David Novotný
New College

Doctor of Philosophy
Michaelmas 2018

Learning the Structure of Object Categories from Incomplete Supervision

Abstract

This thesis aims at learning and predicting the fine-grained structure of visual object categories given input image data. Alleviating the common requirement of collecting an ample amount of manual annotations, we propose several approaches that learn given an incomplete supervisory signal.

Specifically, we begin with an analysis of the amount of supervision needed to learn all visual variations of an object part. Motivated by the gathered observations, a detector of semantic (i.e. nameable) parts supervised with inexpensive web image search data is then proposed. The main challenge of handling a significant amount of annotation noise is addressed with a novel geometry-appearance embedding.

Moving away from semantic part detection, learning generic mid-level elements for understanding the geometry of object categories is brought into focus. A novel architecture that outputs a visual representation suitable for establishing image-to-image semantic correspondences is proposed. The main contribution consists of a new discriminability & diversity objective that facilitates learning of sparse image features sensitive to the changes of the geometry of the input.

A similar feature learning machine leveraging the equivariance constraint is later introduced. Differently from existing alternatives, we adapt the method for the noisy settings of the training dataset by means of a novel probabilistic introspection framework. This allows for a selective representation of image pixels that have the potential to result in a correct match.

Inspired by the ability of deep networks to decompose an object into a constellation of pixel-perfect landmarks, an opposite problem of grouping image pixels belonging to an object is addressed. More specifically, we deal with the instance segmentation problem using a deep convolutional architecture that "colors" image pixels with their instance labels. Identifying the convolutional coloring dilemma, a drawback of standard position-agnostic networks that prevents them from solving this task, we propose a correction comprising a novel position-sensitive semi-convolutional operator.

The last tackled task is learning 3D shapes of object categories. Inspired by the human visual system, a deep network that learns by observing an object category in a sequence of videos is described.

Our final contribution is a probabilistic learning scheme that increases robustness of network training and enables test-time confidence predictions. This is achieved by explicitly modeling the distribution of training errors caused by the insufficiencies of the model or by the noise in ground truth annotations.

Learning the Structure of Object Categories from Incomplete Supervision

D. Phil Thesis

Visual Geometry Group
Department of Engineering Science
University of Oxford

Supervisors:
Professor Andrea Vedaldi
Doctor Diane Larlus

David Novotný

New College
Michaelmas 2018

Declaration

This thesis is submitted to the Department of Engineering Science, 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.

David Novotný, New College

Acknowledgements

For all the time they dedicated, I would like to express the biggest thanks to my supervisors Diane Larlus and Andrea Vedaldi. Diane and Andrea posses a complimentary set of amazing skills that made my DPhil an extremely fun experience during which I learned so much. I thank Diane for being such a nice person and for spending a load of her time discussing new research directions which always helped me to move forward. I thank Andrea for all his time spent saving seemingly lost papers and for all the great ideas for this thesis. I especially appreciated the easy-going friendly research atmosphere where the pre-deadline stress transformed into an exciting process, often causing forgetting about the need to sleep. I am also very grateful to Jiří Matas for giving me the first chance to enter the field of computer vision. These lines would not be written without his involvement.

I want to express my gratitude to all the members of the VGG group in Oxford who made the lab a place full of fun discussions. Special thanks go to Sam Albanie who I really enjoyed collaborating with on several works. These would never get published without his tenacious overnight efforts to squeeze the best out of our ideas.

The year spent in Grenoble would not be so fun without everyone from Naver Labs Europe, to whom I would like to thank as well. There were quite a few great evenings that will stay in my memory for many days to come.

Something I will always remember was all the time spent with the amazing people from the Oxford University Volleyball Club / Oxford NVL club. The very special thanks go to Oxford mNVL/M1 of 2017/2018 and, mainly, to the class of 2015/2016 which was by far the most awesome team I had an opportunity to meet. Captain's incredibly eloquent emails with motivational recordings, defeating Cambridge or pints at the Cape constitute just a meager excerpt from the huge list of unforgettable moments experienced during the gripping season. I feel lucky for being a part of this squad where friends would so remarkably complement each other both on and off the court. I am also grateful for the weekends spent with the beach volleyball community that would always show up for another great game despite the toughest atmospheric conditions the British islands can offer.

Importantly, I am grateful to the people from Oxford kebab vans and Grenoble pizza shops. It is undeniable that none of the works in this thesis would come into existence without the much needed midnight calories that fueled the pre-deadline periods.

I would love to thank my family, especially my parents, for their limitless support. Finally, I would like to express my deepest gratitude to my other half Zuzka for being an amazing partner for many years and for her unparalleled patience with the busy schedule of my DPhil days.

Contents

List of Abbreviations

BA	Bundle adjustment
CNN	Convolutional neural network
CRF	Conditional random field
D&D	Discriminability & Diversity
DA	Domain adaptation
DPM	Deformable Part Model
DSP	Deformable spatial pyramid
ELDA	Exemplar Linear discriminant analysis
HoG	Histogram of Oriented Gradients
IC	Instance Coloring
IoU	Intersection-over-union
LoG	Laplacian of Gaussian
LSVM	Latent SVM
MIL	Multiple instance learning
MMD	Maximum mean discrepancy
MRF	Markov random field
NEIL	Never Ending Image Learner
NRSfM	Non-rigid SfM
P&V	Propose & Verify
PF	Proposal Flow
RANSAC	Random Sample Consensus
RASL	Robust Alignment by Sparse Low-rank
SfM	Structure from Motion
SGD	Stochastic gradient descend
SIFT	Scale Invariant Feature Transform
SURF	Sped Up Robust Features
SVM	Support Vector Machine
TPS	Thin plate spline
WSOD	Weakly supervised object detection
WSOL	Weakly supervised object localization

1 Introduction

1.1 Objective

Modern deep learning methods addressing the main perceptual computer vision tasks such as image classification [Krizhevsky et al. 2012a; Simonyan and Zisserman 2015; He et al. 2016a], object detection [Girshick et al. 2014a; Girshick 2015; Ren et al. 2016] or object instance segmentation [He et al. 2017] have achieved great results and in some cases have even surpassed human performance [Lu and Tang 2015]. However, such perceptual tasks are rather crude and oblivious to the fine-grained structure of individual object categories. In our work, we go beyond object classification and detection and achieve an in-depth understanding of the internal structure, meaning and geometry of objects.

The fine-grained structure of an object category can be represented in many forms. Object categories can be parsed into compositions of *parts*. Recent efforts in the vision community to fuel deep machines with increasing amount of annotated data brought manually labeled datasets [Chen et al. 2014b] that allowed for extraction of semantic (nameable) object parts [Hariharan et al. 2015; Wang et al. 2015b; Zhang et al. 2014b]. However, constraining parts to represent only the human-nameable concepts introduces a "language bottleneck" [Efros 2015] which prevents covering all the possible variations of the visual world.

Dropping the constraint of semanticity enables detecting the *non-semantic parts* which are repeatable mid-level visual elements useful for e.g. boosting the performance of object detectors [Felzenszwalb et al. 2008a; Fergus 2005]. *Keypoints* are related to non-semantic object parts and denote a location of a mid-level visual element with the maximum possible precision by identifying a single pixel in an image. While semantic parts explain the meaning of object categories, the non-semantic ones typically bring understanding of object *geometry*. In fact, keypoints allow to establish precise image-to-image correspondences across intra-category appearance variations allowing for multi-view category-level 3D reconstruction [Kar et al. 2015a; Kanazawa et al. 2018], giving rise to an explicit representation of the category shape.

However, one does not always have an access to multi-view data and must instead only rely on single object views. Given a single image, the 3D shape of the visible part can be described with a *depth map* denoting the distance of every scene surface point from the camera plane. By utilizing hand-designed 3D CAD models [Chang et al. 2015], a depth predictor can be extended to recover the full 3D shape of a depicted object [Choy et al. 2016b; Groueix et al. 2018] in form of a point cloud or a voxel grid. Extracting the 3D structure from a "flat" 2D image is inherently ill-posed and hence benefits tremendously from latent variables that can constrain the space of shape predictions. One of such cues is an object *pose* which, combined with a pose-invariant representation of the global object style, fully specifies the complete 3D shape of the perceived object.

Understanding the fine-grained structure of object categories through parts, keypoints, pose and partial or full 3D shape thus has many practical applications that motivate developing predictors of such representations. From a practical standpoint, a self-evident choice for implementing the fine-grained predictors are deep networks, the crux of modern computer vision algorithms that have been tremendously successful on classic perceptual tasks. However, while modern deep image classifiers and object detectors work well mainly due the availability of large and inexpensive manually annotated datasets [Deng

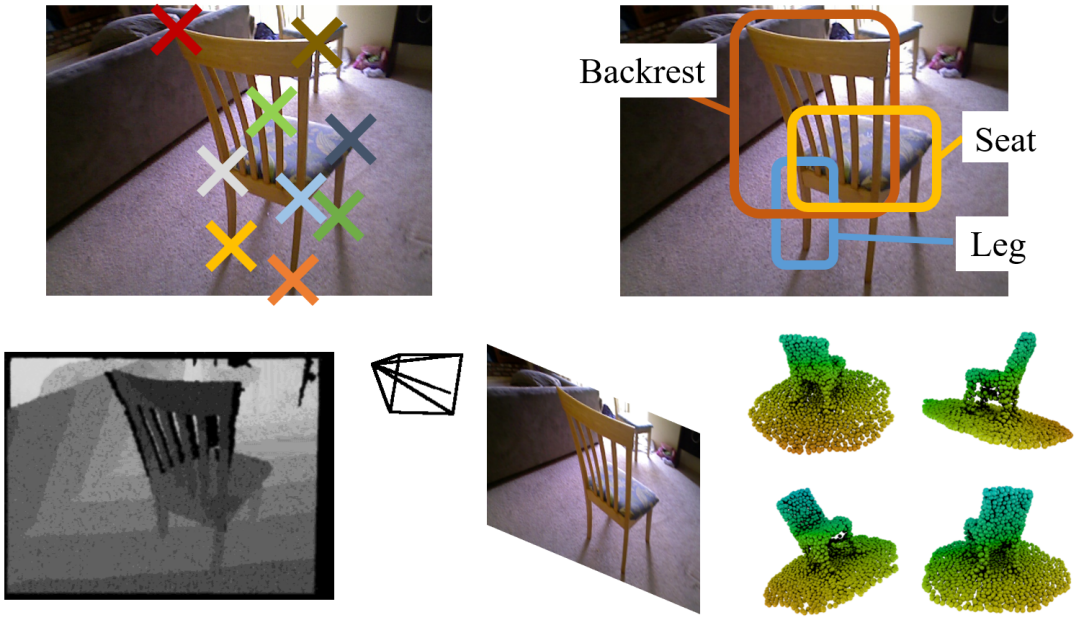


Figure 1.1: Example representations of the fine-grained object structure. From left to right, top row: Keypoints, semantic parts; bottom row: Depth, viewpoint, 3D shape. Each representation type is output by methods proposed in this thesis.

et al. 2009; Lin et al. 2014], part and 3D shape predictors are far from enjoying this benefit. For instance, the number of annotated poses in the largest category-centric 3D dataset [Xiang et al. 2014] is roughly two orders of magnitude lower than the number of class annotations in the popular ImageNet image classification dataset [Deng et al. 2009].

In fact, the significant cost of manually collecting part and 3D annotations constitutes the biggest roadblock on the way to large-scale understanding of fine-grained structure of object categories. Combined with the fact that for some tasks, such as depth prediction, acquiring dense and accurate human annotations is nearly impossible, less supervised alternatives have to be considered. Hence, in this thesis we restrict ourselves to approaches that learn the structure of objects, inexpensively, from incomplete supervision.

1.2 Thesis outline

A brief overview of the thesis content is presented in this section. We start by conducting an initial analysis of the amount of supervision needed to learn the detailed structure (parts) of object categories. Motivated by the observations about generalization capabilities of deep nets, we then tackle the task of learning semantic object parts using web images as a source of inexpensive but noisy supervision. The ensuing sections shift from analyzing semantic parts towards learning geometry-aware features with a set of diversified discriminative feature detectors or through a novel robust probabilistic matching framework. It is later demonstrated that the insights gathered during learning such features could be exploited to boost the performance of instance segmentation algorithms. The final contribution is a novel deep framework for predicting the 3D structure of object categories which is cued solely by observing objects from a moving vantage point.

1.2.1 Seeking the limits of supervision

As outlined above, the main goal of this thesis is to "learn more with less" using deep networks. While such goal seems well motivated, it is meaningful to first analyze the actual amount of supervisory data needed in order to obtain competitive performance within the classical fully supervised setup. Observing that deep nets posses significant generalization capabilities given a small number of annotated samples can then motivate lighter, less supervised approaches.

While it is possible to search for a limit of supervision in the context of many different machine learning tasks, in chapter 3 we conduct such analysis for object part detectors. This not only follows the main theme of the thesis, but is also strongly motivated by the fact that parts are a good candidate for patterns that are learnable given a limited supervision. In fact, parts are often defined as visuals elements that are shareable across instances of the same if not different object categories. It is not obvious, though, whether each part occurrence can

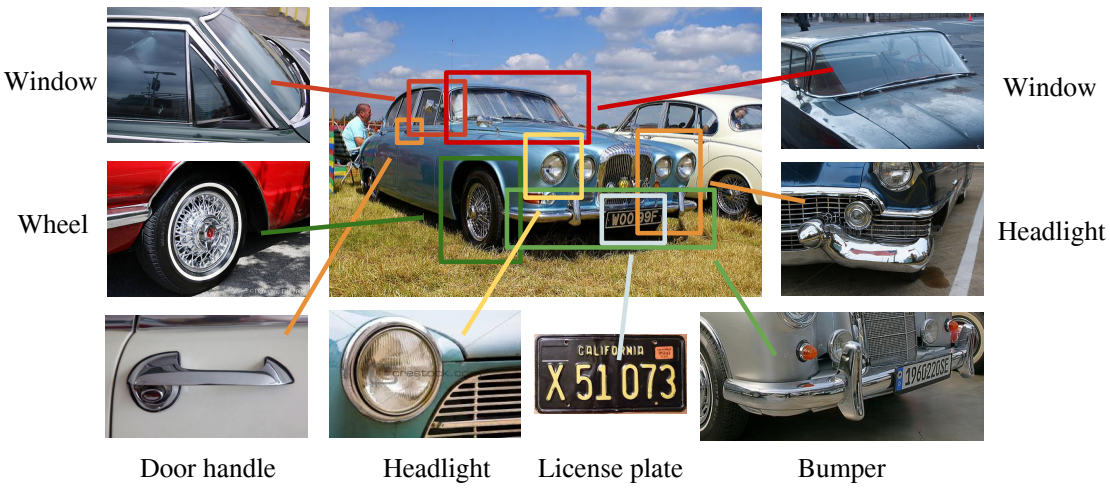


Figure 1.2: Parsing an object into its semantic parts. Chapter 4 addresses learning detectors of human-nameable parts using web supervision.

be detected based on complex semantic cues (e.g. a "door handle" has always the same function despite its shape) or whether the part is mostly defined by its appearance (e.g. the discriminative feature of different animal "eyes" could be the iris which has a fairly similar appearance in all cases). The latter would conveniently entail that object parts can be learned in an unsupervised fashion simply by observing images of instances of object categories. By supervising a deep part predictor with image-based appearance annotations, which are free of the semantic information, we asses the existence of this desired appearance-driven part shareability.

Chapter 3 provides such analysis by studying the limit of supervision on the task of detecting "foot" and "eye" parts of animals. For this purpose, we contributed with a novel dataset because existing alternatives were found unsuitable. Furthermore, it also proposes a novel active-transfer learning method that is designed to boost the generalization capabilities of a baseline deep part detector.

1.2.2 Webly supervised learning of object parts

Motivated by the results from chapter 3, in chapter 4 we focus on learning the semantic object parts from incomplete supervision. More specifically, the goal

is to detect occurrences of different object parts inside images of their parent object categories (fig. 1.2). While fully supervised approaches exist [Chen et al. 2014b; Chai et al. 2013; Wang et al. 2015a], the cost of collecting the part annotations is preventing their deployment at a larger-scale. In fact, to the best of our knowledge, only a single large-scale dataset of generic part annotations exists to this date [Chen et al. 2014b].

Although large clean datasets are hard to collect, we can consider noisy but inexpensive alternatives. To this end, chapter 4 uses images retrieved by web image-search engines in order to supervise the part detection task. Unfortunately, the noisy nature of the retrieved image data in combination with the large appearance variations of the semantic object parts makes the separation of the part occurrences from an irrelevant background visual signal extremely challenging.

The main insight of our approach from chapter 4 is that, while it is hard to detect the semantic nameable parts, it might be much easier to drop the constraint of nameability and first mine for non-semantic mid-level visual elements that can be detected repeatably and reliably across the set of noisy web images. These parts, termed *anchors*, constitute a cornerstone of a novel geometry-appearance embedding that allows for more robust detection of the semantic parts.

1.2.3 Weakly-supervised learning of geometry-sensitive features

Learning semantic part detectors is important for grounding occurrences of human-defined concepts in images. Unfortunately, the limited number of the nameable visual element categories significantly constrains the amount of information that can be extracted from our visual surroundings. Instead, we can focus on detecting more generic visual elements that cannot be easily described with language.

A prominent example of such elements are category-specific *keypoints* (e.g. human skeleton points [Ramanan 2006] or rigid object points [Xiang et al. 2014]).

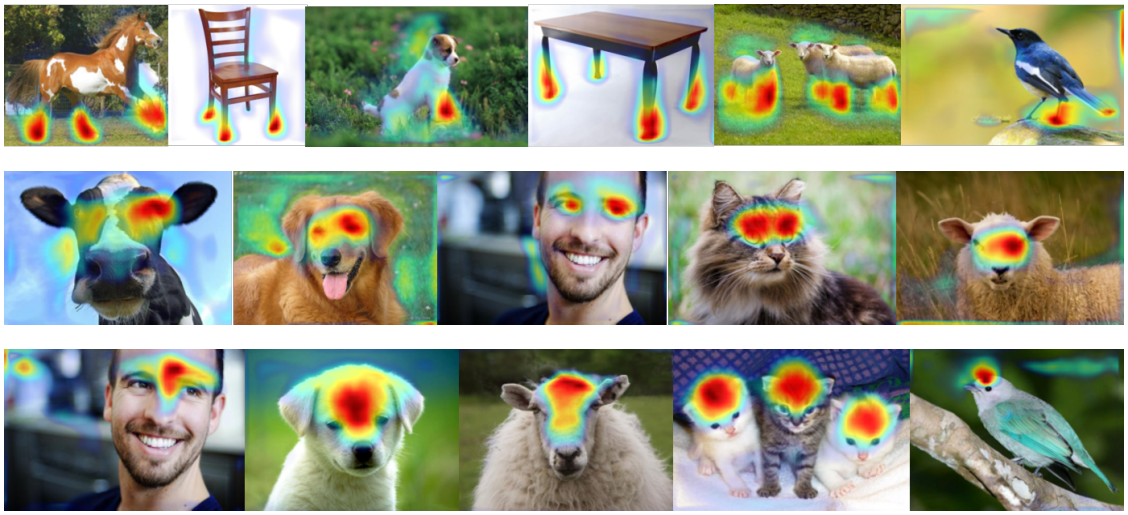


Figure 1.3: Geometry-sensitive features from AnchorNet. Each row depicts a response of a single unsupervised feature detector produced by the AnchorNet architecture from chapter 5. Note that each detector consistently attends to a single mid-level visual element.

Their main use-case is an establishment of pixel-perfect correspondences between views of different instances of an object category. Being able to tackle this task, also known as *semantic correspondence* [Liu et al. 2011], admits solving challenging problems such as category-specific 3D shape learning [Kar et al. 2015a] or inter-image transfer of different pixel-wise annotation types [Liu et al. 2011].

A defining property of keypoints is their sensitivity to the *geometry* of a considered category. More specifically, assuming that an object category shares a common geometric reference (e.g. a 3D human skeleton), a keypoint detector responds to each image-specific realization of a distinct location from the reference (e.g. a detects a wrist in each image of a human). Such requirement of a detectability across all instances of a category implies invariance of keypoints to intra-category appearance variations.

Despite the proven advantages of keypoints, defining them as 2D image locations might be too restrictive. For instance, a nose, an essential part of the geometric coordinate frame of a human face, cannot be represented with a single image point. Instead, other geometric primitives, such as lines or small-scale image segments, can be imagined to strike the intriguing balance between geometry-awareness and the invariance to appearance changes.

In chapter 5 we present a framework for detecting such mid-level geometry-sensitive features without constraining their form to image points. Crucially, we show that the features allow reliable pixel matching across different instances of object categories, just as keypoints do.

While accurate detection of the geometry-sensitive features can be achieved with fully-supervised keypoint detectors [Tulsiani and Malik 2015; Choy et al. 2016a], here, we are again interested in a less costly setup without detailed manual annotations. Our goal is to utilize the popular, inexpensive and abundant image-level classification labels [Deng et al. 2009; Everingham et al. 2011].

Surprisingly, the standard practice of using intermediate features from a pre-trained deep image classifier is inferior to existing hand-coded alternatives such as HoG [Dalal and Triggs 2005a] or SIFT [Lowe 2004a]. Chapter 5 first hypothesizes that the main reason is the geometric invariance acquired as a by-product of optimizing a global image-classification loss. We then proceed with defining a learning setup that removes the harmful invariance in favor of an increased geometric sensitivity of the features. More specifically, this was achieved with a simple pair of constraints enforcing simultaneous discriminability and diversity (D&D) of the trained representation.

1.2.4 Self-supervised learning of geometry-sensitive features through probabilistic introspection

Apart from the combination of discriminability and diversity in chapter 5, several other constraints facilitating geometric feature learning exist. Namely, self-supervised equivariant feature learning from [Rocco et al. 2017; Lenc and Vedaldi 2016; Thewlis et al. 2017a] was shown to produce descriptors suitable for different kinds of geometric tasks. Note that, in case of D&D, the geometric awareness of the learned features is more of an emergent property rather than a direct consequence of optimizing the objective function. This is contrasted with the equivariance constraints [Rocco et al. 2017; Lenc and Vedaldi 2016; Thewlis

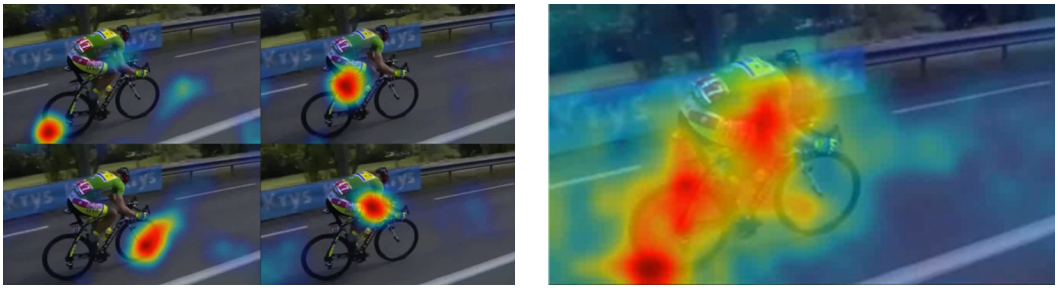


Figure 1.4: Simultaneous learning of a feature detector and descriptor. The probabilistic introspection framework from chapter 6 enables self-supervised learning of geometry-aware features with keypoint-like responses (left) and, simultaneously, learning of a confidence map that highlights interesting foreground patterns (right).

et al. 2017a] which directly steer the feature space to transform with the transformations of the input images, inherently achieving geometric sensitivity. As a result, compared to D&D, equivariance allows for more precise localization of the produced features.

Chapter 6 shows that the increased precision of equivariant features comes at the cost of lower robustness to background clutter and viewpoint changes. Seeking to alleviate this drawback, we propose a novel probabilistic introspection framework which is capable of simultaneous learning of feature descriptors as well as predicting their reliability. This way, it is possible to reject irrelevant background descriptors and focus on learning geometric features of relevant object categories in an annotation-free manner.

An important empirical observation in chapter 6 is that, although the proposed probabilistic introspection objective is in its essence a variant of a matching loss, the output feature responses resemble highly localized keypoint detectors. This indicates an existence of a duality between learning to match and learning to detect keypoints.

1.2.5 Semi-convolutional operators

Chapters 5 and 6 proposed two different ways of decomposing images of object categories into a set of well-localized visual elements that resemble keypoints. A remarkable feature of keypoints is their ability to be predictive of an extent

of the object category they belong to. This property was exploited in the seminal work of Lowe [1999] where votes from several potential object keypoints were accumulated using a hough-voting scheme in order to localize their corresponding objects.

In chapter 7, we revisit this object localization paradigm. More specifically, we tackle the task of instance segmentation as, arguably, it is currently the most challenging form of an object localization task. Similar to the hough voting scheme of Lowe [1999], our proposed solution predicts pixel-wise embeddings that uniquely identify their parent object. Differently from [Lowe 1999], we utilize deep networks - the workhorse of computer vision which, unfortunately, Lowe could not ride in 1999.

While similar formulations were addressed before [De Brabandere et al. 2017; Newell et al. 2017], our main contribution is identifying one of the reasons why these works were unable to outperform other instance segmentation paradigms. This obstacle, termed the *convolutional coloring dilemma*, is connected to the translation invariance of the convolutional operators, the main building blocks of modern deep architectures. We consequently propose semi-convolutional operators, a fix that breaks the translation invariance by making the network outputs position-sensitive.

1.2.6 Capturing the geometry of object categories from video sequences

An important way of representing the structure of an object category is by analyzing its shape in the 3D space. Having the ability to perform category-specific 3D reconstruction finds many applications such as helping robots to interact with their surroundings, driving autonomous cars through complex environments or automatic lifting of movies to three dimensions.

Reconstructing the geometry of an object can be addressed with mature technologies such as structure-from-motion (SfM). The latter does so by exploiting optical constraints of a multi-camera system that observes a given scene from

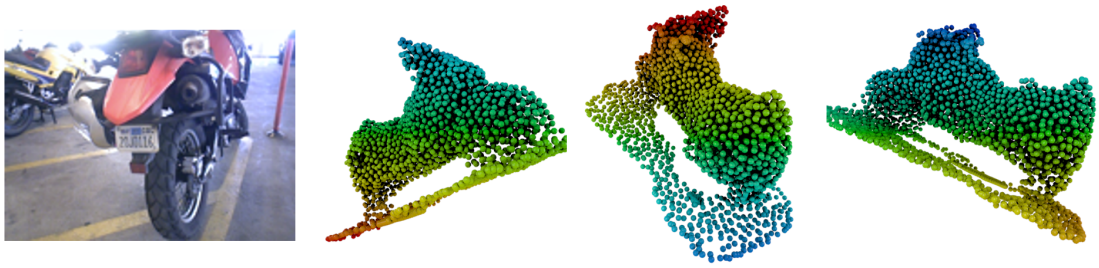


Figure 1.5: Single-view 3D shape estimate of VpDR-Net. Chapter 8 introduces a novel deep architecture, called VpDR-Net, capable of inferring a complete 3D shape of an object (2nd to 4th column) given a single partial view (1st column).

multiple views. This makes such approach inconvenient for our task since these constraints are scene-specific and, as such, do not contain any information about the geometry of the considered object category. Instead, in order to enforce the understanding of the 3D geometry on the level of object categories, we address the task of estimating 3D properties from a single view of an object. Estimating 3D from a single 2D image is an ill-posed problem and, therefore, cannot be tackled with a prior-less approach such as SfM. Instead, models that discover and learn category-specific shape regularities have to be employed.

Following the main narrative of the thesis, we are again facing a problem that is often addressed using an expensive manually annotated dataset (e.g. 3D CAD models [Chang et al. 2015] or manual annotations [Xiang et al. 2014]). We thus propose a solution that significantly decreases the level of supervision by inspiring ourselves with the human visual system. Arguably, humans learn the geometry of object categories by experiencing a sequence of images of an object undergoing a motion. By stitching these individual views together, humans separately reconstruct each scene. They later setup a common geometric reference by collectively analyzing different instances of an object category. In chapter 8 we mimic this learning process by first leveraging SfM for the initial scene reconstruction step. Afterwards a deep architecture that aligns and represents objects in an implicit geometric reference frame of the considered category is proposed. In practice, this approach only requires sufficient number

of videos capturing different objects of the same visual class. Hence, no manual annotations are needed.

1.3 Contributions

This section briefly summarizes the main contributions of this thesis.

Active-transfer learning Chapter 3 describes a learning setup which is a mix of active and transfer learning frameworks. It proposes a novel method, dubbed *auto-validation*, that learns a weighted ensemble of domain specific classifiers in order to increase invariance to domain shifts. The latter is seamlessly connected to an active learning sampler that relies on measuring the disagreement of the individual members of the ensemble. Auto-validation outperforms other alternatives on a publicly available benchmark.

Animal Parts dataset In order to seek the limits of supervision in chapter 3, we collected a novel dataset that is suitable for analyzing active and transfer learning methods. It comprises eye and foot keypoint annotations for 15K images of 100 animal classes from the "vertebrate" subtree of the ImageNet dataset [Deng et al. 2009]. It is freely available at `http://www.robots.ox.ac.uk/~vgg/data/animal_parts/`

Webly-supervised part learning Chapter 4 describes a novel method for parsing objects into semantic parts. The proposed weakly supervised part detector is trained via a multiple instance learning SVM formulation that utilizes a novel *geometry-appearance embedding*. The geometric part of the embedding relies on a set of non-semantic part detectors that serve as *anchors* which help with localization of the semantic parts. State-of-the-art results are reported on public datasets for the semantic matching, part-based image classification and part detection tasks.

IoU is a positive-definite kernel An important theoretical result from chapter 4 is the proof of positive-definiteness of the intersection-over-union similarity measure. Recognizing this property was important to enable the construction of the novel geometry-appearance embedding.

Weakly-supervised geometric feature learning A novel deep architecture, termed AnchorNet, is presented in chapter 5. It is trained with a pair of *discriminability and diversity* objectives that improve the quality of deep features for geometry-related tasks. Importantly, this is achieved with image-level supervision only. It is demonstrated that the features extracted from the proposed architecture significantly improve the performance of methods that estimate semantic correspondences.

Probabilistic instrospection Similar to AnchorNet, chapter 6 revisits learning of geometry-sensitive features. Dropping the discriminability and diversity objectives, we utilize *equivariance* constraints and train a deep architecture in an annotation-free, self-supervised fashion. One of the key contributions is a probabilistic introspection mechanism that allows to automatically identify distinctive features that are likely to result in a correct match.

Resolving the coloring dilemma with semi-convolutions In chapter 7 we introduce the *convolutional coloring dilemma*, a formalization of an obstruction that prevents convolutional architectures from performing well on the instance segmentation task. Later, a replacement of a convolutional operator, termed a *semi-convolutional operator*, that aims to resolve the conundrum is introduced. It is demonstrated that a deep architecture enhanced with the novel operator yields state-of-the-art results on the Pascal VOC instance segmentation task.

Learning 3D object categories by looking around them Chapter 8 presents the first approach that learns monocular 3D shape predictor solely by observing real instances of object categories from a moving camera. The

key novel components are a *viewpoint factorization network* for dealing with ambiguous structure-from-motion supervision and a *point cloud completion network* which represents 3D shape as a probability distribution over a flexible occupancy grid.

Probabilistic learning framework Finally, a generic probability-based framework for measuring uncertainty of deep predictors is proposed. Apart from an assessment of output confidences, it provides a robust learning formulation that explicitly represents and discards noisy ground truth samples. Different variants of the proposed probabilistic learning are employed in chapters 6 and 8.

1.4 Relevant publications

Peer-reviewed publications that are relevant to the content of this thesis are listed below. The references to the corresponding thesis chapters are included as well.

Chapter 3

Novotny, D., Larlus, D., Vedaldi, A.:
I Have Seen Enough: Transferring Parts Across Categories,
BMVC 2016

An extended version was published as a book chapter:

Novotny, D., Larlus, D., Vedaldi, A.:
Generalizing Semantic Part Detectors Across Domains,
Csurka, G., ed.: Domain Adaptation in Computer Vision Applications. Springer 2017

Chapter 4

Novotny, D., Larlus, D., Vedaldi, A.:
Learning the Structure of Objects from Web Supervision,
ECCVW 2016

Chapter 5

Novotny, D., Larlus, D., Vedaldi, A.:
AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching,
CVPR 2017

Chapter 6

Novotny, D.[‡], Albanie, S.[‡], Larlus, D., Vedaldi, A.:
Self-supervised Learning of Geometric Features Through Probabilistic Introspection,
CVPR 2018

Chapter 7

Novotny, D.[‡], Albanie, S.[‡], Larlus, D., Vedaldi, A.:
Semi-convolutional Operators For Instance Segmentation,
ECCV 2018

Chapter 8

Novotny, D., Larlus, D., Vedaldi, A.:
Learning 3D Object Categories by Looking Around Them,
ICCV 2017

An extended version was published in TPAMI:

Novotny, D., Larlus, D., Vedaldi, A.:
Capturing the Geometry of Object Categories from Video Sequences,
TPAMI 2018

[‡]authors contributed equally

2 Background

The contents of chapters 3 to 8 , that thoroughly describe our contributions, are related to many branches of computer vision. Following the ordering of these chapters, an overview of the relevant literature is given below.

2.1 Active-transfer learning

Chapter 3 introduces a domain-invariant keypoint detector trained under limited supervision. The existing methods related to the relevant theoretical frameworks (active learning and domain adaptation) are reviewed in this section.

2.1.1 Active learning

As briefly mentioned above, chapter 3 focuses on scenarios where an annotation budget is assumed. Such tasks are mostly formulated as active learning problems.

The following paragraphs provide an overview of active learning and its applications in computer vision. Active learning [Cohn et al. 1994] differs from the standard inductive learning scenario by allowing an algorithm to choose which data samples should be included in the training process. Usually, active learners alternate between two stages. The first employs a sampling component

which analyzes a large pool of unlabeled training examples and subsequently instructs an annotation oracle to label a fixed-size subset of the examples. In the second stage, a learning algorithm is retrained given the sampled training set. The allowed number of sampled annotations is increased in every round and the goal is to mine annotations such that the performance of the learner on a held-out test set increases at the fastest possible pace.

The selection of the sampling algorithm is an important design choice. A typical active learner defines an expected risk function and aims to minimize it by means of the sampler. For instance, the uncertainty principle [Tong and Koller 2002] minimizes the maximum expected risk by sampling examples which the learner is the least certain about. Additional selection strategies include diversity [Joshi et al. 2009] or relevance [Vendrig et al. 2002].

In the context of computer vision, active learning is a well researched topic. Oldest works focused mainly on image classification or object retrieval and utilized SVM or adaboost based classifiers in combination with the uncertainty samplers [Kapoor et al. 2007; Qi et al. 2008; Holub et al. 2008; Collins et al. 2008]. Subsequent methods aimed at retrieving different levels of annotations at once. For example, [Vijayanarasimhan and Grauman 2009] estimates a cost of different types of annotations and then modulates an expected risk function by it. Russakovsky et al. [2015] design a generic framework aiming at uniformly annotating as many examples in each image as possible. Wah et al. [2011b] proposed a human-assisted active learning approach. [Parkash and Parikh 2012] leverage additional information from different annotation types to improve the convergence properties of the active learner.

2.1.2 Domain adaptation

Chapter 3 addresses learning detectors of animal parts that are shared across a large variety of different species. Intuitively, a successful detector has to be capable of bridging the large intra-species appearance variations. By treating individual animal species as data "domains", we can pose the problem as an

instantiation of the domain adaptation framework. An overview of the relevant literature is presented in this section.

Domain adaptation (DA) [Daume III and Marcu 2006] seeks to learn classifiers that account for a shift between different distributions within the training data. A standard assumption is that the train data consist of a pair of distributions called the Source and Target domains (D_S and D_T respectivelly). The domains are further characterized by the amount of supervision available. Typically all the samples from the source domain D_S are labeled while there are various setups for the target domain. In the unsupervised DA, none of the labels from the target domain are available. Semi-supervised DA assumes an availability of labels for a subset of D_T. A detailed overview of DA approaches is present in [Csurka 2017].

DA has been tackled for the first time in computer vision by Saenko et al. [2010]. Later on, the problem received a lot of attention. A notable line of work was based on projecting features to a set of subspaces spanned by points on a Grassman manifold [Gopalan et al. 2011; Gong et al. 2012]. In a similar spirit, [Ni et al. 2013] learns projections on intermediate dictionaries located between D_S and D_T. Two approaches that learn a linear transformation using a simple alignment procedure of D_S and D_T were introduced in [Fernando et al. 2013; Sun et al. 2016]. A more complex method that learns feature projection to a common domain-invariant space by minimizing the maximum-mean-discrepancy (MMD) was described in [Baktashmotlagh et al. 2013]. Other works [Hoffman et al. 2013; Yang et al. 2007] proposed modifications of SVM classifiers to account for the domain shift.

Deep DA approaches emerged soon after the success of AlexNet [Krizhevsky et al. 2012b]. Chopra et al. [2013] presented an architecture that was first pretrained in an unsupervised fashion to account for the domain shift and then discriminatively finetuned. Tzeng et al. [2015] and Ganin and Lempitsky [2015] introduce a domain confusion classifier which facilitates the domain invariance of the learned representations. [Tzeng et al. 2014] uses a multiple kernel variant

of the MMD loss to make the final layers of their CNN architecture insensitive to the domain shift.

2.2 Learning object parts with weak supervision

A novel webly supervised approach for parsing objects into semantic parts is presented in chapter 4. In order to present relevant literature, below, we firstly discuss methods that utilize web supervision. Since our method is a variant of a weakly supervised detector, weakly supervised recognition systems are revised in section 2.2.2. Finally, a brief overview of semantic part detectors is given.

2.2.1 Learning from web supervision

Modern deep learning algorithms have achieved great results also due to the emergence of large annotated datasets such as ImageNet [Krizhevsky et al. 2012b] or MS COCO [Lin et al. 2014]. However, collecting such vast amount of data requires a large annotation effort. For instance authors of [Krizhevsky et al. 2012b] claim that it "is already the largest clean image dataset available to the vision research community, in terms of the total number of images, number of images per category as well as the number of categories with 50 million cleanly labeled full resolution images".

Several works proposed to minimize the amount of required supervision. One-shot learning [Fei-Fei et al. 2006; Aytar and Zisserman 2012] considers a learning setup with a single available strong annotation per class. Zero-shot learning [Lampert et al. 2009; Parikh and Grauman 2011] uses attribute-based supervision to discover previously unseen object classes. Perhaps the most readily available source of annotated visual data is the world-wide-web. With the proliferation of hand-held image capturing devices, the amount of visual data available in various locations of the internet became nearly infinite. While the relevance of images retrieved by the most profound search engines such as Google or MS BING is far from being perfect, the amount of required explicit supervision shrinks only to designing the text query phrases.

In 2004, Fergus et al. [2004], extended the constellation model [Fergus et al. 2003] to re-rank Google image search results. Differently from then language-based Google image search, the re-ranking system was working on the basis of visual cues. The method was further improved in [Fergus et al. 2005]. Later, Schroff et al. [2007] combined visual features with the textual ones to improve the re-ranking performance.

Approaches from [Li et al. 2007; Berg and Forsyth 2006] introduced the concept of alternating between identifying clean web images and retraining classifiers on those. Vijayanarasimhan and Grauman [2008] formalized this algorithm as a sparse MIL classifier whose optimization resembled the MI-SVM concave-convex procedure [Andrews et al. 2002]. Fergus et al. [2009] presented a semi-supervised framework based on graph laplacians to clean up huge noisy collections of web images. Similarly, Bergamo and Torresani [2010] proposed a domain transfer approach to webly supervised image recognition.

Tang et al. [2014] recently introduced a co-localization approach applicable for the noisy web images. It comprises a quadratic programming problem that picks the most representative bounding box in each image. A limitation are significant memory requirements which scale quadratically with the number of bounding boxes that are being co-localized.

Divvala et al. [2014] aim to cover the visual variance of as many visual concepts as possible with a webly supervised method. The algorithm achieves this goal by identifying a set of visually salient adjectives for each object query phrase a then using those adjectives to query clean images for supervising object detectors. Quantitative results are reported for object detection and action recognition tasks.

Closely related to our work, the never ending image learner (NEIL) [Chen et al. 2013] aims to harvest common sense knowledge from web images. The approach alternates between training a set of ELDA [Hariharan et al. 2012] detectors on a set of retrieved images for a given query phrase, clustering the

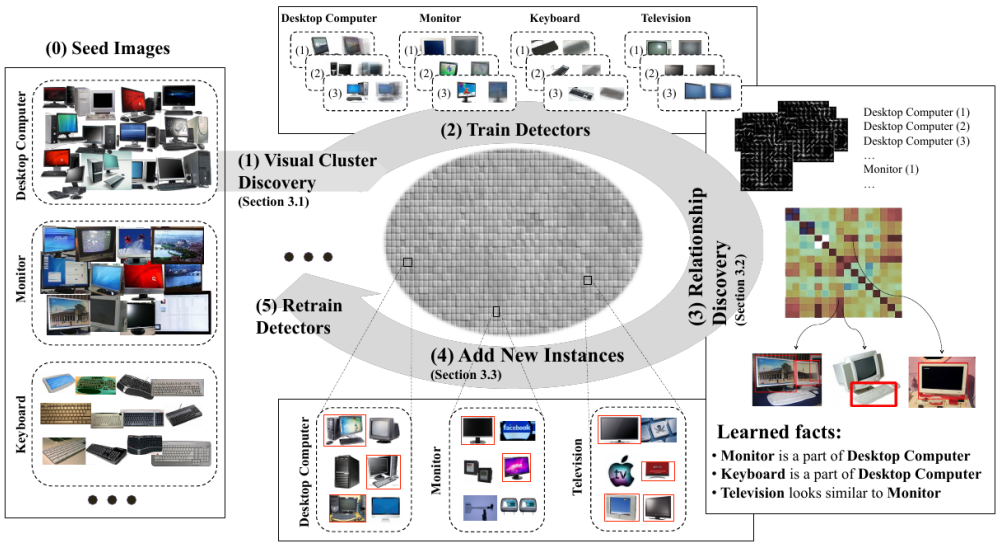


Figure 2.1: Never Ending Image Learner. Chen et al. [2013] learn visual concepts from unconstrained web image data by alternating between clustering images to infer new visual categories, training category detectors and inferring visual relationships between the visual concepts. Figure taken from Chen et al. [2013].

detector responses to obtain new object instances and mining new types of relationships between these objects. The approach is capable of identifying 4 different relationship types including the "part-of" relationship. This relationship is modeled rather vaguely by averaging the relative positions and scales of the parts and their parent objects. The quality of the discovered relationships is not quantitatively assessed.

The robustness of deep classifiers to annotation noise motivated their use for webly supervised learning. Chen and Gupta [2015] first learn an image classification network from scratch using pure web supervision followed by estimating a class confusion matrix on the training set. Features from the pretrained network are later used for training object detectors that are later evaluated on an object detection task. The problem of webly supervised co-segmentation has been studied in [Kim and Xing 2012; Rubinstein et al. 2013; Izadinia et al. 2015]. All the approaches report semantic object segmentation results.

To conclude, there exists a plethora of methods that perform webly supervised image classification, object detection or segmentation. With the exception of [Chen et al. 2013] none of the methods targets webly supervised parsing

of objects into semantic parts. To the best of our knowledge, at the time of the submission of [Novotny et al. 2016b] there did not exist any webly supervised method (including [Chen et al. 2013]) that provides quantitative results for the object part localization task. After the publication of [Novotny et al. 2016b], the task of webly-supervised part learning was revisited in [Modolo and Ferrari 2018].

2.2.2 Weakly supervised image recognition

Since chapter 4 formalizes the addressed webly supervised recognition task as an instance of weakly supervised object localization, the relevant literature is discussed below.

Weakly supervised object localization (WSOL) problems consider learning models with incomplete supervision in form of image-level annotations which denote a presence of an object category. The problem can be further categorized as follows: (a) *weakly supervised object detection (WSOD)* [Nguyen et al. 2009; Pandey and Lazebnik 2011; Deselaers et al. 2012a; Wang et al. 2014; Hoffman et al. 2014; Hoffman et al. 2015; Cinbis et al. 2015; Bilen and Vedaldi 2015] accepts a list of images together with binary labels indicating presence of an object class and outputs an object detector; (b) *co-detection (co-localization)* [Joulin et al. 2014; Tang et al. 2014; Ali and Saenko 2014; Shi et al. 2015] considers a list of images that contain at least one instance of a given object class and outputs a location of each instance. (c) *co-segmentation* [Joulin et al. 2010; Vicente et al. 2011; Joulin et al. 2012; Rubinstein et al. 2013] assumes the same input as co-detection, but produces foreground/background segmentation masks for each input image. As our work considers bounding boxes as an output, we discuss only co-localization and WSOD.

The most common formulation of the weakly supervised object localization is via the multiple instance learning framework (MIL) [Dietterich et al. 1997]. In this scenario, instead of having access to explicit labels of each training sample, labels are assigned to "bags" of instances.

The first method that framed weakly supervised object localization as a MIL problem was authored by Maron and Ratan [1998]. They proposed the Diverse Density algorithm [Maron and Lozano-Pérez 1998] to address recognition of natural images. Although the MIL formulation of WSOL prevailed in the community, the actual Diverse Density algorithm was abandoned in favor of the structured output SVM [Yu and Joachims 2009] or MI-SVM [Andrews et al. 2002] formulations.

Nguyen et al. [2009] were among the first to propose to use MI-SVM for WSOD. Blaschko et al. [2010] used structured output SVMs for WSOD. In [Pandey and Lazebnik 2011], the DPM LSVM formulation [Felzenszwalb et al. 2008b] was adjusted to leverage only image-level labels for WSOD. Li et al. [2013] utilized mi-SVM [Andrews et al. 2002] on top of BoV features to train webly supervised object detectors for localization of semantic mid-level elements. "Object centric pooling", introduced by [Russakovsky et al. 2012], uses foreground-background descriptors in combination with MI-SVM to improve detection performance.

Several works have investigated possible improvements to the crucial initialization step of MIL detectors. The standard practice [Russakovsky et al. 2012; Pandey and Lazebnik 2011] is to initialize the representatives of positive bags to bounding boxes covering whole images. Methods from [Siva et al. 2013; Deselaers et al. 2012b] utilize saliency/objectness measures to indentify likely foreground objects. Multifold-MIL [Cinbis et al. 2014] performs relocalizations and training on independent training data folds.

Smooth relaxations of the non-differentiable LSVM objective function were also proposed. Song et al. [2014] used Nesterov's smoothing [Nesterov 2005], while Bilen et al. [2014] relaxed the max function with the softmax approximation.

Motivated by the success of CNNs on the object detection task [Girshick et al. 2014b], deep features have been adopted for weakly supervised object detectorion. Works from [Wang et al. 2014; Bilen et al. 2014; Song et al. 2014]

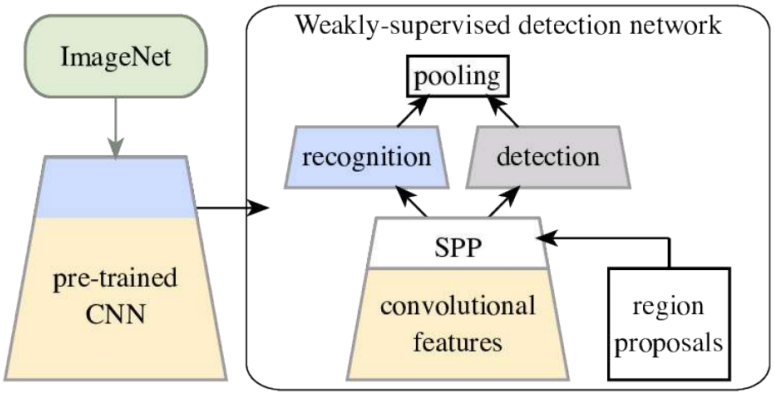


Figure 2.2: Weakly Supervised Deep Detection Network (WSDNN) jointly learns a region selector and classifier from data weakly annotated with image-level labels. Figure taken from [Bilen and Vedaldi 2015].

combined pre-trained deep features with the MIL framework. An early end-to-end trained deep approach was proposed by Oquab et al. [2015]. However, the method was only capable of correctly selecting the most discriminative pixel within each object instance. Recently, Bilen and Vedaldi [2015] surpassed the results of the MIL trained pipelines with an end-to-end trained CNN (fig. 2.2). The cascaded architecture from [Diba et al. 2017] later recorded quantitative results superior to [Bilen and Vedaldi 2015].

2.2.3 Semantic part detection

Chapter 4 proposes a method that addresses detection of semantic (nameable) parts. Although, as outlined above, the webly supervised approaches to this task have not been closely explored before, here we give a brief overview of the fully-supervised ones.

There is a large body of work related to semantic part detection. The field with the highest attention is probably pose-estimation. Some notable approaches are [Yang and Ramanan 2013; Toshev and Szegedy 2014; Tompson et al. 2014; Bourdev et al. 2010]. However we will not discuss these in detail as they are significantly class-specific and do not fit into our generic object part detection scenario. Similarly, there exists a plethora of methods on detecting parts of birds [Wah et al. 2011a].

Earlier generic approaches adopted the famous DPM framework for part detection [Zhang et al. 2013; Chai et al. 2013]. With the proliferation of deep networks, Zhang et al. [2014a], Gkioxari et al. [2015], and Simon and Rodner [2015] replaced the hand-coded appearance descriptors with pre-trained deep features and combined them with a simple deformation model.

Closer to our work, strong DPM [Azizpour and Laptev 2012] was one of first methods that targeted generic detection of animal parts. Later, [Chen et al. 2014a] used a CRF part-based detector to approach a similar task. Hariharan et al. [2015] and Wang et al. [2015a] proposed different CNN architectures for semantic animal part segmentation.

2.3 Geometry-aware representations

Chapters 5 and 6 propose novel strategies for learning geometry-aware feature representations. To put these works into a context, several relevant image representations are reviewed in this section. First, an overview of traditional hand-engineered and learned features for pixel-wise matching is presented. More related to our task of semantic correspondence, existing approaches for unsupervised learning of mid-level elements of object categories are then revised.

2.3.1 Features for dense correspondence

Early hand-engineered features, such as SIFT [Lowe 2004b], HoG [Dalal and Triggs 2005a], SURF [Bay et al. 2008] or Daisy [Tola et al. 2010] accumulated low-level cues (e.g. distributions of image gradient [Dalal and Triggs 2005b; Lowe 2004b] or responses to a Haar wavelet basis [Papageorgiou et al. 1998]) in local neighborhoods of an image point. In order to achieve (limited) invariance to viewpoint changes, patches have been cannonicalized by e.g. finding the dominant gradient orientation [Lowe 2004b].

Learned alternatives to the hand-crafted descriptors have been later proposed. A convex descriptor optimization framework was introduced in [Simonyan et al. 2014]. The recent success of deep learning brought approaches trained on large

amounts of annotated data. MatchNet [Han et al. 2015] proposed a siamese architecture followed by a network that learns a pairwise descriptor similarity metric. Concurrently, Zagoruyko and Komodakis [2015] explored several other variants of this deep architecture. Instead of learning a separate similarity predictor, Simo-Serra et al. [2015] directly embedded pixels into an Euclidean space which makes the representation applicable for a larger variety of tasks. The approach from [Simo-Serra et al. 2015] was later improved in [Yi et al. 2016] by adding differentiable patch-normalization layers. A similar architecture was later adopted for the semantic matching task in [Choy et al. 2016a].

One of the heftiest challenges of descriptor learning is the selection of relevant training pairs/triplets from the vast quantity of available examples. Improving the learning dynamics, Mishchuk et al. [2017] attained state-of-the-art results with a novel hard-negative mining strategy.

2.3.2 Data-driven feature learning

While the previous section discussed either hand-engineered representations or those which are trained from annotated datasets, here, unsupervised data-driven approaches that discover useful visual patterns are reviewed. The feature learning methods proposed in chapters 5 and 6 are members of this category.

Parts. In order to mine for mid-level visual patterns, various types of constraints have been proposed. Perhaps the most obvious property of a useful mid-level element is its "representativeness", i.e. frequent occurrence in the visual world. Several clustering algorithms that identify frequently occurring visual patterns are standard components of part discovery approaches [Doersch et al. 2013; Li et al. 2015b; Sun and Ponce 2015; Doersch et al. 2015].

The predictability of the corresponding parent object class denotes another popular constraint. An empirical loss term which favors discriminative parts was employed in representative part-based detection/classification approaches such as [Felzenszwalb et al. 2010a; Juneja et al. 2013; Doersch et al. 2013; Bossard

et al. 2014]. A recent contribution from [Bristow et al. 2015] highlights discriminative features in an image using a swiftly trained LDA classifier.

A stable geometric arrangement is another potential sign of an informative visual pattern. In the context of object detection, part constellations [Fergus et al. 2003] disambiguate between seldom noisy background regions and geometrically stable part occurrences. Notable approaches that utilize an explicit geometry model are [Felzenszwalb and Huttenlocher 2003; Felzenszwalb et al. 2008b; Zhang et al. 2014a; Yang and Ramanan 2013; Chen et al. 2014a]. Similarly, Singh et al. [2012] and Yang et al. [2012a] discover "doublets" (first introduced in [Sivic et al. 2005]) of co-occurring parts. Geometric arrangements can be also mixed with self-similarity in the appearance space, giving rise to the self-similarity descriptor [Shechtman and Irani 2007]. Its deep variant was later proposed in [Kim et al. 2017] for the semantic matching task.

Learning via auxiliary tasks. While early approaches for data-driven feature learning have mostly focused on variants of discriminative clustering with additional geometric constraints, the remarkable ability of modern deep architectures to learn highly non-linear mappings brought a plethora of new possibilities. Several works have defined a discriminative auxiliary objective that, once optimized, leads to well-performing representations at a limited annotation cost.

Perhaps the most readily available auxiliary task is image classification. Indeed, features from intermediate layers of pre-trained deep image classifiers have been successfully transfered to other tasks such as instance retrieval [Tolias et al. 2015], object detection [Girshick et al. 2014c] or fine-grained image classification [Azizpour et al. 2015]. Regarding the geometric tasks, an evaluation of pre-trained image classification features on the classic descriptor matching is provided in [Fischer et al. 2014]. More related to our problem of semantic matching, Long et al. [2014] analyzed performance of deep features for category-specific keypoint detection and semantic correspondence estimation. Surprisingly, they have not recorded a significant improvement over hand-engineered

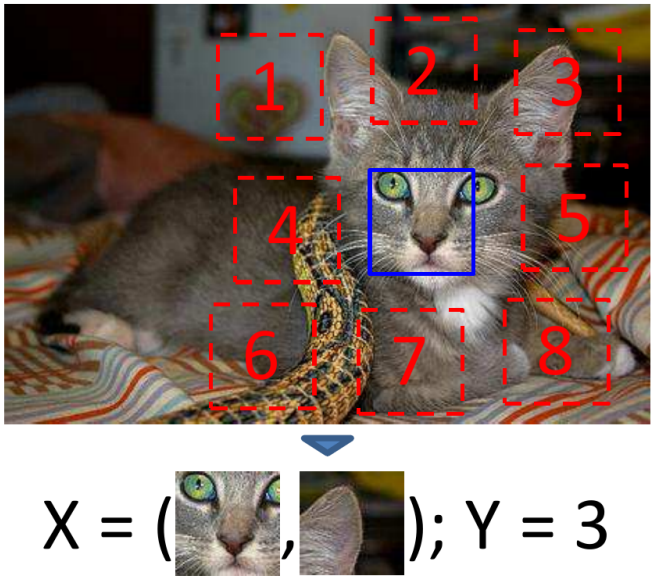


Figure 2.3: Learning representations by predicting context. Doersch et al. [2015] propose a self-supervised network that learns by guessing a relative position between two randomly sampled image patches. Figure taken from Doersch et al. [2015].

alternatives (SIFT). This observation constitutes one of the motivations for the feature learning architecture proposed in chapter 5.

Apart from image classification, many different auxiliary tasks were recently proposed. Dosovitskiy et al. [2014] instructed their network to discriminate between surrogate categories, each formed by a set of random warps of a single image patch. Inspired by the skip-gram models in the natural language processing community [Mikolov et al. 2013], discovering features with a stable visual context was developed by Doersch et al. [2014]. This method was further simplified and improved with deep networks in [Doersch et al. 2015]. Similarly, Noroozi and Favaro [2016] train a network to re-assemble shuffled image patches. Other color-based sources of supervisory signal include image colorization [Zhang et al. 2016] or inpainting [Pathak et al. 2016]. Steering a network towards understanding how temporal constraints manifest in videos gives rise to a convenient video representation [Misra et al. 2016; Fernando et al. 2017].

Equivariant features. Recently, several works addressed learning image representations that are equivariant with geometric transforms of an input image.

Schmidt and Roth [2012] and Kivinen and Williams [2011] alternated the convolutional Restricted Boltzmann Machines [Lee et al. 2009; Norouzi et al. 2009] to make them equivariant with image rotations and translations. Jayaraman and Grauman [2015] proposed a deep convolutional network that produces a global representation equivariant with more plausible image transformations generated by a camera ego-motion.

More related to the probabilistic introspection framework from chapter 6, [Lenc and Vedaldi 2016] learned a deep siamese architecture for local feature detection by enforcing equivariance of the detector response with synthetic image transformations. While the network from [Lenc and Vedaldi 2016] relies on the LoG operator [Lowe 1999] for initial filtering of interest points, Quad-networks [Savinov et al. 2017] overcome this drawback by learning the detector from scratch by enforcing stable ranking of detections under image transformations.

Unsupervised learning of category-specific keypoints has been addressed in [Thewlis et al. 2017a]. Here, feature detectors are constrained to have a peaky response which transforms according to the transformations of an input image. This equivariance constraint is enforced by utilizing a synthetically generated dataset of image pairs with a known ground truth optical flow between them. In more detail, each image in a large dataset biased towards a single object category (e.g. faces of humans or cats) is transformed twice using a randomly generated image transformation. The ground truth pairwise flow is then determined by composing the two transformations. Remarkably, despite the lack of cross-instance training correspondences, the network learns fully-fledged landmark detectors that are invariant to inter-category appearance variations.

Arguably, one of the main motivations of representing mid-level image features with keypoints is the simplicity of obtaining the supervisory data. Once, as proposed in [Thewlis et al. 2017a], an unsupervised setup is introduced, the requirement of sparse keypoint-like responses does not seem relevant anymore. Increasing the power of keypoints to represent category structure, in [Thewlis et al. 2017b], the keypoint detectors have been replaced with a dense pixel-wise

labeling. Here, each pixel-label corresponds to a coordinate from an implicit geometric frame of an object category (e.g. a point on a human face). While the approaches from [Thewlis et al. 2017a; Thewlis et al. 2017a] provide highly accurate features, the inability of the models to represent keypoint occlusions causes a failure to converge on datasets with unconstrained out-of-plane transformations of the input categories. This drawback is addressed in chapter 6.

2.4 Establishing correspondences

In chapters 5 and 6, we tackle the semantic correspondence task. This section revises approaches for the traditional correspondence estimation task and for the semantic correspondence task.

2.4.1 Single-scene correspondence

The traditional correspondence estimation task is defined for different views of the same scene. Relevant methods typically compensate for the brittle nature of hand-crafted features (SIFT, HoG, SURF, etc.) by introducing powerful geometric regularizers. Noting that the literature on this topic is very abundant and that, within this thesis, we focus on the semantic matching, this part summarizes the main methods addressing the traditional correspondence estimation task.

RANSAC. In scenarios where a rigid motion between two images of a scene can be assumed, RANSAC [Fischler and Bolles 1981] provides strong constraints on the plausible arrangement of tentatively estimated correspondences. Combined with descriptor-based matching checks (e.g. second nearest neighbor ratio [Lowe 2004a]), RANSAC-based verification of correspondences still remains a dominant component of state-of-the-art structure-from-motion pipelines [Schönberger and Frahm 2016]. Different variants of RANSAC exist and improve its speed [Nistér 2003; Chum and Matas 2005; Chum et al. 2003] or the quality of the matches [Chum et al. 2003; Torr and Zisserman 2000].

Matching as an optimization problem. In a more principled fashion, Maciel and Costeira [2003] have defined the correspondence estimation task as a combinatorial optimization problem. Relaxing the integral formalization from [Maciel and Costeira 2003], Leordeanu and Hebert [2005] included the geometric constraints within a graph whose nodes were tentative matches. The edges then denoted measures of compatibility between all the pairs of the matches. The adjacency matrix of the graph was then factorized and the dominant cluster, corresponding to the main eigenvector of the matrix, then represented the final set of geometrically verified correspondences. The method was later improved in [Duchenne et al. 2011] by considering higher order interactions between correspondences. In [Zanfir and Sminchisescu 2018], the graph matching framework was converted into a differentiable function and implemented using a deep network for semantic and traditional matching tasks.

Dense correspondence. While RANSAC and spectral matching methods are capable of outputting a sparse set of image-to-image correspondences that satisfy various types of geometric constraints, dense matching methods aim at determining for each pixel from the source view a corresponding one in a target view. The output of such algorithms consists of a flow field containing a displacement vector for every source pixel.

The seminal approach of Horn and Schunck [1993] contained a spatial regularizer that enforced similarity of the displacement field between adjacent pixels. Later, Brox et al. [2004] mixed several improvements into a principled energy minimization framework. While [Horn and Schunck 1993; Brox et al. 2004] operated on raw pixel intensities or image gradients, [Brox et al. 2009] proposed to evaluate appearance similarities in a space of pixel-wise image descriptors. Inspired by deep convolutional architectures, Deep Matching [Revaud et al. 2015a] proposed a fine to coarse way of aggregating similarities between patches. The initial set of sparse matches was later incorporated into an altered version of the matching framework from [Brox et al. 2009] in order to produce a dense flow field. Such dense variant of Deep Matching was called DeepFlow.

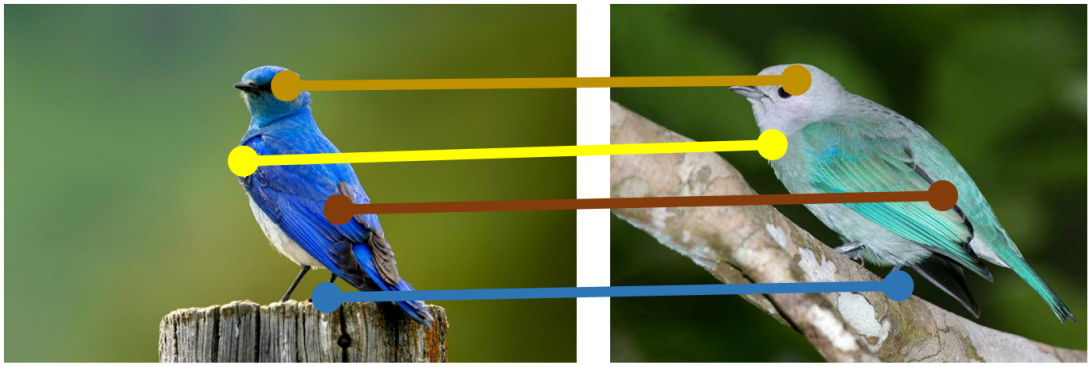


Figure 2.4: Semantic matching aims at estimating pixel-wise correspondences between different scenes that are semantically related. For instance, they contain instances of the same object category.

EpicFlow [Revaud et al. 2015b] improved the dense flow field generation of DeepFlow with an interpolation scheme based on a geodesic distance kernel defined over an edge map.

The advent of deep learning brought several successful architectures suitable for estimating optical flow. FlowNet [Dosovitskiy et al. 2015] demonstrated the possibility of designing a deep architecture for flow estimation. Despite training on a large synthetic dataset, its performance was still inferior to EpicFlow. Thewlis et al. [2016] implemented a fully differentiable variant of Deep Matching and recorded performance improvements over FlowNet. Yet, their network was still inferior to existing shallow approaches. FlowNet 2.0 [Ilg et al. 2017] is an evolution of FlowNet that brought results on par with the state-of-the-art.

2.4.2 Semantic correspondence

While the majority of the matching literature covers the traditional scenario of establishing correspondences between views of a single scene, *semantic matching* generalizes the task to matching views of semantically related scenes (fig. 2.4).

Early approaches. SIFT Flow [Liu et al. 2011], a pioneering work in this field, proposed an energy minimization framework for estimation of an optical flow between pairs of images. The matching objective combined a SIFT descriptor

matching term and a smoothness term that enforced constancy of adjacent displacements. A similar spatial regularizer was used in PatchMatch [Barnes et al. 2009], where high quality matches propagated displacements into their neighborhood. While SIFT Flow independently optimized multi-scale flow fields, DSP [Kim et al. 2013a] minimized matching energy in an end-to-end fashion across all levels of the flow pyramid. DSP was enhanced in [Hur et al. 2015] with geometric regularizers that constraint the parameters of rigid motions of neighboring displacement cells.

Proposal Flow (PF) [Ham et al. 2016] utilized object proposals in order to filter out background image regions that are unlikely to generate reliable cross-scene matches. Furthermore, the displacements generated by region matches were spatially regularized using a hough-voting scheme in order to facilitate rigid motion between scenes. Surprisingly, the best performing variant of PF relied on hand-crafted descriptors (HoG) rather than pre-trained deep features (which have been shown to perform well for the traditional single-scene matching task [Fischer et al. 2014]). One of our objectives in chapter 5 is to learn a deep representation that, after combining with PF, outperforms the hand-engineered features. A fully supervised method for semantic matching, strongly inspired by [Ham et al. 2016], was later proposed in [Han et al. 2017].

Collective alignment. Several methods have tackled the problem of finding stable cross-scene correspondences by exploiting constraints that arise by analyzing a collection of images. The seminal work of Congealing [Learned-Miller 2006] was the first serious attempt to perform such analysis from a geometric standpoint. For each image in a collection, Congealing finds a warp that transforms it into a canonical frame where all images of the collection align. This is done through an iterative process of lowering the entropy of the distribution of the image set in the aligned space. RASL [Peng et al. 2012] poses the joint alignment problem as a rank minimization task. Intuitively, when a batch of images aligns correctly, the rank of the resulting matrix is low, ideally one. In order to discount inevitable errors caused by image corruptions and

occlusions, the low rank recovery is performed up to an additional error term represented as a sparse matrix with a penalty imposed on its ℓ_0 norm. In Collection Flow [Kemelmacher-Shlizerman and Seitz 2012], the transformations into the shared canonical frame are not drawn from a given transformation group, as in [Learned-Miller 2006; Peng et al. 2012], but are rather non-parametrically defined with a per-pixel flow.

Assuming an existence of a common space where all images from the collection align is feasible for restricted domains, such as faces, which do not contain gross out-of-plane rotations. In case the data lie on a complex manifold, where alignment is possible only among the nearest neighbors, alternatives approaches have to be employed. As a replacement for the explicit collective alignment, several works exploited cycle consistency. While loop constraints have been employed in many different contexts before [Zach et al. 2010; Wang et al. 2013], they were first introduced for collective alignment in [Huang and Guibas 2013]. Here, cycle consistency helped obtaining globally consistent matches of vertices among 3D shapes. A significant contribution was an elegant formalization of the 3-cycle consistency constraint as a positive semi-defitiness of a matrix that contains all vertex-to-vertex matches in the shape collection. This methodology was later adopted for collective image matching in [Pachauri et al. 2013]. FlowWeb [Zhou et al. 2015a] introduced an alternative approach to verify 3-cycle consistent matches by traversing the graph of all possible flows between pairs of images, leading to a highly non-convex objective. The optimization process was later improved in [Zhou et al. 2015b] by adopting a low-rank matrix recovery formalization from [Huang and Guibas 2013]. A common drawback of the aforementioned collective aligners is the need to manipulate a large global matching matrix which causes them to leave a large memory footprint.

The approach of [Huang and Guibas 2013] is based on the observation from [Pachauri et al. 2013; Huang and Guibas 2013] that a set of pixels that are matching in a cycle consistent manner is generated by a single common feature from the shared virtual feature "universe". Our method from chapter 5 is motivated

by this intriguing theoretical result. However, instead of recovering the feature universe from a set of tentative matches, we approach the problem in a reverse fashion by directly labeling pixels with their identities subject to the diversity and discriminability constraints. The ability to optimize the D&D objective using SGD, that processes a small number of images at a time, overcomes the large memory requirement of the collective matching methods.

Deep methods. Several methods explored learning deep architectures for semantic matching. WarpNet [Kanazawa et al. 2016] first generated ground truth image-to-image correspondences using an off-the-shelf algorithm. Later, a point transformer network was trained to output parameters of a thin plate spline (TPS) warp between the image pairs. The predicted warps were then used for obtaining single-view 3D reconstructions of birds. In [Rocco et al. 2017], the requirement of producing pairwise image annotations using an off-the-shelf algorithm was alleviated with self-supervision. More exactly, each image, coming from a large unconstrained database, was converted into a flow-annotated image pair by transforming it twice using a randomly generated thin-plate-spline transformation. Rocco et al. [2017] then train a siamese network that, in a reverse fashion, maps the input image pair to the parameters of the generated warp (fig. 2.5). Interestingly, the network was able to generalize from the training setup of matching similar scenes to the test setup of matching pairs of scenes that were only semantically related. In chapter 6, we adopt this self-supervised generation of training pairs. We differ in our main goal which is to learn a geometry sensitive representation instead of predicting image-to-image warps. Outputting a generic pixel-wise embedding allows to transfer the features for a broader range of related tasks such as few-shot keypoint learning.

In order to improve robustness to intra-class variations, Rocco et al. [2018] have later trained a network using pairs of images depicting different instances of the same object category. More specifically, inspired by RANSAC [Fischler and Bolles 1981], their loss maximizes the number of inliers which is consistent with a predicted TPS image-to-image transformation. Similar to [Rocco et al.

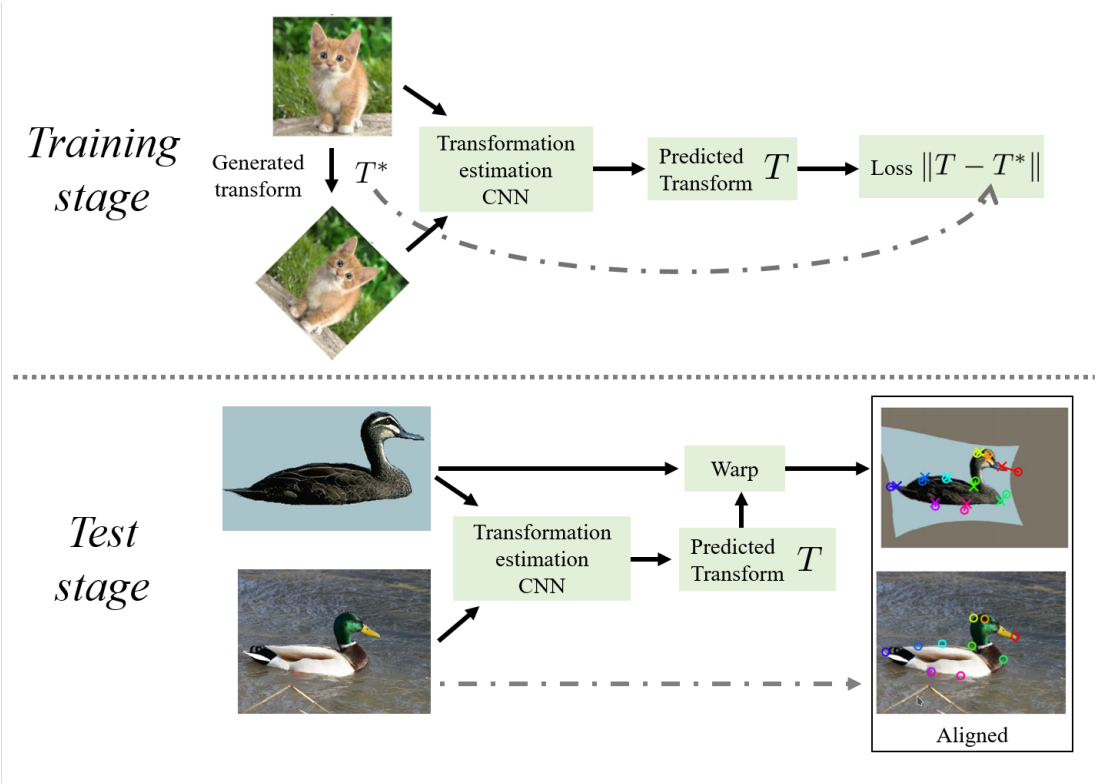


Figure 2.5: Geometric CNN [Rocco et al. 2017] trains a deep self-supervised regressor of transformations between pairs of images of the same scene (top). At test time, the network is capable of aligning different scenes that are only semantically related (bottom).

2017], the transformation parameters are predicted using a FlowNet-style deep architecture. Zhou et al. [2016] supervised their deep semantic matcher with flows generated using a synthetic dataset of 3D models. Their method aligns pairs of real images by composing real-to-synthetic, synthetic-to-synthetic and synthetic-to-real flows. In order to encourage correct composition of flows, the cycle consistency constraint is adopted.

2.5 Instance segmentation

Chapter 7 addresses the instance segmentation task. Existing relevant publications are reviewed in this section.

Originally, the community focused on the semantic segmentation task where the goal is to independently label each image pixel with the semantic class it

belongs to. Many successful solutions, leveraging e.g. normalized cuts [Shi and Malik 2000], graph cuts [Boykov et al. 2001], Conditional Random Fields [Koltun 2011] or convolutional networks [Long et al. 2015] have been proposed.

Propose and Verify

One of the first works that introduced the idea of distinguishing pixels of individual object instances rather than merely recognizing their class was authored by Ramanan [2007]. Here, each candidate detection (output by a sliding window detector) was postprocessed with a heavier algorithm (graphcut) in order to produce pixel-accurate object masks. This instance segmentation paradigm, which we term *propose & verify* (P&V), was later revisited many times. The instance segmentation task was directly addressed for the first time in [Ladický et al. 2010]. Here, the main technical difference from [Ramanan 2007] was the usage of an MRF segmenter instead of graphcut. Parkhi et al. [2011] have demonstrated that the candidate proposals do not have to cover the whole surface of an object category, as an object detector does, but can rather serve as a weak cue to seed the ensuing segmentation process. The task of differentiating instances was later addressed with a novel depth-wise ordering of the object detections in [Yang et al. 2012b].

Deep instance segmentation. Deep approaches have arrived in the form of shallow models trained on top of deep features pre-trained on image categorization Hariharan et al. [2014]. In more detail, MCG candidate segments [Arbelaez et al. 2014] were first filtered using an SVM classifier operating on deep segment descriptors. Later, convolutional features were leveraged to refine the initial segmentations. A similar approach from [Hariharan et al. 2015] enhanced the deep representation by gathering information from multiple image scales.

End-to-end trained deep architectures have been later introduced. DeepMask [Pinheiro et al. 2015] trained a recursive deep network that produced a hierarchy of object candidates which were classified. The successful Faster R-CNN detector [Ren et al. 2015] was adapted for instance segmentation in [Dai

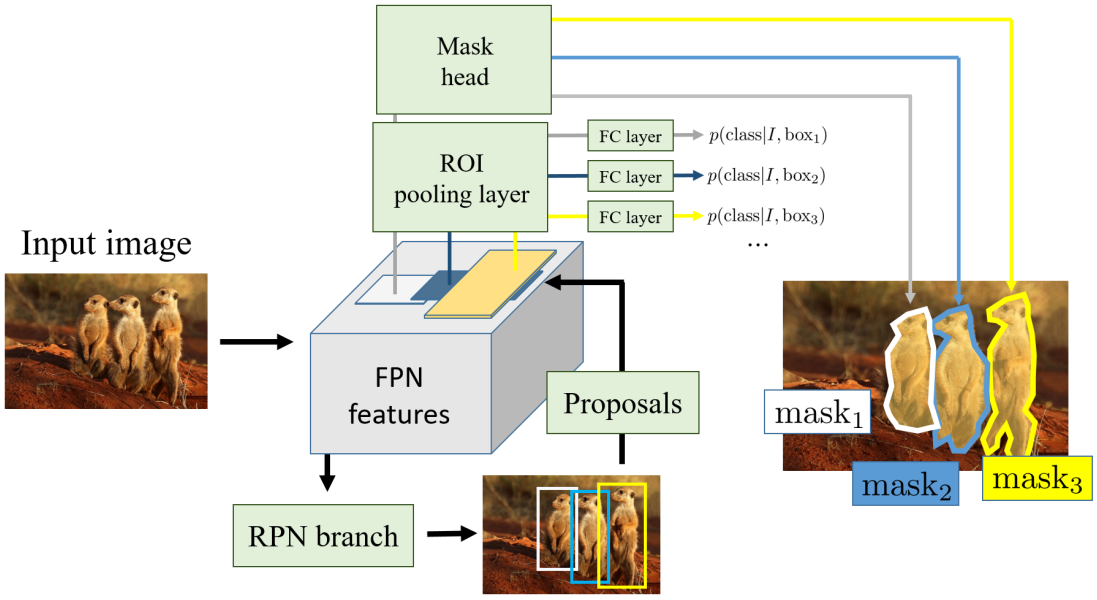


Figure 2.6: Mask R-CNN [He et al. 2017] is a popular example of a Propose & Verify architecture. Given the initial set of box proposals output by Faster R-CNN [Ren et al. 2015], the mask head independently segments the dominant foreground within each proposal crop.

et al. 2016]. This cascaded architecture was enhanced in [Liang et al. 2016] by extending it into a recursive model. Hayder et al. [2017] improved the final segmentation refinement stage by predicting a pixel-wise signed distance from object boundaries instead of a binary segmentation mask.

Mask-RCNN [He et al. 2017] comprises a successful instance segmentation architecture that is a well-tuned evolution of previous works such as [Dai et al. 2016; Liang et al. 2016; Ren et al. 2015]. Like [Dai et al. 2016], it proposes an initial set of candidate windows with Faster R-CNN [Ren et al. 2015]. The verification stage consists of a small network branch which segments the dominant foreground object within each rectangular region (fig. 2.6). One of the biggest drawbacks is the conditioning of each predicted segmentation mask solely on the initial rectangular region. This leads to low-quality segmentations for significantly articulated objects.

Instance Coloring

In order to deal with flexible objects, several works have explored direct labeling of pixels with instance tags. The labels are assigned such that every pair of pixels within the same object instance has to share the same pixel label, while labels of pairs from different instances have to differ. In this thesis, such type of approaches is termed *instance coloring* (IC).

CRF-based methods. A natural way for formalizing the task is by defining a suitable energy that can be optimized with a Conditional Random Field (CRF) [Tighe et al. 2014]. Intuitively, the pairwise CRF potentials should produce high similarity scores for pairs of pixels from the same instance and vice versa. Instead of a CRF, Zhang et al. [2015] leverage depth to determine instance colors. Related to the CRF framework, a deep version of the watershed transform [Beucher 1979] was proposed by Bai and Urtasun [2017].

Metric learning. Several works posed IC as a metric learning problem. More in detail, the goal is to embed pixels into a high dimensional Euclidean space such that embeddings of pixels from the same instance lie close to each other while embeddings of different instances are far apart. Once such embeddings are predicted, an ensuing grouping algorithm (e.g. K-Means) outputs image regions formed by pixels with mutually similar embeddings.

Fathi et al. [2017] identified seed pixels that generated soft instance segmentation masks by evaluating the distance between the seed embedding and the rest of the embedding field. Decreasing the size of the embedding to a single dimension, Newell et al. [2017] introduced a "tagging" network for instance segmentation and grouping of multi-person keypoint detections. A recurrent deep network that implemented mean-shift iterations to refine the instance embeddings was proposed in [Kong and Fowlkes 2018]. De Brabandere et al. [2017] converted the standard pairwise metric learning loss into a discriminative loss function that pulls embeddings shared by an instance towards their common mean while repelling these instance-specific centroids away from each other.

An important downside of the pixel embeddings is related to the convolutional operators that predict them. As a consequence of the translation invariance of the convolution, similar object instances present in different locations of the image cannot be easily distinguished. Our contribution from chapter 7 proposes to break the translational invariance by making the embeddings sensitive to their locations in the image.

2.6 Estimating 3D geometry

Chapter 8 introduces a deep architecture that tackles single-view pose estimation, depth prediction and 3D shape estimation. In this section, an overview of techniques for extracting such 3D information from image data is presented. First, traditional multi-view methods are reviewed followed by discussing the more relevant monocular case.

2.6.1 Multi-view reconstruction

Estimating the 3D structure of a scene is a long lasting problem. Arguably, the most traditional related framework is Structure from Motion (SfM) which addresses recovering the poses of cameras observing a single scene as well as the 3D locations of various points that the scene consists of. The first attempt dates back to the work of Kruppa [1913] who demonstrated that, given a camera pair and correspondences between 5 points, it is possible to compute the relative camera pose together with the 3D point locations up to a scaling factor. Further theoretical foundations for the two-view scenario have been laid by Longuet-Higgins [1981], Ullman [1979], and Faugeras [1993].

The original two-view setup was later extended to multi-view reconstructions. An early multi-view approach from Tomasi and Kanade [1992] proposed a principled formalization of the problem for a weak perspective camera model and derived an optimal solution [Reid and Murray 1994]. However, optimizing the highly non-linear objective function for the more practical perspective

camera case was not addressed. The multi-view reconstruction with perspective cameras was later tackled by a variety of works. Sparr [1996], Pollefeys et al. [1996], and Sturm and Triggs [1996] optimized reconstruction objectives with various kinds of matrix factorization algorithms. However, these minimize algebraic error rather than the geometric reprojection error. The latter is more convenient due to better modeling of the noise coming from the camera sensor. Bundle adjustment [Hartley 1993] (BA) optimizes the reprojection error, but this leads to a non-linear objective susceptible to falling into a poor local minimum if improperly initialized. This has been addressed in [Fitzgibbon and Zisserman 1998] by robustly initializing the camera poses from solutions of the 3-view minimal camera pose problem and later refining the poses with BA.

Several improvements based on iterative enlargement of the set of known camera poses have been later proposed [Beardsley et al. 1997; Koch et al. 1999; Pollefeys et al. 2004]. These works paved the way for large-scale SfM systems capable of reconstructing unconstrained photo collections [Schaffalitzky and Zisserman 2002; Snavely et al. 2006] or urban scenes [Pollefeys et al. 2008]. Further improvements lead to scaling SfM to significantly larger datasets [Agarwal et al. 2009; Frahm et al. 2010] including reconstructions of million-size photo collections [Heinly et al. 2015; Wu 2013]. A recent work by Schönberger and Frahm [2016] constitutes a well-tuned state-of-the-art pipeline dubbed COLMAP.

2.6.2 Single-view reconstruction

SfM systems assume observations from multiple views of a single scene. However, in practice, multiple views are not always available and the geometric properties of an observed scene have to be estimated from a *single-view*. Such scenario is considered in chapter 8 and, in what follows, a review of the monocular viewpoint, depth and 3D shape prediction literature is presented.

Viewpoint estimation

Estimating the SE(3) pose of an object from a single image has a wide range of applications that attracted a large amount of attention from the community. In case correspondences between image pixels and the points on the surface of a predefined 3D model are known, Perspective-n-Point [Fischler and Bolles 1981] solvers can optimally estimate the corresponding pose.

A more practical setup arises by dropping the assumption of manual correspondence annotations between the model and the novel view. The approaches can be split into two groups: (1) Smaller-scale object grasping methods that require high precision with limited robustness to background clutter and intra-class category variations; (2) "In the wild" methods which estimate pose for unconstrained categories such as cars, motorcycles, boats, etc. Since chapter 8 addresses the latter, we give a brief overview of existing approaches of the first type and later delve deeper into the unconstrained viewpoint estimation task.

Object grasping. Early methods, such as [Vacchetti et al. 2004; Gordon and Lowe 2006; Collet et al. 2009; Collet et al. 2011], estimated matches between the novel view and the model using viewpoint-invariant descriptors. The correspondences then lead to a set of constraints for computing the relative pose of the novel view w.r.t. the model. The biggest drawback is a dependence on the local descriptors that often fail for texture-less objects. This was addressed by dense methods. For instance, [Brachmann et al. 2014] utilized the SCORE framework [Shotton et al. 2013] in order to allow each pixel to vote for a possible pose of the parent object. The recent resurgence of deep networks brought several successful solutions such as [Rad and Lepetit 2017].

Unconstrained 3D object detection. Regarding the unconstrained 3D pose estimation, initial approaches were related to the object detection task. More specifically, the detectors contained an explicit viewpoint representation as a component that facilitated viewpoint invariance. Notable examples include [Schneiderman and Kanade 2000] which factorizes an object detector into a collection

of viewpoint-specific detectors. Similar to [Schneiderman and Kanade 2000], [Weber et al. 2000] train a detector for different viewing angles of a human head. A multi-view boosting framework was later introduced in [Torralba et al. 2004] while a similar extension of the implicit shape model [Leibe et al. 2004] was described in [Thomas et al. 2006].

While previous methods mostly contained viewpoint information in order to improve the detection performance, Savarese and Fei-Fei [2007] brought the 6 DoF task into focus. With a limited amount of 3D supervision they built a model of an object geometry based on groups of local invariant features interconnected with mutual homographic transformations. Similarly, [Kushal et al. 2007] designed a viewpoint-aware model based on locally rigid partial surface models. Following the success of DPM [Felzenszwalb et al. 2010b], its 3D extensions were later proposed in [Fidler et al. 2012; Pepik et al. 2012; Zhu et al. 2014].

Deep viewpoints. Similar to other branches of computer vision, convolutional neural networks demonstrated compelling performance for estimating pose. Tulsiani et al. [Tulsiani and Malik 2015] outperformed shallow alternatives (3D DPM) by a significant margin. Their network was fully supervised with Pascal3D [Xiang et al. 2014], a large dataset with manual viewpoint and keypoint annotations for 12 rigid object categories. In [Su et al. 2015], the amount of supervision was increased by generating a synthetic dataset of rendered CAD models from ShapeNet [Chang et al. 2015], bringing quantitative improvements as a result. Xiang et al. [2014] and Tulsiani and Malik [2015] convert the pose regression task into a classification one by quantizing the pose annotations into a set of rotation bins. Differently, Pavlakos et al. [2017] recovered the pose as a solution of an optimization problem that relies on semantic keypoint detections. In more detail, the method first detects semantic keypoints followed by finding an optimal camera matrix that projects the set of 3D keypoints from a CAD model into their corresponding detections in the image.

Aligning CAD models. Several works explored direct alignment of accurate 3D CAD models to their corresponding depictions for the purpose of parsing

Figure 2.7: Unsupervised viewpoint learning was addressed in [Sedaghat and Brox 2015] where a set of car videos was first reconstructed with an SfM algorithm followed by a collective alignment step utilizing a 3D variant of the HoG descriptor. Image taken from [Sedaghat and Brox 2015].

3D indoor scenes [Lim et al. 2013]. The first shallow methods [Lim et al. 2013; Aubry et al. 2014] processed object detections by transferring the pose from the most similar view retrieved from the set of 3D CAD model renders. Other alternatives transferred the geometric information from a larger collection of renders [Huang et al. 2015]. Deep solutions were proposed in [Aubry and Russell 2015; Bansal et al. 2016]. Some works went beyond mere 3D model retrieval and allowed the aligned model to deform in order to better match the depictions [Massa et al. 2016].

Weakly supervised viewpoints. More related to our setup from chapter 8, Sedaghat and Brox [2015] learned a single-view viewpoint estimator from videos of an object category without additional supervision. First, each object-centric video was converted into a scene point cloud using SfM. The point clouds were then described with a hand-engineered 3D variant of HoG and aligned with a graph-optimizer. Given the set of video frames annotated with the correctly aligned camera matrices, a single-view pose estimation CNN was trained.

Depth estimation

The depth estimation task aims at predicting the distance of every image pixel from the camera plane. While the multi-view setup leads to stereo vision [Lucas and Kanade 1981], which is well-researched and leads to a clear-cut optimization

problem, the single view case is inherently ill-posed. Earliest methods addressed the task by analyzing the gradual variation of shading in the image [Horn 1970]. Here, an image was factored into a product of shading, which varied smoothly in the image, and reflectance, that was inferred from sharp edges. The shading factor was later decomposed into a product of illumination and shape which depends on a depth map [Barron and Malik 2015], achieving single-view depth estimation. Several other cues including texture [Witkin 1981], defocus [Favaro and Soatto 2002] or silhouette [Prasad et al. 2006] were also exploited. One of the biggest drawbacks of these methods is their limited applicability to surfaces with roughly uniform color or texture.

Assuming the knowledge of the parent object category of the reconstructed surface brought strong priors for improving the quality of reconstructions. [Nandy and Ben-Arie 2000] proposed a 3D face reconstruction system, termed shape-from-recognition, that relied on face part detectors. Shape-from-knowledge, an improvement from [Nagai et al. 2002], learned correlations between image data and the corresponding 3D models using a Hidden Markov Model. In a more generic fashion, [Hassner and Basri 2006] transfered depth from a database of depth-annotated image patches.

Less constrained depth predictors were later proposed. An initial attempt by Torralba and Oliva [2002] exploited sizes of known objects in a scene in order to estimate the mean image depth. The first full-fledged depth estimator for unconstrained indoor scenes was proposed in [Delage et al. 2006]. Saxena et al. [2008] later introduced a MRF framework that was applicable also to outdoor scenes. Assuming a scene consists of a ground plane and vertical walls, Hoiem et al. [2005] allowed to "pop-up" a visual element from an image plane. A dataset of laser scans and an improved MRF method that identifies 3D locations and orientations of local planes was later described in [Saxena et al. 2009]. Ladicky et al. [2014] argue that the non-linear scaling of depth with the size of perceived objects introduces harmful biases to the predictions of depth regressors. They

alleviate this problem by detecting visual elements that appear at a fixed canonical depth, inherently achieving depth prediction. In a non-parametric manner, [Karsch et al. 2014] warps views with known depth onto a novel test image.

Deep depth estimation. The first deep depth estimator was introduced by Eigen et al. [2014]. Their two-branch architecture that combined coarse and fine predictions outperformed the existing shallow alternatives by a significant margin. The latter was later outperformed by a simpler architecture [Li et al. 2015a] combined with a Conditonal Random Field (CRF) refiner. Cao et al. [2017] converted the depth regression into a classification task in order to tackle the empirically observed inability of deep networks to accurately regress depth values. Laina et al. [2016] adopted the successful residual architectures [He et al. 2016b] together with proposing novel losses and upsampling layers. All the aforementioned deep methods require pixel-perfect ground truth from a depth sensor or a laser scanner.

Unsupervised deep learning of depth was recently addressed in several works. Garg et al. [2016] demonstrated the possibility of learning monocular depth predictors by using disparity constraints between image stereo pairs. Additional constraints improving the quality of the depth estimates were proposed in [Godard et al. 2017]. Less supervised, Zhou et al. [2017] designed a depth and egomotion predictor that is trained solely by observing video sequences. They optimize a photometric loss computed between video frames aligned with an inter-frame warp field obtained by combining the predicted depth and egomotion.

Monocular shape prediction

The first single-view shape estimation pipeline can be traced back to the work of Roberts [Roberts 1963] that describes a computer program which, by means of projective geometry, displays a 3D model of an object depicted in a 2D photograph. The ensuing approaches were able to model non-rigid objects as a composition of parts represented with geometric primitives such as superquadrics [Pentland 1986] or generalized cones [Nevatia and Binford 1977]. Lowe [1987]

aligned a 3D model with its projection in an image using low-level edge-based cues. Model-based 3D reconstruction was further addressed in [Koller et al. 1993] for object tracking.

Faces. There is a large body of work that specializes to two domains of 3D shapes - faces and human bodies. For face reconstruction, the seminal work of Blanz and Vetter [1999] introduced the concept of representing shape deformations as linear combinations of a shape basis. Later, the idea was adopted for the generic Non-rigid Structure From Motion (NRSfM) task in [Bregler et al. 2000]. Following the success of [Blanz and Vetter 1999], many methods for single view face reconstruction were proposed. Briefly mentioning several examples [Hassner and Basri 2006; Kemelmacher-Shlizerman and Basri 2011; Hassner 2013; Richardson et al. 2017], in what follows we discuss other, category-agnostic approaches as these are the the main focus of the thesis.

Data-driven approaches. Overcoming the absence of model-annotated datasets for single-view reconstruction, initially, other types of annotations were utilized for performing data-driven 3D shape prediction. Vicente et al. [2014] leveraged category keypoint annotations in order to reconstruct objects in the Pascal VOC dataset [Everingham et al. 2011]. Similarly, Kar et al. [2015b] reconstructed Pascal VOC with NRSfM, but later additionally supervised a single-view 3D reconstruction system with the obtained 3D models.

Deep predictors with synthetic supervision. The success of deep networks in combination with the emergence of large 3D CAD model libraries [Chang et al. 2015] set off a wave of new deep monocular 3D shape predictors. These differ in the way the output 3D shape is represented. Arguably, the most popular 3D representation is a probability field over a voxel grid. For instance, 3D-R2N2 [Choy et al. 2016b] comprises a 3D equivalent of the standard convolutional architecture enriched with additional LSTM layers. Girdhar et al. [2016] trained a deep autoencoder of voxel data in order to learn a global 3D representation suitable for single-view shape predictions. Generative advesarial networks [Goodfellow

et al. 2014] have been adopted for 3D data in [Wu et al. 2016]. Despite the ability to accurately model multi-modal 3D shape distributions, a significant downside of voxel occupancy grids is their large memory consumption. OctNet [Riegler et al. 2017] alleviates this by exploiting the sparsity of the output and representing only the occupied cells.

Several different types of encoding a 3D shape recently emerged. Surface predictors have been proposed in [Groueix et al. 2018; Sinha et al. 2017; Häne et al. 2017]. Similar to our approach, [Fan et al. 2017; Yang et al. 2018] generate point clouds approximating the surface of the observed shape. Fan et al. [2017] introduced a variational autoencoder for point cloud generation while [Yang et al. 2018] iteratively folds a point cloud around the 3D surface of an object. Similar to point cloud prediction, [Tatarchenko et al. 2016] predicts multi-view depth maps that are later merged into a complete 3D model.

Weakly-supervised shape prediction. While all the aforementioned methods require expensive supervision in form of hand-designed CAD models, Rezende et al. [2016] proposed a deep network capable of learning the 3D structure of object categories from lower levels of supervision. Similarly, silhouettes were exploited as another source of incomplete supervision in [Wiles and Zisserman 2017; Tulsiani et al. 2017]. Nevertheless, viewpoint knowledge is still assumed in both cases. The system from chapter 8 does not require manual annotations as it is only cued by observing videos of objects.

3

Generalizing Semantic Part Detectors Across Domains

This work was published as a chapter [Novotny et al. 2017d] in Csurka, G., ed.: Domain Adaptation in Computer Vision Applications. Springer 2017. [Csurka 2017]

The manuscript is an extension of a work presented as a *poster* presentation at the British Machine Vision Conference, 2016 [Novotny et al. 2016a].

Generalizing semantic part detectors across domains

David Novotny[1,2], Diane Larlus[2], and Andrea Vedaldi[1]

[1] Visual Geometry Group, Department of Engineering Science, University of Oxford
{david,vedaldi}@robots.ox.ac.uk

[2] Computer Vision Group, NAVER LABS Europe
diane.larlus@naverlabs.com

Abstract. The recent success of deep learning methods is partially due to large quantities of annotated data for increasingly big variety of categories. However, indefinitely acquiring large amounts of annotations is not a sustainable process, and one can wonder if there exists a volume of annotations beyond which a task can be considered as solved or at least saturated. In this work we study this crucial question for the task of *detecting semantic parts* which are often seen as a natural way to share knowledge between categories. To this end, on a large dataset of 15,000 images from 100 different animal classes annotated with semantic parts, we consider the two following research questions: i) are semantic parts really visually shareable between classes? and ii) how many annotations are required to learn a model that generalizes well enough to unseen categories? To answer these questions we thoroughly test active learning and DA techniques, and we study their generalization properties to parts from unseen classes when they are learned from a limited number of domains and example images. One of our conclusions is that, for a majority of the domains, part annotations transfer well, and that, performance of the semantic part detection task on this dataset reaches 98% of the accuracy of the fully annotated scenario by providing only a few thousand examples.

1 Introduction

Image understanding has recently progressed dramatically, primarily fueled by the availability of increasingly large quantities of labeled data. It was only with the introduction of large-scale resources such as the ImageNet dataset [8] that deep learning methods were finally able to realize their potential. However, it is unclear whether manual supervision will be able to keep up with the demands of more sophisticated and data-hungry algorithms. Recent initiatives such as the Visual Genome [26], where millions of image regions are labeled with short sentences and supporting bounding boxes, go well beyond standard datasets such as ImageNet and offer new terrific research opportunities. At the same time, however, they raise the obvious question: *when is supervision enough?*

The idea that limitless manual supervision is impractical has motivated research in areas such as unsupervised learning or learning from off-the-shelf resources such as the Web. While these are important research directions, such approaches go to the other extreme of avoiding manual supervision altogether. In this work, we take a *pragmatic approach* and start from the assumption that explicit manual supervision is currently the most effective way of training models. However, we also ask whether there is a limit on the amount of supervision which is actually required and hence of the amount of the data that needs to be annotated.

While answering this question in full generality is difficult, we can still conduct a thorough analysis of this question in representative cases of general interest. In this paper, we focus on the problem of *recognizing and detecting semantic parts in categories*, such as animal eyes (see Figure 1), because semantic parts are highly informative, and, importantly, *semantically shareable* (*e.g.* both monkeys and snakes have faces which, despite important differences, are broadly analogous). In fact, one of the key motivations for looking at semantic parts in computer vision is to transfer the structure of known objects to novel ones, for which no supervision is available. However, an important practical question, which is often neglected, is whether semantically shareable parts are also *visually shareable*, in the sense of being recognizable in novel objects with no or limited further supervision. This assumption has never been challenged beyond a few categories or narrow domains. In this paper, we conduct a careful investigation of *part transferability across a large target set of visually dissimilar classes*. We investigate the two key aspects of transferability under bounded supervision : (i) learning parts from a limited number of example images and (ii) applying known parts to new, unseen domains.

For the first problem, we consider an *active learning* (AL) scenario, where images are annotated with their visible semantic parts in turn, studying which images should be chosen and how many are needed to saturate performance. Working in this AL scenario, we look at part transferability as a *Domain Adaptation* (DA) problem. Differently from the typical transductive learning scenario of DA, where algorithms leverage unlabeled data of the target domain to adapt a source predictor, our goal is to learn semantic part detectors that can be directly applied to novel classes that were never seen before, *i.e.* the model is expected to transfer well to new domains even *without having access to samples from those domains*. This problem has been seldomly studied and is sometimes known as the *domain generalization* (DG) problem [34,16,14].

Another specificity of our work is that, while the majority of the existing DG methods focuses on the standard classification task where the model predictions consist of a discrete categorical label, here we consider DG in the context of a detection task, where predictions are defined as a set of 2D part locations labeled with a real value which represents a confidence score. We thus propose extensions of existing active and transfer learning methods, which were initially designed for domain adaptive classifiers, into methods suitable for the detection task in order to make them applicable to our setting.

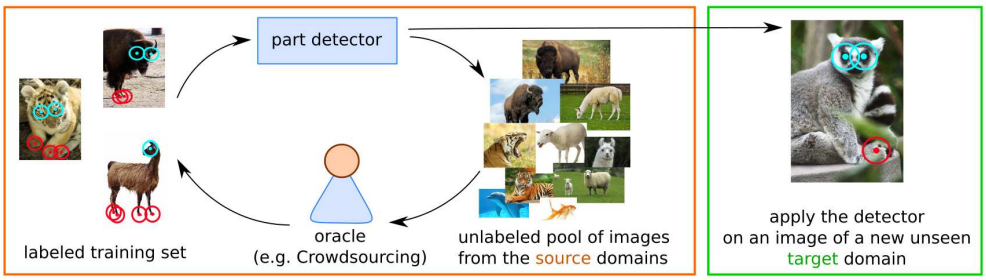


Fig. 1. Overview of the semantic part detector generalization problem. We investigate the ability of semantic part detectors to transfer between different domains, and study the minimal amount of supervision that is required for this task.

In this chapter, we address the DG problem of detecting semantic parts on new unseen target categories by using an efficient ensemble of detectors which is optimized for generalization to new classes and that, at the same time, can be used to guide active learning. We conduct a thorough empirical evaluation of all these research questions and provide insights on how subsets of images may be selected for labeling, and how many such labels may be required to perform well on unseen categories. We also consider several domain adaptation scenario where a small number of annotations are available for the target classes. This chapter extends the work presented in [35].

The rest of this chapter is organized as follows. Section 2 provides a brief overview of related work. Section 3 describes the employed active and transfer learning methods. The dataset considered for our analysis is described in Section 4. Section 5 provides the experimental evaluation of part transferability together with a benchmark of the proposed methods. Section 6 concludes the work.

2 Related work

Domain adaptation. DA seeks to learn predictor functions that are able to account for a shift between the distributions of the source and target data domains. Since the seminal paper of Saenko *et al*. [43], DA has been applied to computer vision by learning feature transformations [18,17,13,48,2], or by adapting the parameters of the predictor [22,60]. Only a few papers consider DA from more than one source domain [49], and most from at most a few, while in our work we consider 50 source domains. Sometimes, the source domains are not given a priory, but discovered implicitly [21].

All these approaches formulate DA as *transductive* learning, for which they require unlabeled samples from the target domain. This is a fundamental difference from our case, where *no target samples are available*, also known as domain generalization (DG) [34,16,14] or predictive DA [62].

Active learning. The goal of active learning (AL) is to reduce the annotation costs by deciding which training samples should be annotated. Each annotation is associated with a cost [6] and the goal is to obtain the best performance within a budget. Many data selection strategies have been proposed, based on uncertainty and entropy [50], used in [25,37,23,7], or diversity and representativeness [24]. The work of [56] estimates a cost of different types of annotations and then modulates an expected risk function while the strategy of [42] annotates as many examples in each image as possible. [36] leverages additional information from different annotation types to improve the convergence properties of the active learner. Only few works have jointly looked at transfer learning and active learning [58,59,38,3,44] as we do here, and none of them for computer vision tasks. Moreover, the transfer learning components of these works approach the transductive DA task whereas we focus on domain generalization.

Related transfer learning problems. Zero-shot learning, *i.e.* the task of recognizing a category with no training samples, is often tackled by explicitly learning classifiers that are transversal to object classes. This can be done by modeling semantic relatedness [40,39], or by transferring other knowledge such as materials [54,5], segments [47], parts [51,10] or attributes [29,11]. However, these works consider only a small number of classes and *assume* that primitives such as parts transfer visually, whereas here we explicitly question this assumption. Fewer works consider transfer learning for localization as we do; these include the works of [33,19,28,9] that transfer bounding box information using the ImageNet hierarchies; the method of [20] that transfer object detectors from seed classes; and [1] which transfers detectors assuming a limited number of annotated examples in the target domain. Differently from such works, we do not transfer whole objects, but individual keypoints, and we do so between very diverse classes. Transferring keypoints was explored in [63], which detects facial landmarks using a deep multi-task learning framework, while [52] induce pose for previously unseen categories.

3 Methods

This work tackles the semantic part detection task, and experiments focus on the dataset introduced in [35], where parts are annotated with keypoints. We first describe the keypoint detector used to locate semantic parts in images. Second, following our AL scenario, we describe how uncertainty sampling can be defined in our particular setting to sample in turn the images to be annotated. Finally, we describe how the fact that we have many source domains (*i.e.* source categories) to sample from can be leveraged in order to combine several detectors that are then applied to the unseen target classes.

Keypoint detector. As our baseline keypoint detector, we use the state-of-the-art method proposed by Tulsiani and Malik [53]. This architecture uses the convolutional layers from the very deep network (VGG-VD) of [46], followed by a linear regressor, that outputs a heat-map expressing the probability for

a keypoint to be centered at a particular location. To improve accuracy, they linearly combine the outputs of a coarse-scale (6×6) and a fine-scale (12×12) network.

As our analysis requires frequent retraining of the model, we adapt the faster 6×6 network of [53] to output finer-scale 12×12 cell predictions. To do so, following [30], we append a bilinear upsampling layer to the final keypoint regressor convolutional layer and sum the resulting upsampled heat-map with a finer-scale convolutional heat-map derived from the pool4 VGG-VD layer. The recombined heat-map is finally followed by a sigmoid activation function. The resulting architecture allows for multi-scale end-to-end training while increasing overall training and testing speed by a factor of 3.

Active learning. The goal of AL is to select a small subset of images for annotation while maximizing the performance of the final predictor. Let U be the set of all available images. The algorithm starts with a pool $L_0 \subset U$ containing $|L_0| = 50$ randomly-selected images from U and collects the corresponding annotations. Then, for every active learning round t ($t = 1, 2, \dots$) the algorithm alternates training a CNN keypoint detector using all annotations in L_k for $k < t$, and collecting annotations for A more images in order to build L_t. For the latter, all non-annotated images in U are examined by the sampling component of the AL algorithm and the A *most informative ones* are selected for annotation.

The standard criterion to select informative images is to pick the ones which leave the current predictor uncertain, also called *uncertainty sampling*. However, while uncertainty is easily defined in classification tasks where the goal is to predict a single label per image, it is not obvious how to do so for keypoint prediction where the predictor produces a score for every image location.

We propose to do so as follows: let $p(y = +1|x, u) = \Phi(x)_u$ be the probability of finding a keypoint at location u in image x as computed by the CNN Φ (unless otherwise specified, we assume that the CNN is terminated by a sigmoid function). The uncertainty score is then given by

$$1 - 2 \times |\max_u p(y = +1|x, u) - \frac{1}{2}| \tag{1}$$

Intuitively, when the model is certain, either (i) there are no part keypoints in that image and $\max_u p(y = +1|x, u) \approx 0$, or (ii) there is at least one keypoint and then $\max_u p(y = +1|x, u) \approx 1$.

Transfer learning by auto-validation. Our problem differs from standard AL in that, as in DA, the target classes are *shifted* compared to the source ones. Furthermore, differently from the standard transductive learning setting in DA, our aim is to learn a "universal" part detector that *generalizes* to unseen target classes without further training. Compared to more common machine learning settings, we can leverage the fact that the source data is split in well-defined domains with a large domain shift and train a set of domain-specific detectors which are later recombined using an ensembling method, assigning higher weights to the models with better generalization capabilities. In other

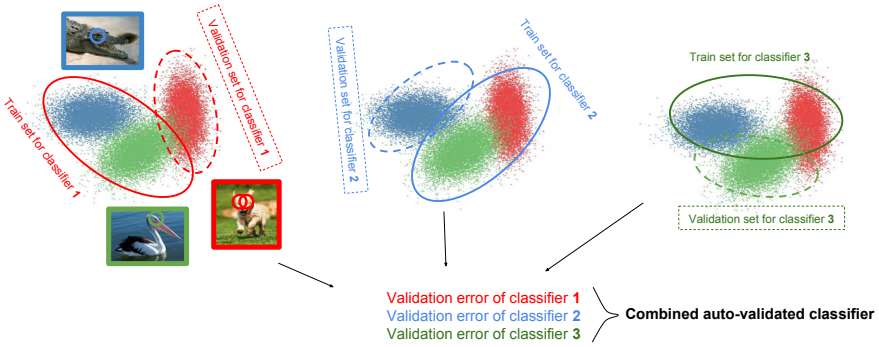


Fig. 2. Auto-validation of the keypoint transfer process across domains. The detectors from the source domains that are more invariant to domain shift have more impact on the final decision on the target domains. Invariance is evaluated by measuring cross-validation error on a held-out set.

words, the detectors from the source domains that are more invariant to the underlying domain distributions will be given a larger impact on the final decision on the target domains.

In more detail, we do so by using an *auto-validation* procedure. Let $D = \{d_1, ...d_N\}$ be a set of source domains (the object categories in our case) and let $\delta \subset D$ be a subset of the source domains. For each possible δ we train a domain-specific (or group of domains specific) part predictor Φ_δ. The set of all Φ_δ, is then recombined into a single domain-invariant model by utilizing the ensembling method of Krogh *et al.* [27]. The recombination procedure from [27] defines the final predictor as a weighted sum of the individual domain-specific models $\sum_{\delta \subset D} \alpha_\delta \Phi_\delta$, where α_δ are the weights of the domain-specific model Φ_δ which favor a diverse set of models with good generalization capabilities. In practice, the ensemble weights are obtained by minimizing the following objective function [27]:

$$\begin{aligned} \underset{\{\alpha_\delta | \delta \subset D\}}{\arg\min} \quad & \sum_{\delta \subset D} \alpha_\delta E_\delta + \sum_{(\delta,\delta') \subset D \times D} \alpha_\delta C_{\delta\delta'} \alpha_{\delta'} - \sum_{\delta \subset D} \alpha_\delta C_{\delta\delta} \\ \text{s.t.:} \quad & \alpha_\delta \geq 0, \ \forall \delta \subset D, \qquad \sum_{\delta \in D} \alpha_\delta = 1 \end{aligned} \tag{2}$$

where $C_{\delta\delta'} = E_{xuv}[\Phi_\delta(x)_{uv} \Phi_{\delta'}(x)_{uv}]$ is the the cross-correlation matrix of the response (heat-maps) of the different models and measures how much different models agree in their predictions. E_δ is the cross-validation error computed on a set of held-out samples that were unseen during the training of Φ_δ. Similar to [27], we define E_δ as the error of Φ_δ on the set of off-domain samples $D - \delta$. This process is illustrated Figure 2.

It is worth mentioning that, although in [27] all the ensemble models were optimized on distinct sets of data samples, the data samples were drawn from the same data distribution. This leads to a set of models that produce similar

predictions which violates one of the main requirements of ensembling which is the fact that the ensemble members should disagree. On the contrary, our choice of creating train-test data splits $(\delta, D - \delta)$ which correspond to diverse animal domains leads to a set of very distinct models with a large degree of disagreement.

The original method from [27] was designed to recombine predictions of independently trained shallow neural networks. Here, we adapt the original method so the different keypoint detectors Φ_δ share weights for the early layers of the network, thus removing the costly requirement of re-optimizing a large number of CNN parameters on every training set of samples δ. In practice, we propose to decompose the CNN architecture as $\Phi_\delta = \phi_\delta \circ \phi_0$, where ϕ_0 is the same for different δ and only ϕ_δ is specific to δ. More precisely, ϕ_0 includes all learnable parameters of the original VGG-VD layers up to conv5_3 and ϕ_δ comprises the final convolutional filter terminated by the sigmoid layer that outputs part detector responses specific to the model trained with the training samples from δ.

The optimization of ϕ_δ and ϕ_0 is easily implemented using SGD: given a data sample x, for all δ in parallel, either ϕ_δ or the cross-validation error E_δ are updated, depending on whether $x \in \delta$. The cross-correlation matrix C is estimated using all samples irrespective of their origin. To ensure numerical stability we add a small constant λ to the diagonal of C ($\lambda = 0.1$ in all experiments). Once the optimization of ϕ_δ, E_δ and C completes, the coefficients α_δ are obtained by solving Eq. (2).

Another advantage of training an ensemble of detectors Φ_δ is that their lack of agreement on the training data can replace uncertainty sampling in guiding the AL process. We implement this query-by-committee [45] criterion (QBC) following [27]: Given a pixel u in a test image x we assess the disagreement between pixel-wise predictors $\Phi_\delta(x)_u$ by evaluating the *ensemble ambiguity*:

$$\begin{aligned} a(x,u) &= \sum_{\delta \subset D} \alpha_\delta (\Phi_\delta(x)_u - \bar{\Phi}(x)_u)^2, \\ \text{where} \quad \bar{\Phi}(x)_u &= \sum_{\delta \subset D} \alpha_\delta \Phi_\delta(x)_u. \end{aligned} \tag{3}$$

Similar to uncertainty sampling (section 3) we label each image x with a disagreement score $A(x)$ by max-pooling over the pixel-wise ensemble ambiguities, *i.e.* $A(x) = \max_u a(x,u)$. During the labeling stage of active learning, samples with highest $A(x)$ are added first.

4 ImageNet Animal Parts

A thorough evaluation of the transferability of parts requires a suitable dataset with a large enough number of classes. Unfortunately, datasets that have keypoints or part-level annotations either consider a handful of classes, such as the PASCAL Parts [4], or are specialized to a narrow set of domains, focusing only

Fig. 3. ImageNet Animal Parts dataset. It contains 100 animal categories from the ImageNet dataset and selected part annotations (the figure shows one annotated example per class). We split the classes into 50 source and 50 target domains

on birds [57], faces [32,63], or planes [55]. Datasets with more categories, such as the Visual Genome [26], do not contain systematic part annotations. In a preliminary version of our work, we introduce the ImageNet Animal Parts dataset[3] [35], following a procedure that we remind here.

Instead of collecting a new set of images, we build on top of the existing ImageNet dataset [8] where the classes are organized in a semantic hierarchy, induced by WordNet [12]. This provides a natural basis to study the semantic structure of this space. A very significant challenge with annotating many parts for many object categories is of course the very large cost, thus, trade-offs must be made.

Here, the singly most important aspect for experimentation is to label a sufficiently large space of categories. This space should also contain a mix of similar and dissimilar categories. Furthermore, the same parts should ideally apply to all categories. Here we select for experimentation 100 classes (see Figure 3) of the 233 classes in the "vertebrate" subtree of the ImageNet ILSVRC [41]. For each

[3] The dataset can be accessed at: `http://www.robots.ox.ac.uk/~vgg/data/animal_parts/`

class we annotate two parts. The first one, *eyes*, exist in all selected animals. The second one, *feet*, exist in a large subset of these (mammals and reptiles but not fish). Beyond their semantic shareability (visual shareability is an assumption that we verify in our work), these parts were selected because they are easily understood by annotators from crowd sourcing platforms, and they can satisfactorily be annotated with keypoints as opposed than by drawing bounding boxes or regions. Both properties were instrumental in collecting a large dataset of part annotations in a reasonable time and budget. While limited, these annotations are sufficient to demonstrate the principles of our analysis. We collected annotations for about 150 images per class, annotating 14711 images in total.

5 Experiments

Experimental protocol. The set of 100 domains (*i.e.* animal classes) is split into 50 source domains and 50 target domains as follows. To achieve uniform coverage of the animal classes in both sets, we first cluster the 100 classes into $K = 50$ clusters using their semantic distance and spectral clustering. The semantic distance between two classes d and d' is defined as $|r \to d \cap r \to d'| / \max\{|r \to d|, |r \to d'|\}$, where $|r \to d|$ is the length of the path from the root of the hierarchy to class d. Then, each cluster representative is included in the target set and the complement is included in the source set. Furthermore, images in each class are divided into a 70/30 training-testing split, resulting in four image sets: source-train, source-testing, target-train and target-test. As common practice [61,31], keypoint detections are restricted to ground-truth bounding boxes for all evaluation measures.

Evaluation measures. We evaluate keypoint detection using two standard metrics [61]: PCK and APK. In **PCK**, for each ground truth bounding box, an algorithm predicts the single most confident keypoint detection. This detection is regarded as true positive if it lies within $\alpha \times \max\{w, h\}$ of the nearest ground truth keypoint, where w, h are the box dimensions and $\alpha \in \langle 0, 1 \rangle$ controls the sensitivity of the evaluation measure to misalignments. For **APK**, keypoints are labeled as positive or negative detections using the same criterion as PCK and ranked by decreasing detection scores to compute average precision. In all of our experiments we set $\alpha = 0.05$ for the eyes, that are small and localized, and $\alpha = 0.1$ for the feet which are more difficult to annotate with a keypoint.

5.1 Baseline detector.

We validated our baseline keypoint detector by comparing it to the original $6\times6+12\times12$ model of [53]. Our implementation of the $6\times6+12\times12$ architecture achieves 61.1% PCK on the PASCAL VOC rigid keypoint detection task – a comparable result to 61.5% PCK reported in [53].

We also experimented on our AnimalParts dataset. Our baseline keypoint detector was compared to the 6×6, 12×12 and $6 \times 6 + 12 \times 12$ models of [53].

Table 1. Validation of the keypoint detector on the AnimalParts dataset.

eye $\alpha = 0.05$; foot $\alpha = 0.1$	**PCK**			**APK**		
Method	eye	foot	**mean**	eye	foot	**mean**
[53] - coarse-scale - 6x6	37.0	73.6	55.3	17.6	58.4	38.0
[53] - fine-scale - 12x12	**81.7**	73.9	**77.8**	**75.4**	64.0	**69.7**
[53] - 6x6+12x12	73.6	**77.3**	75.5	55.9	**67.6**	61.8
Proposed architecture - 6x6 upsample	80.2	74.1	77.2	73.1	59.2	66.2

Each keypoint detector was trained on the *source-train* split and the mean PCK and APK over *target-test* images are reported. Table 1 shows that our modified 6×6 upsample architecture outperforms the low resolution 6×6 by a significant margin while being comparable to 12×12. The only case in which 6×6 performs worse is feet APK (but not PCK); however 6×6 upsample is roughly 3 times faster during both the training and test stages than 12×12. Note also that $6 \times 6 + 12 \times 12$ is not competitive for eye detection while being much heavier. In conclusion, the 6×6 upsample architecture performs on par with state-of-the-art while being much faster than alternatives.

5.2 Visual shareability of parts

In this section we challenge the idea that parts are visually shareable across different classes and that, therefore, it suffices to learn them from a limited number of classes to understand them equally well in all cases. Fig 4 shows part detection performance for individual classes, for different configurations that we discuss below.

Learning from a single class. We first look at the individual target class detection results when learning from annotated samples from the same class (green bar plots). We see that the difficulty of detecting a certain part strongly depends on the specific class. For example, owl's eyes have 100% PCK, whereas turtle's eyes have 38.1% PCK. We then compare with two other training sets of identical size: i) with the nearest class (NC - red bar plot) according to the semantic measure, and ii) with the farthest class (FC - blue bars). As expected, we verify that NC outperforms FC by 20.9% PCK in average, which translates into 39 classes out of 50 for the eyes, and 32 out of 37 for the feet. This demonstrates the relevance of the semantic distance for cross-domain transfer. In average NC still performs 26.9% below training with the target class itself. Next, we consider transferring from more classes.

Increasing the size of the training set. We compare the performance of detectors when these are trained with larger subsets of the data: i) using all classes available (*i.e.* the source and target domains, purple bar plots), and ii) using only the source domains (that does not contain the target class, orange bars). We note several factors. First we observe that using all classes improves performance compared to training only for the target class in average for feet, but not for eyes that perform very well already. Then, we observe that in 61% of

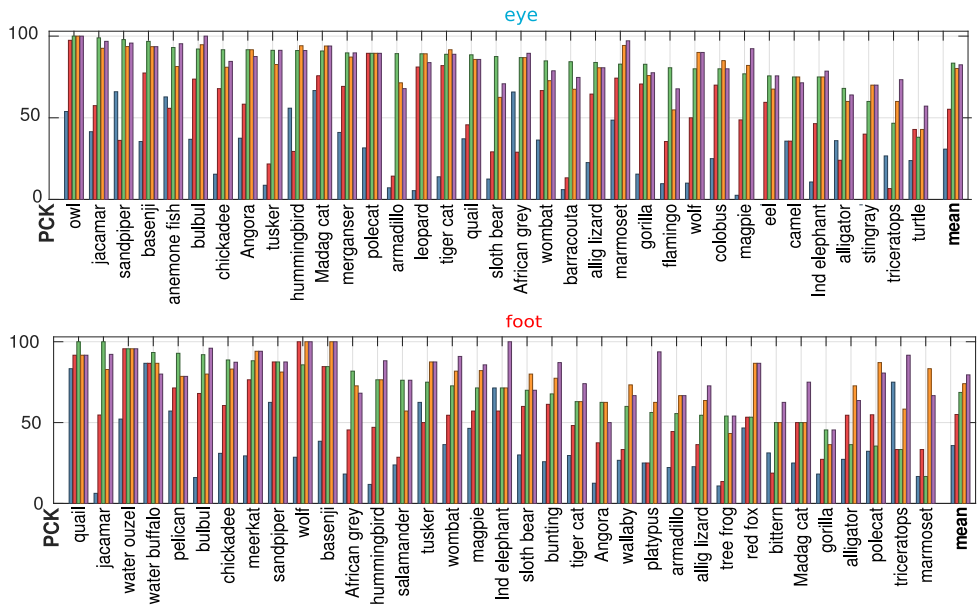


Fig. 4. Relative difficulty of part detection and part transfer. Part detection performance for eyes (top) and feet (bottom) for a subset of the target classes, where the detector has been trained using either: the farthest class (in semantic distance), the nearest class, the same class, the source classes, all source and target classes. Classes are sorted by increasing difficulty.

the cases, learning a part from source classes alone or adding the target classes changes PCK by less than 7%. Hence, if parts are learned form a sufficiently-diverse set of classes, they can be expected to transfer satisfactorily to novel ones as well. In average, training from the source classes only (transfer scenario) is only 2.2 PCK below training from the full set of classes for eyes, and only 5.5 PCK below for feet.

5.3 Active-transfer learning

In the previous section we looked at how well parts transfer from known (source) classes to new (target) classes. Here we investigate how many source images need to be annotated in the source domain. In order to answer this question, we adopt the active-transfer learning framework of Section 3 and we monitor the performance of the detector on the target classes as more annotated images for the source classes become available. The overall performance is summarized by plotting the attained mean PCK (solid lines) and APK (dashed lines) as a function of the number of labeled source images (Figure 5). We compare three methods: AL by random sampling (RS), AL by uncertainty sampling (US), and network ensemble with AL by query-by-committee (ensemble+QBC).

Implementation details. The initial pool contains $|L_0| = 50$ randomly-selected images and 300 additional images are added at every active learning round. For

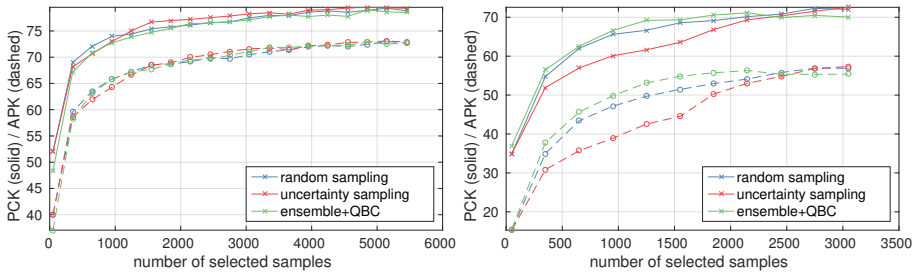


Fig. 5. Active-transfer learning for the eye (left) and foot (right) parts. We show PCK (solid lines) and APK (dashed lines) as a function of the number of labeled examples obtained with random sampling, uncertainty sampling and the network ensemble with query-by-committee sampling (ensemble+QBC) methods.

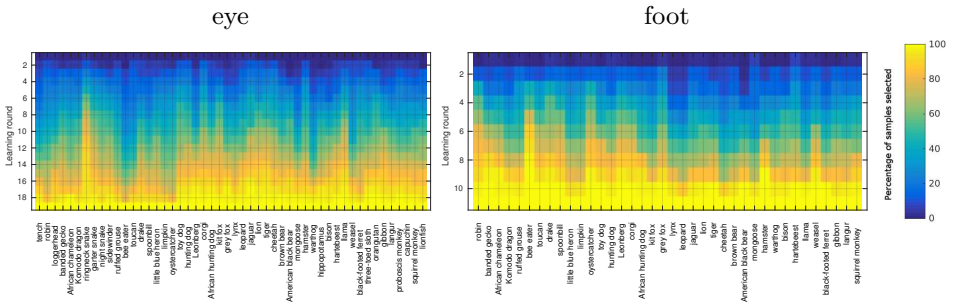


Fig. 6. Comparative domain importance on eye and foot detection. For each source class we show how many more images the AL selects at each round by ensemble+QBC.

pre-training the CNN we first remove all the images of the vertebrate subtree from the original ILSVRC12 dataset and then continue according to the protocol from [46]. In each active learning round, a pre-trained CNN is fine-tuned for 7 epochs, lowering the learning rate tenfold after epoch 5 (this was verified to be sufficient for convergence). Learning uses SGD with momentum and mini-batch size of 20. Mini-batches are sampled to guarantee that all the training classes are equally represented on average (re-balancing). Momentum, weight decay, and initial learning rate were set to 0.9, 0.005, and 0.0003 respectively. All parameters were validated by splitting the source domains in half. For DA by auto-validation, the set D of possible source domains was obtained by regrouping the 50 source domains into 3 super-domains by clustering them using their semantic similarity (hence an ensemble of 7 CNNs is learned). All experiments are repeated 4 times and averages are reported.

Results. First, we observe that US performs slightly better than the other two algorithms on the eye, but is substantially outperformed by RS and ensemble+QBC on the foot class. We believe this because the network is typically most uncertain about images that happen to not contain any part instance, which is fairly frequent with animal feet as they tend to be occluded or truncated. On the

contrary, RS is not affected by this problem. Ensemble+QBC performs as well as RS on the eye part and noticeably better on the foot part. This indicates that guiding active learning using the QBC criterion is more robust than US. The fact that the ensemble+QBC method performs similarly to the others on the eye class is likely due to the fact that there is less visual variability in eyes than feet and therefore all classifiers in the ensemble are similar, with poorer generalization [27]. Ensemble+QBC also benefits from improved generalization by the optimized ensemble of domain-specific models. Finally, we verified that using the ensemble of models with uncertainty sampling strategy is still not competitive. We conclude that ensemble+QBC is an effective active-transfer learning strategy.

Besides the relative merits of individual AL strategies, a main observation for our investigation is how quickly performance of different methods saturates. It can be noticed that for eyes the performance reaches 2% of the maximum with around 3,000 annotations, and for feet, the performance reaches 2% of the maximum with around 2,100 annotations. Combined with the observations in Section 5.2, this indicates that excellent performance can be achieved for part detection in most animal classes by annotating a small representative subset of classes and a small number of corresponding images. This result is somewhat remarkable and can be attributed to the excellent performance of pre-trained deep neural networks as general-purpose representations. Recall that the networks were pre-trained for image classification and not part detection, not using any of the source or target classes.

Sampling strategy analysis. Figure 6 shows the distribution of selected animal classes during individual learning rounds for the QBC strategy. The distribution is clearly non-uniform, and the method seems to select representative classes within groups such as "reptiles", "felines", etc.

5.4 Semi-supervised domain adaptation

While previous experiments focused on the DG task, where target domains are unobserved during training, here we consider a standard DA task where samples from the target domain can be accessed during training. In more detail, we assume a part detector pretrained on all part annotations from a specific animal class and, given a limited number of part annotations from a specified target class, we aim to adapt the pre-trained detector for this target class.

More specifically, for each class T from our pool of 50 target classes we first pre-train the baseline 6×6 upsample part detector on its T's semantically nearest animal class $N(T)$ selected from the set of all animal domains. Then, we expand the training set with the samples from T, keeping track of the evolution of the detection performance on T's testing subset as the training set increases.

This more standard semi-supervised DA scenario allows to utilize existing methods for improving the generalization properties of the part detectors. We thus attached the gradient reversal layer [15] (GRL) to the last layer before the keypoint filters (pool4 of VGG-VD) of our 6×6 upsample architecture. GRL aims

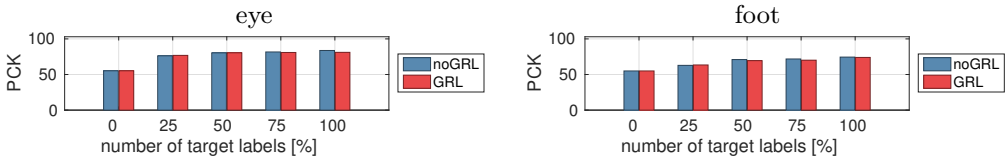


Fig. 7. Semi-supervised DA. For each target class, a pre-trained detector is adapted from its semantically nearest class. Each bar denotes average performance (PCK) over all target classes as more target annotations are added to the training set. We compare the baseline detector (noGRL) and its extension with the gradient reversal layer (GRL) for both eye and foot part detection.

to make the pool4 features invariant to the underlying domain (animal class), possibly improving the generalization capabilities of the succeeding keypoint detection layers.

The results of our experiment are reported in Figure 7. Because both GRL and noGRL perform on par we conclude, in agreement with the previous sections, that, on our task, the baseline deep keypoint detector already exhibits strong domain invariance properties without the need to include additional DA components.

6 Conclusions

In this work we focused on the semantic part detection task and considered the problem of the transferability of the corresponding detectors from source classes for which we have annotations, to target classes which were not seen before. We experimentally demonstrated that, as often hypothesized in previous work, semantic parts are powerful tools with good generalization properties. More precisely, we showed that parts transfer well to the majority of new classes even if trained from a limited number of examples.

These very encouraging results suggest that the pre-trained deep representations have the ability to learn novel concepts quickly and effectively. We further confirmed this by showing that adding a specialized DA component to our architecture does not bring a significant boost in performance. In the future, a more systematic study of the asymptotic properties of supervised training is warranted, in order to assess if, for certain well defined but broad problems such as the detection of certain parts in *all* animals, could be solved essentially by "exhaustion", by collecting once for all a sufficiently large pool of annotated examples.

Acknowledgements This work has been supported by Xerox Research Center Europe and ERC 677195-IDIU.

References

1. Yusuf Aytar and Andrew Zisserman. Tabula rasa: Model transfer for object category detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.
2. Mahsa Baktashmotlagh, Mehrtash Harandi, Brian Lovell, and Mathieu Salzmann. Unsupervised domain adaptation by domain invariant projection. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.
3. Yee Seng Chan and Hwee Tou Ng. Domain adaptation with active learning for word sense disambiguation. In *Annual Meeting of the Association for Computational Linguistics(ACL)*, 2007.
4. Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
5. Mircea Cimpoi, Subhransu Maji, and Andrea Vedaldi. Deep filter banks for texture recognition and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
6. David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine Learning*, 15(2):201–221, 1994.
7. Brendan Collins, Jia Deng, Kai Li, and Li Fei-Fei. Towards scalable dataset construction: An active learning approach. In *European Conference on Computer Vision (ECCV)*, 2008.
8. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
9. Thomas Deselaers, Bogdan Alexe, and Vittorio Ferrari. Localizing objects while learning their appearance. In *European Conference on Computer Vision (ECCV)*, 2010.
10. Ian Endres, Vivek Srikumar, Ming-Wei Chang, and Derek Hoiem. Learning shared body plans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
11. Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
12. Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.
13. Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.
14. Chuang Gan, Tianbao Yang, and Boqing Gong. Learning attributes equals multi-source domain generalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
15. Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning (ICML)*, 2015.
16. Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.
17. Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

18. Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.
19. Matthieu Guillaumin, Daniel Küttel, and Vittorio Ferrari. Imagenet auto-annotation with segmentation propagation. *International Journal of Computer Vision*, 110(3):328–348, 2014.
20. Judy Hoffman, Sergio Guadarrama, Eric S. Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. LSDA: Large scale detection through adaptation. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2014.
21. Judy Hoffman, Brian Kulis, Trevor Darrell, and Kate Saenko. Discovering latent domains for multisource domain adaptation. In *European Conference on Computer Vision (ECCV)*, 2012.
22. Judy Hoffman, Erik Rodner, Jeff Donahue, Trevor Darrell, and Kate Saenko. Efficient learning of domain-invariant image representations. In *International Conference on Learning representations (ICLR)*, 2013.
23. Alex Holub, Pietro Perona, and Michael C. Burl. Entropy-based active learning for object recognition. In *CVPR Workshop on Online Learning for Classification (OLC)*, 2008.
24. Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
25. Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active learning with gaussian processes for object categorization. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.
26. Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *CoRR*, arXiv:1602:07332, 2016.
27. Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 1995.
28. Daniel Küttel and Vittorio Ferrari. Figure-ground segmentation by transferring window masks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
29. Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
30. Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.
31. Jonathan L. Long, Ning Zhang, and Trevor Darrell. Do convnets learn correspondence? In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2014.
32. Stephen Milborrow, John Morkel, and Fred Nicolls. The MUCT Landmarked Face Database. In *Annual Symposium of the Pattern Recognition Association of South Africa*, 2010. `http://www.milbo.org/muct`.
33. Damian Mrowca, Marcus Rohrbach, Judy Hoffman, Ronghang Hu, Kate Saenko, and Trevor Darrell. Spatial semantic regularisation for large scale object detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

34. Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning (ICML)*, 2013.
35. David Novotny, Diane Larlus, and Andrea Vedaldi. I have seen enough: Transferring parts across categories. In *BMVA British Machine Vision Conference (BMVC)*, 2016.
36. Amar Parkash and Devi Parikh. Attributes for classifier feedback. In *European Conference on Computer Vision (ECCV)*, 2012.
37. Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, and Hong-Jiang Zhang. Two-dimensional active learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
38. Piyush Rai, Avishek Saha, Hal Daumé III, and Suresh Venkatasubramanian. Domain adaptation meets active learning. In *ACL Workshop on Active Learning for Natural Language Processing (ALNLP)*, 2010.
39. Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. Transfer learning in a transductive setting. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2013.
40. Marcus Rohrbach, Michael Stark, György Szarvas, Iryna Gurevych, and Bernt Schiele. What helps where – and why? semantic relatedness for knowledge transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
41. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
42. Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of both worlds: human-machine collaboration for object annotation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
43. Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European Conference on Computer Vision (ECCV)*, 2010.
44. Avishek Saha, Piyush Rai, Hal Daumé III, Suresh Venkatasubramanian, and Scott L. DuVall. Active supervised domain adaptation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2011.
45. H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Annual ACM workshop on Computational Learning Theory (CLT)*, 1992.
46. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, arXiv:1409.1556, 2014.
47. Michael Stark, Michael Goesele, and Bernt Schiele. A shape-based object class model for knowledge transfer. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.
48. Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2016.
49. Qian Sun, Rita Chattopadhyay, Sethuraman Panchanathan, and Jieping Ye. A two-stage weighting framework for multi-source domain adaptation. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2011.
50. Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, 2002.
51. Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Sharing visual features for multiclass and multiview object detection. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(5):854–869, 2007.

52. Shubham Tulsiani, João Carreira, and Jitendra Malik. Pose induction for novel object categories. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.
53. Shubham Tulsiani and Jitendra Malik. Viewpoints and keypoints. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
54. Manik Varma and Andrew Zisserman. A statistical approach to material classification using image patch exemplars. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(11):2032–2047, 2009.
55. Andrea Vedaldi, Siddarth Mahendran, Stavros Tsogkas, Subhrajyoti Maji, Ross Girshick, Juho Kannala, Esa Rahtu, Iasonas Kokkinos, Matthew B Blaschko, Daniel Weiss, Ben Taskar, Karen Simonyan, Naomi Saphra, and Sammy Mohamed. Understanding objects in detail with fine-grained attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
56. Sudheendra Vijayanarasimhan and Kristen Grauman. What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
57. Catherine Wah, Steve Branson, Pietro Perona, and Serge Belongie. Multiclass recognition and part localization with humans in the loop. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.
58. Xuezhi Wang, Tzu-Kuo Huang, and Jeff Schneider. Active transfer learning under model shift. In *International Conference on Machine Learning (ICML)*, 2014.
59. Xuezhi Wang and Jeff Schneider. Flexible transfer learning under support and model shift. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2014.
60. Jun Yang, Rong Yan, and Alexander G. Hauptmann. Cross-domain video concept detection using adaptive SVMs. In *ACM Multimedia*, 2007.
61. Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(12):2878–2890, 2013.
62. Yongxin Yang and Timothy M. Hospedales. Multivariate regression on the grassmannian for predicting novel domains. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
63. Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision (ECCV)*, 2014.

4

Learning the Structure of Objects from Web Supervision

This work was presented as an *oral* presentation at the European Conference on Computer Vision Workshops, 2016 [Novotny et al. 2016b].

Learning the structure of objects from Web supervision

David Novotny[1,2] Diane Larlus[2] Andrea Vedaldi[1]

[1]Visual Geometry Group
University of Oxford
{david,vedaldi}@robots.ox.ac.uk

[2]Computer Vision Group
Xerox Research Centre Europe
diane.larlus@xrce.xerox.com

Abstract. While recent research in image understanding has often focused on recognizing *more types of objects*, understanding *more about the objects* is just as important. Learning about object parts and their geometric relationships has been extensively studied before, yet learning large space of such concepts remains elusive due to the high cost of collecting detailed object annotations for supervision. The key contribution of this paper is an algorithm to learn geometric and semantic structure of objects and their semantic parts automatically, from images obtained by querying the Web. We propose a novel embedding space where geometric relationships are induced in a soft manner by a rich set of non-semantic mid-level anchors, bridging the gap between semantic and non-semantic parts. We also show that the resulting embedding provides a visually-intuitive mechanism to navigate the learned concepts and their corresponding images.

Keywords: object part detection, Web supervision, mid-level patches

1 Introduction

Modern deep learning methods have dramatically improved the performance of computer vision algorithms, from image classification [1] to image captioning [2, 3] and activity recognition [4]. Even so, image understanding remains rather crude, oblivious to most of the nuances of real world images. Consider for example the notion of *object category*, which is a basic unit of understanding in computer vision. Modern benchmarks consider an increasingly large number of such categories, from thousands in the ILSVRC challenge [5] to hundred thousands in the full ImageNet [6]. Despite this ontological richness, there is only limited understanding of the internal geometric structure and semantics of these categories.

In this paper we aim at learning the internal details of object categories by jointly *learning about objects, their semantic parts, and their geometric relationships*. Learning about semantic *nameable* parts plays a crucial role in visual understanding. However, standard supervised approaches are difficult to apply to this problem due to the cost of collecting large quantities of annotated example images. A scalable approach needs to discover this information with *minimal or no supervision*.

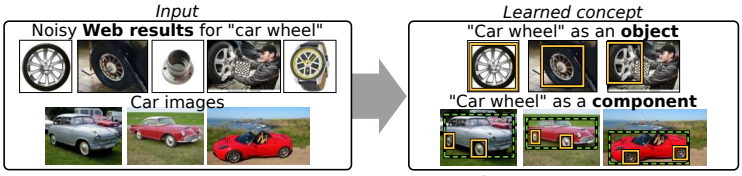


Fig. 1. Our goal is to learn the semantic structure of objects automatically using Web supervision. For example, given noisy images obtained by querying an Internet search engine for "car wheel" and for "cars", we aim at learning the "car wheel" concept, and its dual nature: as an object in its own right, and as a component of another object.

As a scalable source of data, we look at Web supervision to learn the structure of objects from thousands of images obtained automatically by querying search engines (*crf.* fig. 1). This poses two significant challenges: identifying images of the semantic parts in very noisy Web results (crf. fig. 2) while, at the same time, discovering their geometric relationships. The latter is particularly difficult due to the drastic scale changes of parts when they are imaged in the context of the whole object or in isolation.

If parts are looked at independently, noise and scale changes can easily confuse image recognition models. Instead, one needs to account for the fact that object classes have a well-defined geometric structure which constraints how different parts fit together. Thus, we need to introduce a *geometric frame* that can, for any view of an object class, constrain and regularize the location of the visible semantic parts, establishing better correspondences between views.

Traditional representations such as spring models were found to be too fragile to work in our Webly-supervised setting. To solve this issue, we introduce a novel vector embedding that encodes the geometry of parts relatively to a robust reference frame. This reference frame builds on non-semantic anchor parts which are learned automatically using a new method for non-semantic part discovery (section 2.2); we show that this method is significantly better than alternative and more complex techniques for part discovery. The new geometric embedding is further combined with appearance cues and used to improve the performance in semantic part detection and matching.

A byproduct of our method is a large collection of images annotated with objects, semantic parts, and their geometric relationships, that we refer to as a *visual semantic atlas* (section 4). This atlas allows to *visually navigate* images based on conceptual and geometric relations. It also emphasizes the dual nature of parts, as components of an object and as semantic categories, by naturally bridging images that zooms on a part or that contain the object as a whole.

1.1 Related work

Our work touches on several active research areas: localizing objects with weak supervision, learning with Web images, and discovering or learning mid-level features and object parts.

Localizing objects with weak supervision. When training models to localize objects or parts, it is impractical to expect large quantities of bounding box annotations. Recent works have tackled the localization problem with

Fig. 2. Top images retrieved from an Internet search engine for some example queries. Note that part results are more noisy than full object results (the remaining collected images get even noisier, not shown here).

only image-level annotations. Among them, *weakly supervised object localization* methods [7–13] assume for each image a list of every object type it contains. In the *co-detection* [14–17] and *co-segmentation* [18–21] problems, the algorithm is given a set of images that all contain at least one instance of a particular object. They differ in their output: co-detection predicts bounding boxes, while segmentation predicts pixel-level masks. Yet, co-detection, co-segmentation and weakly-supervised object localization (WSOL) are different flavors of the localization problem with weak supervision. For co-detection and WSOL, the task is nearly always formulated as a multiple instance learning (MIL) problem [7, 8, 22, 16, 23]. The formulation in [11, 12] departs from MIL by leveraging the strong annotations for some categories to transfer knowledge to the remaining categories. A few approaches model images using topic models [10, 17].

Recently, CNN architectures were also proved to work well in weakly supervised scenarios [24]. We will compare with [24] in the experiments section. None of these works have considered semantic parts. Closer to our work, the method of [25] proposes unsupervised discovery of dominant objects using part-based region matching. Because of its unsupervised process, this method is not suited to name the discovered objects or matched regions, and hence lack semantics. Yet we also compare with this approach in our experiments.

Learning from Web supervision. Most previous works [26–29] that learn from noisy Web images have focused on image classification. Usually, they adopt an iterative approach that jointly learns models and finds clean examples of a target concept. Only few works have looked at the problem of localization. Some approaches [30, 21] discover common segments within a large set of Web images, but they do not quantitatively evaluate localization. The recent method of [31] localizes objects with bounding boxes, and evaluate the learnt models, but as the previous two, it does not consider object parts.

Closer to our work, [32] aims at discovering common sense knowledge relations between object categories from Web images, some of which correspond to the "part-of" relation. In the process of organizing the different appearance variations of Webly mined concepts, [33] uses a "vocabulary of variance" that may include part names, but those are not associated to any geometry.

Unsupervised parts, mid-level features, and semantic parts. Objects are modeled using the notion of *parts* since the early work on pictorial structure [34], in the constellation [35] and ISM [36] models, and more recently the DPM [37]. Parts are most commonly defined as localized components with consistent ap-

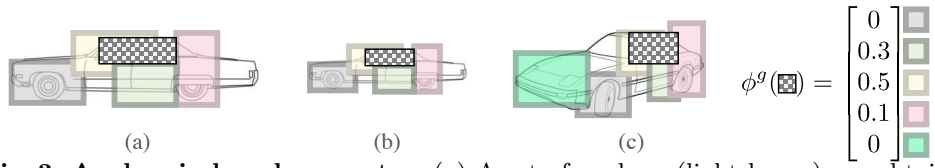


Fig. 3. Anchor-induced geometry. (a) A set of anchors (light boxes) are obtained from a large number of unsupervised non-semantic part detectors. The geometry of a semantic part or object is then expressed as a vector ϕ^g of anchor overlaps. (b) The representation is scale and translation invariant. (c) The representation implicitly codes for multiple aspects.

pearance and geometry in an object. All these works have in common to discover object parts without naming them. In practice, only some of these parts have an actual semantic interpretation. *Mid-level features* [38–43] are discriminative [41, 44] or rare [38] blocks, which are leveraged for object recognition. Again, these parts lack semantic. The non-semantic anchors that we use share similarities with [45] and [42], that we discuss in section 2.2. *Semantic* parts have triggered recent interest [46–48]. These works require strong annotations in the form of bounding boxes [46] or segmentation masks [47, 48] at the part level. Here we depart from existing work and aim at mining semantic nameable parts with as little supervision as possible.

2 Method

This section introduces our method to learn semantic parts using weak supervision from Web sources. The key challenge is that search engines, when queried for object parts, return many outliers containing other parts as well, the whole object, or entirely unrelated things (fig. 2). In this setting, standard weakly-supervised detection approaches fail (section 3). Our solution is a novel, robust, and flexible representation of object parts (section 2.1) that uses the output of a simple but very effective non-semantic part discovery algorithm (section 2.2).

2.1 Learning semantic parts using non-semantic anchors

In this section, we first flesh out our method for weakly-supervised part learning and then dive into the theoretical justification of our choices.

MIL: baseline, context, and geometry-aware. As standard in weakly-supervised object detection, our method starts from the *Multiple Instance Learning* (MIL) [49] algorithm. Let $\mathbf{x}_i$ be an image and let $\mathcal{R}(\mathbf{x}_i)$ be a shortlist of image regions R that are likely to contain objects or parts, obtained for instance using selective search [50]. Each image $\mathbf{x}_i$ can be either positive $y_i = +1$ if it is deemed to contain a certain part or negative $y_i = -1$ if not. MIL fits to this data a (linear) scoring function $\langle \phi(\mathbf{x}_i|R), \mathbf{w} \rangle$, where $\mathbf{w}$ is a vector of parameters and $\phi(\mathbf{x}_i|R) \in \mathbb{R}^d$ is a descriptor of the region R of image $\mathbf{x}_i$, by minimizing:

$$\min_{\mathbf{w} \in \mathbb{R}^d} \frac{\lambda}{2} \|\mathbf{w}\|^2 + \frac{1}{n} \sum_{i=1}^{n} \max\{0, 1 - y_i \max_{R \in \mathcal{R}(\mathbf{x}_i)} \langle \phi(\mathbf{x}_i|R), \mathbf{w} \rangle\} \tag{1}$$

In practice, eq. (1) is optimized by alternatively selecting the maximum scoring region for each image (also known as "re-localization") and optimizing $\mathbf{w}$ for a fixed selection of the regions. In this manner, MIL should automatically discover regions that are most predictive of a given label, and which therefore should correspond to the sought visual entity (object or semantic part). However, this process may fail if descriptors are not sufficiently strong.

For **baseline MIL** the descriptor $\phi(\mathbf{x}|R) = \phi^a(\mathbf{x}|R) \in \mathbb{R}^{d_a}$ captures the region's appearance. A common improvement is to extend this descriptor with *context information* by appending a descriptor of a region $R' = \mu(R)$ surrounding R, where $\mu(R)$ isotropically enlarges R; thus in **context-aware MIL**, $\phi(\mathbf{x}|R) = \text{stack}(\phi^a(\mathbf{x}|R), \phi^a(\mathbf{x}|\mu(R)))$.

Neither baseline or context-aware MIL leverage the fact that objects have a well-defined geometric structure, which significantly constrains the search space for parts. DPM uses such constraints, but as a fixed set of geometric relationships between part pairs that are difficult to learn when examples are extremely noisy. Furthermore, DPM-like approaches learn the most visually-stable parts, which often are *not* the semantic ones.

We propose here an alternative method that captures geometry indirectly, on top of a rich set of unsupervised mid-level non-semantic parts $\{p_1, ..., p_K\}$, which we call *anchors* (fig. 3). Let us assume that, given an image $\mathbf{x}$, we can locate the (selective search) regions $R_{p_k,\mathbf{x}}$ containing each anchor p_k. We define the following geometric embedding ϕ^g of a region R with respect to the anchors:

$$\phi^g(\mathbf{x}|R) = \begin{bmatrix} \rho(R, R_{p_1,\mathbf{x}}) \\ \vdots \\ \rho(R, R_{p_K,\mathbf{x}}) \end{bmatrix}. \tag{2}$$

Here ρ is a measure such as intersection-over-union (IoU) that tells whether two regions overlap. By choosing a function ρ such as IoU which is invariant to scaling, rotation, and translation of the regions, so is the embedding ϕ^g. Hence, as long as anchors stay attached to the object, $\phi^g(\mathbf{x}|R)$ encodes the location of R relative to an object-centric frame. This representation is robust because, even if some anchors are missing or misplaced, the vector $\phi^g(\mathbf{x}|R)$ is not greatly affected. The geometric encoding $\phi^g(\mathbf{x}|R)$ is combined with the appearance descriptor $\phi^a(\mathbf{x}|R)$ in a joint appearance-geometric embedding

$$\phi^{ag}(\mathbf{x}|R) = \phi^a(\mathbf{x}|R) \otimes \phi^g(\mathbf{x}|R) \tag{3}$$

where $\otimes$ is the Kronecker product. After vectorization, this vector is used as a descriptor $\phi(\mathbf{x}|R) = \phi^{ag}(\mathbf{x}|R)$ of region R in **geometry-aware MIL**. The next few paragraphs discuss its properties.

Modeling multiple parts. Plugging ϕ^{ag} of eq. (3) into eq. (1) of MIL results in the scoring function $\langle \mathbf{w}, \phi^{ag}(\mathbf{x}|R) \rangle = \sum_{k=1}^{K} \langle \mathbf{w}_k, \phi^a(\mathbf{x}|R) \rangle \rho(R, R_{p_k,\mathbf{x}})$ which interpolates between K appearance models *based on how the region R is geometrically related to the anchors $R_{p_k,\mathbf{x}}$*. In particular, by selecting different anchors this model may capture simultaneously the appearance of all parts of an object. In order to control the capacity of the model, the smoothness of the

interpolator can be increased by replacing IoU with a softer version, which we do next.

Smoother overlap measure. The IoU measure is a special case of the following family of PD kernels:

Theorem 1. *Let R and Q be vectors in a Hilbert $\mathcal{H}$ space such that $\langle R, R\rangle + \langle Q, Q\rangle - \langle R, Q\rangle > 0$. Then the function $\rho(R, Q) = \frac{\langle R,Q\rangle}{\langle R,R\rangle + \langle Q,Q\rangle - \langle R,Q\rangle}$ is a positive definite kernel.*

Please refer to appendix A for a proof of the theorem. The IoU is obtained when R and Q are indicator functions of the respective regions (because $\langle R, Q\rangle = \int R(x,y)Q(x,y)\,dx\,dy = |R \cap Q|$). This suggests a simple modification to construct a Soft IoU (SIoU) version of the latter. For a region $R = [x_1, x_2] \times [y_1, y_2]$, the indicator can be written as $R(x, y) = H(x - x_1)H(x_2 - x)H(y - y_1)H(y_2 - y)$ where $H(z) = [z \geq 0]$ is the Heaviside step function. SIoU is obtained by replacing the indicator by the smoother function $H_\alpha(z) = \exp(\alpha z)/(1 + \exp(\alpha z))$ instead. Note that SIoU is non-zero even when regions do not intersect.

Theorem 1 provides also an interpretation of the geometric embedding ϕ^g of eq. (2) as a vector of region coordinates relative to the anchors. In fact, its entries can be written as $\rho(R, R_{p_k,\mathbf{x}}) = \langle \psi_{\text{SIoU}}(R), \psi_{\text{SIoU}}(R_{p_k,\mathbf{x}})\rangle$ where $\psi_{\text{SIoU}}(R) \in \mathcal{H}_{\text{SIoU}}$ is the linear embedding (feature map) induced by the kernel ρ[1].

Modeling multiple aspects. So far, we have assumed that all parts are always visible; however, anchors also provide a mechanism to deal with the multiple aspects of 3D objects. As depicted in fig. 3.c, as the object rotates out of plane, anchors naturally appear and disappear, therefore activating and de-activating aspect-specific components in the model. In turn, this allows to model viewpoint-specific parts or appearances. In practice, we extract the L highest scoring detections R_l of the same anchor p_k, and keep the one closest to R.

In order to allow anchors to turn off in the model, the geometric embedding is modified as follows. Let $s_k(R_l|\mathbf{x})$ be the detection score of anchor k in correspondence of the region R_l; then

$$\rho(R, R_{p_k,\mathbf{x}}) = \max_{l \in \{1,\dots,L\}} \text{SIoU}(R, R_l) \times \max\{0, s_k(R_l|\mathbf{x})\}. \tag{4}$$

If the anchor is never detected ($s_k(R_l|\mathbf{x}) \leq 0$ for all R_l) then $\rho(R, R_{p_k,\mathbf{x}}) = 0$. Furthermore, this expression also disambiguates ambiguous anchor detections by picking the one closest to R. Note that in eq. (4) one can still interpret the factors $\text{SIoU}(R, R_l)$ as projections $\langle \psi_{\text{SIoU}}(R), \psi_{\text{SIoU}}(R_l)\rangle$.

Relation to DPM. DPM is also a MIL method using a joint embedding $\phi^{\text{DPM}}(\mathbf{x}|R_1, \dots, R_K)$ that codes simultaneously for the appearance of K parts and their pairwise geometric relationships. Our Webly-supervised learning problem requires a representation that can bridge object-focused images (where several parts are visible together as components) and part-focused images (where

[1] The anchor vectors $\psi_{\text{SIoU}}(R_{p_k,\mathbf{x}})$ are not necessarily orthonormal (they are if anchors do not overlap), but this can be restored up to a linear transformation of the coordinates.

parts are regarded as objects in their own right). This is afforded by our embedding $\phi^{ag}(\mathbf{x}|R)$ but not by the DPM one. Besides bridging parts as components and parts as objects, our embedding is very robust (important in order to deal with very noisy training labels), automatically codes for multiple object aspects, and bridges unsupervised non-semantic parts (the anchors) with semantic ones.

2.2 Anchors: weakly-supervised non-semantic parts

The geometric embedding in the previous section leverages the power of an intermediate representation: a collection of anchors $\{p_k\}_{k=1}^K$, learned automatically using weak supervision. While there are many methods to discover discriminative non-semantic mid-level parts from image collections (section 1.1), here we propose a simple alternative that, empirically, works better in our context.

We learn the anchors using a formulation similar to the MIL objective (eq. (1)):

$$\min_{\boldsymbol{\omega}_1,\dots,\boldsymbol{\omega}_K} \sum_{k=1}^{K} \left[\frac{\lambda}{2}\|\boldsymbol{\omega}_k\|^2 - \frac{1}{n}\sum_{i=1}^{n} y_i \left[\max_{R\in\mathcal{R}(\mathbf{x}_i)} \langle \phi^a(\mathbf{x}_i|R), \boldsymbol{\omega}_k \rangle \right]_+ \right]_+ + \gamma \sum_{k\neq q} \left\langle \frac{\boldsymbol{\omega}_k}{\|\boldsymbol{\omega}_k\|}, \frac{\boldsymbol{\omega}_q}{\|\boldsymbol{\omega}_q\|} \right\rangle^2, \tag{5}$$

where $[z]_+ = \max\{0, z\}$. Intuitively, anchors are learnt as discriminative mid-level parts using weak supervision. Anchor scores $s_k(R|\mathbf{x}) = \langle \phi^a(\mathbf{x}|R), \boldsymbol{\omega}_k \rangle$ are parametrized by vectors $\boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_K$; the first term in eq. (5) is akin to the baseline MIL formulation of section 2.1 and encourages each anchor p_k to score highly in images $\mathbf{x}_i$ that contain the object ($y_i = +1$) and to be inactive otherwise ($y_i = -1$). The last term is very important and encourages the learned models $\{\boldsymbol{\omega}_k\}_{k=1}^K$ to be mutually orthogonal, enforcing *diversity*. Note that anchors use the pure appearance-based region descriptor $\phi^a(\mathbf{x})$ since the geometric-aware descriptor $\phi^{ag}(\mathbf{x})$ can be computed only once anchors are available. Optimization uses stochastic gradient descent with momentum.

This formulation is similar to the MIL approach of [45] which, however, does not contain the orthogonality term. When this term is removed, we observed that the solution degenerates to detecting the most prominent object in an image. [39] uses instead a significantly more complex formulation inspired by mode seeking; in practice we opted for our approach due to its simplicity and effectiveness.

2.3 Incorporating strong annotations in MIL

While we are primarily interested in understanding whether semantic object parts can be learned from Web sources alone, in some cases the precise definition of the extent of a part is inherently ambiguous (*e.g.* what is the extent of a "human nose"?). Different benchmark datasets may use somewhat different definition of these concepts, making evaluation difficult. In order to remove or at least reduce this dataset-dependent ambiguity, we also explore the idea of using a single strongly annotated example to fix this degree of freedom.

Denote by $(\mathbf{x}_a, R_a)$ the single strongly-annotated example of the target part. This is incorporated in the MIL formulation, eq. (1), by augmenting the score

with a factor that compares the appearance of a region to that of R_a:

$$\langle \phi(\mathbf{x}_i|R), \mathbf{w} \rangle \times \begin{cases} \frac{1}{C} \exp \beta \langle \phi^a(\mathbf{x}_i|R), \phi^a(\mathbf{x}_a|R_a) \rangle, & y_i = +1, \\ 1, & y_i = -1. \end{cases} \tag{6}$$

where $C = \text{avg}_{i:y_i=+1} \exp \beta \langle \phi^a(\mathbf{x}_i|R), \phi^a(\mathbf{x}_a|R_a) \rangle$ is a normalizing constant. In practice, this is used only during re-localization rounds of the training phase to guide spatial selection; at test time, bounding boxes are scored solely by the model of eq. (1) without the additional term. Other formulations, that may use a mixture of strongly and Webly supervised examples, are also possible. However, this is besides our focus, which is to see whether parts are learnable from the Web automatically, and the single supervision is only meant to reduce the ambiguity in the task for evaluation.

3 Experiments

This section thoroughly evaluates the proposed method. Our main evaluation is a comparison with existing state-of-the-art techniques on the task of Webly-supervised semantic part learning. In section 3.1 we show that our method is substantially more accurate than existing alternatives and, in some cases, close to fully-supervised part learning.

Having established that, we then evaluate the weakly-supervised mid-level part learning (section 2.2) that is an essential part of our approach. It compares favorably in terms of simplicity, scalability, and accuracy against existing alternatives for discriminability as well as spatial matching of object categories (section 3.2).

Datasets. The Labeled Face Parts in the Wild (LFPW) dataset [51] contains about 1200 face images annotated with outlines for landmarks. Outlines are converted into bounding box annotations and images with missing annotations are removed from the test set. These test images are used to locate the following entities: *face, eye, eyebrow, nose*, and *mouth*.

The PascalParts dataset [47] augments the PASCAL VOC 2010 dataset with segmentation masks for object parts. Segmentation masks are converted into bounding boxes for evaluation. Parts of the same type (*e.g.* left and right wheels) are merged in a single entity (*wheel*). Objects marked as truncated or difficult are not considered for evaluation. The evaluation focuses on the bus and car categories with 18 entity types overall: *car*, *bus*, and their *door*, *front*, *headlight*, *mirror*, *rear*, *side*, *wheel*, and *window* parts. This dataset is more challenging, as entities have large intra-class appearance and pose variations. The evaluation is performed on images from the validation set that contain at least one object instance. Furthermore, following [48], object occurrences are roughly localized before detecting the parts using their localization procedure. Finally, objects whose bounding box larger side is smaller than 80 pixels are removed as several parts are nearly invisible below that scale.

The training sets from both datasets are utilized solely for training the fully supervised baselines (section 3.1), and they are not used by MIL approaches.

Experimental details. Regions are extracted using selective search [50], and described using ℓ_2-normalized Decaf [52] fc6 features to compute the appearance embedding $\phi^a(\mathbf{x}|R)$. The context descriptor $\mu(R)$ is extracted from a region triple the size of R. The joint appearance-geometric embedding $\phi^{ag}(\mathbf{x}|R)$ is obtained by first extracting the top $L = 5$ non-overlapping detections of each anchor and then applying eqs. (3) and (4).

A separate mid-level anchor dictionary $\{p_1, ..., p_K\}$ is learnt for each object class using the Web images for all the semantic parts for the target object (including images of the object as a whole) as positive images and the background clutter images of [53] as negative ones. Eq. (5) is optimized using stochastic gradient descend (SGD) with momentum for 40k iterations, alternating between positive and negative images. We train 150 anchor detectors per object class.

MIL semantic part detectors are trained solely on the Web images and the background class of [53] is used as negative bag for all the objects. The first five relocalization rounds are performed using the appearance only and the following five use the joint appearance-geometry descriptor (the joint embedding performs better with these two distinct steps). The MIL λ hyperparameter is set by performing leave-one-category-out cross-validation.

Web images for parts are acquired by querying the BING image search engine. For car and bus parts, the query concatenates the object and the part names (e.g. "car door"). For face parts, we do not use the object name. We retrieve 500 images of the class corresponding to the object itself and 100 images of all other semantic part classes.

3.1 Webly supervised localization of objects and semantic parts

This section evaluates the detection performance of our approach. We gradually incorporate the proposed improvements, *i.e.* the context descriptor (C) and the geometrical embedding (G) to the basic MIL baseline (B) as defined in section 2.1 and monitor their impact.

We compare our method to the state-of-the-art co-localization algorithm of Cho *et al.* [25] and the state-of-the-art weakly supervised detection method from Bilen and Vedaldi [24]. To detect a given part with [25], we run their code on all images that contain that part (*e.g.* for co-localizing eyes we consider *face* and *eye* images). As reference, we also report a fully supervised detector, trained using bounding-boxes from the training set, for all objects and parts (F). For this, we use the R-CNN method of [54] on top of the same features used in MIL.

We mainly report the Average Precision (AP) per part/object class and its average (mAP) over all parts in each class. We also report the CorLoc (for correct localization) measure, as it is often used in the co-localization literature [55, 14]. As most parts in both datasets are relatively small, following [47], the IoU threshold for correct detection is set to 0.4.

Results. Table 1 reports the average AP and CorLoc over all parts of a given object class for all these methods. First, we observe that even the MIL baseline (B) outperforms off-the-shelf methods such as [25] and [24]. For [24], we have observed that the part detectors degrade to detecting subparts of semantic parts,

measure	mAP			averageCorLoc		
Parent class	{face}	{car}	{bus}	{face}	{car}	{bus}
Cho *et al.* [25]	16.6	16.9	12.4	31.4	29.9	15.5
Bilen & Vedaldi [24]	2.7	12.0	4.7	7.2	15.3	6.7
B	20.6	29.1	22.7	22.0	38.1	29.4
B+C	22.4	27.3	21.4	29.1	37.6	28.4
B+G	29.0	34.1	**23.3**	33.1	45.5	**31.5**
B+C+G	**44.9**	**34.4**	23.0	**52.5**	**47.8**	29.6
F	53.7	51.2	48.2	60.5	62.9	63.8
F+C+G	**61.4**	**60.3**	**54.1**	**67.8**	**71.8**	**66.0**

Table 1. Part detection results averaged for the face, car, and bus parent classes. mAP and average CorLoc for the MIL baseline (B), our improved versions that use context (C), geometrical embedding (G) compared to the fully supervised R-CNN (F).

Class		door	rear	wheel	wind.	side	car	front	headl.	mirror	mean{car}
Web	B	0.4	10.8	34.9	3.6	63.1	92.6	**55.2**	0.7	**0.3**	29.1
	B+C	0.8	11.4	31.3	4.9	58.8	83.0	54.0	**1.0**	0.2	27.3
	B+G	0.7	11.8	**47.9**	**22.7**	71.3	**97.8**	54.5	0.2	0.2	34.1
	B+C+G	**5.1**	**14.7**	43.6	22.6	**72.3**	95.7	54.7	0.3	0.2	**34.4**
Full	F	17.0	**39.0**	66.3	53.3	83.2	95.1	75.9	25.3	5.5	51.2
	F+C+G	**31.1**	30.7	**72.3**	**67.3**	**90.1**	**98.7**	**82.9**	**48.1**	**21.3**	**60.3**

Table 2. Individual part detection results for car: APs for the MIL baseline (B), our improved versions that use context (C), geometrical embedding (G) and the different flavors of the fully supervised R-CNN (F).

suggesting that [24] lacks robustness to drastic scale variations and to the large amount of noise present in our dataset. Second, we see that using the geometric embedding (+G) always improves the baseline results by $1-10$ mAP points. On top of geometry, using context (+C) helps for face and car parts, but not for buses. Overall the unified embedding brings a large improvement for faces (+24.3 mAP) and for cars (+5.3 mAP) and more contained for buses (+0.6 mAP). Importantly, these improvements significantly reduce the gap between using noisy Web supervision and the fully supervised R-CNN (F); overall, Webly supervision achieves respectively 84%, 67%, and 48% of the performance of (F).

Last but not least, we also experimented extending the fully supervised R-CNN method with the joint appearance-geometry embedding and context descriptor (F+C+G), which improves part detections by +7.7, +9.1, +5.9 mAP points respectively. This suggests that our representation may be applicable well beyond weakly supervised learning.

Table 2 shows per-part detection results for the car parts. We see that geometry helps for 6 parts out of 9. Out of the three remaining parts, two are cases where the MIL baseline failed. In the less ambiguous fully-supervised scenario, the geometric embedding improves the performance in 8 out of 9 cases.

Leveraging a single annotation. As noted in section 2.3, one issue with weakly supervised part learning is the inherent ambiguity in the part extent, that may differ from dataset to dataset. Here we address the ambiguity by adding

measure	mAP			averageCorLoc		
Parent class	{face}	{car}	{bus}	{face}	{car}	{bus}
A	29.4 ± 2.6	25.1 ± 2.7	24.5 ± 2.7	38.2 ± 2.5	39.8 ± 3.2	39.6 ± 3.2
A+B	27.3 ± 3.1	33.3 ± 1.1	26.9 ± 1.3	34.6 ± 3.7	46.6 ± 1.5	40.0 ± 2.3
A+B+C	38.2 ± 3.1	32.4 ± 1.2	26.6 ± 1.6	51.7 ± 3.2	49.4 ± 1.5	43.9 ± 3.0
A+B+G	34.5 ± 4.3	35.7 ± 1.1	28.1 ± 1.2	43.5 ± 4.8	48.8 ± 1.6	42.2 ± 2.2
A+B+C+G	**43.0 ± 3.6**	**36.4 ± 1.0**	**30.1 ± 1.8**	**54.7 ± 3.2**	**51.6 ± 1.6**	**45.9 ± 2.8**

Table 3. Part detection results using a single strong annotation (A): mAP and average CorLoc for the MIL baseline (B), our improved versions that use context (C), geometrical embedding (G). Mean and standard deviation over 25 random annotations.

a single strong annotation to the mix using the method described in section 2.3. We asked an annotator to select 25 representative part annotations per part class from the training sets of each dataset. We retrain every part detector for each of the annotations and report mean and standard deviation of mAP. As a baseline, we also consider an exemplar detector trained using the single annotated example (A).

Results are reported in Table 3. Compared to pure Web supervision (B+C+G) in Table 1, the single annotation (A+B+C+G) does not help for faces, for which the proposed method was already working very well, but there is a +2 mAP point improvement for cars and +6.8 mAP for buses, which are more challenging. We also note that the complete method (A+B+C+G) is substantially superior to the exemplar detector (A).

3.2 Validation of weakly-supervised mid-level anchors

This section validates the mid-level anchors (section 2.2) against alternatives in terms of discriminative information content and its ability of establishing meaningful matches between images, which is a key requirement in our application.

Discriminative power of anchors. Since most of the existing methods for learning mid-level patches are evaluated in terms of discriminative content in a classification setting, we adopt the same protocol here. In particular, we evaluate the anchors as mid-level patches on the MIT Scene 67 indoor scene classification task [56]. The pipeline first learns 50 mid-level anchors for each of the 67 scene classes. Then, similar to [43], images are split into spatial grids (2x2 and 1x1) and described by concatenating the maximum scores attained by each anchor detector inside each bin of the grid. All the grid descriptors are then concatenated to form a global image descriptor which is ℓ_2 normalized. 67 one-vs-rest SVM classifiers are trained on top of these descriptors. To be comparable with other methods, we consider both Decaf fc6 and VGG-VD fc7 [57] descriptors.

Table 4 contains the results of the classification experiment. Our weakly-supervised anchors clearly outperform other mid-level element approaches that are not based on CNN features [40, 39, 45, 44]. Among CNN based approaches, our method outperforms the state-of-the-art mid-level feature based method from [43] on both VGG-VD and Decaf features. Remarkably, using our part detectors improves over the baseline which uses the global image CNN descrip-

method	BoP† [40]	DMS† [39]	Jian *et al.*† [45]	RFDC† [44]	FC *Decaf* [52]	FC *VGG-VD* [57]
accuracy (%)	46.1	64.0	58.1	54.4	57.7	68.9

method	BoE† *Decaf* [43]	ours† *Decaf*	FV-CNN *Decaf* [58]	BoE† *VGG-VD* [43]	ours† *VGG-VD*	FV-CNN *VGG-VD* [58]
accuracy (%)	69.7	**71.5**	69.7	77.6	77.8	**81.6**

Table 4. Classification results on MIT Scenes [56]. Methods using mid-level elements are marked with †. For CNN-based approaches, features rely on Decaf or VGG-VD.

tor (FC) by 13.8 and 8.7 average accuracy points for Decaf and VGG-VD features respectively. Compared to other methods which are not based on detecting mid-level elements, our pipeline outperforms state-of-the-art FV-CNN for Decaf features and is inferior for VGG-VD.

Ability of anchors to establish semantic matches. The previous experiment assessed favorably the mid-level parts in terms of discriminative content; however, in the embedding ϕ^g, these are used as *geometric anchors*. Hence, here we validate the ability of the mid-level anchors to induce good semantic matches between pairs of images (before learning the semantic part models).

To perform semantic matching between a source image x_S and a target image x_T, we consider each part annotation R_S in the source and predict its best match $\hat{R}_T$ in the target. The quality of the match is evaluated by measuring the IoU between the predicted $\hat{R}_T$ and ground-truth R_T part. When a part appears more than once (*e.g.* eyes often appear twice), we choose the most overlaping pair. Performance is reported by averaging the match IoU for all part occurrences and pairs of images in the test set, reporting the results for each object category.

Given a source part R_S, the joint appearance-geometry embedding (*anchor-ag*) is extracted for the source part $\phi^{ag}(\mathbf{x}_S|R_S)$ and the target region $\hat{R}_T$ that maximizes the inner product $\langle \phi^{ag}(\mathbf{x}_S|R_S), \phi^{ag}(\mathbf{x}_T|\hat{R}_T)\rangle$ is returned as the predicted match. We also compare *anchor-g* that uses only the geometric embedding $\phi^g(\mathbf{x}|R)$ and the baseline *a* that uses only the appearance embedding $\phi^a(\mathbf{x}|R)$.

We also compare two strong off-the-shelf baselines: DSP [59], state-of-the-art pairwise semantic matching method, and the method of [60], state-of-the-art for joint alignment. To perform box matching with [59] and [60] we fit an affine transformation to the disparity map contained inside a given source bounding box and apply this transform to move this box to the target image. Due to scalability issues, we were unable to apply [60] to the full dataset[2], so we perform this comparison on a random subset of 50 images.

Table 5 presents the results of our benchmark. On the small subset of 50 images the costly approach of [60] performs better than our embedding only on the LFPW faces, where the viewpoint variation is limited. On the car and bus categories our method outperforms [60] by 10% and 16% average IoU respectively. Our method is also consistently better than DSP [59], on both the small and full test set.

[2] More precisely, we were not able to apply [60] on a dataset with more than 60 128×68 pixel images on a server with 120 GB of RAM.

Set	Parent class	Matching method				
		anchor-ag	anchor-g	a	Flowweb [60]	DSP [59]
50 images	{car}	**0.36**	**0.36**	0.31	0.34	0.23
	{bus}	**0.37**	0.36	0.31	0.31	0.22
	{face}	0.41	0.39	0.33	**0.43**	0.19
Full	{car}	**0.36**	**0.36**	0.30	-	0.22
	{bus}	**0.35**	**0.35**	0.29	-	0.21
	{face}	**0.41**	0.39	0.34	-	0.21

Table 5. Semantic matching. For every parent class, we report average overlap (IoU) over all semantic parts. The face class results are obtained on the LFPW dataset while bus and car results come from the PascalParts dataset.

We also note that the matching using geometric embeddings alone (*anchor-g*) achieves similar performance than the appearance-geometry matching (*anchor-ag*) which validates our intuition that the local geometry of an object is well-captured by the anchors.

4 An atlas for visual semantic

As a byproduct of Webly-supervised learning, our method annotates the Web images with semantic parts. By endowing an image dataset with such concepts, we show here that it is possible to browse these annotated images. All of this composes our visual semantic atlas (see a subset of the atlas in Figure 4) that allows to navigate from one image to another, even between an image of a full object and a zoomed-in image of one of its parts.

5 Conclusions

We have proposed a novel method for learning about objects, their semantic parts, and their geometric relationships, from noisy Web supervision. This is achieved by first learning a weakly supervised dictionary of mid-level visual elements which define a robust object-centric coordinate frame. Such property theoretically motivates our approach. The geometric projections are then used in a novel appearance-geometry embedding that improves learning of semantic object parts from noisy Web data. We showed improved performance over co-localization [25], deep weakly supervised approach [24] and a MIL baseline on all benchmarked datasets. Extensive evaluation of our proposed mid-level elements shows comparable results to state-of-the-art in terms of their discriminative power and superior results in terms of the ability to establish semantic matches between images. Finally, our method also provides a visually intuitive way to navigate Web images and predicted annotations.

Acknowledgments.. We would like to thank Xerox Research Center Europe and ERC 677195-IDIU for supporting this research.

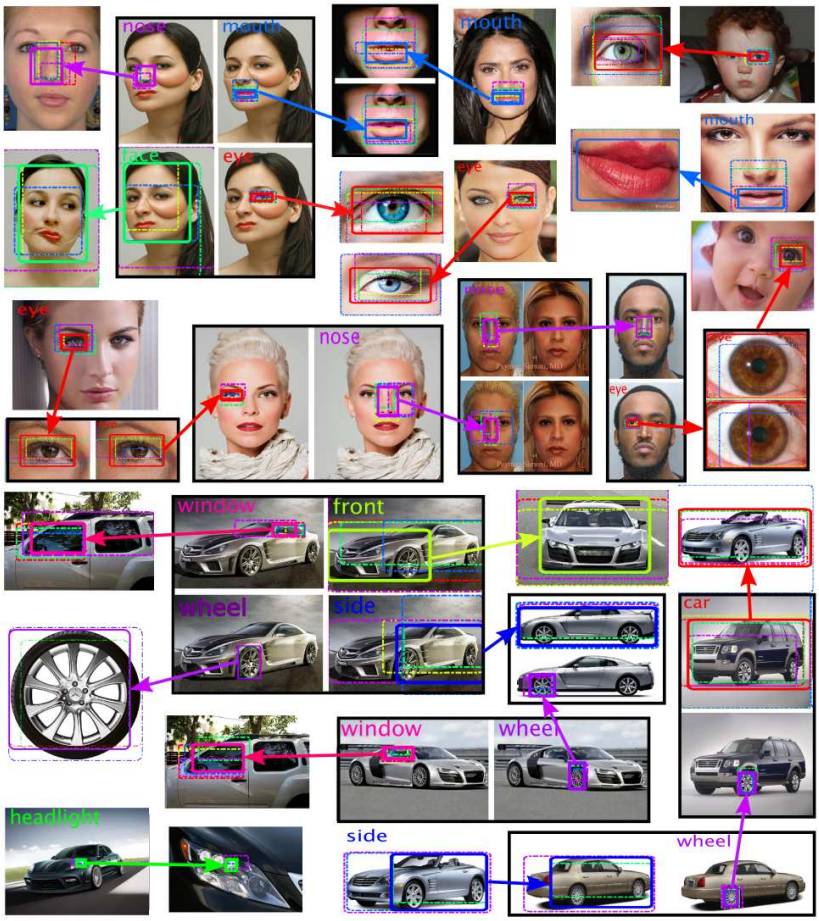


Fig. 4. Navigating the *visual semantic atlas*. Each pair of solid bounding boxes connected by an arrow denotes a preselected part bounding box (near the starting point of an arrow) as *detected by our algorithm* and the most similar semantic match (the endpoint of the arrow). The best matching bounding box is the detection with highest appearance-geometry descriptor similarity among *all the detections in our database* of web images. The dashed boxes denote anchors that contributed the most to the similarity. Please note that the matching gracefully occurs across scales.

Fig. 5. Visualisation of the **car** object class mid-level anchor detectors. Each bounding box corresponds to the max scoring location of an anchor. The locations of corresponding anchors are encoded in the same color across images.

A Appendix

Proof (Proof of Theorem 1). The function $\langle R, Q\rangle$ is the linear kernel, which is PD. This kernel is multiplied by the factor $-1/\bar{k}$ where $\bar{k}(R,Q) = \langle R, Q\rangle - \langle R, R\rangle - \langle Q, Q\rangle$; if this factor is also a PD kernel, then the result holds as the product of PD kernels is PD. According to Lemma 3.2 of [61], $-1/\bar{k}$ is PD if, and only if, $\bar{k}$ is strictly negative (point-wise) and conditionally definite positive (CDP). The first condition is part of the assumptions. To show the second condition that $\bar{k}$ is CDP pick n vectors $R_1, \ldots, R_n$ and real numbers $c_1, \ldots, c_n$ summing to zero $c_1 + \cdots + c_n = 0$; then

$$\sum_{ij} c_i \bar{k}(R_i, Q_i) c_j = \sum_{ij} c_i \langle R_i, Q_j\rangle c_j \geq 0$$

where we used the fact that the terms $\langle R_i, R_i\rangle$ cancel out and the fact that $\langle R_i, Q_j\rangle$ is PD.

B Mid-level anchor visualisations

Figures 5 to 7 contain visualisations of some of the learned unsupervised anchor detectors for the "car", "bus" and "face" classes respectivelly. Please refer to the figure captions for more details.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proc. NIPS. (2012)

Fig. 6. Visualisation of the **bus** object class mid-level anchor detectors. Each bounding box corresponds to the max scoring location of an anchor. The locations of corresponding anchors are encoded in the same color across images.

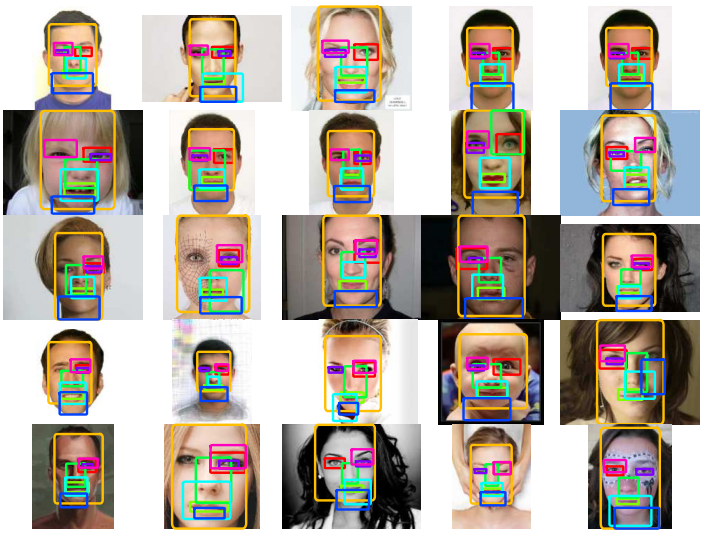


Fig. 7. Visualisation of the **face** object class mid-level anchor detectors. Each bounding box corresponds to the max scoring location of an anchor. The locations of corresponding anchors are encoded in the same color across images.

2. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M.A., Mikolov, T.: Devise: A deep visual-semantic embedding model. In: Proc. NIPS. (2013)
3. Karpathy, A., Joulin, A., Fei-Fei, L.: Deep fragment embeddings for bidirectional image-sentence mapping. In: Proc. NIPS. (2014)
4. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Proc. NIPS. (2014)

5. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. IJCV (2015) 1–42
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: Proc. CVPR. (2009)
7. Nguyen, M.H., Torresani, L., de la Torre, F., Rother, C.: Weakly supervised discriminative localization and classification: a joint learning process. In: Proc. ICCV. (2009)
8. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: Proc. ICCV. (2011)
9. Deselaers, T., Alexe, B., Ferrari, V.: Weakly supervised localization and learning with generic knowledge. Proc. ICCV (2012)
10. Wang, C., Ren, W., Huang, K., Tan, T.: Weakly supervised object localization with latent category learning. In: Proc. ECCV. (2014)
11. Hoffman, J., Guadarrama, S., Tzeng, E.S., Hu, R., Donahue, J., Girshick, R., Darrell, T., Saenko, K.: Lsda: Large scale detection through adaptation. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K., eds.: Proc. NIPS. (2014)
12. Hoffman, J., Pathak, D., Darrell, T., Saenko, K.: Detector discovery in the wild: Joint multiple instance and representation learning. In: Proc. CVPR. (2015)
13. Cinbis, R.G., Verbeek, J., Schmid, C.: Weakly Supervised Object Localization with Multi-fold Multiple Instance Learning. PAMI (September 2015)
14. Joulin, A., Tang, K., Fei-Fei, L.: Efficient image and video co-localization with frank-wolfe algorithm. In: Proc. ECCV. (2014)
15. Tang, K., Joulin, A., Li, L.J., Fei-Fei, L.: Co-localization in real-world images. In: Proc. CVPR. (2014)
16. Ali, K., Saenko, K.: Confidence-rated multiple instance boosting for object detection. In: Proc. CVPR. (2014)
17. Shi, Z., Hospedales, T., Xiang, T.: Bayesian joint modelling for object localisation in weakly labelled images. PAMI **37**(10) (Oct 2015) 1959–1972
18. Joulin, A., Bach, F., Ponce, J.: Efficient optimization for discriminative latent class models. In: Proc. NIPS. (2010)
19. Vicente, S., Rother, C., Kolmogorov, V.: Object cosegmentation. In: Proc. CVPR. (2011)
20. Joulin, A., Bach, F., Ponce, J.: Multi-class cosegmentation. In: Proc. CVPR. (2012)
21. Rubinstein, M., Joulin, A., Kopf, J., Liu, C.: Unsupervised joint object discovery and segmentation in internet images. Proc. CVPR (2013)
22. Song, H.O., Girshick, R., Jegelka, S., Mairal, J., Harchaoui, Z., Darrell, T.: On learning to localize objects with minimal supervision. In: Proc. ICML. (2014)
23. Li, Q., Wu, J., Tu, Z.: Harvesting mid-level visual concepts from large-scale internet images. In: Proc. CVPR. (2013)
24. Bilen, H., Vedaldi, A.: Weakly supervised deep detection networks. arXiv preprint arXiv:1511.02853 (2015)
25. Cho, M., Kwak, S., Schmid, C., Ponce, J.: Unsupervised Object Discovery and Localization in the Wild: Part-based Matching with Bottom-up Region Proposals. In: Proc. CVPR. (2015)
26. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google"s image search. In: Proc. ICCV. (2005) 1816–1823
27. Parkhi, O.M., Vedaldi, A., Zisserman, A.: On-the-fly specific person retrieval. In: International Workshop on Image Analysis for Multimedia Interactive Services, IEEE (2012)

28. Schroff, F., Criminisi, A., Zisserman, A.: Harvesting image databases from the web. In: Proc. ICCV. (2007)
29. Tsai, D., Jing, Y., Liu, Y., Rowley, H., Ioffe, S., Rehg, J.: Large-scale image annotation using visual synset. In: Proc. ICCV. (2011) 611–618
30. Kim, G., Xing, E.P.: On Multiple Foreground Cosegmentation. In: Proc. CVPR. (2012)
31. Chen, X., Gupta, A.: Webly supervised learning of convolutional networks. In: Proc. ICCV. (2015)
32. Chen, X., Shrivastava, A., Gupta, A.: Neil: Extracting visual knowledge from web data. In: Proc. ICCV. (2013)
33. Divvala, S.K., Farhadi, A., Guestrin, C.: Learning everything about anything: Webly-supervised visual concept learning. In: Proc. CVPR. (2014)
34. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. IJCV **61** (2003) 2005
35. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: Proc. CVPR. Volume 2. (June 2003) 264–271
36. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV **77**(1-3) (2008) 259–289
37. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. PAMI **32**(9) (2010) 1627–1645
38. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: Proc. ECCV. (2012)
39. Doersch, C., Gupta, A., Efros, A.A.: Mid-level visual element discovery as discriminative mode seeking. In: Proc. NIPS. (2013)
40. Juneja, M., Vedaldi, A., Jawahar, C.V., Zisserman, A.: Blocks that shout: Distinctive parts for scene classification. In: Proc. CVPR. (2013)
41. Endres, I., Shih, K.J., Jiaa, J., Hoiem, D.: Learning collections of part models for object recognition. In: Proc. CVPR. (2013)
42. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: Proc. ICCV. (2015)
43. Li, Y., Liu, L., Shen, C., van den Hengel, A.: Mid-level deep pattern mining. In: Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on, IEEE (2015) 971–980
44. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101–mining discriminative components with random forests. In: Proc. ECCV. (2014)
45. Sun, J., Ponce, J.: Learning dictionary of discriminative part detectors for image categorization and cosegmentation. Submitted to International Journal of Computer Vision, under minor revision (2015)
46. Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based R-CNNs for fine-grained category detection. In: Proc. ECCV. (2014)
47. Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., Yuille, A.: Detect what you can: Detecting and representing objects using holistic models and body parts. In: Proc. CVPR. (2014)
48. Wang, P., Shen, X., Lin, Z.L., Cohen, S., Price, B.L., Yuille, A.L.: Joint object and part segmentation using deep learned potentials. In: Proc. ICCV. (2015)
49. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence **89**(1-2) (1997) 31 – 71
50. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective search for object recognition. IJCV (2013)
51. Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., Kumar, N.: Localizing parts of faces using a consensus of exemplars. PAMI (2013)

52. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)
53. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. CVIU (2007)
54. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc. CVPR. (2014)
55. Deselaers, T., Alexe, B., Ferrari, V.: Localizing objects while learning their appearance. In: Proc. ECCV. (2010)
56. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: Proc. CVPR. (2009)
57. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
58. Cimpoi, M., Maji, S., Vedaldi, A.: Deep filter banks for texture recognition and segmentation. In: Proc. CVPR. (2015)
59. Kim, J., Liu, C., Sha, F., Grauman, K.: Deformable spatial pyramid matching for fast dense correspondences. In: Proc. CVPR. (2013)
60. Zhou, T., Jae Lee, Y., Yu, S.X., Efros, A.A.: Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In: Proc. CVPR. (2015)
61. Hein, M., Bousquet, O.: Hilbertian metrics and positive definite kernels on probability measures. In: Proc. AISTATS. (2005) 136–143

5

AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching

This work was presented as a *poster* presentation at the IEEE Conference on Computer Vision and Pattern Recognition, 2017 [Novotny et al. 2017a].

AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching

David Novotny[1,2] Diane Larlus[2] Andrea Vedaldi[1]

[1]Visual Geometry Group
Dept. of Engineering Science, University of Oxford
{david,vedaldi}@robots.ox.ac.uk

[2]Computer Vision Group
Xerox Research Centre Europe
diane.larlus@xrce.xerox.com

Abstract

Despite significant progress of deep learning in recent years, state-of-the-art semantic matching methods still rely on legacy features such as SIFT or HoG. We argue that the strong invariance properties that are key to the success of recent deep architectures on the classification task make them unfit for dense correspondence tasks, unless a large amount of supervision is used. In this work, we propose a deep network, termed AnchorNet, that produces image representations that are well-suited for semantic matching. It relies on a set of filters whose response is geometrically consistent across different object instances, even in the presence of strong intra-class, scale, or viewpoint variations. Trained only with weak image-level labels, the final representation successfully captures information about the object structure and improves results of state-of-the-art semantic matching methods such as the deformable spatial pyramid or the proposal flow methods. We show positive results on the cross-instance matching task where different instances of the same object category are matched as well as on a new cross-category semantic matching task aligning pairs of instances each from a different object class.

1. Introduction

Matching, i.e. the problem of establishing correspondences between images, is one of the tent-poles of image understanding. It is well known that, given matches between images of the same object or scene, it is possible to estimate 3D geometry (stereo and structure from motion) and motion (visual odometry, optical flow, and tracking). But matching can be applied to much more abstract levels of understanding as well. For example, aligning different object instances of the same type [32, 21] allows to discover analogies between objects, inducting abstractions such as object categories.

While reliable techniques exist for low-level matching, high-level matching of different object instances remains a heavily-researched topic. Most of the work in this area has focused on finding powerful geometric regularizers, such as hierarchical correspondences [35] or deformable spatial pyramids [32], to compensate for the still brittle visual descriptors. Surprisingly, even powerful convolutional neural network (CNN) descriptors have been found lacking for cross-instance matching [37, 21, 65], and in fact comparable or even inferior to old hand-crafted features such as SIFT [38] and HoG [11] for this task.

It is unclear why CNN representations, which perform well for many challenging vision tasks, including object de-

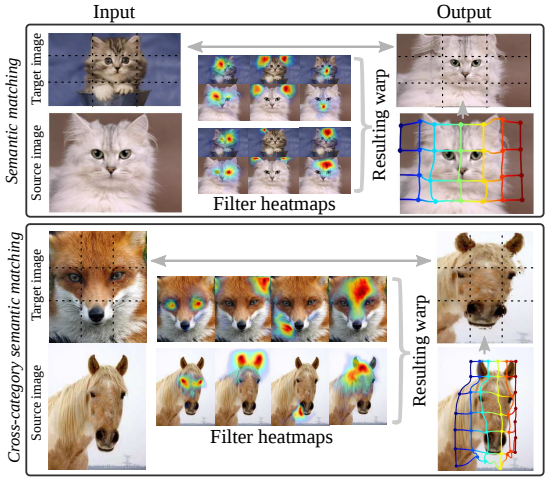
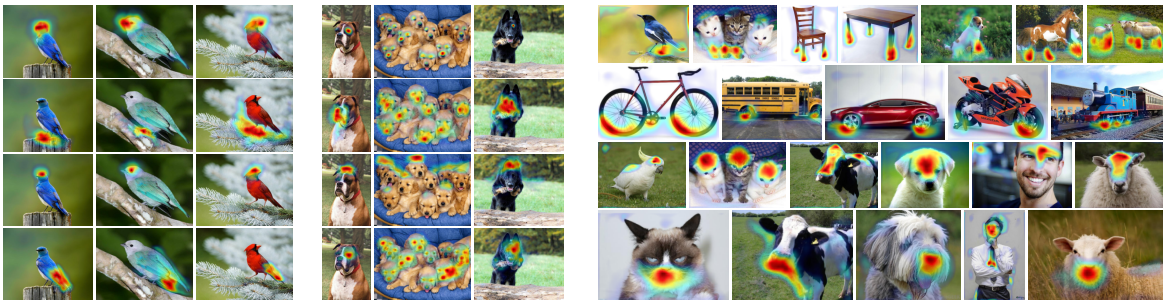


Figure 1: We propose AnchorNet, a novel deep architecture that produces an image representation which significantly improves state-of-the-art semantic matching methods. Key to its success is a set of filters with a sparse response that is geometrically consistent across different instances of a category or of two similar categories. Although these filters are learned in a weakly supervised manner (*i.e.* only image-level labels are used) they tend to anchor reliably on meaningful object parts.



(a) (b) (c)

Figure 2: Example responses of anchor filters discovered by the AnchorNet. (a), (b) show the class specific filters $F_k^{C_i}$ for bird and dog classes respectively while (c) depicts the class agnostic filters F_k^S across different categories (one filter per row).

tection [16] and segmentation [36], image captioning [59], and visual question answering [1], have not been found to work as well for cross-instance matching. Our hypothesis is that this is due to the fact that CNNs are trained on large datasets such as Imagenet ILSVRC [12] purely for the image classification task. By learning with the sole purpose of predicting a global image label, CNNs become insensitive to local details and geometry and hence work poorly for matching. This effect can be reversed by fine-tuning the model on substantial amounts of data strongly supervised with bounding box [16] or keypoint [9] annotations. While this allows to use CNNs as excellent object and keypoint detectors, it defeats the purpose of using CNN features as generic descriptors for *discovering* correspondences in an unsupervised manner, as matching requires.

In this paper, we address this issue by introducing a new deep architecture that can learn *representations that work well for cross-instance matching* (Figure 1), while using *exactly the same supervision* as traditional pre-training – namely image-level labels used to train categorizers on ILSVRC12 [12]. Using only image-level labels for matching amounts to weak supervision since the labels do not provide any information on the geometry of objects or scenes.

Our key insight is that a set of *diverse* and *sparse* filter responses provides a powerful representation for establishing matches. Convolutional features that respond sparsely on an image tend to automatically *anchor* to distinctive image structures such as semantic object parts. Further enforcing diversity of the filter bank responses results in a good coverage. This yields a unique description for *all* object fragments which is an essential property that enables reliable estimation of *dense* semantic correspondences.

We incorporate this idea by extracting from information-rich residual hypercolumns (section 3.1) a bank of distinctive and diverse filters with orthogonal responses (section 3.2; Figure 2). In this framework, which we call *AnchorNet*, geometric consistency is not imposed explicitly, but emerges spontaneously. We also show how to compress banks of class-specific filters into a class-agnostic bank (section 3.3) which works well for all classes.

Extensive experiments show that the proposed representation can be seamlessly leveraged by state-of-the-art semantic matching methods such as the Deformable Spatial Pyramid [32] or Proposal Flow [21] in order to improve their performance (section 4.1). For the first time, we also show that high-level correspondences can be established between objects of different categories, including new ones, unseen during the training of our network (section 4.2).

2. Related Work

Finding dense correspondences. The classical matching methods estimate very accurate pixel correspondences between two images of the same scene, in presence of moderate viewpoint variations [25, 39, 44]. Early methods use different hand-crafted features such as SIFT [38], HoG [11], SURF [4] or DAISY [54]. This task has many applications including stereo matching [44], optical flow [25, 61], or wide baseline matching [39, 63].

Recent works have generalized the notion of flow to image pairs that are only semantically related [34, 47, 32, 52, 21]. This requires handling a higher degree of variability in appearance. The semantic alignment task also finds many applications such as image completion [3], enhancement [20], or segmentation [34], and video depth estimation [30]. The SIFT Flow algorithm [35, 34] pioneered the idea of dense correspondences across different scenes and proposes a multi-resolution image pyramid and a hierarchical optimization algorithm for efficiency. This approach got extended by the Deformable Spatial Pyramid (DSP) algorithm [32] that introduced a multi-scale regularization with a hierarchically connected pyramid of graphs. The generalized deformable spatial pyramid [28] improves over DSP by enforcing additional spatial constraints at a significant computational cost. The Patch Match method [2] and its extension [3] target general purpose matching, including cross-instance matching. The method of [5] builds an exemplar-LDA classifier for every pixel to obtain dense correspon-

dences that improve the performance of scene flows. Proposal Flow [21] leverages the recent development in object proposals and uses local and geometric consistency constraints to establish dense semantic correspondences. Finally, WarpNet [29] learns correspondences by exploiting the relationships within a fine-grained dataset.

A few methods [26, 27, 46, 31, 41, 64] have posed the problem of finding correspondences as the joint alignment of multiple pairs of images, defining the task of collective alignment. These methods assume sets of images that share a category label and consistent viewpoints. The latest method in this field is FlowWeb [64], that builds a fully connected graph with images as nodes, and pairwise flow fields as edges. Yet, this method scales poorly with the size of the image collection, and it is not straightforward to establish pairwise alignments between new samples.

Deep features for correspondences. Long *et al*. [37] studied the application of CNN features pre-trained on large classification datasets for finding correspondences between object instances. They found that CNN features perform on par with hand-crafted alternatives such as SIFT for the weakly-supervised keypoint transfer problems, and can outperform them when keypoint supervision is available. This work paved the way to new deep architectures trained for finding dense correspondences between same object or scene instances [13, 60, 53]. Recently, Choy *et al*. [9] proposed a deep architecture that performs well at cross-instance alignment, but requires strong supervision in form of many keypoint matches.

The question of training deep features without keypoint annotations still remains unanswered, as state-of-the art semantic matching methods [32, 21] still rely on hand-engineered SIFT and HoG respectively.

3. Method

The output of a deep convolutional layer in a CNN is a tensor $\mathbf{x} \in \mathbb{R}^{H \times W \times D}$ of height H, width W, and with D feature channels. Thus, at each spatial location (u, v), one obtains a D-dimensional feature vector $\mathbf{d}_{uv} = (x_{uv1}, \ldots, x_{uvD})$. As noted by [10], such CNN feature vectors are analogous to hand-crafted dense descriptors like HoG and Dense-SIFT and can often be used as a plug-and-play replacement for the latter in applications. However, as noted in e.g. [37] and shown in the experiments, this substitution does not work well for cross-instance matching algorithms such as DSP [32] and Proposal Flow [21].

Since CNNs can be turned in excellent keypoint detectors by fine-tuning on data strongly annotated with keypoint labels [9, 55], the reason for this failure must be in the way most CNNs are pre-trained on image classification tasks. Note that collecting keypoint annotations for every category does not scale and defeats the purpose of cross-instance matching, which is to discover such correspondences automatically. As a solution, we propose a new architecture that, while using the same image-level supervision as the standard pre-training on the classification task, learns features with better geometric awareness.

Our method is motivated by a simple observation. Suppose that learning encourages a feature to respond very locally (ideally a point). A convolutional filter can do this only by responding to a visual structure that occurs uniquely in each image – hence the distinctive part or keypoint of an object. We call the latter the *anchoring principle*. A geometry-aware representation suitable for semantic matching should discover such a complete set of features that ultimately covers the whole object. We can do so by learning a bank of filters that respond to complementary image locations. We call this the *diversity principle*. Note that diversity indirectly encourages anchoring, as, if features respond to different parts of an image, they must also respond locally. Armed with these insights, we propose next an architecture termed AnchorNet that follows the two principles. We then show that these are sufficient to significantly boost the geometric awareness of the resulting features. A diagram of our network is presented in Figure 3.

3.1. Residual hypercolumns

We base our AnchorNet architecture on the powerful residual architectures of [24]. We select the ResNet50 model as a good compromise between speed and accuracy.

In order to improve the geometric sensitivity of the representation, we follow [22] and extract hypercolumns (HC). A HC $\mathbf{d}_{uv}$ at location (u, v) in the image is created by concatenating the convolutional feature responses at that location for different layers of the network. Recall that, in most CNN architectures, deeper features have reduced resolution; HC compensates for this by upsampling the responses to a common size before concatenation. We denote the resulting network $\mathbf{d} = \Phi(I)$, where I is the input image.

In more detail, we bilinearly upsample and concatenate the rectified outputs of the res2c, res4c and res5c layers [24] into a $56 \times 56 \times D$ hypercolumn tensor. Before concatenation, descriptors extracted at each layer are compressed by PCA to 256 dimensions (PCA is implemented as a 1×1 filter bank) and ℓ^2 normalized to balance their energies. This results in $D = 768$ dimensional HC vectors.

3.2. Learning anchoring features for an object type

The residual HC are high-capacity descriptors reflecting both high-level semantics as well as low-level image details. While this suggests that they should contain enough information for establishing matches, their direct utilization leads to suboptimal results. Thus, we train a set of 3×3 convolutional filters $F_1, ..., F_K$ that compress the HC responses into a compact set of *anchor filters* that are suitable

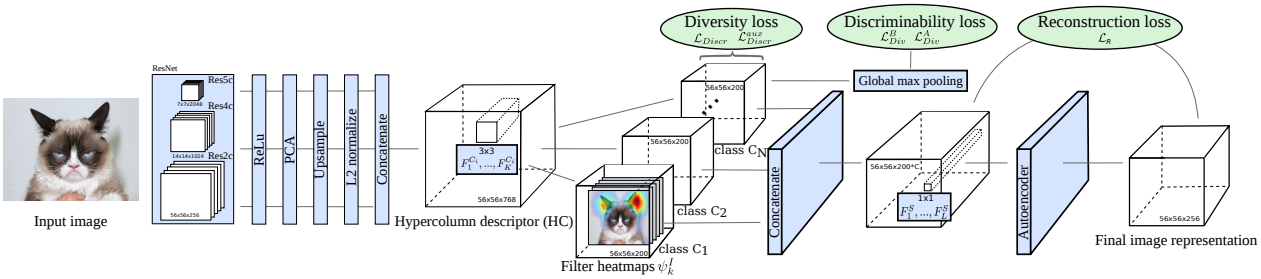


Figure 3: The proposed AnchorNet architecture. First, images are described using hypercolumn descriptors. Sparse filters are discovered for each category using a set of discriminability and diversity losses. Finally a denoising auto-encoder learns how to share these filters between categories, leading to a final category-agnostic representation generalizing to new classes.

for matching. To this end, we learn filters that satisfy two properties: discriminability and diversity.

Discriminability constraints. We start by learning filters F_k predictive of an object category. As a result, the filters tend to focus on relevant foreground objects, and rarely on the background. Without loss of generality, we first consider a binary setting where images I are either containing object instances of a single object category ($y_I = 1$) or irrelevant background ($y_I = -1$). We later extend to multiple categories in section 3.3.

Learning uses a large dataset of images with cheap-to-obtain image-level class labels. We follow common deep networks [51, 24, 33] and use ILSVRC 12 [48] for training. Discriminability is encouraged by minimizing the following loss function:

$$\mathcal{L}_{\text{Discr}}(I, y_I; \Phi, F) = -y_I \sum_{k=1}^{K} \text{gmax}\ \psi(F_k * \Phi(I)), \quad (1)$$

where $\Phi(I)$ denotes the HC tensor extracted from image I. The function $\psi(z) = \log(1 + \exp(z))$ is the smooth version of ReLU [42] and gmax is a global max-pooling operator. Using gmax in order to highlight discriminative features was done in [45]. While [45] finds a single geometry-invariant image pixel within an object instance, here we aim at learning a set of complimentary keypoint detectors sensitive to the geometry of the input.

Minimizing $\mathcal{L}_{\text{Discr}}$ identifies the strongest response of each filter F_k in the image and then enhances or suppresses it depending on whether the image contains the object. A disadvantage is that, due to the global max-pooling, the backpropagated signal is extremely sparse, which makes learning slow. To speed-up the convergence rate, we introduce a secondary loss function that, for negative images only, generates much denser gradients by using global average pooling (gavg) instead of max pooling:

$$\mathcal{L}^{\text{aux}}_{\text{Discr}}(I, y_I; \Phi, F) = \delta_{[y_I=-1]} \sum_{k=1}^{K} \text{gavg}\max\{0, F_k * \Phi(I)\}. \quad (2)$$

Using global average pooling is meaningful for the negative images, where all responses should be suppressed, but not for the positive ones, where only selected responses should be enhanced.

Diversity constraints. Discriminability alone encourages filters to respond to the object; however different filters may learn to respond to redundant highly-distinctive object parts. In order to obtain good coverage (and ultimately good anchoring), we require the filters F_k of one class to be active on *diverse* regions.

The diversity constraint is implemented by two *diversity losses* $\mathcal{L}^A_{\text{Div}}$ and $\mathcal{L}^B_{\text{Div}}$, encouraging orthogonality of the filters and of their responses, respectively. $\mathcal{L}^A_{Div}$ makes filters orthogonal by penalizing their correlations, as follows:

$$\mathcal{L}^A_{\text{Div}}(F) = \sum_{i \neq j} \left| \sum_p \frac{\langle F_i^p, F_j^p \rangle}{\|F_i^p\|_F \, \|F_j^p\|_F} \right| \quad (3)$$

where F_i^p is the column of filter F_i at spatial location p[1]. Note that orthogonal filters are likely to respond to different image structures, but this is not necessarily the case. Thus, we introduce a second term $\mathcal{L}^B_{\text{Div}}$ that directly decorrelates the filters' *response maps* $\psi^I_k \doteq \psi(F_k * \Phi(I))$:

$$\mathcal{L}^B_{\text{Div}}(I; \Phi, F) = \sum_{i \neq j} \left\| \frac{\langle \psi_i^I, \psi_j^I \rangle}{\|\psi_i^I\|_F \|\psi_j^I\|_F} \right\|^2. \quad (4)$$

This term is further regularized by smoothing the response maps $\psi^I_k \doteq g_\sigma * \psi(F_k * \Phi(I))$ prior to computing the loss $\mathcal{L}^B_{\text{Div}}$, where g_σ is a Gaussian kernel; this encourages filter responses to spread farther apart by dilating their activations. Note that inducing diversity among classifier prediction has been explored before [15, 19, 18, 49, 6], however none of these works consider diversity as a loss to train a deep representation as we propose.

Discussion. By making a large number of filters F_k both discriminative and diverse, our method indirectly encour-

[1] *i.e.* for our 3×3 filters F_i, $p \in \{1, 2, \ldots, 9\}$

ages them to become highly-specialized and hence to respond to unique parts of objects (the anchoring principle). This happens automatically, without enforcing such geometric properties explicitly. This intuition is strongly supported by our experiments. Examples of the filters learned for the bird and dog classes are presented in Figure 2 (a) and (b). It is apparent that filters fire on consistent object parts despite large intraclass variations, demonstrating the power of our formulation and its applicability to matching.

3.3. Class-agnostic representation

In the previous section we have defined category specific anchoring filters. In this section, we extend them to be generic to any category. This allows to use the same representation for every image, irrespective of its label, to match instances across different categories (*e.g.* dog vs cat), and to even handle new categories.

First, a filter bank $F_1^{C_i}, ..., F_K^{C_i}$ is learned for each object category $C_1, \ldots, C_N$ using the method above. Each object is learned by considering only images C_i of that object class and a common background class B. Since filters are not learned to discriminate between objects, and since the diversity losses are applied only *within* each bank, different filter banks can develop correlations. Figure 2 illustrates this by showing that filters learned for the "dog" and "bird" classes capture similar concepts such as eyes or nose.

We take advantage of the overlap between different banks by introducing a new bank of 1×1 filters $F_1^S, ..., F_L^S$ that projects the class-specific responses of the filters $F_1^{C_1}, \ldots, F_K^{C_N}$ to L general-purpose response maps applicable to objects of any class.

In order to learn the projections F^S end-to-end, we add a *denoising autoencoder* (DAE) [58] to our architecture. DAE minimizes the *reconstruction loss* $\mathcal{L}_R(F^S, \hat{\Gamma})$

$$\mathcal{L}_R(F^S, \hat{\Gamma}) = \mathcal{D}(\hat{\Gamma}, (F^S)^\top * F^S * c(\hat{\Gamma})) \qquad (5)$$

where $\mathcal{D}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a}/\|\mathbf{a}\| - \mathbf{b}/\|\mathbf{b}\|\|^2$ is the ℓ^2 distance between the ℓ^2 normalized tensors $\mathbf{a}$ and $\mathbf{b}$ and $(F^S)^\top$ is the *convolution transpose* operator [57]. Here $\hat{\Gamma} = \Gamma - \mu(\Gamma)$ denotes the stack of class-specific heatmaps $\Gamma = \text{stack}(\psi_{F_1^{C_1}}, \ldots, \psi_{F_K^{C_N}}) \in R^{W \times H \times (KN)}$ centered by removing their mean $\mu(\Gamma)$, estimated online during training. We have observed that centering followed by ℓ^2 normalization greatly improves the convergence properties of $\mathcal{L}_R$. Function $c(\mathbf{z})$ injects noise by randomly setting to zero 25% of the feature channels of the tensor $\mathbf{z}$.

The decorrelation loss eq. (3) is applied to the compression filters F^S as well in order to encourage their diversity.

Note that the reconstruction loss $\mathcal{L}_R$, when optimized end-to-end with the rest of the model, encourages the maps $\hat{\Gamma}$ to shrink (because, if $\hat{\Gamma} = 0$ everywhere, then the autoencoder has a trivial optimum). This is however prevented by the decorrelation losses $\mathcal{L}^A_{\text{Div}}$, $\mathcal{L}^B_{\text{Div}}$. $\mathcal{L}_R$ thus works as a regularizer enforcing part sharing. Examples of the learned class agnostic filters are in fig. 2 (c).

Denoising autoencoders have been used for domain adaptation before [7, 17]. In a similar spirit, the last part of our network transforms a set of class (domain) specific filters into a domain invariant representation that can accommodate for any class, even the one not seen during training.

Network training. AnchorNet is optimized with stochastic gradient descent (SGD) by minimizing the sum of the proposed losses $\mathcal{L}_{\text{Discr}}$, $\mathcal{L}^{\text{aux}}_{\text{Discr}}$, $\mathcal{L}^A_{\text{Div}}$, $\mathcal{L}^B_{\text{Div}}$ and $\mathcal{L}_R$, with mini batches of size 16, a learning rate of 10^{-2}, and a momentum of 0.0005. Parameters of the network are initialized with the ResNet50 model pre-trained on ILSVRC12. We use two-stage optimization to speed up the training process. First, the class-specific filters $F_i^{C_k}$ are trained on 4×10^4 training images independently for each object class C_k keeping the rest of the network parameters fixed. Then, we attach the autoencoder and the reconstruction loss to fine-tune all the network parameters end-to-end on 12×10^3 images. Further details are provided in the supplementary material.

4. Experiments

We thoroughly compare our method with existing techniques for semantic matching (section 4.1). Then, we assess how well our features allow to establish matches across images of different categories (section 4.2) which, to the best of our knowledge, was never demonstrated before.

Note that for all reported results, *training only uses ILSVRC12* [12] images and labels, where the categories are merged according to the PASCAL-ILSVRC class mapping from [12] (*e.g. sofa* is a merge of "studio couch" and "day bed"). In this manner, 231 ILSVRC classes are used as positive examples spread over the 20 PASCAL VOC classes; the remaining 769 classes are used to form the set B of negative (background) images. Even when we report results on one of the $N = 20$ PASCAL VOC [14] classes, *none* of the PASCAL VOC training data is used.

4.1. Dense pairwise semantic matching

We follow the standard practice [64, 21] of using a dataset with manually annotated semantic keypoints or regions and assess how well a semantic matching method in combination with different types of features transfers the annotations from an image to another. We experiment on three datasets following their evaluation protocol.

Compared methods. The most successful cross-instance matching methods include DSP [32] and Proposal Flow [21] (PF). In their original formulation, these methods performed best with the Dense SIFT [38] feature for DSP, and the whitened version of HoG [23] for PF. In the following

	mean	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	dog	horse	mbike	person	plant	sheep	sofa	table	train	tv
Pairwise alignment methods																					
DSP + ANet-class	**0.45**	**0.31**	**0.49**	**0.32**	0.53	**0.75**	**0.51**	**0.47**	0.23	0.53	**0.37**	0.20	0.33	**0.41**	**0.22**	**0.46**	**0.45**	**0.77**	0.45	0.48	**0.74**
DSP + ANet	**0.45**	0.29	0.47	0.29	0.52	0.73	0.50	0.46	**0.25**	0.53	**0.37**	**0.21**	**0.34**	0.39	0.20	0.44	0.45	**0.77**	0.45	0.51	**0.74**
DSP + HC	0.41	0.29	0.45	0.24	0.51	0.73	0.48	0.44	0.20	0.52	0.32	0.16	0.28	0.35	0.19	0.39	0.37	0.74	0.44	0.48	0.67
DSP + SIFT [32]	0.39	0.25	0.46	0.21	0.48	0.63	0.50	0.45	0.19	0.48	0.30	0.14	0.26	0.35	0.13	0.40	0.37	0.66	0.37	0.48	0.62
Proposal Flow + ANet-class	0.43	0.26	0.43	0.28	**0.54**	0.71	0.50	0.45	0.24	**0.54**	0.32	**0.21**	0.28	0.35	0.21	0.45	0.40	0.74	**0.46**	0.50	0.70
Proposal Flow + ANet	0.42	0.26	0.41	0.26	0.53	0.70	0.49	0.45	**0.25**	**0.54**	0.31	0.19	0.28	0.31	0.17	0.43	0.39	0.74	0.44	**0.52**	0.69
Proposal Flow + HC	0.42	0.26	0.42	0.26	**0.54**	0.70	0.50	0.45	0.23	0.53	0.32	0.18	0.27	0.32	0.18	0.43	0.38	0.74	0.45	0.51	0.64
Proposal Flow + HoG [21]	0.41	0.25	0.45	0.23	**0.54**	0.70	0.49	0.44	0.19	0.53	0.30	0.16	0.25	0.35	0.16	0.41	0.35	0.74	0.44	0.50	0.63
Baseline: NoFlow	0.39	0.27	0.40	0.22	0.50	0.73	0.46	0.42	0.20	0.51	0.30	0.15	0.25	0.32	0.18	0.38	0.34	0.74	0.44	0.47	0.64
Collective alignment methods																					
FlowWeb [64]	0.43	0.33	0.53	0.24	0.51	0.72	0.54	0.51	0.20	0.52	0.32	0.15	0.29	0.45	0.19	0.41	0.39	0.73	0.41	0.51	0.68

Table 1: Weighted IoU for pairwise **semantic part matching** on PASCAL Parts. The proposed methods are in **bold**.

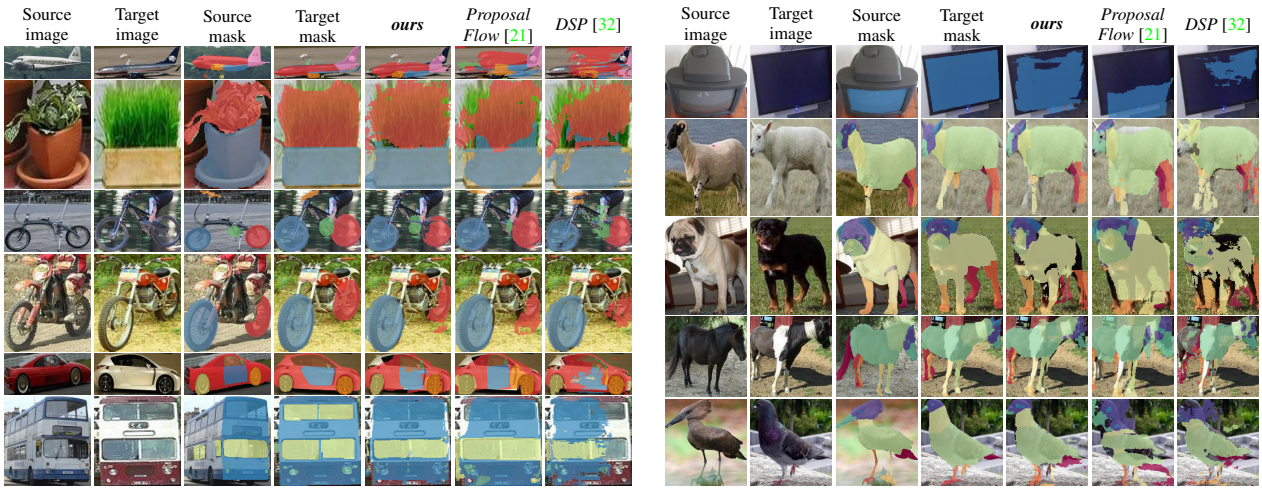


Figure 4: **Segmentation mask transfer** on PASCAL Parts for DSP+ANet (ours), Proposal Flow + HoG, and DSP + SIFT.

experiments, we replace these descriptors with our representation, as follows.

For DSP, the learned filter banks produce a dense field of feature vectors which are bilinearly upsampled to the original image size, ℓ_2 normalized and passed to DSP as a plug-and-play replacement of Dense SIFT. For PF, we mimic their use of HoG: every object proposal serves as a pooling region for the set of filter activations that are extracted once for every image. The pooling is performed by reading-off the filter activations inside the region and resizing them to 8×8 using bilinear interpolation. This tensor is then vectorized and ℓ^2 normalized to form the final descriptor of the proposal region. We use the variant of PF that extracts 1000 selective search boxes [56] per image. The rest of the matching procedure is identical to the original PF algorithm.

We compare both the class-agnostic (**ANet**) and class-specific (**ANet-class**) variants of our anchor filters. The class-agnostic variant uses the 256 dimensional features produced by the autoencoder filters F^S, whereas ANet-class uses the output of the class-specific filters F^{C_i} corresponding to a given PASCAL VOC object category C_i. Thus, ANet-class assumes knowledge of the object class label while ANet is universally applicable without requiring additional image-specific information. As baseline descriptors we consider SIFT, HoG and HC descriptors formed by concatenating the PCA projected layers of ResNet50 (res2c, res4c and res5c - section 3.1). We also report the NoFlow baseline that predicts zero-displacement for every pixel.

While we focus on pairwise matching, an alternative is to align many images together, known as co-alignment. Among various co-alignment methods, including [26, 46, 31], FlowWeb [64] is currently the state of the art. Due to its superior performance, we only report results for FlowWeb; however, while FlowWeb works very well, it is important to note that it is also substantially more expensive than pairwise matching, does not scale well and cannot accommodate for new image pairs.

Evaluation of segmentation masks transfer. We compare the various methods on the task of transferring semantic part segmentation masks, strictly following the protocol of [64]. Dense semantic matches, as determined by DSP or PF given a descriptor, are used to warp the part segmentation mask from a source to a target image. The matching quality is assessed as the average weighted intersection-over-union (IoU) between the predicted masks and the ground-truth

	mean	aero	bike	boat	bottle	bus	car	chair	mbike	sofa	table	train	tv
Pairwise alignment methods													
DSP + ANet-class	**0.24**	**0.23**	0.28	**0.06**	0.38	**0.44**	**0.39**	**0.14**	**0.19**	0.16	0.11	0.13	**0.41**
DSP + ANet	0.23	0.22	0.25	0.06	0.35	0.42	0.34	**0.14**	0.17	**0.17**	**0.13**	**0.14**	0.40
DSP + HC	0.20	0.20	0.23	0.05	**0.39**	0.36	0.25	0.10	0.15	0.12	0.10	0.12	0.28
DSP + SIFT [32]	0.18	0.17	**0.30**	0.05	0.19	0.33	0.34	0.09	0.17	0.12	0.09	0.12	0.18
Proposal Flow + ANet-class	0.17	0.17	0.21	0.05	0.25	0.26	0.27	0.10	0.14	0.12	0.07	0.10	0.24
Proposal Flow + ANet	0.16	0.16	0.19	0.05	0.22	0.26	0.25	0.10	0.12	0.11	0.05	0.12	0.23
Proposal Flow + HC	0.16	0.17	0.21	0.05	0.23	0.27	0.24	0.09	0.13	0.12	0.05	0.11	0.20
Proposal Flow + HoG [21]	0.17	0.20	0.26	0.05	0.20	0.31	0.29	0.10	0.17	0.13	0.05	0.13	0.21
Baseline: NoFlow	0.17	0.18	0.17	0.05	**0.39**	0.31	0.17	0.09	0.12	0.11	0.07	0.11	0.24
Collective alignment methods													
FlowWeb [64]	0.26	0.29	0.41	0.05	0.34	0.54	0.50	0.14	0.21	0.16	0.04	0.15	0.33

Table 2: PCK ($\alpha = 0.05$) for semantic keypoint transfer on the 12 rigid classes of the PASCAL Parts dataset.

	AuCs for PCR		
Matching \ Feature	**ANet-class**	**ANet**	HOG [21]
NAM: baseline	**0.41**	0.36	0.29
LOM: Proposal Flow	**0.46**	0.43	0.43

Table 3: **Region matching** on the PF dataset.

ones for different semantic parts. The results are reported in Table 1, qualitative results are provided in Figure 4.

We make the following observations. First, the ResNet50 features, perform at most marginally better, than SIFT or HoG, while both ANet and ANet-class features improve performance for both DSP (+6% IoU) and PF (+1% IoU). Second, the class-specific features ANet-class perform on par with the class-agnostic features ANet, demonstrating the ability of our domain generalization approach to compress the class-specific filters into the class-agnostic ones. Third, our features, in combination with DSP, exhibit the best average performance among all the compared methods. Remarkably, both ANet and ANet-class outperform all co-alignment methods, including FlowWeb [64], achieving state-of-the-art results on this dataset. This is an interesting finding as the co-alignment methods exploit the small viewpoint and appearance variations in order to improve pairwise alignments.

Evaluation of keypoint matching. We also evaluate performance on matching semantic keypoints. Corresponding annotations are provided by [62] for the 12 rigid PASCAL VOC categories. Similar to the previous section, we use the dataset from [64], and, strictly following their evaluation protocol, we assess the matching accuracy using PCK, setting the misalignment tolerance parameter α to 0.05.

Table 2 contains the results of this experiment. Our features improve the original DSP results by a large margin (+6% PCK), obtaining state-of-the-art results on this dataset among the pairwise alignment methods. Pairwise matching becomes in fact competitive with the results obtained by FlowWeb in co-alignment, although the latter use more information. Proposal Flow is generally weaker on this task and is not helped by the better features.

Matching Alg.	DSP			Proposal Flow			NoFlow
Feature	**ANet**	HC	SIFT	**ANet**	HC	HoG	-
PCK ($\alpha = 0.05$)	0.11	0.08	0.06	**0.13**	0.09	0.06	0.04
PCK ($\alpha = 0.1$)	0.24	0.18	0.12	**0.32**	0.25	0.18	0.12

Table 4: **Semantic matching** on the AnimalParts dataset. For each method, we report the average PCK over all possible 12x12 domain pairs. An overview of individual cross-category results can be found in Figure 5

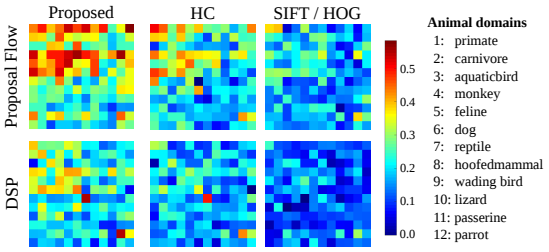


Figure 5: **Per-domain semantic matching** on the AnimalParts dataset. Cells are colored proportionally to the matching performance on a given animal class pair. Columns denote the source domains, rows the targets.

Evaluation of region matching. As a third benchmark dataset, we use the PF dataset and corresponding protocol as described in detail in [21]. The dataset contains 10 image sets of 4 object types and the task is to establish matches between annotated semantic regions within the image sets. We report region matching precision using the definitions specified in [21]. Table 3 contains the results obtained by using the code and data made available by [21].

We evaluate our deep features in combination with the two matching methods presented in [21]: the best performing local offset matching (LOM), and the naive appearance matching (NAM). ANet is compared with the best performing feature from [21], *i.e.* HoG [23]. We observe that using ANet-class features in combination with both matching methods (LOM, NAM) brings a significant performance improvement. Note in particular that ANet-class is sufficiently powerful to make the NAM baseline, which does not use any sophisticated geometric reasoning, competitive with the LOM+HoG, which uses geometric reasoning but

Figure 6: **Cross-class alignments** on the AnimalParts dataset. Given a target (top row) and source images (bottom row) we establish semantic correspondences between parts of animal classes. The alignment warps the source image into the target image. We compare Proposal Flow + ANet (ours - 2nd row) and Proposal Flow + HoG [21] (3rd row).

Source class	**mean**	bicycle ↓	mbike ↓	bus ↓	car ↓	bus ↓	dog ↓	cat ↓	sheep ↓	dog ↓	horse ↓	cow ↓	sheep ↓	cow ↓
Target class		mbike	bicycle	car	bus	car	cat	dog	dog	sheep	cow	horse	cow	sheep
DSP + ANet	**0.37**	**0.35**	**0.45**	**0.52**	**0.35**	0.36	**0.25**	0.25	**0.34**	**0.27**	**0.31**	**0.47**	**0.37**	**0.58**
DSP + HC	0.32	0.27	0.44	0.48	0.32	0.34	0.20	0.21	0.22	0.23	0.27	0.40	0.28	0.54
DSP + SIFT [32]	0.29	0.28	0.40	0.40	0.27	0.30	0.16	0.16	0.20	0.19	0.26	0.31	0.28	0.50
Proposal Flow + ANet	0.35	0.32	0.38	0.50	0.32	**0.37**	0.23	**0.27**	0.30	0.25	0.29	0.41	0.32	0.53
Proposal Flow + HC	0.33	0.31	0.34	0.49	0.29	0.35	0.22	0.24	0.28	0.23	0.28	0.41	0.32	0.53
Proposal Flow + HOG [21]	0.31	0.30	0.43	0.48	0.30	0.35	0.19	0.21	0.22	0.19	0.25	0.37	0.29	0.50
Baseline: NoFlow	0.27	0.26	0.44	0.35	0.26	0.25	0.17	0.18	0.22	0.17	0.22	0.29	0.26	0.49

Table 5: Weighted IoU for **cross instance semantic part matching** on PASCAL Parts.

handcrafted features (LOM+ANet-class is even better).

4.2. Generalization across categories

The previous section experimented on the task of aligning different object instances of the same category. Here, we depart from this scenario and consider instead *cross-category matching*, where correspondences are established between objects of different categories. To the best of our knowledge, this is the first time this task is considered.

For evaluation, we first use the PASCAL Parts [8] data from [64]. Parts with different location qualifiers are merged into one (e.g. "left-leg" and "right-leg" are merged into "leg") to ensure shareability across categories. Overall, there are 9 object categories and 13 shared part types.

Second, we consider the AnimalParts [43] dataset, introduced as a test-bed to study the transferability of semantic part detectors. Here, we reuse the dataset in order to assess transferability of ANet filters trained without explicit supervision. AnimalParts includes only a few part types ("eye" and "foot"), but a large number of different categories – 100 animals from the ILSVRC12 dataset. In order to present results compactly, animals are grouped in 12 families, based on the WordNet [40] hierarchy. For each pair of super-classes, 40 image pairs are randomly sampled for evaluation, resulting in ∼7K image pairs in total. PCK is computed for each pair of super-classes, and the results are averaged over such pairs. The class-specific ANet-class does not apply since the goal is to match across categories and most of these categories were not seen during training.

Tables 4 and 5 and Figure 5 show that ANet works substantially better than other matching methods. For the AnimalParts, the best results are obtained with Proposal Flow in combination with our features, with a 7% PCK improvement over the PF + HoG baseline ($\alpha = 0.05$). The fact that AnimalParts contains categories unseen at train time (*e.g.* reptiles) demonstrates the scalability and generalization of the proposed approach. For PASCAL Parts, similar to the intra-class matching experiment (section 4.1), DSP performs best. Here ANet attains a 16% relative improvement over the best previously published method (Proposal Flow + HoG). Figure 6 provides qualitative results.

5. Conclusion

In this paper we have examined the problem of dense semantic matching. Employing the concept of filter anchoring, we have designed a novel deep architecture, termed AnchorNet. Supervised with only image-level labels, AnchorNet automatically learns a set of filters which respond in a sparse and geometrically consistent manner across object instances. Thanks to these filters, our architecture produces powerful representations for image matching. We experimentally validate these features in conjunction with state-of-the art semantic matching methods attaining state-of-the-art performance on the segmentation transfer and keypoint matching tasks. Versatility of our representation has been

demonstrated on the new task of cross-category matching where we report positive results on two test-beds.

Acknowledgments. We would like to thank Xerox Research Center Europe and ERC 677195-IDIU for supporting this research.

AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching

Appendix

A. Learning details

In this section we provide additional details about the learning protocol of AnchorNet. Training converges after visiting 4×10^4 training samples (for each class) in stage 1 and 1.2×10^4 samples in stage 2 (two days on a single GPU NVIDIA Tesla M40). The learning rate was fixed to a value of 10^{-2} with the minibatch size of 16 and the momentum set to the standard value of 0.0005. The training data were augmented as in [24].

The losses were balanced as follows. The weights of $\mathcal{L}_{\text{Discr}}$ and $\mathcal{L}^{\text{Aux}}_{\text{Discr}}$ were set to 1 and 10 respectively. The weight of $\mathcal{L}_R$ was set to a higher value of 10^6 which is necessary due to the inhibition of the gradient by the ℓ_2 normalization which takes place just before computing $\mathcal{L}_R$. The weights of $\mathcal{L}^{A,B}_{\text{Div}}$ and $\mathcal{L}_{\text{R}}$ were set to be as high as possible (10^5) such that $\mathcal{L}^{A,B}_{\text{Div}} \approx \mathcal{L}_{\text{R}} \approx 0$ are treated approximately as hard constraints. Importantly, $\mathcal{L}_R$ is optimized only during visiting positive samples as reconstructing the activations of negative samples would waste the capacity of the autoencoder. During the first training stage, we sample positive and negative samples with equal probability. Furthermore, during stage 2, we ensure that the distribution of positive samples is uniform over the set of 20 Pascal categories. This causes the positive samples from any given object category to be $20\times$ less frequent than the negative samples. Hence, in order to rebalance losses in stage 2, we decrease the weights of negative samples by a factor of 20. Due to the fact that the gradients from $\mathcal{L}^{A,B}_{\text{Div}}$ exhibit high magnitudes, we decrease the learning rate on the layers bellow the first autoencoder layer by a factor of 10^4 during the second stage.

B. Additional experimental results

Tables F and G provide an extension of Tables 1 and 2 from the paper. On top of the features already provided in Tables 1 and 2, we include more baseline features: res4c and res5c, which are extracted from the ResNet50 architecture and the features from Simon et al. [50]. [50] selects part-like convolutional feature channels using a mixture of constellation models; however, if two different aspects are detected in two images, the set of common features is too sparse for matching. Thus, we converted their output to dense descriptors for use in DSP and PF by 1) modifying the HC from the ResNet50 architecture by retaining their part-like channels across all aspects (denoted as **Constellation-HC**) and 2) by backpropagating the part-like channel activations to the input image as they do, and using the image-level activations as dense descriptors (**Constellation-BP**).

Additionally, to quantify the impact of the diversity losses $\mathcal{L}_{\text{Div}}$, we also report the performance of the features produced by the ANet-class method optimized without the diversity losses with DSP used as the matching algorithm (**DSP + ANet-class w/o $\mathcal{L}_{\text{Div}}$**).

We observe that the res4c, res5c features as well as all the variants of the constellation features perform on par with the hypercolumn features (HC). The apparent drop in performance of DSP + ANet-class w/o $\mathcal{L}_{\text{Div}}$ compared to DSP + ANet-class highlights the contribution of the diversity losses.

C. Additional qualitative results

Segmentation transfer on PascalParts. Figure G complements Figure 4 from the paper and contains additional qualitative results for the segmentation transfer task on the PASCAL Parts dataset.

Semantic matching on AnimalParts. Figure H complements Figure 6 from the paper and contains additional qualitative results for the semantic matching task on the AnimalParts dataset.

References

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *ICCV*, 2015. 2

[2] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. 2009. 2

[3] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *Proc. ECCV*, 2010. 2

[4] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *CVIU*, 110(3):346–359, 2008. 2

[5] H. Bristow, J. Valmadre, and S. Lucey. Dense semantic correspondence where every pixel is a classifier. In *Proc. ICCV*, 2015. 2

[6] T. T. Cai and L. Wang. Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE IT*, 57:4680–4688, 2011. 4

[7] M. Chen, Z. Xu, K. Weinberger, and F. Sha. Marginalized denoising autoencoders for domain adaptation. In *Proc. ICML*, 2012. 5

	mean	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	dog	horse	mbike	person	plant	sheep	sofa	table	train	tv
Pairwise alignment methods																					
DSP + ANet-class	**0.45**	**0.31**	**0.49**	**0.32**	0.53	**0.75**	**0.51**	**0.47**	0.23	0.53	**0.37**	0.20	0.33	**0.41**	**0.22**	**0.46**	**0.45**	**0.77**	0.45	0.48	**0.74**
DSP + ANet-class w/o $\mathcal{L}_{\text{Div}}$	0.41	0.27	0.42	0.25	0.51	0.72	0.46	0.42	0.21	0.52	0.32	0.19	0.30	0.33	0.18	0.44	0.34	0.75	0.42	0.48	0.64
DSP + ANet	**0.45**	0.29	0.47	0.29	0.52	0.73	0.50	0.46	**0.25**	0.53	**0.37**	**0.21**	**0.34**	0.39	0.20	0.44	0.45	**0.77**	0.45	0.51	**0.74**
DSP + res4c	0.41	0.28	0.43	0.23	0.50	0.73	0.47	0.43	0.20	0.52	0.31	0.15	0.27	0.34	0.19	0.39	0.36	0.74	0.44	0.48	0.65
DSP + res5c	0.40	0.27	0.42	0.23	0.50	0.73	0.47	0.42	0.20	0.51	0.31	0.15	0.26	0.33	0.19	0.39	0.35	0.74	0.44	0.48	0.65
DSP + HC	0.41	0.29	0.45	0.24	0.51	0.73	0.48	0.44	0.20	0.52	0.32	0.16	0.28	0.35	0.19	0.39	0.37	0.74	0.44	0.48	0.67
DSP + SIFT [32]	0.39	0.25	0.46	0.21	0.48	0.63	0.50	0.45	0.19	0.48	0.30	0.14	0.26	0.35	0.13	0.40	0.37	0.66	0.37	0.48	0.62
DSP + Constellation-HC	0.40	0.28	0.42	0.23	0.50	0.73	0.47	0.42	0.20	0.52	0.31	0.15	0.27	0.34	0.19	0.38	0.36	0.74	0.44	0.48	0.65
DSP + Constellation-BP	0.40	0.27	0.41	0.23	0.50	0.73	0.46	0.42	0.20	0.51	0.31	0.15	0.26	0.33	0.18	0.38	0.35	0.73	0.44	0.47	0.64
Proposal Flow + ANet-class	0.43	0.26	0.43	0.28	**0.54**	0.71	0.50	0.45	0.24	**0.54**	0.32	**0.21**	0.28	0.35	0.21	0.45	0.40	0.74	**0.46**	0.50	0.70
Proposal Flow + ANet	0.42	0.26	0.41	0.26	0.53	0.70	0.49	0.45	**0.25**	**0.54**	0.31	0.19	0.28	0.31	0.17	0.43	0.39	0.74	0.44	**0.52**	0.69
Proposal Flow + res4c	0.42	0.27	0.44	0.26	0.54	0.70	0.50	0.45	0.23	0.53	0.32	0.18	0.28	0.33	0.17	0.44	0.39	0.74	0.45	**0.52**	0.66
Proposal Flow + res5c	0.39	0.23	0.34	0.25	0.53	0.70	0.47	0.43	0.22	0.52	0.30	0.18	0.26	0.27	0.17	0.41	0.38	0.73	0.45	0.49	0.60
Proposal Flow + HC	0.42	0.26	0.42	0.26	**0.54**	0.70	0.50	0.45	0.23	0.53	0.32	0.18	0.27	0.32	0.18	0.43	0.38	0.74	0.45	0.51	0.64
Proposal Flow + HoG [21]	0.41	0.25	0.45	0.23	**0.54**	0.70	0.49	0.44	0.19	0.53	0.30	0.16	0.25	0.35	0.16	0.41	0.35	0.74	0.44	0.50	0.63
Proposal Flow + Constellation-HC	0.40	0.26	0.39	0.25	0.53	0.68	0.48	0.43	0.21	0.52	0.30	0.17	0.26	0.31	0.15	0.42	0.37	0.72	0.44	0.50	0.62
Proposal Flow + Constellation-BP	0.39	0.25	0.38	0.23	0.53	0.68	0.47	0.41	0.20	0.51	0.29	0.16	0.25	0.30	0.15	0.41	0.35	0.71	0.43	0.49	0.60
Baseline: NoFlow	0.39	0.27	0.40	0.22	0.50	0.73	0.46	0.42	0.20	0.51	0.30	0.15	0.25	0.32	0.18	0.38	0.34	0.74	0.44	0.47	0.64
Collective alignment methods																					
FlowWeb [64]	0.43	0.33	0.53	0.24	0.51	0.72	0.54	0.51	0.20	0.52	0.32	0.15	0.29	0.45	0.19	0.41	0.39	0.73	0.41	0.51	0.68

Table F: Weighted IoU for pairwise **semantic part matching** (not to be confused with object or part detection or segmentation) on PASCAL Parts. The methods that use our proposed features are in **bold**.

	mean	aero	bike	boat	bottle	bus	car	chair	mbike	sofa	table	train	tv
Pairwise alignment methods													
DSP + ANet-class	**0.24**	**0.23**	0.28	**0.06**	0.38	**0.44**	**0.39**	**0.14**	**0.19**	0.16	0.11	0.13	**0.41**
DSP + ANet-class w/o $\mathcal{L}_{\text{Div}}$	0.17	0.19	0.18	0.06	0.31	0.31	0.18	0.10	0.13	0.12	0.08	0.12	0.24
DSP + ANet	0.23	0.22	0.25	0.06	0.35	0.42	0.34	**0.14**	0.17	**0.17**	**0.13**	**0.14**	0.40
DSP + HC	0.20	0.20	0.23	0.05	**0.39**	0.36	0.25	0.10	0.15	0.12	0.10	0.12	0.28
DSP + res4c	0.19	0.20	0.22	0.05	0.39	0.35	0.24	0.10	0.14	0.11	0.09	0.12	0.27
DSP + res5c	0.17	0.19	0.19	0.05	0.38	0.32	0.19	0.09	0.13	0.11	0.08	0.11	0.25
DSP + SIFT [32]	0.18	0.17	**0.30**	0.05	0.19	0.33	0.34	0.09	0.17	0.12	0.09	0.12	0.18
DSP + Constellation-HC [50]	0.18	0.20	0.21	0.05	0.39	0.33	0.20	0.10	0.13	0.12	0.09	0.12	0.26
DSP + Constellation-BP [50]	0.17	0.19	0.19	0.05	0.39	0.32	0.19	0.10	0.12	0.12	0.08	0.12	0.25
Proposal Flow + ANet-class	0.17	0.17	0.21	0.05	0.25	0.26	0.27	0.10	0.14	0.12	0.07	0.10	0.24
Proposal Flow + ANet	0.16	0.16	0.19	0.05	0.22	0.26	0.25	0.10	0.12	0.11	0.05	0.12	0.23
Proposal Flow + HC	0.16	0.17	0.21	0.05	0.23	0.27	0.24	0.09	0.13	0.12	0.05	0.11	0.20
Proposal Flow + res4c	0.17	0.19	0.24	0.05	0.23	0.28	0.27	0.09	0.15	0.12	0.05	0.13	0.21
Proposal Flow + res5c	0.11	0.13	0.11	0.04	0.21	0.21	0.19	0.07	0.08	0.08	0.05	0.09	0.14
Proposal Flow + HoG [21]	0.17	0.20	0.26	0.05	0.20	0.31	0.29	0.10	0.17	0.13	0.05	0.13	0.21
Proposal Flow + Constellation-HC [50]	0.14	0.18	0.17	0.04	0.19	0.25	0.20	0.08	0.12	0.10	0.05	0.10	0.17
Proposal Flow + Constellation-BP [50]	0.13	0.16	0.15	0.04	0.19	0.25	0.18	0.07	0.10	0.10	0.06	0.10	0.17
Baseline: NoFlow	0.17	0.18	0.17	0.05	**0.39**	0.31	0.17	0.09	0.12	0.11	0.07	0.11	0.24
Collective alignment methods													
FlowWeb [64]	0.26	0.29	0.41	0.05	0.34	0.54	0.50	0.14	0.21	0.16	0.04	0.15	0.33

Table G: PCK ($\alpha = 0.05$) for semantic keypoint transfer on the 12 rigid classes of the PASCAL Parts dataset.

[8] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proc. CVPR*, 2014. 8

[9] C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker. Universal correspondence network. In *Proc. NIPS*. 2016. 2, 3

[10] M. Cimpoi, S. Maji, and A. Vedaldi. Deep convolutional filter banks for texture recognition and segmentation. In *Proc. CVPR*, 2015. 3

[11] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005. 1, 2

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. CVPR*, 2009. 2, 5

[13] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazirbas, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proc. ICCV*, 2015. 3

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. 2010. 5

[15] A. Gane, T. Hazan, and T. S. Jaakkola. Learning with maximum a-posteriori perturbation models. In *Proc. AISTATS*, 2014. 4

[16] R. Girshick. Fast r-cnn. In *Proc. ICCV*, 2015. 2

[17] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proc. ICML*, 2011. 5

[18] A. Guzman-Rivera, D. Batra, and P. Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Proc. NIPS*, 2012. 4

[19] A. Guzman-Rivera, P. Kohli, D. Batra, and R. A. Rutenbar. Efficiently enforcing diversity in multi-output structured pre-

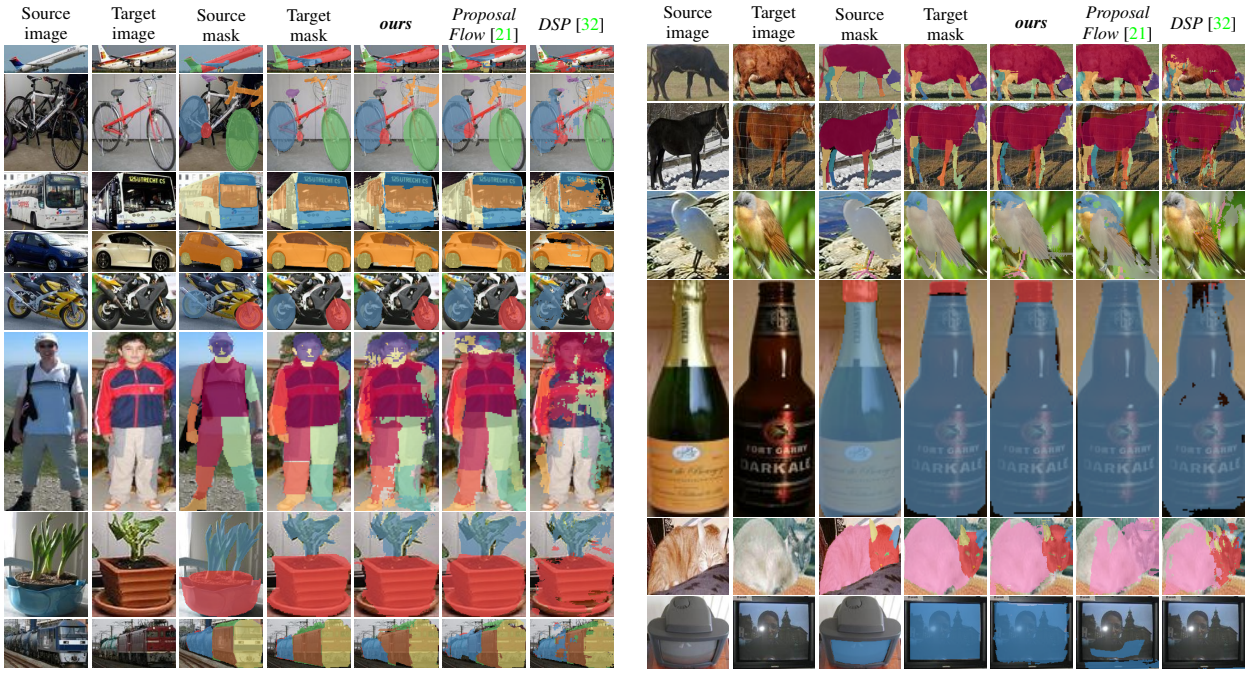


Figure G: **Segmentation mask transfer** on PASCAL Parts for DSP+ANet (ours), Proposal Flow + HoG, and DSP + SIFT.

diction. In *Proc. AISTATS*, 2014. 4

[20] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Non-rigid dense correspondence with applications for image enhancement. 2011. 2

[21] B. Ham, M. Cho, C. Schmid, and J. Ponce. Proposal flow. In *Proc. CVPR*, 2016. 1, 2, 3, 5, 6, 7, 8, 11, 12, 13

[22] B. Hariharan, P. A. Arbeláez, R. B. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proc. CVPR*, 2015. 3

[23] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *Proc. ECCV*, 2012. 5, 7

[24] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Proc. CVPR*, 2016. 3, 4, 10

[25] B. K. P. Horn and B. G. Schunck. Determining optical flow: A retrospective. *Artif. Intell.*, 59(1-2):81–87, 1993. 2

[26] G. B. Huang, V. Jain, and E. G. Learned-Miller. Unsupervised joint alignment of complex images. In *Proc. ICCV*, 2007. 3, 6

[27] G. B. Huang, M. A. Mattar, H. Lee, and E. G. Learned-Miller. Learning to align from scratch. In *Proc. NIPS*, 2012. 3

[28] J. Hur, H. Lim, C. Park, and S. C. Ahn. Generalized deformable spatial pyramid: Geometry-preserving dense correspondence estimation. In *Proc. CVPR*, 2015. 2

[29] A. Kanazawa, D. W. Jacobs, and M. Chandraker. WarpNet: Weakly supervised matching for single-view reconstruction. In *Proc. CVPR*, 2016. 3

[30] K. Karsch, C. Liu, and S. B. Kang. Depth extraction from video using non-parametric sampling. In *Proc. ECCV*, 2012. 2

[31] I. Kemelmacher-Shlizerman and S. M. Seitz. Collection flow. In *Proc. CVPR*, 2012. 3, 6

[32] J. Kim, C. Liu, F. Sha, and K. Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *Proc. CVPR*, 2013. 1, 2, 3, 5, 6, 7, 8, 11, 12

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, 2012. 4

[34] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994, 2011. 2

[35] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. Sift flow: Dense correspondence across different scenes. In *Proc. ECCV*, 2008. 1, 2

[36] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proc. CVPR*, 2015. 2

[37] J. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Proc. NIPS*, 2014. 1, 3

[38] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 1, 2, 5

[39] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. BMVC*, 2002. 2

[40] G. A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38:39–41, 1995. 8

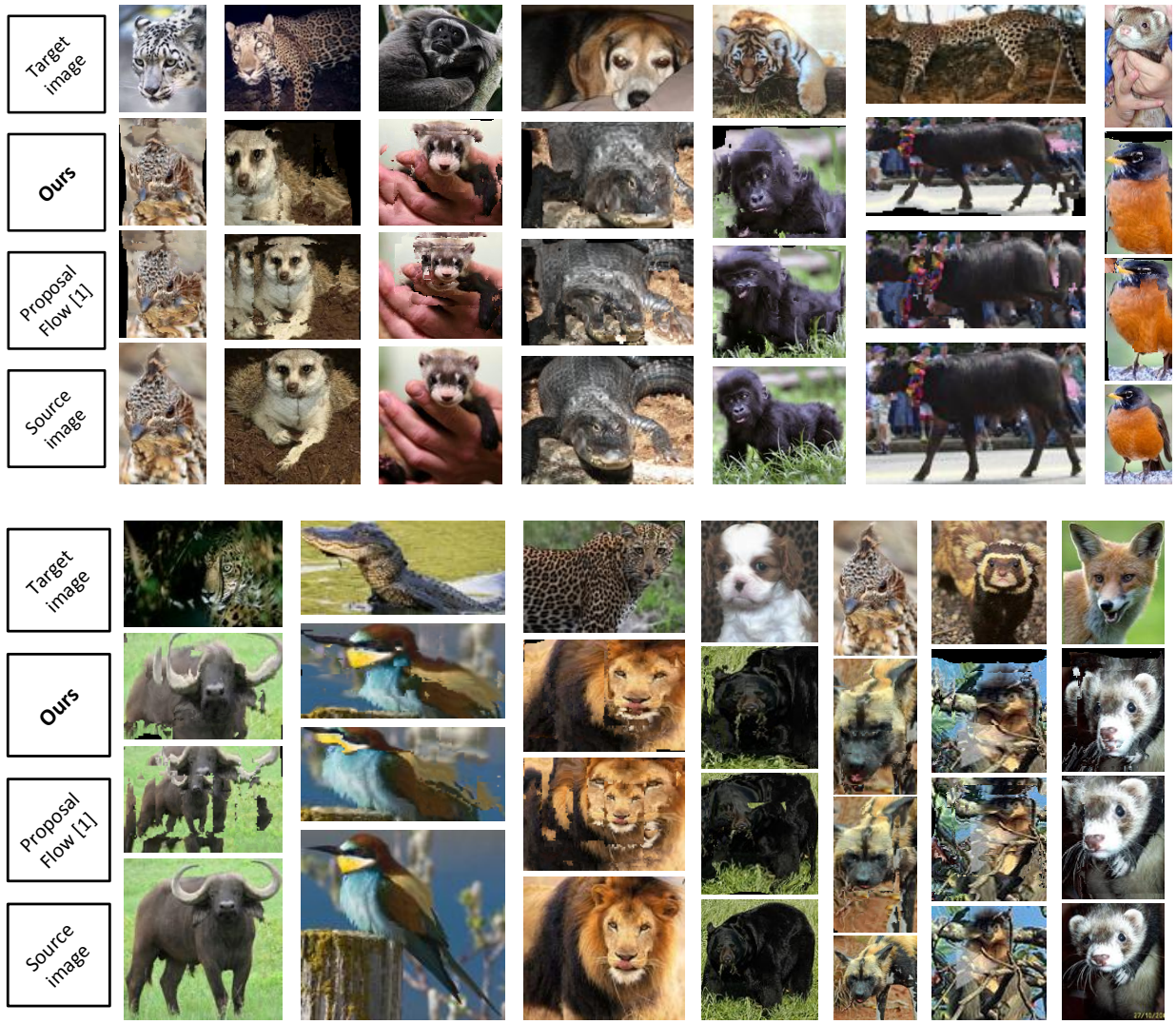


Figure H: **Cross-class alignments** on the AnimalParts dataset. Given a target (top row) and source images (bottom row) we establish semantic correspondences between parts of animal classes. The alignment warps the source image into the target image. We compare Proposal Flow + ANet (ours - 2nd row) and Proposal Flow + HoG [21] (3rd row).

[41] H. Mobahi, C. Liu, and W. T. Freeman. A compositional model for low-dimensional image set representation. In *Proc. CVPR*, 2014. 3

[42] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. ICML*, 2010. 4

[43] D. Novotny, D. Larlus, and A. Vedaldi. I have seen enough: Transferring parts across categories. In *Proc. BMVC*, 2016. 8

[44] M. Okutomi and T. Kanade. A multiple-baseline stereo. *PAMI*, 15(4):353–363, 1993. 2

[45] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free? - weakly-supervised learning with convolutional neural networks. In *Proc. CVPR*, 2015. 4

[46] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. In *Proc. CVPR*, 2010. 3, 6

[47] W. Qiu, X. Wang, X. Bai, A. Yuille, and Z. Tu. Scale-space sift flow. In *Proc. WACV*, 2014. 2

[48] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015. 4

[49] M. Schiegg, F. Diego, and F. A. Hamprecht. Learning diverse models: The coulomb structured support vector machine. In *Proc. ECCV*, 2016. 4

[50] M. Simon and E. Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional net-

works. In *Proc. ICCV*, 2015. 10, 11

[51] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. 2015. 4

[52] M. Tau and T. Hassner. Dense correspondences across scenes and scales. *PAMI*, 38(5):875–888, 2016. 2

[53] J. Thewlis, S. Zheng, P. Torr, and A. Vedaldi. Fully-trainable deep matching. In *Proc. BMVC*, 2016. 3

[54] E. Tola, V. Lepetit, and P. Fua. DAISY: An Efficient Dense Descriptor Applied to Wide Baseline Stereo. *PAMI*, 32(5):815–830, 2010. 2

[55] S. Tulsiani and J. Malik. Viewpoints and keypoints. In *Proc. CVPR*, 2015. 3

[56] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 104:154–171, 2013. 6

[57] A. Vedaldi and K. Lenc. Matconvnet – convolutional neural networks for matlab. In *Proc. ACM Int. Conf. on Multimedia*, 2015. 5

[58] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proc. ICML*, 2008. 5

[59] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proc. CVPR*, 2015. 2

[60] J. Žbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. 17(1):2287–2318, 2016. 3

[61] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *Proc. ICCV*, 2013. 2

[62] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE Winter Conference on Applications of Computer Vision*, 2014. 7

[63] H. Yang, W.-Y. Lin, and J. Lu. Daisy filter flow: A generalized discrete approach to dense correspondences. In *Proc. CVPR*, 2014. 2

[64] T. Zhou, Y. Jae Lee, S. X. Yu, and A. A. Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *Proc. CVPR*, 2015. 3, 5, 6, 7, 8, 11

[65] T. Zhou, P. Krähenbühl, M. Aubry, Q. Huang, and A. A. Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Proc. CVPR*, 2016. 1

6

Self-supervised Learning of Geometrically Stable Features Through Probabilistic Introspection

This work was presented as a *spotlight* presentation at the IEEE Conference on Computer Vision and Pattern Recognition, 2018 [Novotny et al. 2018a].

Self-supervised Learning of Geometrically Stable Features Through Probabilistic Introspection

David Novotny[1,2,*] Samuel Albanie[1,*] Diane Larlus[2] Andrea Vedaldi[1]
[1]Visual Geometry Group
Dept. of Engineering Science, University of Oxford
{david,albanie,vedaldi}@robots.ox.ac.uk
[2]Computer Vision Group
NAVER LABS Europe
diane.larlus@naverlabs.com

Abstract

Self-supervision can dramatically cut back the amount of manually-labeled data required to train deep neural networks. While self-supervision has usually been considered for tasks such as image classification, in this paper we aim at extending it to geometry-oriented tasks such as semantic matching and part detection. We do so by building on several recent ideas in unsupervised landmark detection. Our approach learns dense distinctive visual descriptors from an unlabeled dataset of images using synthetic image transformations. It does so by means of a robust probabilistic formulation that can introspectively determine which image regions are likely to result in stable image matching. We show empirically that a network pre-trained in this manner requires significantly less supervision to learn semantic object parts compared to numerous pre-training alternatives. We also show that the pre-trained representation is excellent for semantic object matching.

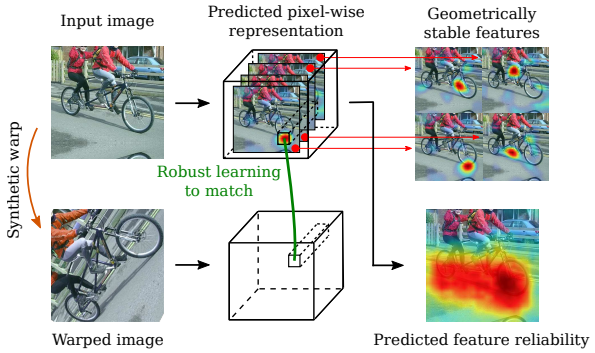


Figure 1. Our approach leverages correspondences obtained from synthetic warps in order to self-supervise the learning of a dense image representation. This results in highly localized and geometrically stable features. The use of a novel robust probabilistic formulation allows to additionally predict a pixel-level confidence map that estimates the matching ability of these features.

1. Introduction

One factor that limits the applicability of deep neural networks to many practical problems is the cost of procuring a sufficient amount of supervised data for learning. This explains the increasing interest in techniques that can learn good deep representations *without the use of manual supervision*. Methods that rely on self-supervision [7, 26, 30], in particular, can initialize deep neural networks from unlabeled image collections. The resulting pre-trained networks can then be fine-tuned to solve a desired task with far fewer manual annotations than would be required if they were trained from scratch.

While several authors have looked at self-supervision for tasks such as image classification and segmentation, less work has been done on tasks that involve understanding the geometric properties of object categories. In this paper, therefore, we propose a self-supervised pre-training technique that obtains image representations suitable for geometry-oriented tasks. We consider two representative problems: semantic part detection and semantic matching, both of which help to characterize the geometric structure of objects.

Our specific goal is to pre-train convolutional neural networks suitable for such geometry-oriented tasks given only a dataset of images of one or more object categories *with no bounding box, part or other types of geometric annotations*. Our approach is based on three ideas. First, we configure the network to compute a dense field of visual descriptors. These descriptors are learned to match corresponding object points in different images using a pairwise loss formulation. However, since no labels are given, correspondences between images are unknown. Thus, the second idea is to generate image pairs for which correspondences are known by means of *synthetic warps* [17, 31, 34, 35]. Learning from this data results in visual descriptors that are invariant to image deformations, but that may not be consistent across

*Authors contributed equally.

intra-class variations. The authors of [35] suggest that intra-class generalization can be achieved by limiting the descriptor dimensionality. However, we found this approach to be too fragile to handle complex 3D object categories, particularly when many landmarks can be occluded in different views. This contrasts with other recent approaches such as AnchorNet [27], which can learn landmarks more robustly, albeit with reduced geometric accuracy.

Seeking to retain the robustness of methods such as AnchorNet [27] while incorporating a geometric prior such as [35], we propose to trade-off robustness for a higher dimensionality of the descriptors. We further improve robustness by casting learning into a probabilistic formulation, our third idea. This formulation allows the network to explicitly learn, along with the visual descriptors, an estimate of their expected matching reliability. In this manner, the network learns failure modalities, such as extracting descriptors in correspondence of background regions instead of the object or occlusions.

The resulting formulation is able to pre-train excellent networks for semantic matching and semantic part detection. This is demonstrated empirically by means of thorough experiments against a range of baselines on standard benchmark datasets. For semantic matching, our results outperform [27] and [35] that use a comparable level of supervision and are on par with the fully supervised method of [11]. For part detection, we consider a few-shot keypoint detection task and show that our method performs better than all competitors when few annotations are available.

The rest of the manuscript is organized as follows. Section 2 discusses related work, section 3 presents the technical details of our method, section 4 conducts the experimental evaluation, and section 5 summarizes our findings.

2. Related Work

Learning features for geometric tasks. Hand-crafted features such as SIFT [24], DAISY [41], or HOG [6], initially designed for geometrical tasks such as matching-based retrieval [33], stereo matching [29], or optical flow [14] formed the gold standard until very recently due to their appealing properties such as repeatability.

Dense semantic matching methods, pioneered by SIFT Flow [21] are designed to deal with more variability in appearance and create dense correspondences across different scenes. Following the success of CNN architectures for recognition tasks like image classification [20], these architectures have been used as feature extractors for other tasks, including semantic matching. Yet, without any further training, they have been shown not to improve over hand-engineered features for geometric tasks [23, 10] and most approaches still combine hand-crafted features and spatial regularization [3, 15, 19, 21, 45]. To overcome this, deep features have been retrained for geometric tasks [4, 45, 11]. Choy *et al*. [4] combine a fully convolutional architecture with a contrastive loss and train with a large number of annotations. Zhou *et al*. [46] require 3D models to link correspondences between images and rendered views. Han *et al*. [11] follow Proposal Flow [10] and replace the hand-crafted features with features trained end-to-end with a large amount of annotations.

Training geometry-aware features without costly annotations such as keypoints or 3D models has only been seldomly studied [27, 34, 35, 31]. The AnchorNet approach [27] builds discriminative parts that match different object instances as well as different object categories using only image-level supervision. Other methods have proposed to replace costly manual annotations by synthetically generating image pairs [34, 35, 31]. Thewlis *et al*. [34] show that placing constraints on matching builds object landmarks that are not only consistently detected across the deformation of a current instance, but also across instances. This work was extended to a dense formulation [35], embedding objects on a sphere. Although this works well for faces, such an approach seems less appropriate for objects with a complex 3D shape. Rocco *et al*. [31] propose a Siamese architecture for geometric matching, composed of a feature extraction part and a matching architecture that is used to predict the parameters of a synthetic transformation applied to the input image. Artificial correspondences were also used in [17] for fine-grained categories.

Keypoint detection. Keypoint detection has been extremely well studied for the case of humans [16, 42, 9, 1] and recent approaches have leveraged deep architectures [37, 36]. Only a few works have considered keypoint detection for generic categories [13, 23, 40, 38]. These methods require large training sets and none of them has considered a few-shot learning scenario.

3. Method

Our aim is to learn a neural network for object part detection and semantic matching. Furthermore, we assume that only a small number of images annotated with information relevant to these tasks is available, but that images labeled only with the presence of a given object category are plentiful. Thus, our goal is to develop a self-supervised method that can use such image-level annotations to pre-train a network that captures the object geometry.

Formally, let $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ be a collection of N unlabeled images $\mathbf{x}_i \in \mathbb{R}^{H \times W \times 3}$ of one or more object categories and let $\phi : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times C}$ be a deep neural network extracting a dense set of feature vectors from the image. We will use the symbol $\phi(\mathbf{x})_u \in \mathbb{R}^C$ to denote the feature vector extracted at lo-

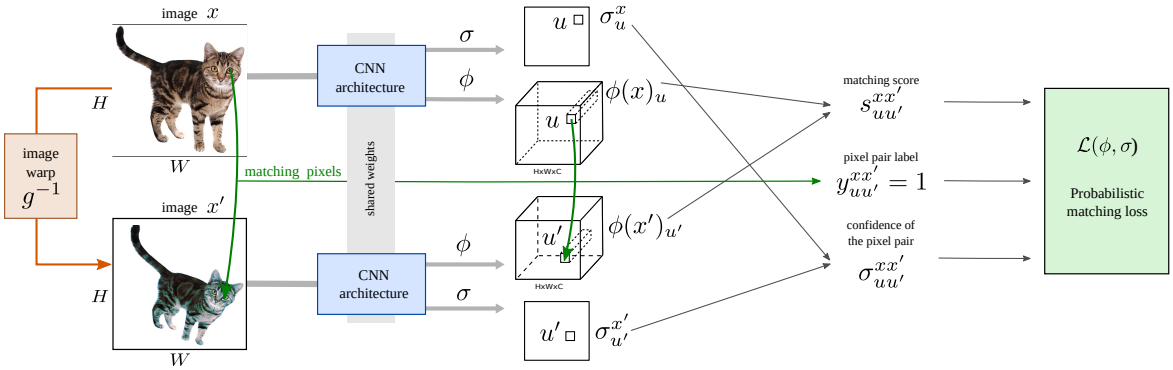


Figure 2. **Overview of our approach**. Image x is warped into image x' using the transformation g^{-1}. Pairs of pixels and their labels (encoding whether they match or not according to g^{-1}) are used together with a probabilistic matching loss to train our architecture that predicts i) a dense image feature $\phi(x)$ and ii) a pixel level confidence value $\sigma(x)$.

cation[1] $u \in \{1, \dots, H\} \times \{1, \dots, W\}$, namely:

$$\forall c \in \{1, \dots, C\} : \quad [\phi(\mathbf{x})_u]_c = [\phi(\mathbf{x})]_{uc}.$$

Each vector $\phi(\mathbf{x})_u$ can be thought of as a descriptor of the image appearance around location u. Since our aim is to recognize and match object parts, we would like such descriptors to be *characteristic of specific object landmarks*.

In a supervised setting, one is given the identity of the object part found at each location u and can use this information to learn the descriptors. However, in our case this information is *not* available, so we must resort to a different supervisory signal. We do so by constraining descriptors to be invariant (section 3.1) and discriminative (section 3.2) with respect to synthetic image transformations, and make this robust using a form of probabilistic introspection (section 3.3). The resulting learning objective is given in section 3.4 and further discussed in section 3.5. Figure 2 provides an overview of the overall approach.

3.1. Invariant description

We say that locations u and u' in image $\mathbf{x}$ and $\mathbf{x}'$ *correspond* if they are projection of the same 3D object point. For object categories, we define correspondences by analogy (such as being centered on the right eyes of two animals).

The *invariance* condition states that the descriptors computed at corresponding image locations u and u' should be identical:

$$\phi(\mathbf{x})_u = \phi(\mathbf{x}')_{u'} \tag{1}$$

While correspondences are not known for arbitrary images in the database $\mathcal{X}$ (short of providing manual annotations), we can at least *synthetically generate* such examples. To this end, let $g : \mathbb{R}^2 \to \mathbb{R}^2, u \mapsto u' = g(u)$ be a random image warp and let $\mathbf{x}' = \mathbf{x} \circ g^{-1}$ be the image obtained by warping $\mathbf{x} \in \mathcal{X}$ accordingly.[2] Then, constraint (1) can be rewritten as:

$$\forall g, u : \quad \phi(\mathbf{x})_u = \phi(\mathbf{x} \circ g^{-1})_{g(u)} \tag{2}$$

While the network ϕ should satisfy constraint (2), the latter is insufficient to characterize good descriptors as it can be trivially satisfied by making all descriptors identical. The missing ingredient is that the descriptors should also *uniquely identify* a specific object point. Building this additional constraint into the model is discussed in the next section.

3.2. Informative invariant description

Invariance (2) must be paired with the fact that descriptors should be able to robustly distinguish between *different* object points. To encode such a constraint, we note first that it does not make sense to check for exact descriptor equality or inequality as literally suggested by eq. (2). Instead, descriptors are compared continuously by considering a *matching score*. We define the latter to be their rectified inner product

$$s_{uu'}^{\mathbf{x}\mathbf{x}'} = \max\{0, \ \langle \phi(\mathbf{x})_u, \phi(\mathbf{x}')_{u'} \rangle\}. \tag{3}$$

In order to guarantee that this score is maximum when a descriptor is compared to itself ($s_{uu'}^{\mathbf{x}\mathbf{x}'} \leq 1, s_{uu}^{\mathbf{x}\mathbf{x}} = 1$), descriptors are L^2 normalized, so that

$$\|\phi(\mathbf{x})_u\|_2 = 1.$$

The inner product is rectified because, while it makes sense for similar descriptors to be parallel, dissimilar descriptors should be orthogonal rather than anti-correlated.

Next, in order to encode invariance and discriminability together, we note that each pair of points (u, u') may or may

[1] In our implementation, features are extracted at a lower resolution than the input image, but for clarity we ignore this difference in the notation.

[2] Here $\mathbf{x}'$ is obtained from $\mathbf{x}$ using inverse warp.

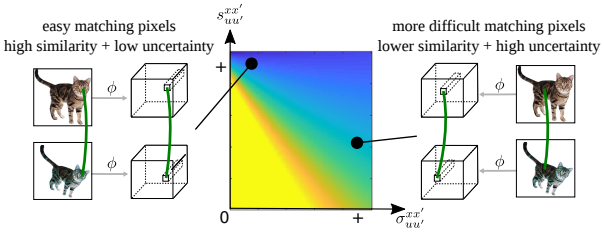


Figure 3. **Illustration of the probabilistic loss.** The plot shows values of the loss for positive pairs ($y_{xx'} = 1$, bluer means a smaller loss) as a function of the similarity between the pixel representations $s_{uu'}^{xx'}$ and the uncertainty $\sigma_{uu'}^{xx'}$ who's inverse ${\sigma_{uu'}^{xx'}}^{-1}$ corresponds to the confidence. The model has several options for decreasing the loss: (1) increasing the similarity while keeping confidence unchanged, (2) decreasing the confidence while keeping similarity and (3) increasing both similarity and confidence.

not represent a valid correspondence for a given image pair $(\mathbf{x}, \mathbf{x}')$. This is captured by a label $y_{uu'}^{\mathbf{x}\mathbf{x}'} \in \{-1, 0, +1\}$, where $+1$ indicates a valid correspondence , -1 an invalid one, and 0 a "borderline" case to be ignored. Given the labels (defined from the synthetic warps in eq. (7)), one can define a *matching loss* $\ell_{uu'}^{\mathbf{x}\mathbf{x}'}$:

$$\ell_{uu'}^{\mathbf{x}\mathbf{x}'} = \begin{cases} 1 - s_{uu'}^{\mathbf{x}\mathbf{x}'} & y_{uu'}^{\mathbf{x}\mathbf{x}'} = 1, \\ 0 & y_{uu'}^{\mathbf{x}\mathbf{x}'} = 0, \\ s_{uu'}^{\mathbf{x}\mathbf{x}'} & y_{uu'}^{\mathbf{x}\mathbf{x}'} = -1. \end{cases} \tag{4}$$

However, ℓ cannot be satisfied for all possible choices of image and pixel pairs $(\mathbf{x}, \mathbf{x}')$ and (u, u'). For example, an object point may be occluded, a pixel may belong to the background, or the match may just be too difficult for the model to express adequately. This problem is addressed in the next section.

3.3. Probabilistic introspection

In order to handle difficult or impossible matches in the loss function, we do not resort to heuristics such as using robust versions of the loss (4), but rather task the neural network with *predicting when descriptors are unreliable*. In order to do so, inspired by [28, 18], the network is modified to compute an additional scalar value $\sigma_u^{\mathbf{x}} \in \mathbb{R}^+$ for each pixel expressing uncertainty about the quality of the descriptor extracted at u and its consequent ability to establish a reliable match. Importantly, this belief is estimated from each image independently *before* matching occurs. In this manner, $\sigma_u^{\mathbf{x}}$ can be interpreted as an assessment of the informativeness of the image region that is used to compute the descriptor.

In more detail (and dropping the superscript $\mathbf{x}\mathbf{x}'$ for simplicity), we define a distribution over matching scores $p(s_{uu'}|y_{uu'}, \sigma_{uu'})$ conditioned on the average predicted uncertainty $\sigma_{uu'} = (\sigma_u + \sigma_{u'})/2$ and on whether pixels are in correspondence or not. The distribution is given by:

$$p(s_{uu'}|y_{uu'}, \sigma_{uu'}) = \frac{1}{\mathcal{C}(\sigma_{uu'})} \exp \frac{1 - \ell_{uu'}(s_{uu'}, y_{uu'})}{\sigma_{uu'}}, \tag{5}$$

where $\mathcal{C}(\sigma_{uu'})$ is a normalization constant ensuring that $p(s_{uu'}|y_{uu'}, \sigma_{uu'})$ integrates to one.

To understand expression (5), note that, due to the fact that $s_{uu'} \in [0, 1]$ and to the particular form (4) of the function $\ell_{uu'}$, $\mathcal{C}(\sigma_{uu'})$ is finite and does not depend on $y_{uu'}$. When the model is confident of the fact that descriptors $\phi(\mathbf{x})_u$ and $\phi(\mathbf{x}')_{u'}$ match or not, the value $\sigma_{uu'}$ is small. In this case, the distribution (5) has a sharp peak around $s_{uu'} = 1$ or $s_{uu'} = 0$, depending on whether pixels (u, u') are in correspondence or not. On the other hand, when the model is less certain about the quality of the descriptors, the score distribution is more spread.

3.4. Learning objective

It is now possible to describe the overall learning objective for our method. The models ϕ and σ are learned by minimizing the negative logarithm of the probability $p(s_{uu'}|y_{uu'}, \sigma_{uu'})$ averaged over images, random transformations, and point pairs. Formally, the learning objective is given by:

$$\mathcal{L}(\phi, \sigma) = \frac{1}{|\mathcal{X}|} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{E}_{g,u,u'} \left[-\log p \left(s_{uu'}^{\mathbf{x}, \mathbf{x} \circ g^{-1}}(\phi) \middle| y_{uu'}^{g}, \frac{\sigma_u^{\mathbf{x}} + \sigma_{u'}^{\mathbf{x} \circ g^{-1}}}{2} \right) \right] \tag{6}$$

Here the score s depends on the neural network ϕ as shown in eq. (3). The function σ is implemented as a small neural network branching off ϕ and is also learned with it. The labels $y_{uu'}^{g}$ are easily obtained as

$$y_{uu'}^{g} = \begin{cases} 1, & \|u' - g(u)\|_2 \leq \tau_1, \\ 0, & \tau_1 < \|u' - g(u)\|_2 \leq \tau_2 \\ -1, & \text{otherwise.} \end{cases} \tag{7}$$

where $\tau_1 < \tau_2$ are matching thresholds (we set $\tau_1 = 1$ and $\tau_2 = 30$ pixels). The value of the probabilistic loss $\mathcal{L}$ as a function of the similarity $s_{uu'}^{\mathbf{x}\mathbf{x}'}$ and the predicted uncertainty $\sigma_{uu'}$ is illustrated in Figure 3.

The set of sampled transformations g consists of random affine warps. To avoid border artifacts, following [31], we mirror-pad each image enlarging its size by a factor of two while biasing the sampled transformations towards zooming into the padded image. In order to avoid potential trivial solutions due to keeping the first image $\mathbf{x}$ unwarped (as the network can catch subtle artifacts induced by warping), we sample two transformations $\hat{g}$, $\hat{g}'$ and then warp the original input image $\hat{x}$ twice to form the input image pair

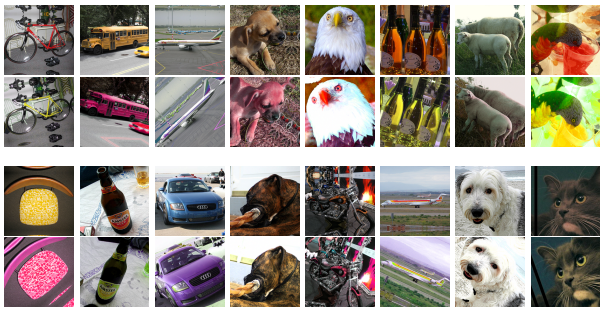


Figure 4. **Example geometric and appearance transformations** used to supervise the learning of our representation. The first (resp. third) row displays original images while the second (resp. fourth) row shows their transformed versions.

$\mathbf{x} = \hat{\mathbf{x}} \circ \hat{g}^{-1}$ and $\mathbf{x}' = \hat{\mathbf{x}} \circ \hat{g}'^{-1}$. The pairwise transformation $g = \hat{g} \circ \hat{g}'^{-1}$ is a straightforward composition of $\hat{g}$ and $\hat{g}'$. In order to sample pairs of pixels (u, u'), we first randomly pick 700 points $\mathcal{U} = \{u_i\}_{i=1}^{700}$ from the first image. For each u_i, we then sample $u_i' = g(u_i)$ from the second image and evaluate the loss $\mathcal{L}$ on all possible pairs $(u_i, u_j') \in \mathcal{U} \times \mathcal{U}'$. We then follow a hard negative mining strategy by selecting the 30 negative samples u' from the second image (out of 700 potential samples) that contribute to $\mathcal{L}$ the most. Backpropagation is then performed only through these "hard negative" examples and all the positive examples while equally balancing the overall weights of the two sets of pixel pairs.

Appearance transformations. While random affine warping makes our features invariant to the geometric transformations, a successful representation should be also invariant to intraclass appearance variations caused by *e.g.* color and illumination changes. Hence, besides warping the input image, we apply a random color transformation $c(\hat{g}(\hat{\mathbf{x}}))$ after the geometric transformation $\hat{g}(\hat{\mathbf{x}})$. The color transformations are generated following the approach of [22]. We increase the intensity of the color shifts in order to introduce substantial appearance changes required to boost the invariance properties of the representation. Examples of the original images and their geometry-appearance transformations are shown in Figure 4.

3.5. Discussion

Besides its robust nature, the formulation so far can be seen as learning discriminative viewpoint invariant features. This does not guarantee *per se* that the learned descriptors are characteristics of particular object parts. For example, since the model is only trained against synthetic warps of individual images, the descriptors computed for analogous parts in different object instances (e.g. the eyes in two different cats) may still differ. Even out-of-plane rotations are in principle sufficient to throw off the model.

Recently, the authors of [35] have suggested to constrain the descriptor capacity to favor generalization. In particular, they argue that using two dimensional descriptors strongly encourages them to attach to specific points on the surface of an object, and thus to generalize across different object instances. Nevertheless, the method of [35] was found to be too fragile to work well in challenging data where significant occlusions may be present. Our approach trades off descriptor generality for robustness. As we will see in the experiments, this pays off as, ultimately, the representation is fine-tuned with a small amount of supervised data which is sufficient to bridge most of the gaps.

3.6. Learning details

We learn our representation using the training images of the 12 rigid PASCAL classes from the ImageNet dataset (but we test it on all 20 classes, including non-rigid ones). As a preprocessing step, we apply a weakly supervised detector [2] and use the resulting image crops instead of the full images. This detector only requires image-level labels and no further supervision is used. This is exactly the same level of supervision used in [27, 31] and weaker than in [34] where bounding box annotations are required.

The representation predictor $\phi(\mathbf{x})$ is a deep convolutional neural network whose architecture is based on the ResNet-50 model [12] due to its good compromise between speed and capacity. We remove the two topmost layers and base the rest of our model on the rectified res5c features. In order to increase the spatial resolution of the produced representation, following [44] we dilate all res5 convolutional filters by a factor of 2 while decreasing their stride to 1. Finally, we attach a 1×1 convolutional layer that produces raw embedding vectors $\hat{\phi}(\mathbf{x}) \in \mathbb{R}^{H \times W \times (C+1)}$. The first C channels of $\hat{\phi}(\mathbf{x})$ are sliced out and ℓ_2 normalized at every spatial location u to form the embedding $\phi(\mathbf{x}) \in \mathbb{R}^{H \times W \times C}$. The last $(C+1)$-th channel $\phi(\mathbf{x})[:, :, C+1]$ of $\hat{\phi}(\mathbf{x})$ is passed through a SoftReLU and lower-bounded by $\epsilon \to 0$ which results in the inverse-confidence predictions $\sigma(\mathbf{x}) = \log(1 + \exp(\hat{\phi}(\mathbf{x})[:, :, C+1])) + \epsilon$.

Our network is optimized using the AdaGrad solver. Learning rate, weight decay and momentum were set to 0.001, 0.0005 and 0.9 respectively. The network is trained until no further loss improvement is observed. Learning converges within 36 hours on a single GPU.

4. Experiments

We first show qualitative results of our self-learning approach (section 4.1). Then, we quantitatively evaluate for the semantic matching (section 4.2) and for the keypoint detection (section 4.3) tasks.

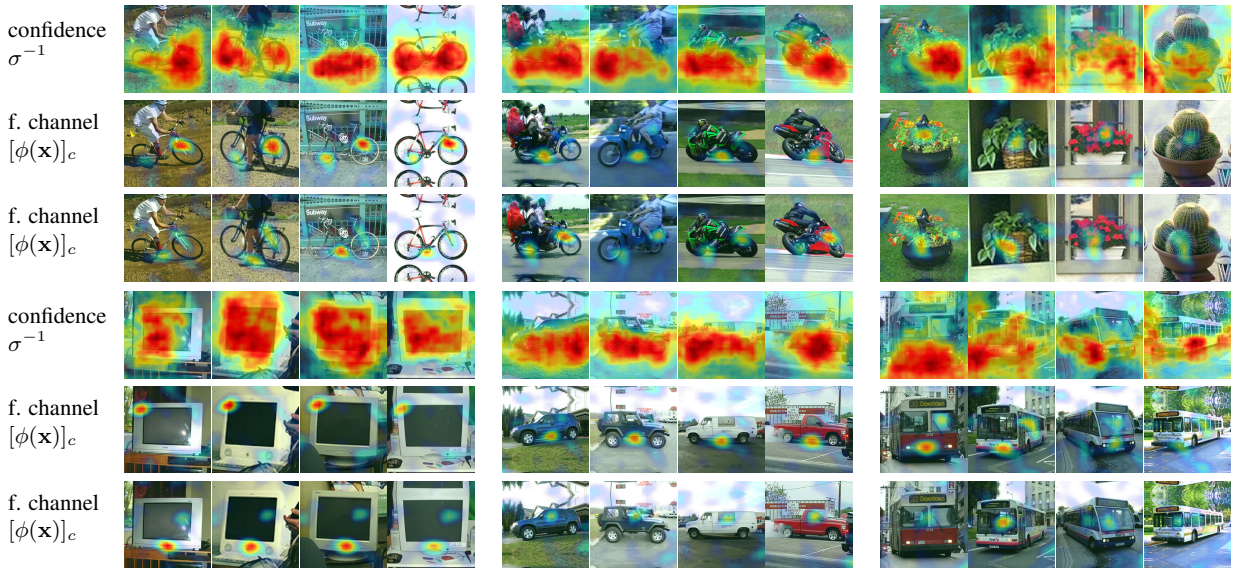


Figure 5. **Qualitative analysis of the learned equivariant feature representation** ϕ visualizing predicted confidence maps σ^{-1} and several responses $\max([\phi(\mathbf{x})]_c, 0)$ of different channels c of the representation, for six different categories.

4.1. Qualitative analysis

We first qualitatively analyze the nature of the learned feature representation. Figure 5 considers six categories and shows, for four images of each category, the confidence maps $\sigma(\mathbf{x})^{-1}$ along with example rectified responses $\max([\phi(\mathbf{x})_u]_c, 0)$ for several feature channels c of the learned representation. It can be observed that the responses resemble distinct keypoint detectors that fire consistently across different instances of a category, even in the presence of large intra-class variations. Furthermore, the confidence predictor $\sigma(\mathbf{x})^{-1}$ can be interpreted as a generic detector of distinct areas of the image foreground.

4.2. Semantic matching

We first assess our method on the problem of semantic matching and compare it to other unsupervised and weakly-supervised approaches for learning geometry-aware representation. In particular, we follow the dataset and experimental protocol of [10] and consider the problem of establishing correspondences between bounding box proposals and keypoints extracted in pairs of images.

Compared approaches.

We compare our learned dense features to five existing feature representations. First, in order to demonstrate the improvement of our self-learning approach over the pre-trained (using only image-level labels) ResNet-50 model, we consider **ResNet-50-HC** which is a hypercolumn architecture that pools features from the res3c, res4c, res5c layers and separately upsamples them to a common spatial size. In order to demonstrate the benefits of the probabilistic introspection, we also present results of **Ours w/o conf.** which is our method trained by optimizing the non-probabilistic loss function from eq. (4). Then, to provide a direct comparison with approaches that tackle the geometric feature learning task, we report the results of [27] and [34]. For **AnchorNet** [27], we use their public class-agnostic model. To provide a fair comparison with the method of **Thewlis *et al.*** [34], we train their method on the same dataset as used for our features. To establish a baseline, we explore three variants of the base architecture proposed in [34]: a model with 10 landmarks (as proposed in the original work), a model with 64 landmarks (to increase model capacity) and finally a modified, class specific architecture which learns a set of 64 landmarks *per-class*. In practice, we found the second design to be most effective, and therefore, all reported results use this option.[3] The last baseline uses pool4 features from the **VGG16** architecture [32] pre-trained on the ImageNet image classification task. We selected these features, since they are the basis of current state-of-the-art semantic matching approaches [31, 11, 10]. Alongside other unsupervised and weakly supervised methods, we also compare against the fully supervised **SCNet-A** architecture introduced in [11].

For our approach, matching descriptors are produced by exploiting the confidence prediction capacity of our model, scaling the outputs of the final layer by the inverse of the predicted uncertainty σ. We then follow the simple approach developed in [11], by applying ROI-pooling with

[3] While this approach has been shown to be effective under more constrained conditions, we were unable to achieve robust learning dynamics when applying it to our task.

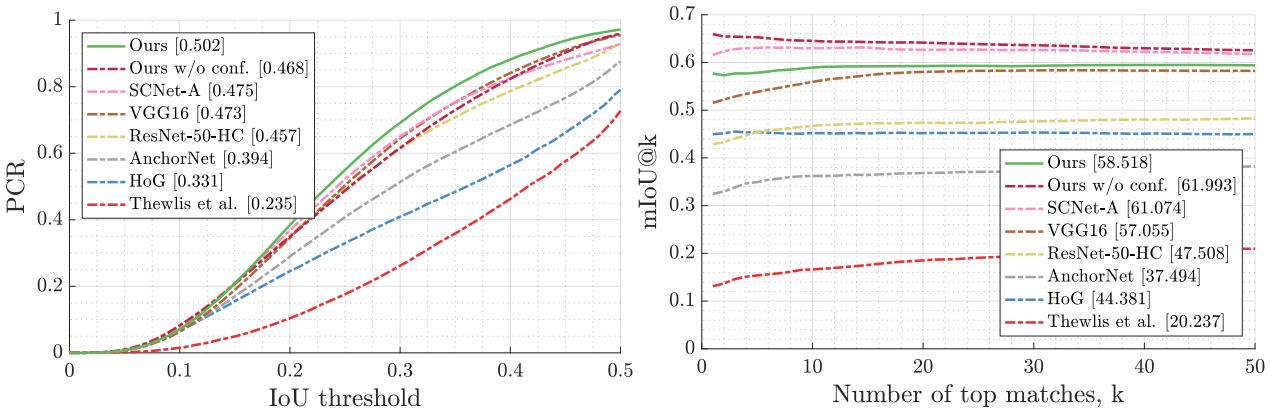


Figure 6. **Region matching performance on PF-Pascal**. Features are matched directly without any spatial regularization. Left: region matching precision (PCR). Right: region matching accuracy (mIoU@k). Note that unlike all other reported approaches, SCNet-A [11] is a fully supervised method.

bin size 7×7 to each proposal region resulting in a feature vector comprising these scaled representations. We further pool and concatenate res4c features from a lower layer of our network. In order to produce a dense warping field for keypoint matching we employ the sd-filtering as done in [10, 11]. For keypoint matching, following other approaches [10, 31, 11], we modify our original ResNet50-based architecture by replacing the network trunk with the VGG16 architecture truncated after the pool4 features and terminated as described in section 3.6. This network was trained on all 20 PASCAL classes of the ImageNet dataset according to the same learning schedule as described in section 3.6. For this architecture, instead of res4c features we pool and concatenate the pool4 features.

Since our objective is to assess *feature quality*, we evaluate each method without using any spatial regularization (such as *e.g.* Local Offset Matching [10], joint warp estimation [31], or MRFs with geometric potentials [39]).[4]

Dataset. We evaluate our approach on the PF-PASCAL dataset [10] which contains pairs of images which have been fully annotated with keypoints for 20 object classes. Each method is evaluated with a set of 1000 object proposals per image, generated with the Randomized Prim (RP) method [25]. Following [11], performance is reported on the *test* partition, which comprises 302 image pairs.

Evaluation. We report results under the standard PCR (probability of correct regions) and mIoU@k (mean intersection over union of the best k matches) metrics introduced in [10]. PCR aims to capture the accuracy of overall assignment, while mIoU@k reflects the reliability of matching scores. Following the common practice on this dataset, keypoint matching is assessed by reporting PCK@α with the misalignment sensitivity threshold α set to 0.1. All evaluations are conducted using the public implementation provided by the authors of [11].

[4]The development of effective spatial regularization methods forms an important, but orthogonal line of research to the focus of our work.

Method	**PCK**	**Method**	**PCK**
Thewlis et al. [34]	14.4	ResNet50-HC [12]	64.0
AnchorNet [27]	56.3	SCNet-A [11]	66.3
VGG16 [10]	62.3	Ours w/o conf.	60.6
gCNN [31]	62.6	**Ours**	66.5

Table 1. **Keypoint matching performance on PF-Pascal** reporting PCK@0.1 for our method and existing approaches.

Results. The region matching results are shown in Figure 6. First, we observe that our approach significantly outperforms previous representations trained with a comparable amount of supervision: AnchorNet [27], the method of Thewlis *et al.* [34], and VGG16 [32]. Second, we see that, interestingly, our self-supervised features perform on par with the model SCNet-A of [11] which is in fact *fully supervised* with keypoint annotations. These observations are encouraging also due to the fact that our representation was trained only on rigid classes while the PF-Pascal dataset also contains a large portion of the non-rigid ones.

Results for keypoint matching are present in Table 1. Similar to region matching, we observe improvements over other approaches trained with comparable level of supervision. Furthermore, our results are again on par with the fully supervised SCNet-A [11]. We observe a decrease in matching performance with Ours w/o conf. which validates the importance of the proposed instrospection mechanism.

4.3. Few-shot keypoint detection

In section 4.1 we have observed that the learned features often correspond to distinctive object parts. Those do not necessarily have a semantic meaning, as demonstrated in

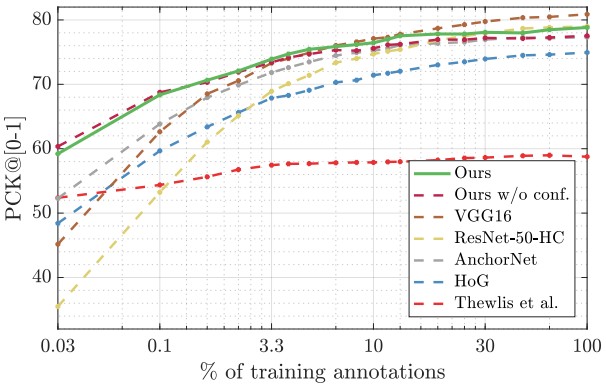


Figure 7. **Keypoint prediction on Pascal3D**. We report the area under the PCK-over-alpha curve as a function of the number of training annotations.

[34], but they can still be used as anchors that facilitate the detection of semantic parts. Following [34], in this section we tackle the task of semantic keypoint detection where our learned representation as well as competitors is used as input features for a keypoint predictor. The keypoint detection performance then serves as an estimate of how well the respective representations encode the geometrical structure of visual categories. We depart from [34] and we consider a significantly more challenging setting with out-of-plane rotations and large appearance variations.

Furthermore, an important feature of successful geometric representations is how well they facilitate transfer of information from a very limited number of annotated samples. Hence, here we consider keypoint detection with few-shot supervision where a training set of object keypoint annotations is gradually extended with new training samples while monitoring the performance on a held-out test set.

Dataset. We use the keypoint annotations from the original Pascal3D dataset [40]. The few-shot keypoint predictors are trained on the "train" set of Pascal3D and evaluated on the held-out "val" set. Following common practice [38], knowledge of a ground truth bounding box as well as the depicted object's class is assumed during both training and testing. The task is evaluated using the probability of correct keypoint measure (PCK) introduced in [43]. A keypoint prediction is regarded as correct if its distance from the corresponding ground truth annotation is lower than $\alpha \times \max\{w, h\}$, where w, h are the object bounding box dimensions and α controls the sensitivity of the measure to misalignments. For each class, PCK corresponds to the ratio between the number of correct predictions and the total number of keypoint annotations. Similar to the PCR metric, we integrate the measure over all possible α values and report the average over the 12 Pascal3D object classes.

Keypoint predictor. Our keypoint predictor consists of a 512-channel 3×3 convolutional layer with stride 1 followed by batch normalization, ReLU and a final 3×3 convolutional layer with stride 1 terminated by the sigmoid activation function. Each channel of the final layer then serves as a response map of the corresponding keypoint class. The loss minimizes the weighted ℓ_2 distance between the ground truth heatmap and the corresponding prediction as proposed in [38]. The evaluation process alternates between training the keypoint detector, evaluating its performance and adding a new set of training annotations consisting of an equal number of randomly sampled images per class. For each round, the detector is trained for 3 epochs making sure that at least 500 training steps are performed for each epoch. Detector parameters are initialized with the model from the previous round. The experiment is run three times with different random seeds and we report an average over PCKs.

Results. Results of the few-shot detection experiments are reported in Figure 7. Our method surpasses all the compared approaches when a small percentage of the training annotations is available, and in particular the methods of [27], [34], and [31], while performing on par with the best competitor on this task (VGG16 [32]) when the full training set is used. Similar to the semantic matching experiments section 4.2, we observe significant drop in performance of the method from [34]. Ours w/o conf. obtains similar results to the proposed method. This is likely due to the fact that the detection dataset does not contain a large quantity of background clutter because the evaluated instances are always cropped using a tight ground truth bounding box.

5. Conclusions

In this paper, we have presented a self-supervised method that can pre-train features useful to reason about the geometry of object categories in tasks such as part localization and semantic matching. The method combines the robustness of recent approaches such as AnchorNet with the geometric prior induced by invariance to synthetic image transformations. This allows to train features that excel at these geometric tasks using only images with class-level annotations. We have shown that these features outperform all other pre-training methods in semantic matching and part localization. In the case of the first task, our features perform on par with a fully-supervised approach.

Acknowledgments. The authors gratefully acknowledge the support of EPSRC AIMS, Seebibyte and ERC 677195-IDIU. The authors would also like to thank James Thewlis for kindly sharing code.

Self-supervised Learning of Geometrically Stable Features Through Probabilistic Introspection

Appendix

In the supplementary material below, we present an ablation study of the components of our method (appendix A). In appendix B, we also provide details of the weakly supervised method that produced the bounding box annotations used to train our model.

A. Ablation studies

In addition to the results reported in sections 4.2. and 4.3. of the paper, we report additional ablation experiments that validate the contribution of the proposed components of our method.

In order to show the improvements over the base architecture that was used to initialize our network, we also compare against the res5c features from the version of the pretrained ResNet-50 model, the filters of which were dilated as explained in section 3.6. in the paper (**ResNet-50-dilated**).

Furthermore, to provide an extended comparison with alternative matching loss formulations, a flavour of our method, abbreviated as **Contrastive**, implements the contrastive loss formulation from [5].

We also test three more methods that assess the sensitivity of the proposed approach to the utilized dataset. We include results for our method trained with ground truth bounding box labels (**Ours-GTbox**), rather than the weakly supervised detections used in the original formulation, to enable an assessment of the method's robustness to the usage of imperfect bounding box annotations. Another variation of our method, **Ours-NObox**, does not use any bounding box annotations. Finally, **Ours-nonrigid** uses all 20 PASCAL categories for training as opposed to the original training setup that used images of the 12 rigid classes.

All variants were evaluated on both the semantic matching and keypoint prediction tasks. The results of the semantic matching experiments are reported in fig. 8 while fig. 9 contains the results of the few-shot keypoint prediction task.

The results indicate that for both semantic matching and keypoint detection the performance of the ground-truth supervised setup is on par with the proposed weakly supervised setup. This shows that, with the inclusion of the probabilistic introspection mechanism, the method has good robustness to annotation noise. The performance of our method trained with the non-rigid categories is on par with the rigid case for proposal matching. We observe a decrease in performance for the keypoint detection task. This is because the few-shot detection dataset consists of only rigid classes and adding the non-rigid ones to the training set makes the features less specialized for the final task. The variant which trains features via the contrastive loss gives lower performance.

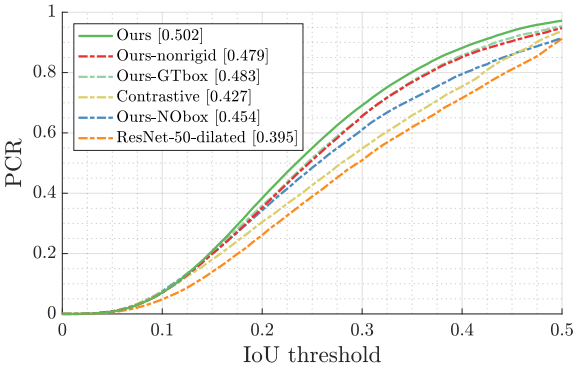


Figure 8. **Ablation study on PF-Pascal**. The region matching performance of several variants of our method (see appendix A for details of each variant).

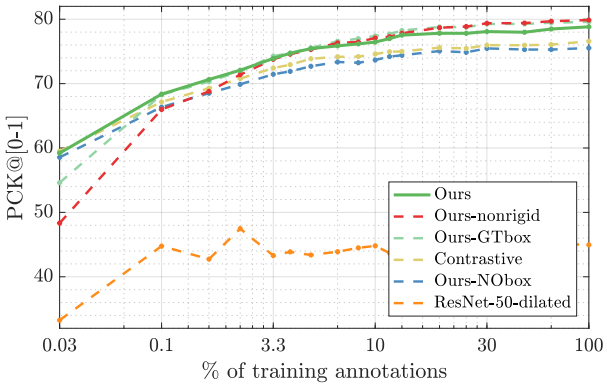


Figure 9. **Ablation study on the few-shot keypoint detection task on Pascal3D**. We report the area under the PCK-over-alpha curve as a function of the number of training annotations for several variants of our method. For details of each variant see appendix A.

A.1. Keypoint detection - detector validation

In section 4.3. in the paper, we reported results for a keypoint detector with a design closely related to that of [38]. In order to validate the implementation of the detector, we provide a comparison against the results of the fully

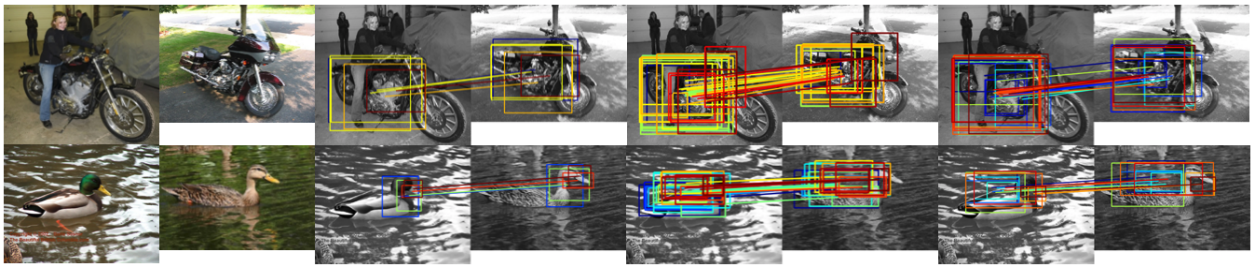


Figure 10. **Region matching examples** for pairs of motorbike (top) and duck (bottom) images. From left to right: source and target images, HOG with NAM matching [10], ours, SCNet-A [11]. We show correctly matched boxes, color-coded according to matching score (red: higher, blue: lower).

supervised detector from [38]. When using all available annotations and the Resnet-50-HC descriptors, the mean PCK ($\alpha = 0.1$) over the 12 rigid classes of the Pascal3D test set is 54.7. This is on par with the best single-model result from [38] (53.3 PCK), validating our keypoint predictor as a representative proxy for evaluating the quality of our feature baselines.

B. Weakly supervised detections

Here we give details of the weakly supervised detector used to provide bounding box annotations for our method, as discussed in Sec. 3.6 of the paper. We use the `vgg-f`-based model described in [2], which is trained using EdgeBox proposals[47] and the image-level labels of the Pascal VOC 2007 detection dataset [8]. To produce bounding box predictions for the ImageNet dataset, we follow the multi-scale evaluation technique described in [2], averaging predictions over five scales and flipped copies of each scale. To form our training set, we then select top scoring box for each class label present in the image. In order to maintain a high quality of box annotation, we do not include boxes whose scores fall below the median detector score of the given class (the median is computed after filtering scores which fall below the noise score threshold of 0.001 given in the public implementation[5] of [2]).

C. Qualitative results

Additional qualitative results for the semantic matching task ok PF-Pascal are present in fig. 10. We show the matching regions for two example pairs, for the method of [10], ours, and the fully-supervised method of SCNet-A.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proc. CVPR*, 2014. 2

[2] H. Bilen and A. Vedaldi. Weakly supervised deep detection networks. Proc. CVPR, 2016. 5, 10

[3] H. Bristow, J. Valmadre, and S. Lucey. Dense semantic correspondence where every pixel is a classifier. In *Proc. ICCV*, 2015. 2

[4] C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker. Universal correspondence network. In *Proc. NIPS*. 2016. 2

[5] C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker. Universal correspondence network. In *Proc. NIPS*, 2016. 9

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005. 2

[7] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, 2015. 1

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 10

[9] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik. Using k-poselets for detecting people and localizing their keypoints. In *Proc. CVPR*, 2014. 2

[10] B. Ham, M. Cho, C. Schmid, and J. Ponce. Proposal flow. In *Proc. CVPR*, 2016. 2, 6, 7, 10

[11] K. Han, R. S. Rezende, B. Ham, K.-Y. K. Wong, M. Cho, and J. P. Cordelia Schmid. Scnet: Learning semantic correspondence. In *Proc. ICCV*, 2017. 2, 6, 7, 10

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 5, 7

[13] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In *Proc. NIPS*, 2012. 2

[14] B. K. P. Horn and B. G. Schunck. Determining optical flow: A retrospective. *Artif. Intell.*, (1-2), 1993. 2

[15] J. Hur, H. Lim, C. Park, and S. C. Ahn. Generalized deformable spatial pyramid: Geometry-preserving dense correspondence estimation. In *Proc. CVPR*, 2015. 2

[16] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proc. BMVC*, 2010. 2

[17] A. Kanazawa, D. W. Jacobs, and M. Chandraker. WarpNet: Weakly supervised matching for single-view reconstruction. In *Proc. CVPR*, 2016. 1, 2

[18] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Proc. NIPS*, 2017. 4

[5] https://github.com/hbilen/WSDDN

[19] J. Kim, C. Liu, F. Sha, and K. Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *Proc. CVPR*, 2013. 2

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, 2012. 2

[21] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994, 2011. 2

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *Proc. ECCV*, 2016. 5

[23] J. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Proc. NIPS*, 2014. 2

[24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 2

[25] S. Manen, M. Guillaumin, and L. Van Gool. Prime object proposals with randomized prim's algorithm. In *Proc. ICCV*, 2013. 7

[26] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, 2016. 1

[27] D. Novotny, D. Larlus, and A. Vedaldi. Anchornet: A weakly supervised network to learn geometry-sensitive features for semantic matching. In *Proc. CVPR*, 2017. 2, 5, 6, 7, 8

[28] D. Novotny, D. Larlus, and A. Vedaldi. Learning 3d object categories by looking around them. In *Proc. ICCV*, 2017. 4

[29] M. Okutomi and T. Kanade. A multiple-baseline stereo. *PAMI*, 15(4):353–363, 1993. 2

[30] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. Efros. Context encoders: Feature learning by inpainting. In *Proc. CVPR*, 2016. 1

[31] I. Rocco, R. Arandjelović, and J. Sivic. Convolutional neural network architecture for geometric matching. In *Proc. CVPR*, 2017. 1, 2, 4, 5, 6, 7, 8

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 6, 7, 8

[33] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, 2003. 2

[34] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *Proc. ICCV*, 2017. 1, 2, 5, 6, 7, 8

[35] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised object learning from dense invariant image labelling. In *Proc. NIPS*, 2017. 1, 2, 5

[36] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Proc. NIPS*. 2014. 2

[37] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proc. CVPR*, 2014. 2

[38] S. Tulsiani and J. Malik. Viewpoints and keypoints. In *Proc. CVPR*, 2015. 2, 8, 9, 10

[39] N. Ufer and B. Ommer. Deep semantic feature matching. In *Proc. CVPR*, 2017. 7

[40] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *Proc. WACV*, 2014. 2, 8

[41] H. Yang, W.-Y. Lin, and J. Lu. Daisy filter flow: A generalized discrete approach to dense correspondences. In *Proc. CVPR*, 2014. 2

[42] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Proc. CVPR*, 2011. 2

[43] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *PAMI*, 35(12):2878–2890, 2013. 8

[44] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *Proc. ICLR*, 2016. 5

[45] T. Zhou, Y. Jae Lee, S. X. Yu, and A. A. Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *Proc. CVPR*, 2015. 2

[46] T. Zhou, P. Krähenbühl, M. Aubry, Q. Huang, and A. A. Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Proc. CVPR*, 2016. 2

[47] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, pages 391–405. Springer, 2014. 10

7

Semi-convolutional Operators for Instance Segmentation

This work was presented as a *poster* presentation at the European Conference on Computer Vision, 2018 [Novotny et al. 2018b].

Semi-convolutional Operators for Instance Segmentation

David Novotny*[1,2] [0000−0002−9517−1464], Samuel Albanie*[1] [0000−0003−1732−9198], Diane Larlus[2], and Andrea Vedaldi[1] [0000−0003−1374−2858]

[1] Visual Geometry Group, Department of Engineering Science, University of Oxford
{david,albanie,vedaldi}@robots.ox.ac.uk

[2] Computer Vision Group, NAVER LABS Europe
diane.larlus@naverlabs.com

Abstract. Object detection and instance segmentation are dominated by region-based methods such as Mask RCNN. However, there is a growing interest in reducing these problems to pixel labeling tasks, as the latter could be more efficient, could be integrated seamlessly in image-to-image network architectures as used in many other tasks, and could be more accurate for objects that are not well approximated by bounding boxes. In this paper we show theoretically and empirically that constructing dense pixel embeddings that can separate object instances cannot be easily achieved using convolutional operators. At the same time, we show that simple modifications, which we call semi-convolutional, have a much better chance of succeeding at this task. We use the latter to show a connection to Hough voting as well as to a variant of the bilateral kernel that is spatially steered by a convolutional network. We demonstrate that these operators can also be used to improve approaches such as Mask RCNN, demonstrating better segmentation of complex biological shapes and PASCAL VOC categories than achievable by Mask RCNN alone.

Keywords: Instance embedding, object detection, instance segmentation, coloring, semi-convolutional

1 Introduction

State-of-the-art methods for detecting objects in images, such as R-CNN [21,20,49], YOLO [47], and SSD [40], can be seen as variants of the same paradigm: a certain number of candidate image regions are proposed, either dynamically or from a fixed pool, and then a convolutional neural network (CNN) is used to decide which of these regions tightly enclose an instance of the object of interest. An important advantage of this strategy, which we call *propose & verify* (P&V), is that it works particularly well with standard CNNs. However, P&V also has several significant shortcomings, starting from the fact that rectangular proposals can only approximate the actual shape of objects; segmenting objects, in particular, requires a two-step approach where, as in Mask R-CNN [25], one first

* Equal contribution

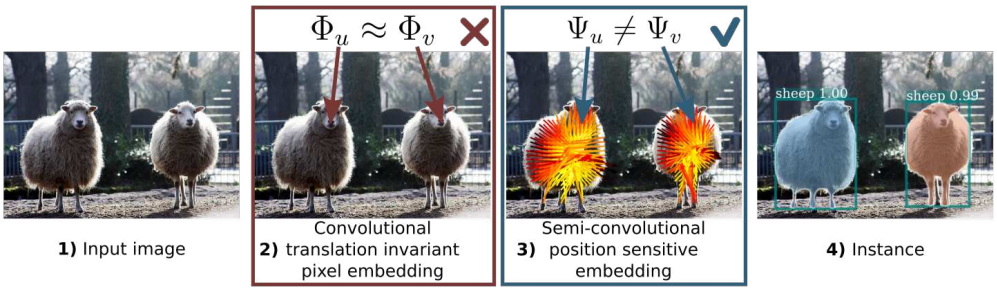


Fig. 1. Approaches for instance segmentation based on dense coloring via convolutional pixel embeddings cannot easily distinguishing identical copies of an object. In this paper, we propose a novel semi-convolutional embedding that is better suited for instance segmentation.

detects object instances using simple shapes such as rectangles, and only then refines the detections to pixel-accurate segmentations.

An alternative to P&V that can overcome such limitations is to label directly individual pixels with an identifier of the corresponding object occurrence. This approach, which we call *instance coloring* (IC), can efficiently represent any number of objects of arbitrary shape by predicting a single label map. Thus IC is in principle much more efficient than P&V. Another appeal of IC is that it can be formulated as an image-to-image regression problem, similar to other image understanding tasks such as denoising, depth and normal estimation, and semantic segmentation. Thus this strategy may allow to more easily build *unified architectures* such as [29,27] that can solve instance segmentations together with other problems.

Despite the theoretical benefits of IC, however, P&V methods currently dominate in terms of overall accuracy. The goal of this paper is to explore some of the reasons for this gap and to suggest workarounds. Part of the problem may be in the nature of the dense labels. The most obvious way of coloring objects is to number them and "paint" them with their corresponding number. However, the latter is a global operation as it requires to be aware of all the objects in the image. CNNs, which are *local and translation invariant*, may therefore be ill-suited for direct enumeration. Several authors have thus explored alternative coloring schemes more suitable for convolutional networks. A popular approach is to assign an arbitrary color (often in the guise of a real vector) to each object occurrence, with the only requirement that different colors should be used for different objects [17,6,30]. The resulting color *affinities* can then be used to easily enumerate object a posteriori via a non-convolutional algorithm.

In this paper, we argue that even the latter technique is insufficient to make IC amenable to computation by CNNs. The reason is that, since CNNs are translation invariant, they must still assign the same color to identical copies of an object, making replicas indistinguishable by convolutional coloring. This argument, which is developed rigorously in sec. 3.6, holds in the limit since in practice the receptive field size of most CNNs is nearly as large as the whole

image; however, it suggests that the convolutional structure of the network is at least an unnatural fit for IC.

In order to overcome this issue, we suggest that an architecture used for IC should not be translation invariant; while this may appear to be a significant departure from convolutional networks, we also show that a small modification of standard CNNs can overcome the problem. We do so by defining *semi-convolutional* operators which mix information extracted from a standard convolutional network with information about the global location of a pixel (sec. 3.1 and fig. 1). We train the latter (sec. 3.2) so that the response of the operator is about the same for all pixels that belong to the same object instance, making this embedding naturally suited for IC. We show that, if the mixing function is additive, then the resulting operator bears some resemblance to Hough voting and related detection approaches. After extending the embedding to incorporate standard convolutional responses that capture appearance cues (sec. 3.3), we use it to induce pixel affinities and show how the latter can be interpreted as a steered version of a bilateral kernel (sec. 3.4). Finally, we show how such affinities can also be integrated in methods such as Mask RCNN (sec. 3.5).

We assess our method with several experiments. We start by investigating the limit properties of our approach on simple synthetic data. Then, we show that our semi-convolutional feature extractor can be successfully combined with state-of-the-art approaches to tackle parsing of biological images containing overlapping and articulated organisms (sec. 4.2). Finally, we apply the latter to a standard instance segmentation benchmark PASCAL VOC (sec. 4.3). We show in all such cases that the use of semi-convolutional features can improve the performance of state-of-the-art instance segmentation methods such as Mask RCNN.

2 Related work

The past years have seen large improvements in object detection, thanks to powerful baselines such as Faster-RCNN [49], SSD [40] or other similar approaches [12,47,36], all from the *propose & verify* strategy.

Following the success of object detection and semantic segmentation, the challenging task of instance-level segmentation has received increasing attention. Several very different families of approaches have been proposed.

Proposal-based instance segmentation. While earlier methods relied on bottom-up segmentations [20,10], the vast majority of recent instance-level approaches combine segment proposals together with powerful object classifiers. In general, they implement a multi-stage pipeline that first generates region proposals or class agnostic boxes, and then classifies them [31,22,8,44,11,45,34]. For instance DeepMask [44] and follow-up approaches [45,9] learn to propose segment candidates that are then classified. The MNC approach [11], based on Faster-RCNN [49], repeats this process twice [11] while [34] does it multiple times. [24] extends [11] to model the shape of objects. The fully convolutional instance segmentation method of [33] also combines segmentation proposal and object detection using a position sensitive score map.

Some methods start with semantic segmentation first, and then cut the regions obtained for each category into multiple instances [28,4,39], possibly involving higher-order CRFs [3].

Among the most successful methods to date, Mask-RCNN [25] extends Faster R-CNN [49] with a small fully convolutional network branch [42] producing segmentation masks for each region of interest predicted by the detection branch. Despite its outstanding results, Mask-RCNN does not come without shortcomings: it relies on a small and predefined set of region proposals and non-maximum suppression, making it less robust to strong occlusions, crowded scenes, or objects with fundamentally non-rectangular shapes (see detailed discussion in sec. 3.6).

Instance-sensitive embeddings. Some works have explored the use of pixel-level embeddings in the context of clustering tasks, employing them as a soft, differentiable proxy for cluster assignments [57,23,17,14,43,30]. This is reminiscent of unsupervised image segmentation approaches [51,18]. It has been used for body joints [43], semantic segmentation [1,23,6] and optical flow [1], and, more relevant to our work, to instance segmentation [17,14,6,30].

The goal of this type of approaches is to bring points that belong to the same instance close to each other in an embedding space, so that the decision for two pixels to belong to the same instance can be directly measured by a simple distance function. Such an embedding requires a high degree of invariance to the interior appearance of objects.

Among the most recent methods, [17] combines the embedding with a greedy mechanism to select seed pixels, that are used as starting points to construct instance segments. [6] connects embeddings, low rank matrices and densely connected random fields. [30] embeds the pixels and then groups them into instances with a variant of mean-shift that is implemented as a recurrent neural network. All these approaches are based on convolutions, that are local and translation invariant by construction, and consequently are inherently ill-suited to distinguish several identical instances of the same object (see more details about the convolutional coloring dilemma in sec. 3.6). A recent work [27] employs position sensitive convolutional embeddings that regress the location of the centroid of each pixel's instance. We mainly differ by allowing embeddings to regress an unconstrained representative point of each instance.

Among other approaches using a clustering component, [52] leverages a coverage loss and [59,53,54] make use of depth information. In particular, [54] trains a network to predict each pixel direction towards its instance center along with monocular depth and semantic labeling. Then template matching and proposal fusion techniques are applied.

Other instance segmentation approaches. Several methods [44,45,35,26] move away from box proposals and use Faster-RCNN [49] to produce "centerness" scores on each pixel instead. They directly predict the mask of each object in a second stage. An issue with such approaches is that objects do not necessarily fit in the receptive fields.

Recurrent approaches sequentially generate a list of individual segments. For instance, [2] uses an LSTM for detection with a permutation invariant loss while

[50] uses an LSTM to produce binary segmentation masks for each instance. [48] extends [50] by refining segmentations in each window using a box network. These approaches are slow and do not scale to large and crowded images.

Some approaches use watershed algorithms. [4] predicts pixel-level energy values and then partition the image with a watershed algorithm. [28] combines a watershed algorithm with an instance aware boundary map. Such methods create disconnected regions, especially in the presence of occlusion.

3 Method

3.1 Semi-convolutional networks for instance coloring

Let $\mathbf{x} \in \mathcal{X} = \mathbb{R}^{H \times W \times 3}$ be an image and $u \in \Omega = \{1, \dots, H\} \times \{1, \dots, W\}$ a pixel. In instance segmentation, the goal is to map the image to a collection $\mathcal{S}_{\mathbf{x}} = \{S_1, \dots, S_{K_{\mathbf{x}}}\} \subset 2^{\Omega}$ of image regions, each representing an occurrence of an object of interest. The symbol $S_0 = \Omega - \cup_k S_k$ will denote the complementary region, representing background. The regions as well as their number are a function of the image and the goal is to predict both.

In this paper, we are interested in methods that reduce instance segmentation to a pixel-labeling problem. Namely, we seek to learn a function $\Phi : \mathcal{X} \to \mathcal{L}^{\Omega}$ that associates to each pixel u a certain label $\Phi_u(\mathbf{x}) \in \mathcal{L}$ so that, as a whole, labels encode the segmentation $\mathcal{S}_{\mathbf{x}}$. Intuitively, this can be done by painting different regions with different "colors" (aka pixel labels) making objects easy to recover in post-processing. We call this process *instance coloring* (IC).

A popular IC approach is to use real vectors $\mathcal{L} = \mathbb{R}^d$ as colors, and then require that the colors of different regions are sufficiently well separated. Formally, there should be a margin $M > 0$ such that:

$$\forall u, v \in \Omega : \quad \begin{cases} \|\Phi_u(\mathbf{x}) - \Phi_v(\mathbf{x})\| \leq 1 - M, & \exists k : u, v \in S_k, \\ \|\Phi_u(\mathbf{x}) - \Phi_v(\mathbf{x})\| \geq 1 + M, & \text{otherwise.} \end{cases} \tag{1}$$

If this is the case, clustering colors trivially reconstructs the regions.

Unfortunately, it is difficult for a convolutional operator Φ to satisfy constraint (1) or analogous ones. While this is demonstrated formally in sec. 3.6, for now an intuition suffices: if the image contains replicas of the same object, then a convolutional network, which is translation invariant, must assign the same color to each copy.

If convolutional operators are inappropriate, then, we must abandon them in favor of non-convolutional ones. While this sounds complex, we suggest that very simple modifications of convolutional operators, which we call *semi-convolutional*, may suffice. In particular, if $\Phi_u(\mathbf{x})$ is the output of a convolutional operator at pixel u, then we can construct a non-convolutional response by mixing it with information about the pixel location. Mathematically, we can define a semi-convolutional operator as:

$$\Psi_u(\mathbf{x}) = f(\Phi_u(\mathbf{x}), u) \tag{2}$$

where $f : \mathcal{L} \times \Omega \to \mathcal{L}'$ is a suitable mixing function. As our main example of such an operator, we consider a particularly simple type of mixing function, namely addition. With it, eq. (2) specializes to:

$$\Psi_u(\mathbf{x}) = \Phi_u(\mathbf{x}) + u, \qquad \Phi_u(\mathbf{x}) \in \mathcal{L} = \mathbb{R}^2. \tag{3}$$

While this choice is restrictive, it has the benefit of having a very simple interpretation. Suppose in fact that the resulting embedding can perfectly separate instances, in the sense that $\Psi_u(\mathbf{x}) = \Psi_v(\mathbf{x}) \Leftrightarrow \exists k : (u, v) \in S_k$. Then for all the pixels of the region S_k we can write in particular:

$$\forall u \in S_k : \quad \Phi_u(\mathbf{x}) + u = c_k \tag{4}$$

where $c_k \in \mathbb{R}^2$ is an *instance-specific point*. In other words, we see that the effect of learning this semi-convolutional embedding for instance segmentation is to predict a *displacement field* $\Phi(\mathbf{x})$ that maps all pixels of an object instance to an instance-specific centroid c_k. An illustration of the displacement field can be found fig. 2.

Relation to Hough voting and implicit shape models. Eq. (3) and (4) are reminiscent of well known detection methods in computer vision: Hough voting [15,5] and implicit shape model (ISM) [32]. Recall that both of these methods map image patches to votes for the parameters θ of possible object occurrences. In simple cases, $\theta \in \mathbb{R}^2$ can be the centroid of an object, and casting votes may have a form similar to eq. (4).

This establishes, a clear link between voting-based methods for object detection and coloring methods for instance segmentation. At the same time, there are significant differences. First, the goal here is to group pixels, not to reconstruct the parameters of an object instance (such as its centroid and scale). Eq. (3) may have this interpretation, but the more general version eq. (2) does not. Second, in methods such as Hough or ISM the centroid is defined a-priori as the actual center of the object; here the centroid c_k has no explicit meaning, but is automatically inferred as a useful but arbitrary reference point. Third, in traditional voting schemes voting integrates local information extracted from individual patches; here the receptive field size of $\Phi_u(\mathbf{x})$ may be enough to comprise the whole object, or more. The goal of eq. (2) and (3) is not to pool local information, but to solve a representational issue.

3.2 Learning additive semi-convolutional features

Learning the semi-convolutional features of eq. (2) can be formulated in many different ways. Here we adopt a simple direct formulation inspired by [14] and build a loss by considering, for each image $\mathbf{x}$ and instance $S \in \mathcal{S}$ in its segmentation, the distance between the embedding of each pixel $u \in S$ and the segment-wise mean of these embeddings:

$$\mathcal{L}(\Psi|\mathbf{x}, \mathcal{S}) = \sum_{S \in \mathcal{S}} \frac{1}{|S|} \sum_{u \in S} \left\| \Psi_u(\mathbf{x}) - \frac{1}{|S|} \sum_{u \in S} \Psi_u(\mathbf{x}) \right\|. \tag{5}$$

Fig. 2. Semi-convolutional embedding. The first two dimensions of the embedding $\Phi_u(\mathbf{x})$ are visualized as arrows starting from the corresponding pixel location u. Arrows from the same instance tend to point towards a *instance-specific* location c_k.

Note that while this quantity resembles the variance of the embedding values for each segment, it is not as the distance is not squared; this was found to be more robust.

Note also that this loss is simpler than the margin condition (1) and than the losses proposed in [14], which resemble (1) more closely. In particular, this loss only includes an "attractive" force which encourages embeddings for each segment to be all equal to a certain mean value, but does not explicitly encourage different segments to be assigned different embedding values. While this can be done too, empirically we found that minimizing eq. (5) is sufficient to learn good additive semi-convolutional embeddings.

3.3 Coloring instances using individuals' traits

In practice, very rarely an image contains exact replicas of a certain object. Instead, it is more typical for different occurrences to have some distinctive individual traits. For example, different people are generally dressed in different ways, including wearing different colors. In instance segmentation, one can use such cues to tell right away an instance from another. Furthermore, these cues can be extracted by conventional convolutional operators.

In order to incorporate such cues in our additive semi-convolutional formulation, we still consider the expression $\Psi_u(x) = \hat{u} + \Phi_u(\mathbf{x})$. However, we relax $\Phi_u(\mathbf{x}) \in \mathbb{R}^d$ to have more than two dimensions $d > 2$. Furthermore, we define $\hat{u}$ as the pixel coordinates of u, u_x and u_y, extended by zero padding:

$$\hat{u} = \begin{bmatrix} u_x & u_y & 0 & \dots & 0 \end{bmatrix}^\top \in \mathbb{R}^d. \tag{6}$$

In this manner, the last $d-2$ dimensions of the embedding work as conventional convolutional features and can extract instance-specific traits normally.

3.4 Steered bilateral kernels

The pixel embedding vectors $\Psi_u(\mathbf{x})$ must ultimately be decoded as a set of image regions. Again, there are several possible strategies, starting from simple K-means clustering, that can be used to do so. In this section, we consider transforming embeddings in an affinity matrix between two pixels, as the latter can be used in numerous algorithms.

In order to define the affinity between pixels $u, v \in \Omega$, consider first the Gaussian kernel

$$K(u,v) = \exp\left(-\frac{\|\Psi_u(\mathbf{x}) - \Psi_v(\mathbf{x})\|^2}{2}\right). \tag{7}$$

If the augmented embedding eq. (6) is used in the definition of $\Psi_u(\mathbf{x}) = \hat{u} + \Phi_u(\mathbf{x})$, we can split $\Phi_u(\mathbf{x})$ into a geometric part $\Phi^g_u(\mathbf{x}) \in \mathbb{R}^2$ and an appearance part $\Phi^a_u(\mathbf{x}) \in \mathbb{R}^{d-2}$ and expand this kernel as follows:

$$K(u,v) = \exp\left(-\frac{\|(u + \Phi^g_u(\mathbf{x})) - (v + \Phi^g_v(\mathbf{x}))\|^2}{2}\right) \exp\left(-\frac{\|\Phi^a_u(\mathbf{x}) - \Phi^a_v(\mathbf{x})\|^2}{2}\right). \tag{8}$$

It is interesting to compare this definition to the one of the *bilateral kernel*:[3]

$$K_{\text{bil}}(u,v) = \exp\left(-\frac{\|u - v\|^2}{2}\right) \exp\left(-\frac{\|\Phi^a_u(\mathbf{x}) - \Phi^a_v(\mathbf{x})\|^2}{2}\right). \tag{9}$$

The bilateral kernel is very popular in many applications, including image filtering and mean shift clustering. The idea of the bilateral kernel is to consider pixels to be similar if they are close in both space and appearance. Here we have shown that kernel (8) and hence kernel (7) can be interpreted as a generalization of this kernel where spatial locations are steered (distorted) by the network to move pixels that belong to the same underlying object instance closer together.

In a practical implementation of these kernels, vectors should be rescaled before being compared, for example in order to balance spatial and appearance components. In our case, since embeddings are trained end-to-end, the network can learn to perform this balancing automatically, but for the fact that (4) implicitly defines the scaling of the spatial component of the kernel. Hence, we modify eq. (7) in two ways: by introducing a learnable scalar parameter σ and by considering a Laplacian rather than a Gaussian kernel:

$$K_\sigma(u,v) = \exp\left(-\frac{\|\Psi_u(\mathbf{x}) - \Psi_v(\mathbf{x})\|}{\sigma}\right). \tag{10}$$

This kernel is more robust to outliers (as it uses the Euclidean distance rather than its square) and is still positive definite [19]. In the next section we show an example of how this kernel can be used to perform instance coloring.

[3] In the bilateral kernel, a common choice is to set $\Phi^a_u(\mathbf{x}) = \mathbf{x}_u \in \mathbb{R}^3$ as the RGB triplet for the appearance features.

3.5 Semi-convolutional Mask-RCNN

The semi-convolutional framework we proposed in sec. 3.1 is very generic and can be combined with many existing approaches. Here, we describe how it can be combined with the Mask-RCNN (MRCNN) framework [25], the current state-of-the-art in instance segmentation.

MRCNN is based on the RCNN *propose & verify* strategy and first produces a set of rectangular regions $\mathcal{R}$, where each rectangle $R \in \mathcal{R}$ tightly encloses an instance candidate. Then a fully convolutional network (FCN) produces foreground/background segmentation inside each region candidate. In practice, it labels every pixel u_i in R with a foreground score logit $s(u_i) \in \mathbb{R}$. However, this is not an optimal strategy for articulated objects or occluded scenes (as validated in sec. 4.2), as it is difficult for a standard FCN to perform individual foreground/background predictions. Hence we leverage our pixel-level translation sensitive embeddings in order to improve the quality of the predictions $s(u_i)$.

Extending MRCNN. Our approach is based on two intuitions: first, some points are easier to be recognized as foreground than others, and, second, once one such *seed point* has been determined, its affinity with other pixels can be used to cut out the foreground region.

In practice, we first identify a seed pixel u_s in each region R using the MRCNN foreground confidence score map $\mathbf{s} = [s(u_1), \ldots, s(u_{|R|})]$. We select the *most confident* seed point as $u_s = \operatorname{argmax}_{1 \leq i \leq |R|} s(u_i)$, evaluate the steered bilateral kernel $K_\sigma(u_s, u)$ after extracting the embeddings Ψ_{u_s} for the seed and Ψ_{u_i} of each pixel u_i in the region, and then defining updated scores $\hat{s}(u_i)$ as $\hat{s}(u_i) = s(u_i) + \log K_\sigma(u_s, u_i)$. The combination of the scores and the kernel is performed in the log-space due to improved numerical stability. The final per-pixel foreground probabilities are obtained as in [25] with $\operatorname{sigmoid}(\hat{s}(u_i))$.

The entire architecture —the region selection mechanism, the foreground prediction, and the pixel-level embedding —is trained end-to-end. For differentiability, this requires the following modifications: we replace the maximum operator with a soft maximum over the scores $\mathbf{p}_s = \operatorname{softmax}(\mathbf{s})$ and we obtain the seed embedding Ψ_{u_s} as the expectation over the embeddings Ψ_u under the probability density $\mathbf{p}_s$. The network optimizer minimizes, together with the MRCNN losses, the image-level embedding loss $\mathcal{L}(\Psi|\mathbf{x}, \mathcal{S})$ and further attaches a secondary binary cross entropy loss that, similar to the MRCNN mask predictor, minimizes binary cross entropy between the kernel output $K_\sigma(u_s, u_i)$ and the ground truth instance masks.

The predictors of our semi-convolutional features Ψ_u were implemented as an output of a shallow subnetwork, shared between all the FPN layers. This subnet consists of a 256-channel 1×1 convolutional filter followed by ReLU and a final 3×3 convolutional filter producing $D = 8$ dimensional embedding Ψ_u. Due to an excessive sensitivity of the RPN component to perturbations of the underlying FPN representation, we downscale the gradients that are generated by the shallow subnetwork and received by the shared FPN tensors by a factor of 10.

3.6 The convolutional coloring dilemma

In this section, we prove some properties of convolutional operators in relation to solving instance segmentation problems. In order to do this, we need to start by formalizing the problem.

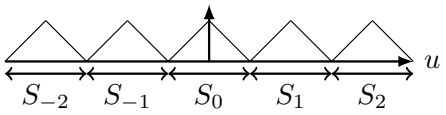
We consider signals (images) of the type $\mathbf{x} : \Omega \to \mathbb{R}$, where the domain Ω is either $\mathbb{Z}^m$ or $\mathbb{R}^m$.[4] In segmentation, we are given a family $\mathbf{x} \in \mathcal{X}$ of such signals, each of which is associated to a certain partition $\mathcal{S}_\mathbf{x} = \{S_1, \dots, S_{K_\mathbf{x}}\}$ of the domain Ω. The goal is to construct a *segmentation algorithm* $\mathcal{A} : \mathbf{x} \mapsto \mathcal{S}_\mathbf{x}$ that computes this function. We look in particular at algorithms that pre-process the signal by assigning a label $\Phi_u(\mathbf{x}) \in \mathcal{L}$ to each point $u \in \Omega$ of the domain. Furthermore, we assume that this labeling operator Φ is *local and translation invariant*[5] so as to be implementable with a convolutional neural network.

There are two families of algorithms that can be used to segment signals in this manner, discussed next.

Propose & verify. The first family of algorithms submits all possible regions $S_r \subset \Omega$, indexed for convenience by a variable r, to a labeling function $\Phi_r(\mathbf{x}) \in \{0, 1\}$ that *verifies* which ones belong to the segmentation $\mathcal{S}_\mathbf{x}$ (i.e. $\Phi_r(\mathbf{x}) = 1 \Leftrightarrow S_r \in \mathcal{S}_\mathbf{x}$). Since in practice it is not possible to test all possible subsets of Ω, such an algorithm must focus on a smaller set of proposal regions. A typical choice is to consider all translated squares (or rectangles) $S_u = [-H, H]^m + u$. Since the index variable $u \in \Omega$ is now a translation, the operator $\Phi_u(\mathbf{x})$ has the form discussed above, although it is not necessarily local or translation invariant.

Instance coloring. The second family of approaches directly colors (labels) pixels with the index of the corresponding region, i.e. $\Phi_u(\mathbf{x}) = k \Leftrightarrow u \in S_k$. Differently from P&V, this can efficiently represent arbitrary shapes. However, the map Φ needs implicitly to decide which number to assign to each region, which is a global operation. Several authors have sought to make this more amenable to convolutional networks. A popular approach [17,14] is to color pixels arbitrarily (for example using vector embeddings) so that similar colors are assigned to pixels in the same region and different colors are used between regions, as already detailed in eq. (1).

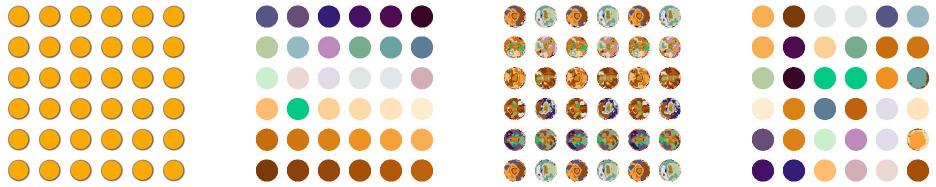
Convolutional coloring dilemma. Here we show that, even with the variants discussed above, IC cannot be approached with convolutional operators even for cases where these would work with P&V.



We do so by considering a simple 1D example. Let $\mathbf{x}$ be a signal of period 2 (i.e. $x_{u+2} = x_u$) where for $u \in [-1, 1]$ the signal is given by $x_u = \min(1-u, 1+u)$.

[4] We assume that the domain extends to infinity to avoid having to deal explicitly with boundary conditions.

[5] We say that Φ is translation invariant if $\Phi_u(\mathbf{x}(\cdot - \tau)) = \Phi_{u-\tau}(\mathbf{x})$ for all translations $\tau \in \Omega$. We say that it is also local if there exists a constant $M > 0$ such that $x_u = x_{u'}$ for all $|u - u'| < M$ implies that $\Phi_u(\mathbf{x}) = \Phi'_u(\mathbf{x})$.



(a) Training image (b) GT instance labels (c) Convolutional embedding (d) Semi-conv. embedding (ours)

Fig. 3. Experiment on synthetic data. An instance segmentation pixel embedding is trained for a synthetic training image consisting of a regular dot pattern (a). After training a model on that image, the produced embeddings are clustered using k-means, encoding the corresponding cluster assignments with consistent pixel colors. A standard convolutional embedding (c) cannot successfully embed each dot into a unique location due to its translational invariance. Our proposed semi-convolutional operator (d) naturally embeds dots with identical appearance but distinct location into distinct regions in the feature space and hence allows for successful clustering of the instances.

Suppose that the segmentation associated to $\mathbf{x}$ is $\mathcal{S} = \{[-1, 1] + 2k, k \in \mathbb{Z}\}$. If we assume that a necessary condition for a coloring-based algorithm is that at least some of the regions are assigned different colors, we see that this cannot be achieved by a convolutional operator. In fact, due to the periodicity of $\mathbf{x}$, any translation invariant function will assign exactly the same color to pixels $2k, k \in \mathbb{Z}$. Thus *all* regions have at least one point with the same color.

On the other hand, this problem can be solved by P&V using the proposal set $\{[-1, 1] + u, u \in \Omega\}$ and the local and translation invariant verification function $\Phi_u(\mathbf{x}) = [x_u = 1]$, which detects the center of each region.

The latter is an extreme example of a convolutional coloring dilemma: namely, a local and translation invariant operator will naturally assign the same color to identical copies of an object even if when they are distinct occurrences (c.f. interesting concurrent work that explores related convolutional dilemmas [38]).

Solving the dilemma. Solving the coloring dilemma can be achieved by using operators that are *not* translation invariant. In the counterexample above, this can be done by using the semi-convolutional function $\Phi_u(x) = u + (1 - x_u)\dot{x}_u$. It is easy to show that $\Phi_u(x) = 2k$ colors each pixel $u \in S_k = [-1, 1] + 2k$ with twice the index of the corresponding region by moving each point u to the center of the closest region. This works because such displacements can be computed by looking only locally, based on the shape of the signal.

4 Experiments

We first conduct experiments on synthetic data in order to clearly demonstrate inherent limitations of convolutional operators for the task of instance segmen-

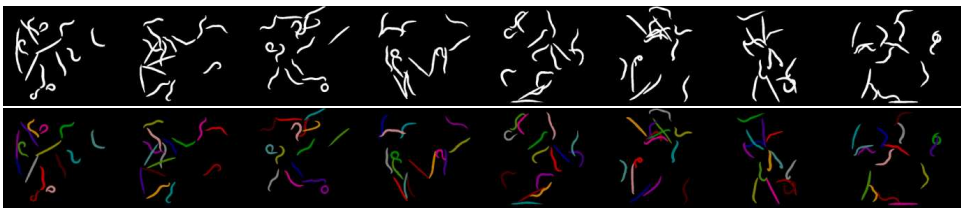


Fig. 4. Sample image crops (top) and corresponding ground-truth (bottom) from the *C. Elegans* dataset.

Table 1. Average precision (AP) for instance segmentation on *C. Elegans* reporting the standard COCO evaluation metrics [37]

AP	AP	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M
Ours	**0.569**	**0.885**	**0.661**	**0.511**	**0.671**
MRCNN [26]	0.559	0.865	0.641	0.502	0.650

tation. In the ensuing parts we demonstrate benefits of the semi-convolutional operators on a challenging scenario with a high number of overlapping articulated instances and finally we compare to the competition on a standard instance segmentation benchmark.

4.1 Synthetic experiments

In sec. 3.1 and 3.6 we suggested that convolution operators are unsuitable for instance segmentation via coloring, but that semi-convolutional ones can do. These experiments illustrate this point by learning a deep neural network to segment a synthetic image x_S where object instances correspond to identical dots arranged in a regular grid (fig. 3 (a)).

We use a network consisting of a pretrained ResNet50 model truncated after the Res2c layer, followed by a set of 1×1 filters that, for each pixel u, produce 8-dimensional pixel embeddings $\Phi_u(x_S)$ or $\Psi_u(x_S)$. We optimize the network by minimizing the loss from *eq.* (5) with stochastic gradient descent. Then, the embeddings corresponding to the foreground regions are extracted and clustered with the k-means algorithm into K clusters, where K is the true number of dots present in the synthetic image.

Fig. 3 visualizes the results. Clustering the features consisting of the position invariant convolutional embedding $\Phi_u(x_S)$ results in nearly random clusters (fig. 3 (c)). On the contrary, the semi-convolutional embedding $\Psi_u(x_S) = \Phi_u(x_S) + u$ allows to separate the different instances almost perfectly when compared to the ground truth segmentation masks (fig. 3 (d)).

4.2 Parsing biological images

The second set of experiments considers the parsing of biological images. Organisms to be segmented present non-rigid pose variations, and frequently form

Fig. 5. Instance segmentation on Pascal VOC 2012. Each pair of rows visualizes instance segmentations produced with method, together with the corresponding semi-convolutional embeddings

clusters of overlapping instances, making the parsing of such images challenging. Yet, this scenario is of crucial importance for many biological studies.

Dataset and evaluation. We evaluate our approach on the C. Elegans dataset (illustrated fig. 4), a subset of the Broad Biomedical Benchmark collection [41]. The dataset consists of 100 bright-field microscopy images. Following standard practice [56,58], we operate on the binary segmentation of the microscopy images. However, since there is no publicly defined evaluation protocol for this dataset, a fair numerical comparison with previously published experiments is infeasible. We therefore compare our method against a very strong baseline (MRCNN) and adopt the methodology introduced by [58] in which the dataset is divided into 50 training and 50 test images. We evaluate the segmentation using average precision (AP) computed using the standard COCO evaluation criteria [37]. We compare our method against the MRCNN FPN-101 model from [25] which attains results on par with state of the art on the challenging COCO instance segmentation task.

Results. The results are given in table 1. We observe that the semi-convolutional embedding Ψ_u brings improvements in all considered instance segmentation metrics. The improvement is more significant at higher IoU thresholds which underlines the importance of utilizing position sensitive embedding in order to precisely delineate an instance within an MRCNN crop.

Table 2. Instance-level segmentation comparison using mean APr metric at 0.5 IoU on the PASCAL VOC 2012 validation set

SDS [22]	PFN [35]	DIN [3]	MNC [11]	FCIS [33]	R2-IOS [34]	DML [17]	R. Emb. [30]	BAIS [24]	MRCNN [26]	**Ours**
43.8	58.7	61.7	63.5	65.7	66.7	62.1	64.5	65.7	69.0	**69.9**

Table 3. Average precision (AP) for instance segmentation on PASCAL VOC 2012 reporting the standard COCO evaluation metrics [37]

AP	*AP*	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L
Ours	**0.412**	**0.699**	**0.424**	0.107	**0.317**	**0.538**
MRCNN [26]	0.401	0.690	0.412	**0.111**	0.313	0.525

4.3 Instance segmentation

The final experiment compares our method to competition on the instance segmentation task on a standard large scale dataset, PASCAL VOC 2012 [16].

As in the previous section, we base our method on the MRCNN FPN-101 model. Because we observed that the RPN component is extremely sensitive to changes in the base architecture, we employed a multistage training strategy. First, MRCNN FPN-101 model is trained until convergence and then our embeddings are attached and fine-tuned with the rest of the network . We follow [25] and learn using 24 SGD epochs, lowering the initial learning rate of 0.0025 tenfold after the first 12 epochs. Following other approaches, we train on the training set of VOC 2012 and test on the validation set.

Results. The results are given in table 2. Our method attains state of the art on PASCAL VOC 2012 which validates our approach. We further compare in detail against MRCNN in table 3 using the standard COCO instance segmentation metrics from [37]. Our method outperforms MRCNN on the considered metrics, confirming the contribution of the proposed semi-convolutional embedding.

5 Conclusions

In this paper, we have considered dense pixel embeddings for the task of instance-level segmentation. Departing from standard approaches that rely on translation invariant convolutional neural networks, we have proposed semi-convolutional operators which can be easily obtained with simple modifications of the convolutional ones. On top of their theoretical advantages, we have shown empirically that they are much more suited to distinguish several identical instances of the same object, and are complementary to the standard Mask-RCNN approach.

Acknowledgments. We gratefully acknowledge the support of Naver, EPSRC AIMS CDT, AWS ML Research Award, and ERC 677195-IDIU.

Semi-convolutional Operators for Instance Segmentation

Appendix

A Ablation studies

Sec. 4 in the paper quantitatively compares the proposed method to existing approaches. In this section, we conduct additional experiments that analyze the contribution of the proposed semi-convolutional operators. In order to do so, we tested several modifications of our method: **NoSteer** suppresses the proposed steerable component of the pixel-wise embedding ($\Psi_u(x) = \phi_u(x)$). **Bilateral** keeps the steerable component, but concatenates it with the dynamic part rather than combining the two via addition ($\Psi_u(x) = [\phi_u(x); u(x)]$) effectively creating a soft-bilateral filter. Finally, **LinearMap** learns a linear transformation M that maps the pixel locations $u(x)$ to a higher dimensional space and sums the result with the dynamic part of the embedding ($\Psi_u(x) = \phi_u(x) + Mu, M \in \mathbb{R}^{d\times 2}$). The results on the C. Elegans and Pascal VOC datasets are in tables 4 and 5 respectively.

Results show that for the C. Elegans dataset the performance drops below the baseline for all considered modifications. For PASCAL-VOC, the soft-bilateral loss performs similarly to the steerable embedding. This shows the value of the semi-convolutional approach when small, flexible and overlapping object instances are present.

	MRCNN [25]	Bilinear	NoSteer	LinearMap	Ours
AP@0.5-0.95	0.559	0.557	0.544	0.535	**0.569**
AP@0.5	0.865	0.865	0.836	0.830	**0.885**

Table 4. Ablation study on the C. Elegans test set comparing different modifications of our method.

Metric	MRCNN [25]	Bilinear	NoSteer	LinearMap	Ours
AP@0.5-0.95	0.401	0.411	0.409	0.411	**0.412**
AP@0.5	0.690	0.695	0.695	0.693	**0.697**

Table 5. Ablation study on the PASCAL VOC 2012 validation set comparing different modifications of our method.

B Alternative Instance Coloring approach

In Sec. 3.5 in the paper, we extended MRCNN [25] with the proposed semi-convolutional embedding in order to improve its performance. Here, we describe and evaluate an alternative approach that makes use of our embedding and can be regarded as a variant of the Instance Coloring (IC) methods. While the extended MRCNN focused on class-specific instance segmentation, here we consider a simpler problem of class agnostic segment proposal generation. This

task was chosen because our embedding is class-agnostic and hence we cannot trivially produce class-specific groupings of pixels.

In order to simplify our pipeline, we re-use the pixel-wise semi-convolutional instance embeddings Ψ_u produced by the semi-convolutional MRCNN model which was benchmarked in Sec. 4.3. The embedding tensors, each corresponding to a particular FPN layer, were first bilinearly upsampled to a common spatial resolution and later concatenated along the channel dimension in order to produce a robust multiscale pixel-wise semi-convolutional representation $\hat{\Psi}$.

The ensuing grouping mechanism is a two-stage process. First, candidate masks are generated by performing KMeans clustering for various different numbers of estimated centroids producing a set of regions $\{\mathcal{S}_i\}$, where $\mathcal{S}_i \subset \Omega^{fg}$. Here, $\Omega^{fg} = \{u \mid p_u^{fg} > \tau\}$ is a subset of image pixels $u \in \Omega$ with their foreground probability p_u^{fg} higher than a threshold $\tau = 0.5$. The foreground probability predictions $p_u^{fg} = 1 - p_u^{bg}$ were obtained by utilizing "background" class predictions p_u^{bg} of a DeepLabv2 model [7] pre-trained using the semantic segmentation annotations of the Pascal VOC 2012 training set.

In the second stage, we compute an "objectness" score $o(\mathcal{S}) \in [0, 1]$ for each candidate $\mathcal{S}$. The latter is a variant of a normalized cut measure [51] weighted by the average probability of the candidate representing a foreground region. More formally:

$$o(\mathcal{S}) \;=\; p^{fg}(\mathcal{S}) \; \frac{\sum_{(u,v)\in\mathcal{S}\times\mathcal{S}} G^N_{u,v}}{\sum_{(r,k)\in\mathcal{S}\times\Omega} G^N_{r,k}}. \tag{11}$$

Here $G^N_{u,v} = \frac{1}{\sqrt{\deg_u}\sqrt{\deg_v}}$, where $G_{u,v}$ is a gaussian kernel $G_{u,v} = \exp\left(-\frac{\|\hat{\Psi}_u - \hat{\Psi}_v\|^2}{\sigma_g}\right)$ with a variance σ_g that measures similarity between a pair of semi-convolutional embeddings $\hat{\Psi}_u$ and $\hat{\Psi}_v$ normalized by the product of the roots of the per-pixel degrees $\deg_u$ and $\deg_v$, where $\deg_u = \sum_{v\in\Omega} G_{u,v}$. The average foreground probability $p^{fg}(\mathcal{S})$ is defined as $p^{fg}(\mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{u\in\mathcal{S}} p_u^{fg}$.

The final set of segmentation candidates is then obtained by filtering out duplicates using the standard greedy non-maximum-suppression algorithm (NMS) [13] with the overlap threshold set to 0.6. For the NMS algorithm, we have empirically observed that approximating each mask with a bounding box and computing overlaps in this bounding box space gives better results than calculation of the overlaps directly in the mask space.

Segment proposal evaluation. We compare our method against state-of-the-art techniques for object proposals on the VOC validation set (1,449 images). We provide an evaluation under the COCO criterion i.e. we report average recall across thresholds ranging from 0.5 to 0.95 for 10 predicted proposals.

Results. Results are reported in table 6. Encouragingly, our method approaches the performance of existing techniques such as Deep-Mask or Sharp-Mask that are designed for the proposal detection task. However, we observe that our grouping pipeline still lags [24] by a considerable margin. While it was not the primary focus of this work, we believe that this gap can be closed by incorporating our

Method	AR@10
Selective Search [55]	7.0
MCG [46]	18.9
Deep-Mask [44]	30.3
Sharp-Mask [45]	33.3
BAIS [24]	**47.8**
Ours	29.3

Table 6. Segmentation proposal on the PASCAL VOC 2012 validation set. Comparison of segmentation proposal methods using the COCO evaluation criterion (average recall at IoU thresholds from 0.5 to 0.95) [37].

approach into a more robust proposal pipeline. This is a direction that we hope to explore in future work.

References

1. Adam W Harley, Konstantinos G. Derpanis, I.K.: Segmentation-aware convolutional networks using local attention masks. In: Proc. ICCV (2017)
2. Andriluka, M., Stewart, R., Ng, A.Y.: End-to-end people detection in crowded scenes. In: Proc. CVPR (2016)
3. Arnab, A., Torr, P.H.S.: Pixelwise instance segmentation with a dynamically instantiated network. In: Proc. CVPR (2017)
4. Bai, M., Urtasun, R.: Deep watershed transform for instance segmentation. In: Proc. CVPR (2017)
5. Ballard, D.H.: Readings in computer vision: Issues, problems, principles, and paradigms. chap. Generalizing the Hough Transform to Detect Arbitrary Shapes, pp. 714–725. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1987)
6. Chandra, S., Usunier, N., Kokkinos, I.: Dense and Low-Rank Gaussian CRFs Using Deep Embeddings. In: Proc. ICCV (2017)
7. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. PAMI **40**(4), 834–848 (2018)
8. Chen, Y.T., Liu, X., Yang, M.H.: Multi-instance object segmentation with occlusion handling. In: Proc. CVPR (2015)
9. Dai, J., He, K., Li, Y., Ren, S., Sun, J.: Instance-sensitive fully convolutional networks. In: Proc. ECCV (2016)
10. Dai, J., He, K., Sun, J.: Convolutional feature masking for joint object and stuff segmentation. In: Proc. CVPR (2015)
11. Dai, J., He, K., Sun, J.: Instance-aware semantic segmentation via multi-task network cascades. In: Proc. CVPR (2016)
12. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: Proc. NIPS (2016)
13. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR (2005)
14. De Brabandere, B., Neven, D., Van Gool, L.: Semantic instance segmentation with a discriminative loss function. arXiv preprint arXiv:1708.02551 (2017)

15. Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. Commun. ACM **15**(1), 11–15 (1972)
16. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
17. Fathi, A., Wojna, Z., Rathod, V., Wang, P., Song, H.O., Guadarrama, S., Murphy, K.P.: Semantic instance segmentation via deep metric learning. CoRR **abs/1703.10277** (2017)
18. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV **59**(2), 167–181 (2004)
19. Feragen, A., Lauze, F., Hauberg, S.: Geodesic exponential kernels: When curvature and linearity conflict. In: Proc. CVPR (2015)
20. Girshick, R.: Fast r-cnn. In: Proc. ICCV (2015)
21. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc. CVPR (2014)
22. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Proc. ECCV (2014)
23. Harley, A.W., Derpanis, K.G., Kokkinos, I.: Learning dense convolutional embeddings for semantic segmentation. In: Proc. ICLR (2016)
24. Hayder, Z., He, X., Salzmann, M.: Boundary-aware instance segmentation. In: Proc. CVPR (2017)
25. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proc. ICCV (2017)
26. Hu, H., Lan, S., Jiang, Y., Cao, Z., Sha, F.: Fastmask: Segment multi-scale object candidates in one shot. In: Proc. CVPR (2017)
27. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. Proc. CVPR (2017)
28. Kirillov, A., Levinkov, E., Andres, B., Savchynskyy, B., Rother, C.: Instancecut: From edges to instances with multicut. In: Proc. CVPR (July 2017)
29. Kokkinos, I.: Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In: Proc. CVPR (2017)
30. Kong, S., Fowlkes, C.: Recurrent pixel embedding for instance grouping. In: Proc. CVPR (2018)
31. Ladický, L., Sturgess, P., Alahari, K., Russell, C., Torr, P.H.S.: What, where and how many? combining object detectors and crfs. In: Proc. ECCV (2010)
32. Leibe, B., Schiele, B.: Interleaved object categorization and segmentation. In: Proc. BMVC (2003)
33. Li, Y., Qi, H., Dai, J., Ji, X., Wei, Y.: Fully convolutional instance-aware semantic segmentation. In: Proc. CVPR (2017)
34. Liang, X., Wei, Y., Shen, X., Jie, Z., Feng, J., Lin, L., Yan, S.: Reversible recursive instance-level object segmentation. In: Proc. CVPR (2016)
35. Liang, X., Wei, Y., Shen, X., Yang, J., Lin, L., Yan, S.: Proposal-free network for instance-level object segmentation. PAMI (2017)
36. Lin, T., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proc. CVPR (2017)
37. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proc. ECCV (2014)
38. Liu, R., Lehman, J., Molino, P., Such, F.P., Frank, E., Sergeev, A., Yosinski, J.: An intriguing failing of convolutional neural networks and the coordconv solution. arXiv preprint arXiv:1807.03247 (2018)

39. Liu, S., Jia, J., Fidler, S., Urtasun, R.: Sgn: Sequential grouping networks for instance segmentation. In: Proc. ICCV (2017)
40. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: Proc. ECCV (2016)
41. Ljosa, V., Sokolnicki, K.L., Carpenter, A.E.: Annotated high-throughput microscopy image sets for validation. Nat Methods **9**(7), 637 (2012)
42. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. CVPR (2015)
43. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: Proc. NIPS (2017)
44. Pinheiro, P.O., Collobert, R., Dollár, P.: Learning to segment object candidates. In: Proc. NIPS (2015)
45. Pinheiro, P.O., Lin, T., Collobert, R., Dollár, P.: Learning to refine object segments. In: Proc. ECCV (2016)
46. Pont-Tuset, J., Arbelaez, P., Barron, J.T., Marques, F., Malik, J.: Multiscale combinatorial grouping for image segmentation and object proposal generation. PAMI **39**(1), 128–140 (2017)
47. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proc. CVPR (2017)
48. Ren, M., Zemel, R.S.: End-to-end instance segmentation with recurrent attention. In: Proc. CVPR (2017)
49. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Proc. NIPS (2015)
50. Romera-Paredes, B., Torr, P.H.S.: Recurrent instance segmentation. In: Proc. ECCV (2016)
51. Shi, J., Malik, J.: Normalized cuts and image segmentation. PAMI **22**(8), 888–905 (2000)
52. Silberman, N., Sontag, D., Fergus, R.: Instance segmentation of indoor scenes using a coverage loss. In: Proc. ECCV (2014)
53. Tighe, J., Niethammer, M., Lazebnik, S.: Scene parsing with object instances and occlusion ordering. In: Proc. CVPR (2014)
54. Uhrig, J., Cordts, M., Franke, U., Brox, T.: Pixel-level encoding and depth layering for instance-level semantic labeling. In: Proc. GCPR (2016)
55. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. IJCV **104**(2), 154–171 (2013)
56. Wählby, C., Riklin-Raviv, T., Ljosa, V., Conery, A.L., Golland, P., Ausubel, F.M., Carpenter, A.E.: Resolving clustered worms via probabilistic shape models. In: Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on. pp. 552–555. IEEE (2010)
57. Wang, L., Lu, H., Ruan, X., Yang, M.H.: Deep networks for saliency detection via local estimation and global search. In: Proc. CVPR (June 2015)
58. Yurchenko, V., Lempitsky, V.: Parsing images of overlapping organisms with deep singling-out networks. Proc. CVPR (2017)
59. Zhang, Z., Schwing, A.G., Fidler, S., Urtasun, R.: Monocular object instance segmentation and depth ordering with cnns. In: Proc. ICCV (2015)

8

Capturing the Geometry of Object Categories from Video Supervision

This work was published in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018 [Novotny et al. 2018c].

The manuscript is an extension of a work presented as an *oral* presentation at the International Conference on Computer Vision, 2017 [Novotny et al. 2017c].

Capturing the Geometry of Object Categories from Video Supervision

David Novotny, Diane Larlus, and Andrea Vedaldi

Abstract—We propose an unsupervised method to learn the 3D geometry of object categories by looking around them. Differently from traditional approaches, this method does not require CAD models or manual supervision. Instead, using only video sequences showing object instances from a moving viewpoint, the method learns a deep neural network that can predict several aspects of the 3D geometry of such objects from single images. The network has three components. The first is a Siamese viewpoint factorization network that robustly aligns the input videos and learns to predict the absolute viewpoint of the object from a single image. The second is a depth estimation network that performs monocular depth prediction. The third is a shape completion network that predicts the full 3D shape of the object from the output of the monocular depth prediction module. While the three modules solve very different task, we show that they all benefit significantly from allowing networks to perform probabilistic predictions. This results in a self-assessment mechanism which is crucial for obtaining high quality predictions. Our network achieves state-of-the-art results on viewpoint prediction, depth estimation, and 3D point cloud estimation on public benchmarks.

Index Terms—monocular pose estimation, monocular depth estimation, point-cloud estimation, geometry reconstruction

1 Introduction

A remarkable ability of human vision is to reliably estimate the 3D geometry of the visible objects, even from single images. Reproducing this capability in artificial vision systems has important and varied applications, such as helping robots to interact with their surroundings, driving autonomous cars through complex environments, or automatically lifting 2D movies to three dimensions.

Nowadays, mature techniques such as structure-from-motion (SfM) [1] and stereo vision [2] allow to reliably reconstruct the geometry of a *particular* scene given *several images* of it seen under sufficiently different viewpoints. Such images may be extracted as the frames of a video sequence captured by a moving camera, or collected from multiple, independent cameras looking at the same scene, famously including the example of unconstrained photos captured by tourists [3]. These reconstruction algorithms are sufficiently mature to be used in industrial applications. However, the human visual system is arguably capable of solving a significantly more complex reconstruction problem than these, namely estimating the geometry of a scene from a *single* image of it. While recovering geometry from multiple views is a matter of exploiting well defined geometric properties of the optical system formed by two or more cameras, the single-view case is inherently ill-posed. A single image is in fact insufficient to uniquely infer the shape of the objects contained in it (e.g. it is not possible to distinguish between an image of a 3D scene or the image of a photo of it, which is is flat). However, statistical reconstructions are still possible provided that one can exploit the regularity of the geometric patterns that exist in the visual world.

An important source of regularity in 3D reconstruction, and image understanding in general, is the existence of *object categories*. The reason is that objects of the same category usually have similar 3D shapes and share a common object-centric coordinate frame. Thus, identifying an object in an image provides a strong constraint in the reconstruction process, significantly reducing uncertainty. However, doing so requires to model and learn the distribution of possible 3D shapes of the objects of a given category, which is a significant challenge in its own right.

Most approaches to learning 3D categories make use of high quality but expensive supervision. CAD models have been used to fully supervise models to recognize the object viewpoint and 3D shape from a single image [4], [5]. Alternatively, standard image datasets such as PASCAL VOC [6], augmented with additional annotations such as object segmentations and keypoints [7], have been used as a supervisory signal. Whether synthetically generated or manually collected, these annotations have helped to overcome the significant challenges of learning 3D object categories, by making available to the learner ground-truth information about viewpoint, geometry, or both.

In this paper, we aim at *significantly lowering the level of supervision* required to learn the 3D geometry of object categories. In particular, we propose an *unsupervised method* (Fig. 1) that replaces synthetic or manual supervision with *motion*. Humans understand visual scenes by experiencing them from different angles, as these diverse viewpoints provide very strong cues on the geometry of specific object instances. They can then generalize such cues to properties of object categories in general. Our goal is to mimic such interaction and learn the 3D geometry of object categories using videos and no manual annotations.

- *D. Novotny is with the VGG, University of Oxford, UK, and NAVER LABS Europe, Meylan, France. E-mail: david@robots.ox.ac.uk*
- *D. Larlus is with NAVER LABS Europe, Meylan, France. E-mail: diane.larlus@naverlabs.com*
- *A. Vedaldi is with the VGG, University of Oxford, UK. E-mail: vedaldi@robots.ox.ac.uk*

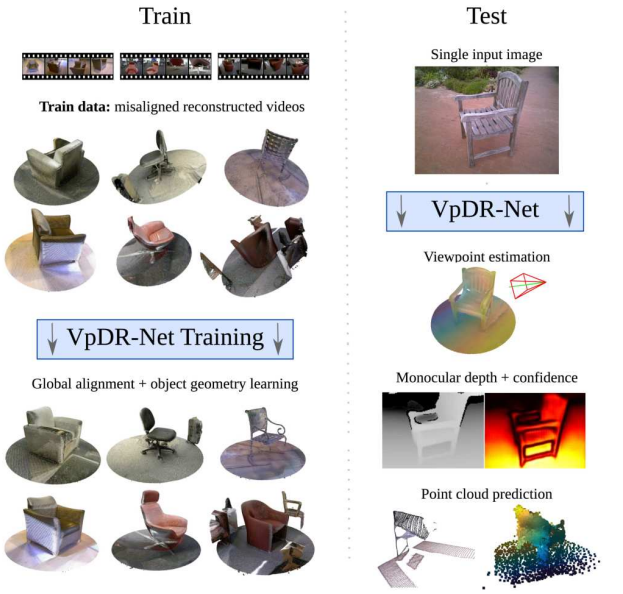


Fig. 1. We propose an architecture to learn the 3D geometry of object categories *from videos only*, without manual annotations. Once learned, the network can predict i) viewpoint, ii) depth, and iii) a point cloud, all from a single image of a new object instance.

In order to automatically generate a supervisory signal from videos, we apply SfM to individual video sequences obtained from moving cameras looking at various instances of a certain object category. As mentioned above, SfM performs well for reconstructing individual object instances, but it is insufficient to learn the shape of such objects *in general*. Thus, the key challenge is to integrate sequence-specific 3D reconstructions into a global geometric model of the object category. This has to be done in a sufficiently robust manner due to the significant level of noise in the SfM reconstructions. To overcome this challenge, we propose three key innovations.

The first innovation is **viewpoint factorization**, a new method to automatically learn to align video sequences of different object instances. Existing approaches to viewpoint alignment [8], [9] try to match 3D shapes by matching corresponding 3D features. We propose instead to learn a network that estimates, given a single image at a time, the *absolute* viewpoint of the object in the image. The network is trained in a Siamese configuration so that the *relative* motion between two images, which can be estimated using SfM, is reconstructed by composition of the absolute viewpoints estimated by the network. In other words, the relative viewpoint is factorized by the network in the product of two absolute viewpoints. We show that this training mechanism implicitly and globally aligns different objects instances while being simpler and more robust than alternatives.

The second innovation is an architecture that can **generate a complete point cloud** for a given object from only a partial reconstruction obtained from monocular depth estimation. This is based on a shape representation that predicts the support of a point probability distribution in the 3D space, akin to a flexible voxelization and a corresponding space occupancy map.

The third innovation is a technique that allows neural networks to **express uncertainty**. More precisely, our networks are designed to automatically predict probability distributions associated to their outputs, which allows them to learn from noisy annotations in a principled manner. We show that, when this mechanism is used, the networks train in a more robust manner.

All three contributions are leveraged by our proposed architecture, a deep network composed of three modules (Fig. 2). The first module estimates the *absolute viewpoint* of objects. This aligns different object instances to a common reference frame where geometric relationships can be modeled more easily. The second module estimates the 3D shape of an object from a given viewpoint, producing a *depth map*. The third module *completes the depth map to a full 3D reconstruction* in a globally-aligned reference frame. Combined and trained end-to-end without manual annotations, from video sequences alone, these components constitute our **VpDR-Net** network, which can jointly estimate the **v**iew**p**oint, the **d**epth and the 3D **r**econstruction of any new object instance from a single image.

This article is an extension and archival version of our previous work [10]. It is organized as follows. Sec. 2 reviews relevant literature. Sec. 3 presents the architecture as well as the training strategy of VpDR-Net. Sec. 4 proposes a novel probabilistic framework for improving the robustness of the VpDR-Net learning process. Sec. 5 shares the learning details. Sec. 6 validates our method empirically and sec. 7 summarizes and discusses our findings.

2 Related work

Capturing the geometry of an object category comprises several subtasks, such as predicting the viewpoint, the depth, and the 3D model of a novel object instance from a single image. This section discusses relevant prior works in these areas.

2.1 Viewpoint estimation

The vast majority of viewpoint estimation methods requires full supervision. Most methods are trained with manual pose annotations [11]–[17]. Full supervision can also be obtained by levering CAD models [14], [18], [19]. In particular, [19] automatically generates viewpoints together with rendered images. This requires to have 3D object models readily available. Both types of approaches rely on an expensive source of supervision. Less supervised, the approach of [20] produces a relative camera pose estimation but takes pairs of images as input.

Only few works have trained models for viewpoint estimation with videos sequences [8], [9]. In order to do so, [8] leverages a generative graphical model to discover object parts while [9] first reconstructs a 3D model per video sequence and then aligns these models.

The task of aligning point clouds is far from trivial. Most existing methods are highly sensitive to noise (see a review in [21]) and require high quality reconstructions. More robust to noise, [22] still requires to match objects of the same shape. Sedaghat et al. [9] solve the shape alignment problem using an appropriate global description of the point

clouds together with a global search strategy based on the pairwise alignment of these point clouds. We depart from this strategy by implicitly aligning point clouds as a part of the training of our Siamese viewpoint factorization network.

2.2 Monocular depth estimation

Depth estimation has been tackled with a large variety of approaches including structure from motion, shape-from-X, or multi-view stereo. In this work, we focus on monocular (i.e. single-view) depth estimation.

Many methods have cast monocular depth estimation as a supervised learning problem, predicting the depth of each pixel using models that have been trained on large datasets annotated with pixel-level ground-truth depth [23]–[25]. Saxena et al. [26] propose a patch-based approach that estimates the 3D location and orientation of local planes to explain each patch, leveraging a dataset of laser scans for training. The predictions are then combined together using an MRF. Liu et al. [25] use a convolutional neural network to learn the weights of the terms of the random fields. Ladicky et al. [23] incorporate semantics into their model to refine the pixel depth estimation. The approach of Karsch et al. [27] retrieves whole depth images from a training set.

More recently, deep learning architectures have been successfully trained for this task. Eigen et al. [24] use a two scale deep network trained with pixel-level depth values. Some works have combined deep architectures with random fields [28] or considered different losses [29], [30].

All these approaches require high quality, pixel aligned, ground truth depth maps at training time. Recently, several works have tackled the problem of learning depth from incomplete or no supervision. Training with image stereo pairs is addressed in [31], [32]. Zhou et al. [33] further decrease the level of supervision and learn a depth and egomotion predictor from unconstrained video sequences.

In this work, we train a neural network architecture for this task using the supervision provided by the reconstructions automatically obtained with an SfM algorithm. In order to cope with the noise in the output of SfM, we devise specific training mechanism including robust probabilistic losses. Our depth predictions are then used to initialize the ensuing 3D shape completion step.

2.3 3D shape prediction

The ability of recovering 3D geometry from a single image is a long standing and challenging problem. Many class-agnostic approaches have been proposed such as shape from shading [34], [35] or from silhouette [36], [37]. Yet, knowing the category of the object to reconstruct allows to leverage useful prior information.

Methods that use a 3D model of the target object go back to the seminal works of Roberts [38] and Lowe [39]. They recently regained popularity with the availability of datasets of 3D CAD models [4], [15]. In one line of research, methods estimate the 3D shape of objects by retrieving and aligning the most similar 3D model from a CAD library [40]–[43]. Other approaches leverage these 3D models to train a network to directly predict the 3D shape of an object in a fully supervised fashion. These methods differ in the type of representation used for the predicted 3D shape. [44], [45] predict a voxel occupancy grid. [46] alleviates the high memory footprint of the voxel-based methods by processing only the voxel grid cells that are predicted to have non-zero occupancy. The approaches from [47]–[49] predict surfaces instead of voxel grids. More related to our approach, [50] learns a variational auto-encoder which outputs a point cloud approximating the surface of an object depicted in a single input image. Yang et al. [51] propose a novel deep point cloud predictor that iteratively folds an initial fixed grid of 3D points. All these methods require handmade 3D CAD models at train time.

2.4 Data-driven approaches for category specific 3D reconstructions

Structure from motion (SfM) [1], [52], [53] can produce high quality 3D reconstruction by matching features across different views of the *same* instance. Matching between *different* instances of a category is much more challenging, and SfM methods generally have difficulties handling the intra-class variations. To overcome this issue, some approaches combine SfM with manual annotations [36], [54], such as keypoints [7], [55] to estimate a rough 3D geometry of objects for unordered sets of images from the same class.

More recently, deep networks have been combined with low-level geometry cues in order to learn category specific shape predictors. Rezende et al. [56] learn 3D structures from various levels of supervision, where the lowest level comprises multiple views of an object. Similarly, [57] exploits multi-view segmentation masks and depth maps while [58], [59] use object silhouettes. All these works assume knowledge of the ground truth camera viewpoint. In this work we do not need any additional annotations as we leverage motion cues.

3 Proposed Architecture

We propose a single Convolutional Neural Network (CNN), VpDR-Net, that learns a *3D object category* by observing it from a *variable viewpoint* in videos and with no supervision (Fig. 2). The key insight is that, while videos do not solve the problem of relating the 3D shape of different object instances, they at least provide powerful if noisy cues about the 3D shape of the individual instances.

At training time, VpDR-Net takes as an input a set of K video sequences $S^1, ..., S^K$ of an object category (such as cars or chairs), where a video $S^i = (f_1^i, ..., f_{N^i}^i)$ contains N^i RGB or RGBD frames $f_t^i \in \mathbb{R}^{H \times W \times \mathcal{C}}$ (where $\mathcal{C} = 3$ for RGB and $\mathcal{C} = 4$ for RGBD data) and learns a model of the 3D category. VpDR-Net, illustrated in Fig. 2, has three components: i) a predictor $\Phi_{\text{vp}}(f_i^t)$ of the *absolute viewpoint* of the object implicitly aligning the different object instances to a common reference frame (sec. 3.1.2); ii) a *monocular depth* predictor $\Phi_{\text{depth}}(f_i^t)$ (sec. 3.2) and iii) a *shape* predictor $\Phi_{\text{pcl}}(f_i^t)$ that extends the depth map to a point cloud capturing the complete shape of the object (sec. 3.3). Learning starts by preprocessing videos to extract instance-specific egomotion and shape information (sec. 3.1.1).

At test time, VpDR-Net takes a single image as input and can estimate simultaneously the viewpoint, the depth map, and the 3D reconstruction of the object contained in it.

3.1 Viewpoint prediction module

3.1.1 Preprocessing

Video sequences are pre-processed to extract from each frame f_t^i a tuple (K_t^i, g_t^i, D_t^i) consisting of: (i) the camera calibration parameters K_t^i, (ii) the camera pose $g_t^i \in SE(3)$, and (iii) a depth map $D_t^i \in \mathbb{R}^{H\times W}$ associating a depth value to each pixel of f_t^i. The camera pose $g_t^i = (R_t^i, T_t^i)$ consists of a rotation matrix $R_t^i \in SO(3)$ and a translation vector $T_t^i \in \mathbb{R}^3$. We use the convention that g_t^i transforms world-relative coordinates $p_{\text{world}} \in \mathbb{R}^3$ to camera-relative coordinates, i.e. $p_{\text{camera}} = g_t^i p_{\text{world}}$.

We extract this information using off-the-shelf methods: the structure-from-motion (SfM) algorithm COLMAP for RGB sequences [60], [61], and an open-source implementation [62] of ORB-slam2 (OS) [63] for RGBD sequences. The information extracted from RGB or RGBD data is qualitatively similar, except that the scale of SfM reconstructions is arbitrary.

3.1.2 Intra-sequence alignment

Methods such as SfM or OS can reliably estimate camera pose and depth information for single objects and individual video sequences, but are not applicable to *different instances and sequences*. In fact, their underlying assumption is that geometry is fixed, which is true for single (rigid) objects, but false when the geometry and appearance differ due to intra-class variations.

Learning 3D object categories requires to relate their variable 3D shapes by identifying and putting in correspondence analogous geometric features, such as the object front and rear. For rigid objects, such correspondences can be expressed by rigid transformations that *align* occurrences of analogous geometric features. The most common approach for aligning 3D shapes, also adopted by [9] for video sequences, is to extract and match 3D feature descriptors. Once objects in images or videos are aligned, the data can be used to supervise other tasks, such as learning a monocular predictor of the absolute viewpoint of an object [9].

One of our main contributions, described below, is to reverse this process by learning a viewpoint predictor *without* explicitly matching 3D shapes. Empirically (sec. 6), we show that, by skipping the intermediate 3D analysis, our method is often more effective and robust than alternatives.

Siamese network for viewpoint factorization. Geometric analogies between 3D shapes can often be detected in image space directly, based on visual similarity. Thus, we propose to train a CNN Φ_{vp} that maps a single frame f_t^i to its *absolute viewpoint* $\hat{g}_t^i = \Phi_{\text{vp}}(f_t^i)$ in the globally-aligned reference frame. We wish to learn this CNN from the viewpoints estimated by the algorithms of sec. 3.1.1 for each video sequence. However, these estimated viewpoints are *not* absolute, but valid only within each sequence; formally, there are unknown sequence-specific motions $h^i = (R^i, T^i) \in SE(3)$ that map the sequence-specific camera poses g_t^i to global poses $\hat{g}_t^i = g_t^i h^i$. Note that h^i composes to the right: it transforms the world reference frame and then moves it to the camera reference frame.

To address this issue, we propose to supervise the network using *relative pose changes within each sequence*, which are invariant to the alignment transformation h^i. Formally, the transformation h^i is eliminated by computing the relative pose change of the camera from frame t to frame t':

$$\hat{g}_{t'}^i(\hat{g}_t^i)^{-1} = g_{t'}^i h^i (h^i)^{-1}(g_t^i)^{-1} = g_{t'}^i (g_t^i)^{-1}. \tag{1}$$

Expanding the expression with $\hat{g}_t^i = (\hat{R}_t^i, \hat{T}_t^i)$, we find equations expressing the relative rotation and translation

$$\hat{R}_{t'}^i (\hat{R}_t^i)^\top = R_{t'}^i (R_t^i)^\top, \tag{2}$$

$$\hat{T}_{t'}^i - R_{t'}^i (R_t^i)^\top \hat{T}_t^i = T_{t'}^i - R_{t'}^i (R_t^i)^\top T_t^i. \tag{3}$$

Eqs. (2) and (3) are used to constrain the training of a *Siamese architecture*, which, given two frames t and t', evaluates the CNN twice to obtain estimates $(\hat{R}_t^i, \hat{T}_t^i) = \Phi_{\text{vp}}(f_t^i)$ and $(\hat{R}_{t'}^i, \hat{T}_{t'}^i) = \Phi_{\text{vp}}(f_{t'}^i)$. The estimated poses are then compared to the ground truth ones, (R_t^i, T_t^i) and $(R_{t'}^i, T_{t'}^i)$, in a relative manner by using losses that enforce the estimated poses to satisfy eqs. (2) and (3):

$$\ell_R(\hat{R}_t^i, \hat{T}_t^i, \hat{R}_{t'}^i, \hat{T}_{t'}^i) \doteq \| \ln \hat{R}_{tt'}^i (R_{tt'}^i)^\top \|_F \tag{4}$$

$$\ell_T(\hat{R}_t^i, \hat{T}_t^i, \hat{R}_{t'}^i, \hat{T}_{t'}^i) \doteq \| \hat{T}_{tt'}^i - T_{tt'}^i \|_2 \tag{5}$$

where $\ln$ is the principal matrix logarithm and

$$R_{t't}^i \doteq R_{t'}^i (R_t^i)^\top, \qquad \hat{R}_{t't}^i \doteq \hat{R}_{t'}^i (\hat{R}_t^i)^\top, \tag{6}$$

$$T_{t't}^i \doteq T_{t'}^i - R_{t't}^i T_t^i, \qquad \hat{T}_{t't}^i \doteq \hat{T}_{t'}^i - \hat{R}_{t't}^i \hat{T}_t^i. \tag{7}$$

Discussion. While this CNN is only required to correctly predict relative viewpoint changes *within each sequence*, since the *same CNN* is used for all videos, the most plausible/regular solution for the network is to assign similar viewpoint predictions $(\hat{R}_t^i, \hat{T}_t^i)$ to images viewed from the same viewpoint, leading to a globally consistent alignment of the input sequences. Furthermore, in a large family of 3D objects, different ones (e.g. SUVs and sedans) tend to be mediated by intermediate cases. This is shown empirically in sec. 6.

3.1.3 Scale ambiguity in SfM

For methods such as SfM, there is an additional ambiguity: reconstructions are known only up to sequence-specific scaling factors $\lambda^i > 0$, so that the camera pose is parametrized as $g_t^i(\lambda^i) = (R_t^i, \lambda^i T_t^i)$. This ambiguity leaves eq. (2) unchanged, but eq. (3) becomes:

$$\hat{T}_{t'}^i - \hat{R}_{t't}^i \hat{T}_t^i = \lambda^i (T_{t'}^i - R_{t't}^i T_t^i) \quad \Rightarrow \quad \hat{T}_{t't}^i = \lambda^i T_{t't}^i.$$

During training, the ambiguity can be removed from loss (5) by dividing vectors $T_{t't}^i$ and $\hat{T}_{t't}^i$ by their Euclidean norm so λ^i is not required to learn Φ_{vp}. Yet, λ^i is important for depth prediction, so we estimate it as well. To do so, we note that, given a pair of frames (t, t') from sequence S^i, one can estimate the sequence scale as

$$\lambda_{t,t'}^i = \frac{\|T_{t'}^i - R_{t't}^i T_t^i\|}{\|\hat{T}_{t'}^i - R_{t't}^i \hat{T}_t^i\|}. \tag{8}$$

This expression allows to conveniently estimate λ^i as a moving average during the training iterations, as sample values of $\lambda_{t,t'}^i$ can be computed for free when training ϕ_{vp}. Note that $\lambda^i = 1$ for OS sequences with metric depth.

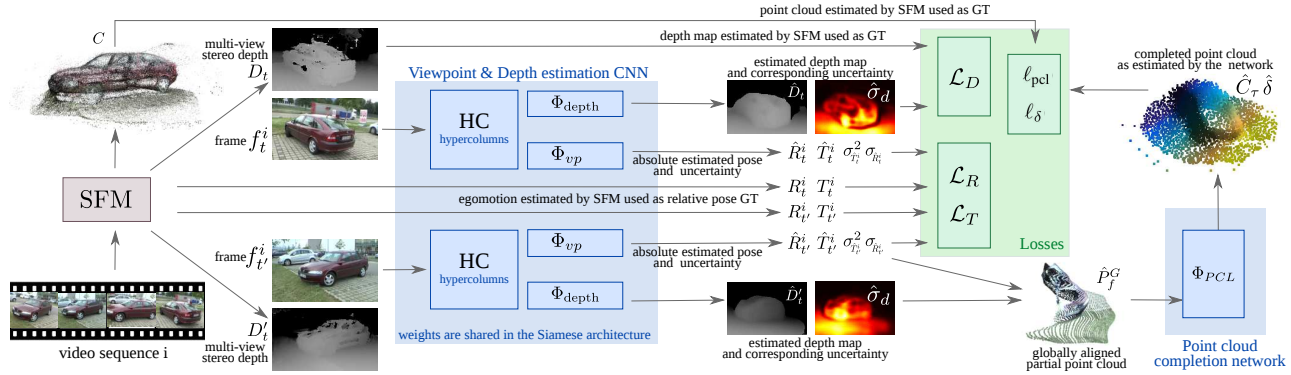


Fig. 2. **Overview of our architecture.** As a preprocessing, structure from motion (SfM) extracts egomotion and a depth map for every frame. For training, our architecture takes pairs of frames f_t, $f_{t'}$ and produces a viewpoint estimate, a depth estimate, and a 3D geometry estimate. At test time, viewpoint, depth, and 3D geometry are predicted from single images.

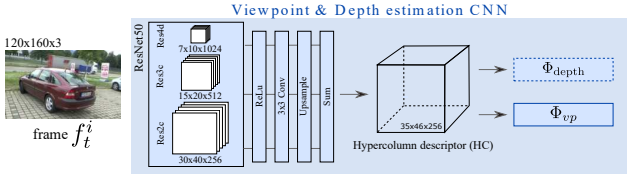


Fig. 3. **The core architecture of VpDR-Net.** This figure describes the architecture of the hypercolumn (HC) module.

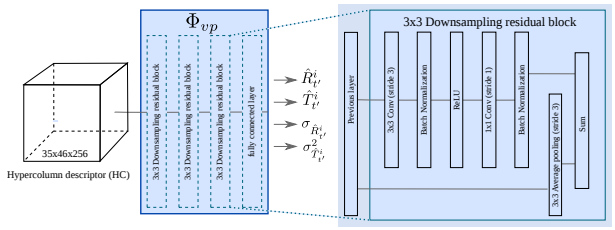


Fig. 4. **The architecture of Φ_{vp}.** Left: the layers of Φ_{vp}. Right: detail of the 3×3 downsampling residual block.

3.1.4 *Architecture*

The viewpoint estimation branch Φ_{vp} of our network is a convolutional architecture. Its lower part is shared between the viewpoint and the depth prediction branches. It is a variant of ResNet-50 [64] with some modifications to improve its performance as viewpoint predictor. First, in order to decrease the degree of geometrical invariance of the network, we replace all 1×1 downsampling filters with full 2×2 convolutions. We then add bilinear upsampling layers that first resize features from three different layers of the architecture (res2c, res3c, res4d) into fixed-size tensors and then sum them in order to create a multiscale intermediate image representation which resembles hypercolumns (HC) [65]. An extension of Fig. 2 that illustrates these layers responsible for the computation of the HC multiscale representation can be found in Fig. 3.

The upper part of Φ_{vp} is specific to the viewpoint prediction branch. HC is followed by 3 modified 3×3 downsampling residual layers that produce the final viewpoint prediction. While the standard downsampling residual layers do not contain the residual skip connection due to different sizes of the input and output tensors, here we retain the skip connection by performing 3×3 average pooling over the input tensor and summing the result with the result of the second 3×3 downsampling convolution branch. We further remove the ReLU after the final residual summation layer. Fig. 4 contains an overview of the viewpoint estimation module together with a detailed illustration of the modified downsampling residual blocks.

3.2 Depth prediction branch

An estimation of the viewpoint of an object is already a powerful geometrical cue allowing to relate it to a 3D scene. In this section, we describe the second branch of our network, a depth prediction module that estimates the 3D structure of the part of the object that is visible in the image.

Monocular depth prediction. The depth predictor module Φ_{depth} of VpDR-Net takes individual frames f_t^i and outputs a corresponding depth map $\hat{D}_t = \Phi_{\text{depth}}(f_t^i)$, performing monocular depth estimation. The depth map $\hat{D}_t$ is the same size as the input image and gives, for each pixel, an estimation of its distance from the camera.

In order to learn Φ_{depth} a standard approach is to minimize a distance metric between the predicted depth $\hat{D}_t$ and the ground truth D_t. Recently, [30] proposed to use the BerHu loss - a reversed version of the Huber loss which adaptively sets the cut-off threshold where the loss transitions from the ℓ_1 into the ℓ_2 part. Note that although VpDR-Net does not use this type of loss, here we describe the approach of [30] as it is later used as a non-probabilistic baseline we compare against.

Architecture. The architecture of Φ_{depth} shares the early HC layers with the viewpoint factorization network Φ_{vp}. The remainder of the pipeline is based on the state-of-the-art depth estimation method of [30]. More precisely, the network is composed of two standard residual blocks, two 2×2 up-projection layers similar to the ones from [30], leading to a 64-dimensional representation of the same size as the input image. These layers are followed by a 1×1 convolutional filter that predicts the depth map $\hat{D}_t$. This is illustrated in Fig. 5.

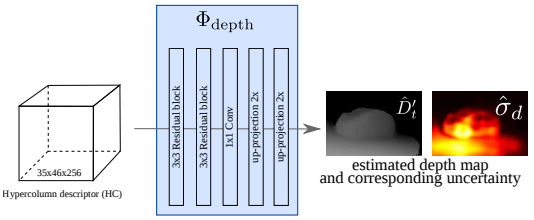


Fig. 5. **The architecture of Φ_{depth}.** Illustration of the layers specific to the depth prediction branch which predicts a depth map (sec. 3.2) and optionally an uncertainty map (sec. 4.3)

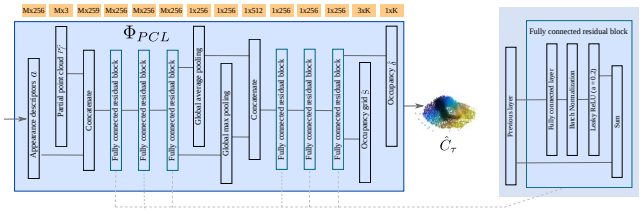


Fig. 6. **The architecture of Φ_{pcl}.** Left: The overview of the point cloud completion network, right: A detail of the fully connected residual block. Orange boxes denote the sizes of the layer outputs.

3.3 Point-cloud completion branch

Given any image f of an object instance, its *aligned 3D shape* can be reconstructed by estimating and aligning its depth map using the output of the viewpoint and depth predictors of sec. 3.1.2 and 3.2. However, since a depth map cannot represent the occluded portions of the object, such a reconstruction can only be partial. In this section we describe the third and last component of VpDR-Net, whose goal is to generate a full reconstruction of the object, beyond what is visible in the given view.

3.3.1 Partial point cloud

The first step is to convert the predicted depth map $\hat{D}_f = \Phi_{\text{depth}}(f)$ into a partial point cloud

$$\hat{P}_f \doteq \{\hat{p}_j : j = 1, \dots, HW\}, \qquad \hat{p}_j \doteq K^{-1} \begin{bmatrix} u_j \\ v_j \\ \hat{d}_i \end{bmatrix},$$

where (u_j, v_j) are the coordinates of a pixel j in the depth map $\hat{D}_f$ and K is the camera calibration matrix. Empirically, we have found that the reconstruction problem is much easier if the data is aligned in the global reference frame established by VpDR-Net. Thus, we transform $\hat{P}_f$ into a globally-aligned point cloud as $\hat{P}^G_f = \hat{g}^{-1}\hat{P}_f$, where $\hat{g} = \Phi_{\text{vp}}(f)$ is the camera pose estimated by the viewpoint-prediction network.

3.3.2 Point cloud completion network

Next, our goal is to learn the point cloud completion part of our network Φ_{pcl} that takes the aligned but incomplete point could $\hat{P}^G_f$ and produces a complete object reconstruction $\hat{C}$. We do so by predicting a 3D occupancy probability field. However, rather than using a volumetric method that may require a discrete and fixed voxelization of space, we propose a simple and efficient alternative. First, the network Φ_{pcl} predicts a set of M 3D points $\hat{S} = (\hat{s}_1, \dots, \hat{s}_M) \in \mathbb{R}^{3\times M}$ that, during training, closely fit the ground truth 3D point cloud C. This step minimizes the fitting error:

$$\ell_{\text{pcl}}(\hat{S}) = \frac{1}{|C|}\sum_{c\in C} \min_{m=1,\dots,M} \|c - \hat{s}_m\|_2 \,. \tag{9}$$

The 3D point cloud $\hat{S}$ provides a good coverage of the ground truth object shape. However, this point cloud is conservative and distributed *in the vicinity* of the ground truth object. Thus, while this is not a precise representation of the object shape, it works well as a support of a probability distribution of space occupancy. In order to estimate the occupancy probability values, the network $\Phi_{\text{pcl}}(\hat{P}^G_f)$ predicts additional scalar outputs

$$\delta_m = |\{c \in C : \forall m' : \|\hat{s}_m - c\|_2 \le \|\hat{s}_{m'} - c\|_2\}| \;/\; |C|$$

proportional to the number of ground truth surface points $c \in C$ for which the support point $\hat{s}_m$ is the nearest neighbor. The network is trained to compute a prediction $\hat{\delta}_m$ of the occupancy masses δ^γ_m by minimizing the squared error loss $\ell_\delta(\hat{\delta}, \delta) = \sum_{m=1}^M (\hat{\delta}_m - \delta^\gamma_m)^2$. Here, raising δ_m to the power of $\gamma < 1$ prevents overweighting the support points with excessive probability masses which often correspond to "sinks" for the distant outlier points. Furthermore, δ^γ_m can be interpreted as a probability of at least one surface point being present in the vicinity of s_m. We set $\gamma = 1/3$ in our experiments.

Given the network prediction $(\hat{S}, \hat{\delta}) = \Phi_{\text{pcl}}(\hat{P}^G_f)$, the completed point cloud is then defined as the subset of points $\hat{C}$ that have sufficiently high occupancy, defined as: $\hat{C}_\tau = \{\hat{s}_m \in \hat{S} : \hat{\delta}_m \ge \tau\}$ where τ is a confidence parameter. The set $\hat{C}_\tau$ can be further refined by using e.g. a 3D Laplacian filter to smooth out noise.

Architecture. The point cloud completion network Φ_{pcl} is modeled after PointNet [66], originally proposed to semantically *segment* a point clouds. Here we adapt it to perform a completely different task, namely 3D shape reconstruction. This is made possible by our model where shape is represented as a cloud of 3D support points $\hat{S}$ and their occupancy masses $\hat{\delta}$.

Differently from Φ_{vp} and Φ_{depth}, the point cloud completion network Φ_{pcl} is not convolutional but uses a sequence of residual fully connected layers to process the 3D points in $\hat{P}^G_f$, after appending an appearance descriptor to each of them. A key step is to add an intermediate orderless pooling operator to remove the dependency on the order and number of input points.

In more details, the network starts by appending to each 3D point $\hat{p}_i \in \hat{P}^G_f \subset \mathbb{R}^3$ an appearance descriptor a_i and processes this input with an MLP with an intermediate pooling operator:

$$(\hat{S}, \hat{\delta}) = \Phi_{\text{pcl}}(\hat{P}^G_f) = \text{MLP}_2\left(\operatorname*{pool}_{1\le i\le|\hat{P}^G_f|} \text{MLP}_1(\hat{p}_i, a_i)\right).$$

The intermediate pooling operator, which is permutation invariant, removes the dependency on the number and order of input points $\hat{P}^G_f$. In practice, the pooling operator

uses both max pooling and sum pooling, stacking the results of the two.

For the appearance descriptors, recall that each point $\hat{p}_i$ is the back-projection of a certain pixel (u_i, v_i) in image f. To obtain the appearance descriptor a_i we reuse the HC features from the core architecture and sample a column of feature channels at location (u_i, v_i) using bilinear sampling. Note that, following [67], the fully connected residual blocks contain leaky-ReLUs with the leak factor set to 0.2. A diagram depicting Φ_{pcl} can be found in Fig. 6. The architecture is configured to predict $M = 10^4$ points $\hat{S}$.

Point cloud sub-sampling. During training the incomplete point cloud $\hat{P}_f^G$ is downsampled by randomly selecting $M = 10^4$ points based on their depth prediction confidence as estimated by Φ_{depth}. This allows the network to implicitly discard background points (as these are assigned low confidence by depth prediction). Due to this reason, at test time, the point cloud sub-sampling is also used with $M = 10^4$.

4 Probabilistic Learning

4.1 Motivation

In the previous section we have presented a basic version of our VpDR-Net network. Although such architecture can be expected to converge and subsequently perform well in standard fully-supervised settings, note that our supervisory signal can contain a significant amount of noise as it is obtained automatically by applying 3D reconstruction to RGB or RBGD images with the COLMAP [60] and ORB-slam2 [63] algorithms respectively. Typically, reconstruction methods fail for transparent regions or around specularities.

Hence, one of our key contributions, which is described in this section, consists of allowing our VpDR-Net to explicitly express this uncertainty in the ground-truth and subsequently use it in order to: (1) obtain more robust training losses; and (2) enable our model to predict the degree of reliability of the predictions.

In order to do so, we present a generic probabilistic framework where, rather than directly predicting the target values, we instruct our network to predict parameters of a distribution that approximates the predicted values. Once our regressor predicts such parameters, the actual output value corresponds to the mean of the predicted distribution (i.e. the most likely value of the distribution), while the variance of the distribution defines how concentrated is the probability mass around the most likely value, hence can be interpreted as a degree of uncertainty.

In what follows, we present a probabilistic extension of the architecture and original training losses described in the previous section.

4.2 Probabilistic predictions for viewpoint estimation

Due to intrinsic ambiguities in the images or to errors in the SfM supervision (caused for example by reflective or textureless surfaces), the viewpoint prediction branch of our network is occasionally unable to predict the ground truth viewpoint accurately. We found beneficial to allow the network to explicitly learn these cases and express uncertainty as an additional input-dependent prediction.

Recall that the viewpoint prediction branch Φ_{vp} predicts an absolute viewpoint $\hat{g}_t^i = \Phi_{\text{vp}}(f_t^i)$ for an input frame f_t^i, where the viewpoint is composed of a translation component $\hat{T}_t^i$ and a rotation component $\hat{R}_t^i$.

For the translation part, we modify the network to predict the absolute pose $\hat{T}_t^i$ as well as its associated confidence score $\sigma_{\hat{T}_t^i}$. We then model the relative translation as a Gaussian distribution with standard deviation $\sigma_T = \sigma_{\hat{T}_{t'}^i} + \sigma_{\hat{T}_t^i}$ and our model is now learned by minimizing the negative log-likelihood $\mathcal{L}_T$ which replaces the loss ℓ_T:

$$\mathcal{L}_T = -\ln \frac{1}{(2\pi\sigma_T^2)^{\frac{3}{2}}} \exp\left(-\frac{1}{2}\frac{\ell_T^2}{\sigma_T^2}\right). \tag{10}$$

The rotation component is more complex due to the non-Euclidean geometry of $SO(3)$, but it was found sufficient to assume that the error term (4) has Laplace distribution and optimize

$$\mathcal{L}_R = -\ln \frac{1}{C_R} \exp\left(-\frac{\ell_R}{\sigma_R}\right), \sigma_R = \sigma_{\hat{R}_{t'}^i} + \sigma_{\hat{R}_t^i}, \tag{11}$$

where $C_R = \sigma_R(1 - \exp(-\sigma_R^{-1}\pi))$ is a normalization term ensuring that the probability distribution integrates to one on the interval of attainable values of $\ell_R \in [0, \pi]$.

Note that this definition of the loss not only allows to predict the degree of uncertainty, but it also allows to increase the robustness of the training. This is because by optimizing the losses $\mathcal{L}_R$ and $\mathcal{L}_T$ instead of ℓ_R and ℓ_T, the network can discount gross errors by dividing the losses by a large predicted variance.

Modification of the architecture. On top of the use of different losses, the network architecture is slightly modified to predict confidence scores. The hypercolumns module remains unchanged. The upper part of Φ_{vp} is updated to predict four values T_t^i, R_t^i, $\sigma_{\hat{T}_t^i}$ and $\sigma_{\hat{R}_t^i}$, instead of two. The confidence scores $\sigma_{\hat{T}_t^i}$ and $\sigma_{\hat{R}_t^i}$ are predicted as the output of a soft ReLU units to ensure positivity.

4.3 Probabilistic predictions for depth estimation

Estimating depth from a single image is inherently ambiguous and requires comparing the image to internal priors of the object shape. Additionally, our supervisory signal is automatically generated from the SfM reconstructions, leading to annotation errors, as discussed in sec. 4.2.

Similar to pose, we allow the network to explicitly *learn and express uncertainty* about depth estimates by predicting a posterior distribution over possible pixel depths. For robustness to outliers, we assume a Laplace distribution with negative log-likelihood loss

$$\mathcal{L}_D = \sum_{j=1}^{WH} -\ln \frac{\sqrt{2}}{2\hat{\sigma}_{d_j}} \exp\left(-\frac{\sqrt{2}\,|d_j - \hat{\lambda^i}^{-1}\hat{d}_j|}{\hat{\sigma}_{d_j}}\right), \tag{12}$$

where d_j is the noisy ground truth depth output by the reconstruction algorithm (COLMAP or ORB-slam2) for a given pixel j, $\hat{d}_j$ and $\hat{\sigma}_{d_j}$ are respectively the corresponding predicted depth mean and standard deviation. Due to a heavy presence of outliers in our ground truth depth data, we selected the Laplace distribution because it is a straightforward extension of the robust ℓ_1 regression loss.

Fig. 7. **Data augmentation**. Training samples generated leveraging monocular depth estimation (**ours**, top) and using depth from ORB-slam2 (baseline, bottom). Missing pixels due to missing depth in red.

An alternative approach consisting of extending the ℓ_2 loss into a Gaussian distribution (as done in sec. 4.2) was not considered because the ℓ_1 loss is known to be more robust to outliers than ℓ_2. We have not used an equivalent of the Laplacian distribution for viewpoint prediction due to the fact that its generalization to higher dimensions leads to non-trivial distributions [68].

The loss $\mathcal{L}_D$ depends on the relative scale $\hat{\lambda}^i$. For RGBD images and the ORB-slam2 algorithm $\hat{\lambda}^i = 1$. For RGB images and SfM, $\hat{\lambda}^i$ is estimated as explained in sec. 3.1.3.

Modification of the architecture. As before, the architecture of Φ_{depth} shares the early HC layers with the viewpoint factorization network Φ_{vp}. The remainder of the architecture is slightly extended with a second 1×1 convolutional filter that predict the confidence maps $\hat{\sigma}_{d_j}$ to complement the first 1×1 convolutional filter predicting the depth map $\hat{D}_t$.

5 Training the Model

The two previous sections described our network in detail, including the architecture of its three modules, respectively responsible for the prediction of an absolute viewpoint, a depth map, and a point cloud. These sections also discussed appropriate losses to train the network. In particular, sec. 4 described how to equip the network with a probabilistic introspection mechanism by training it with probabilistic losses. In this section, we describe implementation details that were found to be crucial for successfully training this network. First, we show how we perform data augmentation for these geometric prediction tasks (sec. 5.1) and we then provide technical details for reproducibility (sec. 5.2).

5.1 Geometry-aware data augmentation

As viewpoint prediction with deep networks benefits significantly from large training sets [19], we increase the effective size of the training videos by *data augmentation*. This is trivial for tasks such as classification, where one can translate or scale an image without changing its identity. The same is true for viewpoint recognition if the task is to only estimate the viewpoint orientation as in [17], [19], as images can be scaled and translated without changing the equivalent viewpoint orientation. However, this assumption is not satisfied if, as in our case, the goal is to estimate all 6 DoF of the camera pose.

Inspired by the approach of [69], we propose to solve this problem by using the estimated scene geometry to *generate new realistic viewpoints* (Fig. 7). Given a sample frame together with its global pose and depth map i.e. a triplet (f_t^i, g_t^i, D_t^i), we apply a random perturbation to the viewpoint (with a forward bias to avoid unoccluding too many pixels) and use depth-image-based rendering (DIBR) [70] to generate a new sample (f_*^i, g_*^i, D_*^i), warping both the image and the depth map, and computing the new global pose.

Sometimes the depth map D_t^i produced by the ORB-slam2 algorithm contains too many holes to yield satisfactory DIBR results (fig. 7, bottom); we found preferable to use the depth $\hat{D}_t^i = \Phi_{\text{depth}}(f_t)$ estimated by our network which is less accurate but more robust, containing almost no missing pixels (fig. 7, top).

5.2 Learning details

The VpDR-Net network is trained with stochastic gradient descent with a momentum of 0.0005 and an initial learning rate of 10^{-2}. The weights of the losses were empirically set to achieve convergence on the training set.[1] When possible, convolutional filters were initialized with the ResNet50 weights pretrained on the ImageNet classification task. Note that, for the motorbike category where the dataset mostly contains extremely zoomed-in frames, we altered the predicted relative translations $\hat{T}^i_{t't} \doteq \hat{T}^i_{t'} - R^i_{t't}\hat{T}^i_t$ from eq. (7) to use the ground truth provided rotation $R^i_{t,t'}$ instead of the predicted $\hat{R}^i_{t,t'}$. In practice, this greatly improved the convergence speed for this category.

Better convergence was observed by training VpDR-Net in two stages. First, Φ_{depth} and Φ_{vp} were optimized jointly, lowering the learning rate tenfold when no further improvement in the training losses was observed. Then, Φ_{pcl} is optimized after initializing the bias of its last layer, which corresponds to an average point cloud of the object category, by randomly sampling points from the ground truth models.

Training minibatches were formed by first sampling a video sequence from a uniform distribution and then randomly picking an image from the sequence twice in order to obtain the final image pair. The batch size was set to 8 image pairs. In order to boost the invariance to input image noise, we blur each training image with a Gaussian filter who's variance is randomly sampled from the interval $(0, 1]$.

6 Experiments

In this section, after introducing the datasets we use in our experimental evaluation (sec. 6.1), we assess our approach on the three geometric inference tasks: viewpoint estimation in sec. 6.2, depth prediction in sec. 6.3, and point cloud prediction in sec. 6.4.

6.1 Datasets

Throughout the experimental section, we consider three datasets for training and benchmarking our network.

FreiburgCars (FrC) [9] is a set of RGB video sequences with the camera circling around 52 different models of cars. The length of videos ranges from 30 seconds to 2 minutes. Each video has been subsampled to roughly 1000 frames.

The **Large Dataset of Object Scans (LDOS)** dataset [71] contains RGBD sequences of man-made objects. We considered the bike, chair, and motorbike categories. We used 126,

1. The exact values of the loss weights are: $w(\mathcal{L}_T) = w(\mathcal{L}_R) = 0.001$, $w(\mathcal{L}_D) = 0.01$, $w(\ell_{pcl}(\hat{S})) = 1000$, $w(\ell_\delta) = 1$, where $w(\mathcal{L})$ is the weight set for the loss $\mathcal{L}$.

object class	test set	manual annot.	method	$\downarrow e_R$	$\downarrow e_C$	$\downarrow e_R^{rel}$	$\downarrow e_T^{rel}$	$\uparrow AP_{e_R}$	$\uparrow AP_{e_C}$
car	Pascal3D	Yes	VPNet + Pascal3D	12.45	1.26	20.35	0.24	0.77	0.74
		No	VPNet + aligned LDOS	49.62	32.29	85.45	0.84	0.15	0.00
		No	**VpDR-Net (ours)**	**29.57**	**7.29**	**62.30**	**0.65**	**0.41**	**0.91**
chair	Pascal3D	Yes	VPNet + Pascal3D	21.63	4.14	39.61	0.48	0.45	0.82
		No	VPNet + aligned LDOS	55.55	41.06	90.94	0.88	0.18	0.00
		No	**VpDR-Net (ours)**	**33.70**	**14.23**	**57.61**	**0.72**	**0.35**	**0.34**
	LDOS	Yes	VPNet + Pascal3D	49.09	8.06	81.57	0.90	0.18	0.00
		No	VPNet + aligned LDOS	40.18	0.60	86.95	0.81	0.24	0.27
		No	**VpDR-Net (ours)**	**27.80**	**0.46**	**55.13**	**0.58**	**0.46**	**0.46**
motorbike	Pascal3D	Yes	VPNet + Pascal3D	21.74	2.64	34.95	0.43	0.56	0.98
		No	VPNet + aligned LDOS	140.21	58.45	128.37	0.99	0.00	0.00
		No	**VpDR-Net (ours)**	**68.67**	**11.88**	**92.09**	**1.05**	**0.08**	**0.52**
	LDOS	Yes	VPNet + Pascal3D	70.24	5.77	98.06	1.03	0.04	0.00
		No	VPNet + aligned LDOS	132.77	1.38	113.09	0.95	0.00	0.01
		No	**VpDR-Net (ours)**	**31.35**	**0.57**	**60.06**	**0.59**	**0.41**	**0.26**
bicycle	Pascal3D	Yes	VPNet + Pascal3D	23.76	3.09	46.29	0.59	0.43	0.95
		No	VPNet + aligned LDOS	114.51	37.25	124.36	1.02	0.00	0.01
		No	**VpDR-Net (ours)**	**81.84**	**24.35**	**91.27**	**1.15**	**0.00**	**0.05**
	LDOS	Yes	VPNet + Pascal3D	56.72	6.67	92.86	0.99	0.12	0.00
		No	VPNet + aligned LDOS	112.06	1.39	106.92	0.95	0.00	0.00
		No	**VpDR-Net (ours)**	**51.25**	**0.77**	**76.26**	**0.81**	**0.11**	**0.14**

TABLE 1

Viewpoint prediction. Angular error e_r and camera-center distance e_c for absolute pose evaluation, and relative camera rotation error e_R^{rel} and translation error e_T^{rel} for relative pose evaluation. AP_{e_R} and AP_{e_C} evaluate absolute angular error and camera-center distance of the pose predictions taking into account the associated estimate confidence values. **VpDR-Net** trained on unconstrained video sequences, is compared to VPNet-unsupervised trained on the same video sequences, aligned with the method of [9] (VPNet + aligned LDOS), and a fully-supervised VPNet (VPNet + Pascal3D). ↑ (resp. ↓) means larger (resp. lower) is better.

77, and 102 videos for the chair, motorbike and bike classes respectively. The average length of each video is 2383 frames which corresponds to 79.5 seconds.

The **Pascal3D** dataset [15] is a standard benchmark for pose estimation [17], [19]. For this dataset, we consider the four previously mentioned categories: cars, bikes, chairs and motorbikes. Following standard practice [17], [19] we only use non-truncated and non-occluded images from each category. We use the "train" set for training some of our baseline networks and for estimation of the global alignment transform $\mathcal{T}_G$ (see sec. A and 6.2 for details) and the held-out "val" set for evaluating performance of all the considered approaches.

For viewpoint estimation, Pascal3D already contains annotations. For LDOS, there are no such absolute viewpoint annotations. To generate ground truth annotations for evaluation, we manually aligned 3D reconstructions of 10 randomly-selected videos for each category and used 50 randomly-selected frames for each video as a test set.

For depth estimation, we evaluate on LDOS as it contains high quality depth maps which provide a suitable ground truth. We use the same 50 randomly selected frames from our pool of test videos, similar to the viewpoint estimation.

For point cloud reconstruction, we use FrC and LDOS. Ground truth point clouds for evaluation are obtained by merging the SFM or RGBD depth maps from all frames of a given test video sequence, picking $3 \cdot 10^4$ points using random subsampling and farthest point sampling for FrC and LDOS respectively. The point clouds were then post-processed with a 3D Laplacian filter. For FrC, five videos were randomly selected and removed from the train set, picking 60 random frames per video for evaluation. For LDOS the pose estimation test frames are used, i.e. the 50 frames extracted from the 10 test videos of each category.

6.2 Pose estimation

First, we evaluate the VpDR-Net viewpoint predictor on the Pascal3D benchmark [15]. Unlike previous works [17], [19] that focus on estimating the object/camera viewpoint represented by a 3 DoF rotation matrix, we evaluate the full 6 DoF camera pose represented by the rotation matrix R together with the translation vector T.

Adjusting the Pascal3D annotations. In Pascal3D, the camera poses are expressed relatively to the whole scenes instead of the objects themselves, so we adjust the dataset annotations. We crop every object using bounding box annotations after reshaping the box to a fixed aspect ratio, and resize the crop to 240×320 pixels. The camera pose is adjusted to the cropped object using the P3P algorithm to minimize the reprojection error between the camera-projected vertices of the ground truth CAD model and the original projection after cropping and resizing.

Absolute pose evaluation. We first evaluate absolute camera pose estimation using two standard measures: the angular error $e_R = 2^{-\frac{1}{2}} \| \ln R^* \hat{R}^\top \|_F$ between the ground truth camera pose R^* and the prediction $\hat{R}$, as well as the camera-center distance $e_C = \|\hat{C} - C^*\|_2$ between the predicted camera center $\hat{C}$ and the ground truth C^*. Following the common practice [17], [19] we report median e_R and e_C over all pose predictions on each test set.

Note that, while object viewpoints in Pascal3D and our method are internally consistent for a whole category, they may still differ between them by an arbitrary global 3D similarity transformation. Thus, the two sets of annotations are

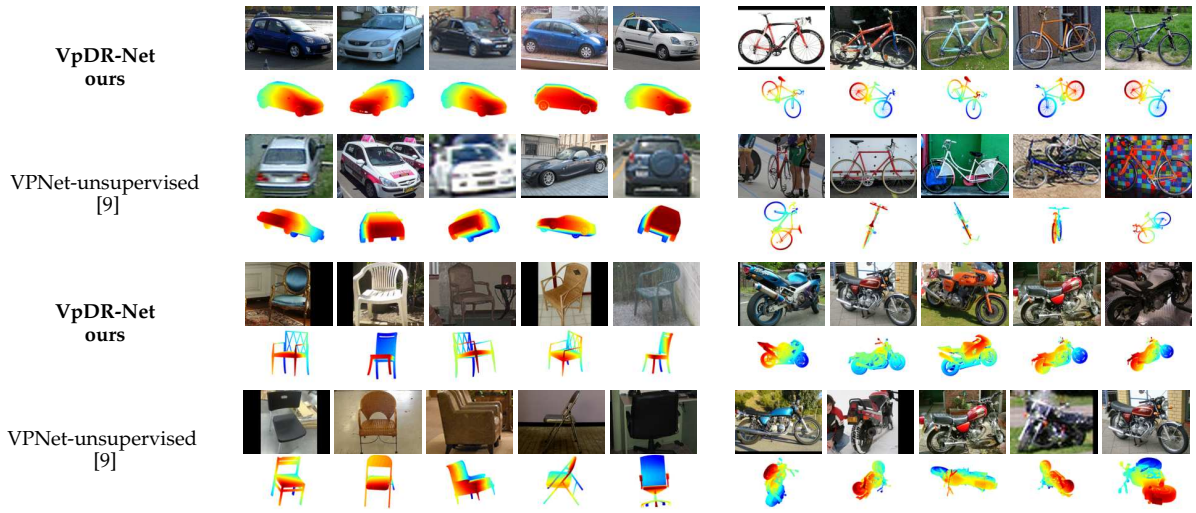


Fig. 8. **Viewpoint prediction.** Qualitative comparison between our VpDR-Net and the baseline VPNet architecture trained on Freiburg Cars / LDOS aligned with the method from [9] (VPNet-unsupervised). For each of the 4 considered object classes, the five most confident viewpoint predictions are visualized (sorted by the predicted confidence from left to right). Each predicted viewpoint is used to align the Pascal3D ground truth CAD model with the corresponding image.

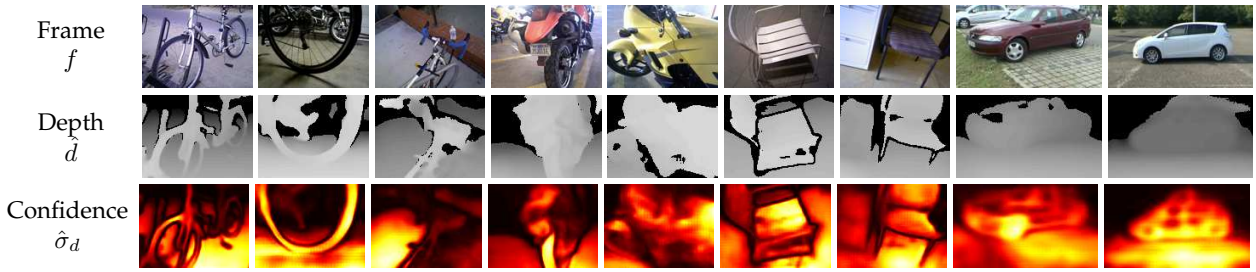


Fig. 9. **Monocular depth prediction.** Visualization of the predicted depth and confidence for different input images of the 4 considered classes. Depth maps are filtered by removing low confidence pixels. Lighter color corresponds to more confident regions.

	$\downarrow e_R$	$\downarrow e_C$	$\downarrow e_R^{rel}$	$\downarrow e_T^{rel}$	$\uparrow AP_{e_R}$	$\uparrow AP_{e_C}$
Test set: **LDOS** - chair						
VpDR-Net (ours)	**27.80**	**0.46**	**55.13**	**0.58**	**0.46**	**0.46**
VpDR-Net-NoAug	28.35	0.51	55.46	0.61	**0.46**	0.38
VpDR-Net-NoDepth	31.60	0.53	60.79	0.66	0.41	0.33
VpDR-Net-NoProb	68.74	0.83	85.78	0.89	0.05	0.06
Test set: **Pascal3D** - chair						
VpDR-Net (ours)	**33.70**	**14.23**	**57.61**	**0.72**	0.35	**0.34**
VpDR-Net-NoAug	33.96	15.26	67.12	0.79	**0.37**	0.29
VpDR-Net-NoDepth	35.34	18.78	68.68	0.85	0.30	0.14
VpDR-Net-NoProb	63.13	56.72	86.15	1.08	0.03	0.00

TABLE 2
Viewpoint prediction. Different flavors of VpDR-Net with removed components to evaluate their respective impact. All variations of VpDR-Net were trained on the LDOS videos of the chair class. ↑ (resp. ↓) means larger (resp. lower) is better.

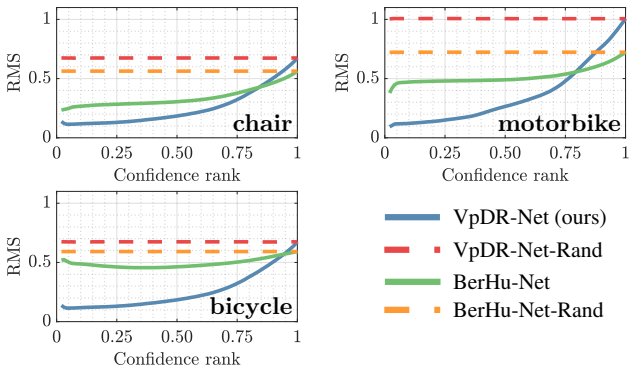


Fig. 10. **Monocular depth prediction.** Cumulative RMS depth reconstruction error for the LDOS data, when pixels are ranked by the predicted pixel-wise confidence.

aligned by a single global similarity $\mathcal{T}_G$ before assessment. The method for estimating $\mathcal{T}_G$ is detailed in sec. A.

Relative pose evaluation. To assess methods with measures independent of $\mathcal{T}_G$ we also evaluate: (1) the relative rotation error between pairs of ground truth relative camera motions $R^*_{tt'}$ and the corresponding predicted relative motions $\hat{R}_{tt'}$ given by $e_R^{\text{rel}} = 2^{-\frac{1}{2}} \| \ln R^*_{tt'} \hat{R}^\top_{tt'} \|_F$ and (2) the normalized relative translation error $e_T^{\text{rel}} = \|\hat{T}_{tt'} - T^*_{tt'}\|_2$, where both $\hat{T}_{tt'}$ and $T^*_{tt'}$ are ℓ_2-normalized so the measure is invariant to the scaling component of $\mathcal{T}_G$. We report the median errors over all possible image pairs in each test set.

Pose prediction confidence evaluation. A feature of our model is to produce confidence scores with its viewpoint estimates. We evaluate the reliability of these scores by correlating them with viewpoint prediction accuracy. In order to do so, predictions are divided into "accurate" and "inaccurate" by comparing their errors e_R and e_C to

thresholds (set to $e_R = \frac{\pi}{6}$ following [17], [19] and $e_C = 15$ and 0.5 for Pascal3D or LDOS respectively). Predictions are then ranked by decreasing confidence scores and the average precisions AP_{e_R} and AP_{e_C} of the two ranked lists are computed.

Baselines. We compare our viewpoint predictor to a strong baseline, called **VPNet**, trained using absolute viewpoint labels. VPNet is a ResNet50 architecture [64] with the final softmax classifier replaced by a viewpoint estimation layer that predicts the 6 DoF pose $\hat{g}_t^i$. Following [17], rotation matrices are decomposed in Euler angles, each discretized in 24 equal bins. This network is trained to predict a softmax distribution over the angular bins and to regress a 3D vector corresponding to the camera translation T. In order to attach a confidence measure to these predictions, empirically we found that it was beneficial to use the average softmax value across the three max-scoring Euler angles.

We test both an unsupervised and a fully-supervised variant of VPNet. **VPNet-unsupervised** is comparable to our setting and is trained on the output of the global camera poses estimated from the videos by the state-of-the-art sequence-alignment method of [9]. In the fully-supervised setting, VPNet is trained by using ground-truth global camera poses provided by the Pascal3D training set.

Quantitative results. Table 1 compares VpDR-Net to the VPNet baselines. First, we observe that our baseline VPNet-unsupervised is very strong, as we report $e_R = 49.6$ error for the full rotation matrix, while the original method of [9] reports an error of 61.5 just for the azimuth component. Nevertheless, VpDR-Net outperforms VPNet-unsupervised in all the cases. The most significant difference in performance can be observed for the motorbike and bicycle classes. Here, the primary reason for the performance drop of VPNet-unsupervised is the inability of the alignment method from [9] to cope with an absence of the ground plane which is the case for the bicycle and motorbike point clouds. This shows the advantage of the proposed viewpoint factorization method compared to aligning 3D shapes as in [9]. Furthermore, the unsupervised VpDR-Net significantly reduces the gap with fully-supervised VPNet. We also observe that the confidence scores estimated by VpDR-Net are significantly more correlated with the accuracy of the predictions than the softmax scores in VPNet-unsupervised, providing a reliable self-assessment mechanism. A qualitative comparison between VpDR-Net and the VPNet-unsupervised baseline are shown in Fig. 8.

Ablation study. We evaluate the importance of the different components of VpDR-Net by turning them off and measuring performance on the *chair* class. In Table 2, **VpDR-Net-NoProb** replaces the robust probabilistic losses $\mathcal{L}_R$ and $\mathcal{L}_T$ with their non-probabilistic counterparts ℓ_R and ℓ_T, and confidence predictions are replaced with random scores for AP evaluation. **VpDR-Net-NoDepth** removes the depth prediction and point cloud prediction branches during training, retaining only the Φ_{vp} subnetwork. **VpDR-Net-NoAug** does not use the data augmentation mechanism of sec. 5.1.

We observe a significant performance drop when each of the components is removed. This confirms the importance of all contributions in the network design.

Additional experiments. We conducted more comparisons to the state of the art on the unaltered Pascal3D dataset, reporting the Average Viewpoint Precision (AVP) measure on the validation set as in [15]. Since AVP requires an object detector, we use the same set of RCNN [72] detections as in [17]. In order to estimate $\mathcal{T}_G$, we use the ground truth annotations from the training set. Due to the additional noise brought by the global alignment $\mathcal{T}_G$ we report results for the coarsest level of 4 orientation bins.

Method	AVP (4 bins) per object class			
	chair	bicycle	mbike	car
VpDR-Net (ours)	**16.6**	23.9	**30.1**	**33.4**
VPNet-unsupervised [9]	12.7	**24.5**	19.8	29.4
3D-DPM [73]	6.1	43.9	31.8	36.9
Vps & Kps [17]	25.1	59.4	61.1	55.2

TABLE 3
Joint viewpoint prediction and object detection on Pascal3D reporting the AVP measure on the validation set. 3D-DPM and Vps & Kps are fully supervised approaches while our VpDR-Net and the VPNet-unsupervised baseline do not require manual annotations.

The results are summarized in Table 3. VpDR-Net outperforms VPNet-unsupervised on 3 out of 4 classes while being comparable to the fully supervised 3D-DPM [73] on 3 out of 4 classes as well.

6.3 Depth prediction

We evaluate the monocular depth prediction module of VpDR-Net and in particular its ability to self-predict the quality of its prediction. These experiments are conducted on the test set of LDOS, since FrC does not contain ground-truth depth annotations.

The depth prediction VpDR-Net is compared against three baselines: **VpDR-Net-Rand** uses VpDR-Net to estimate depth but predicts random confidence scores. **BerHu-Net** is a variant of the state-of-the-art depth prediction network from [30] based on the same Φ_{depth} subnetwork as VpDR-Net (but dropping Φ_{pcl} and Φ_{vp}). Following [30], for training it uses the BerHu depth loss and a dropout layer, which allows it to produce a confidence score of the depth measurements at test time using the sampling technique of [74]. Finally, **BerHu-Net-Rand** is the same network, but predicting random confidence scores.

Quantitative results. Results are presented in Fig. 10, for the three LDOS categories. This figure shows the cumulative root-mean-squared (RMS) depth reconstruction error, after sorting pixels by their confidence as estimated by the network. We observe that, by fitting better to inlier pixels and giving up on outliers, VpDR-Net produces a much better estimate than alternatives for the vast majority of pixels on all considered classes. Our confidence mechanism is more effective in the case of motorbike and bicycle classes which is probably caused by the lower reliability of the ground truth signal obtained by using the IR depth sensor in a suboptimal outdoor setting.

Qualitative results. Fig. 9 shows qualitative results. In the case of chair, motorbike and bicycle depth predictions, we can observe higher uncertainty on the metallic surfaces (e.g. bicycle frames and legs of chairs) or areas lying on the boundaries of the objects. This is expected since the depth sensor provides erroneous signal in these cases. Similarly for

	$\downarrow$ mD_{pcl}				$\uparrow$ mVIoU			
Object class **dataset**	chair LDOS	bicycle LDOS	mbike LDOS	car FrC	chair LDOS	bicycle LDOS	mbike LDOS	car FrC
Aubry et al. [41]	0.49	0.69	0.84	0.41	0.04	0.04	0.04	0.21
VpDR-Net-Fuse (ours)	0.25	0.28	0.37	**0.23**	**0.14**	**0.12**	**0.13**	**0.29**
VpDR-Net (ours)	0.25	0.32	0.40	**0.23**	**0.14**	**0.12**	**0.13**	**0.29**
VpDR-Net-$\hat{P}_f$	0.43	0.53	0.71	0.56	0.09	0.09	0.05	0.11
VpDR-Net-$\hat{S}$	0.39	1.23	0.44	0.70	0.11	0.09	0.11	0.16
VpDR-Net-Chamfer	**0.19**	**0.23**	**0.32**	0.24	0.10	0.07	0.08	0.20

TABLE 4
Point cloud prediction. Comparison between different variants of VpDR-Net and the method of Aubry et al. [41].

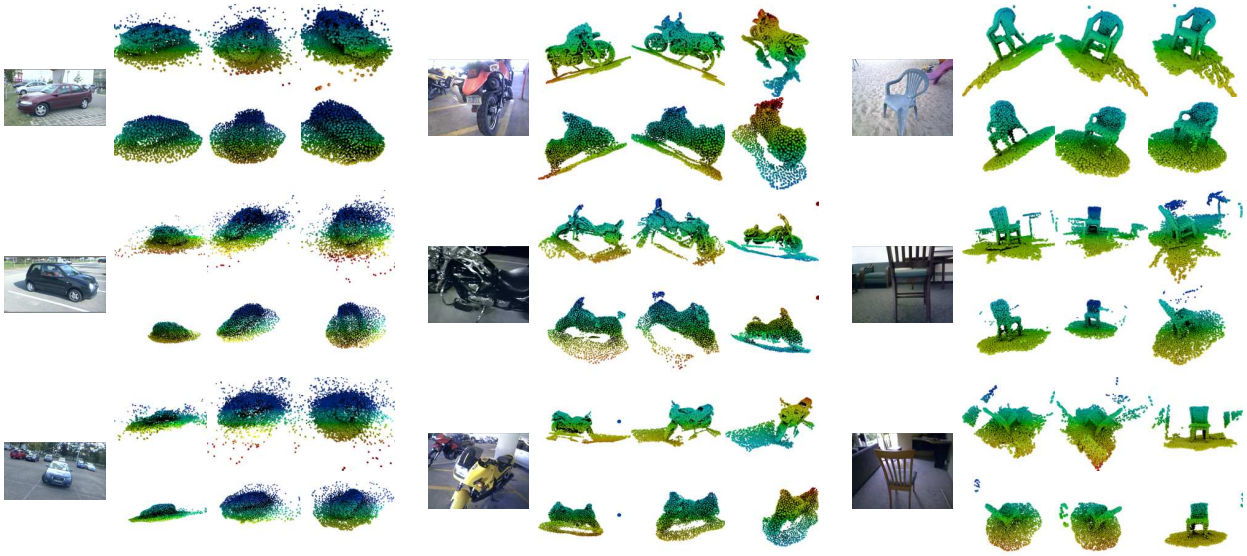


Fig. 11. **Point cloud prediction.** For every input image of an unseen object instance (left), ground-truth point cloud reconstruction using the full video sequence (top) and VpDR-Net point cloud prediction (bottom). For each reconstruction we show three different angles for better visualization.

the depth estimation of cars, the main modes of uncertainty can be observed on the specular areas (e.g. the bodywork) which is the case where ground truth providing multi-view stereo algorithm often fails.

6.4 Point cloud prediction

In this last set of experiments, we evaluate the point cloud completion module of VpDR-Net. The evaluation was conducted on the test sets of FrC and LDOS by comparing the predicted point clouds to the ground truth ones which were obtained as explained in sec. 6.1.

Evaluation measures. We use two evaluation measures: (1) the voxel intersection-over-union (VIoU) measure that computes the Jaccard similarity between the volumetric representations of $\hat{C}$ and C, and (2) the normalized point cloud distance of [75]. We average these measures over the test set leading to mVIoU and mD_{pcl}. The normalized point cloud distance of [75] is computed as

$$D_{\text{pcl}}(C,\hat{C}) = \frac{1}{|C|}\sum_{c\in C}\min_{\hat{c}\in\hat{C}}\|\hat{c}-c\| + \frac{1}{|\hat{C}|}\sum_{\hat{c}\in\hat{C}}\min_{c\in C}\|\hat{c}-c\|.$$

For the VIoU measure, a voxel grid is setup around each ground truth point-cloud C by uniformly subdividing C's bounding volume into 30^3 voxels.

The point clouds are compared within the local coordinate frames of each frame's camera (whose focal length is assumed to be known). Furthermore, since the SfM reconstructions are known only up to a global scaling factor, we adjust each point cloud prediction $\hat{C}$ from the FrC dataset by multiplying it with a scaling factor ζ that aligns the means of $\hat{C}$ and C. Note that ζ can be computed analytically with:

$$\zeta = \frac{\mu_C^T\mu_{\hat{C}}}{\mu_{\hat{C}}^T\mu_{\hat{C}}}, \text{where}$$

$\mu_C = \frac{1}{|C|}\sum_{c_m\in C} c_m$ is the centroid of the point cloud C.

Baselines. VpDR-Net is compared against the approach of Aubry et al. [41] using their code. [41] is a 3D CAD model retrieval method which first trains a large number of exemplar models which, in our case, are represented by individual video frames with their ground truth 3D point clouds. Then, given a testing image, [41] detects the object instance and retrieves the best matching model from the database. We align the retrieved point cloud to the object location in the testing image using the P3P algorithm.

For our VpDR-Net, we evaluate several different flavors: the original VpDR-Net that predicts the point cloud $\hat{C}$, VpDR-Net-Fuse which further merges $\hat{C}$ with the predicted partial depth map point cloud $\hat{P}_f$, VpDR-Net-$\hat{P}_f$ which only predicts the partial point cloud $\hat{P}_f$, VpDR-Net-$\hat{S}$ that predicts the raw unfiltered and untruncated point cloud $\hat{S}$ and finally VpDR-Net-Chamfer which removes the density

predictions $\hat{\delta}$ and replaces $l_{pcl}(\hat{S})$ with a Chamfer distance loss as explained in [50].

Quantitative results. Table 4 shows that our reconstructions significantly outperform [41] on both metrics for both LDOS and FrC. Fusing the results with the original depth map produces a denser point cloud estimate and improves the results for some classes. The drops in performance by predicting solely the raw and partial point clouds $\hat{P}_f$ and $\hat{S}$ emphasize the importance of the point cloud completion and density prediction components respectively. The Chamfer distance loss brings marginal improvements in D_{pcl} but a significant decrease of VIoU due to the inability of the network to represent and discard outliers. Furthermore, as in [50], we have observed that the network tends to predict an average model of the object category with a limited amount of shape variation.

Qualitative results. Qualitative results are shown in Fig. 11. We can observe that in the case of chair and motorbike reconstructions, which are the classes with a large number of training videos and relatively clean ground truth point clouds, the reconstructions exhibit a large amount of details that allow to distinguish different geometric styles (e.g. an enduro vs a chopper). For the car reconstructions, where the number of training videos is lower and the ground truth point clouds are noisy due to erroneous SfM multi-view stereo depth, our model trades off statistical sensitivity for increased smoothness of the predictions.

7 Conclusion

In this work, we have considered the problem of predicting the 3D geometry of an object from a single image. We have demonstrated that motion cues can replace manual annotations and synthetic data for learning the geometry of object categories, and that the learned model successfully generalizes to new unseen instances, predicting the viewpoint, the depth and the shape of that new instance. Learning from motion cues is enabled by two innovations, a new image-based viewpoint factorization method and a new probabilistic shape representation, which we leveraged in a single neural network that simultaneously performs the three prediction tasks. As a third innovation, we have also demonstrated that allowing predictors to explicitly express uncertainty leads to significantly more robust learning. We validated our approach on four object categories demonstrating performance superior to existing approaches.

Acknowledgments. The authors gratefully acknowledge the support of NAVER LABS Europe and ERC 677195-IDIU.

References

[1] A. W. Fitzgibbon and A. Zisserman, "Automatic camera recovery for closed or open image sequences," in *Proc. ECCV*, 1998.

[2] B. D. Lucas, T. Kanade *et al.*, "An iterative image registration technique with an application to stereo vision," 1981.

[3] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building rome in a day," *Communications of the ACM*, vol. 54, pp. 105–112, 2011.

[4] A. X. Chang, T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "Shapenet: An information-rich 3d model repository," *CoRR*, vol. abs/1512.03012, 2015.

[5] Y. Xiang, W. Kim, W. Chen, J. Ji, C. Choy, H. Su, R. Mottaghi, L. Guibas, and S. Savarese, "Objectnet3d: A large scale database for 3d object recognition," in *Proc. ECCV*, 2016.

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, 2010.

[7] J. Carreira, S. Vicente, L. Agapito, and J. Batista, "Lifting object detection datasets into 3d," *PAMI*, vol. 38, no. 7, pp. 1342–1355, 2016.

[8] M. Sun, H. Su, S. Savarese, and L. Fei-Fei, "A multi-view probabilistic model for 3d object classes," in *Proc. CVPR*, 2009.

[9] N. Sedaghat and T. Brox, "Unsupervised generation of a viewpoint annotated car dataset from videos," in *Proc. ICCV*, 2015.

[10] D. Novotny, D. Larlus, and A. Vedaldi, "Learning 3d object categories by looking around them," in *Proc. ICCV*, 2017.

[11] S. Savarese and L. Fei-Fei, "3d generic object categorization, localization and pose estimation," in *Proc. ICCV*, 2007.

[12] M. Ozuysal, V. Lepetit, and P.Fua, "Pose estimation for category specific multiview object localization," in *Proc. CVPR*, 2009.

[13] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich, "Viewpoint-aware object detection and pose estimation," in *Proc. ICCV*, 2011.

[14] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Multi-view priors for learning detectors from sparse viewpoint data," in *Proc. ICLR*, 2014.

[15] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *WACV*, 2014.

[16] R. Mottaghi, Y. Xiang, and S. Savarese, "A coarse-to-fine model for 3d pose estimation and sub-category recognition," in *Proc. CVPR*, 2015.

[17] S. Tulsiani and J. Malik, "Viewpoints and keypoints," in *Proc. CVPR*, 2015.

[18] J. Liebelt, C. Schmid, and K. Schertler, "Viewpoint-Independent Object Class Detection using 3D Feature Maps," in *Proc. CVPR*, 2008.

[19] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views," in *Proc. ICCV*, 2015.

[20] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox, "Demon: Depth and motion network for learning monocular stereo," in *Proc. CVPR*, 2017.

[21] J. W. Tangelder and R. C. Veltkamp, "A survey of content based 3d shape retrieval methods," *Multimedia Tools Appl.*, vol. 39, no. 3, pp. 441–471, Sep. 2008.

[22] T. Shen, H. Li, and X. Huang, "Approximately global optimization for robust alignment of generalized shapes," *PAMI*, vol. 33, pp. 1116–1131, 2010.

[23] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proc. CVPR*, 2014.

[24] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. NIPS*, 2014.

[25] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *PAMI*, vol. 38, no. 10, pp. 2024–2039, 2016.

[26] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *PAMI*, vol. 31, no. 5, pp. 824–840, 2009.

[27] K. Karsch, C. Liu, and S. B. Kang, "Depth transfer: Depth extraction from video using non-parametric sampling," *PAMI*, vol. 36, no. 11, pp. 2144–2158, 2014.

[28] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs," in *Proc. CVPR*, 2015.

[29] Y. Cao, Z. Wu, and C. Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.

[30] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *3DV*, 2016.

[31] R. Garg, V. K. BG, G. Carneiro, and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *Proc. ECCV*, 2016.

[32] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. CVPR*, 2017.

[33] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. CVPR*, 2017.

[34] B. K. P. Horn, "Shape from shading." Cambridge, MA, USA: MIT Press, 1989, ch. Obtaining Shape from Shading Information, pp. 123–171.

[35] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *PAMI*, 2015.

[36] M. Prasad, A. Zisserman, and A. W. Fitzgibbon, "Single view reconstruction of curved surfaces," in *Proc. CVPR*, vol. 2, June 2006, pp. 1345–1354.

[37] E. Toppe, C. Nieuwenhuis, and D. Cremers, "Relative volume constraints for single view 3d reconstruction," in *Proc. CVPR*, 2013.

[38] L. G. Roberts, "Machine perception of three-dimensional solids," Ph.D. dissertation, Massachusetts Institute of Technology. Dept. of Electrical Engineering, 1963. [Online]. Available: http://www.packet.cc/files/mach-per-3D-solids.html

[39] D. G. Lowe, "Three-dimensional object recognition from single two-dimensional images," *Artif. Intell.*, vol. 31, no. 3, pp. 355–395, 1987.

[40] J. J. Lim, H. Pirsiavash, and A. Torralba, "Parsing ikea objects: Fine pose estimation," in *Proc. ICCV*, 2013.

[41] M. Aubry, D. Maturana, A. Efros, B. Russell, and J. Sivic, "Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models," in *Proc. CVPR*, 2014.

[42] S. Gupta, P. A. Arbeláez, R. B. Girshick, and J. Malik, "Aligning 3D models to RGB-D images of cluttered scenes," in *Proc. CVPR*, 2015.

[43] A. Bansal, B. Russell, and A. Gupta, "Marr Revisited: 2D-3D model alignment via surface normal prediction," in *Proc. CVPR*, 2016.

[44] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta, "Learning a predictable and generative vector representation for objects," in *Proc. ECCV*, 2016.

[45] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *Proc. ECCV*, 2016.

[46] G. Riegler, A. O. Ulusoy, and A. Geiger, "Octnet: Learning deep 3d representations at high resolutions," in *Proc. CVPR*, 2017.

[47] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, "Atlasnet: A papier-m\^ ach\'e approach to learning 3d surface generation," *Proc. CVPR*, 2018.

[48] C. Häne, S. Tulsiani, and J. Malik, "Hierarchical surface prediction for 3d object reconstruction," *3DV*, 2017.

[49] A. Sinha, A. Unmesh, Q. Huang, and K. Ramani, "Surfnet: Generating 3d shape surfaces using deep residual networks," in *Proc. CVPR*, 2017.

[50] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3d object reconstruction from a single image," in *Proc. CVPR*, 2017.

[51] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Interpretable unsupervised learning on 3d point clouds," *Proc. CVPR*, 2018.

[52] M. Lhuillier and L. Quan, "A quasi-dense approach to surface reconstruction from uncalibrated images," *PAMI*, vol. 27, no. 3, pp. 418–433, 2005.

[53] M. Crocco, C. Rubino, and A. Del Bue, "Structure from motion with objects," in *Proc. CVPR*, June 2016.

[54] S. Zhu, L. Zhang, and B. M. Smith, "Model evolution: An incremental approach to non-rigid structure from motion," in *Proc. CVPR*, 2010.

[55] A. Kar, S. Tulsiani, J. Carreira, and J. Malik, "Category-specific object reconstruction from a single image," in *Proc. CVPR*, 2015.

[56] D. J. Rezende, S. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess, "Unsupervised learning of 3d structure from images," in *Proc. NIPS*, 2016.

[57] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik, "Multi-view supervision for single-view reconstruction via differentiable ray consistency," in *Proc. CVPR*, 2017.

[58] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision," in *Proc. NIPS*, 2016.

[59] O. Wiles and A. Zisserman, "Silnet: Single-and multi-view reconstruction by learning from silhouettes," *Proc. BMVC*, 2017.

[60] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. CVPR*, 2016.

[61] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *Proc. ECCV*, 2016.

[62] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proc. ICRA*, 2011.

[63] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *RO*, vol. 33, no. 5, pp. 1255–1262, 2017.

[64] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016.

[65] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. CVPR*, 2015.

[66] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. CVPR*, 2017.

[67] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Multi-view 3d models from single images with a convolutional network," in *Proc. ECCV*, 2016.

[68] T. J. Kozubowski, K. Podgórski, and I. Rychlik, "Multivariate generalized laplace distribution and related random fields," *Journal of Multivariate Analysis*, vol. 113, pp. 59–72, 2013.

[69] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. CVPR*, 2016.

[70] Y. Y. Morvan, "Acquisition, compression and rendering of depth and texture for multi-view video," Ph.D. dissertation, Technische Universiteit Eindhoven, 2009.

[71] S. Choi, Q. Zhou, S. Miller, and V. Koltun, "A large dataset of object scans," *CoRR*, vol. abs/1602.02481, 2016.

[72] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. CVPR*, 2014.

[73] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Teaching 3d geometry to deformable part models," in *Proc. CVPR*, 2012.

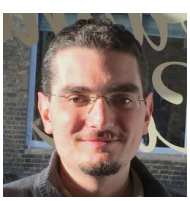
[74] Y. Gal and Z. Ghahramani, "Bayesian convolutional neural networks with Bernoulli approximate variational inference," in *Proc. ICLR*, 2016.

[75] J. Rock, T. Gupta, J. Thorsen, J. Gwak, D. Shin, and D. Hoiem, "Completing 3d object shape from one depth image," in *Proc. CVPR*, 2015.

[76] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *PAMI*, vol. 13, no. 4, pp. 376–380, 1991.

David Novotny received the MS degree (with honors) in computer vision and machine learning from the Czech Technical University, Prague in 2015. He is currently a DPhil student in the VGG group, University of Oxford in collaboration with Naver Labs Europe. His current research interests are object detection, representation learning, matching, single-view 3D reconstruction and pose estimation.

Diane Larlus is a senior research scientist at NAVER LABS Europe. From 2005 to 2008, she worked as a doctoral candidate at INRIA Grenoble (getting a PhD degree in computer science in 2008) and did a robotics internship at the JRL/AIST laboratory in Tsukuba in summer 2007. From 2008 to 2010, she worked as a postdoc at TU Darmstadt and joined what has now become NAVER LABS Europe in 2010.



Andrea Vedaldi is assistant professor of Engineering Science at the University of Oxford, where he is co-PI in the Visual Geometry Group since 2012. He obtained his PhD degree at the computer science department of the University of California at Los Angeles in 2008, and for the BSc in Information Engineering at the University of Padova in 2003. He is currently sponsored by an ERC Starting Grant, and EPSRC Programme Grant, and a number of industrial grants and collaborations.

APPENDIX A

In this section, we detail the procedure from sec. 6.2 that estimates the global alignment transformation $\mathcal{T}_G$. Given a set of ground truth camera poses $g_i^* = (R_i^*, T_i^*)$ and the corresponding predictions $\hat{g}_i = (\hat{R}_i, \hat{T}_i)$, we want to estimate a global similarity transform $\mathcal{T}_G = (R_G, T_G, s_G)$, parametrized by a scale $s_G \in \mathbb{R}$, translation $T_G \in \mathbb{R}^3$ and rotation $R_G \in SO(3)$, such that the coordinate frames of g_i^* and $\hat{g}_i$ become aligned.

In more detail, the desired global similarity transform satisfies the following equation:

$$\hat{R}_i(R_G X + T_G) + s_G \hat{T}_i = R_i^* X + T_i^* \ ; \ \forall X \qquad (13)$$

i.e. given an arbitrary world-coordinate point $X \in \mathbb{R}^3$, its projection into the coordinate frame of g_i^* (the right part of eq. (13)) should be equal to the projection of X into the coordinate frame of $\hat{g}_i$ after transforming X with R_G, T_G and scaling the corresponding camera translation vector $\hat{T}_i$ with s_G (the left side of eq. (13)). Note that for LDOS data $\mathcal{T}_G$ corresponds to a rigid motion and $s_G = 1$. Given $\mathcal{T}_G$, the adjusted camera matrices $\hat{g}_i^{\text{adjust}}$ for which $\hat{g}_i^{\text{adjust}} \approx g_i^*$ are then computed with

$$\hat{g}_i^{\text{adjust}} = (\hat{R}_i R_G, \ \hat{R}_i T_G + s_G \hat{T}_i).$$

In order to estimate $\mathcal{T}_G$, X is substituted in eq. (13) with $X = C_i^* = -{R_i^*}^T T_i^*$, i.e. X is set to be the center of the ground truth camera g_i^* which is a valid point of the world coordinate frame. After performing some additional manipulations, we end up with the following constraint:

$$\forall i: \quad \frac{1}{s_G} R_G C_i^* + \frac{1}{s_G} T_G = \hat{C}_i, \qquad (14)$$

where $\hat{C}_i = -\hat{R}_i^T \hat{T}_i$ is the center of the predicted camera $\hat{g}_i$. Given the corresponding camera pairs $\{(g_i^*, \hat{g}_i)\}_{i=1}^N$ the constraint in eq. (14) is converted to a least squares minimization problem:

$$\underset{R_G, T_G, s_G}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| \frac{1}{s_G} R_G C_i + \frac{1}{s_G} T_G - \hat{C}_i \right\|^2 \qquad (15)$$

and solved using the UMEYAMA algorithm [76].

For Pascal3D we estimate $\mathcal{T}_G$ from the held-out training set and later use it for evaluation on the test set. For LDOS, due to the absence of a held-out annotated training set, we estimate $\mathcal{T}_G$ on the test set.

9 Summary and future work

Within the chapters of this thesis we addressed different topics related to unsupervised learning of object categories from incomplete supervision. This final chapter presents conclusions of our work and suggests future research directions.

9.1 Seeking the limit of supervision

In chapter 3, we sought to find an amount of supervision that saturates the performance of a deep network. We did so on the task of object part detection since parts are often hypothesized to be visually shareable among object categories. Furthermore, learning object parts facilitates understanding of the structure of object classes which is one of the main goals of this thesis.

Our main observation was that deep networks possess strong generalization capabilities that allow them to learn successful part detectors by visiting a relatively limited number of examples. In more detail, the saturation point of the performance of a deep "eye" and "foot" keypoint detector was reached after observing roughly 20% and 30% of the annotated training examples respectively. This suggests that deep networks can solve different types of object understanding problems by being fed with an annotated dataset of a sufficient

and, crucially, finite size. Moreover, this finding validates the hypothesis that certain object parts are mostly defined by their appearance which motivates their usage for representing the structure of object categories without the need of manual annotations.

Chapter 3 also contributed with a dataset of object parts that was later used e.g. for analysis of salient regions that deep classifiers attend to [Fong and Vedaldi 2017].

Future work. An interesting extension would be seeking the limit of supervision for a different task where annotations are more expensive. For instance, 6-DoF pose estimation or 3D shape prediction from chapter 8. Another task that would tremendously benefit from a more efficient generation of annotations is the correspondence estimation. The existing learned solutions are either supervised by less reliable outputs of SfM systems [Yi et al. 2016] or by an optical flow obtained with expensive depth and ego-motion sensors [Choy et al. 2016a].

9.2 Webly-supervised part learning

Chapter 4 addresses learning of semantic object parts from noisy web supervision. In order to do so, a weakly-supervised object detector that leveraged a novel geometry-appearance embedding was proposed. The geometric part of the embedding was formed by a feature map attached to the intersection-over-union kernel which was, for the first time, proven to be positive-definite.

Another important contribution was the introduction of weakly supervised anchors. These were shown to be an excellent mid-level representation applicable for other tasks related to understanding the geometry of object categories. This was verified empirically on the semantic part matching and part-based image categorization tasks.

Future work. The proposed approach from chapter 4 was essentially a shallow classifier trained on top of a feature map derived from pre-trained deep features. An evident extension is to define an end-to-end trained architecture

that contains an intermediate layer of geometry-aware non-sematic features that supports an ensuing detector of semantic parts. More specifically, this is achievable by attaching D&D penalties from chapter 4 to a deep region detector such as Faster R-CNN [Ren et al. 2015].

Another drawback, that deserves rectification, is the dependence of the proposed pipeline on user-specified textual queries that denote the names of the learned object parts. This can be alleviated with a fully-automated solution that mines the names from e.g. WordNet [Chen et al. 2013; Miller 1995].

9.3 Learning geometry-aware representations

Inspired by the performance of the anchor-based detector from chapter 4, in chapter 5 we revised weakly-supervised learning of geometry-aware representations. Additional motivation was given by preceding related works [Long et al. 2014; Ham et al. 2016] that, on the semantic correspondence task, observed lower performance of deep features compared to hand-engineered alternatives.

Proposing a new deep architecture, termed AnchorNet, chapter 5 demonstrates that using its features in combination with off-the-shelf matching algorithms [Ham et al. 2016; Kim et al. 2013b] yields state-of-the-art performance on the semantic matching task and a novel cross-category semantic matching task. This showed the positive effects of the newly introduced discriminability and diversity optimization objective.

After the publication of [Novotny et al. 2017b], several other works that learn geometric features of object categories emerged. Differently from the AnchorNet, a significant portion of them proposed learning of embeddings equivariant with the geometric transformations of input images [Thewlis et al. 2017a; Thewlis et al. 2017b; Rocco et al. 2017]. Hence, in chapter 6 we shed light on learning geometric features using the equivariance constraint.

Firstly, we observed that existing formalizations [Thewlis et al. 2017a; Thewlis et al. 2017b] lack robustness to background clutter and self-occlusions. Seeking to address this drawback, the proposed probabilistic introspection framework

allowed to simultaneously learn a feature descriptor and detector. Such inclusion of the feature detector among the set of network outputs allowed to discard hard examples during training, achieving better specialization of the trained descriptors to important foreground patterns.

The benefits of probabilistic introspection were demonstrated in several ways. At the time of submission, state-of-the-art performance was obtained on the semantic matching task among methods that leverage similar amount of supervision. Compared to a leading fully-supervised approach [Han et al. 2017], our geometry-aware features performed on par. Futhermore, following [Thewlis et al. 2017a], we also quantitatively evaluated on a few-shot keypoint detection task. Similar to semantic matching, the proposed method yielded results superior to existing alternatives.

Interestingly, it has been observed that the activation patterns of the introduced representation were sparse and resembled keypoint localizers. This indicated that there is a strong relation between learning keypoint detectors and learning embeddings for matching.

The idea of learning unsupervised keypoint detectors from [Novotny et al. 2017a] was later re-visited for learning face landmarks in [Thewlis et al. 2017a; Thewlis et al. 2017b]. In general, the proposed AnchorNet architecture helped to increase the interest in the field of unsupervised learning of geometry-aware features [Wiles et al. 2018; Zhang et al. 2018; Jakab et al. 2018].

Future work. Following one of the principal applications of keypoints, it would be beneficial to explore the usage of the geometry-aware representation for category-specific 3D shape estimation. Existing pipelines either rely on synthetic datasets [Chang et al. 2015] or manually annotated keypoints [Kar et al. 2015a]. Such reconstruction of a category shape from a set of images, that could be inexpensively mined from e.g. a web search engine, would enable truly scalable 3D shape learning.

Regarding the methodology, an interesting alternative for unsupervised feature learning is multi-image matching [Zhou et al. 2015b; Zhou et al. 2015a;

Wang et al. 2017]. Unfortunately, the ample memory consumption of such methods prevents their use beyond collectively matching tens of features per image across a few thousands of images. Hence, it would be interesting to explore whether some of the intuitions proposed in chapters 5 and 6 can be mixed with the collective alignment framework in order to extend it to significantly larger datasets.

Finally, designing self-supervised features for other kinds of geometry-related tasks is another potential research direction. Probably the most appealing application would be the classic single-scene matching. Similar to probabilistic introspection, DeTone et al. [2017] recently introduced a novel feature description/detection system. Despite promising quantitative results, the architecture requires ground truth interest point locations. A successful adoption of the annotation-free probabilistic introspection can overcome the difficulty of acquiring dense annotations for the task.

9.4 Semi-convolutional operators

Similar to chapters 5 and 6, in chapter 7 we focused on learning deep architectures that output dense pixel-wise embeddings. Our goal was investigating their usage for detecting instances of object categories within images, effectively addressing the instance segmentation task.

Chapter 7 first compared two groups of existing approaches, *Propose & Verify* (P&Vs) and *Instance Coloring* (IC). It was theoretically demonstrated that existing IC approaches are unsuitable for the instance segmentation task mainly due to the translation invariance of the convolutional operators they rely on. A potential resolution of this problem, in form of novel translation equivariant semi-convolutional operators, was then proposed.

After enhancing existing instance segmentation architectures with the semi-convolutional component, state-of-the-art performance was attained on the Pascal VOC dataset and on a dataset of images of articulated organisms. An additional analysis conducted on synthetic data further clearly demonstrated the

benefits of extending convolutional operators to their semi-convolutional alternatives.

Future work. One of the biggest flaws of the proposed approach was its reliance on the Mask R-CNN detector which, to some extent, defeats the stated advantages of the IC methods. A desirable future goal is thus to remove the initial detector and propose a grouping algorithm that would directly convert the dense embedding field into a set of segmentation masks.

9.5 Learning 3D object categories by looking around them

Our final goal was understanding the 3D geometry of object categories. A human-inspired approach consisting of inferring a 3D structure of an object category by observing it in a set of videos was proposed in chapter 8. More in particular, VpDR-Net, a novel deep network for viewpoint, depth and point cloud prediction, was proposed.

The utilized dataset of videos was further enriched with additional supervisory data obtained using an off-the-shelf structure-from-motion method. In order to remove the inter-scene ambiguity from the SfM viewpoints, we designed a novel viewpoint factorization network that implicitly aligned individual reconstructions into a common coordinate frame. Furthermore, the network simultaneously learned a monocular pose predictor. Benchmarking this viewpoint branch on Pascal 3D, VpDR-Net attained results superior to the best previous comparable method [Sedaghat and Brox 2015].

The viewpoint branch and monocular depth predictor were later seamlessly integrated into a 3D shape prediction branch. Proposing a novel depth completion network that represented the output 3D shapes as a probability distribution supported by a flexible voxel grid, VpDR-Net outperformed other existing methods on the task of monocular 3D shape prediction.

At the time of publication of [Novotny et al. 2017c], VpDR-Net was the first unsupervised single-view 3D reconstruction approach fully trained and tested on real image data. The scientific community appreciated our work by selecting [Novotny et al. 2017c] among 12 best papers of ICCV 2017.

Future work. The related future research directions are twofold. First, due to SfM which rigidly reconstructs the training videos, our method is applicable only to rigid object categories. Hence, an extension that is capable of addressing the non-rigid category reconstruction problem would be a desirable contribution.

Second, it would be convenient to drop the SfM preprocessing step as it introduces another level of complexity. Here, combining the VpDR-Net with the video-supervised monocular architecture of Zhou et al. [2017] would be a meaningful first step.

9.6 Probabilistic learning

Our final contribution was designed to alleviate the difficulties arising from supervising algorithms in a weak manner or using noisy ground truth data. Specifically, a novel probabilistic learning framework for simultaneous uncertainty prediction and robust learning was proposed.

In chapter 6, this contribution takes form of a novel probabilistic introspection mechanism. At test time, it allows to predict a soft confidence map that serves as a detector of salient features. During training, the confidence map weights training annotations according to their difficulty. We have empirically demonstrated quantitative improvements over existing alternatives on the correspondence estimation task. Qualitatively, it was discovered that the confidence map conveniently segments out image regions corresponding to object categories.

In a similar spirit, chapter 8 proposed to explicitly model and identify outliers in noisy ground annotations coming from an off-the-shelf SfM algorithm.

This was done by altering standard regression architectures to output probability distributions supported by the continuous space of possible outputs. An ablation study where a probabilistic monocular pose estimator attained improved accuracy over its non-probabilistic variant demonstrated an increased robustness of the training process. The ability of a successful self-assessment was shown on a monocular depth prediction benchmark which, compared to previous methods, revealed better correlation between the predicted confidences and the actual ground-truth error.

The probabilistic framework was received well by the scientific community. In particular, follow-up applications of the proposed approach include deep SLAM [Bloesch et al. 2018], optical flow [Ilg et al. 2018], odometry [Wang et al. 2018] or video-supervised SfM [Klodt and Vedaldi 2018].

Future work. In chapter 6, it was qualitatively demonstrated that the confidence maps of the feature descriptors attend to instances of object categories. This observation suggests that the introduced probabilistic introspection learning paradigm can be exploited to address the semantic segmentation task. In the future, we can thus envision replacing the matching task from chapter 6 with a different auxiliary objective that leads to a similar segmentation effect.

Secondly, we would like to enhance the learning formulation. Currently, a downside of the probabilistic losses is their inability to distinguish between the errors caused by the incapability of the model to properly minimize the training loss, and the errors introduced due to the noise in ground truth annotations. Revising the formulation such that it disentangles the aforementioned error modes can then lead to an additional improvement in performance and robustness.

References

Agarwal, S., N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski [2009]. "Building Rome in a Day". In: *Proc. ICCV*.

Ali, K. and K. Saenko [2014]. "Confidence-Rated Multiple Instance Boosting for Object Detection". In: *Proc. CVPR*.

Andrews, S., I. Tsochantaridis, and T. Hofmann [2002]. "Support vector machines for multiple-instance learning". In: *Proc. NIPS*.

Arbelaez, P., J. Pont-Tuset, J. Barron, F. Marques, and J. Malik [2014]. "Multiscale Combinatorial Grouping". In: *Proc. CVPR*.

Aubry, M., D. Maturana, A. Efros, B. Russell, and J. Sivic [2014]. "Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models". In: *Proc. CVPR*.

Aubry, M. and B. C. Russell [2015]. "Understanding deep features with computer-generated imagery". In: *Proc. ICCV*.

Aytar, Y. and A. Zisserman [2012]. "Enhancing Exemplar SVMs using Part Level Transfer Regularization." In: *Proc. BMVC.*

Azizpour, H. and I. Laptev [2012]. "Object detection using strongly-supervised deformable part models". In: *Proc. ECCV*.

Azizpour, H., A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson [2015]. "From generic to specific deep representations for visual recognition". In: *Proc. CVPR*.

Bai, M. and R. Urtasun [2017]. "Deep Watershed Transform for Instance Segmentation". In: *Proc. CVPR*.

Baktashmotlagh, M., M. Harandi, B. Lovell, and M. Salzmann [2013]. "Unsupervised domain adaptation by domain invariant projection". In: *Proc. ICCV*.

Bansal, A., B. Russell, and A. Gupta [2016]. "Marr Revisited: 2D-3D Model Alignment via Surface Normal Prediction". In: *Proc. CVPR*.

Barnes, C., E. Shechtman, A. Finkelstein, and D. B. Goldman [2009]. "PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing". In:

Barron, J. T. and J. Malik [2015]. "Shape, Illumination, and Reflectance from Shading". In: *IEEE PAMI*.

Bay, H., A. Ess, T. Tuytelaars, and L. Van Gool [2008]. "Speeded-Up Robust Features (SURF)". In: *CVIU* 110.3, pp. 346–359.

Beardsley, P. A., A. Zisserman, and D. W. Murray [1997]. "Sequential updating of projective and affine structure from motion". In: *IJCV* 23.3, pp. 235–259.

Berg, T. L. and D. A. Forsyth [2006]. "Animals on the web". In: *Proc. CVPR*.

Bergamo, A. and L. Torresani [2010]. "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach". In: *Proc. NIPS*.

Beucher, S. [1979]. "Use of watersheds in contour detection". In: *Proceedings of the International Workshop on Image Processing*. CCETT.

Bilen, H., M. Pedersoli, and T. Tuytelaars [2014]. "Weakly supervised object detection with posterior regularization". In: *Proc. BMVC.*

Bilen, H. and A. Vedaldi [2015]. "Weakly Supervised Deep Detection Networks". In: *Proc. CVPR*.

Blanz, V. and T. Vetter [1999]. "A Morphable Model for the Synthesis of 3D Faces". In: *Proc. ACM SIGGRAPH*.

Blaschko, M., A. Vedaldi, and A. Zisserman [2010]. "Simultaneous object detection and ranking with weak supervision". In: *Proc. NIPS*, pp. 235–243.

Bloesch, M., J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison [2018]. "CodeSLAM-Learning a Compact, Optimisable Representation for Dense Visual SLAM". In: *Proc. CVPR*.

Bossard, L., M. Guillaumin, and L. Van Gool [2014]. "Food-101–mining discriminative components with random forests". In: *Proc. ECCV*.

Bourdev, L., S. Maji, T. Brox, and J. Malik [2010]. "Detecting people using mutually consistent poselet activations". In: *Proc. ECCV*. Springer.

Boykov, Y., O. Veksler, and R. Zabih [2001]. "Fast approximate energy minimization via graph cuts". In: *IEEE PAMI* 23.11, pp. 1222–1239.

Brachmann, E., A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother [2014]. "Learning 6d object pose estimation using 3d object coordinates". In: *Proc. ECCV*.

Bregler, C., A. Hertzmann, and H. Biermann [2000]. "Recovering Non-rigid 3D Shape from Image Streams". In: *Proc. CVPR*. Vol. 2, pp. 690–696.

Bristow, H., J. Valmadre, and S. Lucey [2015]. "Dense Semantic Correspondence Where Every Pixel is a Classifier". In: *Proc. ICCV*.

Brox, T., C. Bregler, and J. Malik [2009]. "Large displacement optical flow". In: *Proc. CVPR*, pp. 41–48.

Brox, T., A. Bruhn, N. Papenberg, and J. Weickert [2004]. "High Accuracy Optical Flow Estimation Based on a Theory for Warping". In: *Proc. ECCV*, pp. 25–36.

Cao, Y., Z. Wu, and C. Shen [2017]. "Estimating Depth from Monocular Images as Classification Using Deep Fully Convolutional Residual Networks". In: *IEEE Transactions on Circuits and Systems for Video Technology*.

Chai, Y., V. Lempitsky, and A. Zisserman [2013]. "Symbiotic segmentation and part localization for fine-grained categorization". In: *Proc. ICCV*.

Chang, A. X., T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu [2015]. "ShapeNet: An Information-Rich 3D Model Repository". In: *CoRR* abs/1512.03012.

Chen, X., R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille [2014a]. "Detect What You Can: Detecting and Representing Objects using Holistic Models and Body Parts". In: *Proc. CVPR*.

— [2014b]. "Detect what you can: Detecting and representing objects using holistic models and body parts". In: *Proc. CVPR*.

Chen, X. and A. Gupta [2015]. "Webly Supervised Learning of Convolutional Networks". In: *Proc. ICCV*.

Chen, X., A. Shrivastava, and A. Gupta [2013]. "NEIL: Extracting Visual Knowledge from Web Data". In: *Proc. ICCV*.

Chopra, S., S. Balakrishnan, and R. Gopalan [2013]. "Dlid: Deep learning for domain adaptation by interpolating between domains". In: *ICML workshop on challenges in representation learning*.

Choy, C. B., J. Gwak, S. Savarese, and M. Chandraker [2016a]. "Universal Correspondence Network". In: *Proc. NIPS*.

Choy, C. B., D. Xu, J. Gwak, K. Chen, and S. Savarese [2016b]. "3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction". In: *Proc. ECCV*.

Chum, O. and J. Matas [2005]. "Matching with PROSAC – Progressive Sample Consensus". In: *Proc. CVPR*.

Chum, O., J. Matas, and J. Kittler [2003]. "Locally optimized RANSAC". In: *DAGM 2003: Proceedings of the 25th DAGM Symposium*. Ed. by G. Goos, J. Hartmanis, and J. van Leeuwen. LNCS 2781. Springer-Verlag, pp. 236–243.

Cinbis, R. G., J. Verbeek, and C. Schmid [2014]. "Multi-fold MIL Training for Weakly Supervised Object Localization". In: *Proc. CVPR*.

— [2015]. "Weakly Supervised Object Localization with Multi-fold Multiple Instance Learning". In: *IEEE PAMI*.

Cohn, D., L. Atlas, and R. Ladner [1994]. "Improving generalization with active learning". In: *Machine learning* 15.2, pp. 201–221.

Collet, A., D. Berenson, S. S. Srinivasa, and D. Ferguson [2009]. "Object recognition and full pose registration from a single image for robotic manipulation". In: *Proc. Intl. Conf. on Robotics and Automation*.

Collet, A., M. Martinez, and S. S. Srinivasa [2011]. "The MOPED framework: Object recognition and pose estimation for manipulation". In: *Intl. J. of Robotics Research* 30.10, pp. 1284–1306.

Collins, B., J. Deng, K. Li, and L. Fei-Fei [2008]. "Towards scalable dataset construction: An active learning approach". In: *Proc. ECCV*.

Csurka, G. [2017]. *Domain adaptation in computer vision applications*. Springer.

Dai, J., K. He, and J. Sun [2016]. "Instance-aware semantic segmentation via multi-task network cascades". In: *Proc. CVPR*.

Dalal, N. and B. Triggs [2005a]. "Histograms of Oriented Gradients for Human Detection". In: *Proc. CVPR*.

— [2005b]. "Histograms of oriented gradients for human detection". In: *Proc. CVPR*.

Daume III, H. and D. Marcu [2006]. "Domain adaptation for statistical classifiers". In: *Journal of Artificial Intelligence Research*, pp. 101–126.

De Brabandere, B., D. Neven, and L. Van Gool [2017]. "Semantic instance segmentation with a discriminative loss function". In: *CoRR*.

Delage, E., H. Lee, and A. Y. Ng [2006]. "A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image". In: *Proc. CVPR*.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei [2009]. "ImageNet: A Large-Scale Hierarchical Image Database". In: *Proc. CVPR*.

Deselaers, T., B. Alexe, and V. Ferrari [2012a]. "Weakly Supervised Localization and Learning with Generic Knowledge". In: *Proc. ICCV*.

— [2012b]. "Weakly supervised localization and learning with generic knowledge". In: *IJCV* 100.3, pp. 275–293.

DeTone, D., T. Malisiewicz, and A. Rabinovich [2017]. "SuperPoint: Self-Supervised Interest Point Detection and Description". In: *CoRR*.

Diba, A., V. Sharma, A. M. Pazandeh, H. Pirsiavash, and L. Van Gool [2017]. "Weakly Supervised Cascaded Convolutional Networks." In: *Proc. CVPR*.

Dietterich, T. G., R. H. Lathrop, and T. Lozano-Pérez [1997]. "Solving the multiple instance problem with axis-parallel rectangles". In: *Artificial intelligence* 89.1, pp. 31–71.

Divvala, S., A. Farhadi, and C. Guestrin [2014]. "Learning everything about anything: Webly-supervised visual concept learning". In: *Proc. CVPR*.

Doersch, C., A. Gupta, and A. A. Efros [2014]. "Context as supervisory signal: Discovering objects with predictable context". In: *Proc. ECCV*.

Doersch, C., A. Gupta, and A. A. Efros [2013]. "Mid-Level Visual Element Discovery as Discriminative Mode Seeking". In: *Proc. NIPS*.

— [2015]. "Unsupervised Visual Representation Learning by Context Prediction". In: *Proc. ICCV*.

Dosovitskiy, A., P. Fischer, E. Ilg, P. Häusser, C. Hazirbas, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox [2015]. "FlowNet: Learning Optical Flow with Convolutional Networks". In: *Proc. ICCV*.

Dosovitskiy, A., J. T. Springenberg, M. Riedmiller, and T. Brox [2014]. "Discriminative unsupervised feature learning with convolutional neural networks". In: *Proc. NIPS*.

Duchenne, O., F. Bach, I.-S. Kweon, and J. Ponce [2011]. "A tensor-based algorithm for high-order graph matching". In: *IEEE PAMI* 33.12, pp. 2383–2395.

Efros, A. [2015]. *Visual Understanding without Naming: Bypassing the "Language Bottleneck"*. URL: `http://www.robots.ox.ac.uk/~seminars/seminars/Extra/2015_07_13_AlyoshaEfros.pdf`.

Eigen, D., C. Puhrsch, and R. Fergus [2014]. "Depth Map Prediction from a Single Image Using a Multi-scale Deep Network". In: *Proc. NIPS*.

Everingham, M., L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman [2011]. *The PASCAL Visual Object Classes Challenge 2011 (VOC2011)*. http://www.pascal-network.org/challenges/VOC/voc2011/.

Fan, H., H. Su, and L. Guibas [2017]. "A point set generation network for 3d object reconstruction from a single image". In: *Proc. CVPR*.

Fathi, A., Z. Wojna, V. Rathod, P. Wang, H. O. Song, S. Guadarrama, and K. P. Murphy [2017]. "Semantic Instance Segmentation via Deep Metric Learning". In: *CoRR* abs/1703.10277.

Faugeras, O. D. [1993]. *Three-Dimensional Computer Vision: a Geometric Viewpoint*. MIT Press.

Favaro, P. and S. Soatto [2002]. "Learning shape from defocus". In: *Proc. ECCV*, pp. 735–745.

Fei-Fei, L., R. Fergus, and P. Perona [2006]. "One-shot learning of object categories". In: *IEEE PAMI* 28.4, pp. 594–611.

Felzenszwalb, P. F., R. B. Girshick, D. McAllester, and D. Ramanan [2010a]. "Object Detection with Discriminatively Trained Part Based Models". In: *IEEE PAMI* 32.9, pp. 1627–1645.

Felzenszwalb, P. F., R. Grishick, D. McAllester, and D. Ramanan [2010b]. "Object Detection with Discriminatively Trained Part Based Models". In: *IEEE PAMI*.

Felzenszwalb, P., D. Mcallester, and D. Ramanan [2008a]. "A Discriminatively Trained, Multiscale, Deformable Part Model". In: *Proc. CVPR*.

Felzenszwalb, P. F. and D. P. Huttenlocher [2003]. "Pictorial Structures for Object Recognition". In: *IJCV* 61, p. 2005.

Felzenszwalb, P., D. McAllester, and D. Ramanan [2008b]. "A discriminatively trained, multiscale, deformable part model". In: *Proc. CVPR*.

Fergus, R. [2005]. "Visual Object Category Recognition". PhD thesis. University of Oxford.

Fergus, R., P. Perona, and A. Zisserman [2003]. "Object Class Recognition by Unsupervised Scale-Invariant Learning". In: *Proc. CVPR*. Vol. 2, pp. 264–271.

Fergus, R., Y. Weiss, and A. Torralba [2009]. "Semi-supervised learning in gigantic image collections". In: *Proc. NIPS*.

Fergus, R., L. Fei-Fei, P. Perona, and A. Zisserman [2005]. "Learning object categories from Google's image search". In: *Proc. ICCV*.

Fergus, R., P. Perona, and A. Zisserman [2004]. "A visual category filter for google images". In: *Proc. ECCV*.

Fernando, B., H. Bilen, E. Gavves, and S. Gould [2017]. "Self-Supervised Video Representation Learning With Odd-One-Out Networks". In: *Proc. ICCV*.

Fernando, B., A. Habrard, M. Sebban, and T. Tuytelaars [2013]. "Unsupervised visual domain adaptation using subspace alignment". In: *Proc. ICCV*.

Fidler, S., S. Dickinson, and R. Urtasun [2012]. "3d object detection and viewpoint estimation with a deformable 3d cuboid model". In: *Proc. NIPS*.

Fischer, P., A. Dosovitskiy, and T. Brox [2014]. "Descriptor matching with convolutional neural networks: a comparison to sift". In: *CoRR*.

Fischler, M. A. and R. C. Bolles [1981]. "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography". In: *Comm. ACM* 24.6, pp. 381–395.

Fitzgibbon, A. W. and A. Zisserman [1998]. "Automatic Camera Recovery for Closed or Open Image Sequences". In: *Proc. ECCV*.

Fong, R. and A. Vedaldi [2017]. "Interpretable Explanations of Black Boxes by Meaningful Perturbation". In: *Proc. ICCV*.

Frahm, J.-M., P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, et al. [2010]. "Building rome on a cloudless day". In: *Proc. ECCV*.

Ganin, Y. and V. Lempitsky [2015]. "Unsupervised domain adaptation by backpropagation". In: *Proc. ICML*.

Garg, R., V. K. BG, G. Carneiro, and I. Reid [2016]. "Unsupervised cnn for single view depth estimation: Geometry to the rescue". In: *Proc. ECCV*.

Girdhar, R., D. F. Fouhey, M. Rodriguez, and A. Gupta [2016]. "Learning a Predictable and Generative Vector Representation for Objects". In: *Proc. ECCV*.

Girshick, R. B. [2015]. "Fast R-CNN". In: *Proc. ICCV*.

Girshick, R. B., J. Donahue, T. Darrell, and J. Malik [2014a]. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proc. CVPR*.

Girshick, R., J. Donahue, T. Darrell, and J. Malik [2014b]. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proc. CVPR*.

— [2014c]. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proc. CVPR*.

Gkioxari, G., R. Girshick, and J. Malik [2015]. "Actions and attributes from wholes and parts". In: *Proc. ICCV*.

Godard, C., O. Mac Aodha, and G. J. Brostow [2017]. "Unsupervised monocular depth estimation with left-right consistency". In: *Proc. CVPR*.

Gong, B., Y. Shi, F. Sha, and K. Grauman [2012]. "Geodesic flow kernel for unsupervised domain adaptation". In: *Proc. CVPR*.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio [2014]. "Generative adversarial nets". In: *Proc. NIPS*, pp. 2672–2680.

Gopalan, R., R. Li, and R. Chellappa [2011]. "Domain adaptation for object recognition: An unsupervised approach". In: *Proc. ICCV*.

Gordon, I. and D. G. Lowe [2006]. "What and where: 3D object recognition with accurate pose". In: *Toward category-level object recognition*. Springer, pp. 67–82.

Groueix, T., M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry [2018]. "AtlasNet: A Papier-M\^ ach\'e Approach to Learning 3D Surface Generation". In: *Proc. CVPR*.

Ham, B., M. Cho, C. Schmid, and J. Ponce [2016]. "Proposal Flow". In: *Proc. CVPR*.

Han, K., R. S. Rezende, B. Ham, K.-Y. K. Wong, M. Cho, and J. P. Cordelia Schmid [2017]. "SCNet: Learning Semantic Correspondence". In: *Proc. ICCV*.

Han, X., T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg [2015]. "Matchnet: Unifying feature and metric learning for patch-based matching". In: *Proc. CVPR*.

Häne, C., S. Tulsiani, and J. Malik [2017]. "Hierarchical surface prediction for 3d object reconstruction". In: *Proc. 3DV*.

Hariharan, B., P. Arbeláez, R. Girshick, and J. Malik [2014]. "Simultaneous detection and segmentation". In: *Proc. ECCV*.

— [2015]. "Hypercolumns for object segmentation and fine-grained localization". In: *Proc. CVPR*.

Hariharan, B., J. Malik, and D. Ramanan [2012]. "Discriminative Decorrelation for Clustering and Classification". In: *Proc. ECCV*.

Hartley, R. I. [1993]. "Euclidean reconstruction from uncalibrated views". In: *Proc. 2nd European-US Workshop on Invariance, Azores*. Ed. by J. L. Mundy, A. Zisserman, and D. Forsyth, pp. 187–202.

Hassner, T. [2013]. "Viewing real-world faces in 3D". In: *Proc. ICCV*.

Hassner, T. and R. Basri [2006]. "Example based 3D reconstruction from single 2D images". In: *Proc. CVPR Workshops*.

Hayder, Z., X. He, and M. Salzmann [2017]. "Boundary-aware instance segmentation". In: *Proc. CVPR*.

He, K., G. Gkioxari, P. Dollár, and R. Girshick [2017]. "Mask r-cnn". In: *Proc. ICCV*.

He, K., X. Zhang, S. Ren, and J. Sun [2016a]. "Deep Residual Learning for Image Recognition". In: *Proc. CVPR*.

— [2016b]. "Deep residual learning for image recognition". In: *Proc. CVPR*.

Heinly, J., J. L. Schonberger, E. Dunn, and J.-M. Frahm [2015]. "Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset)". In: *Proc. CVPR*.

Hoffman, J., S. Guadarrama, E. S. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko [2014]. "LSDA: Large Scale Detection through Adaptation". In: *Proc. NIPS*. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger.

Hoffman, J., D. Pathak, T. Darrell, and K. Saenko [2015]. "Detector Discovery in the Wild: Joint Multiple Instance and Representation Learning". In: *Proc. CVPR*.

Hoffman, J., E. Rodner, J. Donahue, T. Darrell, and K. Saenko [2013]. "Efficient learning of domain-invariant image representations". In: *Proc. ICLR*.

Hoiem, D., A. A. Efros, and M. Hebert [2005]. "Automatic Photo Pop-up". In: *Proc. ACM SIGGRAPH*.

Holub, A., P. Perona, and M. C. Burl [2008]. "Entropy-based active learning for object recognition". In: *Proc. CVPR Workshops*.

Horn, B. K. [1970]. *Shape from shading: A method for obtaining the shape of a smooth opaque object from one view*. Tech. rep.

Horn, B. K. P. and B. G. Schunck [1993]. "Determining optical flow: A Retrospective." In: *Artif. Intell.* 1-2.

Huang, Q.-X. and L. Guibas [2013]. "Consistent shape maps via semidefinite programming". In: *Eurographics Symposium on Geometry Processing* 32.5.

Huang, Q., H. Wang, and V. Koltun [2015]. "Single-view reconstruction via joint analysis of image and shape collections". In: 34.4, p. 87.

Hur, J., H. Lim, C. Park, and S. C. Ahn [2015]. "Generalized Deformable Spatial Pyramid: Geometry-preserving dense correspondence estimation". In: *Proc. CVPR*.

Ilg, E., O. Ciçek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox [2018]. "Uncertainty Estimates and Multi-Hypotheses Networks for Optical Flow". In: *Proc. ECCV*.

Ilg, E., N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox [2017]. "Flownet 2.0: Evolution of optical flow estimation with deep networks". In: *Proc. CVPR*.

Izadinia, H., F. Sadeghi, S. K. Divvala, H. Hajishirzi, Y. Choi, and A. Farhadi [2015]. "Segment-Phrase Table for Semantic Segmentation, Visual Entailment and Paraphrasing". In: *Proc. ICML*.

Jakab, T., A. Gupta, H. Bilen, and A. Vedaldi [2018]. "Conditional Image Generation for Learning the Structure of Visual Objects". In: *Proc. NIPS*.

Jayaraman, D. and K. Grauman [2015]. "Learning image representations tied to ego-motion". In: *Proc. ICCV*.

Joshi, A. J., F. Porikli, and N. Papanikolopoulos [2009]. "Multi-class active learning for image classification". In: *Proc. CVPR*.

Joulin, A., F. Bach, and J. Ponce [2010]. "Efficient Optimization for Discriminative Latent Class Models". In: *Proc. NIPS*.

— [2012]. "Multi-Class Cosegmentation". In: *Proc. CVPR*.

Joulin, A., K. Tang, and L. Fei-Fei [2014]. "Efficient Image and Video Co-localization with Frank-Wolfe Algorithm". In: *Proc. ECCV*.

Juneja, M., A. Vedaldi, C. V. Jawahar, and A. Zisserman [2013]. "Blocks that Shout: Distinctive Parts for Scene Classification". In: *Proc. CVPR*.

Kanazawa, A., D. W. Jacobs, and M. Chandraker [2016]. "WarpNet: Weakly Supervised Matching for Single-View Reconstruction". In: *Proc. CVPR*.

Kanazawa, A., S. Tulsiani, A. A. Efros, and J. Malik [2018]. "Learning Category-Specific Mesh Reconstruction from Image Collections". In: *Proc. ECCV*.

Kapoor, A., K. Grauman, R. Urtasun, and T. Darrell [2007]. "Active learning with gaussian processes for object categorization". In: *Proc. ICCV*.

Kar, A., S. Tulsiani, J. Carreira, and J. Malik [2015a]. "Category-specific object reconstruction from a single image". In: *Proc. CVPR*.

Kar, A., S. Tulsiani, J. Carreira, and J. Malik [2015b]. "Category-specific object reconstruction from a single image". In: *Proc. CVPR*.

Karsch, K., C. Liu, and S. B. Kang [2014]. "Depth Transfer: Depth Extraction from Video Using Non-Parametric Sampling". In: *IEEE PAMI* 36.11, pp. 2144–2158.

Kemelmacher-Shlizerman, I. and R. Basri [2011]. "3D face reconstruction from a single image using a single reference face shape". In: *IEEE PAMI* 33.2, pp. 394–405.

Kemelmacher-Shlizerman, I. and S. M. Seitz [2012]. "Collection flow". In: *Proc. CVPR*.

Kim, G. and E. P. Xing [2012]. "On Multiple Foreground Cosegmentation". In: *Proc. CVPR*.

Kim, J., C. Liu, F. Sha, and K. Grauman [2013a]. "Deformable Spatial Pyramid Matching for Fast Dense Correspondences". In: *Proc. CVPR*.

— [2013b]. "Deformable Spatial Pyramid Matching for Fast Dense Correspondences". In: *Proc. CVPR*.

Kim, S., D. Min, B. Ham, S. Jeon, S. Lin, and K. Sohn [2017]. "Fcss: Fully convolutional self-similarity for dense semantic correspondence". In: *Proc. CVPR*.

Kivinen, J. J. and C. K. Williams [2011]. "Transformation equivariant boltzmann machines". In: *Proc. of the International Conference on Artificial Neural Networks*.

Klodt, M. and A. Vedaldi [2018]. "Supervising the new with the old: learning SFM from SFM". In: *Proc. ECCV*, pp. 698–713.

Koch, R., M. Pollefeys, L. Van Gool, B. Heigl, and H. Niemann [1999]. "Calibration of hand-held camera sequences for plenoptic modeling". In: *Proc. ICCV*.

Koller, D., K. Daniilidis, and H. H. Nagel [1993]. "Model-Based Object Tracking in Monocular Sequences of Road Traffic Scenes". In: *IJCV* 10, pp. 257–281.

Koltun, V. [2011]. "Efficient inference in fully connected crfs with gaussian edge potentials". In: *Proc. NIPS*.

Kong, S. and C. Fowlkes [2018]. "Recurrent Pixel Embedding for Instance Grouping". In: *Proc. CVPR*.

Krizhevsky, A., I. Sutskever, and G. E. Hinton [2012a]. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Proc. NIPS*, pp. 1106–1114.

— [2012b]. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Proc. NIPS*.

Kruppa, E. [1913]. *To determine an object from two perspectives with inner orientation*. Holder.

Kushal, A., C. Schmid, and J. Ponce [2007]. "Flexible object models for category-level 3d object recognition". In: *Proc. CVPR*.

Ladický, L., P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr [2010]. "What, Where and How Many? Combining Object Detectors and CRFs". In: *Proc. ECCV*.

Ladicky, L., J. Shi, and M. Pollefeys [2014]. "Pulling Things out of Perspective". In: *Proc. CVPR*.

Laina, I., C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab [2016]. "Deeper depth prediction with fully convolutional residual networks". In: *Proc. 3DV*.

Lampert, C. H., H. Nickisch, and S. Harmeling [2009]. "Learning to detect unseen object classes by between-class attribute transfer". In: *Proc. CVPR*.

Learned-Miller, E. G. [2006]. "Data driven image models through continuous joint alignment". In: *IEEE PAMI* 28.2, pp. 236–250.

Lee, H., R. Grosse, R. Ranganath, and A. Y. Ng [2009]. "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, pp. 609–616.

Leibe, B., A. Leonardis, and B. Schiele [2004]. "Combined Object Categorization and Segmentation with an Implicit Shape Model". In: *Workshop on Statistical Learning in Computer Vision, ECCV*.

Lenc, K. and A. Vedaldi [2016]. "Learning Covariant Feature Detectors". In: *ECCV Workshop on Geometry Meets Deep Learning*.

Leordeanu, M. and M. Hebert [2005]. "A Spectral Technique for Correspondence Problems using Pairwise Constraints". In: *Proc. ICCV*. Vol. 2, pp. 1482–1489.

Li, B., C. Shen, Y. Dai, A. van den Hengel, and M. He [2015a]. "Depth and Surface Normal Estimation From Monocular Images Using Regression on Deep Features and Hierarchical CRFs". In: *Proc. CVPR*.

Li, L.-J., G. Wang, and L. Fei-Fei [2007]. "OPTIMOL: automatic Online Picture collecTion via Incremental MOdel Learning". In: *Proc. CVPR*.

Li, Q., J. Wu, and Z. Tu [2013]. "Harvesting Mid-level Visual Concepts from Large-Scale Internet Images". In: *Proc. CVPR*.

Li, Y., L. Liu, C. Shen, and A. van den Hengel [2015b]. "Mid-level deep pattern mining". In: *Proc. CVPR*.

Liang, X., Y. Wei, X. Shen, Z. Jie, J. Feng, L. Lin, and S. Yan [2016]. "Reversible recursive instance-level object segmentation". In: *Proc. CVPR*.

Lim, J. J., H. Pirsiavash, and A. Torralba [2013]. "Parsing IKEA Objects: Fine Pose Estimation". In: *Proc. ICCV*.

Lin, T.-Y., M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick [2014]. "Microsoft coco: Common objects in context". In: *Proc. ECCV*.

Liu, C., J. Yuen, and A. Torralba [2011]. "SIFT Flow: Dense Correspondence across Scenes and Its Applications". In: *IEEE PAMI* 33.5, pp. 978–994.

Long, J., E. Shelhamer, and T. Darrell [2015]. "Fully convolutional networks for semantic segmentation". In: *Proc. CVPR*.

Long, J., N. Zhang, and T. Darrell [2014]. "Do Convnets Learn Correspondence?" In: *Proc. NIPS*.

Longuet-Higgins, H. C. [1981]. "A computer algorithm for reconstructing a scene from two projections". In: *Nature* 293, pp. 133–135.

Lowe, D. G. [1987]. "Three-dimensional Object Recognition from Single Two-dimensional Images". In: *Artif. Intell.* 31.3, pp. 355–395.

Lowe, D. [1999]. "Object recognition from local scale-invariant features". In: *Proc. ICCV*, pp. 1150–1157.

— [2004a]. "Distinctive Image Features from Scale-Invariant Keypoints". In: *IJCV* 60.2, pp. 91–110.

Lowe, D. G. [2004b]. "Distinctive image features from scale-invariant keypoints". In: *IJCV* 60.2, pp. 91–110.

Lu, C. and X. Tang [2015]. "Surpassing Human-Level Face Verification Performance on LFW with GaussianFace." In: *Proc. AAAI*.

Lucas, B. D. and T. Kanade [1981]. "An iterative Image Registration Technique with an Application to Stereo Vision". In: *Proc. of the 7th International Joint Conference on Artificial Intelligence*, pp. 674–679. URL: `citeseer.nj.nec.com/lucas81optical.html`.

Maciel, J. and J. P. Costeira [2003]. "A global solution to sparse correspondence problems". In: *IEEE PAMI* 25.2, pp. 187–199.

Maron, O. and T. Lozano-Pérez [1998]. "A framework for multiple-instance learning". In: *Proc. NIPS*.

Maron, O. and A. L. Ratan [1998]. "Multiple-Instance Learning for Natural Scene Classification." In: *Proc. ICML*.

Massa, F., B. C. Russell, and M. Aubry [2016]. "Deep exemplar 2d-3d detection by adapting from real to rendered views". In: *Proc. ICCV*.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean [2013]. "Distributed representations of words and phrases and their compositionality". In: *Proc. NIPS*.

Miller, G. A. [1995]. "WordNet: A Lexical Database for English". In: *Commun. ACM* 38, pp. 39–41.

Mishchuk, A., D. Mishkin, F. Radenovic, and J. Matas [2017]. "Working hard to know your neighbor's margins: Local descriptor learning loss". In: *Proc. NIPS*.

Misra, I., C. L. Zitnick, and M. Hebert [2016]. "Shuffle and Learn: Unsupervised Learning using Temporal Order Verification". In: *Proc. ECCV*.

Modolo, D. and V. Ferrari [2018]. "Learning Semantic Part-Based Models from Google Images". In: *IEEE PAMI* 40.6, pp. 1502–1509.

Nagai, T., T. Naruse, M. Ikehara, and A. Kurematsu [2002]. "HMM-based surface reconstruction from single images". In: *Intl. Conf. Image Proc.*

Nandy, D. and J. Ben-Arie [2000]. "Recovery of 3-D face structure using recognition". In: *Proc. ICPR*.

Nesterov, Y. [2005]. "Smooth minimization of non-smooth functions". In: *Mathematical programming* 103.1, pp. 127–152.

Nevatia, R. and T. O. Binford [1977]. "Description and recognition of curved objects". In: *Artificial Intelligence* 8.1, pp. 77–98.

Newell, A., Z. Huang, and J. Deng [2017]. "Associative Embedding: End-to-end Learning for Joint Detection and Grouping". In: *Proc. NIPS*.

Nguyen, M. H., L. Torresani, F. de la Torre, and C. Rother [2009]. "Weakly supervised discriminative localization and classification: a joint learning process". In: *Proc. ICCV*.

Ni, J., Q. Qiu, and R. Chellappa [2013]. "Subspace interpolation via dictionary learning for unsupervised domain adaptation". In: *Proc. CVPR*.

Nistér, D. [2003]. "Preemptive RANSAC for Live Structure and Motion Estimation". In: *Proc. ICCV*, pp. 199–206.

Noroozi, M. and P. Favaro [2016]. "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles". In: *Proc. ECCV*.

Norouzi, M., M. Ranjbar, and G. Mori [2009]. "Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning". In: *Proc. CVPR*.

Novotny, D., S. Albanie, D. Larlus, and A. Vedaldi [2018a]. "Self-supervised Learning of Geometrically Stable Features Through Probabilistic Introspection". In: *Proc. CVPR*.

— [2018b]. "Semi-convolutional Operators for Instance Segmentation". In: *Proc. ECCV*.

Novotny, D., D. Larlus, and A. Vedaldi [2016a]. "I Have Seen Enough: Transferring Parts Across Categories". In: *Proc. BMVC.*

— [2016b]. "Learning the semantic structure of objects from Web supervision". In: *Proc. ECCV Workshops*.

— [2017a]. "AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching". In: *Proc. CVPR*.

— [2017b]. "AnchorNet: A Weakly Supervised Network to Learn Geometry-sensitive Features For Semantic Matching". In: *Proc. CVPR*.

— [2017c]. "Learning 3D Object Categories by Looking Around Them". In: *Proc. ICCV*.

— [2018c]. "Capturing the Geometry of Object Categories from Video Supervision". In: *IEEE PAMI*.

Novotny, D., D. Larlus, and A. Vedaldi [2017d]. "Generalizing Semantic Part Detectors Across Domains". In: *Domain Adaptation in Computer Vision Applications*. Springer, pp. 259–273.

Oquab, M., L. Bottou, I. Laptev, and J. Sivic [2015]. "Is object localization for free?-weakly-supervised learning with convolutional neural networks". In: *Proc. CVPR*.

Pachauri, D., R. Kondor, and V. Singh [2013]. "Solving the multi-way matching problem by permutation synchronization". In: *Proc. NIPS*.

Pandey, M. and S. Lazebnik [2011]. "Scene Recognition and Weakly Supervised Object Localization with Deformable Part-based Models". In: *Proc. ICCV*.

Papageorgiou, C. P., M. Oren, and T. Poggio [1998]. "A general framework for object detection". In: *Proc. ICCV*.

Parikh, D. and K. Grauman [2011]. "Relative attributes". In: *Proc. ICCV*.

Parkash, A. and D. Parikh [2012]. "Attributes for classifier feedback". In: *Proc. ECCV*.

Parkhi, O. M., A. Vedaldi, C. Jawahar, and A. Zisserman [2011]. "The truth about cats and dogs". In: *Proc. ICCV*.

Pathak, D., P. Krähenbühl, J. Donahue, T. Darrell, and A. Efros [2016]. "Context Encoders: Feature Learning by Inpainting". In: *Proc. CVPR*.

Pavlakos, G., X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis [2017]. "6-dof object pose from semantic keypoints". In: *Proc. Intl. Conf. on Robotics and Automation*.

Peng, Y., A. Ganesh, J. Wright, W. Xu, and Y. Ma [2012]. "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images". In: *IEEE PAMI* 34.11, pp. 2233–2246.

Pentland, A. P. [1986]. "Perceptual Organisation and the Representation of Natural Form". In: *Artificial Intelligence* 28, pp. 293–331.

Pepik, B., M. Stark, P. Gehler, and B. Schiele [2012]. "Teaching 3d geometry to deformable part models". In: *Proc. CVPR*.

Pinheiro, P. O., R. Collobert, and P. Dollár [2015]. "Learning to Segment Object Candidates". In: *Proc. NIPS*.

Pollefeys, M., L. Van Gool, and M. Proesmans [1996]. "Euclidean 3D reconstruction from image sequences with variable focal lengths". In: *Proc. ECCV*. LNCS 1064/1065. Springer-Verlag.

Pollefeys, M., D. Nistér, J.-M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.-J. Kim, P. Merrell, et al. [2008]. "Detailed real-time urban 3d reconstruction from video". In: *IJCV* 78.2-3, pp. 143–167.

Pollefeys, M., L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch [2004]. "Visual modeling with a hand-held camera". In: *IJCV* 59.3, pp. 207–232.

Prasad, M., A. Zisserman, and A. W. Fitzgibbon [June 2006]. "Single View Reconstruction of Curved Surfaces". In: *Proc. CVPR*. Vol. 2, pp. 1345–1354.

Qi, G.-J., X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang [2008]. "Two-dimensional active learning for image classification". In: *Proc. CVPR*.

Rad, M. and V. Lepetit [2017]. "BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth". In: *Proc. ICCV*.

Ramanan, D. [2006]. "Learning to parse images of articulated bodies". In: *Proc. NIPS*. MIT Press.

— [2007]. "Using Segmentation to Verify Object Hypotheses". In: *Proc. CVPR*.

Reid, I. D. and D. W. Murray [1994]. "Active tracking of foveated feature clusters using affine structure".

Ren, S., K. He, R. Girshick, and J. Sun [2016]. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *Proc. NIPS*.

Ren, S., K. He, R. Girshick, and J. Sun [2015]. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *Proc. NIPS*.

Revaud, J., P. Weinzaepfel, Z. Harchaoui, and C. Schmid [2015a]. "Deep convolutional matching". In: *IJCV*, pp. 1164–1172.

— [2015b]. "Epicflow: Edge-preserving interpolation of correspondences for optical flow". In: *Proc. CVPR*.

Rezende, D. J., S. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess [2016]. "Unsupervised learning of 3d structure from images". In: *Proc. NIPS*.

Richardson, E., M. Sela, R. Or-El, and R. Kimmel [2017]. "Learning detailed face reconstruction from a single image". In: *Proc. CVPR*.

Riegler, G., A. O. Ulusoy, and A. Geiger [2017]. "Octnet: Learning deep 3d representations at high resolutions". In: *Proc. CVPR*.

Roberts, L. G. [1963]. "Machine perception of three-dimensional solids". PhD thesis. Massachusetts Institute of Technology. Dept. of Electrical Engineering. URL: `http://www.packet.cc/files/mach-per-3D-solids.html`.

Rocco, I., R. Arandjelović, and J. Sivic [2017]. "Convolutional neural network architecture for geometric matching". In: *Proc. CVPR*.

Rocco, I., R. Arandjelovic, and J. Sivic [2018]. "End-to-end weakly-supervised semantic alignment". In: *Proc. CVPR*.

Rubinstein, M., A. Joulin, J. Kopf, and C. Liu [2013]. "Unsupervised Joint Object Discovery and Segmentation in Internet Images". In: *Proc. CVPR*.

Russakovsky, O., L.-J. Li, and L. Fei-Fei [2015]. "Best of both worlds: human-machine collaboration for object annotation". In: *Proc. CVPR*.

Russakovsky, O., Y. Lin, K. Yu, and L. Fei-Fei [2012]. "Object-centric spatial pooling for image classification". In: *Proc. ECCV*. Springer.

Saenko, K., B. Kulis, M. Fritz, and T. Darrell [2010]. "Adapting visual category models to new domains". In: *Proc. ECCV*.

Savarese, S. and L. Fei-Fei [2007]. "3D generic object categorization, localization and pose estimation". In: *Proc. ICCV*.

Savinov, N., A. Seki, L. Ladicky, T. Sattler, and M. Pollefeys [2017]. "Quad-networks: unsupervised learning to rank for interest point detection". In: *Proc. CVPR*.

Saxena, A., M. Sun, and A. Y. Ng [2009]. "Make3D: Learning 3D Scene Structure from a Single Still Image". In: *IEEE PAMI* 31.5, pp. 824–840.

Saxena, A., S. H. Chung, and A. Y. Ng [2008]. "3-d depth reconstruction from a single still image". In: *IJCV* 76.1, pp. 53–69.

Schaffalitzky, F. and A. Zisserman [2002]. "Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?"" In: *Proc. ECCV*. Vol. 1. Springer-Verlag, pp. 414–431.

Schmidt, U. and S. Roth [2012]. "Learning rotation-aware features: From invariant priors to equivariant descriptors". In: *Proc. CVPR*.

Schneiderman, H. and T. Kanade [2000]. "A Statistical Method for 3D Object Detection Applied to Faces and Cars". In: *Proc. CVPR*.

Schönberger, J. L. and J.-M. Frahm [2016]. "Structure-from-Motion Revisited". In: *Proc. CVPR*.

Schroff, F., A. Criminisi, and A. Zisserman [2007]. "Harvesting Image Databases from the Web". In: *Proc. ICCV*.

Sedaghat, N. and T. Brox [2015]. "Unsupervised Generation of a Viewpoint Annotated Car Dataset from Videos". In: *Proc. ICCV*.

Shechtman, E. and M. Irani [2007]. "Matching local self-similarities across images and videos". In: *Proc. CVPR*.

Shi, J. and J. Malik [2000]. "Normalized Cuts and Image Segmentation". In: *IEEE PAMI* 22.8, pp. 888–905.

Shi, Z., T. Hospedales, and T. Xiang [2015]. "Bayesian Joint Modelling for Object Localisation in Weakly Labelled Images". In: *IEEE PAMI* 37.10, pp. 1959–1972.

Shotton, J., B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon [2013]. "Scene coordinate regression forests for camera relocalization in RGB-D images". In: *Proc. CVPR*.

Simo-Serra, E., E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer [2015]. "Discriminative learning of deep convolutional feature point descriptors". In: *Proc. ICCV*.

Simon, M. and E. Rodner [2015]. "Neural activation constellations: Unsupervised part model discovery with convolutional networks". In: *Proc. ICCV*.

Simonyan, K., A. Vedaldi, and A. Zisserman [2014]. "Learning Local Feature Descriptors Using Convex Optimisation". In: *IEEE PAMI*.

Simonyan, K. and A. Zisserman [2015]. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*.

Singh, S., A. Gupta, and A. A. Efros [2012]. "Unsupervised Discovery of Mid-level Discriminative Patches". In: *Proc. ECCV*.

Sinha, A., A. Unmesh, Q. Huang, and K. Ramani [2017]. "SurfNet: Generating 3D shape surfaces using deep residual networks". In: *Proc. CVPR*.

Siva, P., C. Russell, T. Xiang, and L. Agapito [2013]. "Looking beyond the image: Unsupervised learning for object saliency and detection". In: *Proc. CVPR*.

Sivic, J., B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman [2005]. "Discovering object categories in image collections". In:

Snavely, N., S. Seitz, and R. Szeliski [2006]. "Photo tourism: exploring photo collections in 3D". In: *Proc. ACM SIGGRAPH*. 3, pp. 835–846.

Song, H. O., R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell [2014]. "On learning to localize objects with minimal supervision". In: *CoRR*.

Sparr, G. [1996]. "Simultaneous reconstruction of scene structure and camera locations from uncalibrated image sequences". In: *Proc. ICPR*.

Sturm, P. and W. Triggs [1996]. "A factorization based algorithm for multi-image projective structure and motion". In: *Proc. ECCV*, pp. 709–720.

Su, H., C. R. Qi, Y. Li, and L. J. Guibas [2015]. "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views". In: *Proc. ICCV*.

Sun, B., J. Feng, and K. Saenko [2016]. "Return of Frustratingly Easy Domain Adaptation". In: *Proc. AAAI*.

Sun, J. and J. Ponce [2015]. "Learning Dictionary of Discriminative Part Detectors for Image Categorization and Cosegmentation". In: *IJCV*.

Tang, K., A. Joulin, L.-J. Li, and L. Fei-Fei [2014]. "Co-localization in real-world images". In: *Proc. CVPR*.

Tatarchenko, M., A. Dosovitskiy, and T. Brox [2016]. "Multi-view 3d models from single images with a convolutional network". In: *Proc. ECCV*.

Thewlis, J., H. Bilen, and A. Vedaldi [2017a]. "Unsupervised learning of object landmarks by factorized spatial embeddings". In: *Proc. ICCV*.

— [2017b]. "Unsupervised object learning from dense invariant image labelling". In: *Proc. NIPS*.

Thewlis, J., S. Zheng, P. Torr, and A. Vedaldi [2016]. "Fully-Trainable Deep Matching". In: *Proc. BMVC.*

Thomas, A., V. Ferrar, B. Leibe, T. Tuytelaars, B. Schiel, and L. Van Gool [2006]. "Towards multi-view object class detection". In: *Proc. CVPR*.

Tighe, J., M. Niethammer, and S. Lazebnik [2014]. "Scene Parsing with Object Instances and Occlusion Ordering". In: *Proc. CVPR*.

Tola, E., V. Lepetit, and P. Fua [2010]. "DAISY: An Efficient Dense Descriptor Applied to Wide Baseline Stereo". In: *IEEE PAMI* 32.5, pp. 815–830.

Tolias, G., R. Sicre, and H. Jégou [2015]. "Particular object retrieval with integral max-pooling of CNN activations". In: *CoRR*.

Tomasi, C. and T. Kanade [1992]. "Shape and Motion from Image Streams under Orthography: A Factorization Approach". In: *IJCV* 9.2, pp. 137–154.

Tompson, J. J., A. Jain, Y. LeCun, and C. Bregler [2014]. "Joint training of a convolutional network and a graphical model for human pose estimation". In: *Proc. NIPS*.

Tong, S. and D. Koller [2002]. "Support vector machine active learning with applications to text classification". In: vol. 2, pp. 45–66.

Torr, P. H. S. and A. Zisserman [2000]. "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry". In: *CVIU* 78, pp. 138–156.

Torralba, A., K. P. Murphy, and W. T. Freeman [2004]. "Sharing features: efficient boosting procedures for multiclass object detection". In: *Proc. CVPR*, pp. 762–769.

Torralba, A. and A. Oliva [2002]. "Depth estimation from image structure". In: *IEEE PAMI* 24.9, pp. 1226–1238.

Toshev, A. and C. Szegedy [2014]. "Deeppose: Human pose estimation via deep neural networks". In: *Proc. CVPR*.

Tulsiani, S. and J. Malik [2015]. "Viewpoints and keypoints". In: *Proc. CVPR*.

Tulsiani, S., T. Zhou, A. A. Efros, and J. Malik [2017]. "Multi-view supervision for single-view reconstruction via differentiable ray consistency". In: *Proc. CVPR*.

Tzeng, E., J. Hoffman, T. Darrell, and K. Saenko [2015]. "Simultaneous deep transfer across domains and tasks". In: *Proc. ICCV*.

Tzeng, E., J. Hoffman, N. Zhang, K. Saenko, and T. Darrell [2014]. "Deep domain confusion: Maximizing for domain invariance". In: *CoRR*.

Ullman, S., ed. [1979]. *The interpretation of visual motion*. Cambridge, MA: MIT Press.

Vacchetti, L., V. Lepetit, and P. Fua [2004]. "Stable real-time 3d tracking using online and offline information". In: *IEEE PAMI* 26.10, pp. 1385–1391.

Vendrig, J., I. Patras, C. Snoek, M. Worring, J. den Hartog, S. Raaijmakers, J. van Rest, and D. A. van Leeuwen [2002]. "Feature Extraction by Active Learning". In: *Proc. TREC*.

Vicente, S., J. Carreira, L. Agapito, and J. Batista [2014]. "Reconstructing pascal voc". In: *Proc. CVPR*.

Vicente, S., C. Rother, and V. Kolmogorov [2011]. "Object Cosegmentation". In: *Proc. CVPR*.

Vijayanarasimhan, S. and K. Grauman [2008]. "Keywords to visual categories: Multiple-instance learning forweakly supervised object categorization". In: *Proc. CVPR*.

— [2009]. "What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations". In: *Proc. CVPR*.

Wah, C., S. Branson, P. Welinder, P. Perona, and S. Belongie [2011a]. *The Caltech-UCSD Birds-200-2011 Dataset*. Tech. rep. CNS-TR-2011-001. California Institute of Technology.

Wah, C., S. Branson, P. Perona, and S. Belongie [2011b]. "Multiclass recognition and part localization with humans in the loop". In: *Proc. ICCV*.

Wang, C., W. Ren, K. Huang, and T. Tan [2014]. "Weakly supervised object localization with latent category learning". In: *Proc. ECCV*.

Wang, F., Q. Huang, and L. J. Guibas [2013]. "Image co-segmentation via consistent functional maps". In: *Proc. ICCV*.

Wang, P., X. Shen, Z. L. Lin, S. Cohen, B. L. Price, and A. L. Yuille [2015a]. "Joint Object and Part Segmentation using Deep Learned Potentials". In: *Proc. ICCV*.

Wang, P., X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille [2015b]. "Joint object and part segmentation using deep learned potentials". In: *Proc. ICCV*.

Wang, Q., X. Zhou, and K. Daniilidis [2017]. "Multi-Image Semantic Matching by Mining Consistent Features". In: *Proc. CVPR*.

Wang, S., R. Clark, H. Wen, and N. Trigoni [2018]. "End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks". In: *Intl. J. of Robotics Research* 37.4-5, pp. 513–542.

Weber, M., W. Einhauser, M. Welling, and P. Perona [2000]. "Viewpoint-Invariant Learning and Detection of Human Heads". In: *Proc. 4th IEEE Int. Conf. Autom. Face and Gesture Recog., FG2000*.

Wiles, O., A. S. Koepke, and A. Zisserman [2018]. "Self-supervised learning of a facial attribute embedding from video". In: *Proc. BMVC.*

Wiles, O. and A. Zisserman [2017]. "SilNet: Single-and Multi-View Reconstruction by Learning from Silhouettes". In: *Proc. BMVC.*

Witkin, A. P. [1981]. "Recovering surface shape and orientation from texture". In: *Artificial intelligence* 17.1-3, pp. 17–45.

Wu, C. [2013]. "Towards linear-time incremental structure from motion". In: *Proc. 3DV*.

Wu, J., C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum [2016]. "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling". In: *Proc. NIPS*.

Xiang, Y., R. Mottaghi, and S. Savarese [2014]. "Beyond pascal: A benchmark for 3d object detection in the wild". In: *Proc. WACV*.

Yang, J., R. Yan, and A. G. Hauptmann [2007]. "Cross-domain video concept detection using adaptive svms". In: *Proc. ACMM*.

Yang, S., L. Bo, J. Wang, and L. G. Shapiro [2012a]. "Unsupervised template learning for fine-grained object recognition". In: *Proc. NIPS*.

Yang, Y., C. Feng, Y. Shen, and D. Tian [2018]. "FoldingNet: Interpretable Unsupervised Learning on 3D Point Clouds". In: *Proc. CVPR*.

Yang, Y., S. Hallman, D. Ramanan, and C. C. Fowlkes [2012b]. "Layered object models for image segmentation". In: *IEEE PAMI* 34.9, pp. 1731–1743.

Yang, Y. and D. Ramanan [2013]. "Articulated human detection with flexible mixtures of parts". In: *IEEE PAMI* 35.12, pp. 2878–2890.

Yi, K. M., E. Trulls, V. Lepetit, and P. Fua [2016]. "Lift: Learned invariant feature transform". In: *Proc. ECCV*.

Yu, C.-N. J. and T. Joachims [2009]. "Learning structural SVMs with latent variables". In: *Proc. ICML*.

Zach, C., M. Klopschitz, and M. Pollefeys [2010]. "Disambiguating visual relations using loop constraints". In: *Proc. CVPR*.

Zagoruyko, S. and N. Komodakis [2015]. "Learning to compare image patches via convolutional neural networks". In: *Proc. CVPR*.

Zanfir, A. and C. Sminchisescu [2018]. "Deep Learning of Graph Matching". In: *Proc. CVPR*.

Zhang, N., J. Donahue, R. Girshick, and T. Darrell [2014a]. "Part-based R-CNNs for Fine-grained Category Detection". In: *Proc. ECCV*.

Zhang, N., R. Farrell, F. Iandola, and T. Darrell [2013]. "Deformable part descriptors for fine-grained recognition and attribute prediction". In: *Proc. ICCV*.

Zhang, N., M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev [2014b]. "Panda: Pose aligned networks for deep attribute modeling". In: *Proc. CVPR*.

Zhang, R., P. Isola, and A. A. Efros [2016]. "Colorful image colorization". In: *Proc. ECCV*. Springer, pp. 649–666.

Zhang, Y., Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee [2018]. "Unsupervised Discovery of Object Landmarks as Structural Representations". In: *Proc. CVPR*.

Zhang, Z., A. G. Schwing, S. Fidler, and R. Urtasun [2015]. "Monocular Object Instance Segmentation and Depth Ordering with CNNs". In: *Proc. ICCV*.

Zhou, T., M. Brown, N. Snavely, and D. G. Lowe [2017]. "Unsupervised learning of depth and ego-motion from video". In: *Proc. CVPR*.

Zhou, T., Y. Jae Lee, S. X. Yu, and A. A. Efros [2015a]. "FlowWeb: Joint Image Set Alignment by Weaving Consistent, Pixel-Wise Correspondences". In: *Proc. CVPR*.

Zhou, T., P. Krähenbühl, M. Aubry, Q. Huang, and A. A. Efros [2016]. "Learning Dense Correspondence via 3D-guided Cycle Consistency". In: *Proc. CVPR*.

Zhou, X., M. Zhu, and K. Daniilidis [2015b]. "Multi-image matching via fast alternating minimization". In: *Proc. CVPR*.

Zhu, M., K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis [2014]. "Single image 3D object detection and pose estimation for grasping". In: *Proc. Intl. Conf. on Robotics and Automation*.