning objective and how to sample batches) on a small number of design principles that our analysis has identified.

With this, we make some notable findings for contrastive video representation learning. First, we show that using this wider class of transformations greatly exceeds the performance that can be obtained by a vanilla applications of image-centric methods such as SimCLR to video data. By leveraging time and multiple modalities, we obtain large performance gains, almost *doubling* the performance. Second, we show that just learning representations that are invariant to more and more transformations is *not* optimal, at least when it comes to video data; instead, combining invariance to certain factors with distinctiveness to others performs better. To the best of our knowledge, this is the first time such an effect has been demonstrated in contrastive learning.

We also set the new state of the art in audio-visual representation learning, with both small and large video pre-training datasets on a variety of downstream tasks. In particular, we achieve 94.1% and 67.4% on the standardized UCF-101 [67] and HMDB-51 [42] action recognition benchmarks, when pretrained on HowTo100M [50], and 95.2% and 72.8% respectively when pretrained on IG65M [21].

2. Related work

Self-supervised learning from images and videos. A variety of pretext tasks have been proposed to learn representations from unlabelled **images**. Some tasks leverage the spatial context in images [17, 56] to train CNNs, while others create pseudo classification labels via artificial rotations [23], or clustering features [6, 10, 11, 12, 22, 37]. Colorization [83, 84], inpainting [62], solving jigsaw puzzles [57], as well as the contrastive methods detailed below, have been proposed for self-supervised image representation learning. Some of the tasks that use the space dimension of images have been extended to the space-time dimensions of **videos** by crafting equivalent tasks. These include jigsaw puzzles [40], and predicting rotations [38] or future frames [28]. Other tasks leverage the temporal dimension of videos to learn representations by predicting shuffled frames [53], the direction of time [76], motion [74], temporal ordering [43, 80], and playback speed [9, 14, 19]. These pretext-tasks can be framed as GDTs.

Multi-modal learning. Videos, unlike images, are a rich source of a variety of modalities such as speech, audio, and optical flow, and their correlation can be used as a supervisory signal. This idea has been present as early as 1994 [16]. Only recently, however, has multi-modal learning been used to successfully learn effective representations by leveraging the natural correspondence [2, 4, 5, 7, 54, 60] and synchronization [15, 41, 59] between the audio and visual streams. A number of recent papers have leveraged speech as a weak supervisory signal to train video representations [46, 49, 55, 68, 69] and recently [1], who use speech, audio and video. Other works incorporate optical flow and other modalities [29, 30, 64, 85] to learn representations. In CMC [70], representations are learned with different views such as different color channels or modalities to solely induce multi-view invariance. In contrast, our work extends this to and analyses multi-modal transformations and examines their utility as an invariant *or* distinctive learning signal.

Noise Contrastive Loss. Noise contrastive losses [26, 27] measure the similarity between sample pairs in a representational space and are at the core of several recent works on unsupervised feature learning. They yield good performance for learning image [13, 31, 33, 35, 45, 52, 58, 70, 71, 77] and video [3, 28, 34, 46, 49, 54, 66, 68, 82] representations, and circumvent the need to explicitly specify what information needs to be discarded via a designed task.

We leverage the noise contrastive loss as a learning framework to encourage the network to learn desired invariance and distinctiveness to data transformations. The GDT framework can be used to combine and extend many of these cues, contrastive or not, in a single noise contrastive formulation.

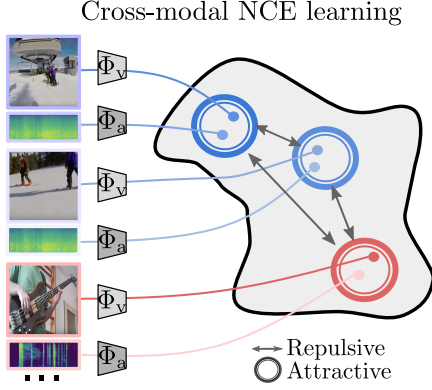


Fig. 2: **Example instantiation.** The embedding is learned via noise contrastive (NCE) learning. Here we show the case of audio-visual sample and time-shift distinctiveness: video-audio embeddings from the same video at the same time are pulled together, while audio-visual sample pairs from different videos and different starting times are pushed apart.

3. Method

We generalize contrastive methods such as CPC [58], PIRL [52], MoCo [31], SimCLR [13], and SwAV [12] to learn representations that can be invariant or distinctive to any number of transformations.

Given a collection x of data such as images or videos, we generate training samples

$$x(t_1, \dots, t_M) \in \mathcal{X}.$$

by applying a sequence of M transformations $T = (t_1, \dots, t_M)$ to the collection. We consider typical transformations such as data augmentations (e.g., randomly cropping an image). We also find it useful to express in the same manner other operations such as extracting a specific image or video from the collection or extracting a specific modality from a video. We call these *generalized data transformations* (GDTs).

To provide a concrete example, in a standard contrastive learning formulation such as SimCLR, the first transformation $t_1 = i \in \{1, \dots, |x|\}$ extracts an image x_i from the collection x and the second transformation $t_2 = g$ applies to it a random augmentation, so that we can write $x(t_1, t_2) = g(x_i)$. The goal is to learn a representation $\Phi : \mathcal{X} \rightarrow \mathbb{R}^d$ that identifies the image regardless of the augmentation; in other words, Φ should be invariant to the choice of t_2 and distinctive for the choice of t_1 .

We wish to generalize this construction to richer data such as videos. Compared to images, videos contain multiple modalities and additional dimensions, which allows to consider qualitatively different transformations such as time

shift, time reversal, and modality slicing. This generalization is however non-trivial. First, when considering $M > 2$ transformations, we have a choice of making the representation invariant or distinctive to each of them independently. For instance, video representations may benefit from being distinctive to time shift and/or time reversal rather than invariant to them. It is not immediately obvious how contrastive learning should be modified to incorporate these different choices. Another less apparent but important issue is how training data batches should be formed. Contrastive learning formulations minimize, in fact, a loss that involves comparing (contrasting) the representations of different samples, and is thus *not decomposable*. In practice, the loss is approximated by sampling batches of data, and how this is done has a major effect on the performance. In the previous example of SimCLR, if transformation (t_1, t_2) is included in the batch, so must be a complementary transformation (t_1, t'_2) that differs only in the second factor $t_2 \neq t'_2$. This is required in order to learn the desired invariance. It also means that transformations in a batch cannot be sampled independently. A way to guarantee that both (t_1, t_2) and (t_1, t'_2) are in the batch is to consider all possible combinations $\mathcal{T}_1 \times \mathcal{T}_2$ of two sets of transformations $\mathcal{T}_1$ and $\mathcal{T}_2$. However this is statistically inefficient because it applies the same augmentations $\mathcal{T}_2$ to all images in the batch. Instead, SimCLR samples at random $B/2$ images and then applies to them B independently sampled augmentations. This is better than the scheme above that would only use $B/|\mathcal{T}_1| = 2$ different augmentations. However, it is unclear how this strategy for sampling diverse transformations can be extended to $M > 2$ factors. This is studied next.

3.1. Compositional contrastive learning

Given a batch $\mathcal{T}$ of data transformations, we consider the learning objective:

$$\mathcal{L}(\Phi; \mathcal{T}) = - \sum_{T, T' \in \mathcal{T}} c(T, T') w(T, T') \cdot \log \left(\frac{e^{\langle \Phi(x(T)), \Phi(x(T')) \rangle / \rho}}{\sum_{T'' \in \mathcal{T}} w(T, T'') e^{\langle \Phi(x(T)), \Phi(x(T'')) \rangle / \rho}} \right). \quad (1)$$

where $\rho > 0$ is a temperature parameter. The *contrast function* $c(T, T') \in \{0, 1\}$ has the following interpretation: when $c(T, T') = 1$, then the representations $\Phi(x(T))$ and $\Phi(x(T'))$ are pulled together (invariance), and when $c(T, T') = 0$ they are pushed apart (distinctiveness). For example, in SimCLR, we set $c(T, T') = c((i, g), (i', g')) = \delta_{i=i'}$ to push apart the representations of different images (i, i') while remaining invariant to a transformation pair (g, g') . The *weight function* w is a second binary function that focuses learning on more informative transformation pairs; for instance, SimCLR sets $w(T, T') = \delta_{T \neq T'}$ to

avoid focusing learning invariance to identical transformation $T = T'$ as this is trivially satisfied. Next, we provide a semi-formal analysis of this formulation, leaving the details to Appendix A.1.

Multiple invariances and distinctiveness. The key to extending eq. (1) to $M > 2$ transformation is to build the function $c(T, T')$. We do this one factor a time. If we wish the representation to be distinctive to factor t_m , we set $c(t_m, t'_m) = \delta_{t_m=t'_m}$. If we wish it to be invariant to it, we set $c(t_m, t'_m) = 1$. In lemmas 4 and 5 (Appendix A.1), we show that, given these choices, the only consistent definition of $c(T, T')$ is the product $\prod_{m=1}^M c(t_m, t'_m)$. The intuition is as follows: The representation Φ should distinguish samples $x(T)$ and $x(T')$ if, and only if, at least one of the distinctive factors in T and T' differs.

Forming a batch. Given c , the remaining challenge is to sample training batches $\mathcal{T}$ appropriately. We start by deriving some requirements for $\mathcal{T}$ and then develop a sampling scheme that satisfies them (none of these are guaranteed by sampling T and T' independently). (i) First, in order for eq. (1) not to be identically zero, $c(T, T')$ should be non-zero for at least *some* choices of T and T' in the batch. (ii) Furthermore, when $c(T, T') = 1$, this should not be for the trivial case $T = T'$ (the one that SimCLR discounts by setting $w(T, T') = 0$). Based on the discussion above, the condition $c(T, T') = 1 \wedge T \neq T'$ means that all distinctive factors in T and T' agree and that at least an invariant factor differs. (iii) Additionally, for the fraction in eq. (1) not to be a constant, if $c(T, T') = 1$, there should be another T'' in the batch such that $c(T, T'') = 0$. The latter means that at least one distinctive factor in T and T'' differs.

Short of considering all possible combinations of transformations (which, as explained above, can be statistically inefficient), we can sample a batch $\mathcal{T}$ that satisfies these constraints as follows. We describe this process for the case of $M = 3$ transformations, but note that it extends immediately to any M (this is done in Appendix A.1.1 and A.1.2). First, we sample K_1 versions of the first distinctive transformations t_1 . Then, for each t_1 , we sample K_2 transformations t_2 , also distinctive. Finally, for each choice of (t_1, t_2) , we sample K_3 invariant transformations t_3 .² We thus obtain a batch of $|\mathcal{T}| = K_1 K_2 K_3$ transformations.

This scheme has several desirable properties. First, for every $T = (t_1, t_2, t_3)$, there is another $T' = (t_1, t_2, t'_3)$ that agrees on the distinctive factors and differs in the invariant one (properties (i) and (ii)). Second, there is a $T'' = (t_1, t'_2, t_3)$ or $T'' = (t'_1, t_2, t_3)$ that differs in one or more distinctive factors (property (iii)). Third, the construction is balanced, in the sense that the number of transformations that share a particular factor value t_m is the same

²Note that the sampling ordering is arbitrary; in particular, it needs not to be the same as the ordering in which transformations are applied to the data.

for all values of t_m (this number is $|\mathcal{T}|/(K_1 \cdots K_m)$). Furthermore, SimCLR is obtained as a special case. Please see lemmas 6 and 7 (Appendix A.1) for an in-depth discussion.

Limitations. Despite the benefits, this scheme has also some limitations. The main issue is that a difference in factor t_m generally implies a difference in all subsequent factors as well, meaning that the representation may be unable to observe and thus learn to discriminate changes in all individual factors. In Appendix A.1.3, we show why this is unlikely to be an issue for the practical cases considered here and in the literature. However, we also suggest other practical cases where this *can* be a significant issue, affecting even methods such as SimCLR.

3.2. Properties of Generalized Data Transformations

In this section, we show that GDT’s batch sampling strategy is statistically more efficient than naively-sampled pairs for contrastive learning. We do this by showing that GDT’s objective has the same mean but a lower variance than sampling batches with eq. 1 directly, which would either enumerate all possible pairs of transformations (which is prohibitively expensive) or subsample it by sampling transformations independently. We assume that the distinctive transformations are injective. This must be approximately true, otherwise it is impossible for any method to be distinctive to such transformations. In fact, we can prove the following result:

Theorem 1. *Given a set of transformations $\mathcal{T}$, of which the distinctive transformations are injective, GDT is an unbiased estimate $\hat{\mathcal{L}}$ of the generalized contrastive loss (eq. 1), i.e. $\mathbb{E}[\hat{\mathcal{L}}] = \mathcal{L}(\Phi; \mathcal{T})$. Furthermore, consider a batch of sampled compositions of M transformations, with size $\prod_j^M K_j$, where K_m is the number of samples for the m th transformation. Define $K_I = \prod_{j \in I} K_j$ and $K_V = \prod_{j \in V} K_j$, where I and V are the subsets of indices corresponding to invariant and distinctive transformations, respectively. Denote by $\mathcal{L}_{jj'}$ and $\sigma_{jj'}^2$, the mean and variance of the partial sum of the objective (eq. 1) on the set $\mathcal{X}_j \times \mathcal{X}_{j'}$, with $\mathcal{X}_j = \{x(T^I, T_j^V) : T^I \in \mathcal{T}_I\}$, i.e. the sample pairs corresponding to distinctive transformations with indices j and j' . Then, the variance of the GDT estimate is*

$$\mathbb{V}[\hat{\mathcal{L}}] = \frac{1}{K_V^4 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma_{jj'}^2.$$

The naive estimate’s variance, on the other hand, is

$$\mathbb{V}[\hat{\mathcal{L}}_d] = \frac{1}{K_V^2 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma_{jj'}^2 + \frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} (\mathcal{L}_{jj'} - \mathcal{L})^2,$$

which is larger by a multiplicative factor of K_V^2 and a further additive factor. Proof: See Appendix A. $\square$

This states that sampling data with GDT yields reduced variance, resulting in higher-quality gradients for learning the *same* objective (since the estimate is unbiased), which is reflected empirically in our strong performance on numerous datasets and benchmarks. We note that this may apply to other methods built on the same sampling strategy but which compose transformations in different ways than GDT, as long as the requirements (i-iii) for forming a batch (sec. 3.1) are satisfied.

3.3. Application to video data

As a concrete instantiation of our framework, we consider video data and transformations of the type $T = (t_1, t_2, t_3, t_4, t_5) = (i, \tau, m, r, g)$, as shown in fig. 1, as follows. The first component i **selects** a video in the dataset. We sample $K_i \gg 2$ indices/videos and assume distinctiveness, so that $c(i, i') = \delta_{i=i'}$. The second component τ contrasts different **temporal shifts**. We sample $K_\tau = 2$ different values of a delay τ uniformly at random, extracting a 1s clip $x_{i\tau}$ starting at time τ . For this contrast, we will test the distinctiveness and invariance hypotheses, as [41] indicate that the former may be preferable. The third component m contrasts **modalities**, projecting the video $x_{i\tau}$ to either its visual or audio component $m(x_{i\tau})$. We assume invariance $c(m, m') = 1$ and always sample two such transformations m_v and m_a to extract both modalities, so $K_m = 2$. The fourth component r contrasts **time reversal** [63, 76], which has not previously been explored in a contrastive or cross-modal setting. This is given by a transformation $r \in \mathcal{R} = \{r_0, r_1\}$, where r_0 is the identity and r_1 flips the time dimension of its input tensor, so $K_r = 2$. The final component g applies a spatial and aural **augmentation** $x(T) = g(r(m(x_{i\tau})))$, also normalizing the data. We assume invariance $c(g, g') = 1$ and pick $K_g = 1$, i.e. augment each datum at this level in the sampling hierarchy. These choices lead to $K = K_i K_\tau K_m K_r K_g = 8K_i$ transformations T in the batch $\mathcal{T}$ (in ablations, we also test a subset of these choices).

While we focus on modality splitting, time reversal and shift, note that we could use any transformation that can yield a useful learning signal, such as speed [9, 14, 18, 36, 75, 81] and temporal ordering [20, 43, 53, 80].

Modality splitting. The modality splitting transformation m is useful to capture correlation between modalities [4, 7, 41, 60, 76]. Modality splitting means that the nature of the sample $x(i, \tau, m, r, g)$ is either a sequence of frames ($m = m_v$) or a sound ($m = m_a$). Formally, this means that $x(i, \tau, m, r, g)$ is an element of the direct sum $\mathcal{X}_v \oplus \mathcal{X}_a$ of visual and audio signals; likewise g , r and Φ are defined on this direct sum. In practice, this means that the transformation g comprises a pair of augmentations (g_v, g_a) , where $g_v(v)$ extracts a fixed-size tensor by resizing to a fixed resolution of a random spatial crop of the input video v , and $g_a(a)$ extracts a spectrogram representation of the audio

signal followed by SpecAugment [61] with frequency and time masking. Likewise, $\Phi = (\Phi_v, \Phi_a)$ comprises a pair of neural networks, one for each modality, both valued in $\mathbb{R}^d$ (refer to Appendix A.3.4 for architectural details). In the Appendix A.3.1, we show that modality splitting is key for performance; thus, we extend SimCLR weight function w to focus learning on only cross-modal pairs: $w(T, T') = \delta_{i \neq i'} \cdot \delta_{m \neq m'}$.

3.4. Discussion: utility of GDT

With our framework, we can now generalize current state of the art contrastive learning approaches such as SimCLR in a systematic and practical manner. The theory above and in Appendix A.1 tells us what is the meaning of composing transformations, how a batch should be sampled and why, how this can be achieved by using a hierarchical sampling scheme that extends SimCLR, and what are the limitations of doing so. A particular benefit is to allow to specify individually, for each transformation, if invariance or distinctiveness is sought, whereas previous works lack this distinction and largely considered learning only invariances (SimCLR [13], AVID [54]), or distinctiveness (AoT [76]) to all factors. This property allows the flexible utilization of dataset specific transformations in the case of prior knowledge, or, as we have shown in this study, the exploration of useful signals by enumeration. Finding the best transformation signals can even be further optimized by methods such as Bayesian optimization. Finally, compared to a direct application of previous state-of-the-art methods image-based methods such as SimCLR [13], PIRL [52], and MoCo [31], we can also seamlessly incorporate important cues such as cross-modal correlation, greatly improving downstream performance (see table A.1).

4. Experiments

We compare self-supervised methods on pretraining audio-visual representations. Quality is assessed based on how well the pretrained representation transfers to downstream tasks. We conduct a study on video-audio, as well as video-text unsupervised representation learning to show the generality of our framework and then compare our best setup to the state of the art.

Self-supervised pretraining. For pretraining, we consider two standard pretraining datasets: Kinetics-400 [39] and HT100M [50] and use R(2+1)D-18 [72] and a 2D ResNet [32] as encoders (see Appendix for further details). We also explore how our algorithm scales to even larger, less-curated datasets and train on IG65M [21] as done in XDC [2].

Downstream tasks. To assess the pretrained representation f_v , we consider standard action recognition benchmark datasets, UCF-101 [67] and HMDB-51 [42]. We test the performance of our pretrained models on the tasks of finetuning

Table 1: **Learning hypothesis ablation.** Results on action classification performance on HMDB-51 is shown for finetuning accuracy (Acc) and frozen retrieval (recall@1) after pretraining on Kinetics-400 for 50 epochs. GDT can leverage signals from both invariance and stronger distinctiveness transformation signals. We consider data-sampling (DS), time-reversal (TR) and time-shifting (TS).

	DS	TR	TS	Mod.	Acc.	R@1
<i>SimCLR-like: DS-distinctiveness only</i>						
(a)	d	.	.	V	44.6	11.8
(b)	d	i	.	V	36.9	13.3
(c)	d	.	i	V	35.9	15.3
(d)	d	i	i	V	37.8	13.9
<i>Cross-modal</i>						
(e)	d	.	.	AV	52.4	21.8
(f)	d	i	.	AV	58.8	22.6
(g)	d	.	i	AV	57.4	23.5
(h)	d	i	i	AV	59.9	24.8
<i>Cross-modal +1 distinctive factor</i>						
(i)	d	d	.	AV	57.8	26.1
(j)	d	.	d	AV	58.7	22.1
(k)	d	d	i	AV	61.1	25.4
(l)	d	i	d	AV	61.4	27.1
<i>Cross-modal + 2 distinctive factors</i>						
(m)	d	d	d	AV	57.2	20.5

the pretrained representation, conducting few-shot learning and video action retrieval. The full details are given in the Appendix.

4.1. Analysis of generalized data transformations

In this section, we conduct an extensive study on each parameter of the GDT transformation studied here, $T = (i, \tau, m, r, g)$, and evaluate the performance by finetuning our network and conducting video retrieval on the HMDB-51 action recognition benchmark.

SimCLR-like baseline. First, we use the framework to test a direct extension of SimCLR to video data, as shown in Table 1(a)-(d). By this, we mean utilizing only the visual modality (V), and only invariance to transformations, which is standard in all recent self-supervised methods [13, 31, 77]. For this, we consider GDTs of the type $T = (i, m, \tau, r, g)$ described above and set $K_i = 512$ (the largest we can fit in our setup). In row (a), we pick only the video modality ($m = m_v$ so $K_m = 1$). We also sample a single shift τ (so $K_\tau = 1$), which results in data augmentation but does not learn shift invariance, and no time reversal $r = 1$ (so $K_r = 1$) — these are denoted with a $\cdot$ in the table. However, we do sample two visual augmentations g ($K_g = 2$), emulating SimCLR and learning invariance to that factor.

Table 2: **GDT on video-text HT100M dataset.** We also find the positive effect of including more modalities and find non-trivial combinations of beneficial transformations previously unexplored.

	DS	TR	TS	Mod.	Acc
<i>SimCLR-like</i>					
(a)	d	.	.	V	36.1
<i>Video-text cross-modal</i>					
(b)	d	.	.	VT	59.2
(c)	d	d	.	VT	61.5
(d)	d	.	d	VT	62.9
(e)	d	d	i	VT	63.8
(f)	d	i	d	VT	64.4
(g)	d	d	d	VT	64.4

We also set all transformation components to invariance ($c(t_m, t'_m) = 1$) except the first that does sample selection. In row (b-d) we also experiment with adding invariance to *time shift* (TS) and *time reversal* (TR), by setting $K_\tau = 2$ and $K_r = 2$. We find that doing so consistently degrades the finetuning accuracy performance, but increases the retrieval performance somewhat, indicating that the model is not able to fully leverage these augmentation signals in a meaningful way.

Cross-modal learning. Next, in rows (e-h) we repeat this experiment, but using both audio-visual modalities (AV) by setting $K_m = 2$. In this case, as explained above, we set the weight w to only consider cross-modal interactions and set $K_g = 1$. We note two facts: First, the performance increases substantially (+7.8% (e) vs (a-d)). Second, now TS and TR invariance leads to significant improvements (up to +7.5%).

Invariance vs distinctiveness. Next, in rows (i-l), we explore the effect of being invariant or distinctive to individual transformations, which is unique to our method. Comparing row (h) to rows (k) and (l), we see that switching to distinctiveness to one of TS or TR further improves performance (up to +1.5%). On the other hand, ‘ignoring’ either ($\cdot$ symbols in lines (g) and (j)) is worse than learning invariance ((h) and (l)), with a difference of around 2.5%. Finally, in row (m) we find that being distinctive for both TS and TR at the same time is worse, suggesting that a mix of distinctiveness and invariance is preferable. This is particularly true for the retrieval metric (column R@1).

4.2. Textual modality

In table 2, we demonstrate the generality of our approach by using ASR captions as an alternative modality (instead of audio) for the HowTo100M dataset [50]. For the text encoder, we use a simple Word2Vec [51] embedding with a MLP (further details are provided in the Appendix). Com-

paring table 2(a) with (b), we find that switching from SimCLR to a cross-modal baseline increases performance by more than +22%. Furthermore, we find gains of 3.7% when switching from data-sampling distinctiveness only (row (b)) to incorporating further distinctivenesses (rows c-d). Finally, we find that – as in the video-audio case – combining time-shift distinctiveness with time-reversal invariance leads to particularly strong representations (row (f)), yielding benefits of over +5% compared to data-sampling distinctiveness alone. Compared to video-audio learning (table 1(m)), we find the case of distinctive-only for video-text learning (table 2(g)) to be highly competitive, highlighting the need to explore the set of possible transformation signals to achieve the best downstream performance.

Intuition. While we only analyse a subset of possible transformations for video data, we nevertheless find consistent signals across both video-audio and video-text learning: Inclusion of further distinctivenesses to TS and TR always improve upon the basecase and the best setup is achieved for TS distinctiveness and TR invariance. One explanation for this might be that there is useful signal in both of these transformations that are not captured by previous “augmentation-only” naive noise-contrastive formulations. For example, for time-shift (TS), the model profits from having to differentiate different points in time, e.g. between an athlete running vs an athlete landing in a sandpit, which could be both in the same video. This intuitively serves as a hard negative for the model, increasing its discriminative power. For time reversal (TR), many actions depicted such as moving an object are inherently invariant to reversing time, as shown in [65], therefore yielding a gain when used as an augmentation. In [76], they show that humans have a 20% error-rate when classifying a video’s direction of time in Kinetics-400, thus demonstrating that Kinetics-400 has subsets of videos that look realistic even when reversed. These findings that additional distinctiveness combined with invariances improve video representation learning are noteworthy, as they contradict results from the image self-supervised learning domain, where learning pretext-invariance can lead to more transferable representations [52]. Even when compared to previous self-supervised learning approaches for video-data, such as predicting the arrow of time [76], our method yields new insights by showing that a unique combination of distinctivenesses and invariances performs best, at least on the training sets considered. Combining these points, the strong performance of GDT is founded in its ability to leverage highly informative, yet “free”, signal that we have from construction of the transformations.

4.3. Qualitative analysis

Here, we study what effect the different transformations we let our model be invariant and distinctive to have on our learned representations. For this, we compare against

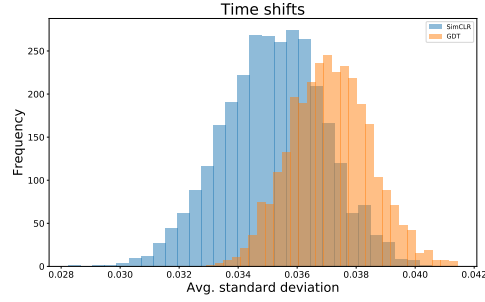


Fig. 3: Learning distinctiveness to time-shifts: our GDT model from Tab.1(j) is able to differentiate features from the same video at different times better than a simple SimCLR variant (Tab.1(a)).

Table 3: **Video retrieval and Few Shot Learning.** Retrieval accuracy in (%) via nearest neighbors at various levels of neighborhood sizes and few shot learning accuracy (%) via a k -nearest neighbor on frozen representations.

		HMDB		UCF	
		1	5	1	5
kNN	Random	3.0	3.5	2.3	4.6
	3DRot [38]	–	–	15.0	31.5
	GDT (ours)	14.3	15.4	26.7	44.6
Retrieval	SP-Net [9]	–	–	13.0	28.1
	VCP [14]	7.6	24.4	18.6	33.6
	M-DPC [29]	7.7	25.7	20.2	40.4
	VSP [14]	10.3	26.6	24.6	41.9
	CoCLR [30]	23.2	43.2	53.3	69.4
	SeLaVi [5]	24.8	47.6	52.0	68.6
	GDT (ours)	26.1	51.7	62.8	79.0

the SimCLR baseline of Tab.1(a) and compare the average standard deviation of the normalized features for 10 time-shifted clips per video for 3000 randomly selected Kinetics-400 validation set.

4.4. Comparison to the state of the art

Given one of our best learning setups from Sec. 4.1 (row (l)), we train for longer and compare our feature representations to the state of the art on standard downstream benchmarks.

4.4.1 Downstream benchmarks

For **few-shot classification**, as shown in table 3, we significantly beat the 3D-Rotnet [38] baseline on UCF-101 by more than 10% on average for each shot with our Kinetics-400 pretrained model.

Table 4: **State-of-the-art on video action recognition with full-finetuning.** Self- and fully-supervisedly trained methods on UCF-101 and HMDB-51 benchmarks.

Method	Data	Top-1 Acc%	
		HMDB	UCF
Supervised [79]	K-400+IN	75.9	96.8
Supervised [2]	K-400	65.1	94.2
AoT [76]	K-400	-	79.4
MultiSensory [59]	K-400	-	82.1
SeLaVi [5]	K-400	47.1	84.2
PEMT [44]	K-400	-	85.2
XDC [2]	K-400	52.6	86.8
AV Sync+RotNet [78]	K-400	54.6	87.0
CoCLR [30]	K-400	54.6	87.9
SeCo [82]	K-400	55.6	88.3
AVTS [41]	K-400	56.9	85.8
CPD [46]	K-400	57.7	88.7
AVID [54]	K-400	60.8	87.5
CM-ACC [47]	K-400	61.8	90.2
GLCM [48]	K-400	61.9	91.2
GDT (ours)	K-400	62.3	90.9
MIL-NCE [49]	HT100M	61.0	91.3
GDT (ours)	HT100M	67.4	94.1
XDC [2]	IG65M	68.9	95.5
GDT (ours)	IG65M	72.8	<u>95.2</u>

For **video retrieval**, we report recall at 1 and 5 retrieved samples for split-1 of the HMDB-51 and UCF-101 datasets in table 3. Using our model trained on Kinetics-400, GDT significantly beats all other self-supervised methods. In particular, we outperform CoCLR [30], a recent state-of-the-art self-supervised method, that uses optical flow as another view to mine hard positive to improve instance discrimination learning for video representations. Moreover, we surpass SeLaVi, an audio-visual clustering and representation learning method, by 2% and 10% on average on recall at 1 and 5 for HMDB-51 and UCF-101.

For **video action recognition**, we finetune our GDT pretrained network for UCF-101 and HMDB-51 video classification, and compare against state-of-the-art self-supervised methods in table 4. When pretrained on the Kinetics datasets, we find that our GDT pretrained model achieves very good results, outperforming all recent methods. In particular, we outperform audio-visual pretraining methods, AVTS [41], SeLaVi [5] and XDC [2], by large margins using the same architecture (R(2+1)D-18) and dataset (Kinetics-400), showing the effectiveness of our GDT pre-training approach. We also surpass AVID [54], the state-of-the-art audio-visual representation learning method, by 1.5% on HMDB-51 and 3.8% on UCF-101. AVID uses a variant of the pre-training

scheme of our baseline approach that extends noise contrastive learning to the audio-visual domain as in Table 1, row (e). However, while AVID simply encodes sample distinctiveness and invariance to modality in its visual representations, we are able to encode invariances and distinctiveness to additional transformations, which significantly improves our performance. Our approach is also more sample efficient, as we are able to achieve our results with 300 less epochs of training. Finally, when pretrained on HT100M, we achieve strong gains of +6.4% on HMDB-51 and +2.8% on UCF-101 compared to the state-of-the-art video text method, MIL-NCE [49]. Similar to AVID, MIL-NCE uses a variant of the baseline cross-modal contrastive framework to learn representations, while we are able to improve upon this baseline by learning invariance and distinctiveness to additional transformations such as time reversal and time shift. Moreover, with HT100M pre-training, we outperform the Kinetics supervised baseline using the same architecture when finetuned on HMDB-51 (67.4 vs 65.1) and are on par for UCF-101 (94.1 vs 94.2). We further show the scalability and flexibility of our GDT framework by pretraining on the IG65M dataset [21]. With this, our visual feature representation sets a new state of the art among all self-supervised methods, particularly by a margin of > 4% on the HMDB-51 dataset. On UCF-101, we set similar state-of-the-art performance with XDC. Along with XDC, we beat the Kinetics supervised pretraining baseline using the same architecture and finetuning protocol.

5. Conclusion

We introduced the framework of Generalized Data Transformations (GDTs), which allows one to capture, in a single noise-contrastive objective, cues used in several prior contrastive and non-contrastive learning formulations, as well as easily incorporate new ones. The framework shows how new meaningful compositions of transformations can be obtained, encoding valuable invariance and distinctiveness that we want our representations to learn. Following this methodology, we achieved state-of-the-art results for self-supervised pretraining on standard downstream video action recognition benchmarks, even surpassing supervised pretraining. Overall, our method significantly increases the expressiveness of contrastive learning for self-supervision, making it a flexible tool for many multi-modal settings, where a large pool of transformations exist and an optimal combination is sought.

Acknowledgements. We are grateful for support from the Rhodes Trust (M.P.), Facebook (M.P.), EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems [EP/L015897/1] (M.P. and Y.A.), the Qualcomm Fellowship (Y.A.), the Open Philanthropy Project (R.F.), and the Royal Academy of Engineering under the Research Fellowship scheme (J.F.H.). We also thank Andrew Owens, Alexei Efros, Triantafyllos Afouras, Gül Varol and Tengda

Han for helpful discussions and feedback. Lastly, the authors would also like to thank Ishan Misra and Bruno Korbar from Facebook for valuable discussions and sharing insights.

References

- [1] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.
- [2] Humam Alwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.
- [3] Ankesh Anand, Evan Racah, Sherjil Ozair, Yoshua Bengio, Marc-Alexandre Côté, and R Devon Hjelm. Unsupervised state representation learning in atari. 2019.
- [4] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.
- [5] Yuki M Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. *NeurIPS*, 2020.
- [6] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.
- [7] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.
- [8] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019.
- [9] Sagie Benaïm, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *CVPR*, 2020.
- [10] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- [11] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, 2019.
- [12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.
- [13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.
- [14] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatio-temporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.
- [15] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.
- [16] Virginia R. de Sa. Learning classification with unlabeled data. In *NeurIPS*, 1994.
- [17] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.
- [18] Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *CVPR*, 2020.
- [19] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proc. CVPR*, 2017.
- [20] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.
- [21] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.
- [22] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *CVPR*, 2020.
- [23] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.
- [24] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.
- [25] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.
- [26] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010.
- [27] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
- [28] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV Workshops*, 2019.
- [29] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.
- [30] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *NeurIPS*, 2020.
- [31] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and

- Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.
- [32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [33] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.
- [34] R Devon Hjelm and Philip Bachman. Representation learning with video deep infomax, 2020.
- [35] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *ICLR*, 2019.
- [36] Simon Jenni, Givi Meishvili, and Paolo Favaro. Video representation learning by recognizing temporal transformations. In *ECCV*, 2020.
- [37] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.
- [38] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.
- [39] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.
- [40] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.
- [41] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-operative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.
- [42] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.
- [43] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.
- [44] Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *ICLR*, 2021.
- [45] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.
- [46] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv preprint arXiv:2001.05691*, 2020.
- [47] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Active contrastive learning of audio-visual video representations. In *ICLR*, 2021.
- [48] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Contrastive self-supervised learning of global-local audio-visual representations, 2021.
- [49] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.
- [50] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.
- [51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [52] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
- [53] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.
- [54] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. 2020.
- [55] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.
- [56] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.
- [57] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, 2017.
- [58] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [59] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
- [60] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.
- [61] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. SpecAugment: A simple data augmentation method for automatic speech recognition. In *INTERSPEECH*, 2019.
- [62] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

- [63] Lyndsey C Pickup, Zheng Pan, Donglai Wei, YiChang Shih, Changshui Zhang, Andrew Zisserman, Bernhard Scholkopf, and William T Freeman. Seeing the arrow of time. In *CVPR*, 2014.
- [64] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020.
- [65] Will Price and Dima Damen. Retro-Actions: Learning ‘close’ by time-reversing ‘open’ videos. 2019.
- [66] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NeurIPS*, 2016.
- [67] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.
- [68] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019.
- [69] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.
- [70] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. 2020.
- [71] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.
- [72] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.
- [73] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(2579-2605):85, 2008.
- [74] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatiotemporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, 2019.
- [75] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020.
- [76] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.
- [77] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.
- [78] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slow-fast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.
- [79] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018.
- [80] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.
- [81] Ceyuan Yang, Yinghao Xu, Bo Dai, and Bolei Zhou. Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*, 2020.
- [82] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. Seco: Exploring sequence supervision for unsupervised representation learning. In *AAAI*, 2021.
- [83] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *Proc. ECCV*, 2016.
- [84] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, 2017.
- [85] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.

A. Appendix

A.1. Foundations of compositional contrastive learning

In this section, we develop more formally a basic theory of compositional contrastive learning formulation, providing rigorous grounds for the approach described in Sec. 3.

Consider the problem of learning a function $f : \mathcal{X} \rightarrow \mathcal{Y}$. In a contrastive setting, we are not given information about the values of f ; instead, we are given a *contrast function* $c : \mathcal{X} \times \mathcal{X} \rightarrow \{0, 1\}$ which only tells for which pairs of points x_1 and x_2 f is the same and for which it differs:

Definition 1. The function f is compatible with the contrast c if, and only if, for all $x_1, x_2 \in \mathcal{X}$:

$$c(x_1, x_2) = \delta_{f(x_1)=f(x_2)}.$$

A contrast function cannot be arbitrary:

Lemma 1. The predicate $c(x_1, x_2) = 1$ is an equivalence relation if, and only if, there exists a function f compatible with c .

Proof. If $c(x_1, x_2) = 1$ defines an equivalence relation on $\mathcal{X}$, then such a function is given by the projection on the quotient $\hat{f} : \mathcal{X} \rightarrow \mathcal{X}/c = \mathcal{Y}$. On the other hand, setting $c(x_1, x_2) = \delta_{f(x_1)=f(x_2)} = 1$ for any given function f is reflexive, symmetric and transitive because the equality $f(x_1) = f(x_2)$ is. $\square$

Definition 2. The contrast function c is admissible if, and only if, $c(x_1, x_2) = 1$ defines an equivalence relation.

Full knowledge of the contrast function c only specifies the level sets of the function f :

Lemma 2. Let f be any function compatible with the admissible contrast c . Then, we can write $f = \iota \circ \hat{f}$ as the composition of an injection $\iota : \mathcal{X}/f \rightarrow \mathcal{Y}$ and the (unique) projection $\hat{f} : \mathcal{X} \rightarrow \mathcal{X}/c$ of $\mathcal{X}$ onto the equivalence classes $\mathcal{X}/c$ of the equivalence relation $c(x_1, x_2) = 1$.

Proof. From elementary algebra, we can decompose any function $f : \mathcal{X} \rightarrow \mathcal{Y}$ as

$$f : \mathcal{X} \xrightarrow{\hat{f}} \mathcal{X}/f \xrightarrow{\iota} \mathcal{Y}$$

where ι is an injective function and $\hat{f}$ projects $\mathcal{X}$ to the quotient $\mathcal{X}/f$, i.e. the collection of subsets $X \subset \mathcal{X}$ where $f(x)$ is constant (level sets). The latter are also the equivalence classes of the relation $f(x_1) = f(x_2)$. Due to definition 1, this is the same equivalence relation given by the contrast c , so that $\mathcal{X}/f = \mathcal{X}/c$. $\square$

Note that, in our contrastive learning formulation, we do not define the contrast c on the sample space $\mathcal{X}$, but rather on the transformation space $\mathcal{T}$. The following lemma suggests that defining a contrast $c(T, T')$ on transformations instead of data samples is usually acceptable:

Lemma 3. Let $c : \mathcal{T} \times \mathcal{T} \rightarrow \{0, 1\}$ be an admissible contrast function defined on a set (e.g., a batch) of generalized data transformations $\mathcal{T}$. Furthermore, let x be a dataset and let $x(T) \in \mathcal{X}$ be the sample indexed by transformation T . If $x(T) = x(T') \Rightarrow T = T'$ (i.e. different transformations output different samples), then setting $\tilde{c}(x(T), x(T')) = c(T, T')$ defines part of an admissible sample contrast function $\tilde{c}$ on $\mathcal{X}$.

Proof. The expression above defines $\tilde{c}$ on the sample space $\tilde{\mathcal{X}} = \{x(T) : T \in \mathcal{T}\} \subset \mathcal{X}$. Reflexivity, symmetry and transitivity are inherited from c . However, if the same data point $x(T) = x(T')$ can be obtained from two different transformations T and T' , the definition is ill posed. The hypothesis in the lemma guarantees that this is not the case. $\square$

A.1.1 Compositional transformations

Next, we consider the case in which $T = (t_1, \dots, t_M)$ is a composition of individual transformations t_m , each with its own contrast t_m :

Definition 3. We say that a contrast function $c(t_m, t'_m)$ is distinctive if it is given by $\delta_{t_m=t'_m}$. We say that it is invariant if it is identically one.

The following lemma provides a formula for the overall contrast function $c(T, T')$ given the contrasts for the individual factors.

Lemma 4. Let $c(t_m, t'_m) = 1$ be admissible contrast functions, either distinctive or invariant. Then, the product $c(T, T') = \prod_{m=1}^M c(t_m, t'_m)$ is also admissible.

Proof. The reflexive and symmetric properties are obviously inherited. For the transitive property, note that $c(T, T') = 1$ if, and only if, $\forall m : c(t_m, t'_m) = 1$. Hence:

$$\begin{aligned} c(T, T') &= c(T', T'') = 1 \\ &\Rightarrow \forall m : c(t_m, t'_m) = c(t'_m, t''_m) = 1 \\ &\Rightarrow \forall m : c(t_m, t''_m) = 1 \Rightarrow c(T, T'') = 1. \end{aligned}$$

$\square$

Finally, we show that, essentially, the formula above is the only reasonable one. For this, we only require $c(T, T')$ to be monotonic in the individual factors; i.e., if more factors become 1, then $c(T, T')$ can only grow:

Definition 4. We say that $c(T, T')$ is monotonic in the individual factors if, and only if, for any three transformations T, T', T'' such that $c(t_m, t'_m) \leq c(t_m, t''_m)$ for all the factors, then we also have $c(T, T') \leq c(T, T'')$.

Next, we show that c can only have a very limited form:

Lemma 5. Suppose that the admissible monotonic contrast $c(T, T')$ is expressible solely as a function of the individual admissible contrasts $c(t_m, t'_m)$ for $m = 1, \dots, M$. Then, up to a permutation of the transformations, we can always write

$$c(T, T') = \prod_{i=1}^m c(t_i, t'_i)$$

where $0 \leq m \leq M$. In particular, $m = M$ is the only option that is guaranteed not to ignore some of the factors.

Proof. From the assumptions, we can write

$$c(T, T') = h \circ v(T, T')$$

where h is a function of the binary vector

$$v(T, T') = (c(t_1, t'_1), \dots, c(t_M, t'_M)) \in \mathbb{B}^M.$$

Furthermore, since invariant factors are constant, they do not affect the function; hence, without loss of generality we can assume that all factors are distinctive.

Since all factors are distinctive, we can construct two transformations $T' = (t'_1, \dots, t'_M)$ and $T'' = (t''_1, \dots, t''_M)$ such that $v(T', T'') = (0, \dots, 0)$ (i.e., all the contrasts $c(t'_m, t''_m)$ are null). If $c(T', T'') = 1$, then, due to monotonicity, $c(T', T'')$ is identically 1 and the lemma is proved for $m = 0$.

If not, let $c(T', T'') = h(0, \dots, 0) = 0$. Then, for any given binary vector v , we can construct a transformation $T = (t_1, \dots, t_M)$ such that $v(T, T') = v$ and $v(T, T'') = \bar{v}$ as follows:

$$t_m = \begin{cases} t'_m, & \text{if } v_m = 1, \\ t''_m, & \text{otherwise.} \end{cases}$$

We cannot have $c(T, T') = h(v) = h(\bar{v}) = c(T, T'') = 1$; otherwise, due to the transitivity of c , we would have $c(T', T'') = h(0, \dots, 0) = 1$, which contradicts our assumption. Hence, h must partition the space of binary vectors in two halves, the ones for which $h(v) = 1$ and their complements $h(\bar{v}) = 0$.

Now let v be a vector with the minimal number of 1 such that $h(v) = 1$. Again without loss of generality, we can assume this is of the type $v = (1, \dots, 1, 0, \dots, 0)$ with m ones in front. Due to monotonicity, all vectors of type $v' = (1, \dots, 1, v_{m+1}, \dots, v_M)$ must also have $h(v') = 1$; by taking their complement, the previous result shows that all vectors $v'' = (0, \dots, 0, v_{m+1}, \dots, v_M)$ must have $h(v'') = 0$. This is also the case for any vector of the type

$(v_1, \dots, v_m, 0, \dots, 0)$ where any $v_i = 0$ for $1 \leq i \leq m$ (because m is the minimum number of ones required for $h(v) = 1$). We conclude that $h(v) = 1$ if, and only if, $(v_1, \dots, v_m) = (1, \dots, 1)$. $\square$

A.1.2 Forming batches

Let $\hat{\mathcal{T}}_1 \times \dots \times \hat{\mathcal{T}}_M$ be a composite space of generalized data transformations, so that data points are indexed as $x(t_1, \dots, t_M)$. Furthermore, let $c(t_m, t'_m)$ be corresponding admissible contrast functions and let $c(T, T')$ be their product, as in lemma 4. As before, we assume that the functions are of two kinds:

- invariant: $c(t_m, t'_m) = 1$.
- distinctive: $c(t_m, t'_m) = \delta_{t_m=t'_m}$.

Let $I \subset \{1, \dots, M\}$ be the subset of indices m corresponding to the invariant transformations and $D = \{1, \dots, M\} \setminus I$ the distinctive ones.

Let $\text{sample}(\hat{\mathcal{T}}_m; K_m)$ be a stochastic operator that samples $K_m \leq |\hat{\mathcal{T}}_m|$ transformations from $\hat{\mathcal{T}}_m$ without replacement. We sample a batch recursively:

- Let $\mathcal{T}_1 = \text{sample}(\hat{\mathcal{T}}_1; K_1)$
- Let $\mathcal{T}_m = \bigcup_{T \in \mathcal{T}_{m-1}} T \circ \text{sample}(\hat{\mathcal{T}}_m; K_m)$

At each level of the recursion, each transformation is extended by sampling K_m more transformations (note that no two identical transformations can be generated in this manner). Hence $|\mathcal{T}_M| = K_1 \cdots K_M$.

Lemma 6. There are exactly $(\prod_m K_m)(\prod_{m \in I} K_m)$ pairs of transformations $(T, T') \in \mathcal{T}_M \times \mathcal{T}_M$ for which $c(T, T') = 1$. Of these, exactly $\prod_m K_m$ are trivial pairs ($T = T'$). Hence, there are $(\prod_m K_m)(\prod_{m \in I} K_m - 1)$ non-trivial pairs for which $c(T, T') = 1$.

Lemma 7. For each $T \in \mathcal{T}_M$, there are exactly $(\prod_m K_m) - (\prod_{m \in I} K_m)$ pairs (T, T'') such that $c(T, T'') = 0$.

For example, in SimCLR $M = 2$, $D = \{1\}$, $I = \{2\}$, $K_1 = B/2$, $K_2 = 2$, $|\mathcal{T}_2| = B$, there are $B(2 - 1) = B$ non-trivial pairs of transformations for which $c(T, T') = 1$, and, for each T , there are $B - 2$ pairs of transformations for which $c(T, T'') = 0$.

The lemmas above suggest that we should pick $K_m \geq 2$ for at least one invariant factor and at least $K_m \geq 2$ for at least one distinctive factor, as otherwise eq. (1) is degenerate.

A.1.3 Limitations

In general, we want more restrictive requirements than the one described above. When learning f , difficult (and therefore interesting) cases amount to: learning to be sensitive

to ‘small’ variations in the distinctive factors and learn to be insensitive to ‘large’ variations in the invariant factors. For the former, we would like f to observe variations in a single distinctive factor at a time, as these are the ‘smallest’. For these individual variations to exist at all in the batch, we should choose $K_m \geq 2$ for all distinctive factors $m \in D$.

Even so, the hierarchical scheme in general *prevents* us from observing all individual variations. In fact, suppose that two transformations T and T' in $\mathcal{T}_M$ differ for factor m (i.e. $t_m \neq t'_m$). Then, the remaining factors $t_{m+1} \neq t'_{m+1}, \dots$ also differ in general because successive transformations are sampled independently in different branches of the tree. This means that we cannot, in general, observe a change in t_m *alone*, so the function f may not learn to be distinctive to this ‘minimal’ change in isolation.

Note that this is a limitation that affects our sampling scheme as well as existing methods such as SimCLR. Fortunately, in practice this is often not an issue. There are in fact two mitigating factors, which apply to most existing formulations, including the new ones presented here.

First, some transformations spaces $\hat{\mathcal{T}}_m$ are very small, and in fact binary (e.g., modality splitting, time reversal). In this case, $K_m = 2$ means that transformations are sampled exhaustively, so for level m the hierarchical sampling scheme does extract all possible combinations of transformations.

Second, in other cases the issue is moot due to the nature of the transformations and the data. For instance, in SimCLR the first transformation t_1 amounts to sampling a certain image x_i , and the second transformation t_2 amounts to sampling two data augmentation $g_{1i}(x_i)$ and $g_{2i}(x_i)$, different for each image. The issue here is that we cannot observe a change in the index i for the same augmentation $g(x_1)$ and $g(x_2)$, as these data points do not exist in the batch. This means that the representation f can only learn to distinguish two different images x_i that also have two different augmentations applied to them. Because of the particular nature of the training data (ImageNet) this is likely irrelevant since different images x_i are unrelated in any case, so applying transformations does not significantly alter their statistical relationships.

However, note that this is not *always* the case. For instance, if SimCLR was applied to a dataset of pre-aligned faces (for the purpose of learning face recognition), then being unable to contrast different faces $g(x_1)$ and $g(x_2)$ under the same transformation g would make negative pairs far too easy to discriminate.

A.2. Reduction in variance theorem

A.2.1 Proof of theorem 1

For ease of notation, we will express eq. 1 as the expected value of a loss function ℓ , which subsumes the weight (w),

contrast (c), feature extractor (Φ) and log-softmax functions:

$$\mathcal{L} = \mathbb{E}_{T, T' \sim \hat{\mathcal{T}}} [\ell(x(T), x(T'))]. \quad (2)$$

The expectation is over pairs of transformations in $\hat{\mathcal{T}} = \hat{\mathcal{T}}_1 \times \dots \times \hat{\mathcal{T}}_M$, the space of all compositions of transformations, which can be applied to the data x . Note that eq. 1 contains a sum over a third transformation (T'') to compute the softmax’s normalization, which is also subsumed by ℓ in eq. 2 as this third transformation is not essential for the rest of the proof. We will separate each transformation into invariant and distinctive parts, $T = (T^I, T^V)$ respectively with $T^I \in \hat{\mathcal{T}}_I$ and $T^V \in \hat{\mathcal{T}}_V$ (see sec. A.1.2). Note that this separation is merely a notational convenience; the individual transformations can be applied to the data in *any* order, with $x(T^I, T^V) = x(t_1 \circ \dots \circ t_M)$, and each t_i belonging to either T^I or T^V . Then, eq. 2 becomes:

$$\mathcal{L} = \mathbb{E}_{T^I, T'^I \sim \hat{\mathcal{T}}_I, T^V, T'^V \sim \hat{\mathcal{T}}_V} [\ell(x(T^I, T^V), x'(T'^I, T'^V))].$$

Consider a mini-batch of data sample pairs and their associated transformation compositions, $\mathcal{B}_{\text{direct}} = \{T_i^I, T_i^V, T'_i{}^I, T'_i{}^V\}_{i=1}^{K_I^2 K_V^2}$, sampled as $T_i^I, T'_i{}^I \sim \hat{\mathcal{T}}_I$ and $T_i^V, T'_i{}^V \sim \hat{\mathcal{T}}_V$. The batch size is a function of $K_I = \prod_{j \in I} K_j$ and $K_V = \prod_{j \in V} K_j$, the number of sampled invariant and distinctive transformations in our method, respectively. The batch size of $K_I^2 K_V^2$ was chosen to allow a direct comparison. The expected value of the loss over this batch is then the simple empirical average:

$$\hat{\mathcal{L}}_d = \frac{1}{K_I^2 K_V^2} \sum_i \ell(x(T_i^I, T_i^V), x(T'_i{}^I, T'_i{}^V)). \quad (3)$$

Now consider the domain of transformed samples $\mathcal{X} = \{x(T^I, T^V) : T^I \in \hat{\mathcal{T}}_I, T^V \in \hat{\mathcal{T}}_V\}$. Due to the assumed injectivity of all $t \in T^I$, we may partition the domain using one partition $\mathcal{X}_j = \{x(T^I, T_j^V) : T^I \in \hat{\mathcal{T}}_I\}$ per distinctive transformation T_j^V , i.e.: $\mathcal{X} = \cup_j^{K_V} \mathcal{X}_j$, with $\mathcal{X}_j \cap \mathcal{X}_{j'} = \emptyset, \forall j, j'$. The probability distribution of the samples has density $p(T^I, T^V)$, and the density in each partition is thus $p_j(T^I) = K_V p(T^I) \delta_{T^I \in \mathcal{X}_j}$, with δ the indicator function.

GDT can then be interpreted as a stratified sampling method, with one stratum (partition) per *pair* of distinctive transformations. The domain being sampled by the expectation in eq. 2 is $\mathcal{X}^2 = \cup_{jj'}^{K_V, K_V} \mathcal{X}_j \times \mathcal{X}_{j'}$, and stratified sampling consists of sampling an equal number of K_I^2 sample pairs from each of the K_V^2 partitions:

$$\hat{\mathcal{L}} = \frac{1}{K_V^2 K_I^2} \sum_{ii'jj'}^{K_I, K_I, K_V, K_V} \ell(x(T_i^I, T_j^V), x(T_{i'}^I, T_{j'}^V)). \quad (4)$$

Note the subtle difference from eq. 3 in the summation ranges, and that the *same* samples and transformations are

reused for both elements of each pair, instead of being sampled independently to fill a mini-batch. To make the following derivations easier, note that we can equivalently express eq. 4 as:

$$\hat{\mathcal{L}} = \frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} \hat{\mathcal{L}}_{jj'},$$

with $\hat{\mathcal{L}}_{jj'} = \frac{1}{K_I^2} \sum_{ii'}^{K_I, K_I} \ell(x(T_i^I, T_j^V), x(T_{i'}^I, T_{j'}^V))$. We will first show that this pairwise stratified sampling is an unbiased estimate of eq. 2:

$$\begin{aligned} \mathbb{E}[\hat{\mathcal{L}}] &= \frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} \mathbb{E}[\hat{\mathcal{L}}_{jj'}] \\ &= \frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} \mathcal{L}_{jj'} \\ &= \mathcal{L}, \end{aligned}$$

where we use the expectation $\mathcal{L}_{jj'}$ of the loss function evaluated on the partition jj' (corresponding to distinctive transformations with indices j and j'), as $\mathcal{L}_{jj'} = \mathbb{E}_{T^I \in \mathcal{X}_j, T'^I \in \mathcal{X}_{j'}} [\ell(x(T^I, T_j^V), x(T'^I, T_{j'}^V))]$.

Similarly, we can also define each partition's loss variance $\sigma_{jj'}^2 = \mathbb{V}_{T^I \in \mathcal{X}_j, T'^I \in \mathcal{X}_{j'}} [\ell(x(T^I, T_j^V), x(T'^I, T_{j'}^V))]$. Then, from eq. 4 we obtain directly

$$\begin{aligned} \mathbb{V}[\hat{\mathcal{L}}] &= \frac{1}{K_V^4} \sum_{jj'}^{K_V, K_V} \mathbb{V}[\mathcal{L}_{jj'}] \\ &= \frac{1}{K_V^4 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma_{jj'}^2. \end{aligned}$$

As a point of comparison, the variance of the direct sampling estimate is:

$$\begin{aligned} \mathbb{V}[\hat{\mathcal{L}}_d] &= \frac{1}{K_V^2 K_I^2} ((\mathbb{E}_{(T^I, T'^I) \in \mathcal{X}^2} [\ell^2(x(T^I, T^V), x(T'^I, T'^V)) - \mathcal{L}^2]) \\ &= \frac{1}{K_V^2 K_I^2} \left(\frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} \mathbb{E}_{T^I \in \mathcal{X}_j, T'^I \in \mathcal{X}_{j'}} [\ell^2(x(T^I, T_j^V), x(T'^I, T_{j'}^V)) - \mathcal{L}^2] \right) \\ &= \frac{1}{K_V^2 K_I^2} \left(\frac{1}{K_V^2} \sum_{jj'}^{K_V, K_V} (\mathcal{L}_{jj'}^2 + \sigma_{jj'}^2) - \mathcal{L}^2 \right) \\ &= \frac{1}{K_V^4 K_I^2} \sum_{jj'}^{K_V, K_V} (\sigma_{jj'}^2 + (\mathcal{L}_{jj'} - \mathcal{L})^2) \end{aligned}$$

$$\geq \frac{1}{K_V^4 K_I^2} \sum_{jj'}^{K_V, K_V} \sigma_{jj'}^2$$

completing the proof. $\square$

A.3. Additional experimental results

A.3.1 Modality ablation

In table A.1, we provide the results of running our baseline model (sample-distinctiveness only) within-modally instead of across modalities and find a sharp drop in performance.

Table A.1: **Within vs cross-modal learning.** Results on action classification performance on HMDB-51 and UCF-101 is shown for finetuning accuracy (Acc) and frozen retrieval (recall@1) after pretraining on Kinetics-400 for 50 epochs.

	HMDB		UCF	
	Acc.	R@1	Acc.	R@1
Within-modal	37.8	13.9	76.4	28.0
Cross-modal	52.4	21.8	87.6	54.8

A.3.2 Dataset details

The Kinetics-400 dataset [39] is human action video dataset, consisting of 240k training videos, with each video representing one of 400 action classes. After filtering out videos without audio, we are left with 230k training videos, which we use for pretraining our model.

HT100M [50] is a large-scale instructional video collection of 1.2 million Youtube videos, along with automatic speech recognition transcripts. There are more than 100 million clips (ASR segments) defined in HowTo100M.

HMDB-51 [42] consists of 7K video clips spanning 51 different human activities. HMDB-51 has three train/test splits of size 5k/2k respectively.

UCF-101 [67] contains 13K videos from 101 human action classes, and has three train/test splits of size 11k/2k respectively.

IG65M [21] is a large-scale weakly supervised dataset collected from a social media website, consisting of 65M videos of human action events. We use the all the videos in the dataset for pretraining.

A.3.3 Preprocessing details

The video inputs are 30 consecutive frames from a randomly chosen starting point in the video. These frames are resized such that the shorter side is between 128 and 160, and a center crop of size 112 is extracted, with color-jittering applied. A random horizontal flip is then applied with probability 0.5, and then the inputs' channels are z-normalized using

mean and standard deviation statistics calculated across each dataset.

One second of audio is processed as a $1 \times 40 \times 99$ image, by taking the log-mel bank features with 40 filters and 99 time-frames after random volume jittering between 90% and 110% is applied to raw waveform, similar to [4]. The spectrogram is then Z-normalized, as in [41]. Spec-Augment is then used to apply random frequency masking to the spectrogram with maximal blocking width 3 and sampled 1 times. Similarly, time-masking is applied with maximum width 6 and sampled 1 times.

For the text, we remove stop words from the narrations as in [50]. For each narration, we take a maximum of 16 consecutive words covering a max duration of 4 seconds as in [49].

A.3.4 Pretraining details

We use R(2+1)D-18 [72] as the visual encoder f_v and ResNet [32] with 9 layers as the audio encoder f_a unless otherwise noted; both encoders produce a fixed-dimensional output (512-D) after global spatio-temporal average pooling. For the text encoder, we use the Google News self-supervised pre-trained word2vec (d=300) embedding [51], that is linearly projected to 2048D and max-pooled as in [49]. After the inputs are encoded by their respective modality encoders, the vectors are then passed through two fully-connected layers with intermediate size of 512 to produce 256-D embeddings as in [8] which are normalized by their L2-norm [77]. The embedding is used for computing the contrastive loss, while for downstream tasks, a linear layer after the global spatio-temporal average pooling is randomly initialized. For NCE contrastive learning, the temperature ρ is set as $1/0.07$. For optimizing these networks, we use SGD. The SGD weight decay is 10^{-5} and the SGD momentum is 0.9. We use a mini-batch size of 8 on each of our 64 GPUs giving an effective batch size of 512 for distributed training. The initial learning rate is set to 0.01 which we linearly scale with the number of GPUs, after following a gradual warm-up schedule for the first 10 epochs [24]. For Kinetics, we train for 100 epochs (3 days), while for HT100M, we train for 40 epochs (3 days).

A.3.5 Ablation experiment details

For the ablations, we only pretrain for 50 epochs on the Kinetics-400 dataset, and 20 epochs on the HT100M dataset, since it is a much larger dataset.

For downstream evaluation, we only evaluate on the first fold of HMDB-51 but found the performance between folds to be close (within 1%).

A.3.6 Evaluation details

All evaluation code is provided in the Supplementary Material.

Video Action Recognition. During training, we take 10 random clips of length 32 frames from each video. For video clip augmentations, we follow a standard protocol as in [41]. During evaluation, we uniformly sample 10 clips from each video, average softmax scores, and predict the class having the highest mean softmax score. We then measure the mean video top-1 accuracy across all videos and all official folds. During training, we use SGD with initial learning rate 0.0025, which we gradually warm up to $2 \cdot 10^{-2}$ in the first 2 epochs. The weight decay is set to $5 \cdot 10^{-3}$ and momentum to 0.9. We use a mini-batch size of 32 and train for 12 epochs with the learning rate multiplied by $5 \cdot 10^{-2}$ at 6 and 10 epochs. We compare our GDT pretrained model with both self-supervised methods, and supervised pretraining, and report average top-1 accuracies on UCF101 and HMDB-51 action recognition task across three folds in table 4.

Few-shot classification We follow the protocol in [38] and evaluate our our GDT pretrained network using few-shot classification on the UCF-101 dataset, and additionally on HMDB-51. We randomly sample n videos per class from the train set, average the encoder’s global average pooled features from ten clips per training sample and measure classification accuracy performance on the validation set using a k -nearest neighbor classifier, with k set to 1.

Video Retrieval. We follow the standard protocol as outlined in [80]. We use the split 1 of UCF101, and additionally HMDB-51. We uniformly sample 10 clips per video, and average the max-pooled features after the last residual block for each clip per video. We use these averaged features from the validation set to query the videos in the training set. The cosine distance of representations between the query clip and all clips in the training set are computed. When the class of a test clip appears in the classes of k nearest training clips, it is considered to be correctly predicted. We report accuracies for $k = 1, 5, 20$ and compare with other self-supervised methods on UCF101 and HMDB-51 in table 3.

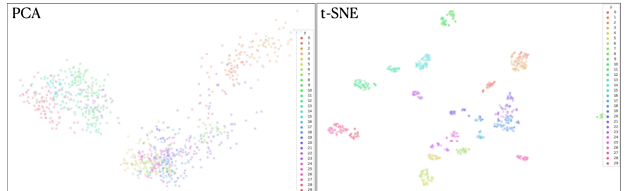


Fig. A.1: Feature visualizations with PCA and t-SNE on 30 videos of a single, random class of HMDB-51. For each video, we sample 10 temporal clips and encode video-ID with color. Embeddings are generated from our time-shift distinct model (Tab.1 (I)).

A.3.7 Additional Qualitative analysis

In fig. A.1, we present a PCA and t-SNE [73] plots of the features obtained by our model (DS-d, TR-d, TS-d) (Tab. 1, row (I)). We observe that even comparing to videos of the same action category, the individual clips are well separated, showing that the model is learning to distinguish between different time intervals.

8

Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning

**This work will be presented at the International Conference
on Computer Vision (ICCV) 2021.**

Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning.

Mandela Patrick*, Po-Yao Huang*, Ishan Misra, Florian Metze, Andrea Vedaldi
Facebook AI Research

mandelapattrick, berniehuang, imisra, fmetze, vedaldi@fb.com

Yuki M. Asano*, João Henriques
Oxford University

yuki, joao@robots.ox.ac.uk

Abstract

The quality of the image representations obtained from self-supervised learning depends strongly on the type of data augmentations used in the learning formulation. Recent papers have ported these methods from still images to videos and found that leveraging both audio and video signals yields strong gains; however, they did not find that spatial augmentations such as cropping, which are very important for still images, work as well for videos. In this paper, we improve these formulations in two ways unique to the spatio-temporal aspect of videos. First, for space, we show that spatial augmentations such as cropping do work well for videos too, but that previous implementations, due to the high processing and memory cost, could not do this at a scale sufficient for it to work well. To address this issue, we first introduce *Feature Crop*, a method to simulate such augmentations much more efficiently directly in feature space. Second, we show that as opposed to naïve average pooling, the use of transformer-based attention improves performance significantly, and is well suited for processing feature crops. Combining both of our discoveries into a new method, *Space-Time Crop & Attend (STiCA)* we achieve state-of-the-art performance across multiple video-representation learning benchmarks. In particular, we achieve new state-of-the-art accuracies of 67.0% on HMDB-51 and 93.1% on UCF-101 when pre-training on Kinetics-400. Code and pretrained models are available¹.

1. Introduction

Visual representations have evolved significantly in the last two decades. The first generation of representations

*Equal contribution.

¹<https://github.com/facebookresearch/GDT>

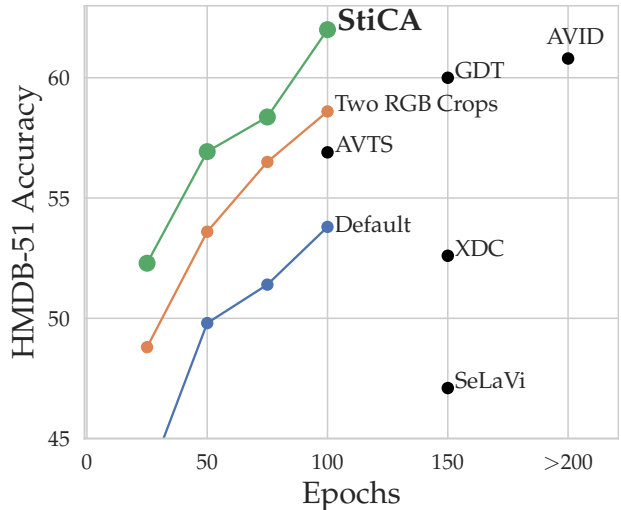


Figure 1: HMDB-51 accuracy vs epoch. Our method, **STiCA**, combines space-time crops in feature space with self-attention of time in latent space. This yields significant benefits not only in performance but also in speed compared to cropping in input space using *two RGB crops*, or simply using the *default* cross-modal only loss. Compared to recent state-of-the-art cross-modal self-supervised learning methods (XDC [6], GDT [106], AVID-CMA [97], SeLaVi [9]) pre-trained on Kinetics-400 [69] STiCA is able to achieve significantly better results in fewer epochs.

comprises algorithms such as SIFT [87] and HOG [30] that were designed manually. The second generation comprises representations learned from data by using deep neural networks and manual supervision [31, 59, 76]. We are now transitioning to the third generation, where representations are learned from data without using any manual annotations by means of self-supervision. Current self-supervised rep-

representations, obtained from methods such as MoCo [57], SimCLR [24] or SwAV [20], convincingly outperform supervised ones on downstream tasks such as image classification, segmentation and object detection. Furthermore, most of these methods are based on noise-contrastive instance discrimination, which was proposed in ExemplarCNNs [39] and put in its current form in [142] and [101]. The idea is to learn representations that are invariant to irrelevant factors of variations, modelled by strong augmentations such as image cropping, while remaining distinctive for the identity of the image.

Noise-contrastive learning is of course not limited to still images. In particular, a number of recent approaches [54, 93, 97, 106] have used noise-contrastive formulations to learn visual or audio-visual representations. However, these methods are not as well developed as their counterparts for still images, with current state-of-the-art methods [54, 106] still lagging behind their supervised counterparts.

In this paper, we identify two areas in which current video representation learning formulations are lacking and improve on them, thus significantly improving upon the current state of the art in this area.

The first shortcoming is the lack of a sufficient encoding of **spatial invariances**. For still images, learning spatial invariances has been shown to be one of the most important factors for performance [20, 24]. Almost all methods achieve some form of spatial invariance simply by applying different spatial augmentations to the images in different epochs of training. However, learning spatial invariances in this manner requires a slow training process that lasts for many epochs (~ 800). Authors have suggested that packing several augmentations of the same image in a single data batch is more effective as it provides a much stronger and more direct incentive for the network to learn invariances [20].

For videos, both strategies are less feasible. Training a model for 200 epochs on Kinetics-400 [69] already requires around 1.5K GPU hours on recent Nvidia V100 architectures, and with recent datasets such as IG65M [45] and HowTo100M [94] only a handful of epochs can realistically be completed. On the other hand, including multiple augmentations of the same video in a batch rapidly exhausts the memory of GPUs. Since batch sizes per GPU are already in the single digits due to the size of video data, including several augmentations is unfeasible. This is particularly detrimental for recent contrastive learning approaches such as [24, 58], where reducing the batch size means reducing the pool of negative contrastive samples.

In order to solve this problem, we propose to move spatial augmentations to the feature space, in a manner specifically tailored to contrastive learning. Instead of extracting a large number R of different augmentations in the input RGB space, we extract only two of them, apply the trunk of the

neural network to extract corresponding features, and then extract $R/2$ more augmentations directly in feature space. In this way, one needs to evaluate the slow and memory taxing feature extraction part of the network only twice, regardless of the number of augmentations that are produced. We show that this *feature-level augmentation* significantly improves representation learning performance.

The second challenge that we tackle is how to best encode **temporal information** in self-supervised video representation learning. Currently, most self-supervised video representation learning approaches use 3D-CNNs [21, 132, 133, 144] that compute convolutions across space and time, but the final representation is generated by naïve global average pooling over space and time, crucially discarding temporal ordering.

In order to address this shortcoming, in this work we propose to use a contextualized pooling function based on the transformer architecture [135] for both self-supervised pretraining and supervised finetuning. The intuition is that, via multi-head self-attention, the transformer can capture temporal dependencies much better than average pooling, especially for longer inputs. Transformers can also benefit from our feature-level crops, as the latter resemble the common approach of randomly masking the inputs to the transformer [62]. Experimental results show that this modification improves the performance of the learned video representations substantially, and is cumulative with the benefit of feature crops, at about the same cost of average pooling.

We combine both of our proposed improvements into a new self-supervised learning approach: Space-Time Attention and Cropping (**STiCA**). To summarize, with STiCA we make the following three main contributions:

- We demonstrate the benefits of stronger spatial invariances in self-supervised video representation learning for the first time and we propose feature-level augmentation to implement the latter efficiently.
- We propose to use transformers to model time more effectively in self-supervised video representations, replacing average as the pooling function.
- We demonstrate strong performance gains by using the two techniques and obtain state-of-the-art performance on two standard benchmarks (67.0% on HMDB-51 and 93.1% on UCF-101).

2. Related Works

Self-supervised Image Representation Learning. Self-supervised learning uses pretext tasks to automatically and easily generate differentiable learning signals from the data itself in order to train convolutional neural networks. A variety of pretext tasks have been proposed such as colorization [148, 149], predicting artificial rotations [46], in-

painting [105], spatial context [35, 100], and clustering features [11, 18, 19, 20, 64, 83]. Recently, contrastive methods [50, 51] have proven to be particularly effective at learning transferable image representations [13, 24, 49, 57, 95, 101, 130].

Self-supervised Video Representation Learning. For videos, pretext tasks often seek to leverage the temporal dimension to learn representations. Such tasks include predicting clip and sequence order [79, 96, 145], future events [52, 53], the arrow of time [140], 3D geometric transformations [65, 71], playback speed [14, 40, 63, 137], or motion statistics [136].

Multi-Modal Learning. The co-occurrence and synchronicity of multiple modalities from videos have been used to learn visual representations from both audio-video [6, 7, 9, 74, 90, 97, 102, 106], and speech-video [5, 73, 85, 92, 93, 98, 107, 123, 125, 126] data. Multi-modal representation learning has several practical applications: lip reading [3, 26, 27], audio-visual source separation and localization [2, 4, 8, 56, 150, 151], speech recognition [1, 111], efficient inference [43, 75], egocentric action recognition [70] and audio-visual navigation [22].

Data Augmentations. Data augmentation has proven to be useful in training deep learning models in many domains, from vision [28, 29, 146] to speech [103]. Data transformations are the foundation of most self-supervised works, and there has been early attempts to even learn the optimal distribution of transformations [16, 29]. Particularly for contrastive learning, the choice of data transformations has been shown to be particularly important to learn desirable invariances and equivariances [95, 106, 130, 131].

Transformations in Feature-Space. Some works have proposed forms of augmentation in feature-space, by adding noise and linear transformations [129], and by associating samples to prototypes in feature-space [78]. These augmentations do not correspond to interpretable geometric operations, however. Crops in feature-space are commonly used in supervised detection pipelines, such as Faster R-CNN and region-based architectures [115], and in earlier detectors based on manually-engineered features [30]. However, the objective of these transformations is to enumerate a space of outputs (e.g. bounding box predictions) for supervised prediction. In self-supervised learning, while [66] uses feature mixing to create harder negatives for contrastive learning, we are instead interested in using feature crop augmentation to achieve spatial invariance.

Temporal Modeling. Videos extend images by adding a temporal dimension. Therefore, there has been a large family of research that has looked into how to model temporal information in videos. Early works incorporated temporal information via average pooling of frame/clip-level

features [48, 68, 138], while later work used 3D convolution neural networks [132, 133, 144] and recurrent-neural networks [37]. Other approaches leverage long-term temporal convolutions [134], self-attention [139], relation networks [152], multi-scale temporal convolutions [61], or optical flow in a two stream network [119].

Transformers in Vision. With the success of the transformer architecture [135] in natural language processing [62], transformers are being used in various vision domains such as image representation learning [23, 32, 38, 118, 141], image generation [104], object detection [17, 86], few-shot learning [36], video action recognition [15, 47, 99, 139], video question-answering [67], image-text [84, 88, 124, 127, 128] and video-text [42, 73, 107, 125, 126, 154] representation learning.

3. Method

Our goal is to learn a general-purpose *data representation* $\Phi : \mathcal{X} \rightarrow \mathcal{Z} = \mathbb{R}^D$ that maps data $x \in \mathcal{X}$ to feature vectors $z = \Phi(x)$. In the supervised setting, representations are learned end-to-end as components of larger systems that solve certain tasks of interest, such as image or video classification, under the assumption that supervision is available to drive the learning process. When supervision is not available, representations can still be learned via self-supervision by means of suitable pretext tasks. Among the latter, *noise contrastive learning* is one of the most popular and successful ones [24, 101]. We summarize this background next and discuss our extensions in the following sections.

3.1. Background: Multi-modal contrastive learning

The idea is to train the representation Φ to identify data points up to the addition of noise or, more generally, the application of certain nuisance transformations. To this end, let $g : \mathcal{X} \rightarrow \mathcal{X}$ be transformations sampled in a set $\mathcal{G}$ of possible nuisances (for example random image crops). Let $\text{sim}(z', z'')$ be a similarity function comparing representations z' and z'' , such as the cosine similarity:

$$\text{sim}(z', z'') = \frac{\langle z', z'' \rangle}{\|z'\| \|z''\|}.$$

Consider a dataset or batch $\mathcal{B} = \{x_1, \dots, x_N\}$ of data samples. Slightly modifying [24], for each sample x_i , draw a set of random nuisance transformations $\{g_{\alpha i}\}_{1 \leq i \leq N}$ and let $z_{\alpha i} = \Phi(g_{\alpha i}(x_i))$ be the representations of the transformed samples. Likewise, consider a second set β of transformations $\{g_{\beta i}\}_{1 \leq i \leq N}$. The noise contrastive loss (NCE) is given by:

$$\mathcal{L}(\alpha, \beta) = -\frac{1}{N} \sum_{i=1}^N \log \frac{e^{\frac{1}{\tau} \text{sim}(z_{\alpha i}, z_{\beta i})}}{\sum_{j=1}^N e^{\frac{1}{\tau} \text{sim}(z_{\alpha i}, z_{\beta j})}} \quad (1)$$

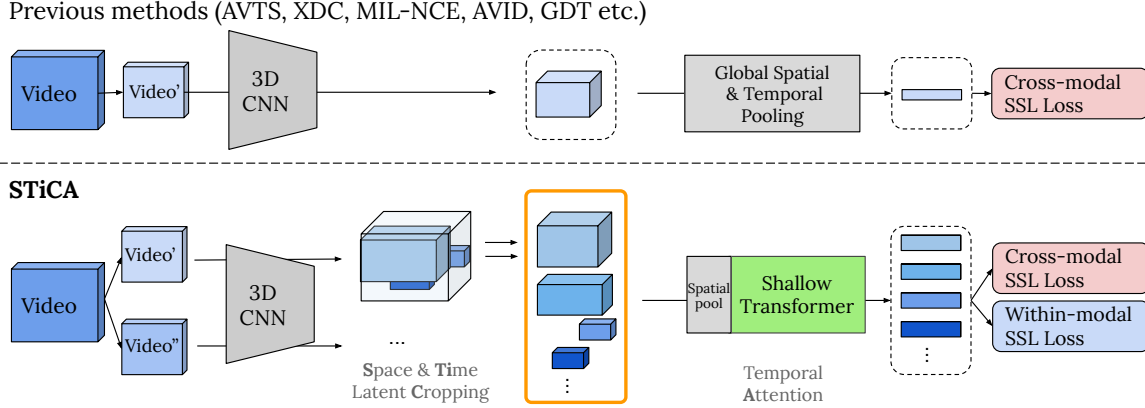


Figure 2: Approach Overview. We present a self-supervised approach that learns video representations without labels. **(Top)** Prior work in video representation learning did not capture spatial invariances, as taking many crops of the input (at varying locations and scales), quickly gets expensive in both compute and memory. **(Bottom)** The proposed method generates a large variety of views from only two RGB-crops by cropping in latent space and is particularly tailored to self-supervised contrastive learning. The latent crops are essentially masked features, which are then further processed by a light-weight temporal transformer. Compared to global pooling, this allows our method to further capture the rich temporal signal.

where $\tau > 0$ is a temperature parameter. This loss pulls together the representations of samples that only differ by the transformation while pushing apart the others. Note that this definition is not symmetric in the two arguments α and β (i.e., $\mathcal{L}(\alpha, \beta) \neq \mathcal{L}(\beta, \alpha)$). Note also that we can introduce any number of transformation sets $\alpha, \beta, \gamma, \dots$ and, for each pair, we can obtain a different variant of eq. (1).

Recently, works such as [106] have ported this technique to the video domain by contrasting modalities. Each video $\mathbf{x} = (\mathbf{v}, \mathbf{a})$ consists of a visual component $\mathbf{v}$ and an audio component $\mathbf{a}$. One consider two sets of transformations g_v , extracting and augmenting the visual component, and g_a , extracting and augmenting the audio component. We still write $\Phi(g_\alpha(\mathbf{x}))$ for the feature computed for either visual and audio components, but the symbol means that a modality-specific neural network is applied as needed.²

With this, we can derive three variants of eq. (1), involving mixed visual-audio and homogeneous visual-visual and audio-audio comparisons. Their combinations are:

$$\lambda_{va}\mathcal{L}(\mathbf{v}, \mathbf{a}) + \lambda_{av}\mathcal{L}(\mathbf{a}, \mathbf{v}) + \lambda_{vv}\mathcal{L}(\mathbf{v}_1, \mathbf{v}_2) + \lambda_{aa}\mathcal{L}(\mathbf{a}_1, \mathbf{a}_2). \quad (2)$$

where λ_{va} , λ_{av} , λ_{vv} and λ_{aa} are non-negative mixing weights.

Challenge 1: Encoding within-modality invariance. While all terms in 2 code for desirable invariances of the representation, several recent papers [90, 97, 106] have found that the mixed term λ_{va} is far more important than the other two; in fact, performance *degrades* if one sets $\lambda_{aa}, \lambda_{vv} \neq 0$, meaning that within-modal invariances are

²In other words, $\Phi = (\Phi_v, \Phi_a)$ is really a pair of networks, producing embedding vectors $\mathbf{z}_\alpha$ that are compatible regardless of the modality $\alpha \in \{\mathbf{v}, \mathbf{a}\}$.

not successfully leveraged. Our hypothesis is that within-modality invariance can be beneficial, and that these early negative results are due to the fact that current learning formulations are ineffective at capitalizing on this signal.

As suggested in Sec. 1, the fact that video data is large means that the batch size used in learning must be small. As a consequence, a batch can contain only a very small number of different augmentations of the same video sample. In current multi-modal learning formulations, each video is already transformed twice in order to extract video and audio components, so cross-modal invariance is learned well. However, the downside is that there is no space left in the batch for multiple visual or audio augmentations. Thus, within-modality invariance is learned only indirectly — in particular, as noted in Sec. 1, two different visual or audio augmentations of the same video are visited by the model only after an entire training epoch. Next, we address this issue by making it feasible to extract several within-modality transformations in the same batch even for video data.

3.2. Efficient spatial cropping for augmentation

It has been found that self-supervised learning benefits from, and requires more and stronger augmentations compared to the supervised counterpart for optimal performance [24]. In particular, several papers [10, 20, 24] have suggested that, in the case of still images, the most important type of augmentation is *cropping*. Namely, given an RGB image $\mathbf{x} \in \mathbb{R}^{3 \times H \times W}$ with three channels and height and width H and W respectively, a crop is given by a box $B = (x_{\min}, x_{\max}, y_{\min}, y_{\max})$. The image tensor is first cropped as:

$$C_B(\mathbf{x}) = \mathbf{x}[:, y_{\min}:y_{\max}, x_{\min}:x_{\max}] \quad (3)$$

where the $:$ symbol is used to denote an index range. The cropped tensor is then resized to a tensor $\tilde{x} = g(x) = R_{H_0 W_0}(C_B(x)) \in \mathbb{R}^{3 \times H_0 \times W_0}$ with a given height and width $H_0 \times W_0$. In practice, $R_{H_0 W_0}$ may also apply additional augmentations such as color jittering, as detailed in the experiments.

As for the visual part $v \in \mathbb{R}^{3 \times T \times H \times W}$ of a video, the situation is similar, except that the video also contains an additional temporal dimension T . To avoid extreme spatial jittering and keep objects aligned, a spatial crop is usually taken at the same location in the input space throughout the whole temporal dimension, so we consider the tube $B = (x_{\min}, x_{\max}, y_{\min}, y_{\max}, t_{\min}, t_{\max})$ and define $\tilde{v} = g_v(v) = R_{H_0 W_0}(C_B(v)) \in \mathbb{R}^{3 \times T_0 \times H_0 \times W_0}$ by extending (3) in the obvious way.

The deep neural network $z = \Phi(\tilde{v})$ mapping v to its corresponding code z is fed with tensors with two spatial dimensions and a temporal one. Such networks, often called 3D for this reason, include R3D [55], S3D [144] and R(2+1)D [133]. As customary in deep convolutional neural networks, they first produce an intermediate tensor with lower space-time resolution and then pool the latter to obtain a single code vector for the entire video. We explicitly break this down into three functions

$$\Phi(\tilde{v}) = (\mathcal{P}_t \circ \mathcal{P}_s \circ \Psi)(\tilde{v}) \quad (4)$$

Here, the first function is a 3D convolutional neural network $\Psi(\tilde{v}) \in \mathbb{R}^{D \times T_1 \times H_1 \times W_1}$ producing a tensor with reduced resolution $T_1 < T_0$, $H_1 < H_0$, $W_1 < W_0$. The operators $\mathcal{P}_s$ and $\mathcal{P}_t$ collapse, respectively, spatial and time dimensions via average pooling.

Now consider implementing term $\mathcal{L}(v_1, v_2)$ in 2. In this case, one samples from each video x_i two different space-time crops $g_{v_1 i}(x_i)$ and $g_{v_2 i}(x_i)$, each corresponding to random tubes B_1 and B_2 respectively. The tubes are not sampled entirely independently, however, as they have the same temporal extent $(t_{\min}, t_{\max})$.

Naïve multiple spatial cropping In practice, [20, 24, 82, 95] show that taking multiple image crops improves self-supervised image representations. We can achieve a similar effect for videos by summing losses $\mathcal{L}(v_\alpha, v_\beta)$ for sets of visual transformations $v_\alpha \neq v_\beta$, obtained by sampling multiple space-time tubes for each video, but this is practically difficult, both due to the large memory footprint and the compute overhead of the slow 3D CNN for each crop.

The Multi-Crop approach introduced by SwAV [20] in the image domain combined with our asymmetric contrastive formulation (1) can partially reduce the complexity. For Multi-Crop, we consider three crop sizes $\alpha \in \{L_1, L_2, S\}$ where L_1 and L_2 stands for large and S for small. The use of a small crop allows to reduce the memory consumption when the representation Φ is computed. We

then have losses:

$$\mathcal{L}(v_{L_1}, v_{L_2}) + \mathcal{L}(v_{L_2}, v_{L_1}) + \mathcal{L}(v_{L_1}, v_S) + \mathcal{L}(v_{L_2}, v_S).$$

While operating on small videos saves some computation, in practice this approach is insufficient to allow using more than a handful of crops in total.

Efficient cropping in feature space. As illustrated in Fig. 2, a much more efficient alternative to cropping the input video is to crop intermediate features.

To do so, we first apply the trunk Ψ of the representation to an input-space crop of the visual component of the video $\tilde{v} = R_{H_0 W_0}(C_B(v)) \in \mathbb{R}^{D \times T_1 \times H_1 \times W_1}$. Then we can efficiently construct a new view of this data by applying the *Feature Crop* $C_{\tilde{B}}$ directly on each intermediate representation, yielding

$$\tilde{v} = C_{\tilde{B}}(\Psi(\tilde{v})) = \Psi(\tilde{v})_{[t_{\min}:t_{\max}, y_{\min}:y_{\max}, x_{\min}:x_{\max}]} \quad (5)$$

Since the operator $C_{\tilde{B}}$ is lightweight, it can be used to compute several such random views efficiently; by comparison, cropping the input RGB images requires recomputing the trunk Ψ multiple times.

In practice, given an input video v , we generate the following views. First, we apply two crops in RGB space, producing two large crops L_1 and L_2 . Then, for each of those, we use the operator (5) to generate m medium-sized and n small-sized crops $\mathcal{T}_i = \{M_1 L_i, \dots, M_m L_i, S_1 L_i, \dots, S_n L_i\}$. We define an overall within-modality loss by summing losses for each pairs of views in $\mathcal{T}$ with exception of pairs where both crops are small:

$$L_{vv} = \sum_{\alpha, \beta} \mathcal{L}(v_\alpha, v_\beta) + \mathcal{L}(v_\beta, v_\alpha), \quad \text{where} \\ (\alpha, \beta) \in (\mathcal{T}_1 \times \mathcal{T}_2) - (\mathcal{S}_1 \times \mathcal{S}_2) \quad (6)$$

Note that there are $2((m+n)^2 - n^2)$ terms in this loss. This is a far greater number of comparison than afforded by the two initial input-space RGB crops.

3.3. Temporal modelling with transformers

We now discuss our second improvement: better modelling of time.

Challenge 2: Modelling time better. Contrary to spatial invariance, models should not be fully invariant to time as the latter can encode causality and with it semantics: a video of someone starting a fire is very different from its reversed version, in which someone extinguishes it. In standard 3D networks, features in the trunk are sensitive to the temporal order, but this information is lost in the final stage, where temporal averaging is applied. We argue that the value of the lost information increases with the length of the video, and that this information can be leveraged by switching to a different pooling function.

Temporal transformer. We propose to tackle this issue by replacing average pooling in time $\mathcal{P}_t$ in eq. (4) with a transformer $\mathcal{P}_{\text{transf}}$. Transformers [135] have been shown effective for representing sequential inputs in the NLP domain [62, 81, 113, 114]. After spatial averaging, the output $\mathbf{h} = \mathcal{P}_s(\Psi(\tilde{\mathbf{v}})) \in \mathbb{R}^{D \times T_1}$ of the network has one feature vector per time step, and is thus amenable to processing by a transformer. The feature $\mathbf{h}$, which differs in latent time-dimension size from its uncropped variant can be seen as masking the transformer’s attention. Masking attention has been used in transformer encoder-decoder training to prevent the model from cheating [33] and encourage it to leverage information from the context. We use a shallow and light-weight transformer on top of our feature cropping procedure, which we show to be sufficient to reap the benefit of better temporal modelling incurring only a very small computational cost. We use 2-layers and 4 self-attention heads and provide further details on the transformer architecture in the Appendix.

3.4. Overall loss

Our combined model, **STiCA**, better learns space-time invariances and relationships by cropping in space-time and leveraging temporal attention with a transformer. For training, we sample N videos in a batch and, for each of them, compute two ‘large’ visual crops in RGB space, $2(n + m)$ small and medium feature crops (Sec. 3.2), and an audio augmentation a . With those, the overall objective is obtained by summing the within-modality loss L_{vv} from eq. (6) to the cross modality losses:

$$L = \lambda_{vv}L_{vv} + \lambda_{va}L_{va}, \quad (7)$$

where $L_{va} = \mathcal{L}(v_{L_1}, a) + \mathcal{L}(v_{L_2}, a) + \mathcal{L}(a, v_{L_1}) + \mathcal{L}(a, v_{L_2})$.

4. Experiments

We first describe the datasets (Sec. 4) and implementation details (Sec. 4) for pretraining. In Sec. 4.1, we describe the downstream tasks for evaluating the representation obtained from self-supervised learning. In Sec. 4.2, we ablate the various components of our method, and the importance of temporal context and multi-modality in Sec. 4.3. Lastly, in Sec. 4.4, we compare with prior work in video and multi-modal representation learning.

Data. We pretrain on the Kinetics-400 dataset [69], which contains about 230K training videos and 13K validation videos belonging to 400 action classes. This dataset is the ‘ImageNet’ for video representation learning due to its moderate size and being public, allowing for broad access and comparability. After pretraining, we evaluate using video action retrieval and action recognition on HMDB-51 [77] and UCF-101 [120]. HMDB-51 [77] consists of 7K

video clips spanning 51 different human activities. HMDB-51 has three train/test splits of size 5K/2K respectively. UCF-101 [120] contains 13K videos from 101 human action classes, and has three train/test splits of size 11K/2K respectively.

Implementation details. Following [106], we use the R(2+1)-18 [133] network as visual encoder and ResNet [59] with 9 layers as audio encoder. We train for 100 epochs and use 30 frames with temporal stride of 1 at sampling rate of 30fps at spatial resolution of 112×112 as input. In our ablations, we evaluate the learned representation by fine-tuning the visual encoder on fold 1 of the HMDB-51 [77] action recognition dataset. Further implementation details are given in the Appendix.

4.1. Downstream tasks

Video action retrieval. For video retrieval, we follow the standard protocol described in [145]. We use the split 1 of UCF-101, and additionally HMDB-51. We uniformly sample 10 clips per video, max pool and then average the features after the last residual block for each clip per video. We use these averaged features from the validation set to query the videos in the training set. If the class of a retrieved video matches the class of query video, we count it as a match. We measure recall at $k=1, 5, 20$.

Video action recognition. As is standard in the literature, we evaluate our pretrained representations by finetuning our visual backbone on the video action recognition task on HMDB-51 and UCF-101 datasets. We closely follow the finetuning schedule of GDT [106]. During finetuning, we use SGD with initial learning rate 0.0025, which we gradually warm up to 0.02 in the first 2 epochs. The weight decay is set to 0.005 and momentum to 0.9. We use a mini-batch size of 32 and train for 12 epochs with the learning rate multiplied by 0.05 at 6 and 10 epochs. For training, we randomly sample 1s clips per video, and during evaluation, we uniformly sample 10 clips from each video and apply 3-crop evaluation as in [41].

4.2. Comparison experiments and ablations

Cropping augmentation. In Tab. 1a, we ablate the importance of spatial augmentation in learning video representations. We compare our proposed Feature Crop augmentation, $C_{\bar{B}}$, to the recently proposed Multi-Crop augmentation strategy [20] and other baseline approaches. Multi-Crop has proven to be effective in image self-supervised learning because it forces the model to learn local-to-global associations, by explicitly enforcing invariance between features of large-crops and those of multiple small crops. While effective, it can be particularly computationally intensive, which, with our hardware, limits its use to only two large crops and one small crop when applied to video representation learn-

Table 1: Comparison experiments and ablations. We compare key parameters and settings of our proposed method. We report results model performance at epoch 100 and with 30 frames and without transformer unless noted otherwise.

Cropping-strategy	Resolution	GPU-h/epoch	Acc.	l -Spatial size M S	l -Temporal size M S	Acc.
Default	1×112^2	17.3	54.0	1×7^2	1×4	54.0
Two RGB Crops	2×112^2	29.3	58.6	1×6^2	2×4	59.9
Multi RGB Crops [20]	$2 \times 112^2 + 1 \times 96^2$	46.7	59.3	1×6^2	2×3	58.4
Ours (Feature Crop)	$2 \times 112^2 + \text{latent}$	29.3	60.4	2×6^2	4×4	60.4

(a) **Cropping** yields benefits but requires more compute. Our feature crops are efficient and outperform [20]. Note that all models are trained for 100 epochs.

(b) **Feature crops.** Heavier augmentations in latent (l) space and time lead to better representations.

Pretraining	Finetuning	Acc	Transf.?	Layers	Params	GFLOPS	Acc.	C_{space}	C_{time}	T?	Acc.
$\mathcal{P}_t$	$\mathcal{P}_t$	54.0	✗	0	37.2M	77.7	54.0	✗	✗	✗	54.0
$\mathcal{P}_t$	$\mathcal{P}_{\text{transf}}$	54.6	✗	0	42.8M	80.0	57.3	✓	✗	✗	59.9
$\mathcal{P}_{\text{transf}}$	$\mathcal{P}_t$	52.1	✓	2	42.4M	77.8	60.3	✓	✓	✗	60.4
$\mathcal{P}_{\text{transf}}$	$\mathcal{P}_{\text{transf}}$	60.3	✓	4	47.7M	77.8	58.3	✓	✓	✓	62.0

(c) **Pooling.** Compared to Average-Pooling ($\mathcal{P}_t$), Transformer-based pooling ($\mathcal{P}_{\text{transf}}$) gives stronger performance.

(d) **Architecture.** Using up to two transformer layers gives gains, not due to more trainable parameters.

(e) **Combined gains.** Feature crop in space C_{space} and time C_{time} and transformer pooling (T) add cumulative benefits.

Method	RGB-Crops			Multi-scale RGB-Crops			Feature Crops	
	1x	2x	4x*	$2 \times 112 + 1 \times 96$	$2 \times 112 + 2 \times 96$	$2 \times 112 + 6 \times 96^*$	(1x7, 1x4)	(2x6 + 4x4, 2x3 + 1x2)
GPU-h/epoch	17.3	29.3	60.0	46.7	53.3	100.7	29.3	30.0

(f) **Speed.** Input-crops are slow: * methods require reducing batch sizes (see Appendix) as activations do not fit on GPU.

ing. Our proposed Feature Crop is not only more efficient, but outperforms Multi-Crop by 1.1% when the learned representations is applied to action classification in HMDB-51. By cropping in feature space, we achieve a similar effect but can increase the number of small crops from 1 to 6 without increasing compute time.

Feature crop parameters. In Tab. 1b, we study the parameters of our Feature Cropping approach. We find that even our basic variant, which does one medium 6×6 crop and two 4×4 small crops (by cropping a 7×7 tensor) increases performance by nearly 6%, which is a relative improvement of more than 10%. If we further increase the number of crops in time and space, the performance increases from 59.9% to 60.4%.

Pooling Function. In Tab. 1c, we test temporal aggregation. We find that using a shallow transformer significantly outperforms simple average pooling by more than 5%; however, transformer pooling must be used both for pre-training the representation and for finetuning it on the target dataset.

Transformer architecture. In Tab. 1d, we test variants of the transformer architecture, including ablating iit altogether. We find that temporal modelling as measured by downstream performance peaks at two layers, likely due to optimization difficulties of deeper transformers with SGD. We also compare to a model with approximately the same number of parameters as our 2-layer transformer (achieved by increasing the networks' last block's hidden dimension

to 640). We find that the transformer still yields gains of 3%, indicating that it not the number of parameters but the modelling of time that is crucial for strong performance.

Combining Feature Crops and Transformer Pooling. In Tab. 1e, we show that combining Feature Crops in space and time, and then adding transformer pooling yield additive gains, with the best result obtained by combining all effects (which corresponds to STICA). This shows that space-time augmentations and transformer pooling are complementary.

Cropping efficiency. In Tab. 1f, we compare training times (normalized to GPUs×hours) for Kinetics-400 epochs for the various spatial crops considered. We make two observations: First, the compute cost of RGB crops scales proportionally to their number because a full forward pass is required for each crop. Second, using a larger number of RGB crops eventually requires to decrease the batch size, which increases significantly the training time. In contrast, the cost of Feature Crop remains roughly constant no matter the number of crops.

4.3. Temporal Context and Multi-modality

Length of temporal context. In Tab. 2, we show the importance of leveraging longer context to improve video self-supervised representation learning. Similar to the supervised regime [134, 139], we observe improved accuracy as we increase the number of frames used during pretraining and fine-tuning. More importantly, the transformer pooling

Frames		Accuracy	
Pretrain	Finetune	GAP	Transf.
30	30	54.0	60.3
60	60	62.4	66.1
90	90	58.0	66.9

Table 2: Temporal context. We report results with different number of frames on finetuning accuracy.

λ_{va}	λ_{vv}	F. Crop?	Acc.
0	1	No	43.3
1	0	No	54.0
0.5	0.5	No	58.6
0.5	0.5	Yes	60.3

Table 3: Loss. Combining within-modal and cross-modal loss with Feature-crops is key.

Method	Architecture	Dataset	Top-1 Acc% HMDB UCF
Supervised	R(2+1)D-18	K-400	70.4 95.0
SeLaVi [9]	R(2+1)D-18	K-400	47.1 83.1
TempTrans [63]	R3D-18	K-400	49.8 79.3
PEMT [80]	SlowFast	K-400	- 85.2
XDC [6]	R(2+1)D-18	K-400	52.6 86.2
MemDPC [53]	R-2D3D	K-400	54.5 86.1
AVSF [143]	AVSF	K-400	54.6 87.0
AVTS [74]	MC3-18	K-400	56.9 85.8
CPD [85]	R3D-50	K-400	57.7 88.7
AVID [97]	R(2+1)D-18	K-400	60.8 87.5
GDT [106]	R(2+1)D-18	K-400	60.0 89.3
ACC [90]	R3D-18	K-400	61.8 90.2
GLCM [91]	R3D-18	K-400	61.9 91.2
CoCLR [54]	S3D	K-400	62.9 90.6
CVLR [112] ³	R3D-50	K-400	66.7 92.2
Ours: STiCA	R(2+1)D-18	K-400	67.0 93.1
SeLaVi [9]	R(2+1)D-18	VGGS	53.1 87.7
Speech2Act [98]	S3D-G	Movie	58.1 -
DynamoNet [34]	ResNext101	Y8M	58.6 87.3
MIL-NCE [93]	S3D	HT	61.0 91.3
AVTS [74]	MC3-18	AS	61.6 89.0
AVID [97]	R(2+1)D-18	AS	64.7 91.5
Textual [123]	S3D-G	WVT-70M	65.3 90.3
GDT [106]	R(2+1)D-18	AS	66.1 92.5
ACC [90]	R(2+1)D-18	AS	67.2 93.5
ELo [110]	R(2+1)D-50	Y2M	67.4 93.8
XDC [6]	R(2+1)D-18	IG65M	68.9 95.5
GDT [106]	R(2+1)D-18	IG65M	72.8 95.2
MMV [5]	TSM-50x2	AS+HT	75.0 95.2

Table 4: Comparison to SoTA for action recognition. Dashed line indicates position of our Kinetics-400 model in comparison to models trained with many more videos. We follow standard evaluation protocol across 3-folds. For linear evaluation results see Tab. 7.

layer is better able to exploit this additional context, outperforming average pooling by over 4% for all frame lengths. Notably, there is a drop in performance when using GAP for extremely long contexts (90 frames).

Loss. Lastly, in Tab. 3, we study the effect of combining multi-modal learning signals with our contributions. In the first row, we have the baseline of naively extending Sim-

³Concurrent work.

Recall @	UCF			HMDB		
	1	5	20	1	5	20
MemDPC [53]	20.2	40.4	64.7	7.7	25.7	57.7
VSP [25]	24.6	41.9	62.7	10.3	26.6	76.8
SeLaVi [9]	52.0	68.6	84.5	24.8	47.6	75.5
CoCLR [54]	55.9	70.8	82.5	26.1	45.8	69.7
GDT [106]	57.4	73.4	88.1	25.4	51.4	75.0
Ours: STiCA	59.1	76.2	88.1	26.3	49.2	76.4

Table 5: Comparison to SoTA for retrieval. Nearest neighbor action retrieval performance @ $k = \{1, 5, 20\}$.

CLR [24] to the video domain, by learning invariances to spatial augmentations of two large-crops. Compared to this, the cross-modal baseline (row 2) already achieves gains of more than 10%. While adding a within-modal invariance adds another 4.6%, we find that the best performance is obtained with our feature crops, adding another 1.7% in performance and showing its unique potential to supplement cross-modal signals.

4.4. Comparison with the state of the art

Video Action Recognition. In Tab. 4, we evaluate our pretraining approach on the standard HMDB-51 and UCF-101 action recognition benchmarks after pretraining on the Kinetics-400 dataset. Firstly, we find our model outperforming the similar NCE-based GDT [106] model by 7.0% and 3.8% on HMDB-51 and UCF-101. We further significantly outperform the current state-of-the-art methods CoCLR [54] by 4.1% and 2.5% and CVLR [112] by 2.6% and 1.0% on HMDB-51 and UCF-101, respectively. Even more impressively, our approach is able to outperform most prior works that use AudioSet [44] pre-training, which is around 10× larger than Kinetics-400. This shows how effective and data-efficient our approach is, significantly closing the gap to supervised learning.

Video Action Retrieval. Lastly, we directly evaluate the transfer-ability of our pretrained representations on action retrieval on UCF-101 and HMDB-51. Similarly to full finetuning setting, we outperform all prior works.

5. Conclusion

We have address two shortcomings of current self-supervised video representation learning: insufficient spatial invariance, especially compared to the image domain, and inadequate modelling of time. We have introduced STiCA, improving spatial invariance at very little cost by implementing cropping in feature space, and improving modelling of time via a shallow transformer. Our method brings self-supervised video representation learning one step closer to the supervised case, providing significant gains w.r.t. the state-of-the-art.

Acknowledgements. We are grateful for support from the Rhodes Trust (M.P.), Facebook (M.P.), EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems [EP/L015897/1] (M.P. and Y.A.), the Qualcomm Fellowship (Y.A.), and the Royal Academy of Engineering under the Research Fellowship scheme (J.F.H.). This work is also supported by the DARPA grants funded under the AIDA program (FA8750-18-2-0018) and the GAILA program (award HR00111990063) (P.H.). We also thank Tengda Han for helpful discussions and feedback.

References

- [1] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [2] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. *Interspeech*, 2018.
- [3] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Deep lip reading: A comparison of models and an online application. 2018.
- [4] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *ECCV*, 2020.
- [5] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020.
- [6] Humam Alwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS*, 2020.
- [7] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.
- [8] Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.
- [9] Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *NeurIPS*, 2020.
- [10] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *ICLR*, 2020.
- [11] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.
- [12] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.
- [13] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019.
- [14] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *CVPR*, 2020.
- [15] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021.
- [16] Uta Buchler, Biagio Brattoli, and Bjorn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *ECCV*, 2018.
- [17] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.
- [18] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- [19] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, 2019.
- [20] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.
- [21] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.
- [22] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *ECCV*, 2020.
- [23] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020.
- [24] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.
- [25] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatio-temporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.
- [26] Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. *CVPR*, 2017.
- [27] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.
- [28] E. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical data augmentation with no separate search. In *CVPRW*, 2020.
- [29] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. In *CVPR*, 2020.
- [30] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005.
- [31] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.
- [32] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations, 2020.
- [33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

- [34] Ali Diba, Vivek Sharma, Luc Van Gool, and Rainer Stiefel-hagen. Dynamonet: Dynamic action and motion network. In *ICCV*, 2019.
- [35] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.
- [36] Carl Doersch, Ankush Gupta, and Andrew Zisserman. Crosstransformers: spatially-aware few-shot transfer. In *NeurIPS*, 2020.
- [37] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.
- [38] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020.
- [39] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *TPAMI*, 38(9), 2015.
- [40] D. Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *CVPR*, 2020.
- [41] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.
- [42] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.
- [43] Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. Listen to look: Action recognition by previewing audio. In *CVPR*, 2020.
- [44] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, 2017.
- [45] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.
- [46] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.
- [47] Rohit Girdhar, João Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019.
- [48] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan C. Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *CVPR*, 2017.
- [49] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.
- [50] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, 2010.
- [51] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
- [52] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCVW*, 2019.
- [53] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020.
- [54] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *NeurIPS*, 2020.
- [55] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. In *ICCVW*, 2017.
- [56] David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *ECCV*, 2018.
- [57] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.
- [58] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv.cs, abs/1911.05722*, 2019.
- [59] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.
- [60] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *CVPR*, 2019.
- [61] Noureldien Hussein, Efstratios Gavves, and Arnold W. M. Smeulders. Timeception for complex action recognition. In *CVPR*, 2019.
- [62] Kenton Lee Jacob Devlin, Ming-Wei Chang and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2018.
- [63] S. Jenni, Givi Meishvili, and P. Favaro. Learning video representations by transforming time. In *ECCV*, 2020.
- [64] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *ICCV*, 2019.
- [65] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.
- [66] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *NeurIPS*, 2020.
- [67] Yash Kant, Dhruv Batra, Peter Anderson, Alex Schwing, Devi Parikh, Jiasen Lu, and Harsh Agrawal. Spatially aware multimodal transformers for textvqa. In *ECCV*, 2020.
- [68] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.
- [69] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang,

- Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.
- [70] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *ICCV*, 2019.
- [71] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.
- [72] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- [73] Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation, 2020.
- [74] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-operative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.
- [75] Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *ICCV*, 2019.
- [76] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.
- [77] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.
- [78] Chia-Wen Kuo, Chih-Yao Ma, Jia-Bin Huang, and Zsolt Kira. Featmatch: Feature-based augmentation for semi-supervised learning. In *ECCV*, 2020.
- [79] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.
- [80] Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *International Conference on Learning Representations*, 2021.
- [81] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, 2020.
- [82] Hao Li, Xiaopeng Zhang, Ruoyu Sun, Hongkai Xiong, and Qi Tian. Center-wise local image mixture for contrastive representation learning, 2020.
- [83] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.
- [84] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language, 2019.
- [85] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv preprint arXiv:2001.05691*, 2020.
- [86] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020.
- [87] David G. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, 1999.
- [88] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. VILBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019.
- [89] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks, 2017.
- [90] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Learning audio-visual representations with active contrastive coding, 2020.
- [91] Shuang Ma, Zhaoyang Zeng, Daniel McDuff, and Yale Song. Contrastive self-supervised learning of global-local audio-visual representations, 2021.
- [92] Antoine Miech, Jean-Baptiste Alayrac, Piotr Bojanowski, Ivan Laptev, and Josef Sivic. Learning from video and text via large-scale discriminative clustering. In *Proc. ICCV*, 2017.
- [93] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020.
- [94] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100M: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019.
- [95] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
- [96] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.
- [97] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. *arXiv preprint arXiv:2004.12943*, 2020.
- [98] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.
- [99] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network, 2021.
- [100] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.
- [101] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [102] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
- [103] D. Park, William Chan, Y. Zhang, Chung-Cheng Chiu, Barret Zoph, E. D. Cubuk, and Quoc V. Le. SpecAugment: A simple data augmentation method for automatic speech recognition. In *INTERSPEECH*, 2019.
- [104] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *ICML*, 2018.
- [105] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.
- [106] Mandela Patrick, Yuki Markus Asano, Ruth Fong, João F.

- Henriques, G. Zweig, and A. Vedaldi. Multi-modal self-supervision from generalized data transformations. *ArXiv*, abs/2003.04298, 2020.
- [107] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander Hauptmann, João Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning, 2020.
- [108] Karol J. Piczak. Environmental sound classification with convolutional neural networks. *MLSP*, 2015.
- [109] Karol J. Piczak. Esc: Dataset for environmental sound classification. In *ACM Multimedia*, 2015.
- [110] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, 2020.
- [111] Gerasimos Potamianos, Chalapathy Neti, Guillaume Gravier, Ashutosh Garg, and Andrew W Senior. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.
- [112] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, H. Wang, Serge J. Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. 2020.
- [113] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.
- [114] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.
- [115] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.
- [116] Guido Roma, Waldo Nogueira, and Perfecto Herrera. Recurrence quantification analysis features for environmental sound recognition. *WASPAA*, 2013.
- [117] Hardik B. Sailor, Dharmesh M Agrawal, and Hemant A Patil. Unsupervised filterbank learning using convolutional restricted boltzmann machine for environmental sound classification. In *INTERSPEECH*, 2017.
- [118] Mert Bulent Sariyildiz, Julien Perez, and Diane Larlus. Learning visual representations with caption annotations. In *ECCV*, 2020.
- [119] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014.
- [120] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.
- [121] Dan Stowell, Dimitrios Giannoulis, Emmanouil Benetos, Mathieu Lagrange, and Mark D. Plumbley. Detection and classification of acoustic scenes and events. *TM*, 2015.
- [122] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley. Detection and classification of acoustic scenes and events. *IEEE Transactions on Multimedia*, 2015.
- [123] Jonathan C. Stroud, D. Ross, Chen Sun, Jun Deng, R. Sukthankar, and C. Schmid. Learning video representations from textual web supervision. *ArXiv*, abs/2007.14937, 2020.
- [124] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vi-bert: Pre-training of generic visual-linguistic representations. In *ICLR*, 2020.
- [125] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019.
- [126] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, 2019.
- [127] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019.
- [128] Hao Tan and Mohit Bansal. Vokenization: Improving language understanding with contextualized, visual-grounded supervision. In *EMNLP*, 2020.
- [129] V Terrance and W Taylor Graham. Dataset augmentation in feature space. In *Proceedings of the international conference on machine learning (ICML), workshop track*, 2017.
- [130] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020.
- [131] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. In *NeurIPS*, 2020.
- [132] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.
- [133] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.
- [134] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term Temporal Convolutions for Action Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [135] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.
- [136] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, 2019.
- [137] Jiangliu Wang, Jianbo Jiao, and Y. Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020.
- [138] Limin Wang, Yuanjun Xiong, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.
- [139] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018.
- [140] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.
- [141] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. 2020.
- [142] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin.

- Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.
- [143] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slow-fast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.
 - [144] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. In *ECCV*, 2018.
 - [145] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.
 - [146] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.
 - [147] Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank J Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models?, 2020.
 - [148] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.
 - [149] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, 2017.
 - [150] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.
 - [151] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, 2018.
 - [152] Bolei Zhou, Alex Andonian, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, 2018.
 - [153] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. In *ICLR*, 2015.
 - [154] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020.

6. Appendix

6.1. Implementation Details

While videos in Kinetics are 10 seconds long, we randomly sample either 1-second (30 frames), or 2-second (60 frames) clips from the 30fps videos. For the R(2+1)-D-18 visual encoder, the dimensions of the *res5* feature map before spatial pooling is $512 \times T \times 7 \times 7$ for a 112×112 resolution video, where $T = 4$ for 30-frame (1 second) input, and $T = 8$ for 60-frame (2 second) input. After spatial pooling, we use either average pooling or a transformer as the temporal pooling function for the visual encoder, but always use average pooling for the audio encoder. The transformer’s layers dimensionality are set to 512-D. Both encoders produce a fixed-dimensional representation vectors after temporal aggregation (512-D). Both vectors are then passed through two fully-connected layers with intermediate size of 512 to produce 256-D embedding vectors z as in [106]. We use these embeddings in our loss eq. (7) and train our model for 100 epochs. For the visual component of the video, we use a 30 frame RGB clip as input, at 30 fps covering 1 second. The video clip has a spatial resolution of 112×112 pixels. For input data augmentation, we apply random crops, horizontal flips, Gaussian blur and color jittering, all clip-wise consistent, following the protocol of SimCLR [24], and we ablate multiple settings for spatial and temporal feature cropping sizes. For the audio input, we extract a 1-second log-mel spectrogram of dimension 257×199 starting at the same time as the visual component. We also apply volume jittering to increase the robustness of our audio features. We optimize this model using SGD with momentum 0.9, weight decay 10^{-5} and learning rate 0.64, with a warm-up period of 10 epochs. For NCE contrastive learning, the temperature τ is set as 0.1 for cross-modal loss, and 0.5 for the within-modal loss. We use a mini-batch size of 8 on each of our 64 GPUs giving an effective batch size of 512 for distributed training. In our ablations, we evaluate the learned representation by fine-tuning the visual encoder on fold 1 of the HMDB-51 [77] action recognition dataset.

6.1.1 State-of-the-Art Experiment Details

For our state-of-the-art model, we train for 100 epochs, using R(2+1)-D-18 visual encoder with transformer temporal attention pooling, and Resnet-9 for audio encoder. We use 60 frames as input, and feature-crop augmentation (space: $2 \times 6^2 + 4 \times 4^2$ & time: $2 \times 3 + 1 \times 2$).

6.2. Transformer Architecture Details

We use a 2-layer transformer, with 4 attention heads, and hidden dimension 512. The input to the transformer is the spatially averaged output of the last convolutional layer of

Method	Pretraining	Acc%	
		DCASE	ESC50
Autoencoder [12]	-	-	39.9
Random Forest [109]	-	-	44.3
Piczak ConvNet [108]	-	-	64.5
RNH [116]	-	72	-
Ensemble [121]	-	77	-
ConvRBM [117]	-	-	86.5
AVTS [74]	K400	91	76.7
XDC [6]	K400	-	78.0
AVID [97]	K400	<u>93</u>	79.1
ACC [90]	K400	-	<u>79.2</u>
Ours: STiCA	K400	94	81.1
SoundNet [12]	SNet	88	74.2
L3-Net [7]	SNet	93	79.3
AVTS [74]	SNet	94	82.3
DMC [60]	SNet	-	82.6
AVTS [74]	AS	93	80.6
XDC [6]	AS	-	85.8
MMV [5]	AS	-	86.1
AVID [97]	AS	<u>96</u>	<u>89.2</u>
GDT [106]	AS	98	<u>88.5</u>
ACC [90]	AS	-	90.8
Human [109]	-	-	81.3

Table 6: Audio classification. Downstream task accuracies on standard audio classification benchmarks on DCASE2014 and ESC50. Dataset abbreviations AudioSet, Kinetics400, SoundNet.

R(2+1)D-18 video backbone. The transformer contextualizes features across time to output a fixed feature length representation of dimension 512, which is then passed to MLP head for contrastive learning. While transformers generally benefit from being optimized with Adam [147], we adhere to using SGD for simplicity. We also do not observe any stability issues, likely because the transformer is quite shallow.

7. Additional experiments

7.1. Audio Classification

For completeness, we also present audio classification results on ESC-50 [108] and DCASE-2014 [122]. ESC-50 [109] is an environmental sound classification dataset which has 2K sound clips of 50 different audio classes. ESC-50 has 5 train/test splits of size 1.6K/400 respectively. DCASE2014 [122] is an acoustic scenes and event classification dataset which has 100 training and 100 testing sound clips spanning 10 different audio classes. We demon-

Method	Architecture	Dataset	Top-1 Acc%	
			HMDB	UCF
RotNet3D [65]	S3D	K600	24.8	47.7
CBT [125]	S3D+BERT	K600	29.5	54.0
MemDPC [53]	R-2D3D	K400	30.5	54.1
AVSF [143]	AVSF	K400	44.1	77.4
CoCLR [54]	S3D	K400	46.1	74.5
Ours: STiCA	R(2+1)D-18	K400	48.2	77.0
MIL-NCE [93]	S3D	HT	53.1	82.7
XDC [6]	R(2+1)D-18	IG65M	56.0	85.3
MMV [5]	R(2+1)D-18	AS	60.0	83.9
ELo [110]	R(2+1)D-50	Y8M	64.5	–

Table 7: Comparison to state-of-the-art. Transfer learning results on UCF-101 and HMDB-51 when video backbone is frozen.

strate competitive performance relative to the state-of-the-art, despite training on a much smaller and less audio-rich Kinetics-400 dataset. We extract 10 equally spaced 2-second sub-clips from each full audio sample of ESC-50 [109] and 60 1-second sub-clips from each full sample of DCASE2014 [122]. We save the activations that result from the audio encoder to quickly train the linear classifiers. We use activations after the last convolutional layer of the ResNet-9 and apply a max pooling with kernel size (1,3) and stride of (1,2) without padding to the output. For both datasets, we then optimize a L2 regularized linear layer with batch size 512 using the Adam optimizer [72] with learning rate 1×10^{-4} , weight-decay set to 5×10^{-4} and the default parameters. The classification score for each audio sample is computed by averaging the sub-clip scores in the sample, and then predicting the class with the highest score. The mean top-1 accuracy is then taken across all audio clips and averaged across all official folds.

7.2. Linear probing results

In Tab. 7, we compute the linear classification results of our model compared to other recent methods. We find that our best model has competitive 3-fold linear evaluation results of 48.2% on HMDB-51 and 77.0% on UCF-101.

7.3. Supervised training on K-400

Here we experiment with training supervisedly on Kinetics-400 and observing the effect of using feature cropping (with the configuration 2 medium and 2 small latent space crops). The experimental results are given in Tab. 8. We find that even though our method is designed for contrastive cross-modal pretraining, using feature crops can help in training in a supervised manner too.

Fm-Crop	HMDB-51 Top-1 Acc.
×	67.6
✓	69.0

Table 8: Supervised Training. We train the R(2+1)D+Transformer architecture supervisedly on Kinetics-400 with and without feature crops enabled.

7.4. Audio-Visual Heatmap Visualizations

In Fig. 3, we show examples that our model truly learned some spatial correspondence between a region and audio. We have done this by visualizing the strength of the dot-product of the visual feature map (without pooling) with the audio feature vector.

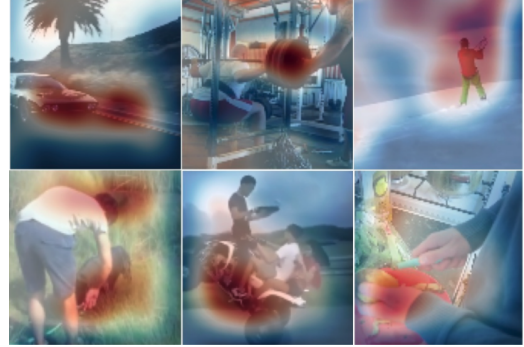


Figure 3: Heatmap visualizations. Heatmaps are obtained by removing the spatial pooling layer and visualizing the strength of the dot-product of the audio feature vector with the video feature-map as in [8]. Here, we show selected samples from Kinetics-400 training set of the resulting heatmaps along with the middle frame of the video.

7.5. Preventing Shortcut Learning with Feature Crops.

Noise contrastive learning works better when you can reduce the mutual information between the input pairs [131] as it's harder for the network to cheat. This can be achieved by taking multiple spatial crops of images in the input space and independently applying different augmentations, such as color jittering and Gaussian blurring, to the cropped inputs. However, as mentioned above, taking more than 2 crops in input space is both memory and computationally infeasible for multi-modal video data. Crops in feature space, on the other hand, allows us to take multiple crops for noise contrastive learning. However, since CNNs have large receptive fields that easily cover the full frame, there may be shortcut learning with feature crops as information may

leak between the crops from same feature map. To alleviate this, we take feature crops from two originally augmented video clips, allowing us to make NCE comparisons *across* modalities and individual augmentations (such as color jitter), leading to a beneficial reduction in mutual information. Furthermore, while the theoretical receptive fields of units in later layers are indeed very large, units tend to be sensitive to an effective area which is significantly smaller than the theoretical receptive field [89, 153], further reducing the mutual information between inputs for noise contrastive learning.

9

Self-supervised object detection from audio-visual correspondence

This work is currently under review and its preprint is public on arXiv.

Self-supervised object detection from audio-visual correspondence

Triantafyllos Afouras^{1*†} Yuki M. Asano^{1*} Francois Fagan² Andrea Vedaldi² Florian Metze²

¹ Visual Geometry Group, University of Oxford

² Facebook AI

afourast@robots.ox.ac.uk

Abstract

We tackle the problem of learning object detectors without supervision. Differently from weakly-supervised object detection, we do not assume image-level class labels. Instead, we extract a supervisory signal from audio-visual data, using the audio component to “teach” the object detector. While this problem is related to sound source localisation, it is considerably harder because the detector must classify the objects by type, enumerate each instance of the object, and do so even when the object is silent. We tackle this problem by first designing a self-supervised framework with a contrastive objective that jointly learns to classify and localise objects. Then, without using any supervision, we simply use these self-supervised labels and boxes to train an image-based object detector. With this, we outperform previous unsupervised and weakly-supervised detectors for the task of object detection and sound source localization. We also show that we can align this detector to ground-truth classes with as little as one label per pseudo-class, and show how our method can learn to detect generic objects that go beyond instruments, such as airplanes and cats.

1. Introduction

While recent progress in learning image and video representations has been substantial [23, 34, 39, 100], this has not yet translated into an ability to learn interpretable and actionable concepts automatically. By that, we mean that some manual labels are still required in order to map unsupervised representations to useful concepts such as image classes or object detections. In this paper, we thus consider the problem of learning interpretable concepts without any manual supervision. In particular, we focus on a problem that has not been explored extensively in the literature: learning to simultaneously detect and classify objects with no manual labels at all.

*Joint first authors.

†Work done during an internship at Facebook.

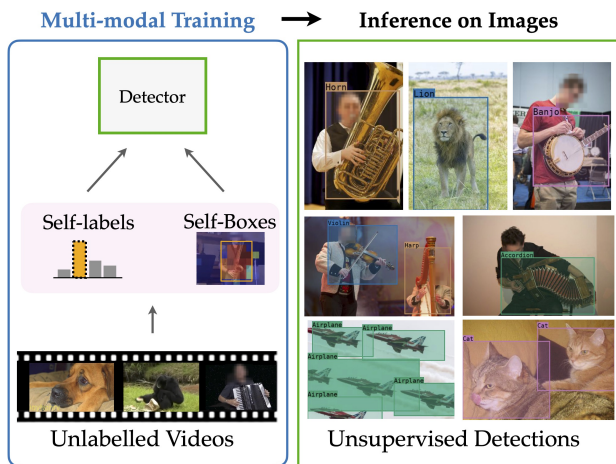


Figure 1: **We train an object detector simply by watching videos.** Without using any manual annotations, we learn to detect different objects in images, by first self-labelling boxes and object categories and then using those as targets to teach a detector. The detection results shown are outputs from our trained model; for visualisation purposes we show Hungarian-matched labels.

This problem is related to weakly supervised object detection (WSOD [16, 65]), with the difference that, in WSOD, the learning algorithm is given image-level labels telling it whether the image contains an occurrence of a given object type or not. Inspired by recent work in self-supervised learning, we seek to replace this source of external supervision with an internal supervisory signal extracted from the observation of video data. Videos are far richer than images, for example because they contain motion. Here, we focus on the multi-modal aspect of videos and use sound as a weak and noisy cue to learn about objects in the visual component of the data.

The power of multi-modal self-supervision has been demonstrated before in self-supervised representation learning, and, closely related, in *video clustering* [9]. However, while video clustering can provide an interpretation

of the data in terms of discrete classes, it does not provide any information about the location of the relevant objects in images. On the other hand, *sound source localisation* [7, 11, 52, 69] has considered precisely the problem of localizing the source of sounds in images. It is therefore tempting to trivially combine image classification and sound source localisation in the hope of learning the type and location of objects automatically.

Unfortunately, such an approach does *not* lead to a satisfactory object detector. To understand why, it is important to note that the goal of sound source localisation is to *localize the sound while it is being heard*. This is insufficient for a detector because many objects emit sounds only occasionally and they become invisible to source localisation when they are silent. Instead, a detector that works in the visual domain should be responsive even when the object cannot be heard. Furthermore, source localisation methods generally only extract a heatmap giving the distribution of possible object locations; in contrast, a detector solves the much harder problem of enumerating all individual object instance that occur in an image by outputting instance-specific bounding boxes.

In order to solve these issues, we should treat the sound component as a useful cue to *learn* an object detector, but not as a cue which is *necessary for detection*. Instead, we consider the problem of taking as input a collection of raw videos and producing a list of object classes and locations, in order to train an image-based detector.

On a high level, our method is based on the following observation: we can use a sound source localisation network to learn about possible locations of sounding objects in videos. From this, we can extract a collection of bounding box pseudo-annotations for the objects and use those to learn a standard object detector. Because the latter only uses the visual modality, it immediately transfers to the detection of objects even when no relevant sound is present.

However, one challenge is that sound source localisation does not provide the necessary class information to train class-specific detectors, effectively resulting in only learning a region proposal network for generic objects, with high rates of false positives. To this end, we note that most sound source localizers are based on noise-contrastive formulations that, together with clustering-based approaches, occupy the current state-of-the-art in self-supervised representation learning. Leveraging this, we derive a joint formulation that can simultaneously benefit from and learn to localize sound sources and classify them without *any* supervision. The resulting output can then be used to train any off-the-shelf object detector such as a Faster-RCNN [76] to obtain a completely unsupervisedly learned object detector, as shown in Fig. 1.

Empirically, we test our method by training and testing on VGGSound [21] and AudioSet [32], as well as testing

only on a subset of OpenImages [56].

2. Related Work

Audio-Visual Sound Source Localisation. Early work in sound-source localisation includes probabilistic models for localisation [27, 42, 53] and segmentation [48], but more recently the focus has shifted to dual-stream neural networks. For example, [7, 38, 82] propose a contrastive learning approach that matches the visual and audio components of the data. The work of [43, 47] instead clusters visual and audio features, associating to them centroids by means of a contrastive loss. Other works [2, 67] learn heatmaps by exploiting audio-visual synchronization in the same video, used previously for lip-to-mouth synchronization and active-speaker detection [24, 61], or by leveraging explicit attention modules [51]. Zhao et al. [108, 109] learn to associate pixels with audio sources by training with a mix-and-separate objective. Others [73] combine activation maps learned from class labels [19, 81] with a contrastive objective, use different levels of supervision and fusion techniques [74], or improve heatmaps by mining hard negative locations [20].

The work most similar to ours is [45], who first train a source localisation model with a contrastive objective and then use the learned heatmaps to extract object representations that are clustered using K-means. The cluster assignments are then used to train classifiers on top of the audio and video encoders. The paper proceeds to use these learned representations to discriminatively localize sound sources while suppressing quiet objects in ‘cocktail party’ scenarios.

Compared to our work, none of the above can detect and thus enumerate individual object occurrences because they produce heatmaps. Furthermore, they all require audio during inference, and therefore cannot be used on individual images or to detect silent objects.

Audio-visual category discovery. Learning visual categories is usually cast as image clustering, for which there is abundant prior work, such as recent ‘deep clustering’ methods [10, 18, 49, 95, 101, 103], or clustering with segmentation [97]. However, there is less work for clustering audio-visual data. In [5] the authors extend Deep Cluster [18] to the video domain by constructing two sets of labels from opposing modalities, which are used for cross-modal representation learning. The work of [80] combines clustering with audio-visual co-segmentation achieving combined audio-visual source separation. In [9], the authors extend the self-labelling method of [10] to multi-modal data by learning a shared set of labels between the two modalities. This work builds on the latter to complement and boost sound source localisation in a joint learning framework.

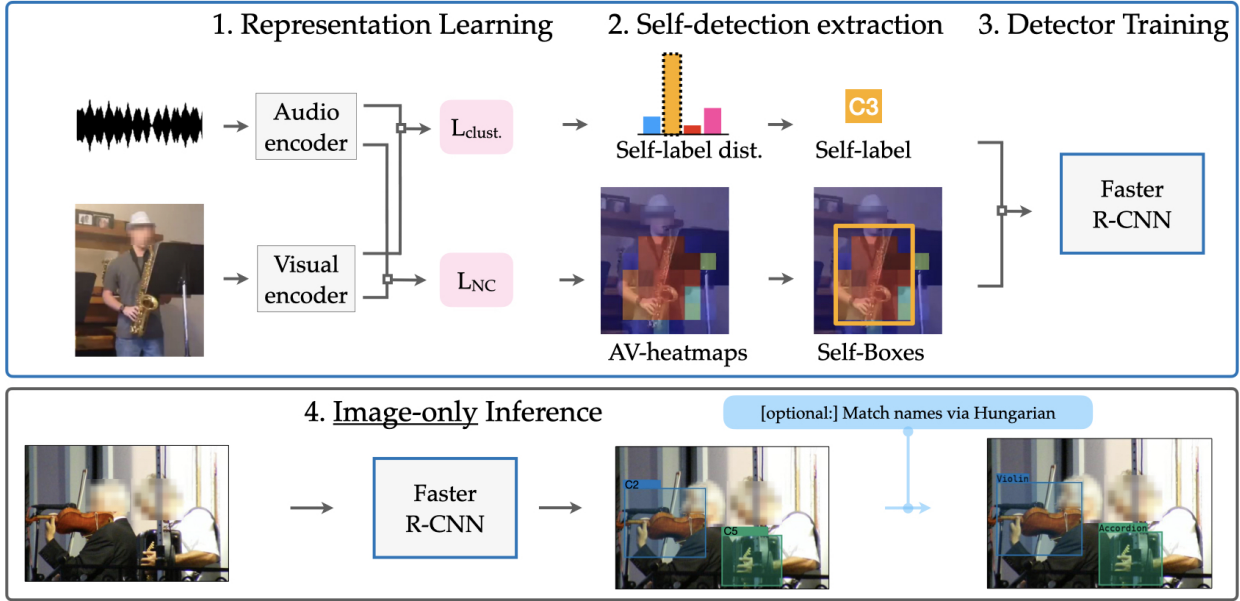


Figure 2: Self-supervised object detection from audio-visual correspondence: We combine noise-contrastive and clustering-based self-supervised learning to generate self-detections (boxes and labels) and use those as targets to train a detector. The trained detector can be used to detect objects from many categories on images without requiring audio.

Weakly Supervised Object Detection (WSOD). Weakly supervised detection uses (manual) image-level category labels without bounding box annotations. Many approaches are based on a form of multiple-instance learning [17, 33, 91, 98, 102, 104, 105, 107], or proposal clustering [90]. Recent works in the area [50, 77] combine a variety of ideas, such as self-training [112] and spatial dropout [99] or explore the use of mixed annotations [78]. Other works obtain improvements by adding curriculum learning [106], using motion cues in videos [86], adversarial training [84], combining segmentation and detection [31, 58, 83], or modelling the uncertainty of object locations [8].

Other methods use technique such as CAM or analogous techniques [15, 19, 28, 81, 85, 111] as a form of weakly supervised saliency or localisation maps. Recent works have suggested that saliency methods can also be applied to self-supervised networks [13, 35], e.g. for object co-localisation [13].

Self-supervised multi-modal learning. Our work is also related to methods that use multiple modalities for representation learning [5, 6, 10, 12, 63, 70, 71] and synchronization [25, 54, 68]. A number of recent papers have leveraged speech as a weak supervisory signal to train video representations [57, 62, 64, 87, 88] whereas [3] uses speech, audio and video. Some works distil knowledge learned from one modality into another [1, 4, 29, 110]. Other works incorporate optical flow and other modalities [36, 37, 72, 108] to learn representations. For instance, the work of [93] learns

to temporally localize audio events through audio-visual attention. CMC [92] learns representations that are invariant to multiple views of the data such as different color channels. Multi-modal self-supervision is also used to learn sound source separation in [30], albeit they assume to have pre-trained detectors.

3. Method

Our goal is to learn object detectors using only unlabeled videos, simultaneously learning to enumerate, localize and classify objects. Our approach consists of three stages summarized in Fig. 2: first, we learn useful representations using clustering and contrastive learning; second, we extract bounding boxes and class categories by combining the trained localisation and classification networks; third, we train an off-the-shelf object detector by using these self-extracted labels and boxes as targets.

Next, we explain each stage and refer the reader to the Appendix for further architecture and training details.

3.1. Representation Learning

Sound source spatial localisation. Our method starts by training a sound source localisation network (SSLN) using a contrastive learning formulation inspired by [7]. The SSLN is learned from pairs (v, a) , where $v \in \mathbb{R}^{3 \times H \times W}$ is a video frame (i.e., a still image) and $a \in \mathbb{R}^{T \times F}$ is the spectrogram of the audio captured in a temporal window centered at that particular video frame.

We consider a pair of deep neural networks. The first network $f^v(v) \in \mathbb{R}^{C \times h \times w}$ extracts from the video frame a field of C -dimensional feature vectors, one per spatial location. We use the symbol $f_u^v(v) \in \mathbb{R}^C$ to denote the feature vector associated to location $u \in \psi = \{1, \dots, h\} \times \{1, \dots, w\}$. Here $h \times w$ is the resolution at which the spatial features are computed and is generally a fraction of the video frame resolution $H \times W$. The second network $f^a(a) \in \mathbb{R}^C$ extracts instead a feature vector for the audio signal.

Importantly, the spatial and audio features share the same C -dimensional embedding space and can thus be contrasted. We further assume that the vectors $f_u^v(v)$ and $f^a(a)$ are L^2 normalized (this is obtained by adding a normalization layer at the end of the corresponding networks). The cosine similarity of the two feature vectors is then used to compute a heatmap of spatial locations, with the expectation that objects that are correlated with the sounds would respond more strongly. This heatmap is given by:

$$h_u(v, a) = \langle f_u^v(v), f^a(a) \rangle / \rho, \quad u \in \psi,$$

where ρ is a learnable temperature parameter.

For the multi-modal contrastive learning formulation [26, 66, 71], the heatmap is converted in an overall score that the video v and audio a are in correspondence. This is done by taking the maximum of the response:

$$S(v, a) = \max_{u \in \psi} h_u(v, a).$$

The contrastive learning objective is defined by considering videos $(v, a) \in \mathcal{B}$ in a batch $\mathcal{B}$. This comprises two terms. The first tests how well a video frame matches with its specific audio among the ones available in the batch:

$$\mathcal{L}_{a \rightarrow v}(\mathcal{B}) = -\frac{1}{|\mathcal{B}|} \sum_{(v, a) \in \mathcal{B}} \log \frac{\exp S(v, a)}{\sum_{(v', a') \in \mathcal{B}} \exp S(v, a')}.$$

The second is analogous, testing how well an audio matches with its specific video frame:

$$\mathcal{L}_{v \rightarrow a}(\mathcal{B}) = -\frac{1}{|\mathcal{B}|} \sum_{(v, a) \in \mathcal{B}} \log \frac{\exp S(v, a)}{\sum_{(v', a') \in \mathcal{B}} \exp S(v', a')}.$$

These two losses are averaged in the *noise-contrastive* loss:

$$\mathcal{L}_{\text{NC}}(\mathcal{B}) = (\mathcal{L}_{a \rightarrow v}(\mathcal{B}) + \mathcal{L}_{v \rightarrow a}(\mathcal{B}))/2 \quad (1)$$

Category self-labeling. Spatial localisation does not provide any class information, whereas our goal is to also associate ‘names’ to the different objects in the dataset. To this end, we consider the self-labelling approach of [9]. To briefly explain the formulation, let $y(v, a) \in \mathcal{Y} = \{1, \dots, K\}$ be a label associated to the training pair (v, a) .

We also consider two classification networks. The first maps a video v to class scores $g^v(v) \in \mathbb{R}^K$ and is optimized by minimizing the standard cross-entropy loss:

$$\mathcal{L}_v(\mathcal{B}|y) = -\frac{1}{|\mathcal{B}|} \sum_{(v, a) \in \mathcal{B}} \log \text{softmax}(y(v, a) | g^v(v)).$$

Note that this classification loss is equivalent to a contrastive loss on the cluster indices (as opposed to image indices) without normalization: As the last classification layer can be viewed as computing dot-products with each corresponding cluster’s feature, it pushes the representation towards the feature of the corresponding cluster and away from the other clusters. The other network $g^a(a)$ is analogous, but uses the audio signal:

$$\mathcal{L}_a(\mathcal{B}|y) = -\frac{1}{|\mathcal{B}|} \sum_{(v, a) \in \mathcal{B}} \log \text{softmax}(y(v, a) | g^a(a)).$$

As noted in [9], the crucial link between the two losses is that the labels y are shared between modalities. This is obtained by averaging the two losses:

$$\mathcal{L}_{\text{clust}}(\mathcal{B}|y) = (\mathcal{L}_v(\mathcal{B}|y) + \mathcal{L}_a(\mathcal{B}|y))/2. \quad (2)$$

Note that the labels y are unknown; following [9] these are learned in an alternate fashion with the classification networks, minimizing the same loss (2). In order to avoid degenerate solutions, the labels’ marginal distribution must be specified, e.g. using a simple equipartitioning constraint:

$$\frac{1}{|\mathcal{D}|} \sum_{(v, a) \in \mathcal{D}} 1_{\{y(v, a) = k\}} = \frac{1}{K} \quad \text{for all } k = 1, \dots, K \quad (3)$$

where $\mathcal{D}$ denotes the entire dataset (union of all batches). Optimizing y can be done efficiently by using the SK algorithm as in [9].

Joint training. To summarize, given the dataset $\mathcal{D}$, we optimize stochastically w.r.t. random batches $\mathcal{B}$ the loss:

$$\mathcal{L}(\mathcal{B}|y) = \lambda \mathcal{L}_{\text{NC}}(\mathcal{B}) + (1 - \lambda) \mathcal{L}_{\text{clust}}(\mathcal{B}|y) \quad (4)$$

where λ is a balancing hyperparameter.

The loss is optimized with respect to the localisation networks f^v and f^a and the classification networks g^v and g^a . These networks share common backbones q^v and q^a and differ only in their heads, so they can be written as $f^v = \hat{f}^v \circ q^v$, $g^v = \hat{g}^v \circ q^v$, $f^a = \hat{f}^a \circ q^a$ and $g^a = \hat{g}^a \circ q^a$.

The model is trained by alternating between updating the labels y with eq. (2) under constraint (3) and updating the networks by optimizing eq. (4).

3.2. Extraction of Self-labels for Detection

Once the localisation and classification networks have been trained, they can be used to extract self-annotations for training a detector. This is done in two steps: extracting object bounding boxes and finding their class labels.

Box extraction. To obtain “self-bounding boxes” for the objects, we use the simple heuristic suggested by [111]: the heatmap $h(v, a)$ is thresholded at a value $\epsilon(h)$, the largest connected component is identified, and a tight bounding box $t^*(v, a) \in \Omega^2$ around that component is extracted (the notation means that the box is specified by the location of the top-left and bottom-right corners).

The threshold is determined dynamically as a convex combination of the maximum and average responses of the heatmap, controlled by hyperparameter β :

$$\epsilon(h) = \beta \max_{u \in \psi} h_u + (1 - \beta) \frac{1}{|\psi|} \sum_{u \in \psi} h_u. \quad (5)$$

Class labelling. As noted above, we only extract a single object from each frame for the purpose of training the detector. Likewise, we only need to extract a single class label for the frame. This is done by taking the maximum response of the visual and audio-based classification networks:

$$y^*(v, a) = \arg \max_{y \in \mathcal{Y}} [g_y^v(v) + g_y^a(a)]. \quad (6)$$

Filtering the annotations. The assumption that frames contain a dominant object introduces noise but simplifies the problem and gives us the ability to use the audio to obtain purer clusters. Notably, we do not require the method above to work for *all* frames but instead rely on our detector to smooth over the specific and noisy self-annotations to learn a holistic detection.

3.3. Training the Object Detector

The process described above results in a shortlist of training triplets $(v, t^*, y^*) \in \mathcal{D}_{\text{det}}$, where v is a video frame (an image), t^* is the extracted bounding box and y^* is its class label. We use this dataset to train an off-the-shelf detector, in particular Faster R-CNN [75] for its good compromise between speed and quality.

Recall that, given an image v , Faster R-CNN detector considers a set of bounding box proposals $m \in M(v) \subset \Omega^2$. It then trains networks $y(m) = f_{\text{det}}^{\text{cls}}(m|v) \in \{1, \dots, K, \text{bkg}\}$ and $t(m) = f_{\text{det}}^{\text{loc}}(m|v) \in \mathbb{R}^4$ inferring, respectively, the class label $y(m)$ and a refined full-resolution bounding box $t(m)$ for the box proposal m . The label space is extended to also include a *background class* bkg , which is required as most proposals do not land on any object.

The detector is trained by finding an association between proposals and annotations. To this end, if $m^* = \arg \max_{m \in M(v)} \text{IoU}(m, t^*)$ is the proposal that matches the pseudo-ground truth bounding box t^* the best, one optimizes:

$$\begin{aligned} \mathcal{L}_{\text{det}}(v, t^*, y^*) &= \mathcal{L}_{\text{reg}}(t(m^*), t^*) + \mathcal{L}_{\text{cls}}(y(m^*), y^*) \\ &+ \sum_{m \in M(v): \text{IoU}(m, t^*) < \tau} \mathcal{L}_{\text{cls}}(y(m), \text{bkg}). \end{aligned}$$

Here $\mathcal{L}_{\text{reg}}$ is the L^1 loss for the bounding box corner coordinates and $\mathcal{L}_{\text{cls}}$ the standard cross-entropy loss. Intuitively, this loss requires the best proposal m^* to match the pseudo-ground truth class y^* and bounding box t^* of, while mapping proposal m that are a bad match ($\tau \leq 0.7$) to class bkg . Further details, including how the region-proposal network that generates the proposals is trained, are given in the Appendix

Discussion. Training a detector is obviously necessary to solve the problem we set out to address. However, it can also be seen as a way of extracting ‘clean’ information from the noisy self-annotations. Specifically: (i) the noise in individual annotations is smoothed over the entire dataset; (ii) because of the built-in NMS step, the detector still learns to extract multiple objects per image even though a single self-annotation is given for each training image; (iii) by learning to reject a large number of false bounding box proposals, the detector learns to be more precise than the self-annotations are.

4. Experiments

We first introduce the datasets, experimental setup and relevant baselines; we then test our method against those, analyse it further via ablations and its capacity to generalize.

4.1. Datasets

AudioSet-Instrument. AudioSet [32] is a large scale audio-visual dataset consisting of 10-second video clips originally from YouTube. For training we use the *AudioSet-Instruments* [7] subset of the “unbalanced” split, containing 110 sound source classes as well as its more constrained subset used by [45] spanning 13 instrument classes. Following previous work we use the “balanced” subset for evaluation on the annotations provided by [45].

VGGSound. VGGSound contains over 200K 10-second clips from YouTube spanning 309 categories of objects where there is some degree of correlation between the audio and the video. We create one subset by keeping only the 50 musical instrument categories yielding around 54K training videos, and one other subset, by keeping from those only the 39 categories that can be roughly mapped to the test-set annotations (details in the Appendix). For VGGSound pseudo-ground truth test-set annotations are obtained using a supervised detector from [30], following [45].

OpenImages. For evaluation, we also use the subset of the OpenImages [56] dataset containing musical instruments, which spans 15 classes.

4.2. Baselines

We are not aware of any prior work that learns an object detector for multiple object classes without any supervision. Instead, we compare against weakly-supervised detectors

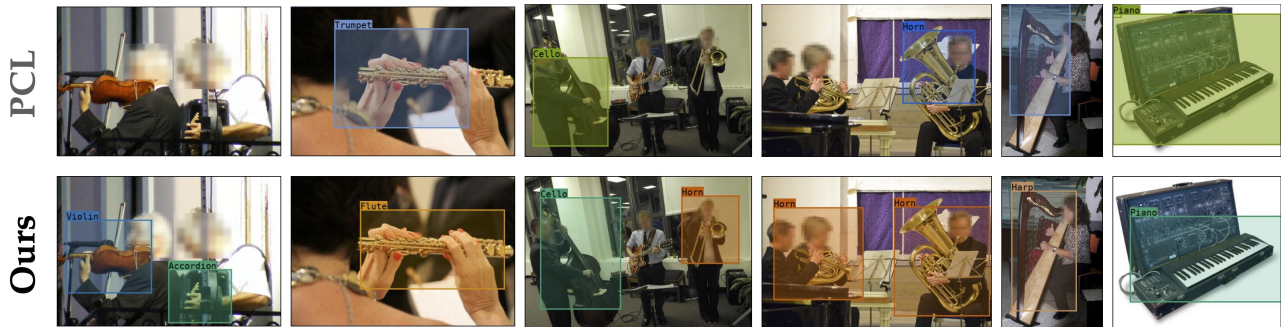


Figure 3: **Qualitative results and comparison** with a weakly supervised object detection method, PCL [90], on the Open-Images test set. Our method accurately detects objects, capturing their boundaries, even though it has been trained without *any* supervision. For visualisation purposes, we show the labels obtained from matching with the Hungarian method. More qualitative results provided in the the Appendix.

(hence using image-level labels) and unsupervised localisation methods that only produce heatmaps (not detections).

Weakly-supervised detection. For weakly-supervised detection, we consider PCL [89], the strongest such baseline for which we could find an implementation. Image-level labels are obtained from the corresponding dataset: for AudioSet the 13 labels of the training set are used, and for VGGSound we manually merge the 39 sub-classes to equivalent 15 classes (e.g., electric guitar, acoustic guitar to “Guitar” etc.); see the Appendix for full details. However, we have found that training PCL directly on the same data as our method (i.e., random clips from the VGGSound and AudioSet-Instrument subsets) does not work, likely due to the high amount of noise present in the labels (e.g., several videos are labelled with an instrument, which is however not visible at all). To avoid this issue, we further preprocess the training data with a supervised instrument detector [30] and only retain frames where at least one relevant detection is found. This of course gives an “unfair” advantage to the baseline, but it is necessary to be able to use it at all.

Heatmap-based localisation. For our second baseline, we consider localisation methods that, similarly to us, use cross-modal self-supervised learning. The state-of-the art DSOL method of [45] is the most relevant, as it produces a heatmap roughly localizing the objects and produces class pseudolabels. While DSOL does not use image-level labels like PCL, it does use audio during inference, and thus strictly more information than our method (which performs localisation only in the visual domain).

Region proposals. Finally, we also compare against other baselines such as simply predicting a large centered box and class-agnostic region proposal methods such as selective search, and using a RPN obtained from supervised training on COCO [59].

4.3. Implementation Details.

Assessing class pseudolabels. Since class pseudolabels do not come with the “name” of the class (they are just cluster indices), they must be put in correspondence with human-labelled classes for evaluation. Following prior work in unsupervised image clustering [10, 14, 49, 96], we apply Hungarian matching [55] to the learned clusters and the ground truth classes. Importantly, the matching is done *after* the detector is trained and only done for assessment; meaning that the detector does not use any manual label.

Training resolutions. We train our method on random square crops of 224 pixels after resizing to 256 pixels. During the training of the detector, we take random 224 crops and obtain the self-supervised bounding boxes on-the-fly from our pretrained model, which are scaled and used to train the detector at the larger detector resolution.

Detector warm-up. We warm-up the Faster R-CNN detector by training in a class-agnostic manner for 20 epochs. This gives the RPN (which is randomly initialized) an opportunity to learn sufficiently stable bounding box proposals; we then switch to full class-aware supervision. We found that this leads to more robust convergence compared to training with pseudo-labels from the start.

Backbone pretraining. We also found it beneficial to pretrain the localizer backbone using only the localisation loss on the full AudioSet-Instruments dataset, and the detector backbone using self-supervised SimCLR [22] on ImageNet (note that the DSOL baseline uses instead *supervised* ImageNet pretraining for the backbone).

Detector training. If not stated otherwise, the localizer and detector are trained on VGGSound and AudioSet whereas OpenImages are only used for evaluation. We do not have any information on the number of instruments in VGGSound and use all videos with no single/multi-object curation. For a fair comparison with DSOL, and only for

Method	No labels?	VGGSound			Audioset			OpenImages		
		mAP ₃₀	mAP ₅₀	mAP _[50:95:5]	mAP ₃₀	mAP ₅₀	mAP _[50:95:5]	mAP ₃₀	mAP ₅₀	mAP _[50:95:5]
PCL (WSOD) [90]	✗	54.9	27.7	7.6	39.0	17.5	4.4	37.9	14.5	3.5
Ours - weak sup.	✗	67.6	42.9	14.2	50.6	30.9	10.3	48.9	33.7	9.5
Center Box*	✓	29.6	5.6	1.5	15.1	3.5	0.7	20.7	4.2	0.8
Selective Search* [94]	✓	5.2	1.1	0.4	2.8	0.4	0.1	7.4	2.1	0.7
COCO-trained RPN*	✗	33.4	7.5	1.6	19.0	4.1	0.8	24.4	11.1	2.6
Ours - self-boxes*	✓	48.1	29.6	10.0	27.8	14.1	4.8	NA	NA	NA
Ours - full	✓	52.3	39.4	14.7	44.3	28.0	9.6	39.9	28.5	7.6

Table 1: **Self-supervised object detection.** We report object detection metrics across three test datasets and find our method is far superior to other unsupervised approaches and outperforms even the weakly supervised baseline in most metrics. For methods denoted by *, we report class-agnostic evaluation numbers. *Center Box* denotes a simple baseline predicting random sized boxes in the middle of the frame. *Ours-weak sup.* is a variant of our model trained with the video-level category annotations in combination with our self-extracted boxes. The class-agnostic performance of the self-boxes that are used to train the detector reveals that the latter greatly outperforms them, which highlights the benefit of our approach.

Method	single-instr.		multi-instr.
	IoU-0.5	AUC	cIoU-0.3
Sound of pixels [109]	38.2	40.6	39.8
Object t. Sound [7]	32.7	39.5	27.1
Attention [82]	36.5	39.5	29.9
DMC [44]	32.8	38.2	32.0
DSOL [45]	38.9	40.9	48.7
Ours	50.6	47.5	52.4

Table 2: **Comparison to sound localisation methods.** Since our detector does not require audio, we obtain detections on the video frames directly. Our model outperforms the baselines. Baselines numbers taken from [46].

the relevant experiment in Table 2, we train on AudioSet using the single-instrument subset for learning the localizer.

Number of clusters. For VGGSound training we use $K = 39$ if not stated otherwise, matching the 39 object types in the training set. Since the dataset is roughly balanced, uniform marginals are used as described in [9]. For AudioSet training we use $K = 30$ and Gaussian marginals.

4.4. Results

Self-supervised object detection. We summarise the results of our evaluations on the three test sets that we consider in Table 1. Following the image object detection literature, we use mAP at different IOU thresholds as the evaluation metric.

Our method clearly outperforms the PCL baseline even though it uses no manual annotations at all during training. PCL outperforms our approach in some of the datasets only if the IoU threshold used for mAP computation is relaxed

substantially (0.3 IoU). However, for stricter thresholds our approach works better, which suggests that our detections have a relatively high spatial accuracy.

To understand the impact of the noisy class self-labels, we also train and test a detector (Ours - weak sup.) with the bounding box labels from our localisation network, but utilising the ground truth video categories. The resulting performance difference is modest, resulting in a 3% AP50 drop in VGGSound and AudioSet. This further demonstrates the accuracy of our class self-labels, but also shows that our method has also the potential to leverage weak supervision if available.

Per-class performance breakdown. To better understand the strengths and weaknesses of our method, we report a performance breakdown by object class in Table 3. We observe that the model obtains good results consistently for classes of large objects with a distinctive appearance (e.g. accordions and harps), while it is weaker for smaller objects such as oboes, or for objects that appear closely in numbers, like drums.

Comparison to audio-visual heatmaps. In Table 2 we compare the performance of our method trained on AudioSet to state-of-the-art sound source localisation methods. For a fair comparison to these methods, we convert the union of our predicted bounding boxes with confidence above a set threshold into a binary map, and use the latter as a pseudo-heatmap to use the same evaluation code. Our approach outperform others for both class-agnostic single object localisation and for class-aware multi-object localisation, *without* using audio signals during inference.

We note however that cIOU is not a very reliable metric for evaluating a detector (or even sound localizer) as it favours high recall over precision: by averaging this metric over all classes the most frequent ones (e.g. drums, guitars,

Dataset	mAP ₅₀	Accordion	Cello	Drum	Flute	Horn	Guitar	Harp	Piano	Saxophone	Violin	Banjo	Trombone	Trumpet	Oboe
OpenImages	28.5	75.3	30.2	6.6	6.5	15.0	14.5	80.4	28.8	22.5	28.8	57.0	9.7	18.1	6.3
Audioset	28.0	41.3	44.9	0.9	5.5	21.7	39.5	82.6	52.7	2.5	17.4	46.7	8.0	-	-
VGGSound	39.4	88.6	39.4	1.8	50.0	3.4	34.9	95.6	50.2	14.4	56.3	100.0	2.2	11.0	3.8

Table 3: **Per-class mAP breakdown** For entries with ‘-’ the test set does not contain any samples for that class.

β	#boxes	mAP ₅₀	
		VGGS	O.Images
0.7	single	39.4	28.5
0.8	single	36.6	29.4
0.9	single	35.8	28.8
0.7-0.9	single	37.5	29.4
0.7-0.9	multi	38.0	29.3

Table 4: **Ablation of hyperparameter β** , which controls the relative width of the bounding box; *multi* denotes the use of multiple self-labelled boxes per sample. Overall the method is fairly stable with respect to the choice of this parameter, however sampling β from a range (0.7-0.9) obtains a better balanced performance. Moreover we do not observe any substantial improvement from using multiple boxes.

# GT-cls.	K	mAP ₅₀	
		VGGS	O.Images
39	20	34.4	24.4
39	30	35.1	25.1
39	39	39.4	28.5
39	50	41.0	27.5

Matching	mAP ₅₀	
	VGGS	O.Images
Hung.	39.4	28.5
Argmax	39.6	30.1
Manual	41.0	29.5
1-shot	36.4	25.1
10-shot	37.1	25.8

Table 5: **Number of clusters K.** Our method is relatively robust ($< 5\%$ decrease in detection AP) to the number of clusters used for self-labelling.

Table 6: **Matching strategies.** Even with as little as 39 labels, our method can detect and classify objects accurately.

pianos) dominate the metric. We therefore propose to the research community – and report in this paper – mean average precision (mAP) values as a more indicative metric.

Ablation: Thresholding parameter. In Table 4 we investigate the influence of the hyper-parameter β and the number of extracted target boxes we use for training the detector. From Eq. (5), a smaller β makes the heatmaps more focused and specific. With regards to this parameter, we find somewhat inverse trends for VGGSound vs OpenImages, where smaller β yields better results for the former and larger β for the latter. We find that a good balance in terms of performance can be achieved by sampling β randomly from a range, as over-specific boxes for some images and under-specific ones are successfully combined during detector training. Even when extracting multiple boxes for training the detector, we find our method performs similarly

well to when only extracting a single box.

Ablation: Number of clusters K. In Table 5, we perform an experiment varying the number of clusters, and as a consequence the number of object categories that the detector learns, while keeping the test-set (containing 15 classes) fixed. We observe that our method achieves reasonable performance for a wide range of the number of clusters. The performance is fairly stable when using more clusters than the ground truth classes, and gradually decreases when fewer clusters are used.

Data-efficient detector alignment. In Table 6 we conduct an investigation into the matching of the clusters to the ground truth labels. First, we compare Hungarian matching to simply taking the argmax of the ground truth class per self-label (i.e. assigning to the most frequent class). We find this yields almost the same results for VGGSound and a gain of 1.6% on OpenImages. By refining the Hungarian assignments via manually grouping similar classes and mapping each group to one of the test classes (for example mapping ‘piano’, ‘electronic organ’ and ‘Hammond organ’ all to ‘piano’; see the Appendix for details), we find another small additional gain can be realized. While the Hungarian and argmax are common evaluation methods for the self-supervised clustering domain, we note that they are unsatisfactory as they still implicitly require a large set of labels. To alleviate this, we devise a “data-efficient” class-matching procedure as follows: Per pseudo-class, obtain the m videos which have the highest response for being in that particular pseudo-class (averaged across audio and video) and obtain the class-label for these m samples. The pseudo-class is then assigned the most frequent ground truth class among these m . Overall, this reduces the number of labels required down to Km , making it a more realistic and scalable evaluation method for self-supervised approaches. Following this procedure, we find that even by just using $m = 1$ (i.e., a total number of 39 annotations), our method still achieves high performances of 37.1% and 25.3% on VGGSound and OpenImages. This 3% drop compared to the Hungarian can be further decreased to around 2.3% by using 10 labels per pseudo-class, for a total labelling budget of 390 images.

Qualitative analysis. We show examples of successfully detected objects in challenging images in Fig. 3, where we also include the outputs of the PCL baseline. Although our model has not been manually shown any objects boundaries during training we see that it can learn very accurate boxes around them and that it can successfully identify multiple

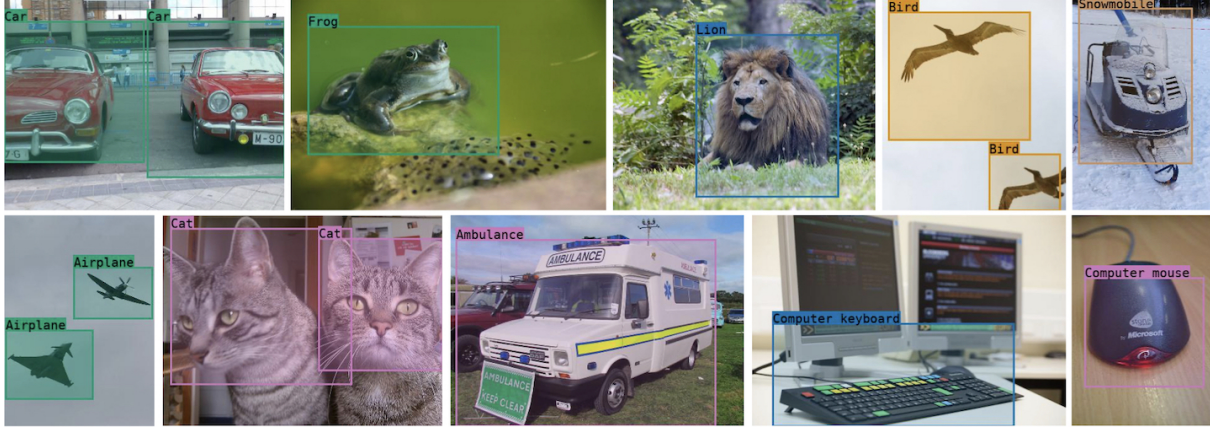


Figure 4: **Object detection beyond musical instruments.** Our proposed method can learn to accurately detect objects from more general categories, as long as they can be associated with a characteristic sound. The results shown here are from a model trained without labels directly on the full VGGSound dataset which includes 309 different video classes. Our method successfully learns to detect non-instrument objects, even in difficult multi-instance cases.

objects in complicated scenes. We provide further examples in the Appendix.

4.5. Towards general object detection

The results presented thus far have focused on subsets of common datasets with instruments solely to ensure comparability with prior works. Since one main goal of self-supervised learning is to leverage the vast amount of unlabelled data, we wish to investigate how general and robust our proposed method when applied on a far larger scale. For this, we increase our pretraining dataset by approximately $10\times$, simply by taking the whole of the VGGSound dataset, without any filtering. We set the number of learned clusters K to 300 and keep all training parameters the same; the result is an unsupervisedly trained object detector that can classify 300 pseudo-classes. As before, we match these to the VGGSound labels with the Hungarian algorithm and out of these take ten categories for which we have annotations in the OpenImages dataset (details in the Appendix).

Class	AP ₃₀	AP ₅₀	AP _[50:95:5]
Mean	45.6	24.4	6.5
Airplane	62.7	27.0	6.5
Ambulance	56.9	30.9	7.1
Bird	26.5	15.8	3.7
Car	29.8	18.4	5.1
Cat	67.7	28.0	7.7
Comp. Keyboard.	53.3	42.6	12.9
Comp. Mouse	35.9	25.4	8.8
Frog	43.5	19.5	4.7
Lion	34.1	22.2	4.9
Snowmobile	64.3	14.3	3.5

Table 7: **Results on general object categories.**

In Fig. 4 we show qualitative results of some detections on OpenImages. The numerical results are given

in Table 7. We find that even for objects that are deformable, such as cats, we get high AP₃₀ values of 67.7% and that even objects that vary in shape, such as airplanes (see Fig. 4, bottom-right), we achieve a good performances 62.7%. While the results for the AP_{50:95:5} metric indicate that there is still room for improvement, these initial results show that leveraging larger and more diverse video datasets for self-supervisedly learning object detectors is a promising avenue. We note that, since minimal curation is performed on the training data, and we use a large number of different object categories in a noisy dataset, this training setting is very challenging. These results further highlight the potential of our proposed method.

5. Conclusion

We have presented a method for training strong object detectors purely with self-supervision by watching unlabelled videos. We demonstrated that our best models perform better than a weakly supervised baseline, even after curating the dataset to filter out noisy samples for training the latter. Our method also outperforms heatmap-based methods in music instruments localisation, while having the ability to detect objects in images directly without requiring audio. We have also addressed one short-coming of using the Hungarian for evaluation by showing that data-efficient alignment of self-supervised detectors is possible with as little as one image per pseudo-label. Finally we applied our method to domains beyond musical instruments and found that it can learn reasonable detectors in this much less curated setting, paving the way to general self-supervised object detection.

Sound is a great natural source of supervision for training detectors; we believe our method will be the first of many to explore this exciting new direction.

References

- [1] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. ASR is all you need: Cross-modal distillation for lip reading. In *International Conference on Acoustics, Speech, and Signal Processing*, 2020. 3
- [2] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *Proc. ECCV*, 2020. 2
- [3] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelovic, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *Proc. NeurIPS*, 2020. 3
- [4] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Emotion recognition in speech using cross-modal transfer in the wild. In *Proc. ACMM*, 2018. 3
- [5] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2, 3
- [6] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017. 3
- [7] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proc. ECCV*, 2018. 2, 3, 5, 7, 14
- [8] Aditya Arun, C. V. Jawahar, and M. Pawan Kumar. Dissimilarity coefficient based weakly supervised object detection. In *CVPR*, 2019. 3
- [9] Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 4, 7
- [10] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 2, 3, 6
- [11] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. SoundNet: Learning sound representations from unlabeled video. In *Proc. NeurIPS*, 2016. 2
- [12] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, 2016. 3
- [13] Kyungjune Baek, Minhyun Lee, and Hyunjung Shim. Psynet: Self-supervised approach to object localization using point symmetric transformation. In *Proc. AAAI*, 2020. 3
- [14] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Bjorn Ommer. Cliquecnn: Deep unsupervised exemplar learning. In *NeurIPS*, pages 3846–3854, 2016. 6
- [15] Loris Bazzani, Alessandra Bergamo, Dragomir Anguelov, and Lorenzo Torresani. Self-taught object localization with deep networks. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016. 3
- [16] Hakan Bilen, Vinay P. Nambodiri, and Luc Van Gool. Object and action classification with latent variables. In *Proc. BMVC*, 2011. 1
- [17] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3
- [18] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 2
- [19] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018. 2, 3
- [20] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Localizing visual sounds the hard way, 2021. 2
- [21] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: A large-scale audio-visual dataset. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020. 2
- [22] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *ICML*, 2020. 6, 14
- [23] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *arXiv.cs*, abs/2002.05709, 2020. 1
- [24] Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Proc. ACCV*, 2016. 2
- [25] Joon Son Chung and Andrew Zisserman. Out of time: Automated lip sync in the wild. In *ACCV Workshop on Multi-view Lip-reading*, 2016. 3
- [26] Soo-Wan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In *Proc. ICASSP*, pages 3965–3969. IEEE, 2019. 4
- [27] John W Fisher III, Trevor Darrell, William T Freeman, and Paul A Viola. Learning joint statistical models for audio-visual fusion and segregation. In *NeurIPS*, 2000. 2
- [28] Ruth Fong and Andrea Vedaldi. Explanations for attributing deep neural network predictions. In *Proc. ICCV*, 2019. 3
- [29] Chuang Gan, Hang Zhao, Peihao Chen, David Cox, and Antonio Torralba. Self-supervised moving vehicle tracking with stereo sound. In *iccv*, 2019. 3
- [30] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *Proc. ICCV*, 2019. 3, 5, 6, 15
- [31] Weifeng Ge, Sibe Yang, and Yizhou Yu. Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In *CVPR*, 2018. 3
- [32] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. ICASSP*, 2017. 2, 5
- [33] Nicolas Gonthier, Yann Gousseau, Said Ladjal, and Olivier Bonfait. Weakly supervised object detection in artworks.

- Computer Vision – ECCV 2018 Workshops*, page 692–709, 2019. 3
- [34] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Dohersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020. 1
- [35] Shir Gur, Ameen Ali, and Lior Wolf. Visualization of supervised and self-supervised neural networks via attribution guided factorization, 2020. 3
- [36] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *Proc. ECCV*, 2020. 3
- [37] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *Proc. NeurIPS*, 2020. 3
- [38] David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *Proceedings of the European conference on computer vision (ECCV)*, pages 649–665, 2018. 2
- [39] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv.cs*, abs/1911.05722, 2019. 1
- [40] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *Proc. ICCV*, 2017. 14
- [41] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 14
- [42] J Hershey and JR Movellan. Audio-vision: Locating sounds via audio-visual synchrony. In *NeurIPS*, volume 12, 1999. 2
- [43] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2
- [44] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *Proc. CVPR*, 2019. 7
- [45] Di Hu, Rui Qian, Minyue Jiang, Xiao Tan, Shilei Wen, Errui Ding, Weiyao Lin, and Dejing Dou. Discriminative sounding objects localization via self-supervised audiovisual matching. In *NeurIPS*, 2020. 2, 5, 6, 7
- [46] Di Hu, Rui Qian, Minyue Jiang, Xiao Tan, Shilei Wen, Errui Ding, Weiyao Lin, and Dejing Dou. Discriminative sounding objects localization via self-supervised audiovisual matching. *arXiv.cs*, abs/2010.05466, 2020. 7
- [47] Di Hu, Zongge Wang, Haoyi Xiong, Dong Wang, Feiping Nie, and Dejing Dou. Curriculum audiovisual learning. *ArXiv*, abs/2001.09414, 2020. 2
- [48] Hamid Izadinia, Imran Saleemi, and Mubarak Shah. Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Transactions on Multimedia*, 15(2):378–390, 2012. 2
- [49] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9865–9874, 2019. 2, 6
- [50] Zequn Jie, Yunchao Wei, Xiaojie Jin, Jiashi Feng, and Wei Liu. Deep self-taught learning for weakly supervised object localization. In *CVPR*, 2017. 3
- [51] Naji Khosravan, Shervin Ardeshtir, and Rohit Puri. On attention modules for audio-visual synchronization. *arXiv preprint arXiv:1812.06071*, 1, 2018. 2
- [52] Einat Kidron, Yoav Y. Schechner, and Michael Elad. Pixels that sound. In *Proc. CVPR*, 2005. 2
- [53] Einat Kidron, Yoav Y. Schechner, and Michael Elad. Pixels that sound. In *Proc. CVPR*, 2005. 2
- [54] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-operative learning of audio and video models from self-supervised synchronization. 2018. 3
- [55] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 6
- [56] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Tom Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018. 2, 5
- [57] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv.cs*, abs/2001.05691, 2020. 3
- [58] Xiaoyan Li, Meina Kan, Shiguang Shan, and Xilin Chen. Weakly supervised object detection with segmentation collaboration. In *cvpr*, 2019. 3
- [59] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proc. ECCV*, 2014. 6
- [60] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *cvpr*, pages 2117–2125, 2017. 14
- [61] Etienne Marcheret, Gerasimos Potamianos, Josef Vopicka, and Vaibhava Goel. Detecting audio-visual synchrony using deep neural networks. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015. 2
- [62] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. *arXiv.cs*, abs/1912.06430, 2019. 3
- [63] Pedro Morgado, Yi Li, and Nuno Vasconcelos. Learning representations from audio-visual spatial alignment. In *Proc. NeurIPS*, 2020. 3
- [64] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *Proc. CVPR*, 2020. 3
- [65] Minh Hoai Nguyen, Lorenzo Torresani, Lorenzo de la Torre, and Carsten Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *Proc. ICCV*, 2009. 1

- [66] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4
- [67] Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. *Proc. ECCV*, 2018. 2
- [68] Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. In *Proc. ECCV*, 2018. 3
- [69] Andrew Owens, Phillip Isola, Josh H. McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. Visually indicated sounds. In *Proc. CVPR*, 2016. 2
- [70] Andrew Owens, Jiajun Wu, Josh H. McDermott, William T. Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *Proc. ECCV*, 2016. 3
- [71] Mandela Patrick, Yuki M Asano, Polina Kuznetsova, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. *arXiv preprint arXiv:2003.04298*, 2020. 3, 4
- [72] A. J. Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *Proc. CVPR*, 2020. 3
- [73] Rui Qian, Di Hu, Heinrich Dinkel, Mengyue Wu, Ning Xu, and Weiyao Lin. Multiple sound sources localization from coarse to fine. In *Proc. ECCV*, 2020. 2
- [74] Janani Ramaswamy and Sukhendu Das. See the sound, hear the pixels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020. 2
- [75] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *arXiv:1506.01497*, 2015. 5
- [76] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Proc. NeurIPS*, 2016. 2
- [77] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G. Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *Proc. CVPR*, 2020. 3
- [78] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Alexander G. Schwing, and Jan Kautz. UFO²: A unified framework towards omni-supervised object detection. In *Proc. ECCV*, 2020. 3
- [79] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G. Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10598–10607, 2020. 14
- [80] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *Proc. ICASSP*, pages 2357–2361. IEEE, 2019. 2
- [81] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 2, 3
- [82] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *Proc. CVPR*, 2018. 2, 7
- [83] Y. Shen, R. Ji, Y. Wang, Y. Wu, and L. Cao. Cyclic guidance for weakly supervised joint detection and segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 697–707, 2019. 3
- [84] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang. Generative adversarial learning towards fast weakly supervised detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5764–5773, 2018. 3
- [85] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *2017 IEEE international conference on computer vision (ICCV)*, pages 3544–3553. IEEE, 2017. 3
- [86] Krishna Kumar Singh and Yong Jae Lee. You reap what you sow: Using videos to generate high precision object proposals for weakly-supervised object detection. In *CVPR*, 2019. 3
- [87] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv.cs, abs/1906.05743*, 2019. 3
- [88] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proc. ICCV*, 2019. 3
- [89] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE transactions on pattern analysis and machine intelligence*, 42(1):176–191, 2018. 6
- [90] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan L. Yuille. PCL: proposal cluster learning for weakly supervised object detection. *CoRR*, abs/1807.03342, 2018. 3, 6, 7
- [91] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *Proc. CVPR*, 2017. 3
- [92] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 3
- [93] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Proc. ECCV*, 2018. 3
- [94] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 2013. 7
- [95] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision (ECCV)*, 2020. 2
- [96] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Proceedings of the European Conference on Computer Vision*, 2020. 6
- [97] Wouter Van Gansbeke, Simon Vandenhende, Stamatios

- Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. *arxiv preprint arxiv:2102.06191*, 2021. 2
- [98] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. C-mil: Continuation multiple instance learning for weakly supervised object detection. In *cvpr*, 2019. 3
- [99] Xiaolong Wang, Abhinav Shrivastava, and Abhinav Gupta. A-Fast-RCNN: Hard positive generation via adversary for object detection. In *Proc. CVPR*, 2017. 3
- [100] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, 2018. 1
- [101] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016. 2
- [102] G. Yan, B. Liu, N. Guo, X. Ye, F. Wan, H. You, and D. Fan. C-midn: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9833–9842, 2019. 3
- [103] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *CVPR*, pages 5147–5156, 2016. 2
- [104] Zhenheng Yang, Dhruv Mahajan, Deepti Ghadiyaram, Ram Nevatia, and Vignesh Ramanathan. Activity driven weakly supervised object detection. In *cvpr*, 2019. 3
- [105] Zhaoyang Zeng, Bei Liu, Jianlong Fu, Hongyang Chao, and Lei Zhang. Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *iccv*, 2019. 3
- [106] Xiaopeng Zhang, Jiashi Feng, Hongkai Xiong, and Qi Tian. Zigzag learning for weakly supervised object detection. In *CVPR*, 2018. 3
- [107] Y. Zhang, Y. Bai, M. Ding, Y. Li, and B. Ghanem. W2f: A weakly-supervised to fully-supervised framework for object detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 928–936, 2018. 3
- [108] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. *Proc. ICCV*, 2019. 2, 3
- [109] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proc. ECCV*, 2018. 2, 7
- [110] M. Zhao, T. Li, M. A. Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi. Through-wall human pose estimation using radio signals. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018. 3
- [111] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 3, 5
- [112] Yang Zou, Zhiding Yu, Xiaofeng Liu, B. V. K. Vijaya Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proc. ICCV*, 2019. 3

Appendix

This Appendix provides: additional qualitative results including failure analysis (Sec. A), further implementation details (Sec. B) and label mapping details (Sec. C).

A. Qualitative analysis & failure cases

In Figure A.1 we show more qualitative results for the general object detection experiment. In Figure A.2 we show some more examples of our trained detector’s output on detecting musical instruments on the OpenImages test set, including failure cases. As noted in the main paper, the model’s detections are very precise, especially for visually distinct categories such as Accordion, Harp, Cello etc.

The failure cases can be attributed to either incorrect box predictions or misclassifications. Box-prediction failure modes most commonly manifest as i) grouping multiple instances into the same detection, ii) missing instances or iii) detection of only parts of an object. All those issues are common and well documented in the WSOD literature [79]. We observe that in the case of objects that appear with high granularity – e.g. drums – although qualitatively the detections seem reasonable, the multiple instance grouping problem is prominent; this explains our model’s low performance by quantitative measures (mAP) for these classes.

Another error is detecting parts of wrong objects that often appear together with the objects of interest due to biases in the data. For example sometimes mouth regions are detected as wind instruments. This is expected when using the audio-visual correspondence as the training signal.

Wrong classification on the other hand occurs often because i) visually similar classes are confused - e.g. Saxophone and Oboe, Horn and Trumpet; ii) of incorrect cluster-to-ground-truth-label matching, e.g. cluster with high Trombone purity matched to Horn; iii) sometimes the model confuses the class due to the object’s orientation – for example a vertical violin is detected as a cello or vice versa a horizontal cello is detected as violin.

B. Additional implementation details

Visual & audio encoder architecture. For the visual and audio encoders (q^v, q^a) we use the AVOL-Net [7] architecture. Each network is a 8-layer CNN with an architecture similar to VGG-M. For a typical $224 \times 224 \times 3$ RGB image input, the video encoder outputs a 14×14 visual feature map. The inputs to the audio encoder are 257×200 dimensional log-mel spectrograms, extracted from 1 second of audio sampled at 24Hz. The output of the encoder is average pooled, resulting in a single audio feature vector.

Localisation and classification heads. As localisation $f^{\{v,a\}}$ and classifier $g^{\{v,a\}}$ networks we use for each modality a 2-layer MLP with a hidden dimension of 512.

The MLPs do not share weights and are applied on top of the common representations extracted by the video and audio encoders. The visual feature maps are average-pooled before being input to the classification network g^v , while f^v is applied directly on the spatial feature map, as discussed in Section 3.1 of the main paper. The output dimension for the localisation networks (i.e. the common embedding dimension) is 128, while the audio and video classifier heads each output K values, one for every cluster.

Detector backbone. As the backbone for the Faster R-CNN detector we use a ResNet50(1x) [22, 41] model pre-trained without labels using the SimCLR [22] method on ImageNet. We use the pretrained weights¹ released by the authors of [22]. We apply a Feature Pyramid Network (FPN) [60] on top of the activations of different backbone layers extracting visual features at five spatial scales, i.e. $/4, /8, /16, /32$, and $/64$ of the input resolution. Feature descriptors are extracted for every proposal with ROI-align pooling [40].

Detector RPN training. The box proposals $m \in M(v)$ used by the Faster R-CNN detector are produced by a Region Proposal Network (RPN). The RPN extracts a set of anchors $n \in N(v) \subset \Omega^2$ at three aspect ratios (0.5, 1.0 and 1.5) on each FPN output scale. The extracted number of anchors depends only on the input image dimensions.

The RPN contains 2 MLP heads in order to model $o(n) = f_{\text{rpn}}^{\text{obj}}(n|v) \in \{0, 1\}$, a binary objectness label indicating whether anchor n contains an object or background, and $m(n) = f_{\text{rpn}}^{\text{loc}}(n|v) \in \mathbb{R}^4$, the bounding box proposal resulting from anchor n .

The RPN heads are trained jointly with the detector heads, using the input box annotations and ignoring the class labels: if $n^* = \arg \max_{n \in N(v)} \text{IoU}(n, t^*)$ is the anchor that matches the pseudo-ground truth bounding box t^* the best, one optimizes:

$$\begin{aligned} \mathcal{L}_{\text{rpn}}(v, t^*) = & \mathcal{L}_{\text{reg}}(m(n^*), t^*) + \mathcal{L}_{\text{obj}}(o(n^*), 1) \\ & + \sum_{a \in A(v): \text{IoU}(n, t^*) < \tau} \mathcal{L}_{\text{obj}}(o(n), 0). \end{aligned}$$

Here $\mathcal{L}_{\text{reg}}$ is the L^1 loss for the bounding box corner coordinates and $\mathcal{L}_{\text{obj}}$ a binary cross-entropy loss. Similarly to the detector loss, the RPN loss requires the best anchor n^* to match bounding box t^* and to have a high objectness score, while reducing the objectness scores of anchors n that are a bad match.

Training details. For all the experiments we pre-train the video and audio backbones using the localisation objective only and average-pooling instead of max-pooling on the heatmap. The pre-training is performed on the Audioset-Instruments training set for 230 epochs on 64 GPUs with a

¹<https://github.com/google-research/simclr>

VGGSound label (representative)	OpenImages label (test)
airplane flyby	Airplane
ambulance siren	Ambulance
mynah bird singing	Bird
racing car	Car
cat purring	Cat
typing on computer keyboard	Computer Keyboard
mouse clicking	Computer Mouse
frog croaking	Frog
helicopter	Helicopter
lions roaring	Lion
driving snowmobile	Snowmobile

Table A.1: Mapping of VGGsound to OpenImages labels for qualitative evaluation on non-instrument object categories shown in Figure A.1.

batch size of 12 per GPU, using the Adam optimizer with a constant learning rate of 6.4×10^{-4} and standard hyperparameters. The learning rate is initially set to 1×10^{-5} and warmed up gradually until reaching the constant value after 10 epochs. The joint localisation and clustering training is performed on 16 GPUs with a batch size of 16 per GPU and SGD optimization with a constant learning rate of 0.005 and momentum 0.9, for a further 300 epochs. We set hyperparameter λ to 0.5. The temperature for the contrastive learning is learned as a single scalar weight. The detector training is performed on 16 GPUs with a batch size of 12, SGD optimization and learning rate 0.008 for 100 epochs. The total training takes approximately 3 days in this setting.

C. Additional label mapping details

In Table A.2 we show the full list of VGGSound video categories used for training our models. We use only samples from the 39 *mappable* classes for the detector training stage if not stated otherwise.

The 15 *test* labels shown are the musical instrument labels found in the OpenImages dataset. As explained in the main paper, the Audioset and VGGSound test sets that we use for evaluation have been pseudo-labelled with high confidence detections from a supervised detector trained on OpenImages by [30], therefore these 15 test labels are common among all our test sets.

We use the same colour to show the manual grouping of VGGSound labels that we use for some of the baseline and ablation experiments. The many-to-one grouping of the *mappable* into *representative* labels is used for training the PCL WSOD baseline and also for evaluating the “manual” matching strategy in Table 6 of the main paper. We note that our proposed method **does not** use this mapping, but instead

uses the Hungarian algorithm for matching our learned clusters directly to the *representative* classes. On the same table we also show the constant mapping of *representative* VGGSound labels to the OpenImages test labels that is necessary for evaluation. The majority of the instrument categories are matched exactly, except for the ‘Guitar’ and ‘Drum’ test labels for which we had to choose one of the candidates from the VGGSound guitar and drum subsets; we chose “playing electric guitar” and “playing snare drum” as the representative labels respectively. Note that since the Hungarian matching only matches one cluster to every test label, only choosing one *representative* class, harms our detectors, which are in fact more specialised. For example instances of acoustic guitar that may be correctly detected and not labelled as electric guitar will not be taken into account, in effect lowering our method’s measured performance.

Similarly in Table A.1 we show the *representative* to *test* mapping used for evaluation and visualisation in the non-instrument object detection experiment.

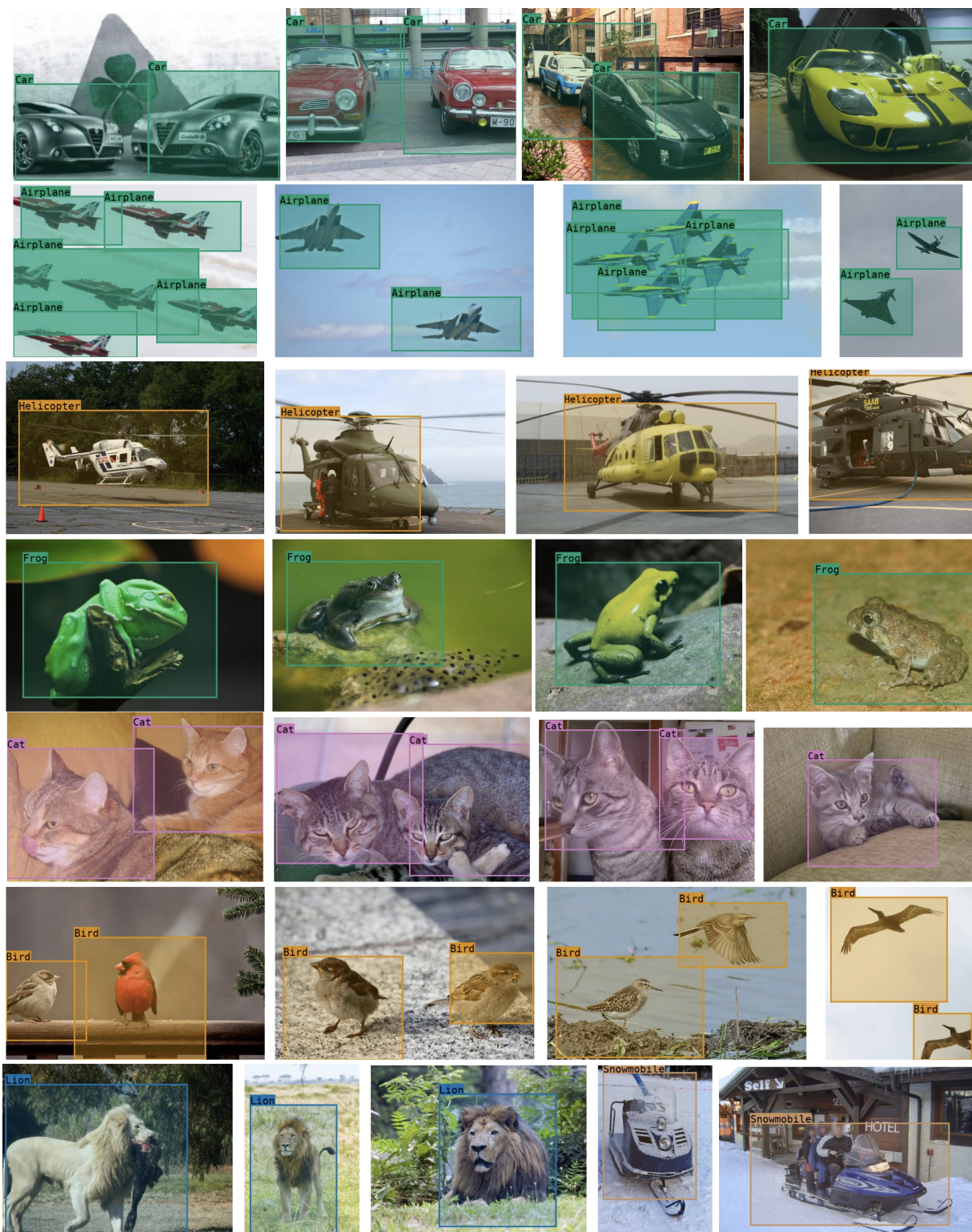


Figure A.1: **Object detection beyond musical instruments.** Our proposed method can learn to accurately detect objects from more general categories, as long as they can be associated with a characteristic sound. The results shown here are from a model trained without labels directly on the full VGGSound dataset which includes 309 different video classes.

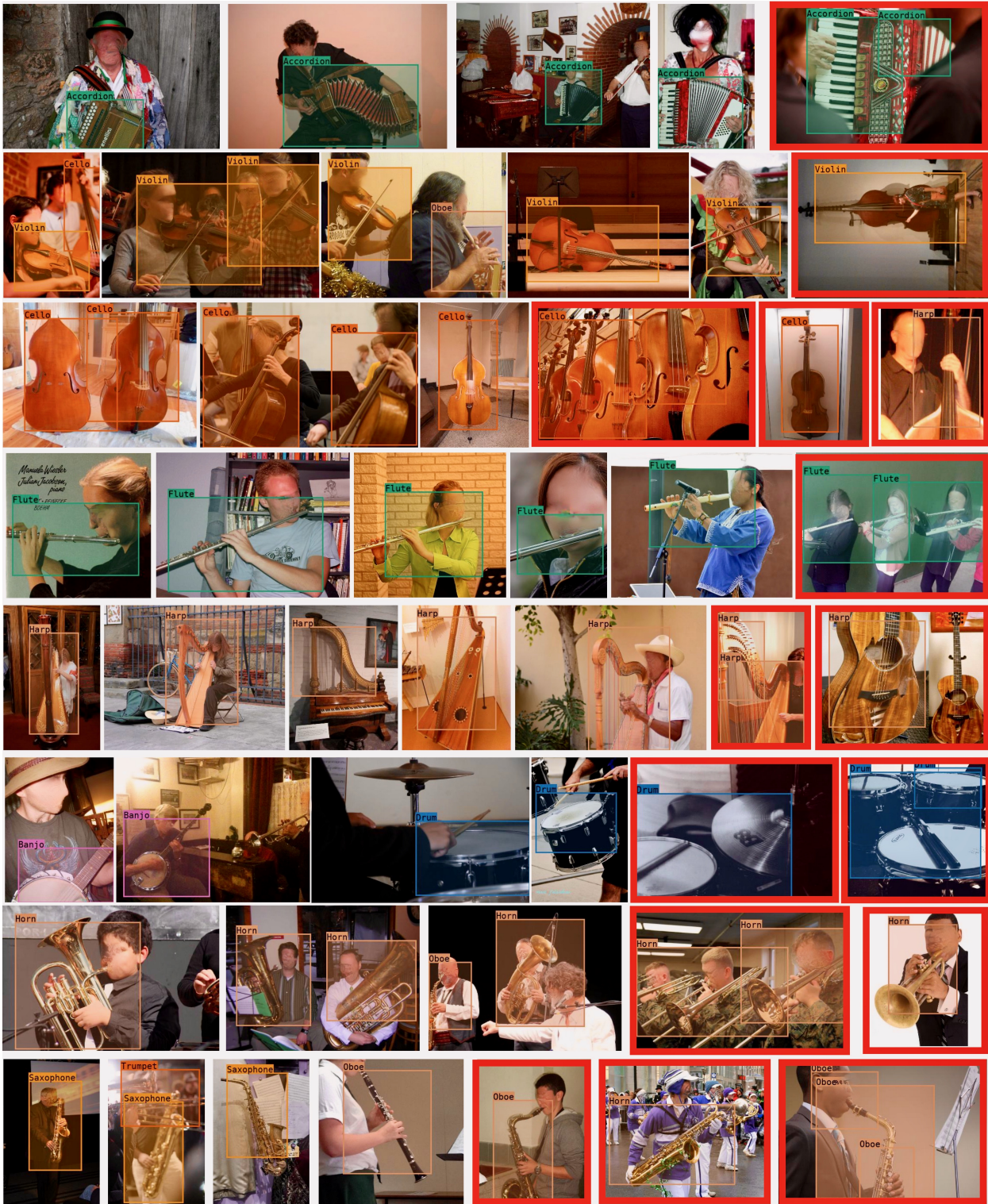


Figure A.2: Additional qualitative examples and failure cases. The images highlighted in red contain failure cases.

VGGS instrument classes (50)	mappable (39)	representative (15)	test (15)
playing accordion	playing accordion	playing accordion	Accordion
playing double bass	playing double bass	playing cello	Cello
playing bassoon	playing bassoon	playing flute	Flute
playing cello	playing cello	playing electric guitar	Guitar
playing clarinet	playing clarinet	playing harp	Harp
playing cornet	playing cornet	playing oboe	Oboe
playing flute	playing flute	playing piano	Piano
playing acoustic guitar	playing acoustic guitar	playing saxophone	Saxophone
playing bass guitar	playing bass guitar	playing trombone	Trombone
playing electric guitar	playing electric guitar	playing trumpet	Trumpet
tapping guitar	tapping guitar	playing violin	Violin
playing harp	playing harp	playing banjo	Banjo
playing harpsichord	playing harpsichord	playing harmonica	Harmonica
playing oboe	playing oboe	playing french horn	Horn
playing electronic organ	playing electronic organ	playing snare drum	Drum
playing hammond organ	playing hammond organ		
playing piano	playing piano		
playing saxophone	playing saxophone		
playing trombone	playing trombone		
playing trumpet	playing trumpet		
playing ukulele	playing ukulele		
playing violin	playing violin		
playing mandolin	playing mandolin		
playing banjo	playing banjo		
playing harmonica	playing harmonica		
playing french horn	playing french horn		
playing drum kit	playing drum kit		
playing bass drum	playing bass drum		
playing snare drum	playing snare drum		
playing tabla	playing tabla		
playing bongo	playing bongo		
playing tambourine	playing tambourine		
playing timpani	playing timpani		
playing tympani	playing tympani		
playing timbales	playing timbales		
playing congas	playing congas		
playing djembe	playing djembe		
playing cymbal	playing cymbal		
playing gong	playing gong		
playing didgeridoo			
playing bugle			
playing shofar			
playing glockenspiel			
playing vibraphone			
playing marimba			
playing bagpipes			
playing theremin			
playing steel guitar			
playing sitar			
playing erhu			

Table A.2: **VGGSound instrument categories and labels mapping.** The 15 *test* labels are the OpenImages instrument labels that all our tests sets are labelled with. The 39 *mappable* labels comprise the VGGSound subset that can be roughly mapped to those 15 test labels – this mapping is necessary for evaluation. The *representative* classes are the ones that we use to link VGGSound labels with the 15 test labels for evaluation. Labels manually matched by us are shown with the same colour.

10

PASS: An ImageNet replacement for self-supervised pretraining without humans

This work will be presented at the conference on Neural Processing Systems (NeurIPS) 2021 Track on Datasets and Benchmarks.

PASS: An ImageNet replacement for self-supervised pretraining without humans

Yuki M. Asano

Christian Rupprecht

Andrew Zisserman

Andrea Vedaldi

Visual Geometry Group, University of Oxford
{yuki, chrisr, az, vedaldi}@robots.ox.ac.uk

Abstract

Computer vision has long relied on ImageNet and other large datasets of images sampled from the Internet for pretraining models. However, these datasets have ethical and technical shortcomings, such as containing personal information taken without consent, unclear license usage, biases, and, in some cases, even problematic image content. On the other hand, state-of-the-art pretraining is nowadays obtained with unsupervised methods, meaning that labelled datasets such as ImageNet may not be necessary, or perhaps not even optimal, for model pretraining. We thus propose an unlabelled dataset **PASS: Pictures without huMans for Self-Supervision**. PASS only contains images with CC-BY license and complete attribution metadata, addressing the copyright issue. Most importantly, it contains *no images of people at all*, and also avoids other types of images that are problematic for data protection or ethics. We show that PASS can be used for pretraining with methods such as MoCo-v2, SwAV and DINO. In the transfer learning setting, it yields similar downstream performances to ImageNet pretraining even on tasks that involve humans, such as human pose estimation. PASS does not make existing datasets obsolete, as for instance it is insufficient for benchmarking. However, it shows that model pretraining is often possible while using safer data, and it also provides the basis for a more robust evaluation of pretraining methods. Dataset and pretrained models are available¹.

1 Introduction

The development of modern machine learning could not have happened without the availability of increasingly large and diverse research datasets. Consider computer vision for example: algorithms were initially developed using small datasets collected in laboratory conditions and, as a consequence, almost no method worked in the real world. This cycle was broken only once researchers adopted datasets such as PASCAL VOC, MS COCO and ImageNet, which are collections of images sampled from the Internet. These collections are not only much larger than prior datasets, but they also provide a much better representation of the statistics of natural images because they arise from the real world. Because of this, algorithms became vastly more robust and were able to better generalize to the real world. Furthermore, these datasets have acquired fundamental scientific functions as well: they allow reproducible and quantitative comparison of algorithms and they enable researchers to efficiently build on each other's work.

However, for all their benefits, these datasets have technical, ethical and legal shortcomings.

One issue is *copyright*. While some datasets contain images explicitly licensed for reuse (e.g. Creative Commons), some do not. Some national laws contain copyright exceptions or otherwise allow the

¹<https://www.robots.ox.ac.uk/vgg/research/pass/>

We also focus on data made available under the most permissive Creative Common license (CC-BY) to address copyright concerns.

Given this data, we then conduct an extensive evaluation of SSL methods, discussing performance differences when these are trained using ImageNet and **PASS**. Compared to ImageNet, there are three essential differences with the **PASS** dataset: the lack of class-level curation and search; the lack of ‘community optimization’ on this dataset; and, of course, the lack of humans. We study via further ablation the contribution of these effects.

We find that: (i) self-supervised approaches such as MoCo, SwAV and DINO train well on our dataset, yielding strong image representations; (ii) excluding images with humans during pretraining has almost no effect on downstream task performances, even if this is done in ImageNet; (iii) in 8/13 frozen encoder evaluation benchmarks, performance of models trained on **PASS** yield better results than pretraining on ImageNet, ImageNet without humans, or Places205, when transferred to other datasets; and for finetuning evaluations, such as detection and segmentation, **PASS** pretraining yields results within $\pm 1\%$ mAP and AP50 on COCO. (iv) Even on tasks involving humans, such as dense pose prediction, pretraining on our dataset yields performance on par with ImageNet pretraining.

2 Related Work

Image datasets for model pretraining. By far the most widely used pretraining dataset is ImageNet ILSVRC12 (IN-1k) [22], containing 1.28M images covering 1000 object categories. This is followed by MIT-Places 205 and 365 datasets [97, 99], containing 2M and upto 10M images with labels, respectively, Webvision [59] containing the same labels as IN-1k, as well as Taskonomy [92], containing 4M images with scene attributes.

YFCC-100M [81] is a much larger dataset with 99M images with licence information and other metadata. OpenImages [6, 54, 56] contains 6M images and 20K classes. These datasets have been further combined and relabeled to yield further datasets, including Tencent ML-images [86] and Conceptual Captions [76].

When measured on downstream task performance, supervised pretraining on ImageNet (IN-1k) has nowadays been surpassed by unsupervised pretraining, but often these methods use the same images for both pretraining and evaluation (with labels). We argue that SSL methods should instead be trained on more diverse and less-curated datasets, so as to disentangle the performance gains from the potential requirements on the underlying data. This is particularly true if ImageNet is *also* used for evaluation.

Bias and privacy. Biases contained in datasets have been investigated in various works [8, 9, 20, 25, 42, 46, 68, 78, 90, 94] and in particular its annotation practices [1, 7, 34, 48, 63, 65, 71, 82, 83]. We focus specifically on ImageNet as this is the defacto standard dataset used for pretraining representations³. Notably, [20] shows that the larger version of ImageNet had stereotypes/slurs as class labels, and further was biased with regards to gender-biased depictions [25], which led to the removal of the person categories from the dataset [90].

More recently, Birhane and Prabhu [8] finds biased and NSFW images in ImageNet LSVRC-12 and criticises its labelling process. This paper and its subsequent blog-post [70] also outline problematic categories, such as the ‘bikini’ class. Another issue is that ImageNet, as well as other datasets, contain biases inherited from the search engines used to collect them, which sometimes results in racist, sexualized, skewed or stereotyped representations [42]. These biases are passed down to models trained on this data, e.g. as shown in the image completion work [78], or face upsampling [62]. Furthermore, the labels included in ImageNet are also imperfect: when [71] reproduced ImageNet’s annotation process, they noted highly-variable results yielding up to 11%–14% drops in accuracy. For a more in-depth review of datasets, their collection practises and current issues we refer to Paullada et al. [68].

In response to some of these issues, a version of ImageNet with blurred faces has now been published [91], in which the authors show that this only marginally affects object classification perfor-

³Although these issues are likely to be present in most other web images datasets too: following the methodology of [8], we easily find several pornographic and troubling images even in Places205 [98]. Better filtering for such content is necessary.

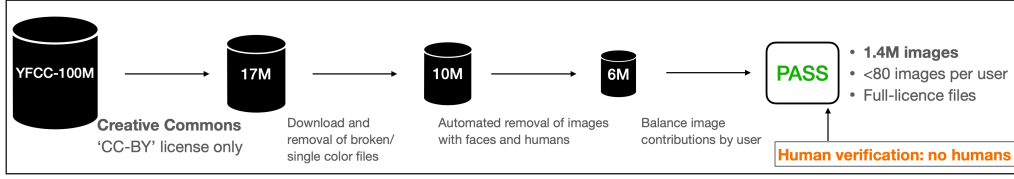


Figure 2: Dataset generation pipeline. Starting from 100M images in YFCC, we select 6M images that do not contain humans. This dataset is further reduced to the size of 1.4M by balancing user contributions and is finally manually checked.

mance. However, as [70] points out, this does not eliminate the problem, as even after blurring faces, reverse image search can be used to retrieve the original picture, thus leaving the privacy concerns unsolved.

In this work, we focus on alleviating some of these privacy and bias issues by constructing a dataset that is completely free of humans, does not rely on search engines or labels, and only contains images with a permissive license (CC-BY) and attribution information.

Self-supervised learning. Starting from early works such as [47, 67], self-supervised representation learning methods are nowadays mainly based on clustering [3, 11, 35, 36, 58] and/or noise-contrastive instance discrimination [14, 29, 38, 41, 44, 64, 88]. In particular, MoCo [44] has sparked several further implementations [4, 15, 51, 58, 77, 89, 100] and therefore we use this model for most of the experiments in this paper. Several papers [30, 53, 95] also compare SSL methods in detail and find among other results that, downstream task performance is dependent on the pretraining dataset, and that self-supervised methods struggle on fine-grained classification tasks compared to their supervised counterparts. We therefore evaluate various distributional splits of our dataset and evaluate also on finegrained datasets.

3 PASS: images minus people

The PASS dataset is obtained as a subset of the YFCC-100M dataset [81]. The latter is freely available on the AWS public datasets page [74], and has complete metadata that includes the individual licenses and creators, thus satisfying our requirement of CC-BY. In case this dataset ceases to exist on the public datasets of AWS, we have made necessary backups, for details see the ‘Datasheets of Datasets’ section in the Appendix.

Collection modality. PASS was obtained as follows (see also Fig. 2). First, the YFCC-100M metadata was used to select images with CC-BY licence, which left us with 17M images. Once, these images were downloaded, corrupted or single-color images⁴ were removed (leaving 10M). Pretrained RetinaFace [23] and Cascade-RCNN (3x) [10] models were used to filter out images containing human faces and humans (6M). All details are provided in the Appendix.

In YFCC-100M, the distribution of images per photographer is highly skewed. Thus, to increase dataset diversity, we set a maximum threshold of 80 images per photographer and uniformly sampled 1.4M images. Finally, these images were submitted for human labelling⁵.

Manual Filtering. The annotators were asked to identify images that contain people or body parts, as well as personal information such as IDs, names, license plates, signatures etc. Additionally, the annotators were asked to flag images with problematic content such as drugs, nudity (mostly included

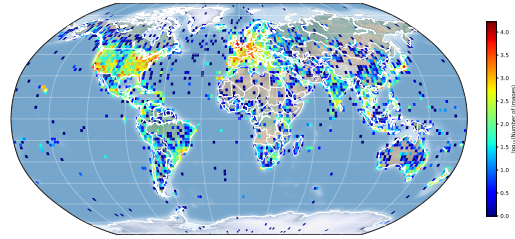
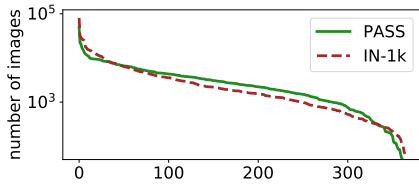


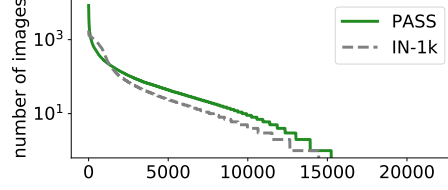
Figure 3: Geographic diversity: 552K of 1.4M images contain geo-locations.

⁴For a surprisingly large fraction of the images all pixels have the same color.

⁵Prior to human verification we further checked the most highly ranked 1K images according to a NSFW classifier [57] for possible harmful content and found none.



(a) **Distribution of places.** As determined by a Places-365 pretrained ResNet-50.



(b) **Distribution of objects.** As determined by a ImageNet-21k pretrained ResNet-50x3 classifier.

Figure 4: **Larger diversity than ImageNet-1k.** Compared to the curated ImageNet-1k dataset, our dataset has longer tails.

Table 1: Downstream evaluation tasks and datasets. (⌘) Indicates presence of ‘person’ annotations or human-oriented downstream tasks.

Frozen	Clustering: IN-1k, ObjectNet, Places205, Flowers
	Linear probing: IN-1k, ObjectNet, Places205, Flowers, CIFAR-10 (⌘), Herbarium2019
	Low-shot classification: Pascal VOC07 (⌘), Places205, Flowers
Finetune	Object detection: COCO17 (⌘), Pascal VOC07+12 (⌘)
	Object segmentation: COCO17 (⌘), LVIS (⌘)
	Dense pose detection (⌘): COCO17
	Human keypoint detection (⌘): COCO17

in the “no people” rule), blood and other offensive content. The annotations were performed over the course of three weeks by an annotation company whose annotators were paid 150% of the minimum wage. During the manual annotation process, around 2% of the images were flagged and subsequently removed. From the remaining images (1.46M) we further removed duplicates (see Appendix C.3) and randomly selected a subset with approximately the same size as IN-1k (1.440.191 images).

Statistics. We first provide some descriptive results of our dataset. Using the meta-data provided, we plot the GPS location of the images in Fig. 3. We find that the subset of the dataset that contains GPS coordinates covers a wide area of the world, but is focused on western countries such as the US, Europe and Japan. This likely reflects the skew of the Flickr user database and is a form of bias that can limit generalisation capabilities [21], as for example images taken in less developed countries might be uploaded by tourists that take images of stereotypes [75, 84]. With this we wish to stress that the **PASS** dataset is by no means bias-free, and its misunderstanding as such might lead to spurious claims of fairness. While the dataset comes with some textual annotations such as tags and descriptions, these are discarded to avoid further biases [83] since they are not necessary for self-supervised learning.

In Fig. 4 we show the results of running pretrained classification models on our dataset and compare the output diversity against IN-1k. Note that in Fig. 4b, the distribution for IN-1k objects should technically be zero for classes after 1000, as IN-1k is a subset of IN-21k. We attribute this deviation to the quality of the model—classifiers with a very high number of classes (here: 21,000) are still very noisy. Nonetheless, for a relative comparison of two datasets, even a noisy classifier is sufficient. We find that, compared to IN-1k, our dataset contains more variety in terms of places and objects, as shown by the higher tail of the distributions.

4 Experimental setup

We evaluate **PASS** by pretraining models on the **PASS** data using self-supervision and then testing the quality of the resulting models on a range of downstream tasks outlined in Table 1.

SSL pretraining methods. We pretrain models using MoCo-v2 [16] as it has become a staple SSL method, can be extended in various ways [4, 15, 51, 58, 77, 89, 100] and, with updates to the learning schedule and most recently its implementation of the loss [17] remains state-of-the-art in feature learning. This makes it an ideal candidate to probe the pretraining performance of our dataset. We therefore conduct most of our experiments using the MoCo-v2 public implementation with a 200

epoch pretraining schedule. We further train and evaluate models trained with SwAV [12], another state-of-the-art representation learning method, but one which is based on unsupervised clustering. Finally, we also pretrain using DINO [13], a recent method that is tailored to vision transformer [24] SSL.

For downstream evaluation, we set all training parameters as proposed in existing literature (which, in practice, means they are tuned for IN-1k). Specific tuning for each pretraining dataset would probably improve performance. For pretraining MoCo-v2 and MoCo-v2 CLD, we follow the official implementation and adhere to the 200 epoch schedule. We use the 200 epoch configuration with multi-crop schedule for SwAV which takes around double the GPU hours as MoCo’s 200 epochs. For DINO, we follow their 100 epoch training code, which also includes multi-crop. Further experimental details are given in the Appendix.

Pretraining Data. Using MoCo-v2, we compare **PASS** against other datasets for pretraining, such as: (a) IN-1k: ImageNet-2012 split [22], the current standard for SSL pretraining; (b) IN-1k*: ImageNet-2012 with humans and human parts removed, as an ablation of IN-1k; (c) Places-205 [98]: to compare the effect of the data distribution on pretraining, as in [53, 95]. We construct (b) by first removing all images with faces [91] and subsequently running our automated human and human part detection pipeline. The selected image IDs are provided in the supplementary material. We do not include manual verification because this dataset is only meant to measure the effect of removing humans from ImageNet. We also compare against supervised pretraining using datasets (a) and (b) for which labels are available.

Frozen encoder evaluation. Frozen encoder evaluations are seen as a gold-standard for evaluating representation learning performance due to their simplicity and ease of use. We include: the two standard evaluation datasets ImageNet [22] and Places [97], as well as ObjectNet [5], CIFAR-10 [55] as an example of a low-resolution dataset (though we note that its classes are contained within IN-1k), and the fine-grained classification datasets Oxford Flowers [66] and Herbarium 2019 [80]. We use linear probing, low-shot classification and clustering as tasks to evaluate the frozen encoders. For linear probing, we use MoCo-v2’s default linear evaluation code, without any adjustments⁶. For low-shot classification we follow the public implementation of [58], again without tuning hyperparameters. For clustering, K -means is run on the embedded features of the validation set and their clustering is compared against the ground-truth labels, as in [73, 96]. Here K is simply set to be equal to the number of ground-truth classes. This evaluation does not contain any hyperparameters.

Finetuning evaluation. For finetuning, we follow standard evaluation practices of the self-supervised literature: we evaluate object segmentation and detection. Beyond this, we also include human-focused efficient keypoint detection and pose estimation on COCO [60] to measure the effect of removing images with humans on person-centric tasks. Finally, we also finetune for object detection on PASCAL VOC [32] and long-tailed object segmentation on LVIS-v1 [40].

Specific details such as number of epoch, learning rates and schedules are provided in the Appendix. Additionally we include all code to reproduce the evaluations.

5 Results

When comparing pretraining datasets, a key factor is whether the pretraining and target datasets used for evaluation coincide or not. The realistic usage scenario is that they differ, which amounts to transfer learning. Yet, for benchmarking SSL methods, the non-transfer setting is often used instead. For example, both pretraining *and* evaluating on IN-1k does not test whether the features generalize beyond ImageNet. In the tables, we clearly distinguish these two cases by *graying the non-transfer learning results*. Naturally, there is a performance gap when training on **PASS** compared to matching pretraining and target datasets. However, for transfer learning we show that **PASS** performs on par to pretraining on ImageNet or similar datasets, even on downstream tasks involving humans, and for different SSL methods. Furthermore, we ablate the effect of using different pretraining splits and pretraining augmentations on downstream performances.

Frozen encoder evaluation. For the clustering evaluation in Table 2a, pretraining on **PASS** yields generalisable features that can outperform IN-1k and IN-1k* pretraining in the transfer learning

⁶For ObjectNet, which is a test set only, we transfer the linear classifier trained on IN-1k and report performance on the overlapping classes.

Table 2: **Frozen encoder evaluations.** We evaluate the encoders by (a) unsupervised clustering, (b) training a linear layer, and (c) training SVMs in a low-shot scenario on top of the frozen representations. Gray numbers indicate “non-transfer” evaluations on datasets that share all classes with the pretraining dataset and thus are expected to perform better than datasets with a different label distribution.

	IN-1k		ObjN		Places		Flowers	
Pretraining	NMI	aRI	NMI	aRI	NMI	aRI	NMI	aRI
Random	43.4	0.1	17.6	0.1	20.7	0.2	24.9	1.7
Supv. IN-1k	79.7	40.7	36.7	6.2	51.5	14.3	70.3	38.8
Supv. IN-1k*	75.9	34.8	33.1	4.3	47.0	10.7	70.4	39.0
MoCo-v2								
on IN-1k	64.0	12.7	24.3	1.1	44.2	7.5	62.1	30.1
on IN-1k*	63.2	12.2	23.7	1.0	42.4	6.4	62.5	32.0
on Places	54.7	5.5	22.8	0.8	51.1	12.0	55.2	21.6
on PASS	55.6	5.9	22.9	0.8	44.4	7.7	63.0	30.1

	IN-1k → ObjN		Plcs		C10 Flwrs		H19	
Pretraining								
Random	4.2	0.4	16.6	14.4	9.0	3.1		
Supv. IN-1k	76.3	27.3	51.5	74.0	90.7	44.0		
Supv. IN-1k*	69.1	23.2	40.0	72.3	89.9	43.8		
MoCo-v2								
on IN-1k	67.5	16.0	50.1	91.6	90.2	42.3		
on IN-1k*	65.6	14.6	50.6	90.7	90.3	42.5		
on Places	57.3	9.9	56.6	55.7	85.9	36.8		
on PASS	59.5	11.7	52.8	81.0	89.2	46.5		

(a) **Clustering evaluation:** K-means is run on the validation sets and the assignments are compared against the ground-truth labels via the NMI and aRI (%).

(b) **Linear probing:** representations are used to train a linear layer using a standard schedule on a target dataset and top-1 accuracy is reported.

Pretraining	k=	Places					Pascal VOC07					Herbarium19				
		1	2	4	8	16	1	2	4	8	16	1	2	4	8	16
Random		0.8	1.1	1.3	1.6	2.1	8.3	8.6	9.4	9.9	10.0	0.4	0.4	0.6	0.6	0.9
Supervised IN-1k		15.3	21.3	27.0	32.3	36.4	54.0	67.9	73.8	79.7	82.3	4.8	7.4	11.5	16.7	22.6
Supervised IN-1k*		12.7	17.5	22.9	27.9	32.2	48.6	62.1	68.7	75.2	78.8	4.8	7.8	11.8	17.2	23.2
MoCo-v2																
on IN-1k		12.2	16.8	22.3	28.1	32.8	46.3	58.3	64.9	72.5	76.1	4.7	7.7	12.4	18.2	24.4
on IN-1k*		11.0	15.4	20.4	25.9	30.6	43.2	55.3	61.9	69.5	73.8	4.8	7.5	12.0	17.7	24.4
on Places		21.5	27.8	34.0	39.6	43.4	40.6	51.5	57.5	64.6	69.0	3.9	6.4	9.8	14.9	20.6
on PASS		12.0	16.6	22.1	27.7	32.8	40.8	51.6	57.6	65.8	70.3	5.3	8.4	12.9	18.7	25.0

(c) **Low-shot classification** for k=1,2,4,8,16 images per class with training one-vs-all SVMs using 5 runs each. We report top-1 accuracy except for the multi-label VOC07, where we report mAP.

Table 3: **Finetuning representation evaluations.** We finetune the encoders on (a) COCO2017 object detection & segmentation (see Appendix for Pascal VOC) and (b) dense human pose estimation. AP is COCO’s AP on IoUs [0.5:0.95:0.05].

Bounding-box					Segmentation			
Init.	AP	AP ₅₀	AP ₇₅	%-AP	AP	AP ₅₀	AP ₇₅	%-AP
Random	26.4	44.0	27.8		29.3	46.9	30.8	
Supv. IN-1k	38.9	59.6	42.7	54.9	35.4	56.5	38.1	47.1
Supv. IN-1k*	38.5	59.3	41.9	54.6	35.0	56.0	37.1	46.7
MoCo-v2								
on IN-1k	38.7	59.2	42.3	55.5	35.2	56.2	37.9	47.6
on IN-1k*	38.4	58.7	41.8	55.4	35.0	55.8	37.4	47.2
on Places	38.3	58.4	41.7	55.9	34.8	55.4	37.4	47.8
on PASS	38.0	58.5	41.5	55.5	34.7	55.4	37.1	47.5

Dense-pose			B-box	Seg.
Init.	AP ^{dp} _{GPS}	AP ^{dp} _{GPSm}	AP ^{bb}	AP ^{sg}
Random	—does not train—			
Supv. IN-1k	64.4	65.7	61.1	67.1
Supv. IN-1k*	64.2	65.5	60.9	66.9
MoCo-v2				
on IN-1k	65.0	66.3	61.7	67.6
on IN-1k*	64.8	66.1	61.4	67.0
on Places	64.9	66.0	61.8	67.1
on PASS	64.9	65.7	61.5	66.8

(a) Object detection & segmentation on COCO using R50-FPN, 1x schedule. %-AP denotes AP on the person class.

(b) Dense human pose estimation with R50-FPN and the ‘s1x’ schedule.

setting. The linear probing benchmark in Table 2b shows a similar pattern: **PASS** yields the best transfer performance on IN-1k, ObjectNet, Places205 and Herbarium19 (HT19). However, for CIFAR-10 transfer from ImageNet works better than Places and **PASS**. This is most likely due to the semantic overlap of these two datasets: all CIFAR-10 classes are contained in ImageNet, reducing the transfer gap.

Finally, we report the low-shot classification results in Table 2c. Here, we find that **PASS** pretraining performs close to IN-1k and surpasses IN-1k* when transferring to Places205. For PASCAL VOC, **PASS** is worse than both ImageNet variants, but outperforms Places205 pretraining. The gap to

Table 4: **Effect of SSL method.** We pretrain with recent SSL methods and evaluate: Linear probing and 4-shot classification (top-1 accuracy), PascalVOC (mAP), object detection COCO (mAP, $\frac{1}{2}$ AP) and for PascalVOC (AP50). For clustering we report aRI.

SSL method	Linear probing					Object detection		4-shot classific.			Clustering	
	IN-1k	ObjN	Plcs	Flwrs	HT19	COCO (m, $\frac{1}{2}$)	PVOC	PVOC07	Plcs	H19	Plcs	Flwrs
MoCo-v2	59.5	11.7	52.8	89.2	46.5	38.0, 55.5	81.1	57.6	22.1	12.9	7.7	30.1
MoCo-v2-CLD	60.2	12.7	53.1	89.8	44.7	38.1, 54.7	81.2	59.6	23.0	13.9	7.9	31.4
SwAV	60.8	12.0	55.5	91.9	54.4	37.2, 52.3	72.9	61.9	23.0	16.0	9.4	36.7
DINO (ViT-S16)	61.3	11.9	54.6	92.2	48.5	NA	NA	64.3	24.9	18.2	10.8	43.1

Table 5: **Effect of pretraining augmentation.** Here we compare the effect of the minimum size parameter of the random-resized crop (RRC) augmentation when pretraining MoCo-v2.

RRC min. size	Linear probing					Object detection		4-shot classific.			Clustering	
	IN-1k	ObjN	Plcs	Flwrs	HT19	COCO (m, $\frac{1}{2}$)	PVOC	PVOC07	Plcs	H19	Plcs	Flwrs
0.2 (default)	59.5	11.7	52.8	89.2	46.5	38.0, 55.5	81.1	57.6	22.1	12.9	7.7	30.1
0.1 (more zoom)	59.5	11.7	53.4	89.6	48.9	37.8, 55.3	81.1	57.6	22.5	13.5	7.7	29.3

IN-1k pretraining might be due to the insufficient tuning of the SVM cost hyperparameter, as well as the semantic similarity of PascalVOC with ImageNet (common objects, object-centric images and indeed overlapping object categories). However, as shown below, different pretraining methods such as DINO can yield significant improvements. For Herbarium19, **PASS** yields the strongest pretraining performance, and is the only example where the self-supervised representation surpasses the supervised baselines for this benchmark.

Finetuning evaluation. Next we analyse the performance of the pretrained representations when they are used as initialisation for downstream tasks (c.f. Table 3). For Mask-RCNN based detection and segmentation on COCO (Table 3a, we find that the gaps between the different pretraining datasets are overall small and within $\pm 1\%$ for the AP, AP50 and AP75 measures for both bounding-boxes (bb) and mask (mk) evaluations. We also report the performance on the ‘person’ object category ($\frac{1}{2}$) and find that the encoder pretrained on **PASS** still matches and surpasses the ones obtained with supervision on IN-1k and IN-1k*. In Table 3b we find similar trend for dense pose estimation, an inherently human-focused task, where the differences between pretraining datasets are even smaller. We provide further experimental results in the Appendix.

Effect of data content. In this section we study the effect of selecting different types of visual content, using the splits specified in Table 6. The first question is *whether humans matter or not* for pretraining. In the Humans split, we proceed as in **PASS** but skip the filtering step (in this case, 57% of the images contain humans and there is a 27% overlap with the filtered version). This leads to a small improvement in object detection 4-shot classification and a small decrease in performance for HT19, Flowers, and ObjectNet, but mostly staying within $\pm 2\%$.

The second question is whether the distribution of *other* classes matters or not. We test biasing the selection of images by using an IN-1k pretrained classifier to match the class statistics of IN-1k as well as possible. The $\approx$ IN800 split is obtained from the 6M images by retaining the most frequent 800 classes and retaining at most 2K samples per class, followed by subsequent sampling to a size of 1.28M. The $\approx$ IN800 split leads to a large increase in performance for IN-1k and ObjectNet linear probing, echoing previous findings that the closeness of the pretraining distribution to the downstream task matters [53, 72]. However, it also leads to small decreases on datasets such as Places, H19 and Flowers.

The third question is whether capping the number of pictures per photographer matters. The Random6M split contains all 6M images left after cleaning and removing humans, as determined by pretrained models, but skipping the photographer balancing and manual cleanup step (the latter is for safety and only changes the dataset composition marginally). The Random split further restricts that to 1.28M. While the Random split generally performs less well than the other splits, Random6M achieves strong performances on many tasks. While this result is also partially due to the larger number of optimization steps (all methods were trained with 200 epochs), it might show how SSL methods can scale with more data.

Table 6: **Effect of pretraining data.** We report the same metrics as in 4.

Data	Linear probing					Object detection		4-shot classific.			Clustering	
	IN-1k	ObjN	Plcs	Flwrs	HT19	COCO (m, $\frac{r}{s}$)	PVOC	PVOC07	Plcs	H19	Plcs	Flwrs
PASS	59.5	11.7	52.8	89.2	46.5	38.0, 55.5	81.1	57.6	22.1	12.9	7.7	30.1
• Humans	59.7	11.2	53.5	87.7	45.3	38.2, 55.6	81.4	59.8	22.2	12.5	7.9	29.9
• $\approx$ IN800	62.4	13.7	51.5	90.3	47.4	38.3, 55.2	81.7	61.6	21.4	12.8	7.5	31.2
• Random	58.7	11.2	52.7	88.6	46.6	37.8, 55.0	81.1	57.8	21.9	12.3	7.6	30.3
• Random6M	61.3	12.9	54.7	90.1	44.9	37.9, 55.3	81.0	61.3	24.4	11.9	8.4	31.3

Overall, **PASS** yields overall good performance without either containing humans, or the need for a pretrained classifier and is of size comparable to IN-1k.

Effect of pretraining method. Finally, we pretrain other self-supervised learning methods on our dataset and show results in Table 4. First, we compare MoCo-v2 against MoCo-v2-CLD [85], and find improvements similar in scale to the ones reported in [85]. Next, from the results of SwAV [12] and BYOL [38], we find that methods that performed better on IN-1k also tend to perform better on our dataset, show that **PASS** can be used instead of IN-1k to compare models.

A note on augmentations. In this paper we have solely used the hyperparameters that come with the methods, and as such, have been optimised by the computer vision community for the last few years on IN-1k. Augmentations play an important role for self-supervised learning [2, 14, 16], and strongly influence the final performance. We cannot replicate several years of augmentation tuning for our **PASS** dataset but can expect that with time, better settings can be found. To support this claim, we show one example of such adaption in Table 5 that we were able to find. By changing the minimum-size parameter used in random-resized crops from 0.2 to 0.1, we observe an improvement in performance in almost every benchmark. While small, the improvement might be explained by the fact that our dataset is less object-centric than ImageNet, and so stronger crops can be used, as there is no need for the crop to cover an object.

6 Discussion

In the introduction, we have motivated our new dataset **PASS** from a technical, ethical and legal perspective. By using CC-BY images, we greatly reduce the risk of using images in a manner incompatible with copyright. By avoiding the usage of search engines and labels to form a dataset, we avoid introducing corresponding biases. By excluding all images that contain humans, as well as other identifiable information and NSFW images, we significantly reduce data protection and other ethics risks for the data subjects. We have shown that, despite these changes, we can effectively pretrain neural networks using this data. By conducting extensive downstream task evaluations, we have shown that pretrained networks obtained using self-supervised training on this dataset are competitive to ImageNet on transfer settings and even on downstream tasks that involve humans.

However, several limitations remain. First, while we put care in filtering the images, automatically and manually, some harmful content might have slipped through.

Second, sampling images randomly from an uncurated large collection removes specific biases such as search engine selection but not others, for example the geographic bias. Furthermore, we added one significant bias: there are no people in these pictures, despite the fact that a large fraction of all images in existence contain people. While this appears to be acceptable for model *pretraining*, **PASS** cannot be used to learn models of people, such as for pose recognition.

Thirdly, since **PASS** contains no labels, **PASS** (in contrast to ImageNet) cannot be used alone for training and benchmarking. For this, curated datasets remain necessary, which continue to carry many of the issues of privacy and copyright described in the paper.

Despite these limitations, we believe that **PASS** is an important step towards curating and improving our datasets to reduce ethical and legal risks for many tasks and applications and at the same time to challenge the SSL community with a new, more realistic training scenario of utilizing images not obtained from a labeled dataset.

Acknowledgements

We are thankful to Abhishek Dutta and Ashish Thandavan for their great support. We thank Rajan and his team of annotators at Elancer for their precise work. We are grateful for support from the AWS Machine Learning Research Awards (MLRA), EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems [EP/L015897/1], the Qualcomm Innovation Fellowship, a Royal Society Research Professorship, and the EPSRC Programme Grant VisualAI EP/T028572/1. C. R. is supported by Innovate UK (project 71653) on behalf of UK Research and Innovation (UKRI) and by the European Research Council (ERC) IDIU-638009.

References

- [1] Lora Aroyo and Chris Welty. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24, 2015.
- [2] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [3] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- [4] Kumar Ayush, Burak Uzcent, Chenlin Meng, Kumar Tanmay, Marshall Burke, David Lobell, and Stefano Ermon. Geography-aware self-supervised learning. *arXiv preprint arXiv:2011.09980*, 2020.
- [5] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32:9453–9463, 2019.
- [6] Rodrigo Benenson, Stefan Popov, and Vittorio Ferrari. Large-scale interactive object segmentation with human annotators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11700–11709, 2019.
- [7] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.
- [8] Abeba Birhane and Vinay Uday Prabhu. Large image datasets: A pyrrhic win for computer vision? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1537–1547, 2021.
- [9] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.
- [10] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.
- [11] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proc. ECCV*, 2018.
- [12] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Un-supervised learning of visual features by contrasting cluster assignments. *arXiv.cs, abs/2006.09882*, 2020.
- [13] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.
- [14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *arXiv.cs, abs/2002.05709*, 2020.
- [15] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [16] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020.

- [17] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv:2104.02057*, 2021.
- [18] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, Oct 2017. ISSN 1558-2256. doi: 10.1109/jproc.2017.2675998. URL <http://dx.doi.org/10.1109/JPROC.2017.2675998>.
- [19] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. URL <http://www.robots.ox.ac.uk/~vgg/data/dtd/>.
- [20] Kate Crawford and Trevor Paglen. Excavating ai: The politics of training sets for machine learning. <https://excavating.ai/>, 2019.
- [21] Terrance de Vries, Ishan Misra, Changhan Wang, and Laurens van der Maaten. Does object recognition work for everyone? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 52–59, 2019.
- [22] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. ImageNet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009.
- [23] Jiankang Deng, Jia Guo, Zhou Yuxiang, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. In *arxiv*, 2019.
- [24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [25] Chris Dulhanty and Alexander Wong. Auditing imagenet: Towards a model-driven framework for annotating demographic attributes of large-scale image datasets, 2019.
- [26] A. Dutta and A. Zissermann. Find identical image (FII). <http://www.robots.ox.ac.uk/vgg/software/fii/>, 2021. Version: 1.01, Accessed: 16.06.2021.
- [27] A. Dutta, A. Gupta, and A. Zissermann. VGG image annotator (VIA). <http://www.robots.ox.ac.uk/vgg/software/via/>, 2016.
- [28] Abhishek Dutta and Andrew Zisserman. The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia, MM ’19*, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6889-6/19/10. doi: 10.1145/3343031.3350535. URL <https://doi.org/10.1145/3343031.3350535>.
- [29] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *arXiv:2104.14548*, 2021.
- [30] Linus Ericsson, Henry Gouk, and Timothy M. Hospedales. How Well Do Self-Supervised Models Transfer? In *CVPR*, 2021.
- [31] European Organization For Nuclear Research and OpenAIRE. Zenodo, 2013. URL <https://www.zenodo.org/>.
- [32] M. Everingham, L. van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2011) Results. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>, 2012.
- [33] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III au2, and Kate Crawford. Datasheets for datasets, 2020.
- [34] Bhavya Ghai, Q. Vera Liao, Yunfeng Zhang, and Klaus Mueller. Measuring social biases of crowd workers using counterfactual queries, 2020.
- [35] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6928–6938, 2020.
- [36] Spyros Gidaris, Andrei Bursuc, Gilles Puy, Nikos Komodakis, Matthieu Cord, and Patrick Pérez. Online bag-of-visual-words generation for unsupervised representation learning. *arXiv:2012.11552*, 2020.

- [37] Priya Goyal, Quentin Duval, Jeremy Reizenstein, Matthew Leavitt, Min Xu, Benjamin Lefaudeaux, Mannat Singh, Vinicius Reis, Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Ishan Misra. <https://github.com/facebookresearch/vissl>, 2021.
- [38] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020.
- [39] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7297–7306, 2018.
- [40] Agrim Gupta, Piotr Dollár, and Ross B. Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proc. CVPR*, 2019.
- [41] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proc. CVPR*, 2006.
- [42] Alex Hanna, Emily Denton, Razvan Amironesei, Andrew Smart, and Hilary Nicole. Lines of sight. In <https://logicmag.io/commons/lines-of-sight/>, 2020.
- [43] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *Proc. ICCV*, 2017.
- [44] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv.cs*, abs/1911.05722, 2019.
- [45] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019.
- [46] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 771–787, 2018.
- [47] Geoffrey E. Hinton and Richard S. Zemel. Autoencoders, minimum description length and Helmholtz free energy. In *Proc. NeurIPS*, 1993.
- [48] Christoph Hube, Besnik Fetahu, and Ujwal Gadiraju. Understanding and mitigating worker biases in the crowdsourced collection of subjective judgments. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2019.
- [49] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.
- [50] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.
- [51] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *Neural Information Processing Systems (NeurIPS)*, 2020.
- [52] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Mean shift for self-supervised learning, 2021.
- [53] Klemen Kotar, Gabriel Ilharco, Ludwig Schmidt, Kiana Ehsani, and Roozbeh Mottaghi. Contrasting contrastive self-supervised representation learning models, 2021.
- [54] Ivan Krasin, Tom Duerig, Neil Alldrin, Andreas Veit, Sami Abu-El-Haija, Serge Belongie, David Cai, Zheyun Feng, Vittorio Ferrari, Victor Gomes, Abhinav Gupta, Dhyanesh Narayanan, Chen Sun, Gal Chechik, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2016.
- [55] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Technical report*, 2009.
- [56] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, pages 1–26, 2020.

- [57] Gant Laborde. Nsfw detection machine learning model. https://github.com/GantMan/nsfw_model, 2020.
- [58] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. *ICLR*, 2021.
- [59] Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data, 2017.
- [60] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *ECCV*, 2014. ISBN 978-3-319-10602-1.
- [61] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *Proc. CVPR*, 2017.
- [62] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2445, 2020.
- [63] Milagros Miceli, Martin Schuessler, and Tianling Yang. Between subjectivity and imposition: Power dynamics in data annotation for computer vision. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–25, 2020.
- [64] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.
- [65] Ishan Misra, C Lawrence Zitnick, Margaret Mitchell, and Ross Girshick. Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2939, 2016.
- [66] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proc. ICVGIP*, 2008.
- [67] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *Proc. CVPR*, 2016.
- [68] Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily Denton, and Alex Hanna. Data and its (dis)contents: A survey of dataset development and use in machine learning research, 2020.
- [69] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [70] Vinay Prabhu. A study of “a study of face obfuscation in imagenet”. In <https://vinayprabhu.medium.com/a-study-of-a-study-of-face-obfuscation-in-imagenet-d7e7591795a2>, 2021.
- [71] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019.
- [72] Colorado J. Reed, Xiangyu Yue, Ani Nrusimha, Sayna Ebrahimi, Vivek Vijaykumar, Richard Mao, Bo Li, Shanghang Zhang, Devin Guillory, Sean Metzger, Kurt Keutzer, and Trevor Darrell. Self-supervised pretraining improves self-supervised pretraining. *arXiv:2103.12718*, 2021.
- [73] Mert Bulent Sariyildiz, Yannis Kalantidis, Diane Larlus, and Karteek Alahari. Concept generalization in visual representation learning. *arXiv:2012.05649*, 2020.
- [74] Amazon Web Services. Aws public datasets: Multimedia commons. <https://registry.opendata.aws/multimedia-commons/>, 2019.
- [75] Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D. Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world, 2017.
- [76] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018.
- [77] Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. *arXiv preprint arXiv:2003.05438*, 3(7), 2020.

- [78] Ryan Steed and Aylin Caliskan. Image representations learned with unsupervised pre-training contain human-like biases. *CoRR*, abs/2010.15052, 2021.
- [79] Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 498–512, 2018.
- [80] Kiat Chuan Tan, Yulong Liu, Barbara Ambrose, Melissa Tulig, and Serge Belongie. The herbarium challenge 2019 dataset. *arXiv preprint arXiv:1906.05372*, 2019.
- [81] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.
- [82] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. From ImageNet to image classification: Contextualizing progress on benchmarks. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9625–9635. PMLR, 13–18 Jul 2020.
- [83] Emiel van Miltenburg. Stereotyping and bias in the flickr30k dataset. In Jens Edlund, Dirk Heylen, and Patrizia Paggio, editors, *Proceedings of Multimodal Corpora: Computer vision and language processing (MMC 2016)*, pages 1–4, 2016. URL <http://www.lrec-conf.org/proceedings/lrec2016/workshops/LREC2016Workshop-MCC-2016-proceedings.pdf>.
- [84] Angelina Wang, Arvind Narayanan, and Olga Russakovsky. REVISE: A tool for measuring and mitigating bias in visual datasets. *European Conference on Computer Vision (ECCV)*, 2020.
- [85] Xudong Wang, Ziwei Liu, and Stella X Yu. Unsupervised feature learning by cross-level discrimination between instances and groups. *arXiv:2008.03813*, 2020.
- [86] Baoyuan Wu, Weidong Chen, Yanbo Fan, Yong Zhang, Jinlong Hou, Jie Liu, and Tong Zhang. Tencent ml-images: A large-scale multi-label image database for visual representation learning. *IEEE Access*, 7: 172683–172693, 2019. doi: 10.1109/ACCESS.2019.2956775.
- [87] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019.
- [88] Zhirong Wu, Yuanjun Xiong, X Yu Stella, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
- [89] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [90] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, FAT* ’20*, page 547–558. Association for Computing Machinery, 2020. ISBN 9781450369367.
- [91] Kaiyu Yang, Jacqueline Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in imagenet, 2021.
- [92] Amir Roshan Zamir, Alexander Sax, William B. Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proc. CVPR*, 2018.
- [93] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruysen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.
- [94] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989, 2017.
- [95] Nanxuan Zhao, Zhirong Wu, Rynson W. H. Lau, and Stephen Lin. What makes instance discrimination good for transfer learning? In *ICLR*, 2021.

- [96] Evgenii Zheltonozhskii, Chaim Baskin, Alex M. Bronstein, and Avi Mendelson. Self-supervised learning for large-scale unsupervised image clustering, 2020.
- [97] Bolei Zhou, Àgata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Proc. NeurIPS*, 2014.
- [98] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Antonio Torralba, and Aude Oliva. Places: An image database for deep scene understanding. *PAMI*, 2016.
- [99] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [100] Benjin Zhu, Junqiang Huang, Zeming Li, Xiangyu Zhang, and Jian Sun. Eqco: Equivalent rules for self-supervised contrastive learning. *arXiv preprint arXiv:2010.01929*, 2020.

Checklist

1. For all authors...
 - (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes] We propose a dataset for pretraining and evaluate it extensively in Tables 2b and 3 to 6.
 - (b) Did you describe the limitations of your work? [Yes] See in particular Section 6
 - (c) Did you discuss any potential negative societal impacts of your work? [Yes] We discuss the issue of our dataset still being US and Euro-centric in Section 3.
 - (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
 - (a) Did you state the full set of assumptions of all theoretical results? [N/A]
 - (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments (e.g. for benchmarks)...
 - (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code is attached as supplementary material and will be released together with the dataset.
 - (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See beginning of Section 5 and further details in Appendix.
 - (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Pretraining is very compute intensive. We could only run pretraining once per experiment.
 - (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] These details are provided in the implementation details in the Appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
 - (a) If your work uses existing assets, did you cite the creators? [Yes] We mainly use YFCC-100M [81] and mention them in Section 1, Section 3. All datasets used for evaluation are cited in Section 5.
 - (b) Did you mention the license of the assets? [Yes] The license is CC-BY, which is mentioned in Section 1 and Section 3.
 - (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We propose a new dataset, which is a subset of YFCC-100M. Download is made possible via code in the Supplementary Material and a link for reviewers to inspect it is provided in Appendix C
 - (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [Yes] We discuss this in Section 3.
 - (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] The data is created to explicitly not contain this information, as stated in Sections 1 and 3.
5. If you used crowdsourcing or conducted research with human subjects...
 - (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See Appendix C.
 - (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We filtered images for pornographic and disturbing content even before sending the images to the annotators.
 - (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes] See Section 3.

Appendix

Table of Contents

A Dataset Access	17
B Image attributions	17
C Dataset Generation Details	18
C.1 Automated pipeline	18
C.2 Human verification	18
C.3 Removal of duplicates	18
D Implementation Details	18
D.1 Representation learning experiments	18
D.2 Downstream tasks	19
E Additional Experimental Results	21
E.1 Object detection on PascalVOC	21
E.2 Data-efficient keypoint detection on COCO	22
E.3 Long-tailed instance segmentation on LVIS-v1	22
E.4 Cross-domain transfer	22
F Dataset Documentation: Datasheets for Datasets	23
F.1 Motivation	23
F.2 Composition	23
F.3 Collection process	24
F.4 Preprocessing/cleaning/labeling	25
F.5 Uses	25
F.6 Distribution	25
F.7 Maintenance	26
F.8 Other questions	26
F.9 Author statement of responsibility	26

A Dataset Access

During review, we had provided instructions for the reviewers to view the dataset in the OpenReview submission. The dataset is hosted by Zenodo [31] and can be found under this URL:

<https://zenodo.org/record/5501843>

B Image attributions

The creators for the images in Fig. 1, are (top-left to bottom-right): *RichTatum, chongeileen, Vlad Iorsh, millstastic, Ross_Angus, nfeli777, semper_fi_brother, jj-photo, BluEyedA73, Pazit Polak, florianplag, Johan Lange, kimadababe, Alzheimer’s Association - Greater Illinois, Esme_Vos, babeltravel, Henrique B Costa, Dude of Lego, matthewreid, Russ Dill, theunwiseman, Wenkan, at8eqeq3, Larry1732, Haroldo Kennedy, DGriebeling, Tez_kuma, Overman Alawami, FreeCat, fczuardi, NorisaI, pescatello, jasonlsraia, semanticwebcompany, jumblejet, Rain San Martin, crackerbelly, Problemkind, M_Hartman Photography, antaeon, andyket, mirven, Quinn Rossi,*

MARCO_QUARANTOTTI, -Tripp-, JulieHagenbuch, frank drewett, MeRyan, Sharib4rd, Schlüsselbein2007, BBR1245, Officer Phil, cadyellow, Andreas Joensen, FuFuWolf, zerojay, Spoon Monkey, Tatters80, gypsygirl.photography, ChristinaT, Irene Vlachou, allenreichert, rs-foto, Sara Cimino, JonoMueller, Lock The Gate, m01229, ebis50, Speculum Mundi, fontosiskola, rakh1, dreamcycle19772006 ON OFF.

All images in this paper and the dataset are licensed by the **CC-BY** (<https://creativecommons.org/licenses/by/2.0/>) license and contain information on their creator. On the above url, this license is described as follows:

1. Copy and redistribute the material in any medium or format
2. Remix, transform, and build upon the material for any purpose, even commercially.
3. This license is acceptable for Free Cultural Works.
4. The licensor cannot revoke these freedoms as long as you follow the license terms.

C Dataset Generation Details

C.1 Automated pipeline

For the detection of faces we use RetinaFace [23] and the MobileNet0.25 model from the PyTorch RetinaFace repository (https://github.com/biubug6/Pytorch_Retinaface). We keep all parameters, including the detection threshold value of 0.02. For the automated detection of humans and human parts we use the highly performant Cascade-RCNN trained using the 3x schedule on COCO, available from the detectron2 library [87]. The threshold for detections is set to its default value of 0.5.

Finally, before passing the images to the professional human annotators, we run the a model pretrained to detect “not safe for work” (NSFW) content on the set of images from ([57]). After visually inspecting the top 500 entries for both the ‘hentai’ and the ‘porn’ categories, and not finding any positive results, we pass the data to the human annotators.

C.2 Human verification

In Fig. 5, we provide two screenshots of the verification instructions given to the annotators. We have manually assessed the quality of these annotations using two independent samples of size 1000 each and only found 15 false positives and 2 minor false negatives. We informed the annotation team of our findings to further improve the quality. The important error type here are false negatives, as this would mean humans or personal information remains undetected.

After human verification 2% of images are flagged and subsequently removed. The vast majority of flagged images stem from the presence of car licence plates.

C.3 Removal of duplicates

We also conduct a basic removal of duplicates process using the Find Identical Images tool [26]. Using this we find and remove 2529 self-duplicates, 114 duplicates with MS COCO, and 9 duplicates with ImageNet LSVRC-12. We further checked against duplicates with Places-205, but found none.

D Implementation Details

D.1 Representation learning experiments

MoCo-v2. We pretrain the MoCo-v2 models using its default settings for 200 epoch using the publicly available repository (<https://github.com/facebookresearch/moco>). This includes training using the cosine learning rate schedule, augmentations consisting of random resized crops, colorjitter, random horizontal flipping and blurring with a batch-size of 256 and a learning rate of 0.03. On two RTX6000 GPUs this training requires around one week to complete.

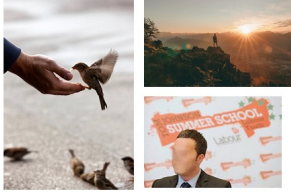
PASS-Dataset Labelling Task

Dataset Statistics

- 1.6M Images
- 2 Classes: OK vs. Not-OK
- The dataset has been pre-filtered with various tools
- We expect 99% of the images to be OK

Examples of Not-OK images:

- Images that contain people, or body parts (hands, faces, legs, etc):



- Images that contain personal information (ID, license plates, names, etc.)



- Images that contain offensive content such as nudity, blood, drugs (the dataset is pre-filtered, so this should not happen)



When an image does not contain any of these things, it is "OK".

A random selection of images (there are 2 Not-Ok images overall):

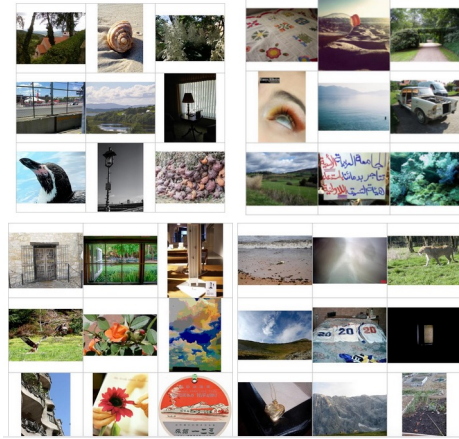


Figure 5: **Human verification instructions.** We have blurred faces and other personally identifiable information solely for in these screenshots. The professional annotators saw unblurred versions to provide a clearer understanding of what images require flagging.

SwAV. We pretrain SwAV [12] using the fixed hyperparameters for 200 epoch pretraining with a batch-size of 256 found at the publicly available repository (<https://github.com/facebookresearch/swav>). For this we use 4 RTX6000 GPUs and this requires training for a bit more than a week. SwAV’s augmentations include those of MoCo-v2 and in addition Multi-Crop, in which in addition to the two 224-sized crops, six 96x96-sized crops are extracted and used for computing the loss.

DINO. We pretrain DINO [13] using for 100 epochs with a batch-size of 256 using the publicly available repository (<https://github.com/facebookresearch/dino>). Training augmentations include those of SwAV, including Multi-Crop, as well as solarization, introduced in [38]. Here, the architecture trained is ViT-B [24] and training takes around 2 days on 8 GPUs.

Pretrained models. We utilise pretrained models for the MoCo-v2 models trained on Places and ImageNet from the MoCo-v2 repository and from the repository of [95]. In addition we use PyTorch’s publicly available models: a supervisedly pretrained ResNet50 model for benchmarking and a pretrained MobileNet-v3 for generating the $\approx$ IN800 split.

D.2 Downstream tasks

Cluster evaluation. We conduct these experiments on the test sets of the datasets by resizing the images to smaller-side 256pixels and taking a centered 224x224 crop. These images’ embeddings after global average pooling are computed and input in k-means, implemented by the FAISS [49] library. For k-means, we use 50 iterations and 5 restarts and set K to be equal to the number of ground-truth classes. NMI, aRI calculations are computed by scikit-learn [69].

Linear probing. We exactly follow MoCo-v2’s implementation. For ease, we describe it here: Training is done for 100 epochs on the global average pooled features of the encoders without the use of an additional BatchNorm layer. The learning rate is dropped by a factor of 0.1 at epochs 60, 80 starting at a value of 30.0. The batchsize is 256, and we use a single GPU. The augmentations during training consist of random horizontal flipping and RandomResizedCrops to size 224x224 with the

Table 7: Differences between linear evaluation implementations.

model	Linear eval code		model	Linear eval code	
	SwAV	MoCo-v2		SwAV	MoCo-v2
SwAV	60.4	57.5	SwAV	72.7	69.2
MoCo-v2	47.0	59.1	MoCo-v2	58.2	67.7

(a) Pretraining on **PASS**.

(b) Pretraining on IN-1k.

Table 8: SwAV linear probing performance. We report the linear evaluation performance of the SwAV pretrained model when using SwAV’s linear evaluation code.

SSL method	Linear probing				
	IN-1k	ObjN	Plcs	Flwrs	HT19
SwAV	60.4	10.4	55.0	88.3	38.7

default parameters (crops of sizes (0.08, 1.0) of the original size and variations in aspect ratio of (3/4, 4/3)). For linear probing on the Vision Transformer [24], we use the 100 epoch linear evaluation code of [13].

Data-efficient SVM classification. We exactly follow the publicly available implementation of PCL [58] (<https://github.com/salesforce/PCL>). Here, a one-vs-all SVM is fit on the data using a cost parameter $c = 0.5$, which we keep at its default value. The features for the SVM are obtained by smaller-side-resizing to 256 and taking a center crop of 224x224.

Finetuning. We follow the publicly available implementation of MoCo [44]. For the R50-C4 architecture used in the Pascal VOC experiments, this includes the use of an additional normalisation layer and training for 24k steps. For FPN architectures (used for COCO detection, segmentation, keypoint, densepose and LVIS segmentation) this includes the use of SyncBatchNorm and training using the ‘1x’ schedule for which trains for 90K steps and has a learning rate warmup for the first 1000 steps from 0.02/100 and subsequent training with an initial learning rate of 0.02 being divided by 10 at steps 60K and 80K. For the data-efficient keypoint detection we use the R50-FPN with a fixed split containing 5% of the annotations of COCO2017’s train split, train for 9K steps and evaluate on the full validation split. For the dense pose estimation on COCO [60] we use an R50-FPN with chart-based embeddings [39] and the ‘s1x’ schedule, which decays the learning rate at steps 100K, 120K and trains for 130K steps in total. Training is done on the 2014 densepose training split and the 2014 ‘valminusminival’ split, and performance tested on the 2014 ‘minival’ split. Note that the default ‘1x’ schedule of LVIS includes 180K iterations, so in these terms our LVIS training can be referred to as ‘0.5x’. Instead of a batch-size of 16 we use a batch-size of 8 on two GPUs and divide the learning rates by half and the multiply the learning-rate schedules by two. We use the detectron2 library [87].

Following the MoCo-v2 implementation and experiments, we evaluate object detection and segmentation on COCO [60] using a Mask-RCNN [43] with the R50-FPN backbone [61] with the ‘1x’ schedule and object detection on Pascal VOC07+12 [32] using a R50-C4 using 24k steps.

D.2.1 A note on evaluation hyperparameters

Finally, we note that comparing different self-supervised methods using linear evaluation is non-trivial, as the optimal settings (learning rate and schedule) can depend on the encoder’s pretraining method. As we show in Table 7, both MoCo-v2 and SwAV suffer strongly when instead of their linear evaluation code, the other’s is used. For completeness, we report the linear probing numbers of SwAV when their linear evaluation code is used in Table 8.

While outside of this work, frozen evaluation of methods which do not require further finetuning and are close to deployment also include the clustering evaluation in the paper (which can be combined with a Hungarian algorithm to yield label predictions), kNN evaluations (as done in e.g. [13]) and the normalisation of feature statistics before training the linear layer, as in [52].

D.2.2 Dataset details

For detection and segmentation in Table 3a we use COCO2017, which contains 80 labeled objects with segmentation masks and boxes. The training set contains 118K images with 850K annotations, and the validation set 5K images with 36K annotations.

For dense human pose estimation in Table 3b, COCO2014 (a subset of COCO2017) is used, which contains masks of people and their bodies’ 3D surface on a mesh. The training and validation sets contain 32K and 1.5K images with 99K and 25K persons instances, respectively.

For object detection in Table 9a we use the common Pascal VOC “07+12” split containing 20 object categories. This split uses the VOC7’s test set (2.5K images and 15K annotations) for evaluation and merges the two trainval sets for finetuning (16.5K images and 47K annotations).

For data-efficient keypoint detection in Table 9b, we generate a random, fixed split of COCO2017, resulting in 1K images with 2K annotations for training and the full test set of 5K images with 10K annotations.

For detection and segmentation in Table 9c, we use LVIS-v1 [40], which contains 1203 labeled objects with segmentation masks and boxes. The training set contains 100K images with 1.2M annotations, and the validation set 5K images with 244K annotations.

For cross-domain transfer in Table 10, we use multiple smaller-scale datasets taken from the Visual Task Adaptation Benchmark (VTAB) [93]: CLEVR-count [50] contains synthetic rendered images of objects and the task is to count the number of objects via 8-way classification and contains 70K, 15K images for training and testing. Our construction of the dataset follows the code from the VISSL repository [37]. DTD [19] contains 47 classes of different describable textures (e.g. ‘honeycombed’ or ‘sprinkled’), and we use the first train and test-split supplied in the paper, containing around 2K images each. Both Eurosat [45] and Resisc45 [18] are remote sensing image classification datasets with 10 and 45 classes respectively (e.g. ‘baseball diamond’, or ‘river’). They do not come with a natural train/test split, so we create a fixed training split using 60% and 20% of the data, following VTAB [93]. This yields around 21K and 25K images for training and 5K and 6K for testing, for Eurosat and Resisc45, respectively.

E Additional Experimental Results

Table 9: **Finetuning representation evaluations.** a) PascalVOC07+12 detection and (b) human keypoint estimation and c) Object segmentation on LVIS. AP denotes COCO’s AP on IoUs [0.5:0.95:0.05].

Init.	Bounding-box		
	AP ₅₀	AP	AP ₇₅
Random	52.5	28.1	26.2
Supv. IN-1k	81.3	53.5	58.8
Supv. IN-1k*	80.9	53.5	58.8
MoCo-v2			
on IN-1k	82.3	56.8	63.1
on IN-1k*	81.5	55.7	62.0
on Places	81.5	56.3	62.8
on PASS	81.1	55.5	61.2

(a) Object detection on Pascal VOC07+12 using Faster-RCNN, R50-C4, and 24K steps.

Init.	AP ^{kp}	AP ^{kp} ₅₀	AP ^{kp} ₇₅
Random	25.9	52.9	21.5
Supv. IN-1k	39.7	69.4	39.5
MoCo-v2 on PASS	40.0	68.7	39.8

(b) Human keypoint detection on 5% of COCO.

Init.	AP	AP _r	AP _f	AP _c
Random	11.4	2.7	18.3	8.7
Supv. IN-1k	18.9	7.6	26.3	16.6
MoCo-v2 on PASS	16.6	4.1	24.9	14.2

(c) Object segmentation on LVIS-v1.

E.1 Object detection on PascalVOC

In Table 9a we report the performances of object detection using R50-C4 on Pascal VOC with 24k steps (as in MoCo). As reported in previous works [44, 95], the self-supervised baselines outperform supervised pretraining especially the more strict AP and AP75 measures. We find that pretraining on **PASS** falls within 1% of the performance of pretraining on IN-1k without humans.

E.2 Data-efficient keypoint detection on COCO

In Table 9b, we show the results of finetuning the encoders using a small sample of ground-truth annotations from the COCO 2014 keypoint annotations. While using 100% of the data results in almost no gains between pretrained and random initialisations (as shown in [44]), we see a large difference when only 5% of the annotations are considered. We find that pretraining on **PASS** yields the strongest keypoint detection results in terms of AP and AP75, surpassing even supervised IN-1k pretraining.

E.3 Long-tailed instance segmentation on LVIS-v1

In Table 9c, we show the results of finetuning for object segmentation using Mask-RCNN with R50-FPN on LVIS-v1 [40], which contains 1203 classes. We find that different to object detection on COCO in Table 3a, the supervisedly trained baseline performs best with an overall AP of 18.9, and the MoCo-v2 pretrained on **PASS** yielding an AP of 16.6, and large gap in the ‘rare’ categories, as measured by AP_r .

E.4 Cross-domain transfer

Table 10: **Cross-domain transfer**. Linear-probing Top-1 accuracies are reported for various datasets whose domain is further from typical web-images.

Model	Resisc45	Clevr-cnt	Eurosat	DTD
sup. IN-1k	90.2	61.8	96.6	64.5
MoCo-v2 Places	91.0	70.4	93.5	62.9
MoCo-v2 IN-1k	90.7	72.6	96.5	66.7
MoCo-v2 PASS	91.4	73.1	95.4	66.8

In Table 10, we report cross-domain transfer results on the following datasets and tasks taken from VTAB [93]: CLEVR-count [50] (object counting), DTD [19] (texture classification) and Eurosat [45] (satellite image land-cover classification) Resisc45 [18] (remote sensing image classification). We use the same linear-evaluation protocol as in the remainder of the paper, i.e. the settings from MoCo-v2, without any hyperparameter tuning.

From Table 10 we find that MoCo-v2 trained on our **PASS** dataset does very well across those datasets, surpassing both IN-1k supervised and MoCo-v2 on IN-1k for three out of four cases. This shows that pretraining on **PASS** is a viable strategy even if the downstream task domain is very different.

F Dataset Documentation: Datasheets for Datasets

Here we answer the questions outlined in the datasheets for datasets paper by Gebru et al. [33].

F.1 Motivation

For what purpose was the dataset created? Neural networks pretrained on large image collections have been shown to transfer well to other visual tasks where there is little labelled data, i.e. transferring a model works better than starting with a randomly initialized network every time for a new task, as many visual features can be repurposed. This dataset has as its goal to provide a safer large-scale dataset for such pretraining of visual features. In particular, this dataset does not contain any humans or human parts and does not contain any labels. The first point is important, as the current standard for pretraining, ImageNet and its face-blurred version only provide pseudo-anonymity and furthermore do not provide correct licences to the creators. The second point is relevant as pretraining is moving towards the self-supervised paradigm, where labels are not required. Yet most methods are developed on the highly curated ImageNet dataset, yielding potentially non-generalizeable research.

Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)? The dataset has been constructed by the research group “Visual Geometry Group” at the University of Oxford at the Engineering Science Department.

Who funded the creation of the dataset? The dataset is created for research purposes at the VGG research group. Individual researchers have been funded by AWS Machine Learning Research Awards (MLRA), EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines & Systems [EP/L015897/1], the Qualcomm Innovation Fellowship, Innovate UK (project 71653) on behalf of UK Research and Innovation (UKRI) and by the European Research Council (ERC) IDIU-638009.

F.2 Composition

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? This dataset only contains photos. In addition we provide tabular meta-data for these images, which contain information such as the creator’s username and image capture date.

How many instances are there in total (of each type, if appropriate)? The dataset contains 1.4M images, resulting in 181GB as a tar file.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? The dataset is a sample of a larger set—all possible digital photographs. As outlined in Section 3 we start from an existing dataset, YFCC-100M, and stratify the images (removing images with people and personal information, removing images with harmful content, removing images with unsuitable licenses, each user contributes at most 80 images to the dataset). This leaves 1.6M images, out of which we take a random sample of 1.28M images to replicate the size of the ImageNet dataset. While this dataset can thus be extended, this is the set that we have verified to not contain humans, human parts and disturbing content.

What data does each instance consist of? Digital photographs uploaded by users of the flickr platform.

Is there a label or target associated with each instance? No. Our dataset deliberately does not contain labels.

Is any information missing from individual instances? Not from the dataset. Note however that the meta-data that we additionally provide is not complete and might have non-uniform missing values.

Are relationships between individual instances made explicit (e.g., users’ movie ratings, social network links)? Not applicable: each image stands on its own and we do not provide relationships between these.

Are there recommended data splits (e.g., training, development/validation, testing)? As outlined in the intended usecases, this dataset is meant for pretraining representations. As such, the models derived from training on this dataset need to be evaluated on *different datasets*, so called down-stream tasks. Thus the recommended split is to use all samples for training.

Are there any errors, sources of noise, or redundancies in the dataset? No.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? No. The dataset contains links to the publicly hosted mirror of the YFCC dataset on Amazon Web Services.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)? No.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? No. Besides checking for human presence in the images, the annotators were also given the choice of flagging images for disturbing content, which once flagged was removed.

Does the dataset relate to people? If not, you may skip the remaining questions in this section. No.

Does the dataset identify any subpopulations (e.g., by age, gender)? NA

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? NA

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? NA

F.3 Collection process

How was the data associated with each instance acquired? The data was collected from the publicly available dataset YFCC-100M which is hosted on the AWS public datasets platform. We have used the meta-data, namely the copyright information to filter only images with the CC-BY licence and have downloaded these using the aws command line interface, allowing for quick and stable downloading. In addition, all files were subsequently scanned for viruses using Sophos SAVScan virus detection utility, v.5.74.0.

What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)? Our dataset is a subset of the YFCC-100M dataset. The YFCC-100M dataset itself was created by effectively randomly selecting publicly available images from flickr, resulting in approximately 98M images.

If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)? See the similar question in the Composition section.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)? As described, the data was collected automatically by simply downloading images from a publicly hosted S3 bucket. The human verification was done using a professional data annotation company that pays 150% of the local minimum wage.

Over what timeframe was the data collected? The images underlying the dataset were downloaded between March and June 2021 from the AWS public datasets' S3 bucket, following the download code provided in the repo. However the images contained were originally and taken anywhere from 2000 to 2015, with the majority being shot between 2010-2014.

Were any ethical review processes conducted (e.g., by an institutional review board)? No.

Does the dataset relate to people? If not, you may skip the remainder of the questions in this section. No, this is one of the specific purposes of this dataset.

F.4 Preprocessing/cleaning/labeling

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? After the download of approx. 17M images, the corrupted, or single-color images were removed from the dataset prior to the generation of the dataset(s) used in the paper. The images were not further preprocessed or edited.

Was the “raw” data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? Yes. The creators of the dataset maintain a copy of the 17M original images with the CC-BY licence of YFCC100M that sits at the start of our dataset creation pipeline.

Is the software used to preprocess/clean/label the instances available? We have only used basic Python primitives for this. For the annotations we have used VIA [27, 28]

F.5 Uses

Has the dataset been used for any tasks already? In the paper we show and benchmark the intended use of this dataset as a pretraining dataset. For this the dataset is used an unlabelled image collection on which visual features are learned and then transferred to downstream tasks. We show that with this dataset it is possible to learn competitive visual features, without any humans in the pretraining dataset and with complete license information.

Is there a repository that links to any or all papers or systems that use the dataset? We will be listing these at the repository.

What (other) tasks could the dataset be used for? We believe this dataset might allow researchers and practitioners to further evaluate the differences that pretraining datasets can have on the learned features. Furthermore, since the meta-data is available for the images, it is possible to investigate the effect of image resolution on self-supervised learning methods, a domain largely underresearched thus far, as the current de-facto standard, ImageNet, only comes in one size.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? Given that this dataset is a subset of a dataset that randomly samples images from flickr, the image distribution is biased towards European and American creators. As in the main papers discussion, this can lead to non-generalizeable features, or even biased features as the images taken in other countries might be more likely to further reflect and propagate stereotypes [84], though in our case these do not refer to stereotypes about humans.

Are there tasks for which the dataset should not be used? This dataset is meant for research purposes only. The dataset should also not be used for, e.g. connecting images and usernames, as this might risk de-anonymising the dataset in the long term. The usernames are solely provided for attribution.

F.6 Distribution

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created? No.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub)? The dataset will be provided as a csv along with code hosted on GitHub that allows the user to download the images in our dataset. In addition, we hope to also host it as a single tarball on our servers.

When will the dataset be distributed? Starting from July 2021.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? CC-BY.

Have any third parties imposed IP-based or other restrictions on the data associated with the instances? No.

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? Not that we are of. Regular UK laws apply.

F.7 Maintenance

Who is supporting/hosting/maintaining the dataset? The dataset is supported by the authors and by the VGG research group. The main contact person is Yuki M. Asano. We host the dataset on zenodo: <https://zenodo.org/record/5528345>.

How can the owner/curator/manager of the dataset be contacted (e.g., email address)? The authors of this dataset can be reached at their e-mail addresses: yuki,chriss,vedaldi,az@robots.ox.ac.uk. In addition, we have added a contact form in which we can be contacted anonymously at <https://forms.gle/tkZugt2DJnFdCE1i6>.

Is there an erratum? If errors are found and erratum will be added to the website.

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? Yes, updates will be communicated via the website. The dataset will be versioned.

If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)? Not applicable.

Will older versions of the dataset continue to be supported/hosted/maintained? If even after our verification we find further images that contain humans or problematic content, we will remove those further images from existing splits to preserve the goal of this dataset.

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? Others are free to reach out to us if their ideas can build on this dataset. All code will be made available.

F.8 Other questions

Is your dataset free of biases? No. There are many kinds of biases that can either be quantified, e.g. geo-location (most images originate from the US and Europe) or camera-model (most images are taken with professional DSLR cameras not easily affordable), there are likely many more biases that this dataset does contain. The only thing that this dataset does not contain are humans and parts of humans, as far as our validation procedure is accurate.

Can you guarantee compliance to GDPR? No, we cannot comment on legal issues.

F.9 Author statement of responsibility

The authors confirm all responsibility in case of violation of rights and confirm the licence associated with the dataset and its images.

Bibliography

- Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2018a.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. *Interspeech*, 2018b.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Deep lip reading: A comparison of models and an online application. In *Interspeech*, 2018c.
- Triantafyllos Afouras, Yuki M. Asano, Francois Fagan, Andrea Vedaldi, and Florian Metze. Self-supervised object detection from audio-visual correspondence. In *arXiv:2104.06401*, 2021.
- Hassan Akbari, Linagzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. In *arXiv:2104.11178*, 2021.
- Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *Proc. NeurIPS*, 2020.
- Humam Alwassel, Bruno Korbar, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *Proc. NeurIPS*, 2020a.
- Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *Proc. NeurIPS*, 2020b.
- R. Arandjelović and A. Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017.
- Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.
- Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *Proc. NeurIPS*, 2020a.
- Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *International Conference on Learning Representations (ICLR)*, 2020b.
- Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020c.

Bibliography

- Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Proc. NeurIPS*, 2016.
- Kumar Ayush, Burak Uzkent, Chenlin Meng, Kumar Tanmay, Marshall Burke, David Lobell, and Stefano Ermon. Geography-aware self-supervised learning. *arXiv:2011.09980*, 2020.
- Shekoofeh Azizi, Basil Mustafa, Fiona Ryan, Zachary Beaver, Jan Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, Ting Chen, Vivek Natarajan, and Mohammad Norouzi. Big self-supervised models advance medical image classification, 2021.
- Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *arXiv:2104.00650*, 2021.
- Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv:2106.08254*, 2021.
- Manel Baradad, Jonas Wulff, Tongzhou Wang, Phillip Isola, and Antonio Torralba. Learning to see by looking at noise. In *arXiv:2106.05963*, 2021.
- Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv:2105.04906*, 2021.
- Horace B Barlow. Unsupervised learning. *Neural computation*, 1(3):295–311, 1989.
- Sagie Benaïm, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos. In *Proc. CVPR*, 2020.
- Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proc. ICML*, pages 517–526. PMLR, 2017.
- Andrew Brown, Vicky S. Kalogeiton, and Andrew Zisserman. Face, body, voice: Video person-clustering with multiple modalities. *arXiv:2105.09939*, 2021.
- Qi Cai, Yu Wang, Yingwei Pan, Ting Yao, and Tao Mei. Joint contrastive learning with infinite possibilities. In *Proc. NeurIPS*, 2020.
- M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *Proc. ECCV*, 2018.
- Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Proc. NeurIPS*, 2020.
- Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv:2104.14294*, 2021.
- Brian Chen, Andrew Rouditchenko, Kevin Duarte, Hilde Kuehne, Samuel Thomas, Angie Boggust, R. Panda, Brian Kingsbury, R. Feris, David F. Harwath, J. Glass, M. Picheny, and Shih-Fu Chang. Multimodal clustering networks for self-supervised learning from unlabeled videos. *arXiv:2104.12671*, 2021a.

Bibliography

- Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Audio-visual embodied navigation. *arXiv:1912.11474*, 2019.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020a.
- Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33:22243–22255, 2020b.
- Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *Proc. CVPR*, 2021.
- Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv:2003.04297*, 2020c.
- Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers, 2021b.
- Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatio-temporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.
- Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proc. CVPR*, volume 1, pages 539–546. IEEE, 2005.
- J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.
- Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. *Proc. CVPR*, 2017.
- Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.
- Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Proc. NeurIPS*, pages 2292–2300, 2013.
- Virginia R. de Sa. Learning classification with unlabeled data. In *Proc. NeurIPS*, 1994.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009a.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, pages 248–255. Ieee, 2009b.
- Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proc. ICCV*, 2017.

Bibliography

- Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, pages 1422–1430, 2015.
- Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning, 2019.
- Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *Proc. ICLR*, 2017.
- A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE PAMI*, 38(9):1734–1747, Sept 2016. ISSN 0162-8828. doi: 10.1109/TPAMI.2015.2496141.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.
- Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.
- Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *arXiv:2104.14548*, 2021.
- Gerald M Edelman. *Neural Darwinism: The theory of neuronal group selection*. Basic books, 1987.
- Aleksandr Ermolov, Aliaksandr Siarohin, Enver Sangineto, and Nicu Sebe. Whitening for self-supervised representation learning. *ICML*, 2020.
- M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.
- Facebook AI Research. <https://ai.facebook.com/blog/a-state-of-the-art-self-supervised-framework-for-video-understanding/>, 2020.
- Facebook AI Research. <https://ai.facebook.com/blog/learning-from-videos-to-understand-the-world/>, 2021.
- Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. In *arXiv:2104.14558*, 2021.
- Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *Proc. CVPR*, pages 10364–10374, 2019.
- Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proc. CVPR*, 2017.
- Valentin Gabeur, Chen Sun, Karteeek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.

Bibliography

- Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Revisiting contrastive methods for unsupervised learning of visual representations. In *arXiv:2106.05967*, 2021.
- Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. Listen to look: Action recognition by previewing audio. In *Proc. CVPR*, 2020.
- Spyros Gidaris, Praveen Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. ICLR*, 2018.
- Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *Proc. CVPR*, 2020.
- Spyros Gidaris, Andrei Bursuc, Gilles Puy, Nikos Komodakis, Matthieu Cord, and Patrick Pérez. Learning representations by predicting bags of visual words. In *Proc. CVPR*, 2021.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.
- Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *Proc. ICCV*, 2019.
- Priya Goyal, Mathilde Caron, Benjamin Lefaudeaux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, and Piotr Bojanowski. Self-supervised pretraining of visual features in the wild. In *arXiv:2103.01988*, 2021.
- Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020.
- Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proc. CVPR*, 2006.
- Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV Workshops*, 2019.
- Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020a.
- Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *Proc. NeurIPS*, 2020b.
- Alex Hanna, Emily Denton, Razvan Amironesei, Andrew Smart, and Hilary Nicole. Lines of sight. In <https://logicmag.io/commons/lines-of-sight/>, 2020.
- David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *ECCV*, 2018.
- Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proc. CVPR*, pages 4918–4927, 2019.

Bibliography

- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.
- Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.
- Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md. Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. In *arXiv:1712.00409*, 2017.
- Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. *arXiv preprint arXiv:2011.08435*, 2020.
- D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of Physiology*, 160:106–154, 1962.
- Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Proc. CVPR*, 2018.
- Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.
- Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. In *Proc. NeurIPS*, 2020.
- Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without natural images. In *arXiv:2101.08515*, 2021.
- Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv:1705.06950*, 2017.
- Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *Proc. ICCV*, 2019.
- Dahun Kim, Donghyeon Cho, Donggeun Yoo, and In So Kweon. Learning image representations by completing damaged jigsaw puzzles. In *WACV*, pages 793–802. IEEE, 2018.
- Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.
- Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Mean shift for self-supervised learning. In *arXiv:2105.07269*, 2021.

Bibliography

- Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *Proc. NeurIPS*, 2018.
- Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *Proc. ICCV*, 2019.
- Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation. In *arXiv:2006.07203*, 2020.
- Klemen Kotar, Gabriel Ilharco, Ludwig Schmidt, Kiana Ehsani, and Roozbeh Mottaghi. Contrasting contrastive self-supervised representation learning models, 2021.
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.
- H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *Proc. ICCV*, 2011.
- Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018.
- Zihang Lai, Erika Lu, and Weidi Xie. Mast: A memory-augmented self-supervised tracker. In *Proc. CVPR*, pages 6479–6488, 2020.
- Iro Laina, Ruth Fong, and Andrea Vedaldi. Quantifying learnability and describability of visual concepts emerging in representation learning. In *Proc. NeurIPS*, 2020.
- Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proc. ECCV*, pages 577–593. Springer, 2016.
- Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proc. CVPR*, 2017.
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proc. ICCV*, 2017.
- Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. *ICLR*, 2021.
- Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv preprint arXiv:2001.05691*, 2020.
- Feng Lin, Haohang Xu, Houqiang Li, Hongkai Xiong, and Guo-Jun Qi. Aetv2: Autoencoding transformations for self-supervised representation learning by minimizing geodesic distances in lie groups. *arXiv:1911.07004*, 2019.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014.

Bibliography

- Yu Liu, Lianghua Huang, Pan Pan, Bin Wang, Yinghui Xu, and Rong Jin. Train a one-million-way instance classifier for unsupervised visual representation learning. In *AAAI*, 2021.
- Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. In *arXiv:2002.06353*, 2020.
- Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofer Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152:166–177, 2019. ISSN 0924-2716.
- Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proc. CVPR*, pages 2437–2445, 2020.
- Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100M: Learning a text-video embedding by watching hundred million narrated video clips. In *Proc. ICCV*, 2019.
- Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proc. CVPR*, 2020.
- Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proc. CVPR*, 2020.
- Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.
- Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. *arXiv:2004.12943*, 2020.
- Pedro Morgado, Ishan Misra, and Nuno Vasconcelos. Robust audio-visual instance discrimination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12934–12945, 2021.
- T. Nathan Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. In *Proc. CVPR*, pages 9339–9348, 2018.
- Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *Proc. CVPR*, 2020.
- Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, pages 69–84. Springer, 2016.
- Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proc. ICCV*, 2017.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Bibliography

- Emin Orhan, Vaibhav Gupta, and Brenden M Lake. Self-supervised learning through the eyes of a child. *Proc. NeurIPS*, 33, 2020.
- Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.
- Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.
- Papers with Code. <https://paperswithcode.com/sota/self-supervised-image-classification-on>, 2021.
- Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proc. CVPR*, pages 2536–2544, 2016.
- Mandela Patrick, Yuki M. Asano, Polina Kuznetsova, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. In *arXiv:2003.04298*, 2020.
- Mandela Patrick, Yuki M. Asano, Bernie Huang, Ishan Misra, Florian Metze, Joao Henriques, and Andrea Vedaldi. Space-time crop & attend: Improving cross-modal video representation learning. In *arXiv:2103.10211*, 2021a.
- Mandela Patrick, Po-Yao Huang, Yuki M. Asano, Florian Metze, Alexander Hauptmann, João Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *ICLR*, 2021b.
- Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily Denton, and Alex Hanna. Data and its (dis)contents: A survey of dataset development and use in machine learning research. In *arXiv:2012.05345*, 2020.
- Karol J. Piczak. Esc: Dataset for environmental sound classification. In *ACM Multimedia*, 2015.
- AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *Proc. CVPR*, 2020.
- Gerasimos Potamianos, Chalapathy Neti, Guillaume Gravier, Ashutosh Garg, and Andrew W Senior. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.
- Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv:1511.06434*, 2015.
- Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, Jean-Bastien Grill, Aäron van den Oord, and Andrew Zisserman. Broaden your views for self-supervised video learning. In *arXiv:2103.16559*, 2021.
- Andrew Rouditchenko, Angie Boggust, David Harwath, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Rogerio Feris, Brian Kingsbury, Michael Picheny, Antonio Torralba, et al. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*, 2020.

Bibliography

- Mert Bulent Sariyildiz, Yannis Kalantidis, Diane Larlus, and Karteek Alahari. Concept generalization in visual representation learning. *arXiv:2012.05649*, 2020.
- Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D. Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world. *arXiv:1711.08536*, 2017.
- Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018.
- Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-mix: Rethinking image mixtures for unsupervised visual representation learning. *arXiv preprint arXiv:2003.05438*, 3(7), 2020.
- Fabian H. Sinz, Xaq Pitkow, Jacob Reimer, Matthias Bethge, and Andreas S. Tolias. Engineering a less artificial intelligence. *Neuron*, 103(6):967–979, 2019. ISSN 0896-6273. doi: <https://doi.org/10.1016/j.neuron.2019.08.034>. URL <https://www.sciencedirect.com/science/article/pii/S0896627319307408>.
- Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.
- Ryan Steed and Aylin Caliskan. Image representations learned with unsupervised pre-training contain human-like biases. *CoRR*, abs/2010.15052, 2021.
- D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley. Detection and classification of acoustic scenes and events. *IEEE Transactions on Multimedia*, 2015.
- Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv:1906.05743*, 2019a.
- Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proc. ICCV*, 2019b.
- Kiat Chuan Tan, Yulong Liu, Barbara Ambrose, Melissa Tulig, and Serge Belongie. The herbarium challenge 2019 dataset. *arXiv:1906.05372*, 2019.
- Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020a.
- Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Proc. NeurIPS*, 2020b.
- Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proc. CVPR*, pages 8769–8778, 2018.
- Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence – video to text. In *Proc. ICCV*, 2015.

Bibliography

- Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proc. CVPR*, 2019a.
- Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proc. ICML*, volume 119, pages 9929–9939. PMLR, 13–18 Jul 2020.
- Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. VateX: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4581–4591, 2019b.
- Chen Wei, Huiyu Wang, Wei Shen, and Alan Yuille. {CO}2: Consistent contrast for unsupervised visual representation learning. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=U4XLJhqwNF1>.
- Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proc. CVPR*, 2018.
- Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, 2018a.
- Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. CVPR*, pages 3733–3742, 2018b.
- Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *ICLR*, 2021.
- Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Delving into inter-image invariance for unsupervised visual representations. In *arXiv:2008.11702*, 2021a.
- Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. *Proc. CVPR*, 2021b.
- Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proc. CVPR*, 2019.
- Jie Xu, Huayi Tang, Yazhou Ren, Xiaofeng Zhu, and Lifang He. Contrastive multi-modal clustering. *arXiv:2106.11193*, 2021.
- Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proc. CVPR*, 2016.
- Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Proc. NeurIPS*, pages 3320–3328, 2014.
- Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv:2103.03230*, 2021.
- M. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

Bibliography

- Xiaohang Zhan, Jiahao Xie, Ziwei Liu, Yew-Soon Ong, and Chen Change Loy. Online deep clustering for unsupervised representation learning. In *Proc. CVPR*, pages 6688–6697, 2020.
- Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *Proc. CVPR*, pages 2547–2555, 2019.
- Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. ECCV*, pages 649–666. Springer, 2016.
- Richard Zhang, Phillip Isola, and Alexei A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proc. CVPR*, 2017.
- Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, 2018.
- Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *Proc. ICCV*, 2019.
- Evgenii Zheltonozhskii, Chaim Baskin, Alex M. Bronstein, and Avi Mendelson. Self-supervised learning for large-scale unsupervised image clustering, 2020.
- Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017.
- Benjin Zhu, Junqiang Huang, Zeming Li, Xiangyu Zhang, and Jian Sun. Eqco: Equivalent rules for self-supervised contrastive learning. *arXiv:2010.01929*, 2020.
- Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proc. CVPR*, 2020.

Appendices

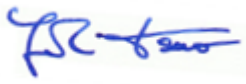
Authorship statements

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

Title of Paper	A critical analysis of self-supervision, or what we can learn from a single image
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Accepted at ICLR 2020. Yuki M. Asano, Christian Rupprecht, Andrew Zisserman, Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. ICLR 2020.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>Y.A. ideated and developed the initial idea, C.R. and A.V. advised. Y.A. ran the experiments. Figures were generated by Y.A. All authors wrote and edited the paper text.</i>		
Signature 	Date	07 / 06 / 2021	

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature 	Date	8 June 2021

Title of Paper	Self-labelling via simultaneous clustering and representation learning
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Accepted at ICLR 2020. Yuki M. Asano, Christian Rupprecht, Andrew Zisserman, Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. ICLR 2020.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>Y.A. ideated and developed the initial idea, C.R. and A.V. advised. Y.A. ran the experiments. C.R. developed the interactive website. Figures were generated by Y.A. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature		Date 8 June 2021

Title of Paper	On Compositions of Transformations in Contrastive Self-Supervised Learning
Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	submitted to: International Conference of Computer Vision (ICCV) 2021 Mandela Patrick*, Yuki Asano*, Polina Kuznetsova, Ruth Fong, João Henriques, Geoffrey Zweig, Andrea Vedaldi. On Compositions of Transformations in Contrastive Self-Supervised Learning. ArXiv.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>M.P and Y.A. ideated, prototyped and developed the initial idea. M.P. and Y.A. designed the experiments and a majority of them were run by M.P. Figures were generated by Y.A. and M.P. and J.H., P.K. ran a large-scale experiment on IG65M. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

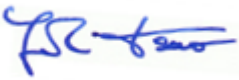
Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:		
Professor Andrea Vedaldi		
Supervisor comments		
The description of the contributions is accurate.		
Signature	Date	
		8 June 2021

Title of Paper	Labelling Unlabelled Videos from Scratch Using Multi-Modal Self-Supervision
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Yuki Asano*, Mandela Patrick*, Christian Rupprecht and Andrea Vedaldi. Labelling Unlabelled Videos from Scratch Using Multi-Modal Self-Supervision. Advances of Neural Information Processing Systems (NeurIPS) 2020.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>M.P and Y.A. developed and prototyped the initial idea. Y.A. and M.P. ran and designed experiments and generated the figures. C.R. developed the webtool for visualisation and aided in fleshing out the method. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

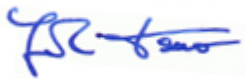
Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature		Date 8 June 2021

Title of Paper	Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning
Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	submitted to International Conference of Computer Vision (ICCV) 2021 Mandela Patrick*, Yuki Asano*, Po-Yao Huang*, Ishan Misra, Florian Metze, João Henriques, Andrea Vedaldi. Space-Time Crop & Attend: Improving Cross-modal Video Representation Learning. ArXiv.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>M.P and Y.A. ideated, prototyped and developed the initial idea. M.P. and Y.A. designed the experiments and a majority of them were run by M.P and P.H.. Figures were generated by Y.A. and M.P.. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

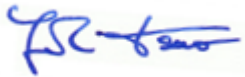
Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title:			
Professor Andrea Vedaldi			
Supervisor comments			
The description of the contributions is accurate.			
Signature		Date	8 June 2021

Title of Paper	Support-Set Bottlenecks for Video-Text Representation Learning
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Mandela Patrick*, Po-Yao Huang*, Yuki Asano*, Florian Metze, Alexander Hauptmann, João Henriques, Geoffrey Zweig, Andrea Vedaldi. Support-Set Bottlenecks for Video-Text Representation Learning. International Conference of Learning Representations (ICLR) 2021.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>M.P, P.H. and Y.A. ideated, prototyped and developed the initial idea. M.P. and P.H. designed and ran the experiments. Figures were generated by Y.A., P.H., M.P. and J.H. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

Supervisor Confirmation

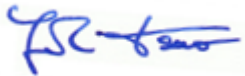
By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature		Date 8 June 2021

Title of Paper	Keeping Your Eye on the Ball: Trajectory Attention in Video Transformers
----------------	--

Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Submitted to Advances in Neural Information Processing Systems (NeurIPS) 2021 Mandela Patrick*, Dylan Campbell*, Yuki Asano*, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, João Henriques. Keeping Your Eye on the Ball: Trajectory Attention in Video Transformers. ArXiv.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>M.P, D.C., Y.A., J.H. and A.V. ideated, prototyped and developed the initial idea. M.P. designed and ran the experiments. Figures were generated by Y.A. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

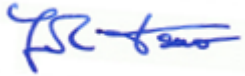
Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi			
Supervisor comments			
The description of the contributions is accurate.			
Signature		Date	8 June 2021

Title of Paper	Self-supervised object detection from audio-visual correspondence
Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Submitted to International Conference on Computer Vision (ICCV) 2021 Triantafyllos Afouras*, Yuki M. Asano*, Francois Fagan, Andrea Vedaldi, Florian Metze. Self-supervised object detection from audio-visual correspondence. ArXiv.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>T.A. and Y.A. ideated and developed the initial idea. T.A. ran the experiments. Figures were generated by Y.A. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature		Date 8 June 2021

Title of Paper	PASS: An ImageNet replacement for self-supervised pretraining without humans
Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Submitted to NeurIPS Datasets Track(NeurIPS) 2021 Yuki M. Asano, Christian Rupprecht, Andrew Zisserman, Andrea Vedaldi. PASS: An ImageNet replacement for self-supervised pretraining without humans. OpenReview.

Student Confirmation

Student Name:	Yuki Asano		
Contribution to the Paper	<i>Y.A. ideated and developed the initial idea. Y.A. ran the experiments. C.R. ran the human verification pipeline. Figures were generated by Y.A. All authors wrote and edited the paper text.</i>		
Signature		Date	07 / 06 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Andrea Vedaldi		
Supervisor comments The description of the contributions is accurate.		
Signature		Date 8 June 2021