Learning to Distill Convolutional Features into Compact Local Descriptors

Jongmin Lee[1,2] Yoonwoo Jeong[1] Seungwook Kim[1] Juhong Min[1,2] Minsu Cho[1,2]
[1]POSTECH [2]NPRC
{ljm1121, jeongyw12382, wookiekim, juhongm999, mscho}@postech.ac.kr

Abstract

Extracting local descriptors or features is an essential step in solving image matching problems. Recent methods in the literature mainly focus on extracting effective descriptors, without much attention to the size of the descriptors. In this work, we study how to learn a compact yet effective local descriptor. The proposed method distills multiple intermediate features of a pretrained convolutional neural network to encode different levels of visual information from local textures to non-local semantics, resulting in local descriptors with a designated dimension. Experiments on standard benchmarks for semantic correspondence show that it achieves significantly improved performance over existing models, with up to a 100 times smaller size of descriptors. Furthermore, while trained on a small-sized dataset for semantic correspondence, the proposed method also generalizes well to other image matching tasks, performing comparable result to the state of the art on wide-baseline matching and visual localization benchmarks.

1. Introduction

Extracting reliable local features is a fundamental part of computer vision problems, including Simultaneous Localization and Mapping (SLAM), Structure-from-Motion (SfM), and 3D reconstruction [13, 18]. The process of extracting local feature representations can be divided into two steps: keypoint localization and descriptor extraction. It has been observed in the literature that local descriptors learned in neural networks [11, 44, 52] can be more effective than traditional, hand-crafted representations [8, 37, 57].

Convolutional neural networks (CNN) are the most prominent amongst such deep neural networks, yielding cutting-edge results for different tasks such as pose estimation [67] and object detection [20, 35]. A common approach to improving local descriptors [11, 30, 41] is to increase the capacity of the descriptors, resulting in an increased size, albeit enhanced performances. The compactness of descriptors, however, is also important in practice. For example, in SfM pipelines [60], heavy descriptors from many images may hinder multi-view matching and triangulation [18, 19, 27].

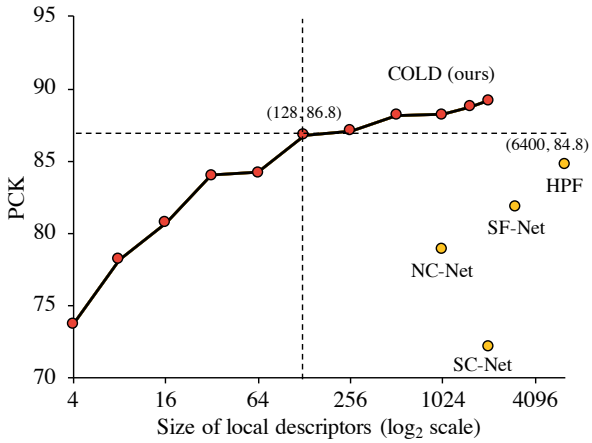


Figure 1. **Descriptor size vs. matching accuracy (PCK) on PF-PASCAL.** The proposed descriptor method, COLD, outperforms other methods with significantly smaller descriptors. Note that it performs best even with the size of 128. See Table 1 for the detail.

In this work, we propose to learn a compact local description method, dubbed COLD, that distills intermediate features of pretrained convolutional networks using multi-layer feature transformation and fusion. Leveraging the compositional feature hierarchy of CNNs [21] pretrained on ImageNet [9], the proposed method extracts multiple intermediate features to encode hierarchical information of the images and learn to compress them into a single compact feature map containing, which contains descriptors for the entire input image. The resultant descriptors show impressive performance on the task of semantic correspondence while being up to 100 times smaller than descriptors obtained from preceding methods [5, 17, 24, 28, 30, 41, 53, 54, 55, 61], suggesting a successful compromise to the effectiveness-compactness trade-off. Figure 1 shows

[1]Pohang University of Science and Technology, Pohang, Korea
[2]The Neural Processing Research Center, Seoul, Korea

the trade-off of COLD compared to existing semantic correspondence models. The efficacy our model can also be observed outside the training domain of semantic correspondence; it performs comparably to state-of-the-art descriptors extraction models [11, 52] trained specifically on the target domain of wide-baseline matching or long-term visual localization. Experiments demonstrate the effectiveness of COLD not only on semantic correspondence benchmarks - PF-PASCAL [16], PF-WILLOW [15], and SPair-71k [42] - but also on wide-baseline matching and visual localization benchmarks, including HPatches [1] dataset and Aachen day-night [58] dataset.

2. Related work

Traditional local descriptors. The most well-known traditional local descriptor extraction approaches include SIFT [37], SURF [3], HOG [8, 37] and BRIEF [31]. However, traditional methods have shown limitations in capturing high-level semantics of target images.

Deep-learned local descriptors. Local descriptors obtained using convolutional neural networks have been able to capture high-level semantics of images. In semantic correspondence tasks, Choy *et al*. [5] make use of correspondence contrastive loss, Rocco *et al*. [55] propose a weakly-supervised trainable network detecting spatially consistent matches, and Lee *et al*. [30] use binary foreground masks with synthetic geometric deformation as supervisory signals for training. Other methods [24, 28, 29, 41, 47, 48] also address local descriptor extraction in the semantic correspondence domain, with various novelties targeted at different aspects.

Local descriptor extraction in wide-baseline matching can be divided into learning patch-based descriptors from predefined patches (detect-then-describe), and learning keypoints and their corresponding descriptors together (detect-and-describe). Mishchuk *et al*. [44] use triplet loss inspired by Lowe's matching criterion [37] to learn the descriptors. Other key methods to learn patch-based descriptors include [38, 39, 66]. Methods learning both keypoints and descriptors together have also been effective. [49, 62] propose an end-to-end method to extract patch-based descriptors. The above methods make use of detect-then-describe approach, which sample the patches on the image.

The following methods obtain the corresponding descriptors of keypoints in a detect-and-describe manner, instead of sampling patches explicitly. Noh *et al*. [46] integrate an attention mechanism for local feature selection on image retrieval. DeTone *et al*. [10] introduce a self-supervised framework to train keypoints detector and dense descriptors. Dusmanu *et al*. [11] increase the stability of the extracted features by postponing keypoint selection. Revaud *et al*. [52] show that reliability and repeatability should also be considered for better descriptors.

Unlike those methods, which does not take into account the size of descriptors, we propose a multi-layer feature fusion method, which directly outputs compact and robust descriptors - cutting on keypoint detection overhead and showing descriptor generalizability.

Methods using multiple layers. Long *et al*. [36] integrate a portion of its deeper intermediate layers for the task of semantic segmentation. The following methods [33, 34] also exploit a subset of layers along a backbone convolutional neural network. More recently, Tan *et al*. [64] propose a bidirectional feature pyramid network for easier and faster feature fusion. Min *et al*. [41] leverage intermediate features along its backbone network to represent images by hyperpixels. Min *et al*. [43] which is a post work of [41] learn to compose the intermediate features by continuous relaxation using Gumbel-softmax [25, 40].

In contrast to the mentioned approaches, our model is as simple as [33, 36] yet effective. Our model exploits all intermediate layers of the backbone network, but outputs more compact representation. Despite this compactness, our approach yields strong results in semantic correspondence and comparable performance on wide-baseline matching and long-term visual localization.

Knowledge distillation. Note that in the literature, knowledge distillation or transfer [23, 56, 50] refers to the process of transferring learned knowledge from a larger teacher model to a smaller student model. In contrast, the term of feature distillation in our context is used for the process of transferring knowledge from a large set of convolutional features to a compact and small local descriptor.

Comparison to most related methods [41, 14]. We adopt HPF [41] as a baseline model and also obtains the base features in the same manner. While our model parameters are trained by backpropagation using the correspondence supervision directly, HPF [41] does not train the network parameters, but instead carries out heuristic beam-search algorithm using PCK values. Unlike [41], we do not use any additional spatial matching, *i.e*., probabilistic Hough matching (PHM) [4]. S2DHM [14] also leverages hypercolumn local descriptors on CNN. While S2DHM [14] only targets visual localization tasks, we concentrate on generally extracting local descriptors, which can be applied to various benchmarks, not limited to visual localization. The usage of element-wise summation as our feature aggregation operation – which reduces the output descriptor channel size – and feature fusion operation are also the unique contribution of ours - S2DHM uses feature concatenation instead.

3. Compact local descriptor networks

In this section we demonstrate our approach in three parts: feature extraction, feature distillation, and the training objective. First, we extract feature maps from interme-

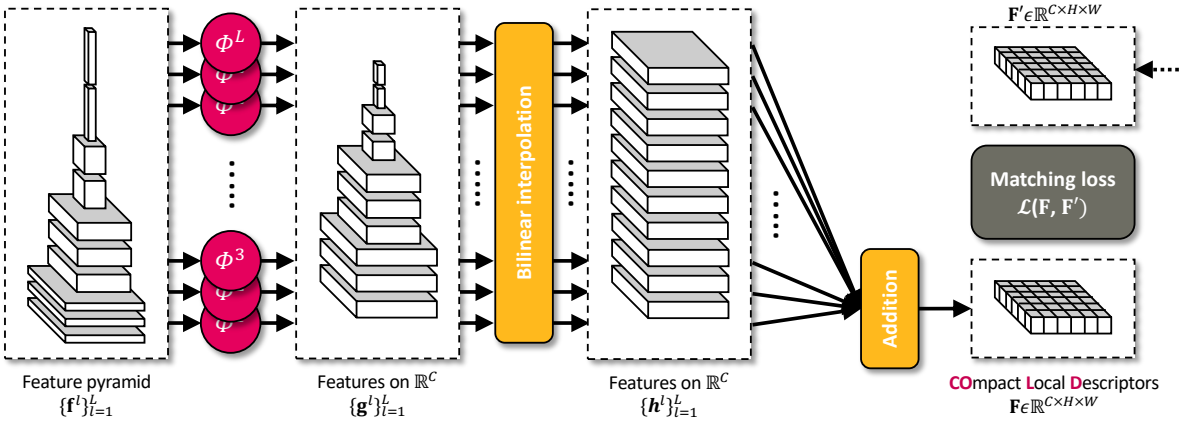


Figure 2. Overall architecture. Extracted features $\mathbf{f}$ from the multiple layers on ImageNet pretrained networks are transformed by MLPs Φ. The outputs $\mathbf{g}$ become the same size features $\mathbf{h}$ by bilinear interpolation. The final results $\mathbf{F}$ obtained by element-wise addition are learned by matching loss $\mathcal{L}$.

diate layers of feature extraction network pretrained on a classification task. Second, each intermediate feature map is then passed to its corresponding transformation module that compactly distills each feature map by projecting onto a subspace $\mathbb{R}^C$ shared across the multi-level features. Third, we train our model using semantically related keypoint correspondences as a supervisory signal. Figure 2 shows the overall architecture of our model.

3.1. Model architecture

Feature extraction. We adopt ResNet [21] as our feature extraction network. Given an input image, our feature extraction network extracts a series of intermediate feature maps $\{\mathbf{f}^l\}_{l=1}^L$. The extracted features are then passed to a learnable module to generate compact local descriptors which is detailed in the next subsection.

Feature transformation. The feature transformation module Φ^l at each layer l consists of two convolutional layers. The first convolutional layer carries out 1x1 convolution on the input features map $\mathbf{f}^l$, reducing their channel dimensions to one eighth of their original dimensions. This operation is followed by batch normalization and ReLU. Each resultant feature is then passed to another 1x1 convolutional layer which performs a linear transformation on each spatial position of the feature map. This convolutional layer projects each feature vector on the same subspace $\mathbb{R}^C$: $\mathbf{g}^l = \Phi^l(\mathbf{f}^l) \in \mathbb{R}^{C\times H^l\times W^l}$ where $l \in \{1, 2, ..., L\}$ denotes the index of an intermediate layer of the backbone network.

Feature fusion. The raw output feature maps of the transformation $\{\mathbf{g}^l\}_{l=1}^L$ have different numbers of local descriptors on their spatial dimension $H^l \times W^l$. To generate compact local descriptors on a dense grid for fine-grained keypoint localisation, we upsample each feature map: $\mathbf{h}^l = \zeta(\mathbf{g}^l)$ where ζ denotes a function that bilinearly interpolates to spatial size of features to $H \times W$. We then aggregate information from multiple visual aspects by simple addition and obtain compact feature map: $\mathbf{F} = \sum_{l\in L} \mathbf{h}^l$. Despite its simplicity, our proposed method shows high reliability in establishing correspondences on semantic correspondence [15, 16, 42] and even generalize effectively when evaluated on different domains [1, 58], *e.g.*, the task of wide-baseline matching and long-term visual localization.

Both upsampling and adding outputs of different layers $\{\mathbf{f}^l\}_{l=1}^L$ capture spatial position information of the original image because we consider strides and paddings of convolutions; the center coordinate $\mathbf{c}^l$ of receptive field for each feature vector $\mathbf{f}^l_{:ij}$ is defined as $\mathbf{c}^l = \mathbf{c}^{l-1} + \left(\frac{k_l-1}{2} - p_l\right)\cdot j_{l-1}$, where k_l and p_l are kernel and padding sizes of the l-th conv layer and j is pixel-level distance between two adjacent feature vectors in input image space; initially, $k_0 = 1$ and $j_0 = 1$ and $\mathbf{c}_0$ is a pixel coordinate. Standard CNNs [21, 63] use kernel size of $k_l = 2p_l + 1$, leading to $\mathbf{c}_l = \mathbf{c}_{l-1}$ as ours does. Thus, the outputs $\{\mathbf{g}^l\}_{l=1}^L$ of simple bilinear upsampling can align different feature maps so that the resultant compact local descriptors $\mathbf{F}_{:ij}$ retain their relative location information of the original image. Aggregating these position-aligned feature maps using a simple addition operation thus results in compact local representation $\mathbf{F}_{:ij}$ which describes semantics of multi-scale receptive fields. The effects of the multi-level fusion will be detailed in Sec. 4.4.

3.2. Training objective

To optimize parameters of the proposed network, we train our network on pairs of semantically related images.Given a pair of compact feature maps $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ and $\mathbf{F}' \in \mathbb{R}^{C\times H'\times W'}$, we first compute a variant of cosine

similarity:

$$\hat{\mathbf{C}}_{ijkl} = \mathrm{ReLU}\Big(\frac{\mathbf{F}_{:ij} \cdot \mathbf{F'}_{:kl}}{\|\mathbf{F}_{:ij}\|\|\mathbf{F'}_{:kl}\|}\Big)^2, \tag{1}$$

where the ReLU non-linearity clamps scores of dissimilar feature pairs to zero and the exponent suppresses noisy match scores. We then normalize each (one) source to (many) target correlation scores $\hat{\mathbf{C}}_{ij::}$ using softmax as follows:

$$\mathbf{C}_{ijkl} = \frac{\exp\left(\hat{\mathbf{C}}_{ijkl}\right)}{\sum_{(y,x)} \exp\left(\hat{\mathbf{C}}_{ijyx}\right)}. \tag{2}$$

such that $\sum_{(y,x)} \mathbf{C}_{ijyx} = 1$. Given semantically paired keypoint correspondences $\{(\mathbf{k}_i, \mathbf{k'}_i)\}_{i=1}^{K} \in \mathcal{K}$, our training objective is formulated as follows:

$$\mathcal{L} = -\frac{1}{|\mathcal{K}|} \sum_{(\mathbf{k},\mathbf{k'}) \in \mathcal{K}} \lambda(\mathbf{k}, \mathbf{k'}) \log \mathbf{C}_{ijkl}, \tag{3}$$

where (i, j) and (k, l) are spatial positions of features nearest to the given source and target ground-truth keypoints, $\mathbf{k}$ and $\mathbf{k'}$, respectively and λ is a weighting function defined as

$$\lambda(\mathbf{k}, \mathbf{k'}) = \begin{cases} (\|\mathcal{T}(\mathbf{k}) - \mathbf{k'}\|/\eta)^2 & \text{if } \|\mathcal{T}(\mathbf{k}) - \mathbf{k'}\| < \eta, \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

where $\mathcal{T}(\cdot)$ is a function that transfers given keypoint using nearest-neighbour assignment based on the predicted correlation matrix $\mathbf{C}$ [41]. The weighting term λ outputs a value proportional to the distance between the ground truth $\mathbf{k'}$ and the prediction $\mathcal{T}(\mathbf{k})$ if the distance is below a certain threshold η, thus helping the network focus on the incorrect matches during training. We set the threshold η to $0.1 \cdot \max(h, w)$, where h and w are respectively the height and weight of the bounding box. Our training objective is to minimize the negative log likelihood of true correspondences with a dynamic weighting term.

4. Experiments

In section 4.1, 4.2 and 4.3, we report the results on various visual correspondence benchmarks [1, 15, 16, 26, 41, 58] and the evaluation setting. In section 4.4, we show the ablation of our proposed model and additional experiments.

4.1. Implementation details

We use the ResNet [21] architecture pre-trained on ImageNet [9] as the backbone network. All the outputs from the bottleneck units of the backbone preceding the final average pooling layers are extracted. We freeze the parameters of the backbone network during training. Our network is trained for 30 epochs using a SGD optimizer with a uniform learning rate of 0.03 and batch size of 8. The training time under these conditions takes approximately 5 hours on a NVIDIA Titan-XP GPU. We resize the spatial size of input images to 240×240, which results in output features of size $\mathbf{F} \in \mathbb{R}^{C \times 60 \times 60}$, and therefore correlation matrix of size $\mathbf{C} \in \mathbb{R}^{60 \times 60 \times 60 \times 60}$. We use the training split of PF-PASCAL [16] with sparse ground-truth keypoints correspondences.

Processing output. We set the output spatial size of our model according to the characteristics of target task. Semantic correspondence is important to capture the high-level semantics. Wide-baseline matching is important to capture the low-level geometry.

The output descriptor sizes are $\mathbb{R}^{C \times 60 \times 60}$ for semantic correspondence. After construction of correlation matrix $\mathbf{C}$, we transfer the source image keypoints to target image by $\mathcal{T}$. To transfer source keypoints, we aggregate the displacement vectors from the center points covered by receptive fields of features, as in [41].

The output descriptor sizes are $\mathbb{R}^{C \times 160 \times 160}$ for wide-baseline matching [1] and long-term visual localization [58]. Here, we use the grid points strategy where all points of the descriptor are regarded as keypoints. There exists a receptive field on the original input image which each pixel on $\mathbf{F}$ represents. We use center points of the receptive fields as keypoints, and designate the output descriptors of the corresponding pixels to the newly declared keypoints. Consequently, these dense grid keypoints may slightly deviate from corners or edges of the input image, but have been verified to include the high majority of sparse keypoints acquired from off-the-shelf keypoint detectors. This approach is a simple baseline to evaluate (quasi-)dense descriptors.

The trained network is then run on different resolutions of the input image in the form of multi-scale inference, yielding better descriptors. We scale the input image with values ranging from 0.25 to 2.5 at an interval of 0.25. The output of multi-scale images are fused into a single output feature descriptor by also using bilinear interpolation and element-wise addition, maintaining the same output descriptor size. We therefore obtain more semantically dense descriptors with a wider diversity of information.

4.2. Datasets and evaluation metrics

To verify the effectiveness of the proposed local descriptors, we evaluate them on two different types of benchmarks: semantic correspondence [15, 16, 42] and wide-baseline matching [1, 58].

PF-PASCAL [16] is composed of 1,351 image pairs and 20 object categories. Each image pair contains distinct object instances with varying appearances. Manually annotated keypoints exist for every pair of images. Following the settings of [54], we use 2,941 training pairs, 309 valida-

Methods	Backbone	Size of desc.	PF-PASCAL				PF-WILLOW			SPair-71k
			τ_{img}			τ_{bbox}	τ_{bbox}			τ_{bbox}
			0.05	0.1	0.15	0.1	0.05	0.1	0.15	0.1
UCN [5]	GoogLeNet	64	29.9	55.6	74.0	-	24.1	54.0	66.5	-
PF [15]	HOG	-	31.4	62.5	79.5	45.0	28.4	56.8	68.2	-
SCNet [17]	VGG-16	2048	36.2	72.2	82.0	48.2	38.6	70.4	85.3	-
A2Net [61]	ResNet-101	512	42.8	70.8	83.3	67.0	36.3	68.8	84.4	20.1
Cnngeo [53, 54]	ResNet-101	1024	49.0	74.8	84.0	72.0	37.0	70.2	79.9	21.1
RTNs [28]	ResNet-101	1024	55.2	75.9	85.2	-	41.3	71.9	86.2	-
NC-Net [55]	ResNet-101	1024	54.3	78.9	86.0	70.0	33.8	67.0	83.7	26.4
SF-Net [30]	ResNet-101	3072	53.6	81.9	90.6	78.7	46.3	74.0	84.2	-
DCC-Net [24]	ResNet-101	1024	55.6	82.3	90.5	-	43.6	73.8	86.5	-
HPF [41]	ResNet-101	6400	60.1	84.8	92.7	78.5	45.9	74.4	85.6	28.2
COLD	ResNet-101	64	67.8	84.2	90.5	81.1	42.4	67.6	80.7	-
	ResNet-101	128	71.2	86.8	92.1	84.2	46.0	70.9	82.4	28.4
	ResNet-101	512	73.3	88.2	92.9	85.1	49.0	73.5	85.6	30.2
	ResNet-152	512	**76.6**	**88.8**	**93.3**	**86.3**	**51.4**	**75.8**	**87.0**	**31.6**

Table 1. Performance on standard semantic correspondence benchmarks [15, 16] with PCK as the evaluation metric.

tion pairs, and 300 test pairs. Training pairs are augmented by flip operation and source-target inversion. The validation split is used to apply early stopping. The probability of correct keypoints (PCK) of an image pair given ground-truth keypoint correspondences $\mathcal{K}$ is defined as follows:

$$\text{PCK}(\mathcal{K}) = \frac{1}{|\mathcal{K}|} \sum_{(\mathbf{k},\mathbf{k}')\in\mathcal{K}} \mathbb{1}[\|\mathcal{T}(\mathbf{k}) - \mathbf{k}'\| < \tau_\alpha \max(h_\alpha, w_\alpha)] \tag{5}$$

where $\mathbb{1}$ is a general indicator function and τ_α is tolerance factor with $\alpha \in \{\text{bbox}, \text{img}\}$.

PF-WILLOW [15] is composed of 900 image pairs from 100 images. Image pairs have four subset categories: car, duck, motorbike, and wine bottle. Upon semantic correspondence evaluation, there are 10 semantic keypoints per image. Each image has background clutter and intra-class variation. We measure PCK with $\tau_{\text{bbox}} \in \{0.05, 0.1, 0.15\}$.

SPair-71k [42] test split is composed of 12,234 image pairs. This dataset contains more challenging pairs of images with higher viewpoint and scale changes, with more truncation and occlusion compared to PF-PASCAL [16] and PF-WILLOW [15]. We use PCK threshold τ to 0.1 at bounding-box-level.

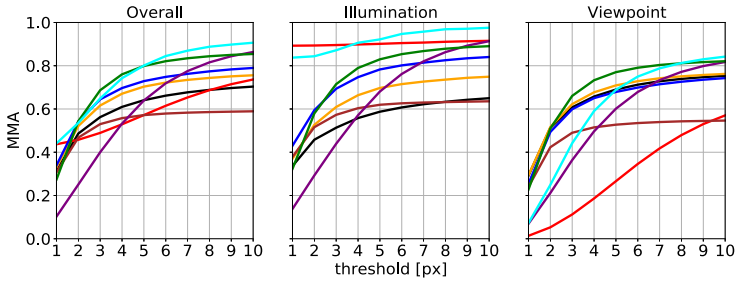
HPatches [1] is a wide-baseline image matching benchmark composed of 648 images from 108 scenes which have 6 images each. It has two main scene categories: 52 scenes with illumination variation and 56 scenes with viewpoint variation. 5 image pairs are for each scene containing six images - the first image against all other images. This results in a total of 540 pairs of images to be evaluated. The mean matching accuracy (MMA) is used as the metric, as in [11].

Aachen day-night [58, 59] is a long-term visual localization benchmark composed of 98 night-time queries with up to 20 relevant day-time images with known camera poses. The night-time images are created using software HDR to obtain high-quality, well-illuminated images. To evaluate the pose accuracy of night-time queries, we follow the evaluation protocol proposed in [58] using SfM pipeline [60]. Three thresholds were used for evaluation: high-precision (0.25m, 2°), mid-precision (0.5m, 5°), and coarse-precision (5m, 10°). All the scores in Table 2 are taken from official benchmark website[1].

4.3. Results

We evaluate our model on semantic correspondence [15, 16, 41], wide-baseline matching [1], and long-term visual localization [58]. Table 1 shows the comparison of recent methods in terms of matching accuracy (PCK) on the standard semantic correspondence benchmarks [15, 16]. The results show that despite having the smallest descriptor size, our model performs the best as illustrated in Table 1 on both datasets for all evaluation criteria. In particular, our model outperforms our baseline model, HPF [41], by up to 16.5%p with 12.5 times fewer size of descriptors on PF-PASCAL high-precision ($\tau_{\text{img}} = 0.05$). This indicates that our descriptors are robust to semantic changes, yet requires minimal memory requirements. Last column of Table 1 shows the PCK achieved by various models compared against ours on the SPair-71k [42] dataset. As SPair-71k is a challenging dataset where feature robustness is crucial to achieve high performances, it can be concluded that our method ex-

[1] https://www.visuallocalization.net/benchmark/



Method	# Features	# Matches
Hes. Det. RootSIFT	6.7K	2.8K
HAN + HN++	3.9K	2.0K
DELF	4.6K	1.9K
Superpoint	1.7K	0.9K
LF-Net	0.5K	0.2K
D2-Net Trained MS	8.3K	2.8K
R2D2	5.0K	1.8K
COLD (ours)	25.6K	9.9K

Figure 3. Evaluation on the image pairs of HPatches [1]. The mean matching accuracy (MMA) by the matching threshold in pixels is used as the evaluation metric. The right table denotes the mean number of local features and the mean number of mutual nearest neighbour matches.

Method	0.5m, 2°	1m, 5°	5m, 10°
RootSIFT [37]	54.1	66.3	75.5
HAN+HN [45]	58.2	72.4	83.7
Superpoint [10]	73.5	79.6	88.8
DELF [46]	54.1	75.5	96.9
D2-Net [11]	**74.5**	86.7	**100.0**
R2D2 [52]	**74.5**	83.7	**100.0**
COLD (ours)	72.4	**87.8**	**100.0**

Table 2. Evaluation on the visual localization benchmark [58]. We only evaluate on night time queries. The evaluation metric is localization accuracy by ground-truth camera pose.

tracts highly robust descriptors compared to other methods. We took all the SPair-71k scores from transferred models trained on PF-PASCAL as well.

Table 2 shows the localization accuracy achieved by various long-term visual localization models compared against ours on the Aachen day-night [58] dataset. While we exhibit on-par performance on the fine-precision (0.5m, 2°) and coarse precision (5m, 10°) thresholds, our model outperforms other models on the medium-precision (1m, 5°) threshold. While our model performs better than R2D2 by 3.1%p on the medium precision threshold, it exhibits a slight drop in performance in the high-precision threshold in comparison. This observation proves that due to the large receptive fields of represented descriptors in our model, our model can well interpret contexts of images - but has weaknesses in precise localization processes.

Figure 3 shows the wide-baseline matching results evaluated on the HPatches [1] dataset, using D2-Net [11] evaluation protocol. The left plot shows that our descriptors are especially robust to illumination changes on all thresholds, and also exhibits high robustness to viewpoint changes - just slightly behind R2D2 [52] on lower thresholds. In the viewpoint variation, our model performs better than the existing dense descriptors matching approaches, DELF [46] and D2-Net [11] at all thresholds. The weakness compared to the sparse matching model is viewpoint variation on high precision, but our model performs best after 9 pixel threshold at viewpoint variation. The table on the right shows that our grid keypoints strategy yields a lot of features and matches. With outlier rejection methods, *e.g.*, RANSAC [12] and its variations [2, 6, 7], our model expect to detect more inlier matches than the other models.

Although our model is trained using only the ground-truth correspondences obtained from this small semantic correspondence dataset [16], our descriptors yield highly competitive results on different domains as well, *e.g.*, visual localization benchmark [58] using a SfM pipeline [60]. The most comparable dense-feature extraction methods include D2-Net [11], which is trained on 327,036 image pairs from MegaDepth [32], and R2D2 [52], which is trained on approximately 12,000 synthetic image pairs from various sources [51, 58, 59]. On the contrary, with only 1,300 image pairs of PF-PASCAL [16], our model achieves comparable performance to D2-Net and R2D2 on both HPatches [1] and Aachen day-night [58].

Figure 4 shows selected qualitative results on [1, 58]. Our model can find image correspondences under strong variation in day-night (row 1) and rotation (row 2). Figure 5 shows selected qualitative comparison on [16, 42]. The correctness threshold is set to 20 pixels. Previous methods [24, 41, 55] fail in cases of large semantic variation, but our model finds correct keypoint matching results. More qualitative results are provided in the supplementary material.

4.4. Discussion

Generalizability. The datasets used in Table 2 and Figure 3, which are Aachen day-night benchmark [58] and HPatches [1] respectively, are on different domains from PF-PASCAL [16], our training data. Even so, our model performs comparably on both datasets [1, 58]. Table 2

Figure 4. Visualization of selected examples. Day/night variation and large viewpoint variation case in HPatches. Best viewed in electronic form.

FT.	FF.	Layers	Size of desc.	PF-PASCAL $\tau_{\text{img}} = 0.1$	HPatches MMA=5px
		N	2048	27.8	0.51
		1	64	6.6	0.35
✓		N	128	20.4	0.41
✓		1	128	3.5	0.19
	✓	$1:N$	2048	57.5	0.69
✓	✓	$1:\frac{N}{2}$	128	31.1	0.75
✓	✓	$\frac{N}{2}:N$	128	85.2	0.75
✓	✓	$1:N$	128	**86.8**	**0.77**

Table 3. Ablation study of feature transformation (FT) and feature fusion (FF). Third column shows the layers used in feature fusion, where $1:N$ denotes all layers, $1:\frac{N}{2}$ denotes first half layers (in this case, 1 to 17), $\frac{N}{2}:N$ denotes last half layers (in this case, 17 to 34), N denotes only the last layer, and 1 denotes only the first layer.

shows that our descriptors perform well on long-term visual localization [58] with integrated SfM pipeline [60], and partially outperform the previous dense descriptors extraction models [11, 52]. The last column of Table 3 displays the improvement in generalizability of our final proposed module against different ablations. The MMA scores at Table 3 are measured on 5 pixel threshold. These results show that our final proposed model yields compact local descriptors with high generalizability.

Ablation study of proposed modules. Table 3 shows the effect of each component in our model. The output of intermediate layers from the backbone network differ in descriptor size, and they cannot be fused by element-wise addition without feature transformation. Therefore, in those cases, we concatenate the smaller-channel layers to themselves to match the layer with the highest descriptor size to enable feature fusion. The fifth row of Table 3 shows an example: the resultant descriptor size is 2048, which was the largest descriptor size among the layers from the backbone network.

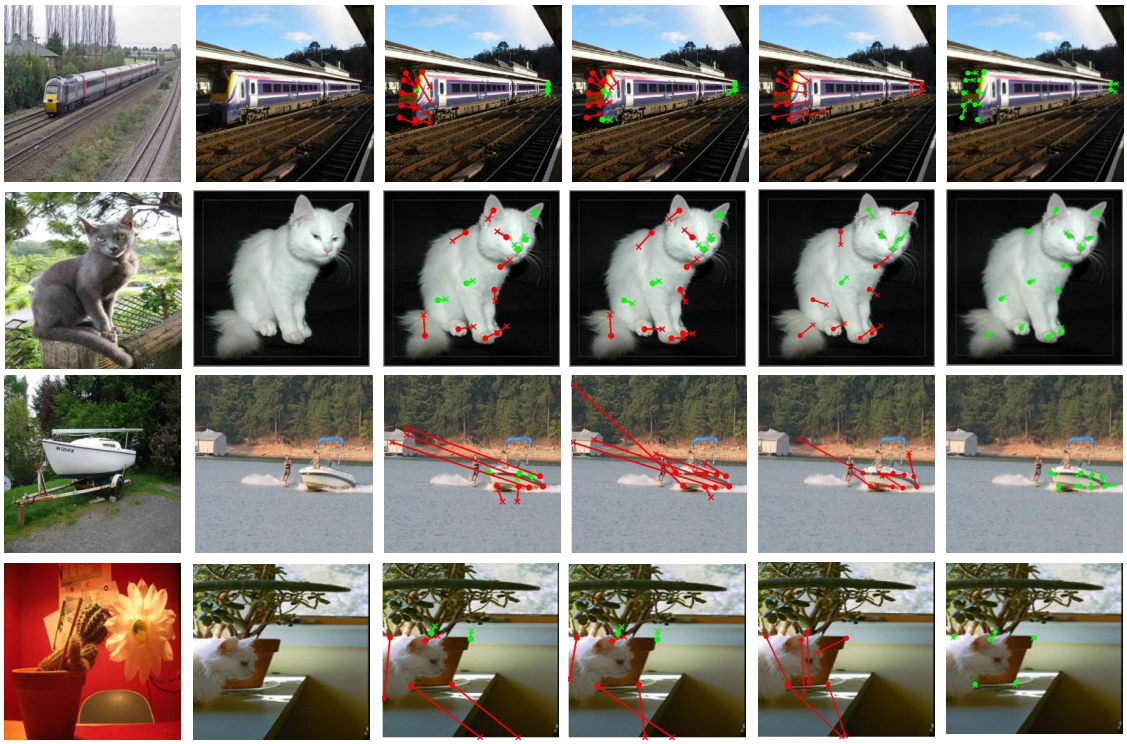
The sixth and seventh rows show the results of evaluation with partial layers. When we use shallower half of the layers (row 6), the results of semantic correspondence is detrimental, but the results of wide-baseline matching is quite comparable. When we use the deeper half of the layers (row 7), the results on both datasets are slightly lower than our proposed model. This is because high-level semantics, which come from deeper layers with larger receptive fields is important in semantic correspondence benchmark, PF-PASCAL. Interestingly, low-level geometric information from shallower layers with smaller receptive fields performs similarly on wide-baseline matching benchmark, HPatches, following to last column of sixth and seventh rows. Our final proposed design with feature transformation and feature fusion exhibits the best performance on both, PF-PASCAL [16] and HPatches [1] benchmark, with the small descriptor size of 128.

	sum	mul	max	concat
PF-PASCAL	**86.8**	25.0	48.1	76.5
PF-WILLOW	**70.9**	0.0	44.2	70.3
SPair-71k	28.4	0.1	12.7	**30.4**
Inference time	**35.2**	35.5	76.8	91.4

Table 4. PCK comparison ($\tau = 0.1$) of different aggregation on semantic correspondence. The unit of time is millisecond by per-pair inference time.

Ablation study of feature aggregation. Table 4 supports our choice of element-wise summation for feature aggregation through an ablation study on several semantic correspondence benchmarks, using ResNet-101 backbone with 34 bottleneck layers. Element-wise multiplication and max pooling demonstrate incomparably poor results. Feature concatenation shows competent performance, but it is approximately 3 times slower and results in descriptors which are 34 times bigger in size.

Integration with Superpoint keypoints detector [10]. Table 5 shows the results of integrating SuperPoint [10] keypoints detector. In this case, we upsampled our local descriptors using bilinear interpolation to match the spatial dimensions of the original input image. This was because we aimed to obtain pixel-level spatial precision to integrate the extracted keypoints directly. Thanks to [26], we could easily evaluate on the Phototourism [26, 22, 65] validation set. Compared to Superpoint [10] with their own proposed descriptors, our descriptors show improved performance on both stereo and multiview matching tasks.



(a) Source (b) Target (c) NCNet (d) DCCNet (e) HPF (f) Ours

Figure 5. Qualitative comparison of semantic correspondence benchmarks. Correct matches are colored as green and incorrect matches as red. o denotes target image keypoints and x denotes source image transferred keypoints. Our model finds correct matches under large intra-class variations such as illumination change (row 1), large deformation (row 2), scale change (row 3), and partial occlusion (row 4). Best viewed in electronic form.

Method	Dim of desc	Stereo	Multiview
		mAA(10°)	
SP [10]	256	0.3054	0.5316
SP + COLD	128	**0.3361**	**0.5705**

Table 5. Integration effect with Superpoint keypoints detector [10] under 2048 keypoints on Phototourism benchmark [26].

5. Conclusion

We have presented a multi-layer feature distillation network architecture to extract highly compact and robust local descriptors of 2D images. Our proposed model exploits intermediate layers of ImageNet pre-trained convolutional neural networks to encode the hierarchical features of the intermediate layers. The aggregation of these layers through element-wise addition yields our compact local descriptors, which can be used for various target tasks, namely semantic correspondence, wide-baseline matching, and visual localization. We demonstrate through our experiments that (1) we could find a successful compromise to the robustness-compactness trade-off, obtaining both compact and robust descriptors that outperform popular existing models, (2) the choice of element-wise summation in place of the conventional concatenation as our choice of feature aggregation improves our descriptors' robustness while resulting in a shorter inference time and higher compactness, and (3) our proposed feature transformation and multi-layer fusion cause our descriptors to exhibit notable generalizability - especially on matching tasks - demonstrating performances on semantic correspondence, wide-baseline matching, and visual localization tasks.

Acknowledgement This research was supported by Samsung Advanced Institute of Technology (NPRC), and also by Basic Science Research Program (NRF-2017R1E1A1A01077999) and Next-Generation Information Computing Development Program (NRF-2017M3C4A7069369), through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (MSIT). This work is also supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH)).

References

[1] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5173–5182, 2017.

[2] Daniel Barath, Jiri Matas, and Jana Noskova. Magsac: marginalizing sample consensus. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10197–10205, 2019.

[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.

[4] Minsu Cho, Suha Kwak, Cordelia Schmid, and Jean Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1201–1210, 2015.

[5] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. In *Proc. Neural Information Processing Systems (NeurIPS)*, pages 2414–2422, 2016.

[6] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Joint Pattern Recognition Symposium*, pages 236–243. Springer, 2003.

[7] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 772–779. IEEE, 2005.

[8] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018.

[11] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*, 2019.

[12] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[13] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002.

[14] Hugo Germain, Guillaume Bourmaud, and Vincent Lepetit. Sparse-to-dense hypercolumn matching for long-term visual localization. In *2019 International Conference on 3D Vision (3DV)*, pages 513–523. IEEE, 2019.

[15] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3475–3484, 2016.

[16] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(7):1711–1725, 2018.

[17] Kai Han, Rafael S Rezende, Bumsub Ham, Kwan-Yee K Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. Scnet: Learning semantic correspondence. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017.

[18] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[19] Richard I Hartley and Peter Sturm. Triangulation. *Computer vision and image understanding*, 68(2):146–157, 1997.

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Jared Heinly, Johannes L Schonberger, Enrique Dunn, and Jan-Michael Frahm. Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3287–3295, 2015.

[23] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[24] Shuaiyi Huang, Qiuyue Wang, Songyang Zhang, Shipeng Yan, and Xuming He. Dynamic context correspondence network for semantic alignment. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

[25] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[26] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *arXiv preprint arXiv:2003.01587*, 2020.

[27] Lai Kang, Lingda Wu, and Yee-Hong Yang. Robust multi-view l2 triangulation via optimal inlier selection and 3d structure refinement. *Pattern Recognition*, 47(9):2974–2992, 2014.

[28] Seungryong Kim, Stephen Lin, SANG RYUL JEON, Dongbo Min, and Kwanghoon Sohn. Recurrent transformer networks for semantic correspondence. In *Proc. Neural Information Processing Systems (NeurIPS)*, pages 6126–6136, 2018.

[29] Seungryong Kim, Dongbo Min, Bumsub Ham, Sangryul Jeon, Stephen Lin, and Kwanghoon Sohn. Fcss: Fully convolutional self-similarity for dense semantic correspondence.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6560–6569, 2017.

[30] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsub Ham. Sfnet: Learning object-aware semantic correspondence. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[31] Stefan Leutenegger, Margarita Chli, and Roland Y Siegwart. Brisk: Binary robust invariant scalable keypoints. In *2011 International conference on computer vision*, pages 2548–2555. Ieee, 2011.

[32] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018.

[33] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[34] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

[35] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[36] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[37] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[38] Zixin Luo, Tianwei Shen, Lei Zhou, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Contextdesc: Local descriptor augmentation with cross-modality context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2527–2536, 2019.

[39] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 168–183, 2018.

[40] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

[41] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2019.

[42] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019.

[43] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to compose hypercolumns for visual correspondence. In *ECCV 2020-16th European Conference on Computer Vision*, 2020.

[44] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017.

[45] Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–300, 2018.

[46] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision*, pages 3456–3465, 2017.

[47] David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Self-supervised learning of geometrically stable features through probabilistic introspection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3637–3645, 2018.

[48] David Novotny, Diane Larlus, and Andrea Vedaldi. Anchornet: A weakly supervised network to learn geometry-sensitive features for semantic matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5277–5286, 2017.

[49] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *Advances in neural information processing systems*, pages 6234–6244, 2018.

[50] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019.

[51] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5706–5715, 2018.

[52] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: Repeatable and reliable detector and descriptor. *arXiv preprint arXiv:1906.06195*, 2019.

[53] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. Convolutional neural network architecture for geometric matching. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[54] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[55] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *Proc. Neural Information Processing Systems (NeurIPS)*, pages 1656–1667, 2018.

[56] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[57] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011*

International conference on computer vision, pages 2564–2571. Ieee, 2011.

[58] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018.

[59] Torsten Sattler, Tobias Weyand, Bastian Leibe, and Leif Kobbelt. Image retrieval for image-based localization revisited. In *BMVC*, 2012.

[60] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[61] Paul Hongsuck Seo, Jongmin Lee, Deunsol Jung, Bohyung Han, and Minsu Cho. Attentive semantic alignment with offset-aware correlation kernels. In *Proc. European Conference on Computer Vision (ECCV)*, 2018.

[62] Xuelun Shen, Cheng Wang, Xin Li, Zenglei Yu, Jonathan Li, Chenglu Wen, Ming Cheng, and Zijian He. Rf-net: An end-to-end image matching network based on receptive field. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8132–8140, 2019.

[63] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[64] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. *arXiv preprint arXiv:1911.09070*, 2019.

[65] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.

[66] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11016–11025, 2019.

[67] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.