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-labelled 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.

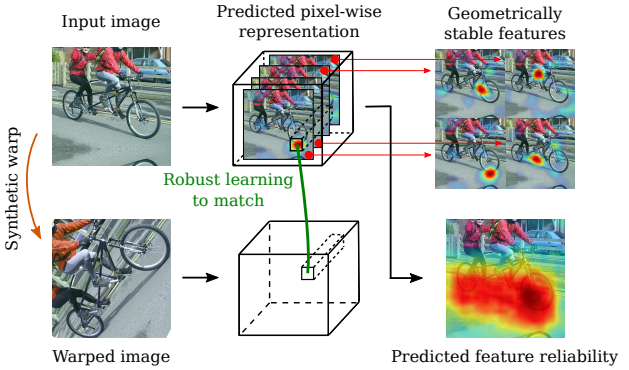

Figure 1. Our approach leverages correspondences obtained from synthetic warps in order to self-supervise the learning of a dense image representation. This results in highly localized and geometrically stable features. The use of a novel robust probabilistic formulation allows to additionally predict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

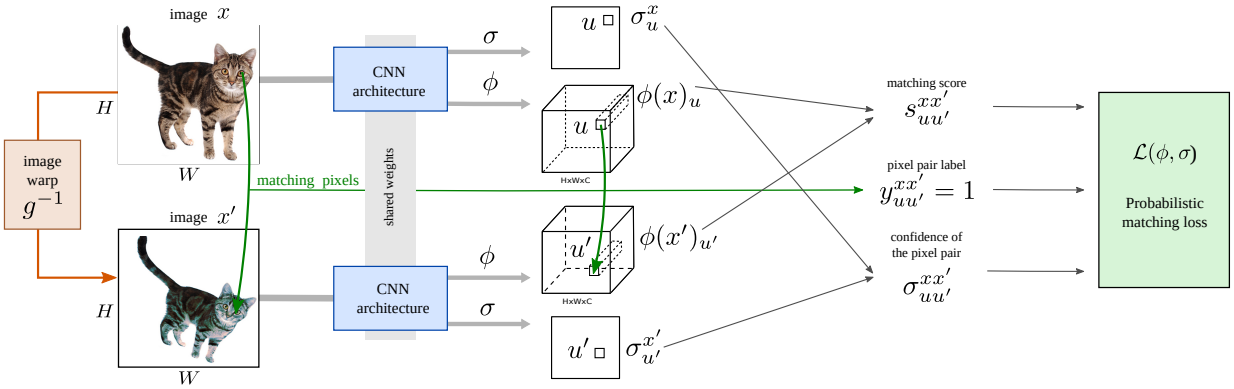

Figure 2. **Overview of our approach**. Image x is warped into image x' using the transformation g^{-1}. Pairs of pixels and their labels (encoding whether they match or not according to g^{-1}) are used together with a probabilistic matching loss to train our architecture that predicts i) a dense image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

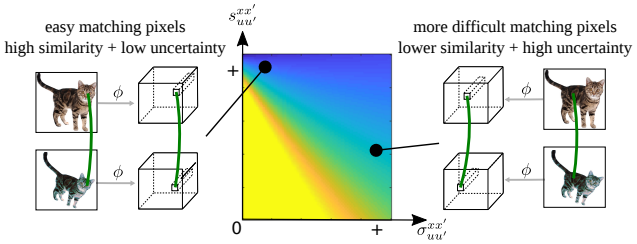

Figure 3. **Illustration of the probabilistic loss.** The plot shows values of the loss for positive pairs ($y_{xx'} = 1$, bluer means a smaller loss) as a function of the similarity between the pixel representations $s^{xx'}_{uu'}$ and the uncertainty $\sigma^{xx'}_{uu'}$ who's inverse $\sigma^{xx'\,-1}_{uu'}$ 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 $y^{\mathbf{x}\mathbf{x}'}_{uu'} \in \{-1, 0, +1\}$, 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* $\ell^{\mathbf{x}\mathbf{x}'}_{uu'}$:

$$\ell^{\mathbf{x}\mathbf{x}'}_{uu'} = \begin{cases} 1 - s^{\mathbf{x}\mathbf{x}'}_{uu'} & y^{\mathbf{x}\mathbf{x}'}_{uu'} = 1, \\ 0 & y^{\mathbf{x}\mathbf{x}'}_{uu'} = 0, \\ s^{\mathbf{x}\mathbf{x}'}_{uu'} & y^{\mathbf{x}\mathbf{x}'}_{uu'} = -1. \end{cases} \tag{4}$$

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 $\sigma^{\mathbf{x}}_u \in \mathbb{R}^+$ 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, $\sigma^{\mathbf{x}}_u$ 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 in the quality of both descriptors $\phi(\mathbf{x})_u$ and $\phi(\mathbf{x}')_{u'}$, the value $\sigma_{uu'}$ is small. In this case, the distribution (5) has a sharp peak around 1 or 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:

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

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 $y^g_{uu'}$ are easily obtained as

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

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 $s^{\mathbf{x}\mathbf{x}'}_{uu'}$ 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

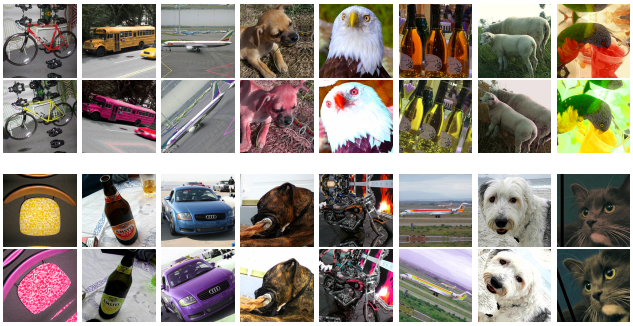

Figure 4. **Example geometric and appearance transformations** used to supervise the learning of our representation. The first (resp. third) row displays original images while the second (resp. fourth) row shows their transformed versions.

$\mathbf{x} = \hat{\mathbf{x}} \circ \hat{g}^{-1}$ 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.

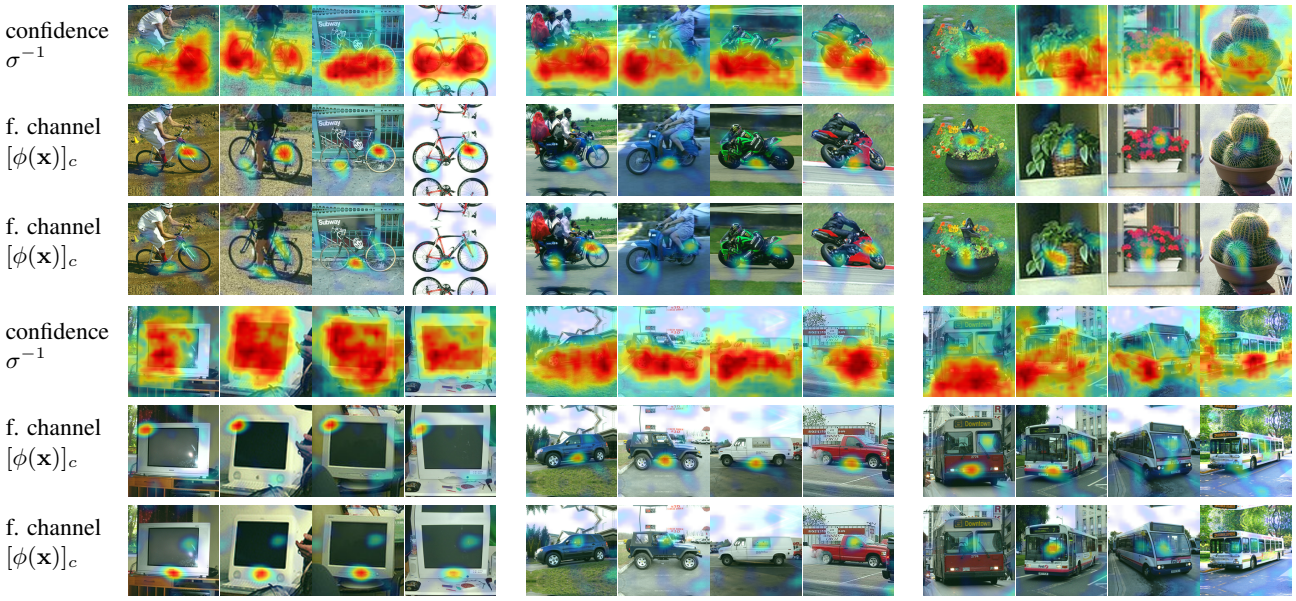

Figure 5. **Qualitative analysis of the learned equivariant feature representation** ϕ visualizing predicted confidence maps σ^{-1} and several responses $\max([\phi(\mathbf{x})]_c, 0)$ of different channels c of the representation, for six different categories.

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.

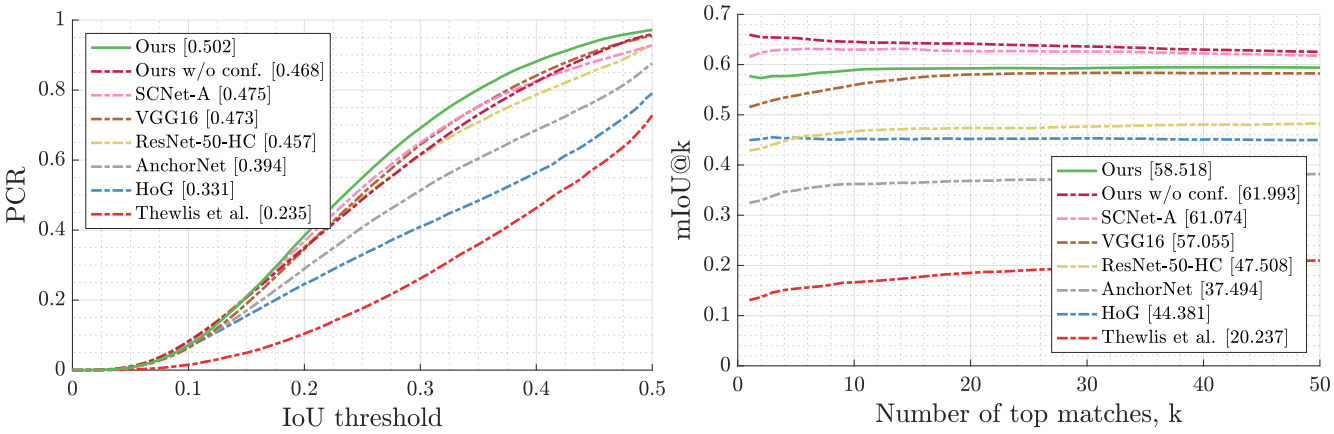

Figure 6. **Region matching performance on PF-Pascal**. Features are matched directly without any spatial regularization. Left: region matching precision (PCR). Right: region matching accuracy (mIoU@k). Note that unlike all other reported approaches, SCNet-A [11] is a fully supervised method.

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

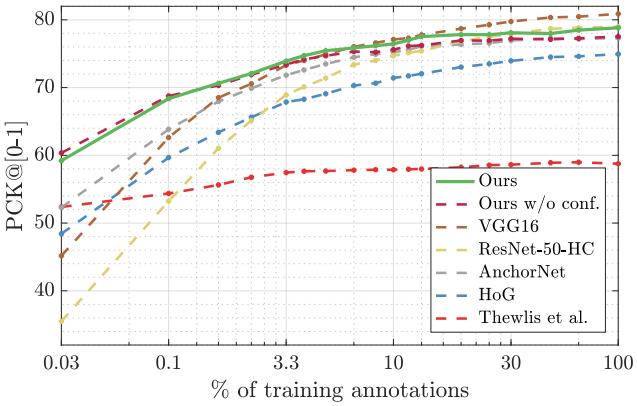

Figure 7. **Keypoint prediction on Pascal3D**. We report the area under the PCK-over-alpha curve as a function of the number of training annotations.

[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.

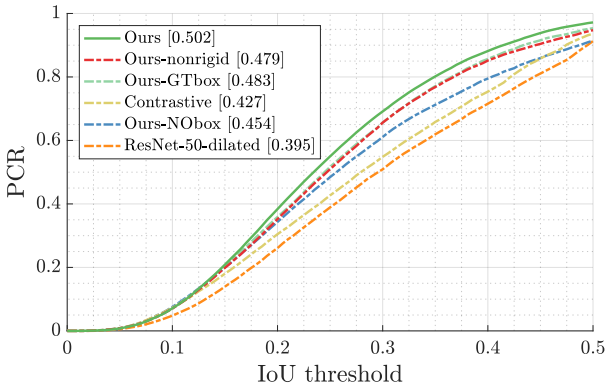

Figure 8. **Ablation study on PF-Pascal**. The region matching performance of several variants of our method (see appendix A for details of each variant).

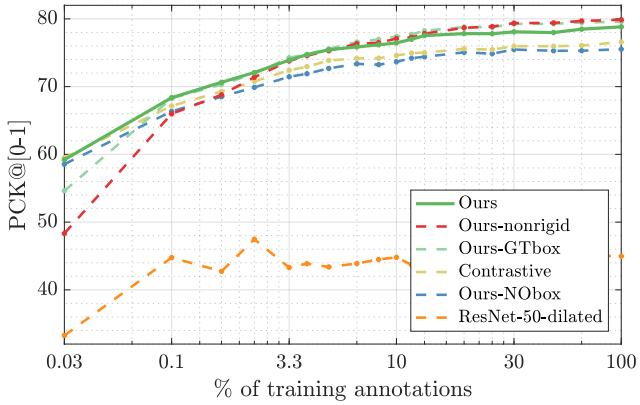

Figure 9. **Ablation study on the few-shot keypoint detection task on Pascal3D**. We report the area under the PCK-over-alpha curve as a function of the number of training annotations for several variants of our method. For details of each variant see appendix A.

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

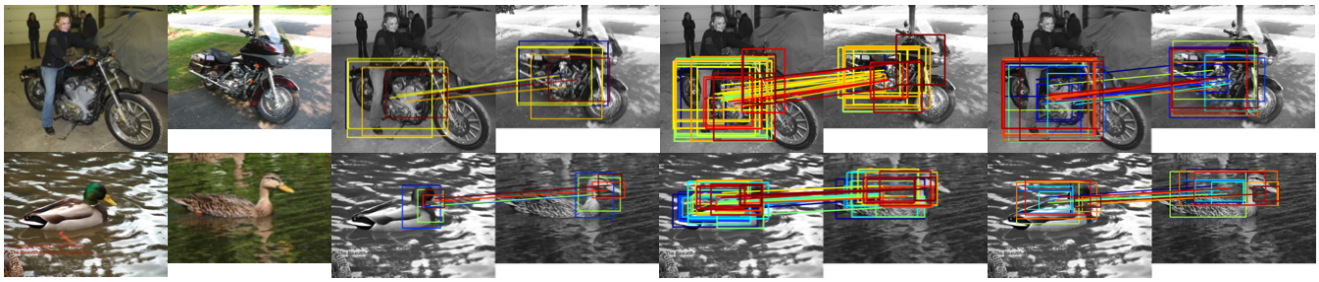

Figure 10. **Region matching examples** for pairs of motorbike (top) and duck (bottom) images. From left to right: source and target images, HOG with NAM matching [10], ours, SCNet-A [11]. We show correctly matched boxes, color-coded according to matching score (red: higher, blue: lower).

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