Image Segmentation with Minimal Human Supervision



Gyungin Shin

St Hugh's College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2024

Abstract

Image segmentation is an indispensable part of computer vision due to its interpretative nature. However, training a robust segmenter presents significant challenges due to the costly labour required to acquire pixel-level labels. This thesis aims to explore various methodologies for learning strong visual representations for segmentation with minimal or no manual supervision.

The structure of this thesis is divided into three parts: (i) Efficient Weakly-supervised Learning for Semantic Segmentation; (ii) Semantic and Instance Segmentation without Manual Annotations; and (iii) Self-supervised Learning for Class-agnostic Image Segmentation.

In the first part, we demonstrate that considering a model's uncertainty in predictions can significantly reduce the number of annotations required to train a segmenter that performs nearly as well as a fully-supervised one. This finding underscores the critical role of pixels around semantic boundaries in helping the model to learn representations suitable for segmentation.

In the second part, we propose a straightforward retrieve and co-segmentation mechanism for semantic segmentation, which mimics how humans learn the visual features of new object categories and enables a segmenter to classify the concept of the object without manual supervision. We also illustrate how this mechanism can be enhanced by substituting the co-segmentation component with a more advanced object segmenter and how it can be extended to instance segmentation.

For the final part, we demonstrate how the strong visual correspondences provided by modern self-supervised models can be effectively utilised to segment an object without any prior knowledge of the object's concept.

Keywords – image segmentation, deep learning, efficient weakly-supervised learning, self-supervised learning

Declaration

This thesis is submitted to the Department of Engineering Science, The University of Oxford, in fulfilment 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.

Gyungin Shin, August 2024.

Acknowledgements

Over the course of my DPhil programme, I have been extraordinarily fortunate to have received invaluable guidance from Samuel Albanie, Weidi Xie, and, in my final year, Andrew Zisserman. In particular, I am deeply grateful to Samuel for his responsible supervision, to Weidi for his astute insights, and to Andrew for his compassionate support during challenging times. Their collective wisdom has not only guided, enlightened, and supported me, but has also been fundamental to my development as a researcher.

I extend my gratitude to Christian Rupprecht and Yuki Asano for their roles as exceptional examiners of this thesis. Their thoughtful suggestions and insightful comments have greatly enhanced the quality of this work.

I thank the Cambridge Service for Data Driven Discovery (CSD3), operated by the University of Cambridge Research Computing Service, for providing the computational resources. Additionally, I thank Dell EMC and Intel for their support, the Engineering and Physical Sciences Research Council for the Tier-2 funding (capital grant EP/T022159/1), and the Science and Technology Facilities Council for the DiRAC funding.

I wish to thank all members of VGG for their consistent friendliness and helpfulness at all times. It has been an honour to be part of such a group where one can expect not only professionalism but also kindness from everyone.

I would like to extend my thanks to several colleagues: Tengda Han for altruistically giving me advice for my internship interview and for providing this thesis template. My gratitude also goes to Lily Momeni and Sagar Vaze for their warm welcome into VGG during my first year. A special thanks goes to Laurynas Karazija for passionate discussions about unsupervised image/video segmentation. In addition, I extend my heartfelt thanks to Yash Bhalgat, Jaesung Huh, and Guanqi Zhan for being sincere friends.

To my dear friends outside VGG, I am grateful to Yibo Wang, Mary Zhou, Lily Su, Jinghan Zheng, Huinmin Hu, Guanyu Dai, Yihan Jin, and Xinyi Shen for being such wonderful friends and creating joyful memories in the UK. To Dylan Li, for an invaluable

friendship that began as a small mustard seed and has blossomed into a large tree. I would also like to thank Freda and John Cammack, who graciously and selflessly helped me settle into Oxford, and the members of Belsyre Court for making it such a lovely neighbourhood.

To my old friends in South Korea—Chang Kyu Lee, Jeongdo Park, and Hyeokjun Shin—I cannot express my gratitude enough for always being there. To Youngrae Cho, thanks are due for maintaining contact despite a busy schedule.

To my families in China and South Korea, I deeply appreciate your unconditional love and prayers, even though I have been sparing with my affection. Without your constant care and support, I wouldn't be able to finish this journey. To my late grandfather, thank you for leaving me with cherished memories and a clear vision of what I want to achieve in life.

Dear my love, Zheng Fang, no words can fully capture my gratitude for your presence in my life. Yet, I must express:

Since I've been with you, the universe no longer seems so vast and cold. Even in the furthest reaches, where space expands beyond the speed of light, there persists the warmth of your love.

Lastly, dear God, thank You for filling each day with love, manifest through cherished ones I scarcely deserve. Each morning, let me remember this grace, loving You by loving them as deeply as I love myself.

Contents

1 Introduction and Background **10**

1.1 Key ideas and literature review 11

1.2 Thesis outline and contributions 15

1.3 Publications 17

I Efficient Weakly-supervised Learning for Semantic Segmentation **20**

2 All You Need Are a Few Pixels: Semantic Segmentation with PixelPick **21**

2.1 Introduction 23

2.2 Related work 25

2.3 Method 28

2.3.1 PixelPick framework 28

2.3.2 PixelPick annotation tool 32

2.4 Experiments 33

2.4.1 Experimental setup 34

2.4.2 Ablation studies 34

2.4.3 Comparison to state-of-the-art methods 39

2.4.4 Practical deployment 40

2.5 Conclusion 41

II Semantic and Instance Segmentation without Manual Annotations 44

3 ReCo: Retrieve and Co-segment for Zero-shot Transfer 45

3.1 Introduction . . . 47
3.2 Related work . . . 49
3.3 Method . . . 52
3.3.1 Unsupervised semantic segmentation with language-image pre-training . . . 52
3.3.2 Retrieve and Co-segment (ReCo) . . . 54
3.3.3 Language-guided co-segmentation and context elimination . . . 57
3.3.4 ReCo+: Fine-tuning ReCo with pseudo-labels . . . 58
3.4 Experiments . . . 59
3.4.1 Datasets . . . 59
3.4.2 Implementation details . . . 60
3.4.3 Ablation studies . . . 61
3.4.4 Comparison to state-of-the-art unsupervised methods . . . 64
3.4.5 Segmenting rare concepts . . . 65
3.5 Limitations . . . 66
3.6 Broader impact . . . 66
3.7 Conclusion . . . 67

4 NamedMask: Distilling Segmenters from Complementary Foundation Models 70

4.1 Introduction . . . 72
4.2 Related work . . . 74
4.3 Method . . . 75
4.3.1 Task formulation and terminology . . . 76
4.3.2 NamedMask . . . 77

4.4 Experiments . . . 79
4.4.1 Datasets . . . 79
4.4.2 Implementation details . . . 81
4.4.3 Ablation study . . . 82
4.4.4 Comparison to state-of-the-art methods . . . 85
4.5 Limitations . . . 89
4.6 Broader impact . . . 90
4.7 Conclusion . . . 90

5 Zero-shot Unsupervised Transfer Instance Segmentation 92
5.1 Introduction . . . 94
5.2 Related work . . . 96
5.3 Method . . . 100
5.3.1 Problem scenario . . . 100
5.3.2 Pseudo-label training . . . 101
5.3.3 Architecture . . . 102
5.4 Experiments . . . 105
5.4.1 Implementation details . . . 105
5.4.2 Ablation study . . . 107
5.4.3 Main results . . . 109
5.4.4 Limitations . . . 112
5.5 Broader impact . . . 113
5.6 Conclusion . . . 114

III Self-supervised Learning for Class-agnostic Image Segmentation 116

6 Unsupervised Salient Object Detection with Spectral Cluster Voting 117

6.1 Introduction . . . 119
6.2 Related work . . . 121
6.2.1 Self-supervised representation learning . . . 121
6.2.2 Unsupervised saliency detection . . . 122
6.3 Method . . . 124
6.3.1 Problem formulation . . . 125
6.3.2 Segmentation with spectral clustering . . . 125
6.3.3 Supervision with pseudo-mask from spectral clusters . . . 126
6.4 Experiments . . . 130
6.4.1 Datasets . . . 130
6.4.2 Implementation details . . . 131
6.4.3 Ablation study . . . 131
6.4.4 Comparison to state-of-the-art unsupervised saliency detection methods . . . 136
6.4.5 Broader impact . . . 137
6.5 Conclusion . . . 137

7 ProMerge: Prompt and Merge for Unsupervised Instance Segmentation 139

7.1 Introduction . . . 141
7.2 Related work . . . 143
7.2.1 Self-supervised visual representation learning . . . 143
7.2.2 Unsupervised single object detection and segmentation . . . 144
7.2.3 Unsupervised instance segmentation . . . 145
7.3 Method . . . 145
7.3.1 Problem formulation . . . 146
7.3.2 ProMerge: Prompt and Merge . . . 146
7.3.3 Background-based mask pruning . . . 148

7.3.4 ProMerge+: Training an object detector with pseudo-labels from ProMerge . . . 151
7.4 Experiments . . . 151
7.4.1 Implementation details . . . 151
7.4.2 Main results . . . 153
7.4.3 Ablation study . . . 154
7.5 Discussion . . . 159
7.6 Conclusion . . . 159

8 Discussion **162**
8.1 Summary . . . 162
8.2 Future work . . . 164
8.3 Conclusion . . . 165

References **166**

A Statement of Authorship **193**

Chapter 1

Introduction and Background

Image segmentation, which aims to differentiate regions for individual entities in an image, is a pivotal component in computer vision—it facilitates visual interpretation of how a computer vision system recognises entities, thereby enabling interaction between humans and the system. For this reason, segmentation is not only a subject of active research within the computer vision community but also finds extensive applications across various industries, including autonomous driving [Cordts et al. 2016], satellite imaging [Demir et al. 2018], and medical analysis [Tobon-Gomez et al. 2015; Milletari et al. 2016; He et al. 2023]. Despite its importance, however, training such a system poses a significant challenge: it requires a vast quantity of pixel-level annotations, manually labelled by humans.

In this thesis, we raise the question, "How can we train a segmenter label-efficiently and, if possible, without human supervision?" and explore a variety of approaches to address this inquiry. Particularly, we cover three aspects of this question: (i) efficient weakly-supervised learning for semantic segmentation; (ii) semantic and instance segmentation without manual annotations; and (iii) self-supervised learning for class-agnostic image segmentation.

We begin by introducing the key ideas behind the work along with the literature review in Section 1.1, then we introduce the three themes of the thesis and their respective contributions in Section 1.2. In Section 1.3, we provide a full list of the works included in the thesis.

1.1 Key ideas and literature review

Humans are extremely adept at identifying the boundary of an object, even if they have never encountered it before. Conversely, even state-of-the-art fully-supervised segmentation models [Kirillov et al. 2023; Ke et al. 2023; Ravi et al. 2024], trained with millions of human-labelled masks, frequently struggle to delineate an accurate boundary when faced with a complex object. This leads to an intriguing question: "Are there effective ways to train a robust segmentation system, while reducing or even eliminating the need for large-scale human-laboured masks?"

To address this, we draw inspiration from how humans learn to segment entities, focusing on uncertainty, common visual features among objects of the same category, and strong affinities between local patches as follows.

Leveraging uncertainty in segmentation models. Not all pixels that comprise an entity hold equal significance. The human visual system instinctively overlooks variations in colour and texture, such as different parts of a car, by grouping adjacent pixels together. Meanwhile, it processes other pixels with greater precision, even though they may seem similar to other parts of the entity (e.g., the boundaries of an object, whose RGB values are often nearly identical to those of neighbouring inner pixels). Interestingly, in a modern deep-learning-based segmenter, this can be observed through uncertainty—when a model makes predictions on pixels near a boundary region, its uncertainty is significantly higher than in a central region.

Building on this observation, in Chapter 2, we investigate the concept of "active learning" within the context of semantic segmentation, which essentially involves a subset selection problem: determining the most informative set of samples to label within a fixed budget [Atlas et al. 1989; Seung et al. 1992; Lewis and Gale 1994; Freytag et al. 2014; Yoo and Kweon 2019]. In semantic segmentation, the goal is to identify the subset of labelled pixels that yields the lowest generalisation error when used to train a supervised model.

For this, different strategies have been employed: selecting superpixels that induce maximum label change using a CRF under weak supervision [Vezhnevets et al. 2012], applying geometric constraints [Konyushkova et al. 2015; Mosinska-Domanska et al. 2016], and propagating foreground masks across large image collections [Jain and Grauman 2016]. In medical image segmentation, methods like FCNs [Long et al. 2015] combined with bootstrapping [Yang et al. 2017] and U-Nets [Ronneberger et al. 2015] using dropout-based Monte Carlo uncertainty estimates [Gorriz et al. 2017] have been utilised to guide label acquisition through uncertainty sampling.

The approach of learning an estimator for challenging regions [Yoo and Kweon 2019] has shown success in selecting images that require dense labelling for semantic segmentation [Xie et al. 2020]. Moreover, adversarial learning has been employed to align labelled and unlabelled image distributions by querying labels [Sinha et al. 2019].

More relevant to our work in Chapter 2, prior research has explored *region-based* sampling strategies for semantic segmentation, utilising reinforcement learning [Casanova et al. 2020], enforcing equivariance constraints [Golestaneh and Kitani 2020], and creating learned estimators for labelling cost [Mackowiak et al. 2018]. Unlike these methods, our approach presents a more efficient active learning paradigm for segmentation, emphasising training models with only sparse pixel annotations, thereby removing the need for the localisation aspect of the annotation task.

This notion of using uncertain pixels as the basic unit for human annotation is adopted and further developed in the context of combining semi-supervised learning and active learning for semantic segmentation [Chakravarthy et al. 2022] and instance segmentation [Tang et al. 2022].

Conceptual visual feature abstraction via image retrieval and co-segmentation. The visual features of an entity is ultimately defined by its concept, subconsciously formed in a human's mind through exposure to numerous instances

of a particular concept, thereby abstracting away their differences. For instance, although every car appears different, when asked to draw one, we typically sketch four wheels—two at the front and two at the back—with a body over the wheels. This communal visual representation of a concept similarly influences the identification of boundaries for an instance belonging to that concept. For this reason, the idea of co-segmentation, which involves segmenting entities of the same type that appear across various images, has been the subject of extensive study using traditional computer vision techniques [Rother et al. 2006; Mukherjee et al. 2009; Vicente et al. 2010; Batra et al. 2010; Joulin et al. 2010; Vicente et al. 2011]. It has recently been explored further with deep learning, utilising shared encoder networks [Chen et al. 2018a; Li et al. 2019; Banerjee et al. 2019; Liu et al. 2020], iterative refinement [Zhang et al. 2021a], and weak supervision [Hsu et al. 2018]. However, these prior works do not address classification of concepts—assuming a collection of images of a certain concept is given, their approaches focus on predicting foreground masks for the commonly appearing regions within the collection.

In Chapter 3, we make the first attempt to jointly solve classification of a concept and co-segmentation, thereby enabling open-vocabulary semantic segmentation. Specifically, we do so without any expensive human-labelled annotations such as pixel-level labels for segmentation. We realise this through two steps: (i) image retrieval with a pre-trained visual-language model and (ii) co-segmentation with shape-aware visual features. These steps effectively mimic how a human learns the visual features of a concept they have never previously encountered, thus allow for generalising to any category that the retrieval system can correctly retrieve.

The idea of our approach has been studied and further developed, notably in Karazija et al. [2024], where a text-to-image diffusion model is used to create a support set of images for a given concept, instead of the image retrieval step.

Strong local correspondences offered by self-supervised features. Modern self-supervised features exhibit strong visual correspondences between local patches [He et al. 2020; Chen et al. 2020; Caron et al. 2020; Wang et al. 2021;

Caron et al. 2021; Oquab et al. 2023]. In other words, similarly to how humans group neighbouring pixels of an object despite their differences (*e.g.*, colour and intensity), these features effectively identify and associate similar patterns and textures across varied contexts, which can serve as an effective base for sophisticated segmentation methods.

Based on these observations, extensive research has been conducted in the area of unsupervised object segmentation [Siméoni et al. 2021; Melas-Kyriazi et al. 2022b; Wang et al. 2022b; Melas-Kyriazi et al. 2022a; Li et al. 2023; Siméoni et al. 2023a; Siméoni et al. 2024]. An early study [Siméoni et al. 2021] extracts features from a self-supervised network [Caron et al. 2021] and isolates the foreground by initially identifying a seed patch with the lowest number of positive correlations with other patches. Another line of approaches [Melas-Kyriazi et al. 2022a; Wang et al. 2022b] employs spectral clustering on self-supervised features to differentiate the foreground from the background. In Chapter 6, we build up on the latter by applying spectral clustering to an ensemble of self-supervised features (e.g., SwAV [Caron et al. 2020], DINO [Caron et al. 2021], and MoCo [He et al. 2020]) to acquire a sufficient support set of foreground masks and deciding a single representative salient mask within the support set via a winner-takes-all voting strategy.

Self-supervised features have also been used to address a more challenging task of unsupervised instance segmentation, which aims to segment multiple objects in an image [Van Gansbeke et al. 2022; Wang et al. 2022a; Wang et al. 2023b]. The dominant type of approaches has been pseudo-label training which involves first obtaining initial masks from object-centric images and training a segmenter using the initial masks as pseudo-masks. Notably, inspired by the success of applying spectral clustering to self-supervised features for object segmentation [Melas-Kyriazi et al. 2022a; Wang et al. 2022b; Shin et al. 2022a], a simple application of repeated spectral clustering to acquire multiple object masks per image was explored in [Wang et al. 2023b]. In Chapter 7, we also introduce a method for

unsupervised instance segmentation, which forgoes the compute-intensive spectral clustering.

1.2 Thesis outline and contributions

This section outlines the subsequent chapters of the thesis and summarises the main contributions. The thesis is organised into three themes: (i) efficient weakly-supervised learning for semantic segmentation; (ii) semantic and instance segmentation without manual annotations; and (iii) self-supervised learning for class-agnostic image segmentation.

Part I: Efficient weakly-supervised learning for semantic segmentation

In this theme, we explore an approach to reducing the number of human labels required to train a robust segmenter by introducing the concept of a model's "uncertainty" in its predictions. Specifically, the segmenter requests labels only for pixels where it is uncertain about its prediction, and a human annotator provides labels for those specific pixels. This method demonstrates that, contrary to the standard practice of using labels for every pixel in an image, a segmenter as strong as a fully-supervised one can be achieved with just a few human labels per image, which is more than 100× fewer than prior state-of-the-art weakly-supervised methods. Interestingly, this approach reveals that regions deemed uncertain by the segmenter are often closely aligned with the semantic boundaries of entities, suggesting that these boundaries are crucial areas to focus on as they delineate the distinctions between different concepts.

Part II: Semantic and instance segmentation without manual annotations

In this section, we delve deeper and explore semantic and instance segmentation methods that do not rely on manual annotations.[1] This raises a fundamental question: "How can we inject semantics into the representations of a segmenter without any human supervision?" We find an answer to this question by utilising a vision-language model (VLM) trained in a webly-supervised manner (*i.e.*, CLIP [Radford et al. 2021]).

In Chapter 3, we propose the novel idea of leveraging the retrieval capability of a pre-trained VLM from a large-scale unlabelled image dataset, in order to form an archive of images for a given concept (*i.e.*, class). For each concept, we apply a co-segmentation algorithm to extract visual features from objects that frequently appear across the images in the archive and use these distinctive features for different categories as prototypes. Entities in new images are then segmented based on the similarities between their visual features and these prototypes. Despite its simplicity, we demonstrate that this retrieve-and-co-segment approach not only shows competitive performance in semantic segmentation compared to previous state-of-the-art methods but also can generalise to extremely rare categories such as "Antikythera mechanism."[2]

In the following chapters, we build upon the method proposed in Chapter 3 and replace the fragile co-segmentation algorithm with a separate unsupervised object segmentation model, which greatly boosts the semantic segmentation performance (Chapter 4). Then, we unify the VLM and the separate object segmenter into a single network, while enabling it for instance segmentation in Chapter 5.

[1] By manual annotations, we refer to labels that are purposefully obtained through human labour for segmentation tasks.

[2] The oldest known example of an analogue computer [Efstathiou and Efstathiou 2018]

Part III: Self-supervised learning for class-agnostic image segmentation

In Parts I and II, segmenters are tasked with differentiating entities based on their semantics, *i.e.*, the concept of an object. While recognising the common visual features shared between instances of the same category (*e.g.*, the black and white stripes of zebras) can be a significant clue for segmenting them, it is also worth noting the generalisation ability of the human visual system that excels at segmenting entities without prior knowledge of their categories. In Part III, we explore how to replicate this ability through strong visual correspondences provided by modern self-supervised features. Specifically, in Chapter 6, inspired by the observation that clustering self-supervised features shows great potential for object segmentation, we leverage an ensemble of different self-supervised features to obtain a single mask for the salient object within an image. In Chapter 7, we take a step further and tackle class-agnostic unsupervised instance segmentation (*i.e.*, multiple instance masks per image) by proposing a simple yet effective prompt-and-merge approach.

1.3 Publications

Chapter 2 to Chapter 7 each contains a research paper. For each paper, we also provide a statement of authorship in Appendix A. The papers included in the thesis are listed below.

Chapter 2: "All you need are a few pixels: semantic segmentation with PixelPick" **Gyungin Shin**, Weidi Xie, Samuel Albanie. In the Workshop on Interactive Labeling and Data Augmentation for Vision (Best Paper Award), *International Conference on Computer Vision*, 2021.

Chapter 3: "ReCo: Retrieve and Co-segment for Zero-shot Transfer" **Gyungin Shin**, Weidi Xie, Samuel Albanie. In *Neural Information Processing Systems*, 2022.

Chapter 4: "NamedMask: Distilling Segmenters from Complementary Foundation Models" **Gyungin Shin**, Weidi Xie, Samuel Albanie. In the Workshop on Learning with Limited Labelled Data for Image and Video Understanding, *Computer Vision and Pattern Recognition*, 2023

Chapter 5: "Zero-shot Unsupervised Transfer Instance Segmentation" **Gyungin Shin**, Samuel Albanie, Weidi Xie. In the Workshop on Learning with Limited Labelled Data for Image and Video Understanding (Best Paper Award & Oral Presentation), *Computer Vision and Pattern Recognition*, 2023

Chapter 6: "Unsupervised Salient Object Detection with Spectral Cluster Voting" **Gyungin Shin**, Samuel Albanie, Weidi Xie. In the Workshop on Learning with Limited Labelled Data for Image and Video Understanding, *Computer Vision and Pattern Recognition*, 2022.

Chapter 7: "ProMerge: Prompt and Merge for Unsupervised Instance Segmentation" Dylan Li[*], **Gyungin Shin**[*✉]. In *European Conference on Computer Vision*, 2024.

Publications not included

[1]**"arXiVeri: Automatic Table Verification with GPT"** **Gyungin Shin**, Weidi Xie, Samuel Albanie. In the AI for Science Workshop, *Neural Information Processing Systems*, 2023.

[2]**"Tails Tell Tales: Chapter-Wide Manga Transcriptions with Character Names"** Ragav Sachdeva, **Gyungin Shin**[†], Andrew Zisserman. In *Asian*

Conference on Computer Vision, 2024.

[1]This publication focuses on utilising a large language model to verify accurate transcription of numerical data in scientific documents, which is not related to this thesis topic and is therefore excluded.

[2]This work proposes a vision model for understanding manga, along with relevant datasets, which is not closely related to the thesis topic and is therefore excluded.

[*]Equal contribution [†]Core contribution [✉]Corresponding author

Part I

Efficient Weakly-supervised Learning for Semantic Segmentation

Chapter 2

All You Need Are a Few Pixels: Semantic Segmentation with PixelPick

This work has been accepted for publication and awarded the Best Paper Award at the Interactive Labeling and Data Augmentation for Vision Workshop, International Conference on Computer Vision (ICCV), 2021.

All You Need Are a Few Pixels: Semantic Segmentation with PixelPick

Gyungin Shin[1] Weidi Xie[1] Samuel Albanie[1,2]

[1]Visual Geometry Group, University of Oxford

[2]Department of Engineering, University of Cambridge

https://www.robots.ox.ac.uk/~vgg/research/pixelpick

Abstract

A central challenge for the task of semantic segmentation is the prohibitive cost of obtaining dense pixel-level annotations to supervise model training. In this work, we show that in order to achieve a good level of segmentation performance, all you need are a few well-chosen pixel labels.

We make the following contributions: (i) We investigate the semantic segmentation setting in which labels are supplied only at sparse pixel locations, and show that deep neural networks can use a handful of such labels to good effect; (ii) We demonstrate how to exploit this phenomenon within an active learning framework, termed PixelPick, to radically reduce labelling cost, and propose an efficient "mouse-free" annotation strategy to implement our approach; (iii) We conduct extensive experiments to study the influence of annotation diversity under a fixed budget, model pretraining, model capacity and the sampling mechanism for picking pixels in this low annotation regime; (iv) We provide comparisons to the existing state of the art in semantic segmentation with active learning, and demonstrate comparable performance with up to two orders of magnitude fewer pixel

annotations on the CamVid, Cityscapes and Pascal VOC 2012 benchmarks; (v) Finally, we evaluate the efficiency of our annotation pipeline and its sensitivity to annotator error to demonstrate its practicality.

2.1 Introduction

The coupling of deep neural networks and large-scale labelled datasets has yielded significant progress on a host of core machine perception tasks. A key challenge of training these models is their need for considerable quantities of annotation, which can be prohibitively expensive to collect for applications that require either specialised annotators such as medical image diagnostics [Abramoff et al. 2016; Gulshan et al. 2016; Shih et al. 2019; Wang et al. 2020], or fine-grained labels, such as for detection and segmentation [Lin et al. 2014].

Semantic segmentation, in particular, has proven valuable for decision making in a variety of applications such as digital pathology [Tokunaga et al. 2019], remote sensing [Wurm et al. 2019] and autonomous driving [Yang et al. 2018]. However, its requirement of per-pixel annotations raises significant scalability challenges—on average more than 1.5 hours of annotation and quality control was required for each image in the Cityscapes segmentation dataset [Cordts et al. 2016].

The objective of this work is to propose a simple yet effective approach for training a good semantic segmentation model at minimal annotation cost. Our approach is motivated by three observations: (1) Within a given image, pixels exhibit significant spatial mutual information; (2) Deep neural networks possess a strong inductive bias that renders them appropriate for modelling these spatial dependencies [Ulyanov et al. 2018]; (3) Collecting mask, scribble or click annotations requires annotators to "localise and classify" using a mouse or trackpad. By contrast, assigning a class to a pixel proposal can be "mouse-free", requiring instead only a "classify" task without a localisation component (and which can be performed via a single key-press). The first two factors imply that densely labelling

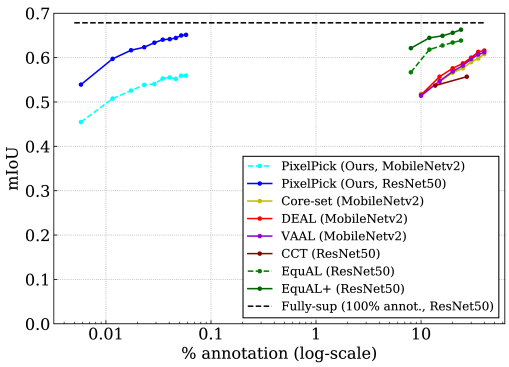


Figure 2.1: **All you need are a few pixels:** We show that deep neural networks can obtain remarkable performance with just a handful of labelled pixels per image whose spatial coordinates are proposed by the model, rather than the human annotator. We compare our approach, PixelPick, with existing active learning and semi-supervised approaches on the CamVid dataset [Brostow et al. 2009] (see Section 2.4 for further details).

all pixels in images may be highly redundant, while the third suggests the possibility of designing an efficient sparse pixel labelling strategy. Several questions then arise: *how many sparse pixel labels are needed to achieve good performance? how should those pixel locations be selected?* and *how can the selected pixels be annotated efficiently?*

In this paper, we address these questions through the lens of *active learning* [Atlas et al. 1989; Seung et al. 1992]. In contrast to passive supervised learning (where the model is tasked to learn the mapping from a fixed set of input-output pairs), active learning considers a dynamic scenario in which a model can interactively request labels for the samples that it believes will be most useful for solving a given task. Our proposed PixelPick framework adopts this paradigm, learning a model for semantic segmentation by alternating between training on previously labelled pixels and requesting new labels.

We make the following contributions: (i) We study the problem setting in which labels are supplied at the level of sparse pixels and show that with only a small collection of such labels, modern deep neural networks can achieve good performance; (ii) We show how this phenomenon can be exploited with an efficient and practical "mouse-free" annotation strategy as part of a proposed PixelPick

active learning framework; (iii) We perform a series of experiments into factors that affect model performance in the low-annotation regime: annotation diversity, architectural choices and the design of the sampling mechanisms for selecting most useful pixels; (iv) We compare with other state of the art active learning approaches on standard segmentation benchmarks: CamVid, Cityscapes and Pascal VOC 2012, where we demonstrate comparable segmentation performance with significantly lower annotation budget (Fig. 2.1); (v) Lastly, we assess PixelPick from the perspective of practical deployment, assessing its annotation efficiency and robustness.

2.2 Related work

Our work is related to several themes of research that have sought to minimise labelling costs for semantic segmentation, as discussed next.

Weakly-supervised semantic segmentation. Many weak supervisory signals have been explored in the literature as a pragmatic compromise between fully supervised [Long et al. 2015] and fully unsupervised approaches to semantic segmentation [Ji et al. 2019]. These cues include scribbles [Lin et al. 2016], eye tracking [Papadopoulos et al. 2014], object pointing [Bearman et al. 2016; Qian et al. 2019; Cheng et al. 2021a], web-queried samples [Jin et al. 2017], bounding boxes [Dai et al. 2015; Khoreva et al. 2017; Song et al. 2019], extreme clicks for objects [Papadopoulos et al. 2017; Maninis et al. 2018] and image-level labels [Zhou et al. 2016; Wei et al. 2018; Fan et al. 2018]. Differently from these approaches, we gather labels at sparse pixel locations proposed by the model itself, rather than at locations selected by the annotator, and show that very few such annotations are needed for good performance.

Interactive annotation. There is rich body of computer vision literature considering the related problem of accelerating *interactive* annotation. The seminal

work of [Boykov and Jolly 2001] demonstrated how to exploit scribbles to indicate the foreground/background appearance model and leverage graph-cuts for segmentation [Boykov and Kolmogorov 2004]. This was later extended to the use of multiple scribbles on both object and background, applied to annotating objects in videos [Nagaraja et al. 2015]. [Rother et al. 2004] exploited 2D bounding boxes provided by the annotator and performed pixel-wise foreground/background labelling using EM. Recent work [Castrejon et al. 2017] tasks a model with sequentially producing the vertices of a polygon outlining an object, given an appropriate crop. As with the weakly-supervised signals described above, these methods are passive in the sense that the labelling process is driven by the human annotator, rather than the model.

Semi-supervised semantic segmentation. Inspired by classical self-labelling approaches which aim to leverage unlabelled data to improve a classifier [Scudder 1965; Yarowsky 1995], a number of semi-supervised approaches have been developed to make use of pseudo-labelling algorithms [Lee 2013] for semantic segmentation in a low-annotation regime. Consistency-based pseudo-labelling methods have recently demonstrated promising results, highlighting the important role of aggressive data augmentations when only a small number of densely annotated images or regions are available [Ouali et al. 2020a; French et al. 2019].

Our approach differs from theirs in several ways: (i) our model is trained from sparse pixel annotations, rather than a small number of densely labelled images, (ii) we employ active learning (samples are dynamically selected and queried for annotation by the model), which, as we show through experiments, brings additional improvements. We compare our approach quantitatively with theirs in Section 2.4.3.

Active learning for semantic segmentation. At its core, active learning is a set selection problem; the aim being to determine the most informative subset of samples to acquire labels for, given a labelling budget [Atlas et al. 1989; Seung et al.

1992; Lewis and Gale 1994; Freytag et al. 2014; Yoo and Kweon 2019]. In this case the maximally informative labelled-pixel subset is the one which yields the lowest generalisation error when used to train a supervised semantic segmentation model. Prior work targeting segmentation has investigated strategies to select superpixels that induce the maximum label change for a CRF on the training set by using weak (image-level category) supervision [Vezhnevets et al. 2012], incorporate geometric constraints [Konyushkova et al. 2015; Mosinska-Domanska et al. 2016] and propagate foreground masks to large-scale image collections [Jain and Grauman 2016]. For foreground segmentation of medical imagery, FCNs [Long et al. 2015] have been coupled with bootstrapping [Yang et al. 2017], and U-Nets [Ronneberger et al. 2015] with dropout-based Monte Carlo estimates of uncertainty [Gorriz et al. 2017] to drive label acquisition via uncertainty sampling. The strategy of learning an estimator for difficult regions [Yoo and Kweon 2019] has proven effective as a basis for selecting which images should be densely labelled for semantic segmentation [Xie et al. 2020]. Adversarial learning has also been explored to query labels that align distributions of labelled and unlabelled images [Sinha et al. 2019].

More closely related to our work, prior studies have considered *region-based* sampling strategies for semantic segmentation, employing reinforcement learning [Casanova et al. 2020], equivariance constraints [Golestaneh and Kitani 2020] and learned estimators of labelling cost [Mackowiak et al. 2018]. In contrast to these lines of research, our work aims to introduce a more efficient paradigm of active learning for segmentation, which is to train models by only querying sparse pixel annotations (removing the localisation component of the annotation task). Concurrent to our work, Cai et al. [Cai et al. 2021] explore the use of super-pixels with active learning. We compare our work with theirs and with other region-based sampling strategies noted above in Section 2.4.

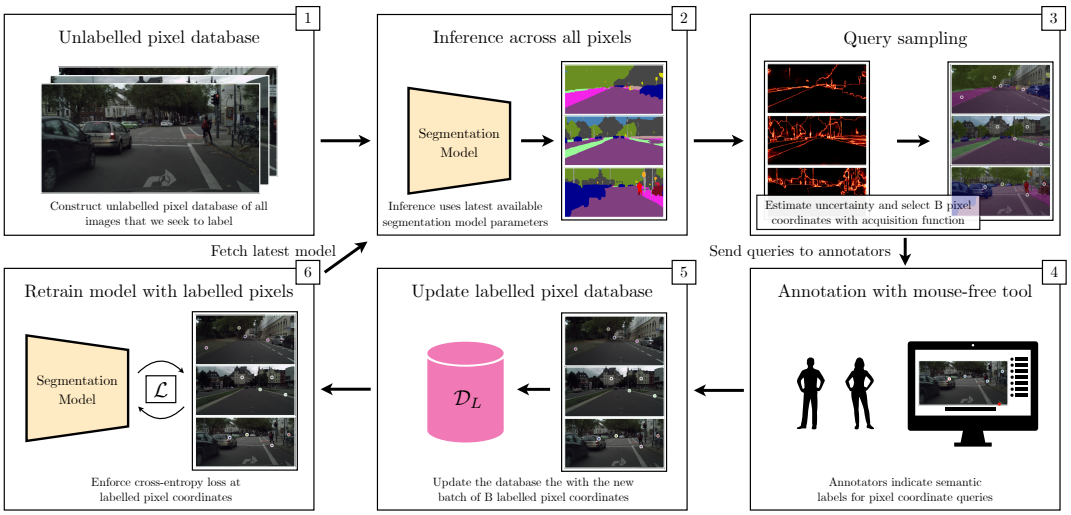


Figure 2.2: **Overview of the PixelPick active learning framework.** Given a database of unlabelled pixels of interest (1) each image is fed to a segmentation model to produce pixel-wise class probabilities (2), which are in turn passed to an acquisition function to estimate per-pixel uncertainties and select a batch of B pixels to be labelled (3). The queries are sent to annotators (4), and the resulting labels are added to the *labelled pixel database*, $\mathcal{D}_L$ (5). Finally, the segmentation model is retrained on the expanded database (6), before the cycle repeats. To bootstrap the process and train the initial segmentation model, we randomly sample B pixels and send them to be annotated. See text in Section 2.3 for further details.

2.3 Method

In this section, we describe the problem formulation and introduce our framework for pixel-level active learning semantic segmentation in Section 2.3.1. We then detail our mouse-free annotation tool to efficiently implement the framework in Section 2.3.2.

2.3.1 PixelPick framework

We seek to train a model for semantic segmentation with *pool-based active learning* [Settles 2009], in which we alternate between training a model on available annotation and requesting labels for unlabelled samples from an oracle (see Fig. 2.2).

More formally, let $\mathcal{X} \subset \mathbb{R}^{H \times W \times 3}$ denote the space of colour images and let

$\Phi(\cdot;\Theta) : \mathcal{X} \to \mathcal{Y}^{H\times W}$ represent a ConvNet with parameters Θ that maps a given image to a grid of elements in a C-class semantic label space (here $\mathcal{Y}$ denotes the $(C-1)$-simplex, i.e. $\mathcal{Y} = \{(p_1,\ldots,p_C) \in [0,1]^C : \sum_{p=1}^{C} p_i = 1, p_i \geq 0\}$). We assume access to an initial unlabelled pool of N images, $\mathcal{D}_U$, indexed by the $H \times W \times N$ pixel coordinate lattice, Ω, and an annotation database, $\mathcal{D}_L^0$, initialised to an empty state.

At the k^{th} round of learning, a batch of $B \in \mathbb{N}$ pixel coordinates, $\omega_k \subset \Omega$, are sampled by an acquisition function, $\mathcal{A}$, using the predictions of the model trained in the previous round, $\Phi(\cdot;\Theta_{k-1})$, on the unlabelled pool $\mathcal{D}_U$, *i.e.*, $\mathcal{A}(\mathcal{D}_U, \Phi(\cdot;\Theta_{k-1})) = \omega_k \subset \Omega$. The sampled pixel coordinates ω_k are then sent to an oracle for labelling to produce a corresponding set of one-hot labels $\{y_u \in \mathcal{Y} : u \in \omega_k\}$ that are added to the latest version of the annotation database, $\mathcal{D}_L^{k-1}$. Finally, the model is retrained on this expanded database, $\mathcal{D}_L^k = \cup_{i=1}^{k}\{(u, y_u) : u \in \omega_i\}$ (comprising all annotations gathered so far), to produce a new model, $\Phi(\cdot;\Theta_k)$, and the process is repeated. We term this framework PixelPick due to its emphasis on selecting appropriate pixels for annotation. The two components of the framework, namely *retraining the segmentation model* and *sampling new pixel coordinates*, are discussed next.

Retraining the segmentation model. At round k of the active learning algorithm, we solve for parameters Θ_k by minimising a cross-entropy loss at each labelled pixel coordinate present in the current annotation database $\mathcal{D}_L^k$:

$$\Theta_k = \underset{\Theta}{\operatorname{argmin}}\, \mathcal{L}(\Theta, \mathcal{D}_L^k) \quad \text{where} \tag{2.1}$$

$$\mathcal{L}(\Theta, \mathcal{D}_L^k) = -\frac{1}{|\mathcal{D}_L^k|} \sum_{(u,y_u)\in\mathcal{D}_L^k}^{n} \sum_{c=1}^{C} y_u(c) \cdot \log(\hat{y}_u(c)). \tag{2.2}$$

In the expression above, $y_u(c)$ and and $\hat{y}_u(c)$ denote the c^{th} channel of the oracle-provided label and corresponding model prediction at pixel coordinate u, respectively.

Sampling new pixel coordinates for labelling. The objective of the acquisition function, $\mathcal{A}$, is to sample the B pixel locations at round k that maximise improvement in segmentation performance for the current model $\Phi(\cdot;\Theta_{k-1})$. Functionally, it acts by examining the predictions of $\Phi(\cdot;\Theta_{k-1})$ across all candidate pixel coordinates among the unlabelled pool $\mathcal{D}_u$ and sampling B such coordinates according to a specified criterion.

Discussion. The distinction between sampling contiguous spatial patches for annotation (*e.g.*, grids of 128x128 pixels or larger as considered in prior work [Mackowiak et al. 2018; Golestaneh and Kitani 2020; Casanova et al. 2020]), and sampling *individual pixel coordinates*, as proposed within the PixelPick framework, is a subtle but important one. It has two key benefits. The first, as noted in Section 2.1, is that it allows us to leverage the powerful inductive biases provided by deep neural network architectures that render them well suited to modelling spatial dependencies in natural images [Ulyanov et al. 2018]. The second is a practical one: by providing annotators with pixel coordinate proposals, the labelling process is transformed from a "localise and classify" task (required when segmenting semantic regions and typically performed with a mouse or trackpad), into simply a "classify" task in which a class label is assigned to a coordinate proposal, and which can often be performed with a single key-press. We validate both claims through experiments in Section 2.4, where we show that (i) deep neural networks achieve strong segmentation performance at extremely sparse annotation levels, and (ii) "mouse-free" annotation can be performed very efficiently.

Acquisition functions. The design of the specific criteria employed by the acquisition function has been the subject of considerable attention in the active learning literature (see [Settles 2009] and [Ren et al. 2020] for surveys of classical and recent approaches, respectively). Since the focus of our work is not the design of another criterion, but rather on the effectiveness of individual pixels as the base unit for annotation, we consider several existing approaches based on the

framework of *uncertainty sampling* [Lewis and Gale 1994] that have been noted as effective in the literature, discussed next.

The *Least Confidence* acquisition strategy [Lewis and Catlett 1994; Culotta and McCallum 2005] draws, at each iteration, the pixel coordinate for which the model has *least confidence* in its *most likely* class label. The *Margin Sampling* strategy [Scheffer et al. 2001] looks for samples that exhibit the smallest difference (i.e. lowest "margin") between the first and second most probable labels. Finally, the *Entropy Sampling* strategy aims to select the pixel coordinate with the greatest conditional entropy [Shannon 1948] under the current model. Formal mathematical descriptions of each method are given in the supplementary.

As noted in prior work [Beluch et al. 2018; Yoo and Kweon 2019], these strategies can suffer from a lack of diversity if applied naively, but can be readily adapted to minimise this effect by first sub-sampling the unlabelled pool and then employing the acquisition function to choose only from this restricted subset. A variation of this diversity heuristic worked well on our task: We first rank all pixels using the acquisition function, then uniformly sampling B/N pixel coordinates from the top $M\%$ ranked locations in each image, where M is a hyperparameter and N denotes the number of images we distribute our budget B amongst. We note that while more sophisticated strategies (e.g. [Sener and Savarese 2018]) could also be considered within our framework, a simple *Margin Sampling* strategy coupled with the modification described above proved effective (shown through experiments in Section 2.4), and thus we adopt it in this work.

Sampling batches. The number of pixel coordinates sampled in each round, B, is set as a hyperparameter. A larger value of B corresponds to fewer rounds of annotation (and therefore a potentially faster deployment cycle), at some cost in performance. A detailed study of the effects of B is provided in the supplementary material.

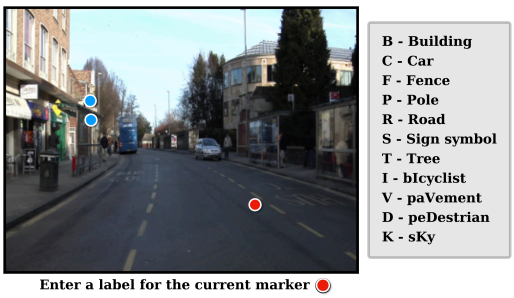


Figure 2.3: **PixelPick mouse-free annotation tool.** The annotator classifies the highlighted point (in red) by pressing the keyboard character of the corresponding class for the dataset. The tool then highlights the next pixel proposal and the process repeats. Note that the task requires the annotator to perform classification, but not localisation.

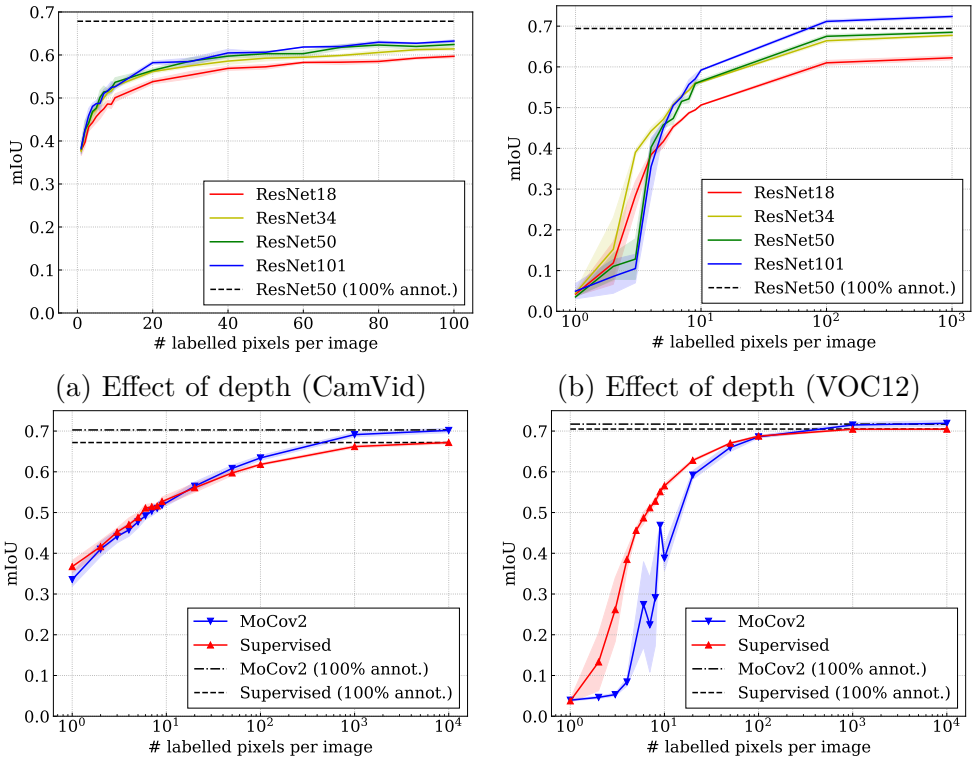
2.3.2 PixelPick annotation tool

To demonstrate the practical utility of the PixelPick framework, we created an annotation tool to support the labelling process (Fig. 2.3). The tool is simple: for each image, the annotator is presented with a few pixels that were selected by the PixelPick acquisition function (described in Section 2.3.1). They are also shown a mapping from keyboard keys to semantic labels (Fig. 2.3, right hand side). The tool iterates over the pixel locations, highlighting the current pixel in red and the annotator simply presses the appropriate key to classify it. The tool then moves on to the next pixel proposal, and the procedure repeats until all proposals in the image are exhausted, when a new image is shown.

We note that an important difference between this annotation technique and those considered in prior work (*e.g.*, scribbles [Lin et al. 2016], object pointing [Bearman et al. 2016; Qian et al. 2019], extreme clicks [Papadopoulos et al. 2017; Maninis et al. 2018] etc.) is that it is "mouse-free"—requiring only key presses from the user—but avoids the complexity of specialised approaches such as eye tracking [Papadopoulos et al. 2014]. In Section 2.4, we conduct experiments to validate the efficiency of the proposed annotation tool.

2.4 Experiments

In this section, we first describe datasets and implementation details for our experiments in Section 2.4.1. In Section 2.4.2, we conduct extensive ablation studies, and we then compare with existing state of the art approaches in Section 2.4.3. Finally, in Section 2.4.4, we demonstrate the practical feasibility and robustness of PixelPick by reporting annotation times and investigating its sensitivity to annotator errors.



(a) Effect of depth (CamVid)

(b) Effect of depth (VOC12)

(c) Effect of pretraining (CamVid)

(d) Effect of pretraining (VOC12)

Figure 2.4: **Ablation studies.** In (a) and (b), we investigate the effect of segmentation encoder *depth* on CamVid and VOC12, respectively. We observe that greater depth consistently helps performance above a threshold of 10 pixel labels per image. In (c) and (d), we compare fully-supervised ImageNet classification pretraining with self-supervised ImageNet (MoCov2 [Chen et al. 2020]) pretraining for the encoder on CamVid and VOC12, respectively, where we see that for lower numbers of pixel labels per image that fully-supervised pretraining is a better choice, but the situation reverses as more annotations become available.

2.4.1 Experimental setup

Datasets. We conduct experiments on the widely used CamVid [Brostow et al. 2009], Cityscapes [Cordts et al. 2016] and Pascal VOC 2012 [Everingham et al. 2015] (abbreviated to VOC12) datasets. Detailed descriptions of each dataset are provided in the supplementary.

Training setting. During each round of active learning, we enforce the cross-entropy loss only on the labelled pixels (*i.e.*, those in $\mathcal{D}_L^k$ for round k), as described in Section 2.3.1. Unless otherwise stated, M, the hyperparameter defining the % of top ranked pixel coordinates used as a basis for uniform sampling is set to 5, while B, the pixel labelling budget per round is set to $10N$ for CamVid and Cityscapes and $5N$ for VOC12, where N is the number of images in the dataset. At the beginning of each round, we reinitialise the model and train from scratch with the updated labelled pixels. For details about optimisation and data augmentation, please refer to the supplementary.

Evaluation metrics. Following standard practice [Golestaneh and Kitani 2020; Xie et al. 2020; Ouali et al. 2020a; Mackowiak et al. 2018], we compute mean intersection over union (mIoU), report our results on the test set for CamVid, and on the validation set for Cityscapes and VOC12 datasets. To provide a measure of variance in our low data regime, we report the average of 3 different runs (i.e., different seeds) on Pascal VOC 2012 and 5 runs on CamVid and Cityscapes for all experiments. We plot their standard deviations as shaded regions (± 1 std. dev.).

2.4.2 Ablation studies

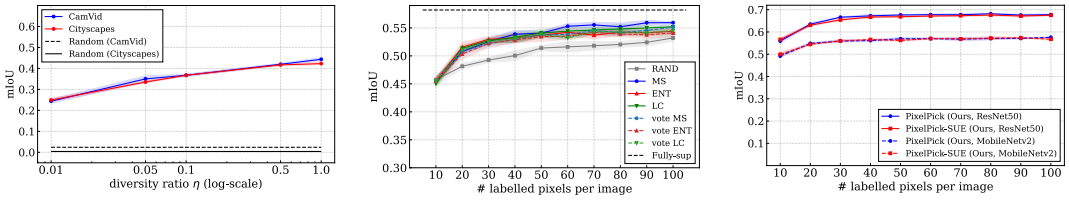
In this section, we explore the effect of four factors that affect performance in the PixelPick framework, with a particular focus on *annotation diversity* (with the goal of finding the most effective way to spend an annotation budget); *encoder depth* (varying the capacity of the encoder); *encoder initialisation* (self-supervised

vs supervised pretraining); and *acquisition function* (determining the best way to select pixels). Note that, while investigating the first three factors, all pixels are selected via simple uniform random sampling, with the goal of validating the effectiveness of inductive bias in modern ConvNets. We simulate the active learning process, following standard practice [Golestaneh and Kitani 2020; Xie et al. 2020; Casanova et al. 2020], *i.e.* to label the queried pixels, we simply reveal labels by querying the ground truth annotations at their spatial coordinates.

Annotation diversity. Given a fixed pixel labelling budget (B pixels), a natural question arises: *is it better to label a small number of images densely or a large number of images sparsely*? To address this question we design a simple experiment, where a fixed annotation budget of n pixels is to be distributed over a dataset of N_{total} images. We define the *annotation diversity ratio*, $\eta = \frac{N_{\text{img}}}{N_{\text{total}}}$, where N_{img} refers to the number of images that have had at least one pixel labelled (for simplicity, we assume the labelling budget is evenly distributed over the selected set of images). Therefore, $\eta \to 1$ refers to a budget uniformly distributed over the full dataset (thereby forming a sparse, but diverse, label set), $\eta \to 0$ denotes the case where the budget is only spent on a few images (yielding a densely annotated subset of images). We then train DeepLabv3+ models on CamVid and Cityscapes, fixing B so as to end up with 10 pixel labels per image when $\eta = 1$, and experiment with 5 different diversity ratios η from 0.01 to 1.0. In Fig. 2.5a, we observe that mean IoU increases monotonically with η. This indicates that, given a fixed budget, it is better to sparsely annotate as many images as possible, rather than a smaller number more densely, motivating our sparse PixelPick approach. In the remaining experiments, we likewise spend our annotation budget evenly across all images within a dataset (as described in Section 2.3.1), with each image being only sparsely labelled.

Encoder depth. We next investigate the effect of encoder capacity in the low annotation regime. Specifically, we experiment with a ResNet-based FPN by chang-

ing the number of layers in the encoder from 18 to 101 layers. All encoders are initialised with a model pretrained for classification on ImageNet [Russakovsky et al. 2015]. We conduct experiments both on CamVid (training each model with 1 to 100 randomly labelled pixel coordinates per image) and VOC12 (training each model with 1 to 1000 randomly sampled labelled pixel coordinates per image), reporting results in Fig. 2.4a and Fig. 2.4b, respectively. We observe that deeper networks yield higher performance above a minimum number of labelled pixels (approximately 10) per image. This implies that, at the cost of greater computational complexity, the use of a deeper network may be a viable way to reduce annotation requirements in low annotation regimes (above some minimum labelling threshold).



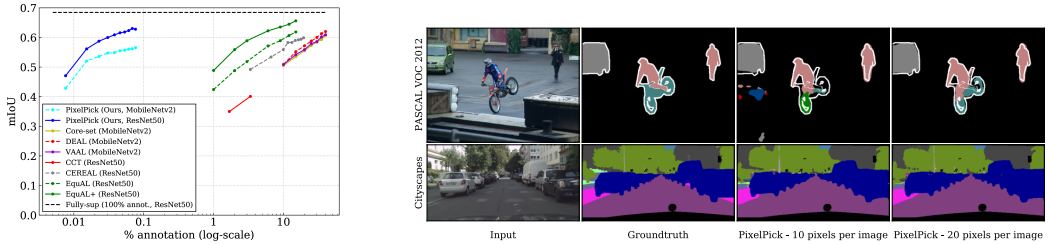
(a) Effect of annotation diversity

(b) Acquisition functions (CamVid)

(c) Sensitivity to annotator error (VOC12)

Figure 2.5: **Ablation studies.** In (a), we observe that sparsely annotating a larger number of images (higher η value) outperforms denser labelling of fewer images, with consistent trends on the CamVid and Cityscapes datasets. In (b), we compare acquisition functions on CamVid and find that *Margin Sampling* performs best. In (c), we investigate the sensitivity of the PixelPick framework to annotator errors by simulating a pixel classification user error (SUE) rate of 10%. We observe that performance is only marginally affected, indicating the practical robustness of the PixelPick framework.

Encoder initialisation. Next, we investigate whether supervised pretraining is necessary for good segmentation performance in a low annotation regime. Concretely, we compare the performance of an FPN-based architecture with a ResNet50 encoder that is initialised using either supervised (ImageNet classification) or self-supervised (MoCov2 [Chen et al. 2020]) pretraining. To study how performance differs with the number of labelled pixels, we vary the annotation budget from 1 to 10^4 randomly sampled labelled pixels per image on CamVid (Fig. 2.4c) and



(a) Comparison to prior work on Cityscapes

(b) Qualitative results for models trained with PixelPick on VOC12 (top) and Cityscapes (bottom)

Figure 2.6: **Comparison to state-of-the-art and qualitative results**. In (a) we observe that PixelPick performs favourably against existing state-of-the-art approaches for active learning and semi-supervised learning on Cityscapes. In (b) we show qualitative results. With only 10 labelled pixels per image, segmentation models trained with PixelPick achieve promising visual quality, which further improves to capture fine details (*e.g.*, the cleanly segmented thin lamppost in the bottom right image) as further labelled pixels are used.

VOC12 (Fig. 2.4d). On CamVid, we observe an interesting biphasic phenomenon: when the number of labelled pixels per image is fewer than 10, the model initialised with supervised ResNet50 shows superior performance. However, as the number of pixel labels increases, self-supervised pretraining gradually outperforms its supervised counterpart. This phenomenon is also observed on the VOC12 dataset, with a cross-over occurring at approximately 10^2 labelled pixels per image. Thus, supervised pretraining may be an appropriate choice for low annotation budgets, when suitable pretraining annotations are readily available, but its advantage wanes the annotation budget grows. Given its superiority at low annotation levels, we adopt supervised pretraining for the remaining experiments.

Acquisition function. Thus far, we have only labelled pixels selected via simple uniform random sampling, showing that modern CNNs—with their strong inductive biases—can be trained for semantic segmentation with just a handful of pixel annotations per image. Here, we go one step further, investigating whether a better choice of acquisition function can further improve learning efficiency. To this end, we experiment on CamVid with three popular uncertainty sampling methods (described in Section 2.3.1): *Least Confidence* (LC), *Margin Sampling* (MS)

and *Entropy Sampling* (ENT). In addition, we also experiment with a Query-By-Committee (QBC) [Seung et al. 1992] approach that queries labels using model ensembles [Settles 2009]. We implement this with dropout after each convolutional layer, repeating inference 20 times to obtain a Monte Carlo estimate following [Gal and Ghahramani 2016]. Due to the large number of models to be trained (*i.e.*, different acquisition functions, each trained five times to estimate variance), we employ the lightweight MobileNetv2-based DeepLabv3+ model. We initialise training with 10 uniform randomly selected labelled pixels per image. Once training converges, we query 10 additional pixel labels with the given acquisition function. As described in Section 2.3.1, we first take top $M\%$ ranked pixels (here, $M = 5$) per image under the uncertainty estimation ranking and uniformly sample 10 pixels from these pixels. Fig. 2.5b shows the results. We see that all uncertainty-based methods outperform the random baseline in every round. Interestingly, dropout-based voting variants of LC, MS and ENT each show worse performance than their counterparts voting—a similar observation was also made in [Casanova et al. 2020]. We note that in our problem setting, *Margin Sampling* (MS) outperforms other strategies, reaching about 96% of the performance of the fully supervised baseline with only 100 pixels per image (0.06% of the annotations). Therefore, we use MS as our sampling method for PixelPick to compare against previous work in the following section.

Discussion. To summarise, we can draw the following conclusions from the ablation studies: First, given a fixed pixel annotation budget, it is best to spread it over as many images as possible; Second, the inductive bias in modern ConvNets makes them well-suited to capture local correlations within an image, evidenced by the first three experiments, where models trained with randomly sampled pixels still perform well; Third, although it might be thought that deeper networks with greater capacity would suffer significantly from over-fitting in the low-annotation regime, we found that for many budget choices, deeper networks are the preferred option. Fourth, in terms of acquisition functions, active learning outperforms ran-

Method	Backbone	Train set (anno. type)	mIoU
Weakly-supervised methods			
GAIN [Li et al. 2018]	VGG16	10.5K imgs (classes)	55.3
MDC [Wei et al. 2018]	VGG16	10.5K imgs (classes)	60.4
DSRG [Huang et al. 2018]	ResNet101	10.5K imgs (classes)	61.4
FickleNet [Lee et al. 2019]	ResNet101	10.5K imgs (classes)	64.9
BoxSup [Dai et al. 2015]	VGG16	10.5K imgs (boxes)	62.0
ScribbleSup [Lin et al. 2016]	VGG16	10.5K imgs (scribbles)	63.1
Interactive weak supervision			
PixelPick (Ours)	ResNet50	1.5K imgs (sparse pixels)	**65.6**

Table 2.1: **Comparison to existing weakly-supervised methods on VOC12 validation set.** PixelPick is competitive against existing methods, using a budget of 20 pixel annotations per image when trained on a much smaller number of images.

dom sampling, and in particular, *Margin Sampling* performs best in our setting.

2.4.3 Comparison to state-of-the-art methods

We next validate our framework by comparing against prior work in active/semi-supervised learning on CamVid (Fig. 2.1) and Cityscapes (Fig. 2.6a) and in weakly-supervised learning on Pascal VOC 2012 (Table 2.1). To strike a balance between computation complexity and performance, we adopt the FPN model with a ResNet50 backbone, and query additional samples each round with *Margin Sampling*, as suggested by the ablation study. We train for 10 query rounds, with each round adding 10 labelled pixels per image for CamVid and Cityscapes and 5 pixels for VOC12.

Comparison to region-based active learning and semi-supervised methods. In Fig. 2.1 and Fig. 2.6a, we observe that our approach performs favourably to existing semi-supervised and region-based active learning approaches in terms of label efficiency on both CamVid and Cityscapes, highlighting that the strong inductive bias of CNNs can make good use of pixel-level supervision.

Comparison to previous weakly-supervised learning work. In Table 2.1, we compare our results on VOC12 to weakly-supervised models which leverage different forms of ground truth annotation. We observe that PixelPick compares favourably to existing methods (each trained on 10.5K images) when PixelPick uses a budget of 20 pixel annotations per image across only 1.5K training images (since active learning involves retraining the model multiple times, we use a limited size of training set). For reference, note that other methods such as BoxSup [Dai et al. 2015] and ScribbleSup [Lin et al. 2016] use multiple annotations per image, depending on the number of objects present.

Comparison to concurrent work of active learning for semantic segmentation In Fig. 2.7, we compare with the concurrent work of [Cai et al. 2021], which employs sparse superpixel annotation. To account for the differences between the architectures and training image resolutions used in each method, we compare the methods according to the % of fully supervised performance attained by the model. On the x-axis, we indicate the annotation cost (measured as the number of keyboard inputs for querying pixel labels, *i.e.*, the "class clicks" referred to in [Cai et al. 2021]). We observe that our proposed PixelPick approach is competitive with the concurrent work. In particular, we note that PixelPick is able to reach 90% performance with fewer annotation cost on both Cityscapes and VOC12 benchmarks.

2.4.4 Practical deployment

Thus far, we have largely followed the common practice in previous active learning segmentation work, mimicking the labelling process by simply disclosing the corresponding labels from the fully-annotated dataset. In this section, we evaluate the efficiency of PixelPick (Fig. 2.3) and its sensitivity to annotator noise during model training.

In detail, we ask eleven annotators to label 100 images from VOC12 dataset,

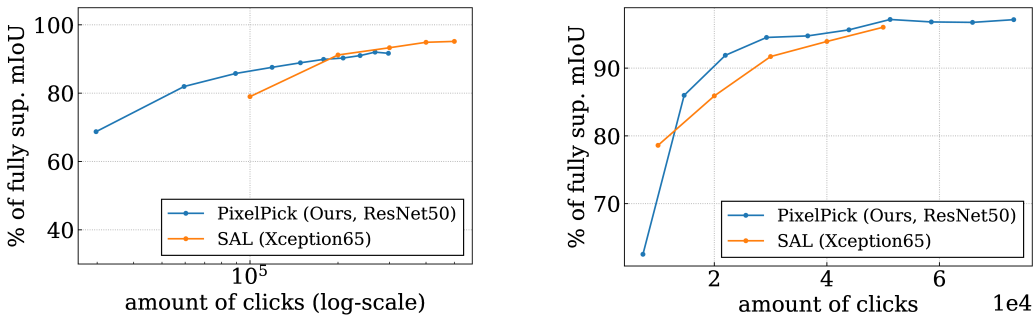


Figure 2.7: **A comparison based on estimated annotation cost.** We compare PixelPick against the concurrent work of [Cai et al. 2021] in terms of mIoU ratio vs annotation cost approximated by the amount of clicks on (left) Cityscapes and (right) VOC12 datasets.

with 10 pixels per image, we measure the average time and accuracy (between annotator and the groundtruth from original dataset). As a result, with our simple unoptimised annotation tool, it takes 1.42 seconds on average to label the queried pixel (14s per image), with 87.7% average accuracy. To our knowledge, this annotation speed is significantly faster than drawing bounding boxes or scribbles [Dai et al. 2015; Lin et al. 2016], and approximately twice as fast as picking extreme points according to times reported by [Papadopoulos et al. 2017]. Additionally, given the observed annotation error rate, we conduct an experiment to assess the influence of these noisy annotations, that is, we artificially jitter 10% of the groundtruth annotations to simulate errors during the annotation process and train a model on pixels containing this label noise. As shown in Fig. 2.5c, the performance gap incurred from annotation noise is negligible, indicating that our framework is not only efficient with respect to annotation time but also robust to potential errors caused by annotators. This is an encouraging sign for the practical potential of the PixelPick framework for real-world deployment.

2.5 Conclusion

In this work we proposed PixelPick, a framework for semantic segmentation that employs a small number of sparsely annotated pixels to train effective segmentation

models. We showed that PixelPick requires considerably fewer annotations than existing state-of-the-art to achieve comparable performance. Finally, we showed how annotation for pixel-level active learning can be obtained efficiently with a mouse-free labelling tool, facilitating real-world deployment. We hope that our work encourages further research into the promising use of sparse pixel-level annotation for image understanding.

Acknowledgements

GS is supported by AI Factory, Inc. in Korea. WX and SA are supported by Visual AI (EP/T028572/1). The authors would like to thank Tom Gunter for suggestions. SA would also like to thank Z. Novak and S. Carlson for support.

Appendices

Appendices for this chapter can be found in the online version of the paper.[1]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[1] https://www.robots.ox.ac.uk/~vgg/research/pixelpick

Part II

Semantic and Instance Segmentation without Manual Annotations

Chapter 3

ReCo: Retrieve and Co-segment for Zero-shot Transfer

This work has been accepted for publication at the Thirty-Sixth Conference on Neural Information Processing Systems (NeurIPS), 2022.

ReCo: Retrieve and Co-segment for Zero-shot Transfer

Gyungin Shin[1] Weidi Xie[1,2] Samuel Albanie[3]

[1]Visual Geometry Group, University of Oxford

[2]Coop. Medianet Innovation Center, Shanghai Jiao Tong University

[3]Department of Engineering, University of Cambridge

https://www.robots.ox.ac.uk/~vgg/research/reco

Abstract

Semantic segmentation has a broad range of applications, but its real-world impact has been significantly limited by the prohibitive annotation costs necessary to enable deployment. Segmentation methods that forgo supervision can side-step these costs, but exhibit the inconvenient requirement to provide labelled examples from the target distribution to assign concept names to predictions. An alternative line of work in language-image pre-training has recently demonstrated the potential to produce models that can both assign names across large vocabularies of concepts and enable zero-shot transfer for classification, but do not demonstrate commensurate segmentation abilities.

We leverage the retrieval abilities of one such language-image pre-trained model, CLIP, to dynamically curate training sets from unlabelled images for arbitrary collections of concept names, and leverage the robust correspondences offered by modern image representations to co-segment entities among the resulting collections. The synthetic segment collections are then

employed to construct a segmentation model (without requiring pixel labels) whose knowledge of concepts is inherited from the scalable pre-training process of CLIP. We demonstrate that our approach, termed **Re**trieve and **Co**-segment (ReCo) performs favourably to conventional unsupervised segmentation approaches while inheriting the convenience of nameable predictions and zero-shot transfer. We also demonstrate ReCo's ability to generate specialist segmenters for extremely rare objects.

3.1 Introduction

The objective of *semantic segmentation* is to partition an image into coherent regions and to assign to each region a semantic label. This task has myriad applications across domains such as medical image analysis, autonomous driving, industrial process monitoring and wildlife tracking. However, there are several key challenges that have hindered the deployment of existing semantic segmentation approaches to date: (1) **Cost**: collecting manual pixel-level annotations is extraordinarily expensive (*e.g.*, 90 minutes per image for high quality labels [Cordts et al. 2016]), limiting the use of fully-supervised approaches; (2) **Flexibility**: supervised approaches have typically been trained with limited lists of pre-defined categories and lack the ability to recognise rare or novel categories (such as those described by free-form text); (3) **Complexity of deployment**: unsupervised segmentation methods have dramatically reduced annotation costs, but still exhibit the inconvenience of requiring labelled examples to assign names to predictions; (4) **Data access**: many existing approaches (both supervised and unsupervised) are trained on the *target data distribution*, requiring both that this distribution is known at training time (limiting flexibility) and that this data is accessible, which may not be the case for legal/ethical reasons (*e.g.*, medical image data).

There is a rich body of semantic segmentation literature that proposes solutions to subsets of these challenges, but to our knowledge no existing work addresses their full combination. To tackle all four challenges, we draw inspiration from two

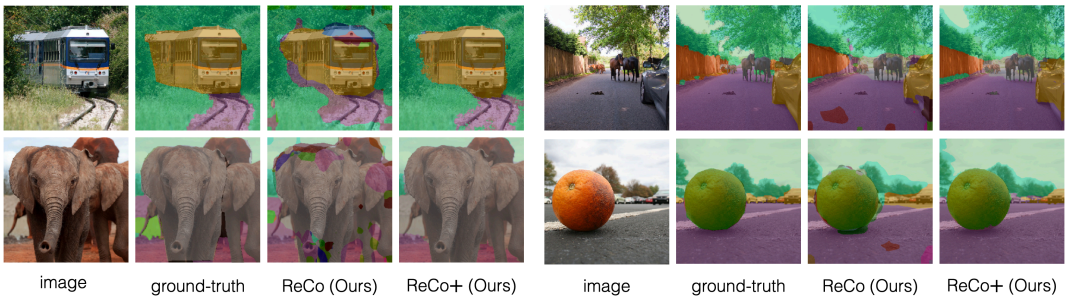


Figure 3.1: We propose ReCo, a new framework for semantic segmentation zero-shot transfer without pixel supervision. The figure depicts ReCo segmentations for COCO-Stuff [Caesar et al. 2018], indicating promising results in the challenging zero-shot transfer setting. We also illustrate results with unsupervised adaptation to the target distribution (ReCo+), trading flexibility for improved segmentation quality.

lines of recent research. The first line of research has shown that modern deep architectures (most notably vision transformers [Dosovitskiy et al. 2021]) develop the ability to infer the spatial extent of objects without pixel supervision [Caron et al. 2021; Naseer et al. 2021; Siméoni et al. 2021; Wang et al. 2022b; Shin et al. 2022a; Melas-Kyriazi et al. 2022a] and moreover can establish semantically consistent correspondences across images [Collins et al. 2018; Hamilton et al. 2022]. The second line of research has demonstrated that large-scale visual-language pre-training [Radford et al. 2021; Jia et al. 2021] produces models that possess both a large vocabulary and remarkable *zero-shot transfer* potential—the ability to recognise concepts on target datasets without access to the target data distribution during training.

In this work, we target a synthesis of these two approaches that draws on their respective strengths for the semantic segmentation task. We first employ CLIP [Radford et al. 2021] to curate training sets from unlabelled images for any desired list of concepts. We then exploit the robust semantic correspondences offered by modern vision backbones to co-segment concepts among the resulting curated collections. Finally, we use these co-segmented concepts to construct a segmentation model for the given concepts, *without* requiring training. This framework, which we term **Re**trieve and **Co**-segment (ReCo), performs favourably to

existing unsupervised segmentation approaches while preserving the benefits of a wide vocabulary, named predictions and zero-shot transfer exemplified by CLIP. Since ReCo utilises a pre-trained image-language model, we call the task considered in this work *unsupervised semantic segmentation with language-image pre-training* to differentiate it from conventional unsupervised semantic segmentation task formulations.

Our contributions are three-fold: (1) We propose the ReCo framework, enabling open vocabulary semantic segmentation, without pixel annotations or the need to provide labelled examples from the target domain thereby enabling zero-shot transfer; (2) We compare our approach to prior work on standard semantic segmentation benchmarks (COCO-Stuff [Caesar et al. 2018], Cityscapes [Cordts et al. 2016], and KITTI-STEP [Weber et al. 2021]), and further illustrate the ability of ReCo to segment rare concepts beyond these benchmarks; (3) For cases when the target image distribution *is* available, we demonstrate that a simple extension of our approach, ReCo+, can exploit this data access via unsupervised adaptation to bring further gains.

3.2 Related work

Our work is connected to several themes in the literature, which we describe next.

Unsupervised semantic segmentation. There has been considerable recent progress towards unsupervised semantic segmentation with deep neural networks by leveraging ideas from self-supervised learning. Learning objectives based on maximising mutual information between views [Ji et al. 2019; Ouali et al. 2020b], metric learning across proposals [Zhang and Maire 2020; Van Gansbeke et al. 2021], equivariance and invariance constraints [Cho et al. 2021], distillation of self-supervised feature correspondences [Hamilton et al. 2022] and cross modal cues (vision and LiDAR) [Vobecky et al. 2022] have all shown their potential for this task.

One drawback of these approaches is a reliance on either nearest neighbour search on a held-out set with pixel level annotations or the Hungarian algorithm [Kuhn 1955] (optimally matching predictions against ground-truth semantic masks) to produce segments with names. In contrast, by leveraging the ability of a language-image model to name concepts, ReCo is independent of labelled examples during both training and inference.

Weakly-supervised semantic segmentation. To reduce annotation costs, a number of works have explored weaker cues such eye tracking [Papadopoulos et al. 2014], pointing [Bearman et al. 2016; Qian et al. 2019; Cheng et al. 2021a], sparse pixel labels [Shin et al. 2021], scribbles [Lin et al. 2016], web-queried samples [Jin et al. 2017], boxes [Dai et al. 2015; Khoreva et al. 2017; Song et al. 2019], extreme clicks [Papadopoulos et al. 2017; Maninis et al. 2018], image-level labels [Zhou et al. 2016; Wei et al. 2018; Fan et al. 2018; Ahn and Kwak 2018; Chang et al. 2020; Pathak et al. 2015; Pinheiro and Collobert 2015] and free-form text [Xu et al. 2022]. However, such approaches still require the weak annotation to be attached to the data used to train the segmentation model—by contrast, ReCo can in principle train a segmenter from any unlabelled collection of images. ReCo also bears a conceptual similarity with *webly-supervised* approaches [Fergus et al. 2005] for semantic segmentation [Jin et al. 2017; Shen et al. 2018]. These methods employ an image search engine such as Google to provide training samples for concepts. However, their flexibility is limited (this approach cannot be applied to private or commercially sensitive data, for example) and they lack the ability to leverage the knowledge of the search engine itself during inference (we demonstrate that integrating the vision-language model into the inference procedure brings significant gains in performance).

Zero-shot semantic segmentation with pretrained language/vision-language embeddings. A diverse body of work has explored zero-shot semantic segmentation, broadly defined as the task of segmenting categories for which no labels were provided during training (often termed *zero-label* semantic segmenta-

tion [Xian et al. 2019]). The key idea underlying many of these works is to leverage relationships encoded in pretrained word embeddings (such as word2vec [Mikolov et al. 2013] or GloVe [Pennington et al. 2014]) to enable generalisation to unseen categories [Zhao et al. 2017; Bucher et al. 2019; Gu et al. 2020; Xian et al. 2019; Kato et al. 2019; Hu et al. 2020a; Li et al. 2020; Pastore et al. 2021]. More recent work has sought to leverage the vision-language embedding space learned by CLIP [Radford et al. 2021] to improve dense prediction in various settings [Zhou et al. 2021; Rao et al. 2021; Li et al. 2022; Xu et al. 2021; Ding et al. 2022; Lüddecke and Ecker 2022]. We adopt a variation of DenseCLIP [Zhou et al. 2021] as a component of our framework. Additionally, differently from the above, we pursue the formulation of *zero-shot transfer* popularised by CLIP [Radford et al. 2021] which evaluates performance on *unseen datasets* rather than *unseen categories*. Consequently, unlike these works, our model has no access to either labelled or unlabelled examples from the target data distribution (or pixel-level labels from the source distribution, as investigated by [Xu et al. 2021]). We note one exception: in addition to their primary zero-shot evaluations, DenseCLIP [Zhou et al. 2021] also report an "annotation-free" evaluation without access to the target dataset—we compare our approach with theirs under an equivalent setting.

Large-vocabulary/rare concept segmenters. To scale up the number of concepts that can be segmented by a model several strategies based on captions [Ghiasi et al. 2021b], grounded text descriptions [Kamath et al. 2021] and annotation transfer [Hu et al. 2018; Huynh et al. 2021] have been explored. In a differing direction, various losses and incremental learning techniques have been employed [Hu et al. 2020b; He et al. 2022] to better segment rare concepts. Unlike ReCo, however, each of the above approaches still requires costly pixel-level annotations.

Co-segmentation which aims to segment common regions among a collection of images, has been widely studied with classical computer vision approaches [Rother et al. 2006; Mukherjee et al. 2009; Vicente et al. 2010; Batra et al. 2010; Joulin

et al. 2010; Vicente et al. 2011]. The topic has been revisited with deep learning using shared encoder networks [Chen et al. 2018a; Li et al. 2019; Banerjee et al. 2019; Liu et al. 2020], iterative refinement [Zhang et al. 2021a] and weak (class-label) supervision [Hsu et al. 2018]. While ReCo can in principle make use of any unsupervised co-segmentation algorithm, we find that a simple correlation strategy works well, and thus we adopt it for our approach.

3.3 Method

In this section, we first formalise the task of unsupervised semantic segmentation with language-image pre-training (Section 3.3.1). We then introduce the ReCo framework (Section 3.3.2) which enables zero-shot transfer for semantic segmentation with arbitrary categories. We describe a language-based gating mechanism to enhance segmentation quality (Section 3.3.3), and a pseudo-labelling scheme, ReCo+, which adapts to a target distribution using predictions from ReCo (Section 3.3.4).

3.3.1 Unsupervised semantic segmentation with language-image pre-training

Let us denote by $x \in \mathbb{R}^{3\times h\times w}$ an image of interest. Let $\Omega = \{1,\dots,h-1\} \times \{1,\dots,w-1\}$ denote its spatial domain and $\omega \in \Omega$ a pixel location. The objective of *unsupervised semantic segmentation with language-image pre-training* (USLIP) is to assign to each pixel location $\omega \in \Omega$ a category, $c \in C$, that falls among one of $|C|$ mutually exclusive target categories. The key characteristic distinguishing USLIP from traditional unsupervised segmentation is that in order to tackle the task, a model may assume access to *pre-training corpus* of paired image and text data, without *pixel-level supervision*. We distinguish this approach from weakly-supervised learning in that it does not rely on weak human annotations (*e.g.* image-level labels) during training for a downstream task (*i.e.* segmentation).

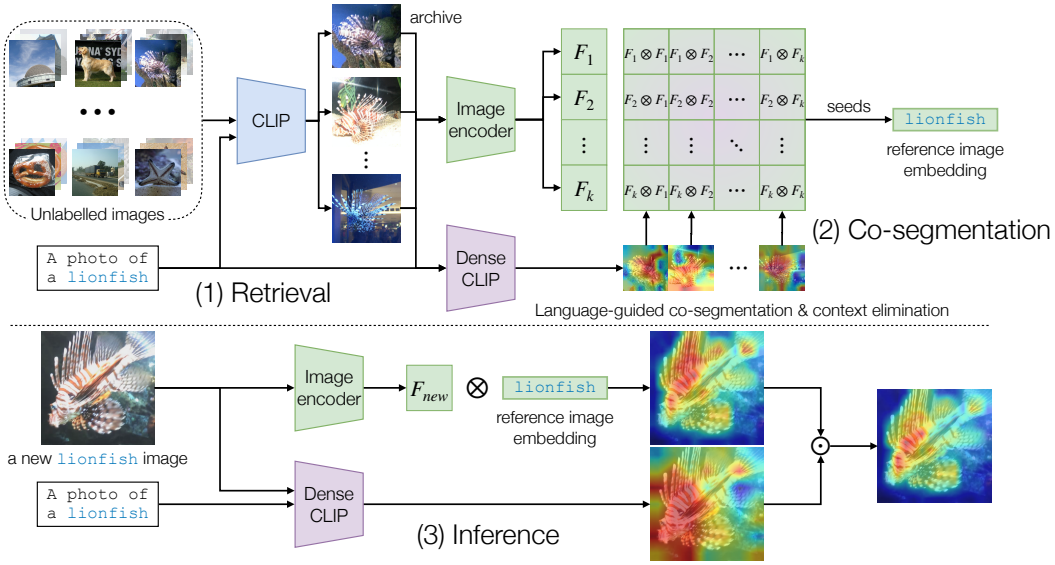


Figure 3.2: In this work, we propose ReCo, a framework for open vocabulary semantic segmentation zero-shot transfer utilising a language-image pre-training model (*i.e.*, CLIP). **Top:** (1) Given a large-scale unlabelled dataset and a category to segment, we first curate an archive of k images from the unlabelled collection using CLIP [Radford et al. 2021]. (2) Using a pre-trained visual encoder (*e.g.*, MoCov2 [He et al. 2020], DeiT-S/16-SIN [Naseer et al. 2021]), we extract dense features from the archive images, which are used to generate a reference image embedding for the given category via a co-segmentation process. **Bottom:** (3) During inference, the reference image embedding is employed to produce an initial segmentation of the target concept which is refined with DenseCLIP. $\otimes$ and $\odot$ denote inner product and Hadamard product, respectively. See Section 3.3 for details.

3.3.2 Retrieve and Co-segment (ReCo)

The inputs to ReCo are a collection of unlabelled images, and a list of text descriptions of concepts to be segmented. Through a combination of *image retrieval* and *co-segmentation* across the image collection, ReCo constructs a segmenter for the given concepts on the fly. During *inference*, this segmenter is applied *without* fine-tuning to images from a target distribution of interest, thus supporting the zero-shot transfer. The interaction of these three stages is illustrated in Fig. 3.2 and discussed in more detail below.

Curating exemplars through image retrieval. The first hypothesis underpinning our approach is that it is possible to construct a sufficiently large and diverse unlabelled image collection that contains examples (or closely related examples) of any concept we wish to segment. We consider this hypothesis to be reasonable in light of the fact that recent research has produced a number of diverse datasets spanning billions of images [Mahajan et al. 2018; Goyal et al. 2022; Jia et al. 2021; Schuhmann et al. 2022].

To make use of this hypothesis for segmentation, we employ a vision-language model to curate an archive of images for each concept of interest. In this work, we use CLIP [Radford et al. 2021], a model comprising an image encoder and a text encoder that enables efficient image retrieval for free-form text queries. In more detail, for a given text query describing a concept c the CLIP text encoder $\psi_{\mathcal{T}}$ produces an embedding $\psi_{\mathcal{T}}(c) \in \mathbb{R}^e$ which can be compared against the embeddings produced by the image encoder $\psi_{\mathcal{I}}$ for each image x in the unlabelled collection $\mathcal{U}$: $\{\psi_{\mathcal{I}}(x) \in \mathbb{R}^e : x \in \mathcal{U}\}$. We then build an archive from the k images that form nearest neighbours for each concept c. While a more sophisticated strategy (for instance, adjusting the archive size according to concept difficulty or prevalence) is possible with ReCo, we found that this simple approach worked well, and so we adopt it here. Note that, thanks to mature approximate nearest neighbour search techniques [Johnson et al. 2019], this archive construction process can be readily

applied to a collection $\mathcal{U}$ containing billions of images.

Co-segmentation with seed pixels. The second hypothesis underpinning our approach is that modern vision-language models such as CLIP are capable of constructing archives of *high purity* (*i.e.*, a high ratio of the archive images contain the concept of interest) provided that such exemplars exist in the underlying image collection $\mathcal{U}$ (our first hypothesis). Our second hypothesis, which we validate through experiments in Section 3.4, enables the use of *co-segmentation* to obtain concept segments.

Concretely, given the k images comprising the archive for category c constructed by CLIP, we aim to segment regions corresponding to concepts that re-occur across the images. Since we expect the archive to be of high purity, and since modern visual backbones provide consistent semantic correspondences across images [Hamilton et al. 2022], we adopt a strategy of first identifying a *seed pixel* in each image that we are confident belongs to the target category, c. Intuitively, good seed pixels are ones that have close neighbours (*i.e.*, strong support) across each of the archive images, since we assume that: (i) the target concept is common to all images in the archive, (ii) our visual backbone will produce consistent features for pixels belonging to the same concept. We then use these seed pixels to construct a reference embedding that can be used to classify pixels belonging to c.

In detail, we first extract dense features for each image in the archive using a pre-trained image encoder $\phi_{\mathcal{I}}$:

$$\{F_1, \ldots, F_k\} = \{\phi_{\mathcal{I}}(x_1), \ldots, \phi_{\mathcal{I}}(x_k)\} \tag{3.1}$$

where $F_i \in \mathbb{R}^{d \times h \times w}$ denotes (spatially) dense features for the i^{th} image with height h, width w and d channels. Each such feature is L2-normalised along its channel dimension.[1] Note that any image encoder $\phi_{\mathcal{I}}$ can be employed here (it need not

[1]For clarity purposes, we assume that all images share the same spatial extent (and thus their dense features also do). However, the proposed co-segmentation approach can be applied to

be CLIP).

We identify seed pixels in four steps: First, we construct an adjacency matrix $A^{khw \times khw}$ among all features in the archive. Here, each of the $k \times k$ submatrices, $A_{ij} \in \mathbb{R}^{hw \times hw}$, encodes pairwise similarities between features from image i and image j for $i, j \in \{1, ..., k\}$. Second, we aim to identify, for each pixel in the archive, the similarity of its nearest neighbour among each of the k images. To do so, we apply a `max` operator along the columns of each submatrix (reducing the hw colums of each submatrix to 1 and reducing the overall adjacency matrix dimensions to $khw \times k$). Third, we aim to identify the average *support* that each pixel has among the k archive images. For this, we apply a `mean` operator over the columns of each submatrix such that each row of the resulting $khw \times 1$ matrix encodes the mean maximum similarity across k images. Finally, we identify the seed pixel locations by applying an `argmax` operator to each of the k submatrices of size $hw \times 1$, yielding the spatial indices of the features in each image with highest average maximum similarity across the archive.

To construct a classifier for concept c, we simply average the embeddings of the k seed pixels from its archive and L2-normalise the resultant vector to produce the reference embedding $f_c \in \mathbb{R}^d$.

Inference. To localise instances of the category in a new image x_{new}, we first compute the dot-product between f_c and the L2-normalised dense features F_{new} from the new image, and pass the result through a sigmoid:

$$P^c_{\text{new}} = \sigma(f_c \cdot F_{\text{new}}) \in [0, 1]^{h \times w} \tag{3.2}$$

where P^c_{new} denotes an initial estimate of the probability map corresponding to category c.

To refine this probability map, we draw inspiration from recent work [Zhou et al. 2021; Rao et al. 2021] showing that dense visual CLIP features can be use-

images with different resolutions.

fully correlated against a given CLIP text embedding. Concretely, we employ the DenseCLIP mechanism of Zhou et al. [Zhou et al. 2021] to highlight regions of the input that are salient for the target category c as follows. For image x_{new}, we extract features $V_{\text{new}} \in \mathbb{R}^{e_v \times h \times w}$ from the last self-attention layer values of the CLIP image encoder (here e_v denotes the value feature dimension), project the features into the joint space $\mathbb{R}^e$ with the CLIP image encoder's final linear projection and L2-normalise the result. We then compute the CLIP text embedding $\psi_{\mathcal{T}}(c) \in \mathbb{R}^e$ for the target concept c and L2-normalise it before producing a saliency map $\mathcal{S}^c_{\text{new}} \in [0,1]^{h \times w}$ by applying the text embedding as a 1×1 convolution to the visual features and applying a sigmoid activation function to the result. Note that our use of a sigmoid activation function differs from the softmax used by [Zhou et al. 2021], since we process each concept independently.

Our final probability map for category c is produced by the Hadamard product of these estimates:

$$\bar{P}^c_{\text{new}} = P^c_{\text{new}} \odot \mathcal{S}^c_{\text{new}} \tag{3.3}$$

In case of multiple categories predictions, we concatenate all the category prediction maps and apply `argmax` to the category dimension. As a simple post-processing step, we also experiment with the effect of applying a CRF [Krähenbühl and Koltun 2011a] (similarly to [Hamilton et al. 2022]). Pseudocode for ReCo can be found in the supplementary material.

3.3.3 Language-guided co-segmentation and context elimination

Even when an archive has high purity, co-segmenting the target concept c from the collection of images can be challenging due to the potential presence of distractor categories $\tilde{c}$ that often co-occur with c. For example, *cars* often co-occur with *roads* and likewise *aeroplanes* with *sky*. To minimise the risk of the co-segmentation algorithm anchoring on unintended categories, we introduce two mechanisms into the co-segmentation procedure, namely, *language-guided co-segmentation* and *context*

elimination.

Language-guided co-segmentation. Specifically, for each x_i in the archive, we first compute the corresponding saliency map $\mathcal{S}_i^c \in [0,1]^{h \times w}$ using DenseCLIP. We then vectorise this map and use it to filter the co-segmentation similarities by replacing each row of $A_{j,i}$ with the Hadamard product of itself with $\text{vec}(\mathcal{S}_i^c) \in [0,1]^{1 \times hw}$ (where $\text{vec}(\cdot)$ denotes vectorisation) for each submatrix $j = \{1, ..., k\}$. Intuitively, this serves to act as a *gate* that preserves similarities only for pixels identified by CLIP as salient for the target concept.

Context elimination. To reduce the influence of widely co-occurring distractors, we select a few common background categories $\tilde{c}$ that frequently appear in images (*e.g. tree, sky, road*) and compute their reference embeddings $f_{\tilde{c}}$ as described in the seeding procedure of Section 3.3.2. We then use the resulting attention maps $P_i^{\tilde{c}}$ produced by these reference embeddings to suppress regions of common background context in the subsequent co-segmentation processes for different categories. This is done similarly as above, by vectorising and replacing each row of $A_{j,i}$ with the Hadamard product of itself with $\text{vec}(1 - P_i^{\tilde{c}}) \in [0,1]^{1 \times hw}$ for each submatrix $j = \{1, ..., k\}$. If a target concept c is identical to one of the common background categories $\tilde{c}$, we replace f_c with the corresponding $f_{\tilde{c}}$.

3.3.4 ReCo+: Fine-tuning ReCo with pseudo-labels

The ReCo framework outlined above requires no training and no access to the target distribution. However, it is possible to consider a simple extension to this idea when access to the target image distribution is available. Our proposed extension, ReCo+, simply trains a segmentation architecture (*e.g.* DeepLabv3+ [Chen et al. 2018b]) with the segmentation masks from ReCo as pseudo-masks.

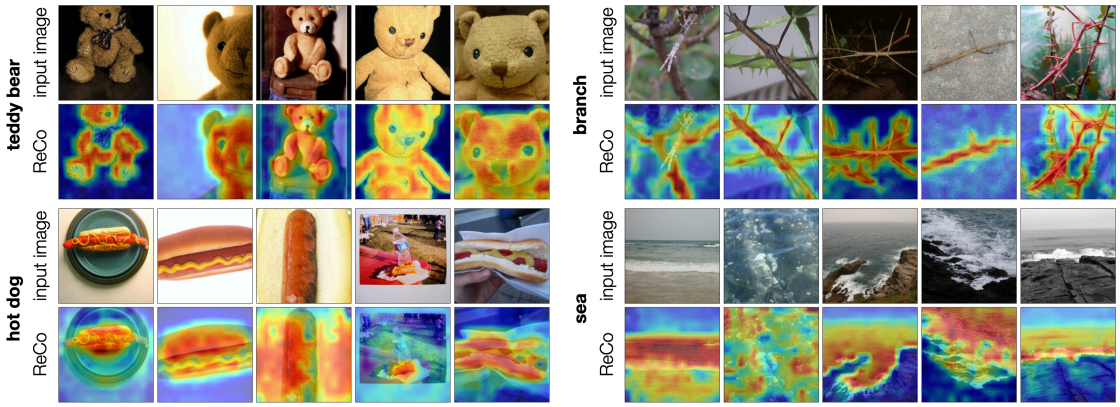


Figure 3.3: **Sample visualisations of the co-segmentation approach used by ReCo.** We display the top 5 ranked images among an archive of 50 retrieved images for a given concept. Highlighted regions are shown in red. For both *things* (left) and *stuff* (right) categories, ReCo reliably recognises appropriate regions. Best viewed in colour.

3.4 Experiments

In this section, we start by describing the datasets used for our experiments (Section 3.4.1) and implementation details (Section 3.4.2). Then, we conduct an ablation study (Section 3.4.3) and compare our model to state-of-the-art methods for unsupervised semantic segmentation with and without language-image pre-training (Section 3.4.4). Finally, we showcase our model's ability to segment rare-category objects (Section 3.4.5).

3.4.1 Datasets

For our ablation study, we use the ImageNet1K [Deng et al. 2009] validation set to curate archive for concepts of interest. The dataset covers 1K classes with 50 images for each class. To measure segmentation performance in the zero-shot transfer setting, we use the PASCAL-Context [Mottaghi et al. 2014] validation set for evaluation, which has 5,104 images of 59 categories excluding the background class.

To compare with previous unsupervised segmentation methods, we use the

ImageNet1K training set to construct the reference embedding for the concept we wish to segment. The dataset has 1K classes with 1.2M images. We evaluate on standard benchmarks including the Cityscapes [Cordts et al. 2016] validation split, which has 500 urban scene images of 27 categories, KITTI-STEP [Weber et al. 2021] validation set, which is composed of 2,981 urban scene images of 19 categories, and COCO-Stuff [Caesar et al. 2018] validation split, which has 4,172 images of 171 low-level thing and stuff categories excluding background class. Following [Cho et al. 2021; Hamilton et al. 2022], we use the 27 mid-level categories for evaluation. For unsupervised adaptation with ReCo+ (Section 3.3.2), we train on ReCo pseudo-labels on the Cityscapes training set with 2,975 images, KITTI-STEP training set which contains 5,027 images, and the COCO-Stuff10K subset which has 9,000 images for each respective benchmark. We emphasise that *no ground-truth labels are used for training.*

Finally, to demonstrate our model's ability to segment rare concepts, we use the LAION-5B dataset [Schuhmann et al. 2022] with 5 billion images as a large collection of images that we expect to satisfy our first hypothesis, namely that it will have coverage of rare concepts. To assess performance, we use the FireNet dataset [Panella et al. n.d.] which has 1,452 images spanning rare fire safety-related classes. For our experiment, we select the *fire extinguisher* class as an example of a concept that is important but rare in vision datasets (it is not contained in ImageNet1K [Russakovsky et al. 2015], for example) and evaluate ReCo on 263 images containing at least one instance of the category. As a further proof of concept, we also demonstrate co-segmentations of the *Antikythera mechanism* (a historical item that does not appear in WordNet [Miller 1995], or any labelled vision datasets that we are aware of).

3.4.2 Implementation details

Here, we describe the hyperparameters used to train ReCo+, inference details and evaluation metrics. Our implementation is based on the PyTorch library [Paszke

et al. 2019] and made publicly available.[2]

ReCo+ Training. While ReCo does not require training, we train ReCo+ based on the DeepLabv3+ [Chen et al. 2018b] segmentation architecture with a ResNet101 [He et al. 2016] backone on the predictions from ReCo as described in Section 3.3.2. All training images are resized and center-cropped to 320×320 pixels and data augmentations such as random scaling, cropping, and horizontal flipping are applied with random color jittering and Gaussian blurring. We use the Adam optimiser [Kingma and Ba 2014] with an initial learning rate of 5×10^{-4} and a weight decay of 2×10^{-4} with the Poly learning rate schedule as in [Liu et al. 2015; Chen et al. 2018b]. Training consists of 20K gradient iterations with a batch size of 8 and takes about 5 hours on a single 24GB NVIDIA P40 GPU.

Inference. For each benchmark, we pre-compute reference image embeddings for a list of categories for the benchmark and store the embeddings to form a classifier. Whenever DenseCLIP is employed, we use the ResNet50x16 model (following [Zhou et al. 2021]) to construct a saliency map for each image. Unless otherwise stated, for the COCO-Stuff and Cityscapes benchmarks, we resize and center crop the input images to 320×320 pixels as in [Hamilton et al. 2022]. For the KITTI-STEP validation set, we use the original resolution of each image as in [Ke et al. 2022]. For the FireNet benchmark, we resize the shorter side of images to 512 pixels and predict a single class of *fire extinguisher* by thresholding the predicted heatmap with probability of 0.5.

Evaluation metrics. Following the common practice [Ji et al. 2019; Cho et al. 2021; Hamilton et al. 2022], we report pixel accuracy (Acc.) and mean intersection-over-union (mIoU).

3.4.3 Ablation studies

Ability of CLIP to curate archives. We begin by assessing the validity of our second hypothesis—namely that CLIP is capable of achieving *high purity* archives

[2]Code available at https://github.com/NoelShin/reco

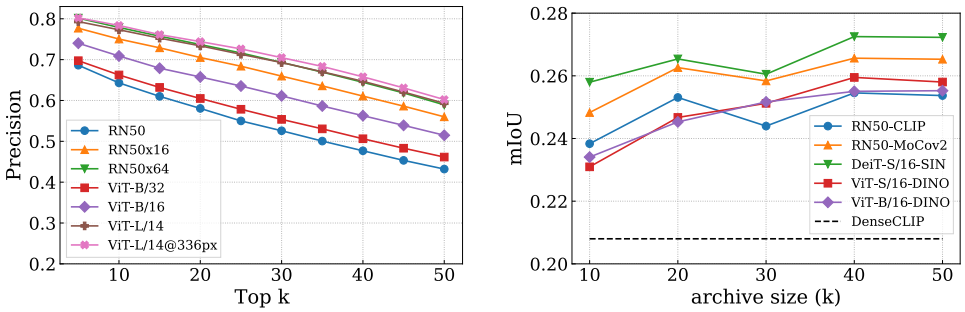


Figure 3.4: **Ablation studies. Left:** Image retrieval performance of different CLIP models on the ImageNet1K validation set with k ranging from 5 to 50. ViT-L/14@336px performs particularly strongly, suggesting the ability to curate archives of high purity. **Right:** Co-segmentation performance on PASCAL-Context validation set as we vary the archive size and choice of visual encoders. We observe a general trend towards improved performance with increasing archive size for all encoders.

DenseCLIP	LGC	CE	CRF	Acc.	mIoU
✗	✗	✗	✗	16.8	5.7
✓	✗	✗	✗	41.1	21.8
✓	✓	✗	✗	43.1	23.1
✓	✗	✓	✗	49.7	26.0
✓	✓	✓	✗	50.9	26.6
✓	✓	✓	✓	**51.6**	**27.2**

Table 3.1: **Influence of ReCo components for zero-shot transfer on PASCAL-Context** [Mottaghi et al. 2014]**.** We observe that integrating DenseCLIP during inference, *Language-guided co-segmentation* (LGC), *Context elimination* (CE), and CRF [Krähenbühl and Koltun 2011a] post-processing each contribute to improved performance. All comparisons use a DeiT-SIN visual backbone for co-segmentation and ViT-L/14@336px for archive curation.

from unlabelled images. To this end, we evaluate the retrieval performance of different CLIP models on the ImageNet1K validation set when constructing different archive sizes. In detail, for each archive size, k, we compute the precision of the top-k retrieved images based on whether the the ground-truth image-labels match the query text. As can be seen in Fig. 3.4 (left), all CLIP models achieve solid retrieval performance, suggesting their potential for curating high purity archives as part of ReCo. Since ViT-L/14@336px performs best, we employ this as our retrieval model in the remaining experiments.

Influence of archive size and visual encoder used for co-segmentation. In Fig. 3.4 (right) we illustrate the effect of using different pre-trained architectures, *e.g.* MoCov2 [He et al. 2020], DINO [Caron et al. 2021], CLIP [Radford et al. 2021], DeiT-SIN [Naseer et al. 2021], as the archive size (and thus the number of images used for co-segmentation) changes. The y-axis depicts segmentation performance for ReCo with these configurations on the PASCAL-Context benchmark. We observe that using larger archives tends to improve performance (likely due to their reasonably high purity) albeit non-monotonically, and that features from DeiT-SIN perform best. We therefore use DeiT-SIN with an archive size of k=50 for the remaining experiments unless otherwise stated. See Fig. 3.3 for qualitative examples of our co-segmentation results.

Influence of ReCo framework components. We next assess the effect of employing DenseCLIP during inference, the *language-guided co-segmentation* and *context elimination* components of ReCo which seek to improve the quality of co-segmentation achieved across each archive to boost downstream segmentation performance. When applying context elimination, we select *tree, sky, building, road*, and *person* as common background concepts appearing in natural images to be suppressed. In Table 3.1, we show the effect of three of these strategies, together with the effect of applying a CRF [Krähenbühl and Koltun 2011a] as post-processing. We observe that integrating DenseCLIP into the inference procedure brings a significant gain in performance which we believe is driven by the notable robustness of CLIP features under zero-shot transfer [Radford et al. 2021]. In addition, language-guided co-segmentation and context elimination further boost co-segmentation performance, while the CRF brings a small gain. We therefore use each of these strategies (including CRF post-processing) in the remaining experiments.

Model	Acc.	mIoU
Zero-shot transfer		
DenseCLIP‡ [Zhou et al. 2021]	32.3	19.8
ReCo‡ (Ours)	**46.6**	**27.2**
Unsupervised adaptation		
IIC [Ji et al. 2019]	21.8	6.7
MDC [Cho et al. 2021]	32.2	9.8
PiCIE [Cho et al. 2021]	48.1	13.8
PiCIE + H [Cho et al. 2021]	50.0	14.4
STEGO [Hamilton et al. 2022]	**56.9**	28.2
ReCo+‡ (Ours)	54.5	**33.0**

Model	Acc.	mIoU
Zero-shot transfer		
DenseCLIP⋆‡ [Zhou et al. 2021]	35.9	10.0
MDC⋆† [Cho et al. 2021]	-	7.0
PiCIE⋆† [Cho et al. 2021]	-	9.7
D&S⋆† [Vobecky et al. 2022]	-	16.2
ReCo⋆‡ (Ours)	**65.4**	**22.0**
Unsupervised adaptation		
IIC [Ji et al. 2019]	47.9	6.4
MDC [Cho et al. 2021]	40.7	7.1
PiCIE [Cho et al. 2021]	65.5	12.3
STEGO [Hamilton et al. 2022]	73.2	21.0
ReCo+‡ (Ours)	**83.7**	**24.2**

Model	Acc.	mIoU
Zero-shot transfer		
DenseCLIP‡ [Zhou et al. 2021]	34.1	15.3
ReCo‡ (Ours)	**70.6**	**29.8**
Unsupervised adaptation		
SegSort [Hwang et al. 2019]	69.8	19.2
HSG [Ke et al. 2022]	73.8	21.7
ReCo+‡ (Ours)	**75.3**	**31.9**

Table 3.2: **Comparison to state-of-the art approaches on COCO-Stuff (left), Cityscapes (middle), and KITTI-STEP (right) validation sets.** ⋆Evaluated at the original resolution. †Models trained on Waymo Open [Sun et al. 2020] (reported from [Vobecky et al. 2022]). ‡Models that leverage a language-image pre-training model to assign a concept name to a prediction. The best score for each metric under each protocol is highlighted in **bold**. We observe that ReCo and ReCo+ perform strongly relative to prior work under zero-shot transfer and unsupervised adaptation protocols, respectively.

3.4.4 Comparison to state-of-the-art unsupervised methods

We compare ReCo and ReCo+ to state-of-the-art unsupervised semantic segmentation models with and without vision-language pre-training on standard benchmarks, including COCO-Stuff [Caesar et al. 2018], Cityscapes [Cordts et al. 2016] and KITTI-STEP [Weber et al. 2021] under both zero-shot transfer and unsupervised adaptation (training without labels on the target distribution). For COCO-Stuff, we observe that the mid-level categories used for evaluation are somewhat abstract for retrieval (for instance, one mid-level category is "outdoor objects," which may include many low-level categories beyond the target hierarchy). To avoid introducing ambiguity to the co-segmentation procedure, we instead directly use the low-level categories and then merge the predictions into the mid-level categories. Additionally, we rephrase two category names to reduce ambiguity (*parking* to *parking lot* and *vegetation* to *tree*) in Cityscapes and KITTI-STEP based on the descriptions found in [Cordts et al. 2016]. A detailed discussion can be found in the supplementary material.

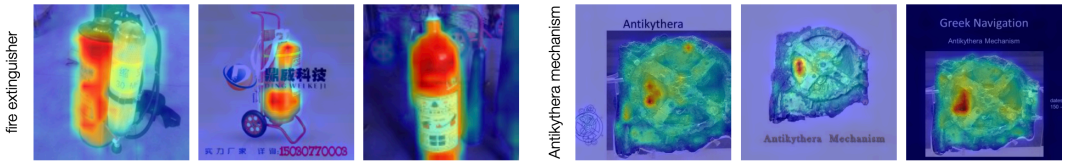


Figure 3.5: **Co-segmentations of rare concepts. Left:** *fire extinguisher*. **Right:** *Antikythera mechanism*. We show selected samples from ReCo archives for each concept, together with their co-segmentations. In each case, ReCo successfully identifies the regions associated with the concept.

As shown in Table 3.2, ReCo strongly outperforms prior models on all benchmarks for zero-shot transfer. Under an unsupervised adaptation protocol, ReCo+ outperforms the state-of-the-art by a large margin on the Cityscapes and KITTI-STEP. On COCO-Stuff, ReCo+ achieves slightly lower pixel accuracy compared to [Hamilton et al. 2022] but a considerably higher mIoU. In Fig. 3.1, we visualise the sample predictions of our models on COCO-Stuff.

3.4.5 Segmenting rare concepts

By virtue of inheriting CLIP's diverse knowledge of nameable visual concepts, ReCo exhibits the ability to segment rare categories. We first demonstrate this ability for *fire extinguisher* objects, which have important fire-safety implications but seldom appear in popular semantic segmentation benchmarks. To assess performance, we evaluate segmentation quality on the FireNet dataset (as described in Section 3.4.1) and achieve reasonable performance on pixel accuracy (93.3) and IoU (44.9) metrics. In Fig. 3.5 (left) we visualise the co-segmentation produced by ReCo across sample images container fire extinguishers. As an additional demonstration (Fig. 3.5, right), we also show co-segmentations for images containing the views of the (unique) *Antikythera mechanism*. In both cases, we observe that ReCo is capable of co-segmenting the concept of interest without labelled examples.

3.5 Limitations

Our work has several limitations: (1) There may be cases for which our first hypothesis does not hold—concepts so rare that they do not appear in billion-image scale datasets. In such cases, *e.g.* "a purple elephant with square orange feet wearing an inverted cowboy hat in front of the Doge's Palace", ReCo will struggle. (2) During inference, we make use of a visual encoder and a pre-trained language-image model, *i.e.* CLIP (a computationally heavy model). Future work could address this with training a light student network which distills knowledge from both models. (3) We employ ImageNet as the unlabelled image collection for our primary investigations, which is known to exhibit an object-centric bias. While qualitative results suggest that ReCo can learn from extremely diverse data [Schuhmann et al. 2022], a more comprehensive empirical evaluation would strengthen this result. (4) Models like CLIP are expensive to train and are consequently updated infrequently. If a new concept emerges (*e.g.* a new product) after CLIP was trained, ReCo will be unable to build an archive to enable co-segmentation for this concept. (5) Although our work is unsupervised in the sense that it is not trained on pixel-level annotation, the ablation studies on PASCAL-Context have guided the development of our method, and thus it benefits from a form of indirect supervision. As a result, our design likely contains choices that subtly favour performance on the evaluation tasks.

3.6 Broader impact

Semantic segmentation is a *dual use* technology. It enables many applications with the potential for significant positive societal impact across domains in medical imaging, wildlife monitoring, improved fault detection in manufacturing processes etc. However, it is also vulnerable to abuse: it may enable unlawful surveillance or invasions of privacy, for example. By removing the requirement to collect pixel

masks for concepts of interests, ReCo lowers the barrier to entry for any individual that wish to make practical use of segmentation, but makes no distinction on the ethical implications of the use case, positive or negative.

ReCo makes use of large-scale, unlabelled image collections. By their nature, such images are subject to minimal curation and sanitisation, and thus may contain not only biases across demographics, but also content that does not align with the ethical values of the user. Consequently, we emphasise that ReCo represents a research proof of concept that is not appropriate for real-world usage without extensive additional analysis of the specific deployment context in which it will be used and, in particular, safeguards to moderate the archive curation process.

3.7 Conclusion

In this work, we introduced the ReCo framework for semantic segmentation zero-shot transfer. By drawing on the strengths of large-scale language-image pre-training and modern visual backbones, ReCo attains the ability to segment rare concepts and to directly assign names to concepts without labelled examples from the target distribution. In addition, experimental comparisons demonstrate that ReCo strongly outperforms existing zero-shot transfer approaches that forgo pixel supervision.

Acknowledgements

This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk). GS is supported by AI Factory, Inc. in Ko-

rea. GS would like to thank Yash Bhalgat, Tengda Han, Charig Yang, Guanqi Zhan, and Chuhan Zhang for proof-reading. SA would like to acknowledge the support of Z. Novak and N. Novak in enabling his contribution.

Appendices

Appendices for this chapter can be found in the online version of the paper.[3]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[3] https://www.robots.ox.ac.uk/~vgg/research/reco

Chapter 4

NamedMask: Distilling Segmenters from Complementary Foundation Models

This work has been accepted for publication at the 2nd Workshop on Learning with Limited Labelled Data for Image and Video Understanding, Conference on Computer Vision and Pattern Recognition (CVPR), 2023.

NamedMask: Distilling Segmenters from Complementary Foundation Models

Gyungin Shin[1,3] Weidi Xie[1,2] Samuel Albanie[3]

[1]Visual Geometry Group, University of Oxford

[2]Coop. Medianet Innovation Center, Shanghai Jiao Tong University

[3]Cambridge Applied Machine Learning Lab, University of Cambridge

https://www.robots.ox.ac.uk/~vgg/research/namedmask

Abstract

The goal of this work is to segment and name regions of images without access to pixel-level labels during training. To tackle this task, we construct segmenters by distilling the complementary strengths of two foundation models. The first, CLIP [Radford et al. 2021], exhibits the ability to assign names to image content but lacks an accessible representation of object structure. The second, DINO [Caron et al. 2021], captures the spatial extent of objects but has no knowledge of object names. Our method, termed NamedMask, begins by using CLIP to construct category-specific archives of images. These images are pseudo-labelled with a category-agnostic salient object detector bootstrapped from DINO, then refined by category-specific segmenters using the CLIP archive labels. Thanks to the high quality of the refined masks, we show that a standard segmentation architecture trained on these archives with appropriate data augmentation achieves impressive semantic segmentation abilities for both single-object and multi-object images. As a result, our proposed NamedMask performs favourably against a range of prior work on five benchmarks including the VOC2012, COCO and large-scale ImageNet-S datasets.

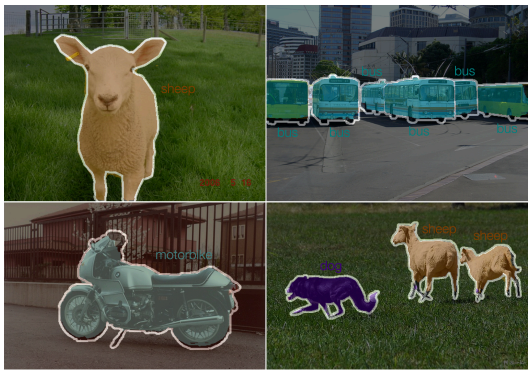


Figure 4.1: We propose NamedMask, a segmenter distilled from the namedmask capabilities of CLIP and DINO. *With no access to pixel-level annotation*, NamedMask not only accurately segments single objects (left) but also multiple objects (right) within an image. White pixels denote ignored regions. The images are drawn from the VOC2012 benchmark.

4.1 Introduction

Semantic segmentation is a task that entails grouping and naming coherent regions of images. It has a broad range of applications spanning domains such as autonomous driving, manufacturing and medicine. A key barrier to automating this task through supervised learning is the requirement for pixel-level segmentation annotations, which can be extremely costly to obtain (*e.g.*, 1.5 hours per image when accounting for quality control [Cordts et al. 2016]).

The emerging paradigm of *foundation models* (models that have been pretrained on broad data and can be adapted to a wide range of downstream tasks) has yielded striking gains for many machine perception problem domains [Bommasani et al. 2021]. There is therefore considerable interest in determining whether such models can alleviate the prohibitive annotation costs associated with semantic segmentation. In this vein, MaskCLIP [Zhou et al. 2022] demonstrated the potential benefits of leveraging the representation learned by CLIP [Radford et al. 2021] to perform "annotation-free"[1] segmentation with no prior knowledge of the

[1]This setting can be equivalently referred to as *zero-shot transfer* in the terminology of [Radford et al. 2021].

target domain. However, unless segmentation masks are available from at least some of the categories to guide pseudo-labelling, segmentation quality remains far from supervised performance. ReCo [Shin et al. 2022b] considered an alternative formulation in which only the *names* of the target categories (but no images from the target domain) are available during training. By coupling the retrieval capabilities of CLIP to curate archives of images belonging to specific categories with a co-segmentation algorithm, ReCo obtained improvements over MaskCLIP but still struggles to produce precise object segmentations.

In this work, we build upon the ReCo framework and revisit the mechanism through which it obtains pseudo-masks with semantic labels. Drawing inspiration from recent work showing that DINO features can be employed to perform unsupervised salient object detection [Shin et al. 2022a; Melas-Kyriazi et al. 2022a; Wang et al. 2022b], our first step is to replace the fragile co-segmentation of ReCo with a more robust category-agnostic object segmentation facilitated by DINO. We then exploit the naming capabilities of CLIP to assign the category label from each archive of images to these segmentations to enable the training of category-specific "expert" segmenters that refine the quality of the archive segmentations. Finally, we train a single semantic segmentation model on the resulting collection that is capable of segmenting objects from any category that is represented in the archives, using copy-paste augmentation [Ghiasi et al. 2021a] to improve generalisation to images of multiple objects. We show that our approach, which we term NamedMask, achieves substantial gains in performance for semantic segmentation of objects (see Fig. 4.1 for examples).

Our contributions are: (1) We propose NamedMask, a framework for segmenting and naming objects without pixel-level annotation by distilling the complementary strengths of CLIP and DINO; (2) We provide extensive experiments to demonstrate the improvements brought by NamedMask over prior semantic segmentation approaches that also make use of language-image pretraining.

4.2 Related work

Our approach relates to prior work on *unsupervised semantic segmentation*, *semantic segmentation with language-image pretraining* and *salient object detection*. We discuss connections to each of these next.

Unsupervised semantic segmentation. By coupling deep neural networks with creative learning objectives, substantial progress has been made towards unsupervised semantic segmentation. Examples of learning signals that have been constructed without labels include expectation-maximisation over segments [Hwang et al. 2019], mutual information maximisation [Ji et al. 2019; Ouali et al. 2020b], contrasting proposals [Van Gansbeke et al. 2022], complementary signals from LiDAR and vision [Vobecky et al. 2022] and feature correspondence distillation [Hamilton et al. 2022]. In contrast to name-only segmentation, these methods do not make use of language-image pretraining or require access to the target category list during training. They do, however, require the use of a small number of images labelled with segments (typically the test set itself) together with the Hungarian algorithm to assign names to segment predictions, or otherwise employ nearest-neighbour lookup on a held-out set of images with segmentation masks.

Annotation-free semantic segmentation using language-image model. Several recent works have sought to leverage the zero-shot transfer capabilities of CLIP [Radford et al. 2021] to perform semantic segmentation with no access to paired data (images labelled with categories or segments) from the target domain. MaskCLIP [Zhou et al. 2022] illustrated the potential of using CLIP for semantic segmentation in a zero-shot transfer setting (a setting that they term "annotation-free"). A recent example of such line of research is ReCo [Shin et al. 2022b], which curates unlabelled images into examples of concepts with CLIP, then applies a co-segmentation algorithm to derive semantic segmentation training data. While ReCo achieves promising results, it fails to coherently pseudo-label objects and thus (as we show through experiments) does not lead to high-quality object

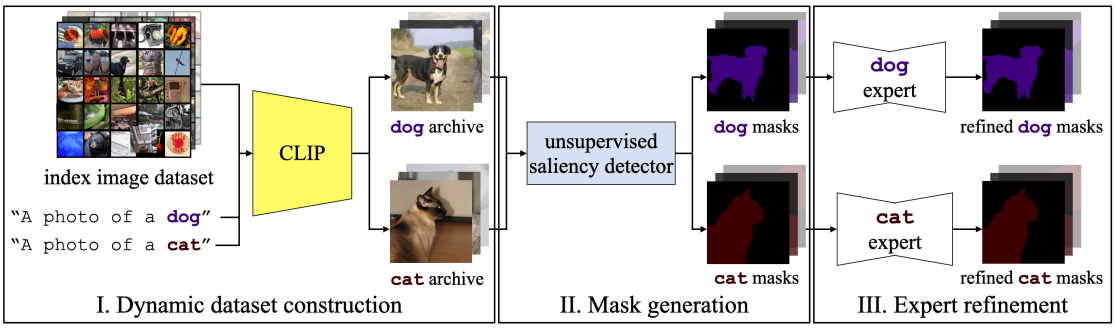


Figure 4.2: Overview of the proposed pipeline used to construct the NamedMask training dataset for semantic segmentation. Given an image archive for a concept retrieved by CLIP (left), we generate masks using an unsupervised saliency detector (middle). We refine the segmentations of each category by a class expert trained with the constructed image-mask pairs (right). Using the retrieved images and their refined segments, we train NamedMask to generate a segmenter capable of predicting a set of pre-defined categories (omitted in the figure for simplicity).

segmentations. In this work, we compare directly with ReCo and demonstrate the substantial gains in performance that can be attained by bootstrapping the category-agnostic pseudo-labels enabled by DINO.

Unsupervised salient object detection. A range of work has sought to perform salient object detection (the task of segmenting foreground objects) without human annotation [Zhang et al. 2017; Bielski and Favaro 2019; Voynov and Babenko 2020]. One notable trend amongst recent approaches has been the application of spectral clustering in combination with self-supervised features [Melas-Kyriazi et al. 2022a; Wang et al. 2022b; Shin et al. 2022a]. In this work, we build on the SelfMask approach of [Shin et al. 2022a] to provide a robust category-agnostic segmenter for NamedMask.

4.3 Method

In this section, we formulate the semantic segmentation task considered in this work (Section 4.3.1) and the method, NamedMask, that we propose to tackle it (Section 4.3.2).

4.3.1 Task formulation and terminology

Our objective is to perform *semantic segmentation*: for a given image, $x \in \mathbb{R}^{3\times H\times W}$, we aim to assign a label, c, from among a finite set of categories, $\mathcal{C}$, to each pixel location $\omega \in \{1,\dots,H\} \times \{1,\dots,W\}$ of x. To facilitate cost-effective scaling, we aim to do so without access to any form of pixel-level annotation. To this end, we propose to exploit the perceptual grouping of objects and their semantic categorisation offered by two foundation models. Specifically, we leverage the semantic categorisation capabilities of CLIP derived through large-scale language-image pretraining and the perceptual grouping capabilities of DINO derived from vision-only pretraining.

Terminology. To date, a wide array of methods have been proposed to tackle the problem of semantic segmentation with different levels of supervision (*fully unsupervised*, *unsupervised but with supervised pretraining*, *weakly-supervised* etc.). However, the terminologies used to describe these levels of supervision are not always clear or consistent. We therefore first aim to clarify the annotation regime in which we operate and how it is closely related to prior work.

In particular, we consider a setting that we term *Segmentation Leveraging Only Weak Pretraining* (SLOWP). SLOWP methods make no use of pixel-level annotation and are characterised by pretraining on data that is either: (1) *weakly-labelled* (e.g. with class labels or alt-text) and does not derive from the target distribution; or (2) *unlabelled* and may or may not derive from the target distribution. Within the space of SLOWP methods, we further distinguish three sub-categories that more precisely characterise the knowledge that the method possesses about the segmentation task used to evaluate the model: (i) *Zero-shot transfer* assumes no knowledge of the target distribution (images or category names) during training; (ii) *Name-only transfer* assumes access (during training) to the list of category names that are to be used for the target segmentation task, but does not assume access to any images from the target distribution; (iii) *Name-and-image trans-*

fer assumes access (during training) to the list of category names in the target segmentation task *and* access to unlabelled images from the target distribution.

To relate these categories to prior work, note that MaskCLIP [Zhou et al. 2022] represents a *SLOWP zero-shot transfer* method: it uses language-image pretraining (via CLIP) and does not make use of the target categories during training. ReCo [Shin et al. 2022b] typically represents a *SLOWP name-only transfer* method: it uses language-image pretraining (via CLIP) and image classification pretraining (via DeiT-S/SIN [Naseer et al. 2021]) and has access to target category names for constructing classifiers.

In this work, we propose a method, NamedMask, that operates effectively in both the *SLOWP name-only transfer* and *SLOWP name-and-image transfer* scenarios. We describe NamedMask next.

4.3.2 NamedMask

NamedMask is trained in a sequence of four stages: (1) For a given list of target categories, we perform dynamic dataset construction by curating archives of images for each category from an unlabelled image collection using CLIP; (2) For each image in each archive, we predict a category-agnostic object mask with an unsupervised saliency detector; (3) We refine the predicted masks with a category-specific "expert" segmenter, which is self-trained with the generated image-mask pairs within each archive; (4) We distill a segmenter using the image archives and their refined masks as pseudo-labels. An overview of the first three stages is provided in Fig. 4.2, and each stage is detailed in the following.

Dynamic archive construction. To create a data set containing images of categories of interest, we follow the approach proposed by ReCo [Shin et al. 2022b] and curate an archive of images for each concept using an image-language model (*i.e.* CLIP). Formally, given an image encoder $\phi_{\mathcal{I}}$ and a text encoder $\phi_{\mathcal{T}}$ of CLIP, we curate one archive from an unlabelled image collection $\mathcal{U}$ for each category c of interest. We do so by selecting the n images among the collection whose visual

embeddings $\phi_{\mathcal{I}}(x_i) \in \mathbb{R}^e$ lie closest to the text embedding[2] $\phi_{\mathcal{T}}(c) \in \mathbb{R}^e$ of c. That is,

$$\mathcal{A}_c = \{x_i \mid i \in \arg\ \text{top}k\ [\phi_{\mathcal{I}}(\mathcal{U}) \cdot \phi_{\mathcal{T}}(c)]\}, \tag{4.1}$$

where $\mathcal{A}_c$ denotes the image archive for category c and arg top$k$ returns the indices of its arguments with the k largest values. In this way, we dynamically construct a data set comprising a collection of $|\mathcal{C}|$ archives (one for each category).

Mask generation. To produce category-agnostic object segmentations for the images within each archive, we adopt the SelfMask [Shin et al. 2022a] unsupervised salient object detection method. SelfMask learns to perform salient object detection by first performing spectral clustering on DINO features across unlabelled images, then using these clusters as pseudo-labels to train a variant of MaskFormer segmenter [Cheng et al. 2021b]. Given the SelfMask saliency detector ψ_s, we first predict a category-agnostic saliency map $y_i = \psi_s(x_i) \in \{0,1\}^{H \times W}$ for each image $x_i \in \mathbb{R}^{3 \times H \times W}$ in each archive. We then simply assign to each category-agnostic saliency map the category label c of the archive that contains the image. This produces a collection of images annotated with saliency masks and corresponding category labels.

Mask refinement through category experts. The category-agnostic saliency detector employed in the previous stage is unaware of the category of objects that it is being used to segment. We hypothesise that a segmenter will produce superior segmentations when it is given knowledge of the specific category of objects that it is required to segment, and thus will produce improved pseudo-masks for training NamedMask. To instantiate this idea, we refine the category-agnostic predictions made by the saliency detector with a segmenter ψ_c, which specialises in segmenting regions corresponding to category c. To this end, we train a segmenter ψ_c to assign regions to either the category c or the background class for each image in $\mathcal{A}_c$, as a pixel-level one-vs-all binary classification task. We then use the predictions

[2]Details of the prompt used to construct the text embedding can be found in the supplementary material.

obtained by ψ_c as pseudo-masks for category c. We show through experiments in Section 4.4.3 that this simple approach yields superior segmentation training data relative to using SelfMask predictions directly.

Note that for cases when there are a large number of target categories (*e.g.* 919 categories in ImageNet-S [Gao et al. 2021]), training one expert per class can be computationally expensive. For such cases, we group the relevant categories by applying k-means clustering to the text embeddings of the categories extracted from a CLIP text encoder and train an expert for each category group.

Training NamedMask. Given the resulting collection of image archives annotated by category-specific segmenters, NamedMask is produced by simply training a standard semantic segmentation architecture using a cross-entropy loss. Thus, self-training produces a segmenter that exploits the complementary information encoded by two different foundation models, where the visual-only model (*i.e.* DINO) implicitly captures the perceptual grouping of objects, and the ability to name categories derives from the language-image model (*i.e.* CLIP).

4.4 Experiments

In this section, we begin by describing the datasets considered for our experiments, implementation details, and our ablation study. We then compare our model to state-of-the-art unsupervised semantic segmentation (USS) methods and approaches that leverage only weak pretraining (SLOWP).

4.4.1 Datasets

Evaluation benchmarks. We consider five segmentation benchmarks including COCO [Lin et al. 2014], CoCA [Zhang et al. 2020], Cityscapes [Cordts et al. 2016], PASCAL VOC2012 [Everingham et al. 2015], and ImageNet-S [Gao et al. 2021]. COCO consists of 118,287 and 5,000 images for train and validation splits with 80 object categories and a background class and CoCA comprises 1,295 images

model	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	dining-table	dog	horse	motorbike	person	potted plant	sheep	sofa	train	tv/monitor	**avg.**
SelfMask	78.5	26.8	68.0	59.2	22.9	69.3	43.1	80.7	14.0	70.2	20.3	74.2	70.1	57.7	41.5	20.6	74.3	18.4	64.7	25.0	50.0
experts	80.2	29.0	72.9	65.0	30.4	74.9	48.9	82.4	15.8	77.8	27.2	75.2	74.5	62.1	43.1	21.3	74.9	26.7	69.8	36.4	54.4

Table 4.1: **Category experts produce better quality segmentation masks than the baseline unsupervised saliency detector.** We report segmentation performance for each method on Pascal VOC2012. The performance metric is (class-wise) IoU (%).

of 80 object categories with a background. Cityscapes contains 2,975 and 500 urban scene images for training and validation splits with 30 categories among which we pick 14 object categories to evaluate based on the original paper [Cordts et al. 2016]. VOC2012 is composed of 1,464 training and 1,449 validation images with 21 categories including background, and the large-scale ImageNet-S [Gao et al. 2021] dataset comprises 9,190 train, 12,419 validation, and 27,423 test images with precise pixel-level annotations. There are three variations of ImageNet-S: ImageNet-S_{50}, ImageNet-S_{300}, and ImageNet-S_{919}, consisting of 50, 300, and 919 semantic categories of ImageNet1K [Deng et al. 2009], respectively.

We use the VOC2012 train split and the ImageNet-S_{300} validation split for our ablation studies, and compare our models to previous USS and SLOWP methods on CoCA, the validation split of COCO, Cityscapes, and VOC2012, and the test split of ImageNet-S.

Image collections. To curate image archives for each category, we use two unlabelled image collections: (1) For experiments on PASCAL VOC2012, we use the ImageNet1K training set without labels, following [Shin et al. 2022b]. (2) For experiments on ImageNet-S, we use unlabelled images from LAION-5B [Schuhmann et al. 2022]. For the latter, we implement the archive curation process using the CLIP feature index provided with the LAION-5B release[3]. Since the LAION-5B dataset was collected with limited manual curation, we apply a face detector to all images and discard any image containing a visible human face [Deng et al. 2020]. We refer the reader to the supplementary material for further details about the

[3] https://laion.ai

usage of the LAION-5B dataset.

4.4.2 Implementation details

We conduct the experiments on a single A100 NVIDIA graphic card with PyTorch [Paszke et al. 2019]. Code will be made publicly available.

Network architecture and optimisation. We use DeepLabv3+ [Chen et al. 2018b] with a ResNet50 [He et al. 2016] backbone for both category experts and NamedMask. We initialise the backbone with DINO [Caron et al. 2021] that is pretrained on ImageNet [Russakovsky et al. 2015] in a self-supervised manner. For expert training, we adopt a lightweight learning schedule of 5K gradient updates with a batch size of 8 for COCO, CoCA, Cityscapes, and VOC2012 and 10K updates with a batch size of 16 for ImageNet-S. For NamedMask, we train the model with 20K iterations for COCO, CoCA, Cityscapes, and VOC2012 and 80K iterations for ImageNet-S. We use standard data augmentations such as random scaling, random cropping and colour jittering. We use Adam optimiser with an initial learning rate of 0.0005 and a weight decay of 0.0002. We decay the learning rate with the Poly learning rate [Chen et al. 2018b].

To curate category archives from ImageNet and LAION-5B, the ViT-L/14@336px and ViT-L/14 variants of CLIP are employed repectively. For our unsupervised saliency detection method, we adopt the model from SelfMask [Shin et al. 2022a], and apply a bilateral solver [Barron and Poole 2016] to predictions from SelfMask as a post-processing step.

Inference. When evaluating on ImageNet-S, images are resized such that their larger dimension is 1024 pixels while preserving their aspect ratio. For evaluation, the predictions of the model are then resized back to the original resolution to match the ground-truth mask by using a bilinear upsampler. For the ImageNet-S_{300} validation set (used in our ablation study), we resize the shorter side of images to 384 with a maximum length for the larger dimension of 512 pixels. For the other benchmarks, we use the original resolution of the images.

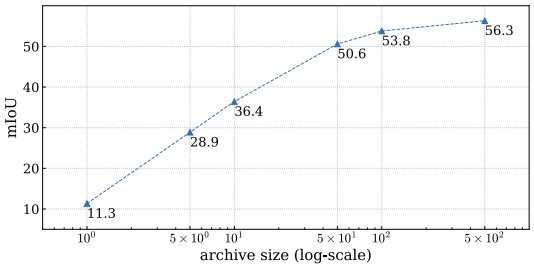


Figure 4.3: **Larger image archives produce better segmenters.** Here, "archive size" denotes the number of images retrieved by CLIP to curate an image archive for each category.

Metric. Following the common practice, we employ intersection-over-union (IoU) to measure a class-agnostic mask quality and mean IoU (mIoU) to evaluate the performance of semantic segmentation.

4.4.3 Ablation study

In this section, we present a thorough ablation study on each component of our proposed NamedMask, namely, the influence of archive size, the influence of adopting category experts and the effect of the number of category experts. We also investigate the influence of adopting copy-paste augmentation for segmenting images with multiple objects.

Effect of archive size. Unlike supervised approaches for which it is costly to acquire annotations, the dataset creation process for NamedMask can be easily scaled. To investigate the influence of the number of images used for training, we vary the size of archive curated by CLIP for each category, from 1 to 500 images, and train NamedMask on the resulting images with corresponding pseudo-labels obtained from SelfMask. As for quantitative evaluation, we adopt the VOC2012 training set and report numbers in Fig. 4.3. Note that, the training is done only on the constructed archive of ImageNet images, with pseudo labels acquired from SelfMask (*i.e.*, no category experts have been introduced at this stage).

As shown in Fig. 4.3, that the archive size plays an important role in the performance of our model, monotonically increasing with the number of images

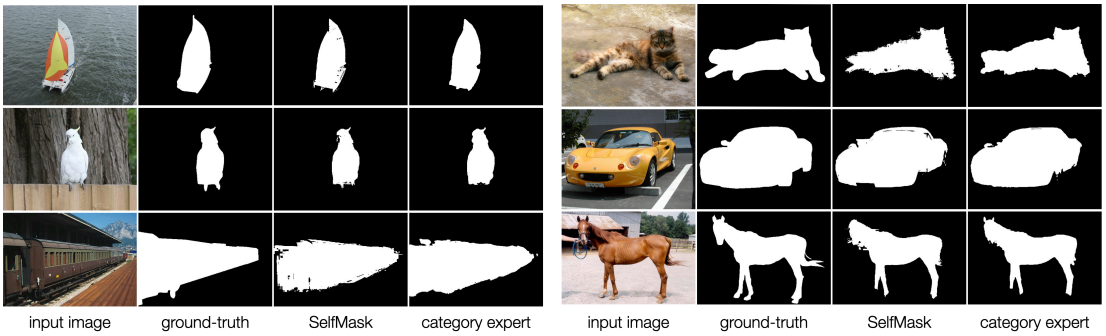


Figure 4.4: **Category experts refine the masks provided by an unsupervised saliency detector (*i.e.* SelfMask).** The images are selected from VOC2012. Zoom in for details.

model	# experts	avg. IoU
SelfMask [Shin et al. 2022a]	-	62.7
category experts	1	63.3
	30	**64.1**
	60	64.0
	90	63.9

Table 4.2: Effect of grouping semantically relevant categories for category expert training on the ImageNet-S_{300} validation split.

for an archive. For the remaining experiments, we curate 500 images per archive.

Effect of category experts on mask quality. As described in Section 4.3.2, we propose to refine the pseudo-labels from SelfMask with category-specific experts, which are trained to distinguish foreground and background pixels.

To compare the quality of category-agnostic masks generated by SelfMask and class-specific masks by an expert, we evaluate their predictions on 20 object categories from the VOC2012 train split. Specifically, we train 20 category experts on image archives constructed by retrieving from ImageNet1K. In Table 4.1, we show the (class) IoU of each category. We observe that the experts consistently outperform SelfMask across all categories. For a qualitative comparison, we visualise the examples predictions from both SelfMask and category experts in Fig. 4.4. In the following experiments, we produce pseudo-masks from an expert for each category by default.

model	copy-paste	single obj.	multi-obj.	all
SelfMask + CLIP	-	63.3	42.1	50.4
NamedMask (Ours)	✗	67.0	50.5	56.6
NamedMask (Ours)	✓	**68.0**	**53.6**	**58.7**

Table 4.3: **Copy-paste augmentation helps the model to segment multiple objects in an image.** The performance is measured in mIoU (%). A baseline model is marked in grey. Best score for each column is highlighted in **bold**.

Training an expert for a category group. When there are numerous classes that are semantically close to one another, training individual category experts may become prohibitively expensive. We therefore group categories by applying k-means to text embeddings of the categories into k concept groups as described in Section 4.3.2.

To show the effect of k, we evaluate the category experts, by varying the number of category groups on the ImageNet-S_{300} validation set. In Table 4.2, we report average IoU over 300 categories for k={30, 60, 90} which corresponds to 10%, 20%, and 30% of the total number of classes, respectively. As baselines, we also show average IoU for SelfMask and a single expert for 300 categories (*i.e.*, k=1). As can be seen, training an expert always shows higher score than the baseline, even when k is set to 1. However, when we set the number of experts higher than 30, the performance appears to saturate. We conjecture that this is because there is a trade-off between the number of different images which an expert can see during training and average semantic relevance within a category group. That is, when k is large, the average number of categories assigned to one group tends to be low, which decreases the total number of images used for training an expert for the group. In contrast, when k is small, the total number of images for an expert become large at the cost of reduced semantic relevance in a class group. For this reason, we group categories of ImageNet-S_{300} and -S_{919} into 30 and 90 which (approx.) accounts for 10% of total classes, respectively.

Effect of copy-paste on segmenting multiple objects. In contrast to the

model	dataset
MaskCLIP [Zhou et al. 2022]	WebImageText
ReCo [Shin et al. 2022b]	WebImageText, (Stylized-)ImageNet
NamedMask (Ours)	WebImageText, ImageNet

Table 4.4: Datasets employed in training each model. ReCo utilises both Stylized-ImageNet and ImageNet.

model	transfer type	COCO	CoCA	Cityscapes$_{obj}$
MaskCLIP [Zhou et al. 2022]	zero-shot	5.3	3.1	6.1
ReCo† [Shin et al. 2022b]	name-only	17.1	16.9	14.1
NamedMask	name-only	**28.4**	**27.3**	**18.2**

Table 4.5: Comparison to previous segmentation leveraging only weak pre-training (SLOWP) methods on the COCO, CoCA, and Cityscapes$_{obj}$ benchmarks in terms of mIoU. Highest scores on each benchmark are in **bold**. †initialises the backbone with Stylized-ImageNet pre-training.

salient object detectors and category experts which segment an object of a single category or a group of similar categories within an image, our model can be readily trained to segment multiple objects of different categories by employing the copy-paste augmentation [Ghiasi et al. 2021a]. To demonstrate this, we evaluate two NamedMask models, trained with and without copy-paste augmentation on the images containing a single object or multi-objects from possibly different categories. As a baseline, we also evaluate segmentation of SelfMask whose semantic label is decided by applying CLIP to a given image. As shown in Table 4.3, when validated on the VOC2012 training split, copy-paste brings a notable gain in performance by 2.1 mIoU compared to the model trained without copy-paste. We therefore adopt the copy-paste augmentation as a default setting in the remaining experiments.

4.4.4 Comparison to state-of-the-art methods

To describe the effectiveness of our approach, we compare NamedMask against existing approaches that fall into the proposed segmentation leveraging only weak

model	transfer type	backbone	mIoU
USS			
Inst. Disc. [Wu et al. 2018]	-	ResNet50	4.3
MoCo [He et al. 2020]	-	ResNet50	3.7
InfoMin [Tian et al. 2020]	-	ResNet50	4.4
SwAV [Caron et al. 2020]	-	ResNet50	4.4
MaskCon. [Van Gansbeke et al. 2021]	-	ResNet50†	35.0
MaskDist. [Van Gansbeke et al. 2022]	-	ResNet50†	**45.8**
SLOWP			
MaskCLIP* [Zhou et al. 2022]	zero-shot	ResNet50	29.1
ReCo*‡ [Shin et al. 2022b]	name-only	DeiT-S/16	34.2
NamedMask	name-only	ResNet50	**59.2**

Table 4.6: Comparison to existing unsupervised semantic segmentation (USS) and segmentation leveraging only weak pre-training (SLOWP) methods on the PASCAL VOC2012 validation set. Numbers for *USS* methods are from MaskDistill. *Re-implemented and adapted by us to predict a background class. †uses dilated ResNet [Yu and Koltun 2016]. ‡initialises the backbone with Stylized-ImageNet [Geirhos et al. 2019] pre-training. Highest scores of each kind of methods are in **bold**.

pretraining (SLOWP) setting. Specifically, we consider and re-implement MaskCLIP [Zhou et al. 2022] with the *zero-shot transfer* setting (*i.e.* the annotation-free setting in their paper) and ReCo [Shin et al. 2022b] with the *name-only transfer* or and *name-and-image transfer* setting. As described in Section 4.3.1, the transfer type of each SLOWP method is determined by whether it has access to either category names or unlabelled images from the evaluation benchmark during training. As such, the transfer type of a method varies with its evaluation benchmark (see Table 4.4 for datasets of which categories and images each SLOWP approach has access to during training).

As MaskCLIP and ReCo do not explicitly define background categories, we classify the pixels as background if their highest class probability is lower than a certain threshold t. We set t as 0.25 and 0.9 for MaskCLIP and ReCo (see the supp. mat. for more details on how t is selected).

We evaluate on popular segmentation benchmarks including COCO, CoCA,

VOC2012, and large-scale ImageNet-S datasets. Additionally, we also evaluate on the object categories (*e.g.*, car, person) in Cityscapes (denoted Cityscapes$_{obj}$). In addition to SLOWP methods, we also compare with state-of-the-art unsupervised semantic segmentation (USS) including MaskContrast [Van Gansbeke et al. 2021] and MaskDistill [Van Gansbeke et al. 2022] on the VOC2012 and ImageNet-S benchmarks, as they share with SLOWP the similar goal of training without manual annotations.

In Table 4.5, we compare NamedMask to previous SLOWP methods on COCO, CoCA, and Cityscapes$_{obj}$ benchmarks. We make two observations: (i) ReCo and NamedMask, which have access to the category names, outperform MaskCLIP, which is unaware of the concepts of the target benchmarks during training; (ii) when comparing the two name-only transfer methods, NamedMask performs better than ReCo by a large margin on each dataset.

In Table 4.6, we report the results of NamedMask on the VOC2012 validation split. Our approach shows favourable performance over the existing models for both SLOWP and USS. In detail, while the previous SLOWP methods fall behind the state-of-the-art USS models, NamedMask outperforms them by some (≈13.4 mIoU). We also observe that the proposed method is competitive on ImageNet-S, which consists of significantly more number of categories than VOC2012. Here, NamedMask corresponds to the *name-and-image transfer* setting since it has implicit access to unlabelled images from the target distribution through its use of SelfMask (which is bootstrapped from DINO). Similarly, ReCo is categorised as *name-and-image transfer*, as it uses ImageNet1K training images for constructing classifiers. With the caveat that each method has access to different information, NamedMask outperforms the state-of-the-art methods by 15.5, 14.7, and 11.9 mIoU on ImageNet-S$_{50}$ (in Table 4.7), ImageNet-S$_{300}$ (in Table 4.8), and ImageNet-S$_{919}$ (in Table 4.9), respectively.

For qualitative results, we show sample visualisations of our method in Fig. 4.1. More visualisation examples including failure cares are shown in the supplementary

model	transfer type	mIoU	S	MS	ML	L
USS						
MDC [Cho et al. 2021]	-	3.6	0.4	2.6	3.8	4.9
MDC† [Cho et al. 2021]	-	14.3	2.6	10.9	14.6	19.1
PiCIE [Cho et al. 2021]	-	4.5	0.2	3.1	5.0	5.3
PiCIE† [Cho et al. 2021]	-	17.6	4.4	13.1	20.1	23.1
MaskCon. [Van Gansbeke et al. 2021]	-	24.2	**12.2**	25.6	24.7	20.4
LUSS_s [Gao et al. 2021]	-	29.3	6.6	25.0	33.2	32.6
LUSS_p [Gao et al. 2021]	-	**32.0**	9.7	**26.2**	**36.5**	**40.5**
SLOWP						
MaskCLIP* [Zhou et al. 2022]	zero-shot	17.9	3.6	13.1	18.6	20.6
ReCo*† [Shin et al. 2022b]	name & image	22.6	10.0	24.6	22.1	18.8
NamedMask	name & image	**47.5**	**23.5**	**48.7**	**49.3**	**38.0**

Table 4.7: Comparison to existing *USS* and *SLOWP* methods on the ImageNet-S_{50} benchmark. Scores for *USS* are drawn from LUSS [Gao et al. 2021]. We also report mIoU under different object sizes from small (S), medium-small (MS), medium-large (ML), and large (L). *Re-implemented and adapted by us to predict a background class. †initialises the encoder with supervised ImageNet (for MDC and PiCIE) or Stylized-ImageNet pre-training (for ReCo). Best score for each column within a same method type is in **bold**.

model	transfer type	mIoU	S	MS	ML	L
USS						
LUSS_s [Gao et al. 2021]	-	16.0	2.8	12.0	16.4	21.7
LUSS_p [Gao et al. 2021]	-	**18.1**	**4.2**	**13.6**	**19.5**	**23.5**
SLOWP						
MaskCLIP* [Zhou et al. 2022]	zero-shot	1.6	0.4	0.6	1.2	2.5
ReCo*† [Shin et al. 2022b]	name & image	8.5	5.4	9.7	8.4	5.6
NamedMask	name & image	**32.8**	**9.9**	**29.1**	**34.9**	**26.0**

Table 4.8: Evaluation on the ImageNet-S_{300} benchmark. *Re-implemented and adapted by us to predict a background class. †initialises the encoder with Stylized-ImageNet pre-training.

material.

model	transfer type	mIoU	S	MS	ML	L
USS						
LUSS_s [Gao et al. 2021]	-	6.6	1.3	4.6	7.1	8.4
LUSS_p [Gao et al. 2021]	-	**11.0**	**2.4**	**8.3**	**11.9**	**13.4**
SLOWP						
MaskCLIP* [Zhou et al. 2022]	zero-shot	0.5	0.1	0.2	0.3	0.8
ReCo*† [Shin et al. 2022b]	name & image	3.8	2.6	4.6	3.6	2.5
NamedMask	name & image	**22.9**	**5.1**	**19.4**	**24.4**	**19.8**

Table 4.9: Evaluation on the ImageNet-S_{919} benchmark. *Re-implemented and adapted by us to predict a background class. †initialises the encoder with Stylized-ImageNet pre-training. NamedMask is able to segment reasonably well even when numerous categories (*i.e.*, 919 classes) are present in the target dataset.

4.5 Limitations

We note several limitations of our approach: (1) We need to train a new segmenter each time we wish to include another category which is not considered in the previous training of NamedMask. Future work could potentially address this by developing a segmenter that directly predicts embeddings in the shared textual embedding space of CLIP. These could subsequently be used for naming predictions beyond the categories seen during training (*i.e.* generalisation to unseen categories without retraining). (2) While we primarily focus on object semantic segmentation by leveraging an unsupervised saliency detector, it would strengthen our approach to incorporate cues to segment "stuff" categories such as water, sky, etc. This could potentially be done by building prior work such as ReCo or MaskCLIP, which are capable of predicting stuff categories, into our pseudo-label generation step. (3) We note that NamedMask struggles to differentiate a category which often appears with another concept. For example, a rider of a motorbike sometimes is classified as part of the motorbike. We conjecture that this is due to the use of a (unsupervised) saliency detector for pseudo-mask generation which highlights dominant regions in an image without account for a semantic of interest. We believe this can be alleviated by considering a semantic prediction for an expected category in an

image (similary to the language-guided attention mechanism used in ReCo [Shin et al. 2022b]) in addition to the prediction made by the saliency detector.

4.6 Broader impact

NamedMask distills segmenters from foundation models. While powerful, these models have been shown to exhibit biases across different racial and religious groups [Bommasani et al. 2021]. It is therefore likely that NamedMask inherits these biases to some degree. As such NamedMask represents a research prototype that is not appropriate for real-world usage without additional consideration of the deployment setting and the design of appropriate mitigation mechanisms.

NamedMask aims to achieve semantic segmentation with a methodology that can be scaled up without the prohibitive cost of manually-collected segment annotation. In doing so, we hope that it will help enable the deployment of semantic segmentation for applications that yield positive societal impact. As with many powerful computer vision technologies, however, NamedMask is a tool that is subject to *dual use* and is therefore vulnerable to abuse. We are likely unable to anticipate all such possible abuses, but examples could include applications that entail unlawful surveillance.

4.7 Conclusion

In this work, we introduced NamedMask, a method for semantic segmentation that is trained by distilling the complementary capabilities of two foundation models, CLIP and DINO, into a single segmenter. By doing so, NamedMask achieves impressive segmentation quality across both single-object and multi-object images without pixel-level annotation. We demonstrate the effectiveness of NamedMask by comparing to prior methods on several standard semantic segmentation benchmarks including the large-scale ImageNet-S_{919} dataset, where we observe

that NamedMask achieves a significant boost in segmentation performance.

Acknowledgements

This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk). GS is supported by AI Factory, Inc. in Korea. GS would like to thank Guanqi Zhan for proof-reading. SA would like to acknowledge the support of Z. Novak and N. Novak in enabling his contribution.

Appendices

Appendices for this chapter can be found in the online version of the paper.[4]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[4] https://www.robots.ox.ac.uk/~vgg/research/namedmask

Chapter 5

Zero-shot Unsupervised Transfer Instance Segmentation

The paper has been accepted for publication as an oral presentation and awarded the Best Paper Award at the 2nd Workshop on Learning with Limited Labelled Data for Image and Video Understanding, Conference on Computer Vision and Pattern Recognition (CVPR), 2023.

Zero-shot Unsupervised Transfer Instance Segmentation

Gyungin Shin[1,2] Samuel Albanie[2] Weidi Xie[1,3]

[1]Visual Geometry Group, University of Oxford

[2]Cambridge Applied Machine Learning Lab, University of Cambridge

[3]Coop. Medianet Innovation Center, Shanghai Jiao Tong University

https://www.robots.ox.ac.uk/~vgg/research/zutis

Abstract

Segmentation is a core computer vision competency, with applications spanning a broad range of scientifically and economically valuable domains. To date, however, the prohibitive cost of annotation has limited the deployment of flexible segmentation models. In this work, we propose **Z**ero-shot **U**nsupervised **T**ransfer **I**nstance **S**egmentation (ZUTIS), a framework that aims to meet this challenge. The key strengths of ZUTIS are: (i) no requirement for instance-level or pixel-level annotations; (ii) an ability of zero-shot transfer, *i.e.*, no assumption on access to a target data distribution; (iii) a unified framework for semantic and instance segmentations with solid performance on both tasks compared to state-of-the-art unsupervised methods. While comparing to previous work, we show ZUTIS achieves a gain of **2.2 mask AP** on COCO-20K and **14.5 mIoU** on ImageNet-S with 919 categories for instance and semantic segmentations, respectively. The code is made publicly available.[1]

[1]https://github.com/NoelShin/zutis

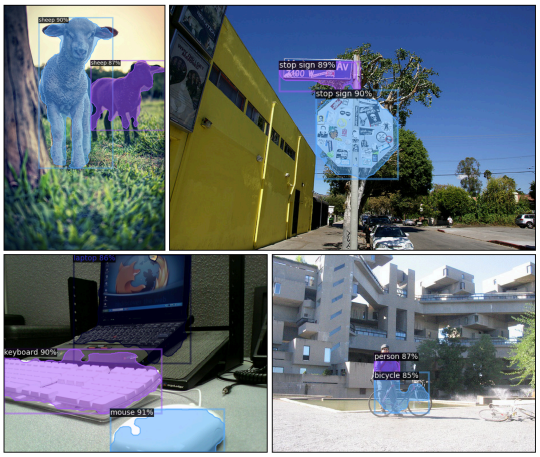


Figure 5.1: We propose ZUTIS, a framework for zero-shot unsupervised transfer instance segmentation. The figure depicts instance segmentations made by ZUTIS on COCO-20K [Vo et al. 2020] and VOC2012 [Everingham et al. 2010]. Without pixel-level annotation or access to the target distribution, ZUTIS acquires the ability to reliably segment instances within an image.

5.1 Introduction

In computer vision, the task of segmentation aims to group pixels within an image into coherent, meaningful regions. Accurate segmentation unlocks a host of applications such as tumour assessment in medical images [Bilic et al. 2019], land cover estimation [Tong et al. 2020] for logistical planning, scene segmentation for autonomous driving [Cordts et al. 2016], to name a few. The central challenge that limits the deployment of such applications is the high cost of obtaining large, accurate collections of pixel-level annotations to train appropriate segmenters. For example, it was reported that when constructing the Cityscapes dataset, it took 90 minutes to fully annotate and validate individual images [Cordts et al. 2016].

To overcome this challenge, a range of unsupervised segmentation methods have been developed that forgo pixel-level supervision [Ji et al. 2019; Cho et al. 2021; Van Gansbeke et al. 2022; Zhang and Maire 2020; Hamilton et al. 2022; Vobecky et al. 2022]. One particularly promising line of work has focused on a setting known as *unsupervised semantic segmentation with language-image pre-training* (USSLIP) [Shin et al. 2022b; Shin et al. 2023], which leverages a vision-

language foundation model [Bommasani et al. 2021] that has been pretrained on a large corpus of internet-sourced image-text pairs. USSLIP methods exhibit strong segmentation performance, category label flexibility and zero-shot transfer—the ability to perform well on a downstream task without access to images from the target distribution. However, while USSLIP methods enable *semantic segmentation*, no such method developed to date possesses the ability to differentiate between *instances within a semantic category*, a key functionality for many fine-grained applications.

In this paper, we consider a challenging task, **Z**ero-shot **U**nsupervised **T**ransfer **I**nstance **S**egmentation, *i.e.*, to segment the instances present in an image and infer its semantic classes without relying on manual supervision or access to a target dataset. To tackle such challenge, we start from the recent progress in USSLIP [Shin et al. 2023], retrieving images for given concept with a pretrained visual language model (*e.g.*, CLIP), then generating pseudo-masks for the collected images with an unsupervised saliency detector. To take one step further, we extend the prior USSLIP architectures with two critical abilities, namely, instance-level segmentation and generalisation to unseen categories. In specific, we couple a query-based Transformer [Vaswani et al. 2017] decoder, which generates instance mask proposals, with an image encoder, which is trained to output dense features (*i.e.*, patch tokens) aligned with text embeddings for a set of concepts from a *frozen* CLIP [Radford et al. 2021] text encoder. Notably, the design of the proposed approach allows to do inference for both semantic and instance segmentations with strong performance compared to prior state-of-the-art approaches.

In summary, our contributions are three-fold: (i) We introduce a challenging task, namely, zero-shot unsupervised transfer instance segmentation, which aims to segment object instances without human supervision or access to a target data distribution; (ii) We propose a simple yet effective framework, termed ZUTIS, that goes beyond prior USSLIP approaches, and enables to concurrently perform instance segmentation in addition to semantic segmentation; (iii) We show that

ZUTIS performs favourably against state-of-the-art methods on standard unsupervised segmentation benchmarks (*e.g.*, COCO [Lin et al. 2014], ImageNet-S [Gao et al. 2021]) by a large margin in both zero-shot transfer and unsupervised domain adaptation settings.

5.2 Related work

Our work relates to diverse themes in the literature including *zero-shot semantic/instance segmentation*, *unsupervised semantic segmentation* (with and without language-image pretraining), *unsupervised object segmentation*, *class-agnostic unsupervised instance segmentation*, *universal architectures*, and *open-vocabulary segmentation*.

Zero-shot semantic/instance segmentation with (image-)language pretraining. Zero-shot semantic/instance segmentation aims to generalise to unseen categories after training for seen categories with ground-truth annotations. The dominant approach exploits the relationships between category label embeddings produced by a language model (*e.g.*, word2vec [Mikolov et al. 2013] or GloVe [Pennington et al. 2014]) [Zhao et al. 2017; Bucher et al. 2019; Gu et al. 2020; Xian et al. 2019; Kato et al. 2019; Hu et al. 2020a; Li et al. 2020; Pastore et al. 2021; Zheng et al. 2021] to facilitate generalisation. More recently, there has been growing interest in leveraging the joint image-text embedding space produced by a pretrained vision-language model (*e.g.*, CLIP) to enable dense predictions [Rao et al. 2021; Li et al. 2022; Xu et al. 2021; Ding et al. 2022; Lüddecke and Ecker 2022]. In a similar vein, we build our approach on a pretrained vision-language model to enable generalisation to novel categories, but with two key differences. First, we do not assume access to a target data distribution, a setting termed *zero-shot transfer* in [Radford et al. 2021]. Second, we do not use any manual annotations during training. Note that the "annotation free" variant of MaskCLIP [Zhou et al. 2022] enables semantic segmentation in a similar regime in which neither access to

the target distribution nor ground-truth annotations are available. We compare our method to MaskCLIP on semantic segmentation tasks in Section 5.4.

Unsupervised semantic segmentation. A rich line of work has considered the problem of unsupervised semantic segmentation, creatively constructing learning signals from proxy tasks [Ji et al. 2019; Ouali et al. 2020b; Zhang and Maire 2020; Van Gansbeke et al. 2021; Cho et al. 2021; Hamilton et al. 2022; Vobecky et al. 2022]. One practical challenge associated with these approaches is their reliance on a matching stage to enable deployment (typically performed with Hungarian matching [Kuhn 1955] on pixel-level segmentation annotations) that establishes correspondences between segments and category names. By contrast, ZUTIS requires no access to pixel-level supervision during either training or inference. Furthermore, unlike the above, ZUTIS is capable of instance-level predictions as well as semantic segmentation. We note one exception: MaskDistill [Van Gansbeke et al. 2022] also reports on unsupervised instance segmentation in addition to semantic segmentation (also using Hungarian matching to assign categories to predictions). In Section 5.4, we compare ZUTIS with MaskDistill on unsupervised instance segmentation.

Unsupervised semantic segmentation with language-image pretraining (USSLIP). To achieve independence from pixel-level annotations during both training and inference, a recent line of work targeting unsupervised semantic segmentation proposes to leverage a vision-language model (*e.g.*, CLIP [Radford et al. 2021]) to assign names to categories [Shin et al. 2022b; Shin et al. 2023]. To do so, images are curated from an unlabelled image collection using the retrieval abilities of the vision-language model, and then segmented via co-segmentation [Shin et al. 2022b] or salient object detection [Shin et al. 2023]. However, while these methods avoid pixel-level annotations, they are either fragile (*i.e.*, co-segmentation used in [Shin et al. 2022b]) or rigid (*i.e.*, a new segmenter needs to be retrained from scratch for each new category [Shin et al. 2023]). Moreover, no USSLIP method to date supports instance segmentation. ZUTIS builds on this line of work, but ad-

dresses its limited functionality by enabling instance segmentation, and improves both robustness and flexibility.

Unsupervised object segmentation. Unsupervised object segmentation, also referred to as saliency detection, aims to train a detector to segment prominent object regions in images without human supervision. Traditionally, handcrafted methods have been proposed utilising low-level cues such as centre prior [Judd et al. 2009], contrast prior [Itti et al. 1998], and boundary prior [Wei et al. 2012]. A more recent line of research uses objectness properties emerging from self-supervised features extracted from modern vision architectures [Siméoni et al. 2021; Wang et al. 2022b; Shin et al. 2022a] (*i.e.* DINO [Caron et al. 2021]). In this work, we adopt SelfMask [Shin et al. 2022a] to generate object masks for images that are used as pseudo-masks for our training.

Class-agnostic unsupervised instance segmentation. Recently, FreeSOLO [Wang et al. 2022a] proposed a self-supervised framework for the class-agnostic instance segmentation task. For this, coarse object masks are first generated by using the object localisation property of self-supervised features (*e.g.*, DenseCL [Wang et al. 2021]), then a class-agnostic object detector is trained with the initial masks via a self-training scheme [Wang et al. 2022a]. Concurrent work, CutLER [Wang et al. 2023b], follows the similar framework, but with better initial masks produced by proposed MaskCut. Unlike the above, we focus on the conventional *class-aware* instance segmentation. The classification of each instance mask is made possible as ZUTIS is built on the recent progress in unsupervised semantic segmentation with language-image pretraining (*i.e.*, ReCo [Shin et al. 2022b]).

Universal architectures. Recently, universal architectures that deliver multiple object detection/segmentation tasks in a unified manner have gained considerable attention [Carion et al. 2020; Cheng et al. 2021b; Zhang et al. 2021b; Cheng et al. 2022]. Similarly, we propose an architecture that can tackle both semantic and instance segmentations with a single architecture with two key differences: (i) ZUTIS is flexible in terms of categories to segment as we use a text encoder

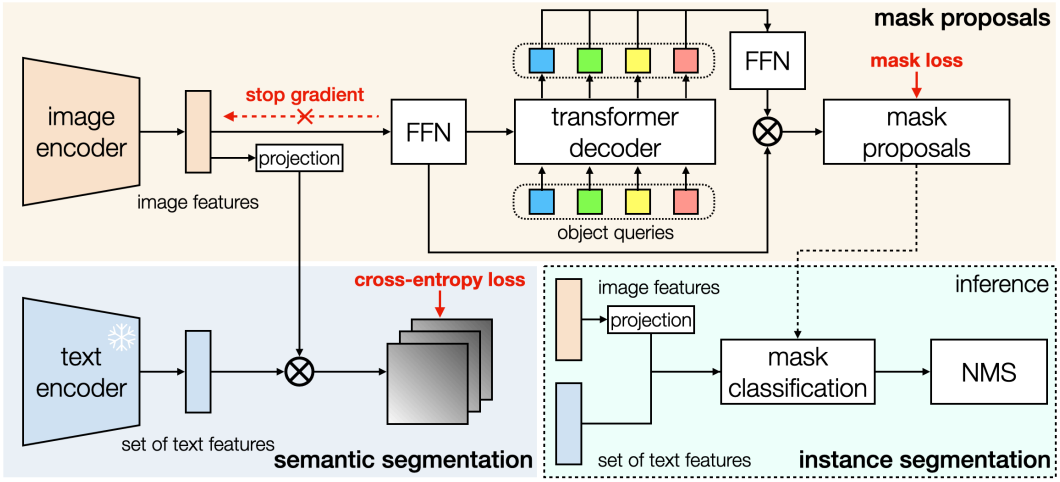


Figure 5.2: **Overview of ZUTIS**. Given an image encoder and a text encoder from a language-image model (*e.g.*, CLIP), ZUTIS learns to perform both semantic and instance segmentation. (Top) image features for an image are fed to a feed-forward network (FFN) followed by a transformer decoder to produce mask proposals, which are used to make predictions for instance segmentation at inference (bottom right). At the same time, the image features are projected into a text embedding space in which semantic predictions are made via a dot-product between the projected image features and frozen text features for a set of categories (bottom left). For simplicity, the pseudo-mask generation step is omitted. See the text for details.

as a classifier; (ii) unlike the above which need to train a model from scratch for a different task, ZUTIS requires only a single training for semantic and instance segmentations.

Open-vocabulary segmentation. Increasing the number of object categories to be segmented has been explored by utilising class-incremental few-shot learning [Hu et al. 2020b], captions [Ghiasi et al. 2021b], grounded text descriptions [Kamath et al. 2021], as well as annotation transfer [Hu et al. 2018; Huynh et al. 2021] and pairwise class balance regularisation [He et al. 2022]. Similarly, we seek to scale the number of classes to be segmented, but without human supervision.

5.3 Method

In this section, we start by introducing the considered problem scenario, namely, zero-shot unsupervised transfer instance segmentation in Section 5.3.1, and describe the core building blocks of our proposed approach in Section 5.3.2, followed by the architecture details for addressing unsupervised semantic and instance segmentation tasks with pretrained language-image models in Section 5.3.3.

5.3.1 Problem scenario

We consider the problem of zero-shot unsupervised transfer instance segmentation, which aims to jointly segment objects present in an image and predicts their semantic categories in both unsupervised and zero-shot transfer manner. The unsupervised property of the task prohibits any reliance on manual supervision for instance segmentation, while the zero-shot transfer property assumes that a segmenter has no access to a target data distribution (*e.g.*, a training split of an evaluation benchmark). Note that, such properties pose significant differences from the existing zero-shot instance segmentation, which leverages human supervision (*e.g.*, pixel-level annotations) in a training split (of an evaluation benchmark) for a certain group of classes (*e.g.*, seen categories) during training.

To tackle this challenge, we propose a simple yet effective framework in which we first predict class-agnostic object masks (mask proposal), then classify each mask (mask classification) based on the pixelwise classification obtained in a joint image-text space. Formally, we seek to train a segmenter Φ_{seg}, consisting of an image encoder $\Phi_{\mathcal{I}}^{\text{enc}}$, an image decoder $\Phi_{\mathcal{I}}^{\text{dec}}$, and a text encoder $\Phi_{\mathcal{T}}$. The segmenter ingests an image $x \in \mathbb{R}^{3\times H\times W}$, and a set of concepts/object categories ($\mathcal{C}$), and outputs a set of masks for semantic segmentation (SS) and instance segmentation

(IS):

$$\Phi_{\text{seg}}(x, \mathcal{C}) = \begin{cases} \Phi_{\mathcal{T}}(\mathcal{C})\mathbf{W}\Phi_{\mathcal{I}}^{\text{enc}}(x) \in \{0,1\}^{|\mathcal{C}|\times H\times W} \text{for SS,} \\ \Phi_{\mathcal{I}}^{\text{dec}} \circ \Phi_{\mathcal{I}}^{\text{enc}}(x) \in \{0,1\}^{n\times H\times W} \text{ for IS.} \end{cases} \tag{5.1}$$

where $\mathbf{W}$ is a matrix projecting image features into a text embedding space and n denotes a pre-defined number of object mask predictions. Note that, at this stage, the object masks from the image decoder are class-agnostic. To decide a class of the mask proposals, each mask is assigned a category via a dot-product between its average image embedding (from the image encoder) and text embeddings followed by a softmax (detailed in Section 5.3.3). It is worth noting that the design of our framework allows the model to tackle both instance and semantic segmentations concurrently—we show performance of our model on both tasks in Section 5.4.

In the following sections, we introduce the key components for our framework in an unsupervised and zero-shot transfer fashion: generating pseudo-labels for unlabelled images with existing pretrained foundation models, and an efficient transformer-based architecture for simultaneous semantic and instance segmentation.

5.3.2 Pseudo-label training

To train a segmenter without relying on manual labels, we adopt pseudo-label training as in [Shin et al. 2022b; Shin et al. 2023]. Here, we briefly describe our pseudo-mask generation process, composed of archive construction, unsupervised saliency detection, and copy-paste augmentation used to generate synthetic images containing multiple objects.

Archive construction. Given an image encoder $\phi_{\mathcal{I}}$ and a text encoder $\phi_{\mathcal{T}}$ from a pretrained vision-language model (*e.g.*, CLIP), we first build archives of images for a set of categories $\mathcal{C}$ by curating images for each concept from an unlabelled image dataset $\mathcal{U}$ (called an index dataset). Formally, we extract a set of normalised

image embeddings $\mathcal{F}_\mathcal{I}$ as follows:

$$\mathcal{F}_\mathcal{I} = \{\phi_\mathcal{I}(x_i) \in \mathbb{R}^d,\ i = 1, ..., N\} \tag{5.2}$$

where $x_i \in \mathbb{R}^{3\times H\times W}$ and N denotes the total number of images in $\mathcal{U}$. Similarly, we extract a set of normalised text embeddings $\phi_\mathcal{T}(c) \in \mathbb{R}^d$ for a name of each category $c \in \mathcal{C}$ from the text encoder. Then we select k images with highest similarities between image and text embeddings to form an archive for a category c:

$$\mathcal{U}_c = \{x_i \in \mathcal{U} \mid i \in \text{argtop}_k[\mathcal{F}_\mathcal{I}\phi_\mathcal{T}(c)]\} \tag{5.3}$$

where argtop_k returns indices of k largest values.

Unsupervised saliency detection. Given the image archives for the categories of interest, we generate category-agnostic saliency masks $\mathcal{S}_i \in \{0, 1\}^{H\times W}$ by feeding each image x_i into an unsupervised saliency detector (*e.g.*, SelfMask [Shin et al. 2022a]). We then assign the corresponding category name (from archive construction) and an instance id to the inferred saliency mask, which allows for training a segmenter for semantic and instance segmentations as described in Section 5.3.3.

Synthetic image generation with copy-paste augmentation. To train a segmenter which can segment multiple objects within an image, we follow [Shin et al. 2023] and use copy-paste augmentation [Ghiasi et al. 2021a] to synthesise an image with multiple objects. A pseudo-mask is created accordingly by copy-pasting the binary pseudo-masks of the images used for the synthetic image, with a unique instance id and a category label allocated to each mask.

5.3.3 Architecture

To tackle both semantic and instance segmentation tasks while preserving zero-shot ability of a pretrained vision-language model (VLM), we propose a simple framework termed, ZUTIS, which operates on features from image and text en-

coders of VLM (shown in Fig. 5.2).

Semantic segmentation. Given an image encoder $\psi_{\mathcal{I}}$ and a text encoder $\psi_{\mathcal{T}}$ from a pretrained VLM, we extract dense features $\psi_{\mathcal{I}}(x_i) \in \mathbb{R}^{e_v \times h \times w}$ (*e.g.*, patch tokens from a vision transformer) for an image x_i from the image encoder where e_v, h, and w denote the dimensionality of a visual embedding space, height and width of the features, respectively. The dense features are projected into a text embedding space by a projection matrix $\mathbf{W} \in \mathbb{R}^{e_t \times e_v}$, where e_t is a dimension of the text space. With text embeddings $\psi_{\mathcal{T}}(\mathcal{C}) \in \mathbb{R}^{|\mathcal{C}| \times e_t}$ from the text encoder for a set of categories, we compute logits via dot-product between the projected image features and text features which are followed by a softmax function:

$$P_i = \text{softmax}(\psi_{\mathcal{T}}(\mathcal{C})\widetilde{\psi}_{\mathcal{I}}(x_i),\ \texttt{dim=0}) \tag{5.4}$$

where $\widetilde{\psi}_{\mathcal{I}}(.)$ and P_i denote $\mathbf{W}\psi_{\mathcal{I}}(.)$ and the probability map, respectively. Then a cross-entropy loss $\mathcal{L}_{ce}$ is used to minimise differences between the prediction and the corresponding pseudo-mask generated in Section 5.3.2. To inherit the zero-shot ability of pretrained VLM, we only optimise the parameters of the image encoder, leaving the text encoder frozen. Note that this approach is related to MaskCLIP, but with a key difference: we update the parameters in image encoder, while MaskCLIP keeps the parameters fixed and instead uses value features from the last self-attention layer of the image encoder to produce a semantic prediction. We compare our method to MaskCLIP in Section 5.4.3.

Instance segmentation. Here, we first produce object mask proposals using a query-based transformer decoder. In detail, given dense image features $\psi_{\mathcal{I}}(x_i)$ before projection to the textual space, we pass the features to a feed-forward network (FFN) with a hidden layer (*e.g.*, an MLP with three layers) whose output features are used as values $V \in \mathbb{R}^{d \times h \times w}$ for the transformer decoder. Given n_q object queries $Q \in \mathbb{R}^{n_q \times d}$ and V, the decoder outputs query vectors that are fed into another FFN before producing mask proposals $\mathcal{M} \in \mathbb{R}^{n_q \times h \times w}$ via a dot-

product between the resulting Q and V. Then, we update the model with a bipartite matching loss [Carion et al. 2020; Cheng et al. 2021b; Cheng et al. 2022] $\mathcal{L}_{mask}$ between the proposals and the pseudo-masks for the image. We find that it is essential to **stop gradients** from the transformer decoder flowing to the image encoder, otherwise the model fails to converge (see Section 5.4.2). For $\mathcal{L}_{mask}$, we use a mixture of dice coefficient [Milletari et al. 2016] and binary cross-entropy losses $\mathcal{L}_{mask} = \mathcal{L}_{dice} + \mathcal{L}_{bce}$ with equal weights following [Cheng et al. 2022].

During inference, we assign each mask proposal $m_l \in \mathcal{M}$ a category whose text embedding shares the highest similarity with an average image embedding of the mask. For this, we first binarise m_l with a threshold t and compute the average image embedding $\overline{\psi}_{\mathcal{I}}(x_i, m_l; t)$:

$$\overline{\psi}_{\mathcal{I}}(x_i, m_l; t) = \text{mean}(\widetilde{\psi}_{\mathcal{I}}(x_i)[m_l > t]) \in \mathbb{R}^{e_t}. \tag{5.5}$$

Then, we assign the mask that category with highest similarity to the average image embedding:

$$\underset{c \in \mathcal{C}}{\text{argmax}} \Big[\psi_{\mathcal{T}}(\mathcal{C}) \overline{\psi}_{\mathcal{I}}(x_i, m_l; t) \Big]. \tag{5.6}$$

Note that both text and average image embeddings are L2-normalised before dot-product. In addition, we compute a confidence score $s_l \in [0, 1]$ for each mask proposal, defined as the average value of the mask region multiplied by the maximum class probability for the mask (similarly to [Cheng et al. 2022]). Lastly, to reduce false positives occurring from redundant predictions for a single object, we apply non-maximum suppression (NMS) to the proposals before outputting final instance predictions. We show the effect of NMS in Section 5.4.2.

Discussion. The key differences of ZUTIS from prior work for unsupervised semantic segmentation with language-image pretraining are two-fold: (i) rather than using a fixed n-way classifier, we use a pretrained, frozen text encoder as a classifier, and optimise an image encoder to output dense features aligned with the textual features from the text encoder, a design choice that allows the model

to be open-vocabulary; (ii) we enable instance segmentation by training a query-based transformer decoder by bootstrapping the results from saliency detection via copy-paste augmentation.

5.4 Experiments

In this section, we first describe the details of our experiments including datasets, network architecture, training and inference settings, and evaluation metrics. Next, we ablate components of our method such as the use of stop-gradient and non-maximum suppression, and report the performance of the model on both semantic and instance segmentation.

5.4.1 Implementation details

Datasets. We evaluate our model on COCO2017 [Lin et al. 2014] `val` split, PASCAL VOC2012 [Everingham et al. 2010], CoCA [Zhang et al. 2020], and ImageNet-S [Gao et al. 2021] `test` split for semantic segmentation and COCO-20K [Vo et al. 2020] for instance segmentation following [Van Gansbeke et al. 2022]. To demonstrate our model's zero-shot ability to **new concepts**, we additionally consider the CUB-200-2011 [Wah et al. 2011] `test` split. In detail, VOC2012 `trainval` split has 2,913 images with 21 categories including a background. COCO2017 `val` and CoCA are composed of 5,000 and 1,295 images with 80 object categories and a background class. ImageNet-S `test` consists of 27,423 images with 919 object classes which are a subset of ImageNet1K [Russakovsky et al. 2015] classes. COCO-20K comprises 19,817 images from the COCO2014 `train` split with the same 80 object classes as COCO2017. CUB-200-2011 `test` is composed of 5,794 images with 200 fine-grained bird breeds.

Note that, in this paper, we primarily consider the zero-shot transfer setting, in which the model has no access to training data sharing a data distribution with a downstream benchmark. Thus, throughout our experiments, we use images

retrieved from ImageNet1K (1.2M images) and PASS [Asano et al. 2021] (1.4M images) by the ViT-L/14@336px CLIP model to form an index image dataset except an experiment in the unsupervised domain adaptation setting for the ImageNet-S benchmark where we only retrieve ImageNet1K images. For prompt engineering, we average text embeddings from 85 templates to obtain a textual feature for a category following [Zhou et al. 2022; Shin et al. 2022b; Shin et al. 2023]. In all cases, we fix the number of images for an archive as 500 (*i.e.* 500 images for a category) as in [Shin et al. 2023].

Architecture. We use transformer-based CLIP models for the image encoder (*e.g.*, ViT-B/16) and text encoder. We use 6 transformer layers for the transformer decoder and three-layer MLP for the FFN. We feed patch tokens from the last layer of the image encoder to the decoder after bilinearly upsampling them by a factor of 2 to enable predictions at a higher resolution.

Training. We compute the final loss $\mathcal{L}$ for the model as $\mathcal{L} = \mathcal{L}_{ce} + \lambda_{mask}\mathcal{L}_{mask}$ with λ_{mask} set to 1.0. We optimise our model with the AdamW optimiser [Loshchilov and Hutter 2019], with an initial learning rate of 5e-5 and a weight decay of 0.05. For the image encoder, we use a smaller learning rate of 5e-6. We train for 20K iterations with the Poly learning rate scheduler [Chen et al. 2018b], except when training for 919 categories of ImageNet-S where the model is updated for 80K iterations. We use standard data augmentations such as random resizing, cropping and colour jittering. Following [Shin et al. 2023], we adopt copy-paste augmentation [Ghiasi et al. 2021a] and set the maximum number of images used for copy-pasting to 10. To further encourage the model to differentiate objects of the same category, we select images used for copy-paste from a randomly selected archive 50% of the time. As in [Carion et al. 2020; Cheng et al. 2022], we compute a mask loss for predictions by each transformer decoder layer.

Inference. We perform inference on images at their original resolution, except for the large-scale ImageNet-S benchmark where we resize images with a longer side larger than 1024 while preserving its aspect ratio. For such cases, the orig-

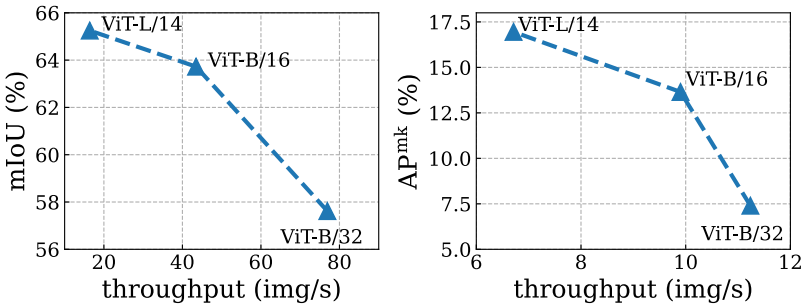


Figure 5.3: With more computation measured in throughput, ZUTIS can produce better performance in both semantic segmentation (left) and instance segmentation (right).

inal resolution is restored with a bilinear upsampler following [Shin et al. 2023]. In addition, we apply NMS for instance segmentation predictions as mentioned in Section 5.3.3.

Evaluation metrics. To measure our model's performance, we use the standard metrics such as mean intersection-over-union (mIoU) for semantic segmentation and COCO-style mask average precision (AP^{mk}) for instance segmentation.

5.4.2 Ablation study

Here, we study the influence of the components in our method, including the choice of encoder architecture, stop-gradient and NMS. For experiments in the ablation study, we report the results on the VOC2012 `trainval` split.

Effect of encoder architecture. In Fig. 5.3, we show the performance of our model with different transformer-based CLIP image encoders such as ViT-B/32, ViT-B/16, and ViT-L/14 [Dosovitskiy et al. 2021].[2] We observe that at the cost of computation measured in throughput, heavier models consistently outperform lighter models in both mIoU (left) and AP^{mk} (right). While ViT-L/14 performs best, we report results with either ViT-B/32 or ViT-B/16 in the following experiments to limit differences in performance due to model size (ResNet50 [He et al.

[2]While there are also ResNet-based CLIP encoders, we found them not suitable for instance segmentation as they directly output features in the joint image-text space via an attention pooling [Radford et al. 2021].

stop-grad	NMS	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
✗	✗	0.3	0.4	0.3
✓	✗	4.4	8.7	4.2
✓	✓	**13.7**	**30.9**	**11.1**

Table 5.1: Stopping gradients from the transformer decoder to the image encoder plays a crucial role in our framework while non-maximum suppression (NMS) produces a substantial boost in performance. The performance is measured in COCO-style AP metrics for instance segmentation.

	class-agnostic			*class-aware*		
stop-grad	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
✗	8.4	19.9	6.4	1.0	1.9	0.9
✓	9.9	24.1	7.4	13.7	30.9	11.1

Table 5.2: Applying a stop-grad operation between the image encoder and decoder allows the encoder features to keep semantic representations.

2016] is typically used by previous unsupervised methods). Note that ViT-B/32 and ViT-B/16 are the lightest CLIP models compatible with our framework.

Effect of stop-gradient and non-maximum suppression. As described in Section 5.3.3, we prevent gradients from backpropagating to the encoder parameters when optimising the transformer decoder to generate mask proposals. We observe in Table 5.1 that this is crucial, otherwise the optimisation does not converge to a reasonable solution. For our model trained with stop-gradient, applying NMS to its mask proposals brings a noticeable gain in performance from 4.4 to 13.7 AP^{mk}. This is because the model tends to predict redundant mask proposals for a single object, increasing false positives during evaluation. We therefore employ both stop-gradient and NMS throughout the experiments.

Analysis on a stop-gradient operation. To further investigate why applying a stop-grad operation is essential in our framework, we hypothesise that if stop-grad is not applied, gradients from the mask loss for instance segmentation could potentially dominate the visual feature learning, thus harming the visual-language alignment in pretrained VLM. To verify this, we evaluate the models trained w/ and w/o the stop-grad w.r.t. class-agnostic and -aware AP^{mk}. As shown in Ta-

ble 5.2, while the model trained w/o stop-grad performs poorly on class-aware AP^{mk}, it performs reasonably on class-agnostic AP^{mk}, indicating that the resulting features are capable of segmenting objects but not suitable for classification. Thus, it is crucial to separate features used for semantic and instance predictions via a stop-grad for enabling class-aware instance segmentation.

5.4.3 Main results

Here, we compare ZUTIS to existing unsupervised instance segmentation and semantic segmentation approaches. While we mainly focus on the zero-shot transfer setting in which the model has no access to training data for the target downstream task, we also report results for semantic segmentation in the unsupervised domain adaptation setting to draw comparison with existing approaches, where the target data distribution is exposed to the model.

Unsupervised instance segmentation. In Table 5.3, we evaluate our model on unsupervised instance segmentation on the COCO-20K dataset. As a baseline for our method, we evaluate MaskCLIP for instance segmentation by treating its semantic segmentation masks for an image as mask proposals. When comparing to the state-of-the-art approach [Van Gansbeke et al. 2022], our model shows comparable (with ViT-B/32) or better performance (with ViT-B/16) by 3.3 AP^{mk}_{50}. For qualitative visualisations of our model's predictions, see Fig. 5.1.

Unsupervised semantic segmentation. For semantic segmentation, we primarily compare to unsupervised approaches that leverage language-image pre-training such as NamedMask, ReCo, and MaskCLIP. As ReCo and MaskCLIP do not predict a "background" category, we re-implement their methods to predict background class since it appears in all the benchmarks considered in our experiments. Specifically, for ReCo, we follow [Shin et al. 2023] and treat the pixels as background if their maximum class probability is lower than a threshold (=0.9). For MaskCLIP, we simply provide a text embedding for "background" along with other object category embeddings which we find more effective than thresholding.

model	backbone	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
unsupervised methods w/o language-image pretraining				
DINO [Caron et al. 2021]	ViT-S/16	0.7	2.0	0.4
LOST [Siméoni et al. 2021]	ViT-S/16	1.2	3.3	0.6
MaskDistill [Van Gansbeke et al. 2022]	ViT-S/16	1.7	4.1	1.4
MaskDistill [Van Gansbeke et al. 2022]†	RN50-C4	3.5	7.7	2.9
unsupervised method w/ language-image pretraining				
MaskCLIP [Zhou et al. 2022]	ViT-B/32	0.3	0.8	0.2
ZUTIS (Ours)	ViT-B/32	3.4	8.0	2.6
MaskCLIP [Zhou et al. 2022]	ViT-B/16	1.3	3.4	0.8
ZUTIS (Ours)	ViT-B/16	**5.7**	**11.0**	**5.4**

Table 5.3: Comparison to previous unsupervised instance segmentation methods on COCO-20K. †Mask R-CNN [He et al. 2017] trained with pseudo-masks from MaskDistill. The numbers for the methods without language-image pretraining are quoted from [Van Gansbeke et al. 2022].

model	arch.	COCO	CoCA
initialised with different encoder features			
ReCo† [Shin et al. 2022b]	DeiT-S/16 & RN50x16	23.8	28.8
NamedMask‡ [Shin et al. 2023]	RN50 & DLv3+	28.4	27.3
initialised with CLIP encoder features			
MaskCLIP [Zhou et al. 2022]	ViT-B/16	20.6	20.2
ZUTIS (Ours)	ViT-B/16	**32.8**	**32.7**

Table 5.4: Comparison to previous unsupervised semantic segmentation methods leveraging image-language pretraining on COCO and CoCA in terms of mIoU (%). †Initialised with supervised Stylised-ImageNet pretraining [Naseer et al. 2021]. ‡Initialised with DINO [Caron et al. 2021].

In Table 5.4, we evaluate our model in the zero-shot transfer setting on the COCO `val` and CoCA benchmarks and compare to unsupervised methods. Note that ReCo and NamedMask have different settings from ours in terms of initialisation and architecture for a backbone (*i.e.*, ReCo initialises its backbone with supervised Stylised-ImageNet training and NamedMask with DINO), thus a direct comparison is not possible. Relative to MaskCLIP (which is comparable), our model shows improvements of 12.2 and 12.5 mIoU on COCO and CoCA, respectively.

In Table 5.5, we evaluate our method in the unsupervised domain adaptation

model	arch.	# params (M)	throughput (img/s)	mIoU	S	MS	ML	L
unsupervised methods w/o language-image pretraining								
PASS_p [Gao et al. 2021]	RN50	25.6	-	6.6	1.3	4.6	7.1	8.4
PASS_s [Gao et al. 2021]	RN50	25.6	-	11.0	2.4	8.3	11.9	13.4
unsupervised methods w/ language-image pretraining								
ReCo† [Shin et al. 2022b]	DeiT-S/16 & RN50x16	170.4	32.3	10.3	6.0	11.6	10.2	6.7
NamedMask‡ [Shin et al. 2023]	RN50 & DLv3+	26.6	**125.0**	22.9	5.1	19.4	24.4	19.8
ZUTIS (Ours)	ViT-B/32	87.8	76.9	27.5	5.6	22.3	28.9	26.5
ZUTIS (Ours)	ViT-B/16	86.2	43.5	**37.4**	**10.7**	**32.1**	**40.2**	**33.4**

Table 5.5: Comparison to existing unsupervised methods on the ImageNet-S benchmark with 919 object categories in the unsupervised domain adaptation setting. We also show mIoU in diverse object sizes from small (S), medium-small (MS), medium-large (ML), and large (L). †Encoder initialised with supervised Stylised-ImageNet pretraining. ‡Encoder initialised with unsupervised pretraining (*i.e.*, DINO).

category-specific label	CUB-200-2011
✗	72.5
✓	72.6

Table 5.6: High-level to low-level zero-shot transfer on the CUB-200-2011 benchmark. When given a fine-grained bird breed, ZUTIS can segment the corresponding bird regions as good as when it is given a high-level category "bird."

setting, where the model is trained with images retrieved from the ImageNet1K `train` split, and evaluated on ImageNet-S. Compared to the state-of-the-art unsupervised method (*i.e.* NamedMask), our approach achieves a gain of 4.6 mIoU with ViT-B/32 and 14.5 mIoU with ViT-B/16 at the expense of lower throughput.

Generalisation to new categories. Since we optimise our image encoder to produce visual embeddings aligned to the text embedding from the frozen text encoder, we expect the resulting model to be capable of segmenting objects of novel concepts which are unseen during training. To verify this, we consider two scenarios: (i) a high-level to low-level category transfer, *i.e.*, the model is evaluated on categories that it did not encounter during training but only their superset category; (ii) transfer to unseen categories semantically far from those it has seen during training.

For the first scenario, we evaluate our model, trained for 80 categories in COCO

including 'bird', on the `test` split of CUB-200-2011 benchmark which has 200 fine-grained bird categories. Here, given a high-level category (*i.e.*, "bird") or a low-level category for an image (*i.e.*, image-specific fine-grained bird categories), we encode the category with the text encoder and proceed with segmentation as usual. Then we compare the segmentation result with the groundtruth mask. It is worth mentioning that the performance is measured in IoU rather than mIoU, as we do not expect the model to distinguish between the fine-grained categories. This is because the image archive for "bird" is likely to contain images of different bird breeds, which encourages the model to learn the invariance between birds. However, we expect the model to also identify the "bird" regions given a specific bird breed as a target. As shown in Table 5.6, when given low-level categories (bird breeds) as input, our model can perform equally well as when given a high-level category ("bird"). Note that the semantically closest category to the fine-grained categories among the 80 classes in COCO is "bird" and that only 16 out of 200 fine-grained bird categories contain "bird" as a part of its name (*e.g.*, "Anna Hummingbird"). This implies that the model is equipped with knowledge about fine-grained bird species.

For the second scenario, we split 65 seen and 15 unseen classes in the COCO dataset and evaluate our model on the unseen classes following prior work on zero-shot instance segmentation [Zheng et al. 2021]. For this, we train our model with image archives constructed only for the seen categories. As shown in Table 5.7, when compared to a baseline unsupervised method, our model performs favourably by a notable margin (*i.e.*, +5.2 AP_{50}^{mk}). It indicates good generalisability of our model to the unseen categories which are semantically remote from the concepts seen during training.

5.4.4 Limitations

Although we show strong performance on unsupervised semantic segmentation and instance segmentation leveraging only image-language pretraining, we also

model	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
MaskCLIP [Zhou et al. 2022]	0.7	2.0	0.4
ZUTIS (Ours)	3.3 (+2.6)	7.2 (+5.2)	2.8 (+2.4)

Table 5.7: Zero-shot unsupervised instance segmentation for 15 unseen categories on COCO-20K.

note two limitations to our work: (i) While the use of a vision-language model such as CLIP [Radford et al. 2021] significantly simplifies deployment relative to unsupervised approaches that employ Hungarian matching, it also represents the source of potential error. For instance, our method will be unable to segment categories that are not present in CLIP's pretraining data (extremely rare concepts, for example). (ii) Our pipeline for sourcing pseudo-masks with a vision-language model (VLM) and a saliency detector has drawbacks. Images retrieved for a given concept by VLM can contain objects of a distracting class which the detector can highlight together with the desired category object. For example, both a skateboard and a person riding it can be segmented by the detector when we retrieve the image for "skateboard" and generate a pseudo-mask for it.

5.5 Broader impact

The goal of this work is to propose a practical framework for instance and semantic segmentation. As such, we hope that our work facilitates many useful applications of segmentation (medical image analysis, fault detection in manufacturing, security monitoring etc.). However, automatic segmentation represents a dual-use technology and is therefore subject to misuse (unlawful surveillance, for example). We also note that we build ZUTIS on top of foundation models like CLIP [Radford et al. 2021]. These models are known to reflect biases present in large, minimally curated internet corpora and thus our model is likely to inherit these biases also. Consequently, any practical deployment of ZUTIS will require assessment (and potentially also mitigation) of the risks posed by such biases.

5.6 Conclusion

In this work, we introduced ZUTIS, the first framework for joint instance segmentation and semantic segmentation in a zero-shot transfer setting that requires no pixel-level or instance-level annotation. We employ a query-based transformer architecture for instance segmentation and train it on pseudo-labels generated from applying an unsupervised saliency detector to images retrieved by CLIP. Through careful experiments, we demonstrated the effectiveness of ZUTIS across both instance segmentation and semantic segmentation tasks. In future work, we intend to explore the application of ZUTIS to other modalities such as video.

Acknowledgements

This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk). GS is supported by AI Factory, Inc. in Korea. GS would like to thank Guanqi Zhan for proof-reading and Zheng Fang for the enormous support. SA would like to acknowledge the support of Z. Novak and N. Novak in enabling his contribution.

Appendices

Appendices for this chapter can be found in the online version of the paper.[3]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[3] https://www.robots.ox.ac.uk/~vgg/research/zutis

Part III

Self-supervised Learning for Class-agnostic Image Segmentation

Chapter 6

Unsupervised Salient Object Detection with Spectral Cluster Voting

The paper has been accepted for publication at the Learning with Limited Labelled Data for Image and Video Understanding Workshop, Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

Unsupervised Salient Object Detection with Spectral Cluster Voting

Gyungin Shin[1] Samuel Albanie[2] Weidi Xie[1,3]

[1]Visual Geometry Group, University of Oxford

[2]Department of Engineering, University of Cambridge

[3]Shanghai Jiao Tong University

https://www.robots.ox.ac.uk/~vgg/research/selfmask

Abstract

In this paper, we tackle the challenging task of unsupervised salient object detection (SOD) by leveraging spectral clustering on self-supervised features. We make the following contributions: (i) We revisit spectral clustering and demonstrate its potential to group the pixels of salient objects across various self-supervised features, *e.g.*, MoCov2, SwAV, and DINO; (ii) Given mask proposals from multiple applications of spectral clustering on image features computed from different self-supervised models, we propose a simple but effective winner-takes-all voting mechanism for selecting the salient masks, leveraging object priors based on framing and distinctiveness; (iii) Using the selected object segmentation as pseudo groundtruth masks, we train a salient object detector, termed SelfMask, which outperforms prior approaches on three unsupervised SOD benchmarks. Code is publicly available at https://github.com/NoelShin/selfmask.

6.1 Introduction

Salient object detection (SOD)[1], which aims to group pixels that attract human visual attention, has been extensively studied in the field of computer vision due to its wide range of applications such as photo cropping [Suh et al. 2003; Marchesotti et al. 2009], re-targeting in images [Ding et al. 2011; Sun and Ling 2011] and video [Rubinstein et al. 2008].

In the literature, early work tackled this problem by utilising low-level features (*e.g.*, colour [Cheng et al. 2015]) together with priors on salient regions in an object such as contrast priors [Itti et al. 1998], boundary priors [Wei et al. 2012] and centre priors [Judd et al. 2009]. Recent SOD models have approached this task from the perspective of representation learning, typically training deep neural networks (DNNs) on a large-scale dataset with manual annotations. However, the scalability of such supervised learning approach is limited because it is a costly process to collect ground-truth mask annotations.

To overcome the necessity of large-scale human annotation, many unsupervised methods for saliency detection/object segmentation have recently been proposed [Bielski and Favaro 2019; Chen et al. 2019; Voynov and Babenko 2020; Voynov et al. 2021; Melas-Kyriazi et al. 2022c; Zhang et al. 2017; Zhang et al. 2018; Nguyen et al. 2019]. Despite these efforts, the gap between unsupervised and fully supervised SOD methods remains significant.

Interestingly, however, it has been noted that recent self-supervised models such as DINO [Caron et al. 2021] exhibit significant potential for object segmentation despite the fact that their training objective does not explicitly encourage pixel grouping. The focus of this work is to leverage this observation to propose a *simple yet effective mechanism for extracting object regions from self-supervised features* that can be employed for the task of unsupervised salient object detection.

[1] In contrast to *object detection* (which aims to localise and recognise objects with bounding boxes), *salient object detection* aims to segment foreground objects by predicting pixel-wise masks for them.



Figure 6.1: In this work, we propose SelfMask, a framework for salient object detection that employs no human annotation. The figure depicts example segmentations produced by our model on DUT-OMRON [Yang et al. 2013], DUTS-TE [Wang et al. 2017] and ECSSD [Shi et al. 2016], with blue and orange regions denoting the intersection and the difference between the predicted mask and ground-truth label, respectively. Despite no supervision, SelfMask reliably segments a close approximation to the full spatial extent of salient regions. Best viewed in colour.

To this end, we explore the use of spectral clustering [Shi and Malik 2000], a classical graph-theoretic clustering algorithm, and find that it can generate useful segmentation candidates across a range of self-supervised features (*i.e.*, DINO [Caron et al. 2021], MoCov2 [He et al. 2020], and SwAV [Caron et al. 2020]). Motivated by this finding, we propose a simple *winner-takes-all* voting strategy to select the salient object masks among a collection of clusters produced by repeated applications of spectral clustering to self-supervised features.[2] In particular, we base our voting strategy on two priors: The first is a *framing prior* that a salient object should not occupy the full image height or width; the second is a *distinctiveness prior* that assumes that salient regions are sufficiently distinctive that they will appear as clusters among an appropriately constructed collection of redundant re-clusterings of the data. We then show that the selected salient masks can be employed as *pseudo-labels* to train a saliency estimation network that achieves state-of-the-art results on a variety of benchmarks (see Fig. 6.1 for

[2]In this work, we use the terms *cluster* and *mask* interchangeably. Specifically, by mask, we mean a (one-hot) mask which encodes the spatial extent of a cluster.

example visualisations).

In summary, we make the following contributions: (i) We revisit spectral clustering and highlight that, when applied to self-supervised features, it brings clear benefits over k-means clustering as a proposal mechanism on three popular salient object detection (SOD) datasets; (ii) We propose an effective voting strategy to select the most salient object mask in an image among multiple segmentations generated from different self-supervised features by leveraging saliency priors; and (iii) Using the salient masks as pseudo ground truth masks (pseudo-masks), we train an object segmentation model, SelfMask, that outperforms the previous unsupervised saliency detection approaches on three SOD benchmarks.

6.2 Related work

Our work relates to two themes in the literature: *self-supervised representation learning* and *unsupervised saliency detection*. Each of these is discussed next.

6.2.1 Self-supervised representation learning

There has been a great deal of interest in self-supervised approaches to learning visual representations that obviate the requirement for labels. These include techniques for solving proxy tasks, such as predicting patch locations [Doersch et al. 2015], patch discrimination [Dosovitskiy et al. 2015], grouping through co-occurrence [Isola et al. 2015], colourisation [Larsson et al. 2017], jigsaw puzzles [Noroozi and Favaro 2016], common fate principles [Mahendran et al. 2018], clustering [Caron et al. 2018; Asano et al. 2020], and instance discrimination [Wu et al. 2018; He et al. 2020]. Our work is inspired by recent work exploring the use of self-supervised transformers [Chen et al. 2021; Caron et al. 2021], and by the analysis provided by [Caron et al. 2021; Naseer et al. 2021] who noted that the self-attention of Vision Transformers (ViT) [Dosovitskiy et al. 2015] are capable of highlighting spatially coherent object regions in their input. While prior

work [Zhan et al. 2018] has demonstrated that self-supervised pretraining can be effective for semantic segmentation when coupled with end-to-end supervised metric learning on the target dataset, we instead seek a simple way to exploit self-supervised features for object segmentation without annotation.

6.2.2 Unsupervised saliency detection

Supervised object segmentation requires pixel-wise annotations which are time-consuming to acquire. Seeking to avoid this cost, many attempts have been made to solve the task in an unsupervised fashion. Prior to the dominance of deep neural networks, a broad range of handcrafted methods were proposed [Liu et al. 2007; Achanta et al. 2009; Klein and Frintrop 2011; Goferman et al. 2012; Yan et al. 2013; Zhu et al. 2014; Cheng et al. 2015] based on one or more priors relating to foreground regions within an image such as the contrast prior [Itti et al. 1998], centre prior [Judd et al. 2009], and boundary prior [Wei et al. 2012]. However, these handcrafted approaches suffer from poor performance relative to recent DNN-based models, described next.

Generative models. A common approach for DNN-based unsupervised object segmentation is to utilise generative adversarial networks (GANs) [Goodfellow et al. 2014]. Specifically, given an image, a generator is adversarially trained to produce an object mask which will be used to composite a realistic image by copying the corresponding object region in the image into a background which is either synthesised [Bielski and Favaro 2019] or taken from a different image [Arandjelović and Zisserman 2019]. An alternative family of approaches aims to discover a direction in the latent space of a pre-trained GAN that can be used to segment foreground and background regions [Voynov and Babenko 2020; Voynov et al. 2021; Melas-Kyriazi et al. 2022c]. Then, a saliency detector is trained on a synthetic data set composed of pairs of images together with their foreground masks generated via the discovered latent space structure. In contrast, we seek to exploit representations learned via self-supervision by discriminative, rather than generative,

models.

Noisy supervision with pseudo-labels. More closely related to our work, the use of weak pseudo saliency masks for training a DNN has been proposed. SBF [Zhang et al. 2017], the first attempt to train saliency detector without human annotation, proposed to train a model with superpixel-level pseudo-masks generated by fusing weak saliency maps from multiple unsupervised methods (*i.e.*, [Zhang et al. 2015; Zhang and Sclaroff 2016; Shi et al. 2016]). Similarly, USD [Zhang et al. 2018] aims to learn from diverse noisy pseudo-labels obtained via distinct unsupervised handcrafted methods in such a way that a saliency detector trained with the pseudo-masks can predict a saliency map free from label noise. DeepUSPS [Nguyen et al. 2019] proposed to refine the pseudo-masks for images produced by handcrafted saliency methods, by training segmentation networks via a self-supervised iterative refinement process. The refined pseudo-masks are then combined from different handcrafted methods to train a final segmentation network. In contrast to the methods above, we use neither handcrafted saliency methods nor an iterative refinement strategy, resulting in a simpler learning framework.

Object segmentation properties of self-supervised vision transformers. Another line of related work has sought to investigate the observation that self-supervised ViTs [Caron et al. 2021] exhibit object segmentation potential. LOST [Siméoni et al. 2021] propose to pick a seed patch from such a ViT that is likely to contain part of a foreground object, and then expand the seed patch to different patches sharing a high similarity with the seed patch. Concurrent work, TokenCut [Wang et al. 2022b], proposes to use Normalised Cuts [Shi and Malik 2000] to segment the salient object among the final layer self-attention key features of ViT. SelfMask differs from the TokenCut approach to salient object detection in two key ways: (1) While we similarly employ spectral clustering as part of our pipeline, we demonstrate the significant additional value of integrating cues from diverse re-clusterings via voting to bootstrap a pseudo-labelling process; (2) Rather

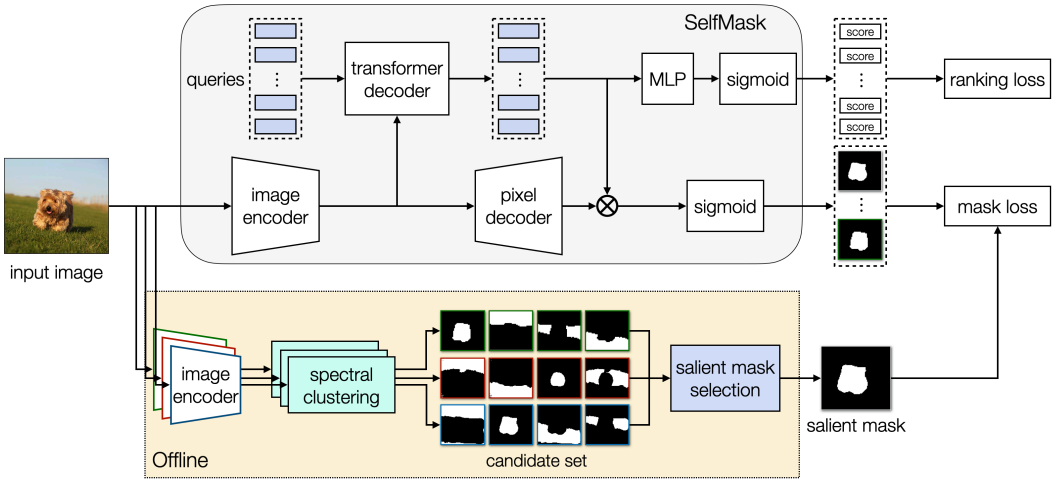


Figure 6.2: Overview of our approach. Given different self-supervised encoders, we first generate a set of pseudo-mask candidates per image using spectral clustering before the training step. In the figure we show 12 masks from clusterings from three different encoder features with k=4. We select the most salient mask among them via the proposed voting strategy and use it as a pseudo-mask for the image. Then we train our model to predict n_q queries (*i.e.*, predictions), all of which are encouraged to be similar to the salient mask. To make the model aware of the objectness of each prediction, we use the ranking loss which encourages the objectness score of a prediction closer to the salient mask to be higher. At inference time, we select the prediction with the highest objectness score. Please see the text for details.

than focusing solely on transformers, we leverage saliency cues from self-supervised convolutional neural networks (CNNs) as well as ViT architectures, and show the benefits of doing so. We compare our approach with theirs in Section 6.4.

6.3 Method

In this section, we begin by formalising the problem scenario (Section 6.3.1) and briefly summarise spectral clustering (Section 6.3.2). Then, we introduce our approach to address unsupervised salient object detection by selecting pseudo ground truth masks via spectral clustering, and train a saliency prediction network, called SelfMask (Section 6.3.3).

6.3.1 Problem formulation

Here, we consider the task of unsupervised salient object detection (SOD), with the goal of training a segmenter (Φ_{seg}) that seeks to partition the image into two disjoint groups, namely foreground and background. That is,

$$\Phi_{\text{seg}}(\mathbf{I}; \Theta) = \mathbf{M}_{\text{seg}} \in \{0, 1\}^{H \times W} \tag{6.1}$$

where $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ refers to an input image, and $\boldsymbol{\Theta}$ represents learnable parameters. $\mathbf{M}_{\text{seg}}$ denotes a binary segmentation mask, with 1s denoting a foreground region and 0s denoting background.

Traditionally this has been treated as a clustering problem where the key challenge lies in designing effective features for accurately describing salient regions. In this work, we instead look for a simple yet effective solution by leveraging self-supervised visual representations.

6.3.2 Segmentation with spectral clustering

Conceptually, segmentation is obtained via spectral clustering [Shi and Malik 2000] with pixel-wise image features by projecting the features onto a representation space such that image partitions can be decided by directly comparing similarities between their corresponding features.

Concretely, we first extract dense features from the image, *i.e.*, $\mathbf{F} \in \mathbb{R}^{h \times w \times D}$ with a pre-trained convolutional- or transformer-based encoder. Then, each feature vector $\mathbf{f}_i \in \mathbb{R}^D$ in the dense feature map $\mathbf{F}$ can be seen as a vertex in an undirected graph with vertex set $\mathbf{V} = \{\mathbf{f}_1, \ldots, \mathbf{f}_N\} \in \mathbb{R}^{N \times D}$, where $N = h \cdot w$. Each edge between two vertices $\mathbf{f}_i$ and $\mathbf{f}_j$ is associated with a non-negative weight $w_{ij} \geq 0$ defined by feature similarity. In particular, the weighted *adjacency matrix* $\mathbf{W}$ of

the graph is computed as

$$\mathbf{W} = (w_{ij}) = \frac{\mathbf{V}\mathbf{V}^T}{\|\mathbf{V}\|\|\mathbf{V}\|} \in \mathbb{R}^{N \times N} \tag{6.2}$$

where the *degree* of a vertex $\mathbf{f}_i \in \mathbf{V}$ is defined as $d_i = \sum_{j=1}^{N} w_{ij}$, and the *degree matrix* $\mathbf{D}$ is defined with the degrees $d_1, \ldots, d_N$ on the diagonal. Given the adjacency matrix $\mathbf{W}$ and the degree matrix $\mathbf{D}$, the (un-normalised) graph Laplacian $\mathbf{L}$ is defined as:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{6.3}$$

Given $\mathbf{L}$, we can solve the generalised eigenproblem:

$$\mathbf{L}\mathbf{u} = \lambda \mathbf{D}\mathbf{u} \tag{6.4}$$

where $\mathbf{u} \in \mathbb{R}^N$ and λ represent an eigenvector and its eigenvalue. We take the k eigenvectors with the lowest eigenvalues and form a matrix $\mathbf{U} \in \mathbb{R}^{N \times k}$ which has the eigenvectors as its columns.

Finally, a set of clusters $\mathcal{C}$ is obtained by running k-means algorithm on the row vectors of a matrix $\mathbf{U}$ (see supplementary for more detail), producing regions with all pixels from the corresponding cluster. Note that, at this stage, the resulting clusters are composed of both object and background masks—the object mask itself will be selected by our selection strategy described next.

6.3.3 Supervision with pseudo-mask from spectral clusters

Here, we first introduce our voting strategy for selecting a salient mask from a set of spectral clusters from different features and multiple k values (*i.e.*, cluster numbers), which utilises a *framing prior* and *distinctiveness prior*. Then, we describe our model, SelfMask, which is trained by using the selected salient masks as pseudo-masks for supervision.

Spectral cluster winner-takes-all voting

To choose the salient object among the mixture of foreground and background masks generated by spectral clustering, we propose a voting strategy based on two observations: (1) The spatial extent of an object rarely occupies the entire height and width of an image. (2) Salient object regions are likely to appear in multiple clusters across different self-supervised features as well as with different cluster numbers k. In other words, among k clusters from an application of clustering, we assume that at least one cluster encodes an object region within the image and that this holds for different features (*e.g.*, DINO, MoCov2, or SwAV). We call these priors the *framing prior* and *distinctiveness prior*, respectively. Note that the framing prior bears a resemblance to the centre prior [Judd et al. 2009], which states that salient objects are likely to be located near the center of an image, and the boundary prior [Wei et al. 2012], which presumes that the foreground object rarely touches the boundary of an image. However, the framing prior differs from these priors in that it is not related to a location of an object but rather the scale of an object within an image.

To use the framing prior and distinctiveness prior in practice, we first form a candidate set of masks by repeatedly applying spectral clustering to different features with multiple k values. Then, we treat masks whose spatial extent is as long as the width or height of the image as background masks, and eliminate them from the candidate set. Finally, we employ *winner-takes-all* voting: we pick the mask with the highest average pair-wise similarity w.r.t. IoU among all remaining masks as the final mask for salient objects (bottom of Fig. 6.2). There are two edge cases in the background elimination process we handle explicitly: (i) when no masks are left in the candidate set and (ii) when only two masks are left, sharing the same IoU. For the former case, we simply keep all the masks in the candidate set as every mask highlights regions spanning the spatial extent of the image, breaking the assumption of the framing prior. For the latter, we break ties randomly to pick one of the two masks.

SelfMask

Here, we describe our model architecture, training, and inference procedure.

Architecture. We base our salient object detection network, called SelfMask, on a variant of the MaskFormer architecture [Cheng et al. 2021b] which was originally proposed for the semantic/instance segmentation task.

SelfMask has two different sets of outputs: mask predictions and objectness scores for each mask. In detail, our model comprises an image encoder, a pixel decoder, and a transformer decoder (upper part of Fig. 6.2). The image encoder takes an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ as input and outputs feature maps $f(\mathbf{I}) \in \mathbb{R}^{h \times w \times D}$. Then, the feature maps are fed into the pixel decoder to produce upsampled features $g(f(\mathbf{I})) \in \mathbb{R}^{H \times W \times D}$. The feature maps are also passed to the transformer decoder which outputs n_q per-mask embeddings by using the feature maps as keys and values and the learnable embeddings as queries. The final mask predictions $\mathbf{M}$ are produced via matrix multiplication between the upsampled features and per-mask embeddings, followed by an element-wise sigmoid function $\sigma(\cdot)$:

$$\mathbf{M} = \{\mathbf{M}_i \mid \mathbf{M}_i = \sigma\left(g\left(f\left(\mathbf{I}\right)\right)\mathbf{q}_i\right), i = 1, ..., n_q\} \tag{6.5}$$

where $\mathbf{q}_i \in \mathbb{R}^D$ denotes the ith query (*i.e.*, per-mask embedding). For each mask $\mathbf{M}_i$, an objectness score $o_i \in [0, 1]$ is estimated by feeding the corresponding per-mask embedding $\mathbf{q}_i$ to a simple MLP with two hidden layers followed by a sigmoid function. Note that other encoder and decoder architectures can also be used here.

Objective. For training, we employ two objective functions: a mask loss and a ranking loss, denoted by $\mathcal{L}_{\text{mask}}$ and $\mathcal{L}_{\text{rank}}$, respectively. Given n_q mask predictions for an image from the model, we encourage all predictions to be similar to the pseudo-mask. Specifically, following [Cheng et al. 2021b], we use the Dice coefficient [Milletari et al. 2016], which considers the class-imbalance between foreground and background regions within an image, as the mask loss. It is worth noting that, unlike [Cheng et al. 2021b], we do not include the focal loss [Lin et al.



Figure 6.3: Sample visualisations of the pseudo-masks and predictions from our model on the ECSSD, DUT-OMRON, and DUTS-TE benchmarks. From left to right, the input image, ground-truth mask, a pseudo-mask decided by the proposed voting-based salient mask selection, and the prediction of our model are shown. Blue and orange coloured regions denote the intersection and difference between a ground-truth and a predicted mask. Best viewed in colour.

2017] in the mask loss since we find that it hinders convergence.

To decide which prediction best highlights the salient region in the image among the proposed candidates when $n_q > 1$, we rank the predicted masks based on their objectness score. Specifically, we first re-order the indices of the predicted masks by their mask loss in ascending order such that

$$\mathcal{L}_{\text{mask}}\left(\mathbf{M}_i, \mathbf{M}_{\text{pseudo}}\right) \leq \mathcal{L}_{\text{mask}}\left(\mathbf{M}_j, \mathbf{M}_{\text{pseudo}}\right) \tag{6.6}$$

for any $i < j$, where $\mathbf{M}_{\text{pseudo}}$ denotes the target pseudo-mask for the image. Then, we enforce the objectness score o_i of the mask $\mathbf{M}_i$ to be higher than the scores o_j for any $j > i$. As a consequence the model is encouraged to produce a higher score for a predicted mask that more closely resembles the pseudo-mask than other predictions. We instantiate this ranking loss as a *hinge loss* [Cortes and Vapnik 1995]:

$$\mathcal{L}_{\text{rank}} = \sum_{i=1}^{n_q-1} \sum_{j>i}^{n_q} \max\left(0,\ o_j - o_i\right). \tag{6.7}$$

Overall, our final objective function is as follows:

$$\mathcal{L} = \mathcal{L}_{\text{mask}} + \lambda \mathcal{L}_{\text{rank}} \tag{6.8}$$

where λ is a weighting factor, which is set to 1.0 across our experiments. Following [Carion et al. 2020; Cheng et al. 2021b], we compute the loss for outputs from each layer of the transformer decoder.

Inference. During inference, given n_q predicted masks for an image and their objectness score, we pick the mask with the highest score as the salient object detection and binarise it with a fixed threshold of 0.5.

6.4 Experiments

In this section, we first describe the datasets used in our experiments (Section 6.4.1) and provide implementation details (Section 6.4.2). We then conduct ablation study (Section 6.4.3) and report our results for salient object detection (Section 6.4.4).

6.4.1 Datasets

We use *DUTS-TR* [Wang et al. 2017], which contains 10,553 images, to train our model with the pseudo-masks generated by following Section 6.3.3. We emphasize that only images are used for generating pseudo-masks and training, without the corresponding labels. For our ablation study and comparison to previous work, we consider five popular saliency datasets including *DUT-OMRON* [Yang et al. 2013], which comprises 5,168 images of varied content with ground-truth pixel level masks; *DUTS-TE* [Wang et al. 2017], containing 5,019 images selected from the SUN dataset [Xiao et al. 2010] and ImageNet test set [Deng et al. 2009]; *ECSSD* [Shi et al. 2016] which contains 1,000 images that were selected to represent complex scenes; *HKU-IS* [Li and Yu 2015] which consists of 4,447 scene images with foreground/background sharing the similar appearances; *SOD* [Movahedi and Elder 2010] which contains 300 images with many images having multiple salient objects.

6.4.2 Implementation details

Networks. We use the ViT-S/8 architecture [Dosovitskiy et al. 2021] for the encoder, a bilinear upsampler with a scale factor of 2 for the pixel decoder, and 6 transformer layers [Vaswani et al. 2017] for the transformer decoder. For the MLP applied to per-mask queries (with a dimensionality of 384) that outputs a scalar value for the objectness score, we use three fully-connected layers with a ReLU activation between them. We set the same number of units for the hidden nodes as the input (*i.e.*, 384) and output a single value followed by a sigmoid.

Training details. We train our models for 12 epochs and optimise all parameters including the backbone encoder using AdamW [Loshchilov and Hutter 2019] with a learning rate of 6e-6 and the Poly learning rate policy [Ouali et al. 2020a; Xie et al. 2020; Chen et al. 2018b]. For data augmentation, we use random scaling with a scale range of [0.1, 1.0], random cropping to a size of 224×224 and random horizontal flipping with a probability of 0.5. In addition, photometric transformations include random color jittering, color dropping, and Gaussian blurring are applied. We run each model with three different seeds and report the average.

Metrics. In our experiments, we report intersection-over-union (IoU), pixel accuracy (Acc) and maximal F_β score (max F_β) with β^2 set to 0.3 following [Wang et al. 2022b]. Please refer to the supplementary for more details on these metrics.

6.4.3 Ablation study

In this section, we first conduct experiments to compare the effectiveness of spectral clustering and k-means when applied to self-supervised image encoders. Next, we quantitatively verify the performance of our winner-takes-all voting strategy for foreground mask selection and compare it to different saliency selection methods. Lastly, we investigate the effect of different number of queries on SelfMask.

Model	Arch.	Cluster.	DUT-OMRON k={2, 3, 4}	DUTS-TE k={2, 3, 4}	ECSSD k={2, 3, 4}
Convolutional Nets					
MoCov2 [He et al. 2020]	ResNet50	k-means	.375	.415	.500
MoCov2 [He et al. 2020]	ResNet50	spectral	**.387**	**.454**	**.627**
SwAV [Caron et al. 2020]	ResNet50	k-means	.399	.444	.542
SwAV [Caron et al. 2020]	ResNet50	spectral	**.401**	**.458**	**.590**
Vision Transformer					
DINO [Caron et al. 2021]	ViT-S/16	k-means	.377	.392	.541
DINO [Caron et al. 2021]	ViT-S/16	spectral	**.394**	**.417**	**.577**
DINO [Caron et al. 2021]	ViT-S/8	k-means	.369	.377	.551
DINO [Caron et al. 2021]	ViT-S/8	spectral	**.398**	**.411**	**.587**

Table 6.1: **Spectral clustering dominates k-means for self-supervised features.** We report upper bound IoUs to compare the quality of masks produced by k-means and spectral clustering on *self*-supervised features with two different encoder architectures (i.e., convolution- and transformer-based encoder). We report the average of the results from k={2, 3, 4}.

Model	Arch.	Cluster.	DUT-OMRON k={2, 3, 4}	DUTS-TE k={2, 3, 4}	ECSSD k={2, 3, 4}
Convolutional Nets					
ResNet [He et al. 2020]	ResNet50	k-means	**.337**	**.354**	**.444**
ResNet [He et al. 2020]	ResNet50	spectral	.310	.327	.437
Vision Transformer					
ViT [Dosovitskiy et al. 2021]	ViT-S/16	k-means	**.394**	**.411**	.542
ViT [Dosovitskiy et al. 2021]	ViT-S/16	spectral	.380	.400	**.551**

Table 6.2: **Spectral clustering and k-means perform comparably for fully-supervised features.** We report upper bound IoUs to provide a comparison of mask quality comparison between k-means and spectral clustering applied to *fully*-supervised features. We report the average of the results from k={2, 3, 4}.

Spectral clustering vs k-means clustering

We compare spectral clustering against a k-means [Lloyd 1982] clustering baseline on three salient object detection benchmarks. As the resulting segmentations from each algorithm are agnostic to foreground/background regions, we consider a best-case evaluation for both algorithms. In detail, given a groundtruth mask, we pick the cluster with the highest IoU w.r.t. the groundtruth. Such IoUs act as an upper-bound score among the clusters (*i.e.*, average best overlap in [Arbeláez et

al. 2014]).

To account for the effect of the cluster number k on performance of the clustering algorithms, we consider different k values from 2 to 4 and average the results. For the full results with each k value, please refer to the supplementary.

As shown in Table 6.1, we observe that object masks from spectral clustering consistently outperform k-means masks by a large margin. Interestingly, however, when using fully-supervised image encoders, the performance gain of spectral clustering diminishes (Table 6.2). These findings boil down to a simple summary: *while using self-supervised visual representations for grouping, spectral clustering is considerably superior to k-means, regardless of the choice of encoder architecture and self-supervised learning algorithm.*

Voting for salient object masks

Here, we conduct experiments to assess our voting method for selecting a foreground mask among the mask candidates. Since these experiments include ablations across hyperparameter choices, we conduct them on the HKU-IS [Li and Yu 2015] and SOD [Movahedi and Elder 2010], rather than the benchmarks used to compare to prior work.

In detail, we construct an initial corpus of mask candidates by clustering different self-supervised features with different number of clusters, as described in Section 6.3.2. For this, we build the candidate set with spectral clusters from different combinations of self-supervised features, *e.g.*, MoCov2/DINO or SwAV/MoCov2. It is important to do so to allow our voting-based method to leverage the distinctiveness prior across different features. In addition, for each combination, we experiment with 3 different k value settings: k=2, $\{2, 3\}$ or $\{2, 3, 4\}$ to account for various object scales, *e.g.*, a lower k tends to cover large regions, while a higher k segments smaller objects. We then evaluate the selected masks on the *HKU-IS* set. For reference we also compute an upper bound IoU, which is computed in a similar way as done in the previous section.

Effectiveness of clustering various self-supervised models with different number of clusters. As shown in Table 6.3, we make two observations: (i) IoU of both selected masks and upper bound masks, denoted by pseudo-mask and UB improves by increasing k across all feature combinations; (ii) using all three features (*i.e.*, DINO, MoCov2 and SwAV) results in better pseudo-masks than using two of the three features (*e.g.*, MoCov2 and SwAV). These support the distinctiveness prior, which assumes that at least one cluster represents a foreground region, and its application to our voting-based saliency selection method.

Effectiveness of the proposed voting approaches. We further validate the effectiveness of our voting scheme by comparing to different selection methods, *i.e.*, random selection and a centre prior [Judd et al. 2009]-based strategy. Specifically, we form a candidate set using DINO/MoCov2/SwAV features with $k = \{2, 3, 4\}$, which amounts to 27 masks in total. For the random strategy, we simply pick one of the masks uniformly from the candidates. For the centre prior selection strategy, we choose a mask whose average Euclidean distance to the image centre from its constituent pixel locations is lowest. In addition, we consider each method with or without utilising the framing prior to assess the influence of filtering out background mask. We evaluate the IoU of each case on the *HKU-IS* and *SOD* benchmarks.

As can be seen in Table 6.4, deploying the framing prior boosts IoU in all considered selection methods, with the proposed voting selection method performing best. The framing prior plays a crucial role in the voting process: voting without this prior performs much worse than its counterparts in both random and center-based selections on *HKU-IS*, and performs similarly to the random strategy on *SOD*. This is caused in large part by mistakes when selecting background masks as salient objects.

Features			k	Pseudo-mask	UB
DINO [Caron et al. 2021]	MoCov2 [He et al. 2020]	SwAV [Caron et al. 2020]			
✗	✓	✓	2	.508	.562
			2, 3	.561	.626
			2, 3, 4	.580	.658
✓	✗	✓	2	.473	.553
			2, 3	.538	.644
			2, 3, 4	.559	.682
✓	✓	✗	2	.459	.546
			2, 3	.536	.648
			2, 3, 4	.566	.688
✓	✓	✓	2	.511	.584
			2, 3	.567	.664
			2, 3, 4	**.590**	**.698**

Table 6.3: **Forming a candidate set with various self-supervised features and multiple k values improves IoU of both pseudo-masks and upper bound masks (UB).** We compare cases with different combinations of self-supervised features and cluster numbers of k=2, $\{2, 3\}$ or $\{2, 3, 4\}$ on the HKU-IS benchmark.

The influence of the number of queries

As described in Section 6.3.3, we train SelfMask using the selected salient masks as pseudo-masks. Here, we investigate the effect of the number of queries n_q in the Transformer decoder. For this, we train our model with n_q={5, 10, 20, 50, 100} on *DUTS-TR* and evaluate performance on the *HKU-IS* benchmark in terms of maxF_β for two settings, (i) using ground-truth masks to pick the best mask out of n_q mask predictions, denoted as the SelfMask upper bound (UB); (ii) taking the mask with the highest object score, denoted SelfMask.

As shown in Fig. 6.4, both SelfMask and SelfMask UB are fairly robust to the number of queries, initially increasing slightly with this hyperparameter (*i.e.*, predictions) before degrading after 20 queries. We conjecture that this is because a handful of queries are enough to localise the salient objects, while further predictions may make it challenging to appropriately rank the objectness of each prediction. For this reason, in the section that follows, we consider SelfMask with 20 queries, and pick the query with the highest objectness as our prediction during inference.

Selection	Framing prior	HKU-IS	SOD
random	✗	.206	.197
	✓	.464	.277
center	✗	.362	.122
	✓	.442	.392
voting (ours)	✗	.081	.200
	✓	**.590**	**.447**

Table 6.4: **Winner-takes-all voting and the framing prior both significantly improve mask quality.** We compare our voting strategy to different selection strategies along with the effect of framing prior under the IoU metric. Selection is performed from a candidate set including DINO, MoCov2 and SwAV features with $k = \{2, 3, 4\}$.

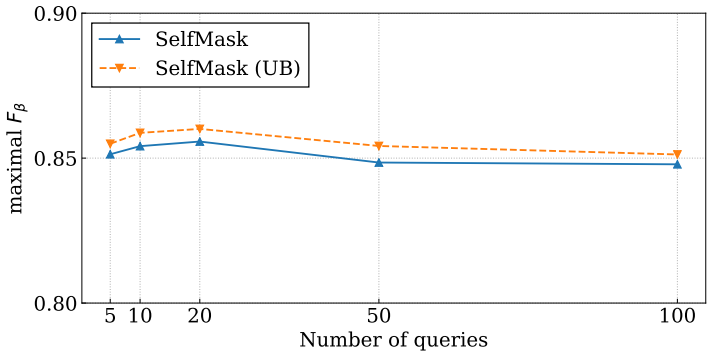


Figure 6.4: **Effect of number of queries on the performance of SelfMask on the HKU-IS dataset.** The model's prediction and its upper bound, denoted by SelfMask and SelfMask (UB) each, are shown.

6.4.4 Comparison to state-of-the-art unsupervised saliency detection methods

To compare with existing works on unsupervised SOD, we evaluate on three popular SOD benchmarks in terms of Acc., IoU, and maxF_β. Following [Wang et al. 2022b], we also report results after post-processing predictions with the bilateral solver [Barron and Poole 2016]. As shown in Table 6.5, while the pseudo-masks from spectral cluster voting already perform reasonably well compared to previous models, our self-trained model outperforms all existing approaches on all benchmarks. This suggests both that the model can learn to generalise effectively from noisy masks, and that the objectness score trained with the ranking loss is effective

Model	DUT-OMRON			DUTS-TE			ECSSD		
	Acc	IoU	maxF_β	Acc	IoU	maxF_β	Acc	IoU	maxF_β
HS [Yan et al. 2013]	.843	.433	.561	.826	.369	.504	.847	.508	.673
wCtr [Zhu et al. 2014]	.838	.416	.541	.835	.392	.522	.862	.517	.684
WSC [Li et al. 2015]	.865	.387	.523	.862	.384	.528	.852	.498	.683
DeepUSPS [Nguyen et al. 2019]	.779	.305	.414	.773	.305	.425	.795	.440	.584
BigBiGAN [Voynov et al. 2021]	.856	.453	.549	.878	.498	.608	.899	.672	.782
E-BigBiGAN [Voynov et al. 2021]	.860	.464	.563	.882	.511	.624	.906	.684	.797
Melas-Kyriazi et al. [Melas-Kyriazi et al. 2022c]	.883	.509	-	.893	.528	-	.915	.713	-
LOST [Siméoni et al. 2021]	.797	.410	.473	.871	.518	.611	.895	.654	.758
LOST [Siméoni et al. 2021] + BS	.818	.489	.578	.887	.572	.697	.916	.723	.837
TokenCut [Wang et al. 2022b]	.880	.533	.600	.903	.576	.672	.918	.712	.803
TokenCut [Wang et al. 2022b] + BS	.897	.618	.697	.914	.624	.755	.934	.772	.874
pseudo-masks (Ours)	.811	.403	-	.845	.466	-	.893	.646	-
SelfMask (Ours)	.901	.582	.680	.923	.626	.750	.944	.781	.889
SelfMask (Ours) + BS	**.919**	**.655**	**.852**	**.933**	**.660**	**.882**	**.955**	**.818**	**.956**

Table 6.5: **Comparison to state-of-the-art unsupervised saliency detection methods on three salient object detection benchmarks.** For all metrics, higher number indicates better results. The best score per column is highlighted in bold. BS denotes the application of the bilateral solver [Barron and Poole 2016] for post-processing. We observe that SelfMask yields improved performance over prior state-of-the-art approaches across all benchmarks.

for picking the best salient mask.

6.4.5 Broader impact

This work contributes a new framework to deliver performant unsupervised salient object detection. As such, it offers the potential to underpin a range of societally beneficial applications that are bottlenecked by annotation costs. These include improved low-cost medical image segmentation, crop measurement from aerial imagery, and wildlife monitoring. However, low-cost segmentation is a powerful dual-use technology, and we caution against its deployment as a tool for unlawful surveillance and oppression.

6.5 Conclusion

In this work, we address the challenging problem of unsupervised salient object detection (SOD). For this, we first observe that self-supervised features exhibit significantly greater object segmentation potential with spectral clustering than with

k-means. Inspired by this observation, we extract foreground regions among multiple masks generated with multiple types of features, and varying cluster numbers based on winner-takes-all voting. By using the selected masks as pseudo-masks, we train a saliency detection network and show promising results compared to previous unsupervised methods on various SOD benchmarks.

Acknowledgements

GS is supported by AI Factory, Inc. in Korea. WX is supported by Visual AI (EP/T028572/1). SA would like to thank Z. Novak and N. Novak for enabling his contribution. GS would like to thank Jaesung Huh for proof-reading.

Appendices

Appendices for this chapter can be found in the online version of the paper.[3]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[3] https://www.robots.ox.ac.uk/~vgg/research/selfmask

Chapter 7

ProMerge: Prompt and Merge for Unsupervised Instance Segmentation

This work has been accepted for publication at the European Conference on Computer Vision (ECCV), 2024.

ProMerge: Prompt and Merge for Unsupervised Instance Segmentation

Dylan Li[1*] Gyungin Shin[2,*✉]

[1]Meta Reality Labs

[2]Visual Geometry Group, University of Oxford

https://www.robots.ox.ac.uk/~vgg/research/promerge

Abstract

Unsupervised instance segmentation aims to segment distinct object instances in an image without relying on human-labelled data. This field has recently seen significant advancements, partly due to the strong local correspondences afforded by rich visual feature representations from self-supervised models (*e.g.*, DINO). Recent state-of-the-art approaches use self-supervised features to represent images as graphs and solve a generalised eigenvalue system (*i.e.*, normalised-cut) to generate foreground masks. While effective, this strategy is limited by its attendant computational demands, leading to slow inference speeds. In this paper, we propose Prompt and Merge (ProMerge), which leverages self-supervised visual features to obtain initial groupings of patches and applies a strategic merging to these segments, aided by a sophisticated background-based mask pruning technique.

*Equal contribution

✉Correspondence to: gyungin@robots.ox.ac.uk



Figure 7.1: **Qualitative examples of ProMerge, a simple yet effective *training-free* approach for unsupervised instance segmentation.** Despite its simplicity, it demonstrates strong segmentation performance.

ProMerge not only yields competitive results but also offers a significant reduction in inference time compared to state-of-the-art normalised-cut-based approaches. Furthermore, when training an object detector using our mask predictions as pseudo-labels, the resulting detector surpasses the current leading unsupervised model on various challenging instance segmentation benchmarks.

7.1 Introduction

Instance segmentation identifies and delineates each distinct object within an image, providing both its class and precise pixel-wise location. This capability is crucial for a wide range of applications, from autonomous driving systems [Cordts et al. 2016] that must navigate complex environments to medical imaging technologies that require accurate tumor segmentation [Bilic et al. 2019; He et al. 2023]. However, the cost of manually annotating dense masks for training data is prohibitively high, especially for domains such as medical imaging that require deep expertise.

To overcome the challenges with dense annotations, multiple endeavours have attempted to tackle category-agnostic instance segmentation in an unsupervised manner,[1] among which normalised-cut-based approaches [Wang et al. 2022b; Wang

[1]In this paper, we consider the *class-agnostic* setting following the prior works [Wang et al. 2022b; Wang et al. 2023b] on unsupervised instance segmentation.

et al. 2023b] have recently shown promise. These methods partition a graph representation of an image encoded in feature space into similar parts and solve a generalised eigenvalue system, specifically through spectral clustering with normalised-cut [Shi and Malik 2000]. Notably, the recently introduced MaskCut method [Wang et al. 2023b], which iteratively employs the TokenCut algorithm [Wang et al. 2022b] on a single image multiple times to generate numerous instance masks, demonstrates state-of-the-art performance. However, repeatedly solving this generalised eigenvalue problem incurs significant computational demands that delay the per-image inference time. Furthermore, its reliance on a fixed criterion to determine the number of segmentation masks per image (with MaskCut using three) may not capture the complete taxonomy of objects in dense, intricate scenes.

In our paper, we propose a simple yet effective framework called ProMerge, which sidesteps the aforementioned limitations. We start with generating initial masks of locally grouped patches by point-prompting self-supervised visual features (*e.g.*, DINO [Caron et al. 2021]). The initial masks, generated through computing the affinity between individual local patches and global patches, constitute a large set of mask proposals covering different parts of a given image. Following this mask generation process, we iteratively merge these local masks based both on their pixel overlap and their similarity in feature space. The effectiveness of ProMerge is demonstrated through its faster inference speed (about 3.6 times) and competitive performance compared to existing training-free[2] unsupervised methods on six benchmarks, including the densely annotated LVIS [Gupta et al. 2019] and SA-1B [Kirillov et al. 2023] datasets. Moreover, by training an object detector (*i.e.*, Cascade Mask R-CNN [Cai and Vasconcelos 2018]) with the mask predictions by ProMerge as pseudo-labels, we show that our framework outperforms the current leading unsupervised model (*i.e.*, CutLER [Wang et al. 2023b]).

Our contributions are three-fold: (i) We propose the ProMerge framework, composed of an initial mask generation step using point-prompted self-supervised

[2]Here, training-free is defined as not requiring training for segmentation.

visual features, an iterative mask merging that considers similarities in pixel and feature spaces, and a sophisticated mask pruning strategy; (ii) We compare the competitive performance of our approach to the popular normalised-cut-based methods on six standard instance segmentation benchmarks, including COCO2017 [Lin et al. 2014], COCO-20K [Vo et al. 2020], LVIS [Gupta et al. 2019], KITTI [Geiger et al. 2012], and subsets of Objects365 [Shao et al. 2019] and SA-1B [Kirillov et al. 2023]; and (iii) When training a class-agnostic object detector using the predictions by ProMerge as pseudo-labels, we show the resulting detector outperforms the leading unsupervised detector on the above datasets.

7.2 Related work

Our work is connected to three themes in the literature, including self-supervised visual representation learning, unsupervised single object detection/segmentation and unsupervised instance segmentation.

7.2.1 Self-supervised visual representation learning

Self-supervised learning in computer vision has advanced significantly by adopting the principle of learning from the intrinsic structure of data, drawing inspiration from how language models such as GPT [Brown et al. 2020] and BERT [Devlin et al. 2019] achieve semantic understanding from text. One strategy in self-supervised learning involves leveraging pretext tasks, such as those employed by Masked Autoencoders [He et al. 2021], wherein models learn by predicting the obscured parts of an image. Another set of strategies, embodied by SwAV [Caron et al. 2020], MoCo [He et al. 2020; Chen et al. 2020; Chen et al. 2021], and DINO [Caron et al. 2021; Oquab et al. 2023; Darcet et al. 2023], uses data augmentations to generate varied perspectives of images and aligns feature representations of these perspectives. Among these self-supervised paradigms for training encoders, DINO in particular encodes detailed segmentation information, a capability diminished

in models trained with supervised labels [Caron et al. 2021; Shin et al. 2022a]. In this work, by leveraging DINO's inherent grouping ability, we propose a simple yet effective approach to instance segmentation without explicit labels.

7.2.2 Unsupervised single object detection and segmentation

Unsupervised object detection and segmentation aim to localise a single, dominant object in an image in an unsupervised manner with a bounding box or a segmentation mask, respectively.

LOST [Siméoni et al. 2021] extracts features from a self-supervised network and isolates the foreground by first identifying the seed patch with the lowest count of positive correlations with other patches. A seed expansion strategy is then used to include additional patches that correlate positively with the original seed patches.

Another line of work uses normalised-cut-based methods [Wang et al. 2022b; Melas-Kyriazi et al. 2022a; Shin et al. 2022a] to distinguish the foreground from the background. This class of methods uses the eigendecomposition of the Laplacian matrix derived from a feature affinity matrix constructed with self-supervised features. The resulting eigenvectors, processed with traditional clustering or thresholding methods, can be translated into meaningful segmentations in pixel space.

This process facilitates the identification of coherent regions, enabling the separation of primary image elements from the background. While these prior works serve as foundational methods for unsupervised multi-object localisation and segmentation, they primarily focus on segmenting or identifying the location of a *single* salient object, limiting their generalisability and performance on multi-object localisation tasks.

7.2.3 Unsupervised instance segmentation

Unsupervised instance segmentation aims to identify *multiple* objects in an image without human labels, introducing a more complex challenge than the aforementioned single object detection and segmentation. An early attempt [Van Gansbeke et al. 2022] generates a single salient mask per image, and trains the Mask R-CNN detector [He et al. 2017] using the generated masks as pseudo-labels. However, the single mask generation approach does not provide sufficient mask instances per image, resulting in suboptimal detection performance.

Recent approaches [Wang et al. 2022b; Wang et al. 2023b], on the other hand, propose methods for generating multiple instance masks per image, that are used to train a general object detector. MaskCut [Wang et al. 2023b] has demonstrated promising outcomes by iteratively applying a normalised-cut-based single object segmentation technique (*i.e.*, TokenCut [Wang et al. 2022b]) to images and training a robust object detector with pseudo-labels generated through the MaskCut algorithm. Despite its effectiveness, MaskCut uses repeated eigenvalue system resolutions for the normalised cut, incurring significant computational demands. Additionally, MaskCut limits itself to three segmentations per image. In contrast, ProMerge eschews computationally intensive eigenvalue calculations in favour of using raw feature affinities and does not impose a restriction on the number of segmentations per image. By doing so, it offers a more computationally efficient alternative for the instance segmentation task while achieving higher precision and recall.

7.3 Method

In this section, we first describe the problem scenario (Section 7.3.1) and introduce ProMerge, a simple yet effective prompt-and-merge method to tackle unsupervised instance segmentation (Section 7.3.2). We then describe a background-based mask pruning strategy to extract foreground masks from a set of prompted masks that

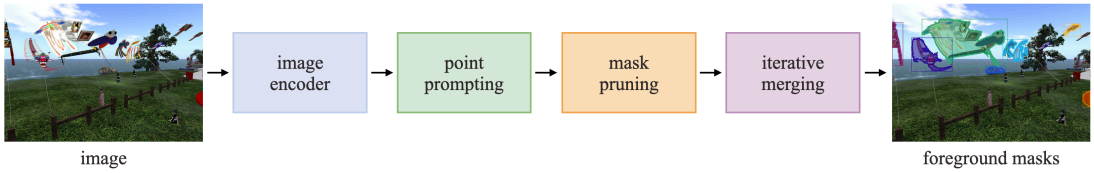


Figure 7.2: **An overview of ProMerge.** Given an input image, we obtain initial mask proposals by prompting visual features from an image encoder using a 2D point grid. Then, the noisy proposals are filtered through the proposed background-based mask pruning. The final predictions are made by iteratively merging the remaining foreground masks.

increases the effectiveness of prompting and merging (Section 7.3.3). We finally describe a pseudo-label training scheme in which an object detector is trained with pseudo-labels generated by ProMerge (Section 7.3.4).

7.3.1 Problem formulation

We address the challenging task of instance segmentation in an unsupervised manner. Given an image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$, where H and W denote the height and width of the image, we aim to produce a set of N instance binary masks $\mathcal{M} = \{\mathbf{M}_l \in \{0,1\}^{H\times W} | l = 1, ..., N\}$ without relying on any human-labelled data.

7.3.2 ProMerge: Prompt and Merge

Point-prompting visual features. Our approach begins with generating preliminary mask proposals. We use visual features, denoted as $\mathbf{F} = \{\mathbf{f}_{ij} \in \mathbb{R}^c | i = 1, ..., h,\ j = 1, ..., w\}$, that are obtained by feeding an image into an image encoder (*e.g.*, patch tokens for Vision Transformers (ViT) [Dosovitskiy et al. 2021]). Here, c, h, and w represent the channel, height, and width dimensions of the features. To generate initial masks from the visual features (*i.e.*, patch tokens), we consider the technique of point-prompting, wherein a 2D grid of K equally spaced patch tokens is selected as seeds for mask generation. This subset of the selected tokens $\mathcal{P} = \{\mathbf{p}_l \in \mathbb{R}^c | l = 1, ..., K\}$, which we call the set of *prompt tokens*, is individually compared with all of the patch tokens in the image. By comparing each prompt

token in a one-to-all manner via a similarity measure, we generate an affinity matrix $\mathbf{A}_l \in [-1, 1]^{h \times w}$ for each prompt token $\mathbf{p}_l$. In this paper, we use *key* features from the last attention layer of a self-supervised image encoder (*i.e.*, DINO) as patch tokens [Wang et al. 2023b] and compute the cosine similarity between them. That is,

$$\mathbf{A}_l = (A_{l;ij}) = \frac{\mathbf{p}_l \cdot \mathbf{f}_{ij}}{||\mathbf{p}_l||_2 ||\mathbf{f}_{ij}||_2} \tag{7.1}$$

where $||.||_2$ denotes L2 norm. We then apply a bipartition threshold, τ_b, to the affinity matrix to obtain a binary mask $\mathbf{M}_l$.

Merging prompted masks. Given the prompted masks above, we consider an iterative clustering method, wherein masks, sorted by area in descending order, are sequentially merged with a set of larger masks processed at the previous iterations. The processed masks serve as bases for merging smaller masks that are introduced in later iterations.

In comparing a new, smaller mask with a previously processed, larger mask, we consider two straightforward conditions. First, we use the Intersection-over-Area (IoA) metric to determine if the smaller mask should be merged with the larger mask. If the ratio of the intersection area between two masks to the area of the smaller mask exceeds a certain threshold, denoted as $\tau_{\text{IoA}}^{\text{merge}}$, the smaller mask is combined with the larger. We choose IoA over the more conventional Intersection-over-Union (IoU) due to IoU's diminished effectiveness in cases where a large mask merges with a significantly smaller one, such as when combining an object's main body with an appendage. Note that the IoA criterion alone is insufficient, as it prevents merging two masks unless they exhibit substantial overlap in pixel space. To allow for merging semantically similar masks with a small overlap, we consider the second condition based on feature similarity. Given a pair of masks, we compute the cosine similarity between their average patch embeddings and merge them if the feature similarity is over a threshold τ_f^{merge}. If a new mask meets at least one of the merging conditions with more than one previously processed mask, the new mask and all compatible masks are merged together. Conversely, if

it does not satisfy a merge condition with any of the previously processed masks, it remains unchanged. This mask will then be compared with subsequent masks in later iterations for potential merging.

In summary, through our merge algorithm, we start with disparate, potentially overlapping masks, and end up with semantically-related mask groupings.

7.3.3 Background-based mask pruning

While the prompt-and-merge framework thus far is intuitive, we observe that it suffers from poor performance in isolation, due to the noisy background masks among the prompted masks. We introduce a mask pruning strategy based on background prediction between the prompting and merging steps. We rely on a two-step process that (i) groups the initial K masks into the foreground or background, and aggregates the background masks via pixel-wise voting to produce a single, fine background mask for the image and (ii) uses the predicted background mask to filter out noisy foreground masks from the initial mask proposals. Each step in the mask pruning strategy is detailed below.

Background aggregation. Recall that after prompting the visual features with K equally spaced points in a 2D grid, we obtain K binary masks. We then classify each of the masks as either a foreground or background candidate. We employ a simple heuristic: a background mask is likely to contain a majority of pixels along the edge for at least two of the edges of the image. Specifically, we consider a mask as background if more than one of its sides contains a number of positive pixels that exceeds half the length of that side. Then, we create a single representative background mask for the image by applying a pixel-wise voting scheme to the background candidates. Formally, given the set of background candidate masks $\mathcal{B} = \{\mathbf{M}_l^{\text{bg}}\}$, a pixel value at (i, j) of the aggregated background mask $\tilde{M}_{ij}^{\text{bg}}$ is determined with:

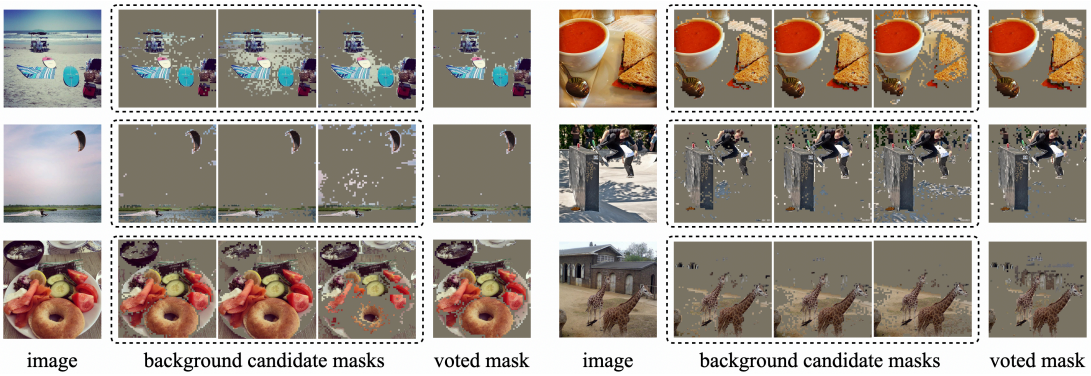


Figure 7.3: **Qualitative examples of the pixel-wise voting.** For each case, an original image, background candidate masks (only three masks are shown for visual purposes), and the voted mask are visualised. The voted background mask, $\tilde{M}^{bg}$ effectively filters out the background, leaving only foreground regions despite the noisy candidate masks.

$$\tilde{M}^{\text{bg}}_{ij} = \left[\frac{\sum_{l=1}^{|\mathcal{B}|} M^{\text{bg}}_{l;ij}}{|\mathcal{B}|} > 0.5 \right] \tag{7.2}$$

where the [.] operator is the indicator function, which returns 1 if the condition within the operator is satisfied, and 0 otherwise. The condition in the operator checks whether over the half of the background candidate masks have a value of 1 at (i, j). As such, we obtain a single representative background mask per image, represented by $\tilde{\mathbf{M}}^{\text{bg}} \in \{0, 1\}^{h \times w}$. Some visual examples are shown in Fig. 7.3.

Foreground filtering. With the background mask $\tilde{\mathbf{M}}^{\text{bg}}$, we exclude prompted masks that are more likely to belong to the background before the merge process. For this filtering step, we consider three separate approaches: (i) intersection-based (ii) similarity-based filtering and (iii) the proposed Cascade filtering.

For intersection-based filtering, we use a simple prior that any foreground masks considered for the following merge process should not significantly intersect with the voted background $\tilde{\mathbf{M}}^{\text{bg}}$. Similarly to Section 7.3.2, we use IoA, as we want the metric to be invariant to the size disparity between a candidate foreground mask and $\tilde{\mathbf{M}}^{\text{bg}}$. If the intersection of a foreground mask and $\tilde{\mathbf{M}}^{\text{bg}}$ divided by the area of the mask is greater than $\tau^{\text{bg}}_{\text{IoA}}$, we regard the mask as belonging to the background

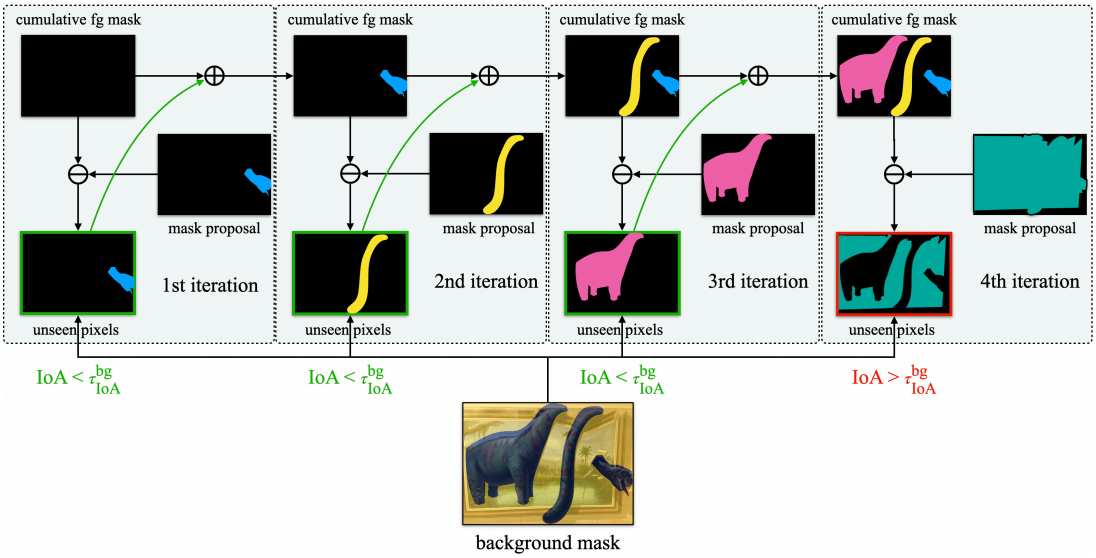


Figure 7.4: **An illustration of the proposed Cascade mask filtering process.** For each iteration of the proposed method, we evaluate the newly proposed mask by focusing on the pixels that have not yet been covered by the cumulative foreground mask, which is an aggregation of pixels from mask proposals in preceding iterations. If these previously unseen pixels demonstrate a significant overlap with the background mask, quantified by the Intersection-over-Area (IoA) metric, the mask proposal for that iteration is subsequently disregarded. An example of this can be observed in the fourth iteration (rightmost), in which the mask proposal is eliminated due to its high IoA with the background mask. Note that in the figure, the feature similarity condition is not shown for visual clarity. See the text for details.

and exclude it in the merging process.

For similarity-based filtering, we prune masks with a high similarity with the background mask in feature space. We again compute the mean patch embeddings for the candidate foreground mask and $\tilde{\mathbf{M}}^{\text{bg}}$, before calculating their normalised cosine similarity. If the similarity value is over a threshold, we exclude the mask from the merging process.

In the proposed Cascade filtering approach (shown in Fig. 7.4), we initially sort the prompted masks in ascending order based on their area. We then proceed through these masks sequentially, maintaining a cumulative ledger of pixels already incorporated into previous masks. At each step, we identify pixels in the current mask that have not yet been considered in a prior mask. We then calcu-

late the IoA between these 'new' pixels and $\tilde{\mathbf{M}}^{\text{bg}}$. We also compute the feature similarity between the current mask and $\tilde{\mathbf{M}}^{\text{bg}}$ in a manner similar to the aforementioned similarity-based filtering. If either of these measures with $\tilde{\mathbf{M}}^{\text{bg}}$ exceeds their respective thresholds, the mask is excluded from the subsequent merge process.

7.3.4 ProMerge+: Training an object detector with pseudo-labels from ProMerge

Following [Wang et al. 2023b], we train an object detector on ProMerge predictions generated from inference on images in a large-scale image dataset (*i.e.*, ImageNet2012 [Deng et al. 2009]). The purpose of this pseudo-label training is twofold: firstly, to obtain an object detector with better performance by learning from the noisy pseudo-labels; and secondly, to assess the detector's ability to generalize across different data distributions by training on images from one dataset (*e.g.*, ImageNet2012) and evaluating on another (*e.g.*, SA-1B [Kirillov et al. 2023]). The trained detector, ProMerge+, surpasses performance and zero-shot transfer capabilities of the current leading model.

7.4 Experiments

In this section, we first provide implementation details (Section 7.4.1) and compare our method with the state-of-the-art-methods (Section 7.4.2). Then, we provide extensive ablation study to analyse our approach (Section 7.4.3).

7.4.1 Implementation details

Our implementation is based on the PyTorch [Paszke et al. 2019] and Detectron2 [Wu et al. 2019] libraries, and A100 GPUs are used for our experiments unless otherwise stated.

Datasets. We evaluate our ProMerge on six benchmarks including COCO2017 [Lin

et al. 2014], COCO-20K [Vo et al. 2020], LVIS [Gupta et al. 2019], KITTI [Geiger et al. 2012], subsets of Objects365 [Shao et al. 2019] and SA-1B [Kirillov et al. 2023] (44K and 11K images, respectively). Among these, SA-1B is the most challenging benchmark due to the densely-annotated fine-grained masks, with an average of 101 segmentation masks per image. To train ProMerge+, we use unlabelled ImageNet2012 training images [Deng et al. 2009] (1.2M images) and evaluate the resulting model on the six benchmarks in a zero-shot manner (i.e., the model is not trained with images sharing the same data distribution as evaluation data). For the ablation study, we use COCO2017 following [Wang et al. 2023b].

Evaluation metrics. We evaluate our methods based on average precision (AP) and recall (AR), the standard metrics for the instance segmentation task.

Inference of ProMerge. We follow the previous work [Wang et al. 2023b] for our inference setting. Specifically, we use DINO [Caron et al. 2021] with the ViT-B/8 architecture [Dosovitskiy et al. 2021] as an image encoder and input images are resized to 480×480 pixels before being fed into the encoder. We apply Conditional Random Field (CRF) [Krähenbühl and Koltun 2011b] to an output mask for post-processing. Additionally, we split connected components of initial masks before merging, which we find beneficial in terms of both AP and AR (shown in Section 7.4.3).

Pseudo-label training. When we train an object detector with pseudo-masks generated by ProMerge, we use the Cascade Mask R-CNN model [Cai and Vasconcelos 2018] with the ResNet50 backbone [He et al. 2016] initialised with DINO features [Caron et al. 2021]. We train the detector for 160K iterations using the SGD optimizer with a batch size of 16, a momentum of 0.9, and a learning rate of 0.005, which is decreased by a factor of 5 during training.

method	COCO2017 AP^{mk}	COCO2017 AR^{mk}_{100}	COCO-20K AP^{mk}	COCO-20K AR^{mk}_{100}	LVIS AP^{mk}	LVIS AR^{mk}_{100}	KITTI AP^{mk}	KITTI AR^{mk}_{100}	Objects365† AP^{bb}	Objects365† AR^{bb}_{100}	SA-1B AP^{mk}	SA-1B AR^{mk}_{100}	Average AP	Average AR_{100}
Training-free methods														
TokenCut [Wang et al. 2022b]	2.0	4.4	2.7	4.6	0.9	1.8	**0.3**	1.5	1.1	2.1	1.0	0.3	1.3	2.5
MaskCut [Wang et al. 2023b]	2.2	6.9	**3.0**	6.7	0.9	2.6	0.2	**2.2**	1.7	4.0	0.8	0.6	1.5	3.5
ProMerge	**2.4**	**7.5**	**3.0**	**7.4**	**1.3**	**3.3**	**0.3**	1.9	**2.2**	**6.0**	**1.2**	**0.8**	**1.7**	**4.5**
Models trained with pseudo-labels														
CutLER‡ [Wang et al. 2023b]	8.7	24.9	8.9	25.1	3.4	16.6	3.9	23.3	11.5	34.3	5.5	13.5	7.0	22.9
ProMerge+	**8.9**	**25.1**	**9.0**	**25.3**	**4.0**	**17.7**	**5.4**	**25.7**	**12.2**	**35.8**	**7.8**	**16.3**	**7.9**	**24.3**

Table 7.1: **Comparison between training-free methods (top) and models trained with pseudo-labels (bottom).** CutLER and ProMerge+ are trained with pseudo-labels generated by MaskCut and ProMerge, respectively. †Only ground-truth bbox annotations available. ‡Re-implemented with a single round of training for a fair comparison.

method	FPS
TokenCut [Wang et al. 2022b]	0.34
MaskCut [Wang et al. 2023b]	0.15
ProMerge	**0.54**

Table 7.2: **Speed comparison.** Our method is approximately 3.6 times faster in FPS compared to MaskCut.

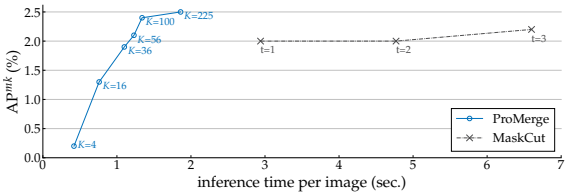


Figure 7.5: **AP^{mk} vs inference time on COCO2017.** K denotes the number of prompt tokens. MaskCut uses t=3 by default in the original paper.

7.4.2 Main results

In this section, we compare our methods to the state-of-the-art methods with both standard evaluation metrics and inference speed.

Comparison to state-of-the-art methods. We first compare ProMerge to *training-free* methods including TokenCut [Wang et al. 2022b] and MaskCut [Wang et al. 2023b] algorithms. In Table 7.1 (top), ProMerge demonstrates superior performance in both average precision (AP) and average recall (AR_{100}) compared to the existing methods. The overall higher recall of ProMerge is attributed to its flexibility in not requiring a predetermined number of masks per image. Qualitative results of each method can be found in the supplementary.

We next compare ProMerge+, a class-agnostic object detector trained with the predictions of ProMerge on ImageNet as pseudo-labels, to the state-of-the-art model (i.e., CutLER [Wang et al. 2023b]). As can be seen in Table 7.1 (bottom),

ProMerge+ outperforms CutLER [Wang et al. 2023b] by 0.9 and 1.4 in AP and AR_{100}, respectively, on average. Note that we use the same detector architecture (i.e., Cascade Mask R-CNN [Cai and Vasconcelos 2018]) and the training recipe as CutLER. The sole difference is that CutLER is trained with predictions from MaskCut and ProMerge+ is trained with ones from ProMerge. These results demonstrate the effectiveness of using our higher quality pseudo-labels for training an object detector.

Speed comparison. Unlike the state-of-the-art training-free methods that depend on solving the computationally intensive normalised-cut algorithm, our method avoids this complexity, enabling faster inference. To demonstrate this, we compare the inference speed of our method with that of normalised-cut-based methods in terms of Frames Per Second (FPS). In detail, we measure the timing for each of TokenCut, MaskCut, and ProMerge across 200 images, using the first 100 images as a warm-up phase to ensure accurate measurements.[3] As shown in Table 7.2, ProMerge runs at 0.54 FPS, which is about 1.6 and 3.6 times faster than TokenCut and MaskCut, respectively. In addition, we compare the speed differential between ProMerge and MaskCut under various inference configurations. For MaskCut, we adjust the number of repetitions (t) for solving the eigenvalue problem, which significantly impacts its runtime. For our method, we vary the number of prompt tokens (K). As can be seen in Fig. 7.5, ProMerge provides a more advantageous trade-off between inference speed and performance in AP^{mk} compared to MaskCut. It achieves higher AP^{mk} values when K exceeds 100, while also being significantly faster—approximately 4.9 times faster at $K = 100$.

7.4.3 Ablation study

Here, we conduct an extensive ablation study to analyse the effect of each component in ProMerge. Specifically, we explore the effects of pixel-wise voting for

[3]We use a single RTX 3080 GPU and a 12th Gen Intel(R) Core(TM) i7-12700K chipset for this experiment.

Prompting	Merging	Mask Pruning	CC Splitting	AP^{mk}_{50}	AP^{mk}	AR^{mk}_{100}
✓	✗	✗	✗	0.8	0.5	1.6
✓	✓	✗	✗	0.6	0.4	1.0
✓	✓	✓	✗	4.4	2.1	4.7
✓	✓	✓	✓	**5.6**	**2.4**	**7.5**

Table 7.3: **Effect of individual components.** A naive prompting and merging approach suffers poor performance, while applying the proposed background-based mask pruning (Mask Pruning) allows for a notable increase in performance, which is further enhanced by connected component splitting (CC Splitting).

background aggregation, foreground filtering methods, different merging conditions, and varied hyperparameter choices.

Effect of each component. We identify major components that affect the performance of our approach: (i) prompting and merging; (ii) background-based mask pruning (denoted as Mask Pruning); and (iii) connected component splitting (denoted as CC Splitting). In Table 7.3, we show the influence of each component. Notably, a naive approach that relies solely on prompting and merging suffers from poor performance, with 0.4 AP^{mk} and 1.0 AR^{mk}_{100}, which are worse than a prompting-only approach. However, adding background-based mask pruning greatly boosts both AP^{mk} and AR^{mk}_{100} by 1.7% and 3.7%. CC-splitting further increases AP^{mk} by 0.3% and AR^{mk}_{100} by 2.8%. These results demonstrate that though the core of prompt-and-merge is straightforward, the competitive performance of our approach is facilitated by incorporating sophisticated components such as background mask pruning.

Effect of pixel-wise voting. In the background-based mask pruning process, we use a pixel-wise voting strategy to aggregate background candidate masks to produce a single, representative background mask for a given image. We consider the case where we substitute voting with a simpler non-voting mechanism, and compare it with the voting based mechanism in ProMerge. In the non-voting experiment, we sum up all background candidate masks. If a pixel at location (i, j) has at least one mask with a value of one (*i.e.*, positive pixel), it is regarded as a background pixel. As highlighted in Table 7.4a, voting leads to a significant en-

voting	AP_{50}^{bb}	AP^{bb}	AR_{100}^{bb}	AP_{50}^{mk}	AP^{mk}	AR_{100}^{mk}
✗	4.1	1.6	5.0	3.2	1.2	4.2
✓	**6.0**	**3.0**	**8.6**	**5.6**	**2.4**	**7.5**

(a) Effect of voting

filter. method	AP_{50}^{bb}	AP^{bb}	AR_{100}^{bb}	AP_{50}^{mk}	AP^{mk}	AR_{100}^{mk}
IoA	4.7	2.1	6.8	4.0	1.7	5.8
feat.	5.0	2.3	5.9	4.6	2.0	5.2
Cascade	**6.0**	**3.0**	**8.6**	**5.6**	**2.4**	**7.5**

(b) Impact of filtering criteria

Table 7.4: **Influence of the voting strategy and filtering conditions. (Left)** We note that using the proposed pixel-wise voting for obtaining a representative background mask allows for a notable gain in performance. **(Right)** Comparing the Intersection-over-Area-based filtering (denoted as IoA) and the feature similarity-based filtering (denoted as feat.), the former demonstrates higher recall, whereas the latter excels in precision. Our proposed Cascade filtering outperforms both in all evaluated metrics, showcasing its effectiveness.

hancement in both average precision and recall. In the absence of pixel-wise voting, the aggregated background often encompasses not only the actual background but also parts of the foreground in the image, which detracts from the overall performance. Conversely, the application of the voting strategy more accurately isolates the background region.

Effect of foreground filtering method. Given the background mask obtained from the pixel-wise voting above, we filter prompted masks based on the background mask as described in Section 7.3.3. Here, we compare three different methods including (i) IoA-based, (ii) feature-similarity-based, and (iii) our proposed Cascade filtering strategies. For the IoA-based filtering, we discard a mask proposal if it has an IoA with the background mask exceeding 0.5, meaning more than half of the proposal's pixels are part of the background. In the feature-similarity-based approach, we classify a proposal as background if the cosine similarity between the normalised mean embeddings of the candidate foreground mask and the background exceeds 0, implying that the proposal's embedding closely aligns with that of the background.

From Table 7.4b, we make the following key observations. First, IoA-based filtering, demonstrates higher recall but lags in precision compared with similarity-based filtering. This discrepancy can be attributed to the latter's tendency to filter out more masks, irrespective of their degree of area overlap with the background,

feat.	IoA	AP^{bb}_{50}	AP^{bb}	AR^{bb}_{100}	AP^{mk}_{50}	AP^{mk}	AR^{mk}_{100}
✗	✓	5.3	2.3	7.9	4.4	1.9	6.9
✓	✗	5.8	2.7	8.4	4.9	2.2	7.2
✓	✓	**6.0**	**3.0**	**8.6**	**5.6**	**2.4**	**7.5**

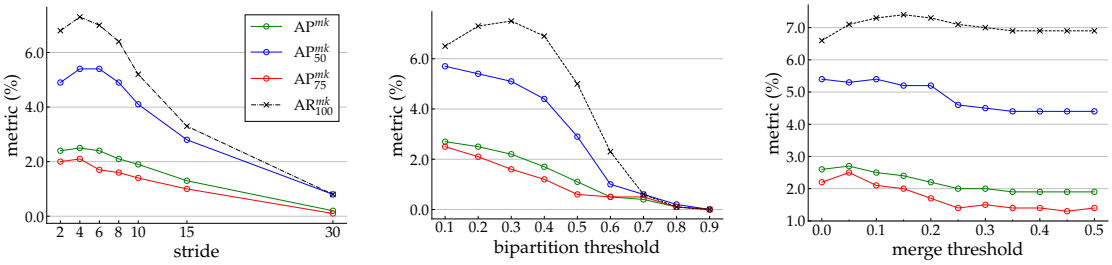
Table 7.5: **Effect of merging conditions.** Considering both feature similarity (denoted as feat.) and intersection-over-area (denoted as IoA) for merging a pair of masks yields further gain.

which in turn affects recall. Second, our proposed Cascade filtering emerges as the superior method. Its effectiveness stems from retaining smaller object masks while filtering larger masks that might incorrectly merge these smaller ones (in the following merge step). In other words, as Cascade filtering focuses specifically on the new, unseen pixels (and their respective relationships with the background), it lifts precision and recall by preventing the amalgamation of smaller entities into a larger, encompassing mask during mask merging.

Effect of merging conditions. In the merging process, we consider two different conditions for merging: (i) Intersection-over-Area (denoted as IoA) and (ii) feature similarity (denoted as feat.) between two masks. As shown in Table 7.5, we observe that the feature similarity condition plays a more significant role than the IoA condition when used separately. However, allowing both conditions leads to the best performance, indicating that they are complementary conditions.

Hyperparameter analysis. Lastly, we conduct experiments to explore the impact of various hyperparameters, including grid size, bipartition threshold for obtaining initial masks (τ_b), and feature similarity threshold for merging (τ_f^{merge}). We refer to the grid size parameter as 'stride', where the stride is inversely proportional to the grid size. For instance, with a stride of 2 and a spatial dimension of 60×60 of DINO feature embeddings, the grid size is reduced to 30×30, resulting in 900 initial prompted masks (i.e., $K = 900$). Conversely, a stride of 30 results in a grid size of 2×2, yielding only 4 initial prompted masks (i.e., $K = 4$).

In Fig. 7.6a, we present the performance of ProMerge using strides of {2, 4, 6, 8, 10, 15, 30}. The stride significantly influences performance, with a stride of 4 yielding the best results, while larger strides such as 15 and 30 lead to diminished



(a) Effect of stride (b) Effect of τ_b (c) Effect of τ_f^{merge}

Figure 7.6: **Hyperparameter analysis.** We note that stride (*i.e.*, the Euclidean distance between two nearest point prompts in a grid), bipartition threshold (τ_b) for binarising prompted masks, and feature similarity threshold (τ_f^{merge}) for merging masks play important roles in our approach. The default setting uses stride=4, τ_b=0.2, and τ_f^{merge}=0.1, respectively. Best viewed in colour.

performance. This decline is attributed to the reduced number of masks generated at higher stride values.

We next examine the performance variation associated with different values of the bipartition threshold τ_b, which is employed to binarise initial soft masks derived from prompting. A lower τ_b results in prompted masks having a larger area, while a higher τ_b leads to smaller mask areas. As illustrated in Fig. 7.6b, optimal average precision and recall are achieved between 0.1 and 0.3, with the highest precision at $\tau_b = 0.1$ and the highest recall at $\tau_b = 0.3$. Performance in both metrics declines as τ_b increases, suggesting that ProMerge benefits more from larger initial masks. Consequently, we set τ_b to 0.2 for all our experiments.

We also investigate the impact of the feature similarity threshold τ_f^{merge}, which determines whether two masks should be merged. A lower τ_f^{merge} value leads to the merging of mask pairs even with minimal similarity, while a higher value requires a high degree of similarity for merging. Fig. 7.6c demonstrates that our method's performance significantly varies with τ_f^{merge} values between 0.0 and 0.3, but levels off at higher values. This plateau suggests that beyond a τ_f^{merge} of 0.3, few mask pairs meet the merging criterion, thereby minimally impacting the overall performance metrics. Consequently, we opt for a $\tau_f^{\text{merge}} = 0.1$ in our experiments.

7.5 Discussion

We observe that a simple approach of prompting and merging masks is competitive with computationally intensive normalised-cut-based approaches [Wang et al. 2022b; Wang et al. 2023b]. We primarily attribute the success of ProMerge to two properties: (i) unlike in [Wang et al. 2022b; Wang et al. 2023b], we do not assume a fixed number of mask predictions per image and allow the algorithm to flexibly make predictions, which we find particularly helpful when multiple objects are present in a given image; (ii) we use the proposed sophisticated background-based filtering method, which excludes masks that overlap with the background of an image.

ProMerge shows promise but still has room to improve compared to fully-supervised methods, such as SAM [Kirillov et al. 2023]. We believe that the gap partly results from utilizing self-supervised features that are not trained with a pretext task driven by object localisation or segmentation. Training and adopting self-supervised features specifically tailored for object localisation or segmentation could significantly bridge this gap.

7.6 Conclusion

Our work introduces Prompt and Merge (ProMerge), a novel method to unsupervised instance segmentation that capitalises on the strong local correspondences afforded by self-supervised visual features. By iteratively merging initial patch groupings and employing a sophisticated background-based mask pruning technique, ProMerge achieves competitive performance with a significant reduction in inference time compared to state-of-the-art normalised-cut-based methods on standard benchmarks. Moreover, the application of our method in training an object detector with pseudo-labels demonstrates superior performance, surpassing the leading unsupervised segmentation model.

Acknowledgements. This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk). All authors appreciate the anonymous reviewers for the thoughtful suggestions and Fengting Yang for the kind advice on the rebuttal. DL would like to also thank Eric Tran and Robert Wang for their support while DL conducted his research at Meta. GS would like to thank Zheng Fang for the enormous support.

Appendices

Appendices for this chapter can be found in the online version of the paper.[4]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[4] https://www.robots.ox.ac.uk/~vgg/research/promerge

Chapter 8

Discussion

Here, we first summarise the key works presented in this thesis (Section 8.1), and suggest directions for future research (Section 8.2). We then conclude the thesis (Section 8.3).

8.1 Summary

Point-level active learning. In Chapter 2, we present the paper, "All You Need Are a Few Pixels: Semantic Segmentation with PixelPick," where we introduce a point-level active learning scheme for segmentation for the first time. This method significantly reduces the number of annotations required to train a performant segmenter by allowing the model to request labels only for uncertain pixels. Interestingly, our observations include: (i) the pixels requested for labels tend to lie near the boundaries of objects, revealing that boundary pixels are of greater importance than the inner pixels; and (ii) only a handful of labels for such pixels per image are necessary to train a segmenter nearly as effective as a fully-supervised one.

Due to its label-efficient nature, this work has subsequently been cited by papers in various fields such as wildlife conservation [Bakana et al. 2024], microscopy [Hoshina et al. 2024], and skin lesion detection [Liang et al. 2023].

Retrieval-augmented segmentation. In Chapter 3, we introduce the paper,

"ReCo: Retrieve and Co-segment for Zero-shot Transfer," where we propose using a pre-trained visual-language model to enable the categorisation of objects via its image retrieval capability, without relying on manual supervision. This method differs from conventional semantic segmentation approaches, which typically limit the range of categories a segmenter can classify to those included in a training dataset. Instead, our proposed method allows the model to generalise to as many categories as the visual-language model can recognise. We further strengthen the approach by replacing the co-segmentation module with a strong unsupervised object segmenter (Chapter 4) and expand it for instance segmentation (Chapter 5). We believe that as more advanced vision-language models are developed, these methodologies will further enhance the capabilities of open-vocabulary segmenters.

Strong objectness cues offered by self-supervised features. In Part III, we demonstrate that modern self-supervised features exhibit a significant potential for grouping pixels by object, despite not being explicitly trained for segmentation during the pre-training phase.

Specifically, in Chapter 6, we show that diverse self-supervised features exhibit such an objectness nature, and thus an ensemble of the features can be effectively used to produce a quality mask for a dominant object in an image. We further introduce a network, termed SelfMask, which is trained with the masks produced above as pseudo-labels. SelfMask has been widely adopted as a standard baseline in subsequent studies on unsupervised saliency detection [Ravindran and Basu 2023; Li et al. 2023; Siméoni et al. 2023b; Zhang et al. 2024].

In Chapter 7, we take a step further and illustrate that self-supervised features can facilitate unsupervised instance segmentation using a straightforward prompt-and-merge technique. This approach contrasts with the conventional methods of solving a generalised eigenvalue system with self-supervised features, which typically require complex and intensive computations [Wang et al. 2022b; Wang et al. 2023b].

We posit that these two methodologies will benefit greatly from leveraging

self-supervised features that are explicitly trained for segmentation as a pretext task.

8.2 Future work

In this thesis, we investigate various approaches to developing a robust segmenter, with a primary focus on annotation efficiency. Additionally, we briefly explore potential directions for future research beyond this aspect.

Unification of segmentation and language. To acquire a segmenter capable of functioning robustly in complex real-world scenarios (*e.g.*, objects reflected in a mirror or seen through a transparent medium), integrating common sense knowledge/reasoning ability into a segmentation model using a large language model (LLM) could be an effective solution. Indeed, with the emergence of large multimodal models (LMMs) [Liu et al. 2023; Peng et al. 2024], the idea of equipping a segmentation model with a strong language ability offered by LLM has been explored. This includes from generating a caption along with a predicted segmentation mask [Wang et al. 2023a] to reasoning segmentation [Lai et al. 2023]. However, it is not straightforward how to effectively integrate a segmenter and an LLM into a single model due to inherent discrepancies in their input and output spaces—a segmenter requires visual inputs and outputs dense masks (*i.e.*, pixel-wise classification), whereas an LLM takes in and produces a series of texts. These typically necessitate two distinct types of both encoders and decoders (*i.e.*, image and text encoders, and segmentation and text decoders). While some efforts have been made to unify either the input forms of images and texts by feeding text-rendered images, instead of texts, to a vision-language model [Tschannen et al. 2023], or the output forms of segmentations and texts by predicting segmentations as sets of polygon vertex tokens along with text tokens [Wang et al. 2023a], no works have attempted to streamline the entire pipeline using a single network for both segmentation and text generation. Exploring a streamlined, unified archi-

tecture for both tasks will bring us closer to emulating the human visual system, which operates on complex communications between visual stimuli and high-level reasoning.

Leveraging different low-level modalities. In Part III, we solely rely on RGB values as inputs to group pixels constituting an object. While RGB values and their spatial distributions in a 2D grid convey rich information about the entities in an image, it also comes with noise—after all, what we want for a segmenter to learn is the shapes of entities by learning invariance between an object's pixels with different low-level features such as colour and texture. In this regard, a segmenter can benefit from a processed modality such as optical flow [Choudhury et al. 2022; Xie et al. 2022] and depth, both of which assign values smoothly vary within a single entity, thus can provide a strong objectness signal to a segmenter.

Beyond region-based segmentation. The emergence of SAM [Kirillov et al. 2023] represents a significant milestone in region-based segmentation approaches [He et al. 2017; Cheng et al. 2021b]. However, it also highlights a limitation of such methods: they still require the input and output of high-resolution images (e.g., 1024×1024) to produce precise segmentation masks, which can be a bottleneck for inference speed. As an alternative, exploring polygon-based [Castrejon et al. 2017; Peng et al. 2020; Feng et al. 2024; Zhang et al. 2022] and point-based representations [Yang et al. 2020] for segmentation could help mitigate the limitation of the region-based models.

8.3 Conclusion

In this thesis, we explore various approaches for efficient weakly-supervised and self-supervised learning for image segmentation. We hope these methodologies will pave the way for liberating segmenters from manual supervision and enabling them to surpass their fully-supervised counterparts. This will facilitate their application across diverse fields at a low cost.

References

Michael David Abramoff, Yiyue Lou, Ali Erginay, Warren Clarida, Ryan Amelon, James C Folk, and Meindert Niemeijer. "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning". In: *Investigative Ophthalmology & Visual Science* (2016).

Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. "Frequency-tuned salient region detection". In: *CVPR*. 2009.

Jiwoon Ahn and Suha Kwak. "Learning Pixel-Level Semantic Affinity With Image-Level Supervision for Weakly Supervised Semantic Segmentation". In: *CVPR*. 2018.

Relja Arandjelović and Andrew Zisserman. "Object discovery with a copy-pasting gan". In: *arXiv:1905.11369* (2019).

Pablo Arbeláez, Jordi Pont-Tuset, Jonathan Barron, Ferran Marques, and Jitendra Malik. "Multiscale Combinatorial Grouping". In: *CVPR*. 2014.

Yuki M. Asano, Christian Rupprecht, Andrew Zisserman, and Andrea Vedaldi. "PASS: An ImageNet replacement for self-supervised pretraining without humans". In: *NeurIPS Track on Datasets and Benchmarks* (2021).

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. "Self-labelling via simultaneous clustering and representation learning". In: *ICLR*. 2020.

Les E Atlas, David A Cohn, and Richard E Ladner. "Training connectionist networks with queries and selective sampling". In: *NeurIPS*. 1989.

Sibusiso Reuben Bakana, Yongfei Zhang, and Shan Yang. “Digital Eye on Endangered Wildlife: Crafting Recognition Datasets through Semi-Automated Annotation”. In: *ICAIP*. 2024.

Sayan Banerjee, Avik Hati, Subhasis Chaudhuri, and Rajbabu Velmurugan. “CoSegNet: Image Co-segmentation using a Conditional Siamese Convolutional Network.” In: *IJCAI*. 2019.

Jonathan T. Barron and Ben Poole. “The Fast Bilateral Solver”. In: *ECCV*. 2016.

Dhruv Batra, Adarsh Kowdle, Devi Parikh, Jiebo Luo, and Tsuhan Chen. “icoseg: Interactive co-segmentation with intelligent scribble guidance”. In: *CVPR*. 2010.

Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. “What’s the point: Semantic segmentation with point supervision”. In: *ECCV*. 2016.

William H. Beluch, Tim Genewein, A. Nürnberger, and J. Köhler. “The Power of Ensembles for Active Learning in Image Classification”. In: *CVPR*. 2018.

Adam Bielski and Paolo Favaro. “Emergence of object segmentation in perturbed generative models”. In: *NeurIPS*. 2019.

Patrick Bilic, Patrick Ferdinand Christ, Eugene Vorontsov, Grzegorz Chlebus, Hao Chen, Qi Dou, Chi-Wing Fu, Xiao Han, Pheng-Ann Heng, Jürgen Hesser, et al. “The liver tumor segmentation benchmark (lits)”. In: *arXiv:1901.04056* (2019).

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. “On the opportunities and risks of foundation models”. In: *arXiv:2108.07258* (2021).

Yuri Boykov and Vladimir Kolmogorov. “An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision”. In: *TPAMI* (2004).

Yuri Y Boykov and M-P Jolly. “Interactive graph cuts for optimal boundary & region segmentation of objects in ND images”. In: *ICCV*. 2001.

Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. “Semantic object classes in video: A high-definition ground truth database”. In: *Pattern Recognition Letters* (2009).

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. “Language Models are Few-Shot Learners”. In: *NeurIPS*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. 2020.

Maxime Bucher, Tuan-Hung Vu, Mathieu Cord, and Patrick Pérez. “Zero-Shot Semantic Segmentation”. In: *NeurIPS*. 2019.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. “Coco-stuff: Thing and stuff classes in context”. In: *CVPR*. 2018.

Lile Cai, Xun Xu, Jun Hao Liew, and Chuan Sheng Foo. “Revisiting Superpixels for Active Learning in Semantic Segmentation with Realistic Annotation Costs”. In: *CVPR*. 2021.

Zhaowei Cai and Nuno Vasconcelos. “Cascade R-CNN: Delving Into High Quality Object Detection”. In: *CVPR*. 2018.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. “End-to-End Object Detection with Transformers”. In: *ECCV*. 2020.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. “Deep clustering for unsupervised learning of visual features”. In: *ECCV*. 2018.

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.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. “Emerging Properties in Self-Supervised Vision Transformers”. In: *ICCV*. 2021.

Arantxa Casanova, Pedro O Pinheiro, Negar Rostamzadeh, and Christopher J Pal. “Reinforced active learning for image segmentation”. In: *ICLR*. 2020.

Lluis Castrejon, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler. “Annotating Object Instances with a Polygon-RNN”. In: *CVPR*. 2017.

Adithi D. Chakravarthy, Dilanga Abeyrathna, Mahadevan Subramaniam, Parvathi Chundi, and Venkataramana Gadhamshetty. “Semantic Image Segmentation Using Scant Pixel Annotations”. In: *Machine Learning and Knowledge Extraction* (2022).

Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. “Weakly-supervised semantic segmentation via sub-category exploration”. In: *CVPR*. 2020.

Hong Chen, Yifei Huang, and Hideki Nakayama. “Semantic aware attention based deep object co-segmentation”. In: *ACCV*. 2018.

Liang-Chieh Chen, Y. Zhu, G. Papandreou, Florian Schroff, and H. Adam. “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation”. In: *ECCV*. 2018.

Mickaël Chen, Thierry Artières, and Ludovic Denoyer. “Unsupervised Object Segmentation by Redrawing”. In: *NeurIPS*. 2019.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. “Improved Baselines with Momentum Contrastive Learning”. In: *arXiv:2003.04297* (2020).

Xinlei Chen, Saining Xie, and Kaiming He. “An Empirical Study of Training Self-Supervised Vision Transformers”. In: *ICCV*. 2021.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. “Masked-attention Mask Transformer for Universal Image Segmentation”. In: *CVPR*. 2022.

Bowen Cheng, Omkar Parkhi, and Alexander Kirillov. “Pointly-Supervised Instance Segmentation”. In: *arXiv:2104.06404* (2021).

Bowen Cheng, Alex Schwing, and Alexander Kirillov. “Per-Pixel Classification is Not All You Need for Semantic Segmentation”. In: *NeurIPS*. 2021.

Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. “Global Contrast Based Salient Region Detection”. In: *TPAMI* (2015).

Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. “PiCIE: Unsupervised Semantic Segmentation using Invariance and Equivariance in Clustering”. In: *CVPR*. 2021.

Subhabrata Choudhury, Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. “Guess What Moves: Unsupervised Video and Image Segmentation by Anticipating Motion”. In: *BMVC*. 2022.

Edo Collins, Radhakrishna Achanta, and Sabine Susstrunk. “Deep feature factorization for concept discovery”. In: *ECCV*. 2018.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. “The Cityscapes Dataset for Semantic Urban Scene Understanding”. In: *CVPR*. 2016.

Corinna Cortes and Vladimir Vapnik. “Support-Vector Networks”. In: *Machine Learning* (1995).

Aron Culotta and Andrew McCallum. “Reducing Labeling Effort for Structured Prediction Tasks”. In: *AAAI*. 2005.

Jifeng Dai, Kaiming He, and Jian Sun. “Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation”. In: *ICCV*. 2015.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. “Vision Transformers Need Registers”. In: *arXiv:2309.16588* (2023).

I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raskar. “DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images”. In: *CVPRW*. 2018.

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.

Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. “RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild”. In: *CVPR*. 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *NAACL*. 2019.

Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. “Decoupling Zero-Shot Semantic Segmentation”. In: *CVPR*. 2022.

Yuanyuan Ding, Jing Xiao, and Jingyi Yu. “Importance filtering for image retargeting”. In: *CVPR*. 2011.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. “Unsupervised visual representation learning by context prediction”. In: *ICCV*. 2015.

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.

Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. “Discriminative unsupervised feature learning with exemplar convolutional neural networks”. In: *TPAMI* (2015).

Kyriakos Efstathiou and Marianna Efstathiou. “Celestial Gearbox”. In: *Mechanical Engineering* (2018).

Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. “The pascal visual object classes challenge: A retrospective”. In: *IJCV* (2015).

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. “The Pascal Visual Object Classes (VOC) Challenge”. In: *IJCV* (2010).

Ruochen Fan, Qibin Hou, Ming-Ming Cheng, Gang Yu, Ralph R. Martin, and Shi-Min Hu. “Associating Inter-Image Salient Instances for Weakly Supervised Semantic Segmentation”. In: *ECCV*. 2018.

Hao Feng, Zhou Keyi, Wengang Zhou, Yufei Yin, Jiajun Deng, Qi Sun, and Houqiang Li. “Recurrent Generic Contour-based Instance Segmentation with Progressive Learning”. In: *TCSVT*. 2024.

Robert Fergus, Li Fei-Fei, Pietro Perona, and Andrew Zisserman. “Learning object categories from google’s image search”. In: *ICCV*. 2005.

Geoffrey French, Timo Aila, Samuli Laine, Michal Mackiewicz, and Graham Finlayson. “Consistency regularization and cutmix for semi-supervised semantic segmentation”. In: *BMVC*. 2019.

Alexander Freytag, Erik Rodner, and Joachim Denzler. “Selecting Influential Examples: Active Learning with Expected Model Output Changes”. In: *ECCV*. 2014.

Yarin Gal and Zoubin Ghahramani. “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning”. In: *ICML*. 2016.

Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, Ming-Ming Cheng, Junwei Han, and Philip Torr. “Large-scale Unsupervised Semantic Segmentation”. In: *arXiv:2106.03149* (2021).

Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for autonomous driving? The KITTI vision benchmark suite”. In: *CVPR*. 2012.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. “ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness.” In: *ICLR*. 2019.

Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. “Simple Copy-Paste Is a Strong Data Augmentation Method for Instance Segmentation”. In: *CVPR*. 2021.

Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. “Open-Vocabulary Image Segmentation”. In: *arXiv:2112.12143* (2021).

Stas Goferman, Lihi Zelnik-Manor, and Ayellet Tal. “Context-Aware Saliency Detection”. In: *TPAMI* (2012).

S Alireza Golestaneh and Kris M Kitani. “Importance of Self-Consistency in Active Learning for Semantic Segmentation”. In: *BMVC*. 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. “Generative Adversarial Nets”. In: *NeurIPS*. 2014.

Marc Gorriz, Axel Carlier, Emmanuel Faure, and Xavier Giro-i-Nieto. “Cost-effective active learning for melanoma segmentation”. In: *arXiv:1711.09168* (2017).

Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Mannat Singh, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. “Vision models are more robust and fair when pretrained on uncurated images without supervision”. In: *arXiv:2202.08360* (2022).

Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. "Context-Aware Feature Generation For Zero-Shot Semantic Segmentation". In: *ACM MM*. 2020.

Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, and et al. Jorge Cuadros. "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs". In: *JAMA* (2016).

Agrim Gupta, Piotr Dollár, and Ross Girshick. "LVIS: A Dataset for Large Vocabulary Instance Segmentation". In: *CVPR*. 2019.

Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T. Freeman. "Unsupervised Semantic Segmentation by Distilling Feature Correspondences". In: *ICLR*. 2022.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. "Masked Autoencoders Are Scalable Vision Learners". In: *arXiv:2111.06377* (2021).

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. "Momentum Contrast for Unsupervised Visual Representation Learning". In: *CVPR*. 2020.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask R-CNN". In: *ICCV*. 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition". In: *CVPR*. 2016.

Sheng He, Rina Bao, Jingpeng Li, Jeffrey Stout, Atle Bjornerud, P. Ellen Grant, and Ou Yangming. "Computer-Vision Benchmark Segment-Anything Model (SAM) in Medical Images: Accuracy in 12 Datasets". In: *arXiv:2304.09324* (2023).

Yin-Yin He, Peizhen Zhang, Xiu-Shen Wei, Xiangyu Zhang, and Jian Sun. "Relieving Long-tailed Instance Segmentation via Pairwise Class Balance". In: *arXiv:2201.02784* (2022).

Yutaka Hoshina, Takuma Yamamoto, and Shigeaki Uemura. “Wire-tracking of bent electric cable using X-ray CT and deep active learning”. In: *Microscopy* (2024).

Kuang-Jui Hsu, Yen-Yu Lin, Yung-Yu Chuang, et al. “Co-attention CNNs for unsupervised object co-segmentation.” In: *IJCAI*. 2018.

Ping Hu, Stan Sclaroff, and Kate Saenko. “Uncertainty-aware learning for zero-shot semantic segmentation”. In: *NeurIPS*. 2020.

Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. “Learning to segment every thing”. In: *CVPR*. 2018.

Xinting Hu, Yi Jiang, Kaihua Tang, Jingyuan Chen, Chunyan Miao, and Hanwang Zhang. “Learning to segment the tail”. In: *CVPR*. 2020.

Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang. “Weakly-Supervised Semantic Segmentation Network With Deep Seeded Region Growing”. In: *CVPR*. 2018.

Dat Huynh, Jason Kuen, Zhe Lin, Jiuxiang Gu, and Ehsan Elhamifar. “Open-Vocabulary Instance Segmentation via Robust Cross-Modal Pseudo-Labeling”. In: *arXiv:2111.12698* (2021).

Jyh-Jing Hwang, Stella X Yu, Jianbo Shi, Maxwell D Collins, Tien-Ju Yang, Xiao Zhang, and Liang-Chieh Chen. “Segsort: Segmentation by discriminative sorting of segments”. In: *ICCV*. 2019.

Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. “Learning visual groups from co-occurrences in space and time”. In: *arXiv:1511.06811* (2015).

L. Itti, C. Koch, and E. Niebur. “A model of saliency-based visual attention for rapid scene analysis”. In: *TPAMI* (1998).

Suyog Dutt Jain and Kristen Grauman. “Active image segmentation propagation”. In: *CVPR*. 2016.

Xu Ji, João F Henriques, and Andrea Vedaldi. “Invariant information clustering for unsupervised image classification and segmentation”. In: *ICCV*. 2019.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. “Scaling up visual and vision-language representation learning with noisy text supervision”. In: *ICML*. 2021.

Bin Jin, Maria V Ortiz Segovia, and Sabine Susstrunk. “Webly supervised semantic segmentation”. In: *CVPR*. 2017.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. “Billion-scale similarity search with gpus”. In: *IEEE Transactions on Big Data* (2019).

Armand Joulin, Francis Bach, and Jean Ponce. “Discriminative clustering for image co-segmentation”. In: *CVPR*. 2010.

Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. “Learning to predict where humans look”. In: *ICCV*. 2009.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. “MDETR-modulated detection for end-to-end multi-modal understanding”. In: *ICCV*. 2021.

Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. “Diffusion Models for Open-Vocabulary Segmentation”. In: *ECCV*. 2024.

Naoki Kato, Toshihiko Yamasaki, and Kiyoharu Aizawa. “Zero-shot semantic segmentation via variational mapping”. In: *ICCVW*. 2019.

Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. “Segment Anything in High Quality”. In: *NeurIPS*. 2023.

Tsung-Wei Ke, Jyh-Jing Hwang, Yunhui Guo, Xudong Wang, and Stella X. Yu. “Unsupervised Hierarchical Semantic Segmentation with Multiview Cosegmentation and Clustering Transformers”. In: *CVPR*. 2022.

Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. “Simple does it: Weakly supervised instance and semantic segmentation”. In: *CVPR*. 2017.

Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *ICLR*. 2014.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. “Segment Anything”. In: *ICCV*. 2023.

Dominik A. Klein and Simone Frintrop. “Center-surround divergence of feature statistics for salient object detection”. In: *ICCV*. 2011.

Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. “Introducing geometry in active learning for image segmentation”. In: *ICCV*. 2015.

Philipp Krähenbühl and Vladlen Koltun. “Efficient inference in fully connected CRFs with Gaussian edge potentials”. In: *NeurIPS*. 2011.

Philipp Krähenbühl and Vladlen Koltun. “Efficient inference in fully connected CRFs with Gaussian edge potentials”. In: *NeurIPS*. 2011.

Harold W. Kuhn. “The Hungarian Method for the Assignment Problem”. In: *Naval Research Logistics Quarterly* (1955).

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. “LISA: Reasoning Segmentation via Large Language Model”. In: *arXiv:2308.00692* (2023).

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. “Colorization as a proxy task for visual understanding”. In: *CVPR*. 2017.

Dong-Hyun Lee. “Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks”. In: *ICMLW*. 2013.

Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. “FickleNet: Weakly and Semi-Supervised Semantic Image Segmentation Using Stochastic Inference”. In: *CVPR*. 2019.

D. Lewis and W. A. Gale. “A sequential algorithm for training text classifiers”. In: *SIGIR*. 1994.

David D Lewis and Jason Catlett. “Heterogeneous uncertainty sampling for supervised learning”. In: *ICML*. 1994.

Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. “Language-driven Semantic Segmentation”. In: *ICLR*. 2022.

Guanbin Li and Yizhou Yu. “Visual Saliency Based on Multiscale Deep Features”. In: *CVPR*. 2015.

Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. “Tell Me Where to Look: Guided Attention Inference Network”. In: *CVPR*. 2018.

Nianyi Li, Bilin Sun, and Jingyi Yu. “A weighted sparse coding framework for saliency detection”. In: *CVPR*. 2015.

Peike Li, Yunchao Wei, and Yi Yang. “Consistent structural relation learning for zero-shot segmentation”. In: *NeurIPS*. 2020.

Weihao Li, Omid Hosseini Jafari, and Carsten Rother. “Deep Object Co-segmentation”. In: *ACCV*. 2019.

Xiang Li, Chung-Ching Lin, Yinpeng Chen, Zicheng Liu, Jinglu Wang, Rita Singh, and Bhiksha Raj. “PaintSeg: Painting Pixels for Training-free Segmentation”. In: *NeurIPS*. 2023.

Qiaokang Liang, Hai Qin, Hui Zeng, Jianyong Long, Wei Sun, Dan Zhang, and Yaonan Wang. “Active Learning Integrated Portable Skin Lesion Detection System Based on Multimodel Fusion”. In: *IEEE Sensors Journal* (2023).

Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. “Scribblesup: Scribble-supervised convolutional networks for semantic segmentation”. In: *CVPR*. 2016.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. “Focal Loss for Dense Object Detection”. In: *ICCV*. 2017.

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.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. “Visual Instruction Tuning”. In: *NeurIPS*. 2023.

Tie Liu, Jian Sun, Nan-Ning Zheng, Xiaoou Tang, and Heung-Yeung Shum. “Learning to Detect A Salient Object”. In: *CVPR*. 2007.

Wei Liu, Andrew Rabinovich, and Alexander C. Berg. “ParseNet: Looking Wider to See Better”. In: *arXiv:1506.04579* (2015).

Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu. “Crnet: Cross-reference networks for few-shot segmentation”. In: *CVPR*. 2020.

Stuart Lloyd. “Least squares quantization in PCM”. In: *IEEE transactions on information theory* (1982).

Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *CVPR*. 2015.

Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *ICLR*. 2019.

Timo Lüddecke and Alexander Ecker. “Image Segmentation Using Text and Image Prompts”. In: *CVPR*. 2022.

Radek Mackowiak, Philip Lenz, Omair Ghori, Ferran Diego, Oliver Lange, and Carsten Rother. “Cereals-cost-effective region-based active learning for semantic segmentation”. In: *BMVC*. 2018.

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: *ECCV*. 2018.

Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. “Cross pixel optical-flow similarity for self-supervised learning”. In: *ACCV*. 2018.

Kevis-Kokitsi Maninis, Sergi Caelles, Jordi Pont-Tuset, and Luc Van Gool. “Deep extreme cut: From extreme points to object segmentation”. In: *CVPR*. 2018.

Luca Marchesotti, Claudio Cifarelli, and Gabriela Csurka. “A framework for visual saliency detection with applications to image thumbnailing”. In: *ICCV*. 2009.

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. “Deep Spectral Methods: A Surprisingly Strong Baseline for Unsupervised Semantic Segmentation and Localization”. In: *arXiv:2205.07839* (2022).

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. “Finding an Unsupervised Image Segmenter in each of your Deep Generative Models”. In: *ICLR*. 2022.

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. “Finding an Unsupervised Image Segmenter in each of your Deep Generative Models”. In: *ICLR*. 2022.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. “Efficient estimation of word representations in vector space”. In: *arXiv:1301.3781* (2013).

George A Miller. “WordNet: a lexical database for English”. In: *Communications of the ACM* (1995).

F. Milletari, N. Navab, and S. Ahmadi. “V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation”. In: *3DV*. 2016.

Agata Mosinska-Domanska, Raphael Sznitman, Przemyslaw Glowacki, and Pascal Fua. “Active learning for delineation of curvilinear structures”. In: *CVPR*. 2016.

Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. “The Role of Context for Object Detection and Semantic Segmentation in the Wild”. In: *CVPR*. 2014.

Vida Movahedi and James H. Elder. “Design and perceptual validation of performance measures for salient object segmentation”. In: *CVPRW*. 2010.

Lopamudra Mukherjee, Vikas Singh, and Charles R Dyer. “Half-integrality based algorithms for cosegmentation of images”. In: *CVPR*. 2009.

Naveen Shankar Nagaraja, Frank R. Schmidt, and Thomas Brox. “Video Segmentation with Just a Few Strokes”. In: *ICCV*. 2015.

Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. “Intriguing properties of vision transformers”. In: *NeurIPS*. 2021.

Tam Nguyen, Maximilian Dax, Chaithanya Kumar Mummadi, Nhung Ngo, Thi Hoai Phuong Nguyen, Zhongyu Lou, and Thomas Brox. “DeepUSPS: Deep Robust Unsupervised Saliency Prediction via Self-supervision”. In: *NeurIPS*. 2019.

Mehdi Noroozi and Paolo Favaro. “Unsupervised learning of visual representations by solving jigsaw puzzles”. In: *ECCV*. 2016.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. “DINOv2: Learning Robust Visual Features without Supervision”. In: *arXiv:2304.07193* (2023).

Yassine Ouali, Celine Hudelot, and Myriam Tami. “Semi-Supervised Semantic Segmentation With Cross-Consistency Training”. In: *CVPR*. 2020.

Yassine Ouali, Céline Hudelot, and Myriam Tami. “Autoregressive Unsupervised Image Segmentation”. In: *ECCV*. 2020.

Fabio Panella, Victor Melatti, and Jan Boehm. *FireNet dataset*. http://www.firenet.xyz. Accessed: 2022-05-17.

Dim P Papadopoulos, Alasdair DF Clarke, Frank Keller, and Vittorio Ferrari. “Training object class detectors from eye tracking data”. In: *ECCV*. 2014.

Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. "Extreme clicking for efficient object annotation". In: *ICCV*. 2017.

Giuseppe Pastore, Fabio Cermelli, Yongqin Xian, Massimiliano Mancini, Zeynep Akata, and Barbara Caputo. "A closer look at self-training for zero-label semantic segmentation". In: *CVPR*. 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *NeurIPS*. 2019.

Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell. "Constrained convolutional neural networks for weakly supervised segmentation". In: *ICCV*. 2015.

S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, and X. Zhou. "Deep Snake for Real-Time Instance Segmentation". In: *CVPR*. 2020.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. "Grounding Multimodal Large Language Models to the World". In: *ICLR*. 2024.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global vectors for word representation". In: *EMNLP*. 2014.

Pedro O Pinheiro and Ronan Collobert. "From image-level to pixel-level labeling with convolutional networks". In: *CVPR*. 2015.

Rui Qian, Yunchao Wei, Honghui Shi, Jiachen Li, Jiaying Liu, and Thomas Huang. "Weakly supervised scene parsing with point-based distance metric learning". In: *AAAI*. 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark,

Gretchen Krueger, and Ilya Sutskever. “Learning Transferable Visual Models From Natural Language Supervision”. In: *ICML*. 2021.

Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. “DenseCLIP: Language-Guided Dense Prediction with Context-Aware Prompting”. In: *arXiv:2112.01518* (2021).

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. “SAM 2: Segment Anything in Images and Videos”. In: *arXiv:2408.00714* (2024).

Sriram Ravindran and Debraj Basu. “SEMPART: Self-supervised Multi-resolution Partitioning of Image Semantics”. In: *ICCV*. 2023.

Pengzhen Ren, Y. Xiao, Xiao-jun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and X. Wang. “A Survey of Deep Active Learning”. In: *arXiv:2009.00236* (2020).

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *MICCAI*. 2015.

Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. “GrabCut interactive foreground extraction using iterated graph cuts”. In: *ACM transactions on graphics* (2004).

Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov. “Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs”. In: *CVPR*. 2006.

Michael Rubinstein, Ariel Shamir, and Shai Avidan. “Improved Seam Carving for Video Retargeting”. In: *SIGGRAPH* (2008).

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. “Imagenet large scale visual recognition challenge”. In: *IJCV* (2015).

T. Scheffer, Christian Decomain, and S. Wrobel. “Active Hidden Markov Models for Information Extraction”. In: *IDA*. 2001.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Theo Coombes, Cade Gordon, Aarush Katta, Robert Kaczmarczyk, and Jenia Jitsev. *LAION-5B: LAION-5B: A NEW ERA OF OPEN LARGE-SCALE MULTI-MODAL DATASETS*. https://laion.ai/laion-5b-a-new-era-of-open-large-scale-multi-modal-datasets/. 2022.

H Scudder. “Probability of error of some adaptive pattern-recognition machines”. In: *IEEE Transactions on Information Theory* (1965).

Ozan Sener and Silvio Savarese. “Active Learning for Convolutional Neural Networks: A Core-Set Approach”. In: *ICLR*. 2018.

Burr Settles. *Active learning literature survey*. 2009.

H. S. Seung, M. Opper, and H. Sompolinsky. “Query by committee”. In: *COLT*. 1992.

C. Shannon. “A mathematical theory of communication”. In: *Bell Syst. Tech. J.* (1948).

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. “Objects365: A Large-Scale, High-Quality Dataset for Object Detection”. In: *ICCV*. 2019.

Tong Shen, Guosheng Lin, Chunhua Shen, and Ian Reid. “Bootstrapping the performance of webly supervised semantic segmentation”. In: *CVPR*. 2018.

Jianbo Shi and Jitendra Malik. “Normalized cuts and image segmentation”. In: *TPAMI* (2000).

Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. “Hierarchical Image Saliency Detection on Extended CSSD”. In: *TPAMI* (2016).

George Shih, Carol C Wu, Safwan S Halabi, Marc D Kohli, Luciano M Prevedello, Tessa S Cook, Arjun Sharma, Judith K Amorosa, Veronica Arteaga, and

et al. Maya Galperin-Aizenberg. “Augmenting the National Institutes of Health chest radiograph dataset with expert annotations of possible pneumonia”. In: *Radiology: Artificial Intelligence* (2019).

Gyungin Shin, Samuel Albanie, and Weidi Xie. “Unsupervised Salient Object Detection With Spectral Cluster Voting”. In: *CVPRW*. 2022.

Gyungin Shin, Weidi Xie, and Samuel Albanie. “All you need are a few pixels: semantic segmentation with PixelPick”. In: *ICCVW*. 2021.

Gyungin Shin, Weidi Xie, and Samuel Albanie. “ReCo: Retrieve and Co-segment for Zero-shot Transfer”. In: *NeurIPS*. 2022.

Gyungin Shin, Weidi Xie, and Samuel Albanie. “NamedMask: Distilling Segmenters from Complementary Foundation Models”. In: *CVPRW*. 2023.

Oriane Siméoni, Gilles Puy, Huy V. Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. “Localizing Objects with Self-Supervised Transformers and no Labels”. In: *BMVC*. 2021.

Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonín Vobecký, Éloi Zablocki, and Patrick Pérez. “Unsupervised Object Localization: Observing the Background To Discover Objects”. In: *CVPR*. 2023.

Oriane Siméoni, Éloi Zablocki, Spyros Gidaris, Gilles Puy, and Patrick Pérez. “Unsupervised Object Localization in the Era of Self-Supervised ViTs: A Survey”. In: *arXiv:2310.12904* (2023).

Oriane Siméoni, Éloi Zablocki, Spyros Gidaris, Gilles Puy, and Patrick Pérez. “Unsupervised Object Localization in the Era of Self-Supervised ViTs: A Survey”. In: *IJCV* (2024).

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. “Variational Adversarial Active Learning”. In: *ICCV*. 2019.

Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. “Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation”. In: *CVPR*. 2019.

Bongwon Suh, Haibin Ling, Benjamin B. Bederson, and David W. Jacobs. “Automatic Thumbnail Cropping and Its Effectiveness”. In: *ACM Symposium on User Interface Software and Technology*. 2003.

Jin Sun and Haibin Ling. “Scale and object aware image retargeting for thumbnail browsing”. In: *ICCV*. 2011.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. “Scalability in Perception for Autonomous Driving: Waymo Open Dataset”. In: *CVPR*. 2020.

Chufeng Tang, Lingxi Xie, Gang Zhang, Xiaopeng Zhang, Qi Tian, and Xiaolin Hu. “Active Pointly-Supervised Instance Segmentation”. In: *ECCV*. 2022.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. “What Makes for Good Views for Contrastive Learning?” In: *NeurIPS*. 2020.

Catalina Tobon-Gomez, Arjan J. Geers, Jochen Peters, Jürgen Weese, Karen Pinto, Rashed Karim, Mohammed Ammar, Abdelaziz Daoudi, Jan Margeta, Zulma Sandoval, Birgit Stender, Yefeng Zheng, Maria A. Zuluaga, Julian Betancur, Nicholas Ayache, Mohammed Amine Chikh, Jean-Louis Dillenseger, B. Michael Kelm, Saïd Mahmoudi, Sébastien Ourselin, Alexander Schlaefer, Tobias Schaeffter, Reza Razavi, and Kawal S. Rhode. “Benchmark for Algorithms Segmenting the Left Atrium From 3D CT and MRI Datasets”. In: *IEEE Transactions on Medical Imaging* (2015).

Hiroki Tokunaga, Yuki Teramoto, Akihiko Yoshizawa, and Ryoma Bise. “Adaptive weighting multi-field-of-view CNN for semantic segmentation in pathology”. In: *CVPR*. 2019.

Xin-Yi Tong, Gui-Song Xia, Qikai Lu, Huanfeng Shen, Shengyang Li, Shucheng You, and Liangpei Zhang. “Land-cover classification with high-resolution remote sensing images using transferable deep models”. In: *Remote Sensing of Environment* (2020).

Michael Tschannen, Basil Mustafa, and Neil Houlsby. “CLIPPO: Image-and-Language Understanding from Pixels Only”. In: *CVPR*. 2023.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. “Deep image prior”. In: *CVPR*. 2018.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. “Unsupervised Semantic Segmentation by Contrasting Object Mask Proposals”. In: *ICCV*. 2021.

Wouter Van Gansbeke, Simon Vandenhende, and Luc Van Gool. “Discovering Object Masks with Transformers for Unsupervised Semantic Segmentation”. In: *arXiv:2206.06363* (2022).

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.

Alexander Vezhnevets, Joachim M Buhmann, and Vittorio Ferrari. “Active learning for semantic segmentation with expected change”. In: *CVPR*. 2012.

Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. “Cosegmentation revisited: Models and optimization”. In: *ECCV*. 2010.

Sara Vicente, Carsten Rother, and Vladimir Kolmogorov. “Object cosegmentation”. In: *CVPR*. 2011.

Huy V. Vo, Patrick Pérez, and Jean Ponce. “Toward Unsupervised, Multi-object Discovery in Large-Scale Image Collections”. In: *ECCV*. 2020.

Antonin Vobecky, David Hurych, Oriane Siméoni, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, and Josef Sivic. “Drive&Segment: Unsupervised Semantic Segmentation of Urban Scenes via Cross-modal Distillation”. In: *arXiv:2203.11160* (2022).

Andrey Voynov and Artem Babenko. “Unsupervised discovery of interpretable directions in the gan latent space”. In: *ICML*. 2020.

Andrey Voynov, Stanislav Morozov, and Artem Babenko. “Object Segmentation Without Labels with Large-Scale Generative Models”. In: *ICML*. 2021.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. *The Caltech-UCSD Birds-200-2011 Dataset*. Tech. rep. 2011.

Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. “Learning to detect salient objects with image-level supervision”. In: *CVPR*. 2017.

Linda Wang, Zhong Qiu Lin, and Alexander Wong. “COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images”. In: *Scientific Reports* (2020).

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, and Jifeng Dai. “VisionLLM: Large Language Model is also an Open-Ended Decoder for Vision-Centric Tasks”. In: *NeurIPS*. 2023.

Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M. Alvarez. “FreeSOLO: Learning To Segment Objects Without Annotations”. In: *CVPR*. 2022.

Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. “Dense Contrastive Learning for Self-Supervised Visual Pre-Training”. In: *CVPR*. 2021.

Xudong Wang, Rohit Girdhar, Stella X Yu, and Ishan Misra. “Cut and learn for unsupervised object detection and instance segmentation”. In: *CVPR*. 2023.

Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz. “Self-supervised Transformers for Unsupervised Object Discovery using Normalized Cut”. In: *CVPR*. 2022.

Mark Weber, Jun Xie, Maxwell Collins, Yukun Zhu, Paul Voigtlaender, Hartwig Adam, Bradley Green, Andreas Geiger, Bastian Leibe, Daniel Cremers, Aljosa Osep, Laura Leal-Taixe, and Liang-Chieh Chen. “STEP: Segmenting and Tracking Every Pixel”. In: *NeurIPS Track on Datasets and Benchmarks*. 2021.

Yichen Wei, Fang Wen, Wangjiang Zhu, and Jian Sun. “Geodesic Saliency Using Background Priors”. In: *ECCV*. 2012.

Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S. Huang. “Revisiting Dilated Convolution: A Simple Approach for Weakly- and Semi-Supervised Semantic Segmentation”. In: *CVPR*. 2018.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. *Detectron2*. https://github.com/facebookresearch/detectron2. 2019.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. “Unsupervised feature learning via non-parametric instance discrimination”. In: *CVPR*. 2018.

Michael Wurm, T. Stark, X. X. Zhu, M. Weigand, and H. Taubenböck. “Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* (2019).

Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. “Semantic projection network for zero-and few-label semantic segmentation”. In: *CVPR*. 2019.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. “Sun database: Large-scale scene recognition from abbey to zoo”. In: *CVPR*. 2010.

Junyu Xie, Weidi Xie, and Andrew Zisserman. “Segmenting Moving Objects via an Object-Centric Layered Representation”. In: *NeurIPS*. 2022.

Shuai Xie, Zunlei Feng, Y. Chen, Songtao Sun, Chao Ma, and Ming-Li Song. “DEAL: Difficulty-aware Active Learning for Semantic Segmentation”. In: *ACCV*. 2020.

Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. “GroupViT: Semantic Segmentation Emerges from Text Supervision”. In: *CVPR*. 2022.

Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. “A Simple Baseline for Zero-shot Semantic Segmentation with Pre-trained Vision-language Model”. In: *arXiv:2112.14757* (2021).

Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. “Hierarchical Saliency Detection”. In: *CVPR*. 2013.

Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. “Saliency Detection via Graph-Based Manifold Ranking”. In: *CVPR*. 2013.

Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. “Suggestive annotation: A deep active learning framework for biomedical image segmentation”. In: *MICCAI*. 2017.

Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. “Denseaspp for semantic segmentation in street scenes”. In: *CVPR*. 2018.

Ze Yang, Yinghao Xu, Han Xue, Zheng Zhang, Raquel Urtasun, Liwei Wang, Stephen Lin, and Han Hu. “Dense RepPoints: Representing Visual Objects with Dense Point Sets”. In: *ECCV*. 2020.

David Yarowsky. “Unsupervised word sense disambiguation rivaling supervised methods”. In: *ACL*. 1995.

Donggeun Yoo and I. Kweon. “Learning Loss for Active Learning”. In: *CVPR* (2019).

Fisher Yu and Vladlen Koltun. “Multi-Scale Context Aggregation by Dilated Convolutions”. In: *ICLR*. 2016.

Xiaohang Zhan, Ziwei Liu, Ping Luo, Xiaoou Tang, and Chen Loy. “Mix-and-match tuning for self-supervised semantic segmentation”. In: *AAAI*. 2018.

Chi Zhang, Guankai Li, Guosheng Lin, Qingyao Wu, and Rui Yao. “CycleSegNet: Object Co-Segmentation With Cycle Refinement and Region Correspondence”. In: *IEEE TIP* (2021).

Dingwen Zhang, Junwei Han, and Yu Zhang. “Supervision by Fusion: Towards Unsupervised Learning of Deep Salient Object Detector”. In: *ICCV*. 2017.

Jianming Zhang and Stan Sclaroff. “Exploiting Surroundedness for Saliency Detection: A Boolean Map Approach”. In: *TPAMI* (2016).

Jianming Zhang, Stan Sclaroff, Zhe Lin, Xiaohui Shen, Brian Price, and Radomír Mech. “Minimum Barrier Salient Object Detection at 80 FPS”. In: *ICCV*. 2015.

Jing Zhang, Tong Zhang, Yuchao Dai, Mehrtash Harandi, and Richard Hartley. “Deep Unsupervised Saliency Detection: A Multiple Noisy Labeling Perspective”. In: *CVPR*. 2018.

Tao Zhang, Shiqing Wei, and Shunping Ji. “E2EC: An End-to-End Contour-based Method for High-Quality High-Speed Instance Segmentation”. In: *CVPR*. 2022.

Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. “K-Net: Towards Unified Image Segmentation”. In: *NeurIPS*. 2021.

Xiao Zhang and Michael Maire. “Self-supervised visual representation learning from hierarchical grouping”. In: *NeurIPS*. 2020.

Xin Zhang, Jinheng Xie, Yuan Yuan, Michael Bi Mi, and Robby T. Tan. “HEAP: Unsupervised Object Discovery and Localization with Contrastive Grouping”. In: *AAAI*. 2024.

Zhao Zhang, Wenda Jin, Jun Xu, and Ming-Ming Cheng. “Gradient-Induced Co-Saliency Detection”. In: *ECCV*. 2020.

Hang Zhao, Xavier Puig, Bolei Zhou, Sanja Fidler, and Antonio Torralba. “Open vocabulary scene parsing”. In: *ICCV*. 2017.

Ye Zheng, Jiahong Wu, Yongqiang Qin, Faen Zhang, and Li Cui. “Zero-Shot Instance Segmentation”. In: *CVPR*. 2021.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. “Learning deep features for discriminative localization”. In: *CVPR*. 2016.

Chong Zhou, Chen Change Loy, and Bo Dai. “Denseclip: Extract free dense labels from clip”. In: *arXiv:2112.01071* (2021).

Chong Zhou, Chen Change Loy, and Bo Dai. “Extract Free Dense Labels from CLIP”. In: *ECCV*. 2022.

Wangjiang Zhu, Shuang Liang, Yichen Wei, and Jian Sun. “Saliency Optimization from Robust Background Detection”. In: *CVPR*. 2014.

Appendix A

Statement of Authorship

A statement of authorship is provided for each multi-authored paper included in this thesis. The statements describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication, there exists a complete statement that is filled out and signed by the candidate and supervisor.

Statement of Authorship for the paper "All You Need Are a Few Pixels: Semantic Segmentation with PixelPick" in Chapter 2.

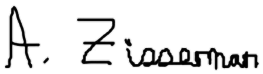
Paper title	[Best Paper Award] All You Need Are a Few Pixels: Semantic Segmentation with PixelPick
Authors	**Gyungin Shin**, Weidi Xie, Samuel Albanie
Publication status	Published
Publication details	The Interactive Labeling and Data Augmentation for Vision Workshop, International Conference on Computer Vision (ICCV), 2021

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Design and implementation of models • Running experiments • Writing the paper	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024

Statement of Authorship for the paper "ReCo: Retrieve and Co-segment for Zero-shot Transfer" in Chapter 3.

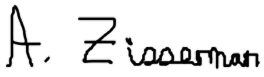
Paper title	ReCo: Retrieve and Co-segment for Zero-shot Transfer
Authors	**Gyungin Shin**, Weidi Xie, Samuel Albanie
Publication status	Published
Publication details	Neural Information Processing Systems (NeurIPS), 2022.

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Implementation and training of models • Inventing the algorithm • Running experiments • Writing the paper	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024

Statement of Authorship for the paper "NamedMask: Distilling Segmenters from Complementary Foundation Models" in Chapter 4.

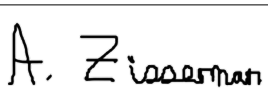
Paper title	NamedMask: Distilling Segmenters from Complementary Foundation Models
Authors	**Gyungin Shin**, Weidi Xie, Samuel Albanie
Publication status	Published
Publication details	The 2nd Workshop on Learning with Limited Labelled Data for Image and Video Understanding, Conference on Computer Vision and Pattern Recognition (CVPR), 2023

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Conception of research ideas • Implementation and training of models • Running experiments • Writing the paper	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024

Statement of Authorship for the paper "Zero-shot Unsupervised Transfer Instance Segmentation" in Chapter 5.

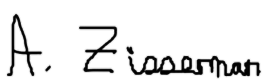
Paper title	Zero-shot Unsupervised Transfer Instance Segmentation
Authors	**Gyungin Shin**, Samuel Albanie, Weidi Xie
Publication status	Published
Publication details	[Best Paper Award & Oral] The Workshop on Learning with Limited Labelled Data for Image and Video Understanding, Conference on Computer Vision and Pattern Recognition (CVPR), 2023

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Conception of research ideas • Implementation and training of models • Running experiments • Writing the paper	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024

Statement of Authorship for the paper "Unsupervised Salient Object Detection with Spectral Cluster Voting" in Chapter 6.

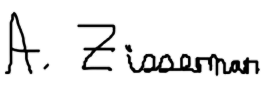
Paper title	Unsupervised Salient Object Detection with Spectral Cluster Voting
Authors	**Gyungin Shin**, Samuel Albanie, Weidi Xie
Publication status	Published
Publication details	The Workshop on Learning with Limited Labelled Data for Image and Video Understanding, Conference on Computer Vision and Pattern Recognition (CVPR), 2022

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Implementation and training of models • Inventing the algorithm • Running experiments • Writing the paper	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024

Statement of Authorship for the paper "ProMerge: Prompt and Merge for Unsupervised Instance Segmentation" in Chapter 7.

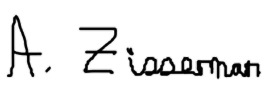
Paper title	ProMerge: Prompt and Merge for Unsupervised Instance Segmentation
Authors	Dylan Li[*], **Gyungin Shin**[*✉] [*]Equal contribution [✉]Corresponding author
Publication status	Published
Publication details	European Conference on Computer Vision (ECCV), 2024

Student Confirmation

Student name	Gyungin Shin	
Contribution to the paper	The candidate's contributions: • Advising the project • Conception of research ideas • Designing and conducting ablation studies • Training models • Writing papers	
Signature and Date		22 August 2024

Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Professor Andrew Zisserman	
Supervisor comments	The description of the contributions is accurate.	
Signature and Date		22 August 2024