

Self-Supervised Learning of Image Scale and Orientation - supplementary material -

This supplementary material consists of four parts. Section 1 describes a more detailed process of our PatchPose dataset generation. Section 2 shows our scale and orientation estimation results on the image matching tasks and in-depth analysis. Section 3 analyzes the prediction range of scale and orientation compared to the existing methods. Section 4 shows various qualitative results to verify our model.

1 PatchPose dataset generation

We generate our PatchPose dataset, extracted from 1,793 images of SPair-71k [1] from PASCAL-VOC [2]. The PatchPose dataset consists of two parts, PatchPose-A and PatchPose-B. Figure 1 shows to generate an example of the PatchPose, $\mathcal{D} = \{(I_n, I'_n, \Delta_n)\}_{n=1}^N$, where Δ_n denotes the ground-truth relative pose from I_n to I'_n .

Dataset specification. The PatchPose-A dataset contains all combinations of scale and orientation shifts over 36 rotation and 13 scale values. We first produce 2,517,372 patches and then prune them as described in the next subsection of this supplementary material. After pruning, we obtain 2,013,804 patches, which make pairs with their original patches from the original images. We split the PatchPose-A dataset into 1,973,527 training pairs, 20,138 validation pairs, and 20,139 test pairs, whose split ratio is train : val: test = 98 : 1 : 1. The PatchPose-A dataset has $\frac{1}{3}$ interval of \log_2 -scale in the range of $[-2, 2]$ and $\frac{1}{18}\pi$ interval of orientation in the range of $[0, 2\pi)$, covering wide ranges of scale and orientation changes. In the PatchPose-A dataset, the orientation value is fixed when the scale value varies and vice versa. The PatchPose-B dataset is designed to evaluate the robustness under simultaneous and continuous changes of scale and orientation. Unlike the PatchPose-A, the scale and

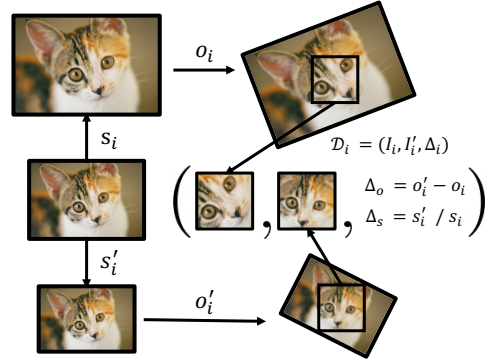


Figure 1: Illustration of training data generation. Each entry of \mathcal{D} stores two cropped patches extracted about the same keypoint from differently augmented versions of an image and their ground-truth scale/orientation differences.

orientation of an image patch are simultaneously transformed by random value in $[\frac{1}{4}, 4]$ and $[0, 2\pi)$. The PatchPose-B contains the same number of patches as the PatchPose-A, which are randomly transformed by $\Delta_s, \Delta_o \in \mathbb{R}$. Finally, we make the PatchPose dataset to merge PatchPose-A and -B, which dataset split is train: val: test = 3,947,054 : 40,276 : 40,278.

Dataset pruning. Among the generated image patches, there are ambiguous patches that do not have distinct patterns for characteristic scale and orientation. We find that those ambiguous samples often distract the learning process.

In order to prune the scale- and rotation-agnostic examples, we adopt an instance discrimination scheme inspired by [8]. We measure the average standard deviation σ of the transformed feature vectors $g(\cdot) \in \mathbb{R}^N$:

$$\bar{g}(I) = \frac{1}{|\mathcal{A}|} \sum_{(s,o) \in \mathcal{A}} g(\mathcal{T}_{s,o}(I)), \quad (1)$$

$$\sigma^2(I) = \sum_{i=1}^N \sum_{(s,o) \in \mathcal{A}} (g(\mathcal{T}_{s,o}(I))_i - \bar{g}(I)_i)^2, \quad (2)$$

where \mathcal{A} is a set of all possible pairs using 13 scaling and 36 rotating, *i.e.*, $\mathcal{A} = \{(s_i, o_i)\}_{i=1}^{468}$ and I is the transformed input patches. g is a function that forwards to CNN to generate features from the input image. $\bar{g}(\cdot)$ denotes the mean feature vector from the set of transformed feature vectors.

The standard deviation σ means how the sample can be discriminated by scale and rotation. For feature extraction of pruning, we use a ResNet-18 [9] model pre-trained on ImageNet [9] to extract features before the last fully connected layer. We aim to filter the unrelated samples with scale/orientation variation (e.g., textureless, round shape), to focus on learning with clear examples. Finally, we prune 20% of patches with low discriminability. Figure 2 illustrates the discriminability of feature vectors and several examples of acquired and pruned patches. Most of the pruned patches are less discriminative about scale and rotation compared to the acquired patches.

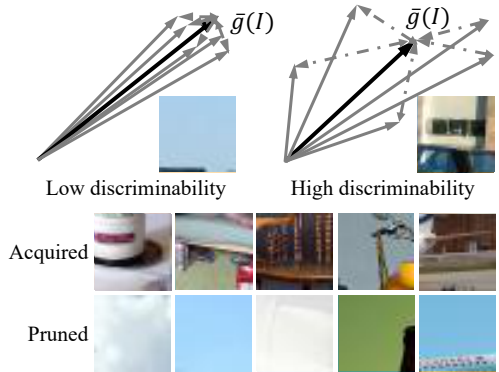


Figure 2: Illustration of the instance discrimination scheme. The left side of the top illustrates an example of a pattern-less patch with low discriminability. The source vectors of transformed patches have fewer separations from their mean vector \bar{g} . The right side of the top illustrates a patch with high discriminability. The source vectors are sufficiently separated from the mean vector \bar{g} .

2 Additional analysis on image matching

We experiment with our scale and orientation estimation on image matching methods for in-depth analysis. We use the HPatches [10] benchmark, and the evaluation scheme is the same as Section 4.2 and 4.4 in the main paper. We add more baselines [2, 7, 11, 12, 13] and evaluate additional thresholds and report separated results of the illumination and viewpoint variations. To evaluate the effect of our method, we replace the scale and orientation values in the image matching pipelines [2, 7, 11, 12, 13] at the patch extraction stage. We use all the pre-trained models and source codes released by the authors.

Table 1 summarizes the results on the HPatches dataset. For keypoint detectors, we use SIFT [2] and Key.Net [2]; we use HardNet [13] for descriptor extraction. Additionally, we use detection-then-description image matching methods LF-Net [13] and RF-Net [13]. In addition to Figure 4 in the main paper, we report the lower/upper bounds of patch extraction. Column ‘Det.’ denotes the keypoints detection methods, and ‘Des.’ denotes descriptor extraction methods, and ‘Pose.’ denotes patch extraction methods. The row with ‘ours’ denotes the results with argmax selection. The row with ‘ours (top-k)’ denotes top-k candidates selection on our output histogram as scale and orientation values. To measure the lower bound, we use identity matrix to extract patches centered on the keypoints as denoted the row with ‘lower’. To measure the upper bound, we use the ground-truth homography matrix to extract patches centered on the keypoints as denoted the row with ‘upper’.

Our model consistently performs better than the baseline methods on the mean matching accuracy (MMA) of the overall sequences and the viewpoint variation sequences. In the illumination variations, RF-Net [13] and Key.Net+HardNet [2, 13] show robust results on several thresholds. This is because RF-Net [13] trains their model using the HPatches dataset, and Key.Net [2] uses synthetic training data to consider photometric variations. In the viewpoint variations, our model consistently performs better than the existing methods in the same settings that all the methods are trained to consider the geometric variations at training time.

Upper bound and lower bound. Row ‘lower’ and ‘upper’ denote the lower bound and upper bound accuracies of patch extraction on the image matching pipeline. We measure the lower bound using no pose extraction at all, *i.e.*, patch sampling by an identity matrix, and the upper bound using ground-truth pose values to evaluate the test oracle of patch extraction on image matching pipelines. The results imply that there is still room for improvement as the upper bound (*i.e.* using ground-truth pose values for pose extraction) yields the best results, motivating further research in this aspect.

3 Analysis on prediction ranges.

A plot of prediction range. Figure 3 plots the predicted scale and orientation of different models on the PatchPose dataset, where we can observe the ranges of predicted scale and orientation. We obtain the scale value of SIFT [2] using the scale-space maxima and octave index, where a single octave-level difference is the same to enlarge twice the image size. We obtain the orientation value of SIFT [2] using the dominant direction of the histogram bin. We use the scale and orientation values from the output of LF-Net [13] and RF-Net [13] keypoints detector. In the case of our model, we convert histogram bins to the corresponding scale and orientation values by argmax selection. All the scale values are plotted in \log_2 scale. SIFT [2] and our model show a large range in both scale prediction and orientation prediction. In contrast, LF-Net [13] is limited to a small range in scale prediction, and RF-Net [13] is to a small range in both scale prediction and orientation predictions.

Evaluation of scale/orientation estimation under varying differences. Figure 4 shows the accuracy of patch pose estimation under varying differences in scale or orientation. Among the baselines, SIFT [2] makes relatively accurate predictions across various ranges of scale/orientation differences. On the other hand, the existing learning-based methods [13, 13] exhibits high accuracy only in specific ranges of scale/orientation differences. As can be in Figure 3, this problem occurs due to the limited prediction range. This indicates that these models [13, 13] have limited power in predicting large changes in scale/orientation.

methods			K	M	MMA					
Det.	Des.	Pose.			Overall		Illumination		Viewpoint	
					3px	6px	3px	6px	3px	6px
SIFT [0]	HardNet [0]	lower	122.9	35.5	44.9	46.1	49.9	51.0	40.1	41.3
		SIFT		35.2	41.7	44.6	45.8	48.9	37.7	40.4
		ours		36.3	51.3	54.3	53.8	57.2	48.9	51.5
		ours (top-4)		36.7	54.4	57.7	57.3	61.1	51.7	54.5
		upper		38.3	65.5	69.2	60.7	64.6	70.2	73.7
		lower	487.6	151.3	47.8	49.1	53.0	54.5	42.7	43.8
		SIFT		148.9	45.2	49.0	50.1	54.5	40.5	43.8
		ours		154.6	57.5	61.4	60.2	64.7	54.9	58.1
		ours (top-4)		157.6	61.1	65.1	64.2	69.0	58.0	61.3
		upper		166.0	72.5	77.0	67.7	72.6	77.1	81.1
LF-Net [0]	LF-Net [0]	lower	128.0	45.2	27.9	28.5	37.1	37.8	19.1	19.6
		LF-Net		44.2	22.2	22.6	29.7	30.2	14.9	15.3
		ours		42.6	35.6	36.6	42.1	43.1	29.4	30.2
		ours (top-4)		44.3	44.4	45.6	51.6	52.8	37.4	38.6
		upper		46.1	60.9	63.0	60.1	62.1	61.7	63.8
		lower	512.0	164.4	27.0	27.9	35.7	36.7	18.6	19.3
		LF-Net		160.3	21.0	21.7	28.2	29.1	14.0	14.6
		ours		155.0	37.1	38.8	43.6	45.5	30.7	32.3
		ours (top-4)		168.5	46.6	48.7	54.0	56.2	39.4	41.4
		upper		184.6	66.4	69.6	64.4	67.4	68.3	71.7
RF-Net [0]	RF-Net [0]	lower	127.6	59.7	27.7	29.5	38.0	39.9	17.9	19.5
		RF-Net		70.4	35.5	45.1	52.6	64.3	19.1	26.5
		ours		66.5	41.6	50.6	49.9	57.3	33.6	44.1
		ours (top-4)		66.7	45.0	55.7	53.5	62.5	36.7	49.2
		upper		65.2	48.9	61.1	55.2	64.9	42.9	57.5
		lower	510.4	217.2	27.9	29.3	38.8	40.3	17.4	18.6
		RF-Net		235.5	37.6	47.8	56.8	68.3	19.1	28.1
		ours		226.7	45.7	54.9	54.2	61.2	37.5	48.8
		ours (top-4)		230.9	49.6	60.0	58.3	66.1	41.2	54.1
		upper		225.1	55.3	67.0	60.5	69.0	50.4	65.1
Key.Net [0]	HardNet [0]	lower	125.5	46.5	55.7	57.1	62.8	64.0	48.8	50.3
		Key.Net		58.9	70.6	74.3	73.6	77.3	67.6	71.4
		ours		55.6	66.2	69.5	68.9	72.0	63.7	67.0
		ours (top-9)		58.9	71.1	74.9	73.2	76.8	69.0	73.1
		upper		47.6	75.6	78.3	72.3	74.9	78.7	81.5
		lower	503.9	177.9	54.7	56.1	60.7	62.0	48.9	50.3
		Key.Net		241.5	73.0	77.9	74.5	80.0	71.5	75.9
		ours		221.5	68.6	72.8	69.4	73.8	67.9	71.8
		ours (top-9)		240.7	73.3	78.2	74.0	79.2	72.7	77.2
		upper		187.2	78.4	81.7	74.2	77.4	82.5	85.8

Table 1: Additional results with off-the-shelf keypoints detectors and descriptors on the HPatches. Column ‘K’ denotes the number of extracted keypoints, and ‘M’ denotes the average number of predicted matches. Row with ‘ours’ means argmax selection, and row with ‘ours (top-k)’ is k multiple candidates extraction on our histogram representation. Row with ‘lower’ means lower bound accuracy with no patch alignment, identity patch sampling. Row with ‘upper’ means upper bound accuracy with patch extraction using ground-truth homography matrix.

In contrast, our model makes more diverse and more accurate predictions than the other methods.

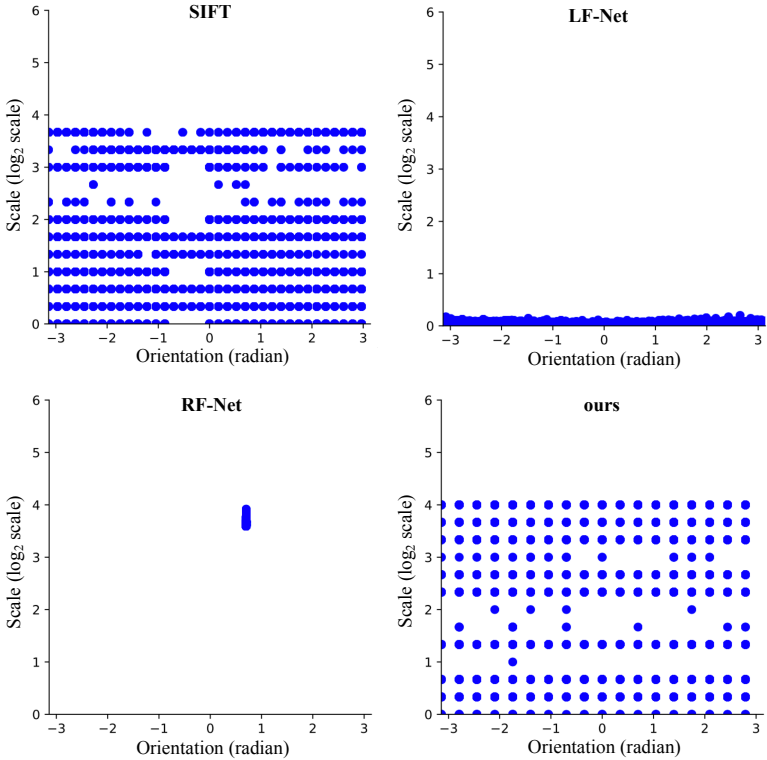


Figure 3: Plot of scale and orientation predictions on the PatchPose dataset. The objective of this experiment is to show the limitation of the range of scale and orientation predicted by each model. SIFT [1] and our model have a large range in scale and orientation prediction, but LF-Net [12] has a restricted range in scale prediction, and RF-Net [13] has a small range in scale and orientation prediction.

4 More qualitative results

We visualize the scale and orientation estimation on example cases with large pose differences in Figure 5. Each model is tested on the local image patches centered on the circles shown. Compared to the other methods, our method predicts the scale and orientation of the local region more accurately. In these challenging cases with extreme differences, the previous learning-based methods, LF-Net [12] and RF-Net [13], fail to estimate large scale/orientation changes accurately, which are rarely observed in their training. The hand-crafted method, SIFT [1], performs relatively better than the learning-based methods.

Figure 5 shows qualitative examples by different ranges. We use two more rotation angles for the experiment: $\frac{5\pi}{18}$, $\frac{7\pi}{9}$ and $\frac{16\pi}{9}$. We first rotate each image using the target angle, then enlarge the image to be 2.52 times the original size. Then, we search for the corresponding locations of sampled points in the transformed image to predict the characteristic scale and orientation of corresponding locations using each method [1, 12, 13]. We compare the predicted pose differences with the ground truth pose differences and visualize them with

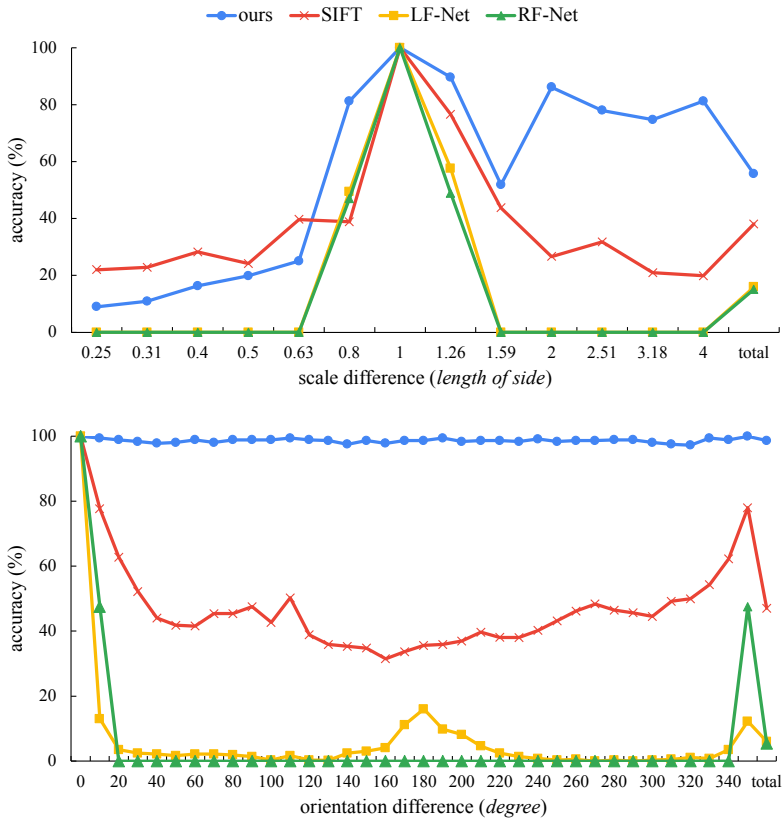


Figure 4: Range-wise evaluation of scale estimation (upper) and orientation estimation (be-low) on the PatchPose dataset. We evaluate the accuracy of scale and orientation estimation by each difference level. We set the accuracy threshold $\pm 2^{1/3}$ and $\pm 10^\circ$, scale and orienta-tion, respectively. The value of the last index ‘total’ indicates the average of the accuracy.

colored circles and lines. These examples show that the previous learning-based methods tend to learn a bias in their training setup, and thus often do not generalize well to different unseen cases. In contrast, due to self-supervised and explicit learning, our method performs very robustly to such a wide range of variations.

Figure 6 shows qualitative examples on zoom/rotated image pair [8, 9]. Compared to the existing models [0, 12, 13], our model shows better patch alignment results based on the predicted scale/orientation values.

Figure 7 visualizes examples of image matching on the HPatches dataset [10]. Our model generates more matches with better precision on the examples.

Figures 8 and 9 show example sequences of the PatchPose-A and the PatchPose-B. All the visualized patches are cropped by 64×64 size from the original images. PatchPose-A has grid-level scale/orientation variations, while the PatchPose-B has randomly selected, continuous scale/orientation variations.

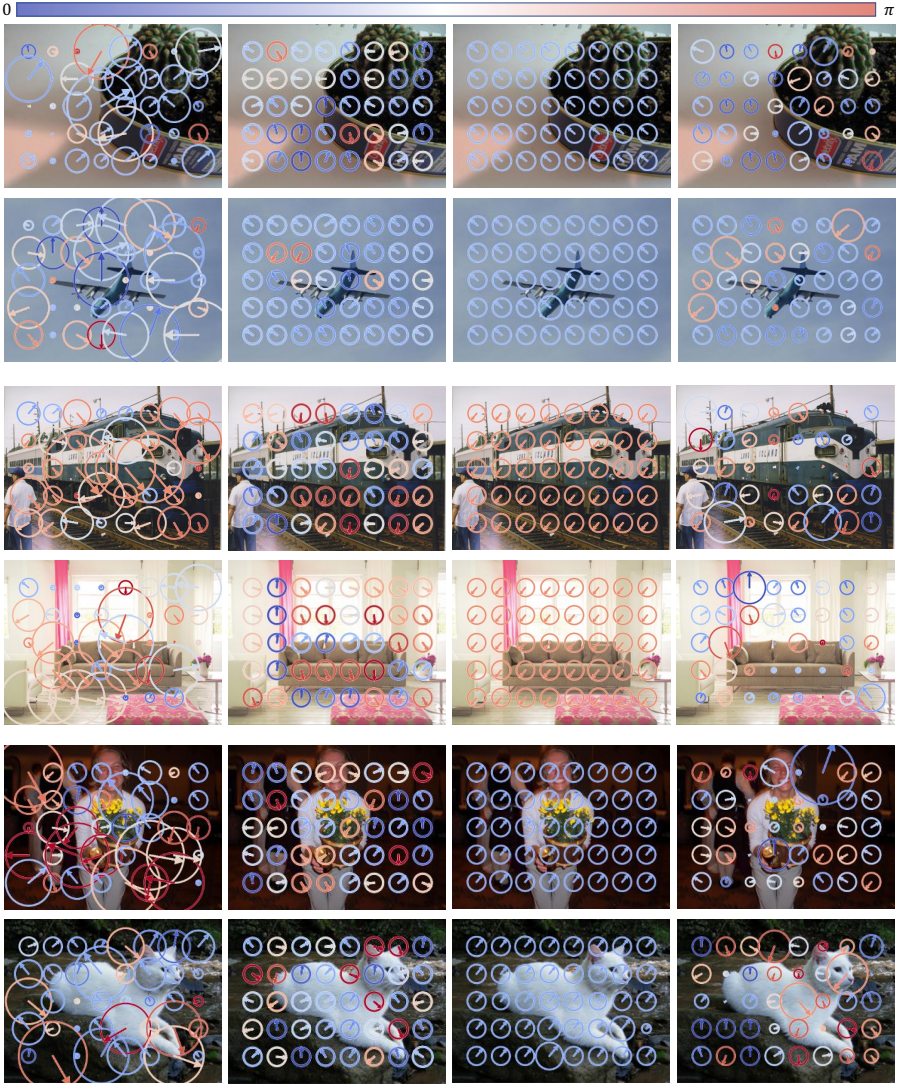


Figure 5: Visualization of scale and orientation prediction error using [10, 12, 13] and ours. For each image, we generate an image pair using 2.52 times upscale and 50° rotation (Row 1, 2), 140° rotation (Row 3, 4), and 320° rotation (Row 5, 6). We apply each method to the pair and estimate the difference of scale and orientation for corresponding regions. The line direction denotes the orientation error, *i.e.*, pointing upwards shows 100% accuracy while pointing downward shows 0% accuracy. For enhanced visibility, we add colors to the circles, as the error distribution at the top of the color bar. The redder the color, the less accurate the orientation predictions are; the bluer the color, the more accurate. The circle size represents the scale prediction error, where the larger circle size denotes the larger prediction error.

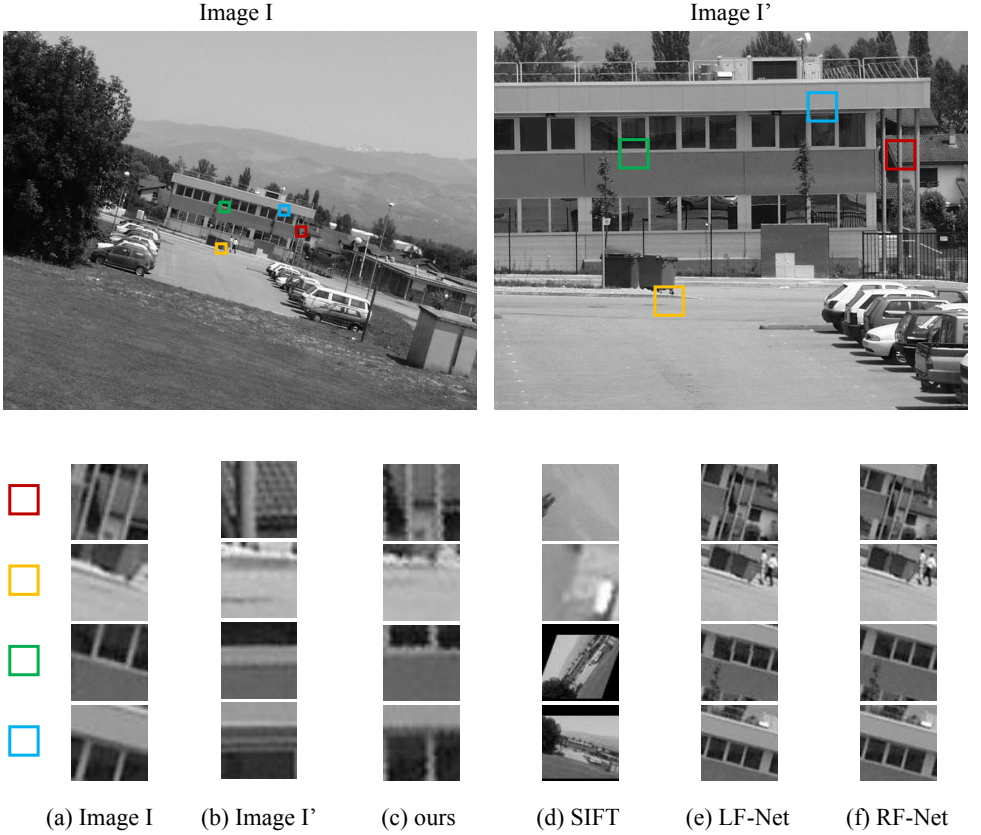


Figure 6: Selected visualization of patch pose estimation on zoom/rotated image pair [8, 9]. The colors denote the corresponding patches on image I and I' . Columns (a) and (b) show the cropped patches from the input image I and I' . Columns (c), (d), (e), and (f) show the aligned patches from I to I' using the predicted scale/orientation values of each model. In this dataset, the ground-truth homography has not only scale and orientation factors but also other perspective transformation parameters (*e.g.*, shearing, tilting). Therefore, the patches may not be perfectly aligned from I to I' by only using scale and orientation factors. Nevertheless, our model consistently aligns several patches better than the other models.

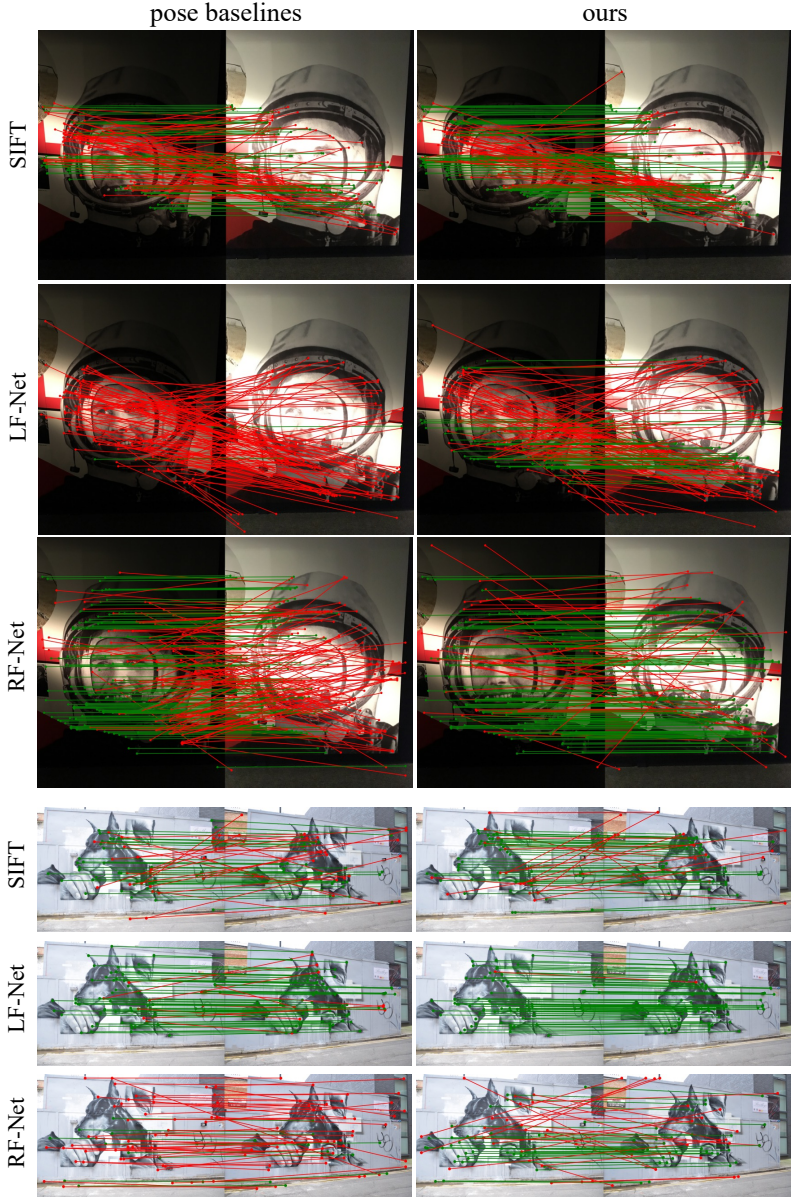


Figure 7: Visualization of image matching using patch extraction on the HPatches [10]. The circles are keypoints detected by each method, and the lines denote predicted matches to satisfy mutual nearest neighbour. The color denotes correctness by ground-truth homography, *i.e.*, green color is a correct, red color is an incorrect match. We set the threshold of correctness as 10 pixels. The left side image pairs denote image matching results using the internal pose estimator of each method. The right side image pairs denote image matching results to replace the pose estimation results with our pose estimation results.

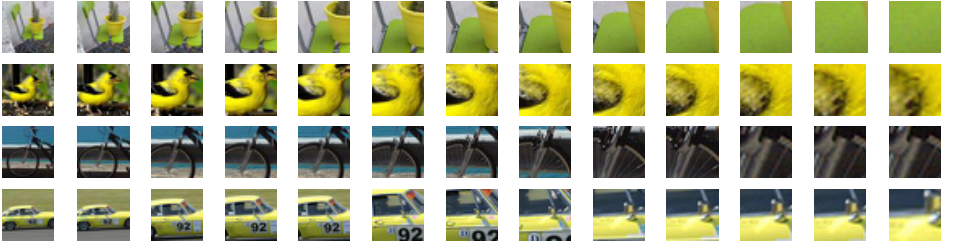


Figure 8: Visualization of scale variation examples of the PatchPose-A (row 1, 2) and the PatchPose-B (row 3, 4). From left to right, the columns show the resized patches which are scaled at factors from 2^{-2} to 2^2 at an interval of $2^{\frac{1}{3}}$ about rows 1 and 2. The patches of rows 3 and 4 unlock the interval restriction and randomly resized on the range of factors $[2^{-2}, 2^2]$.

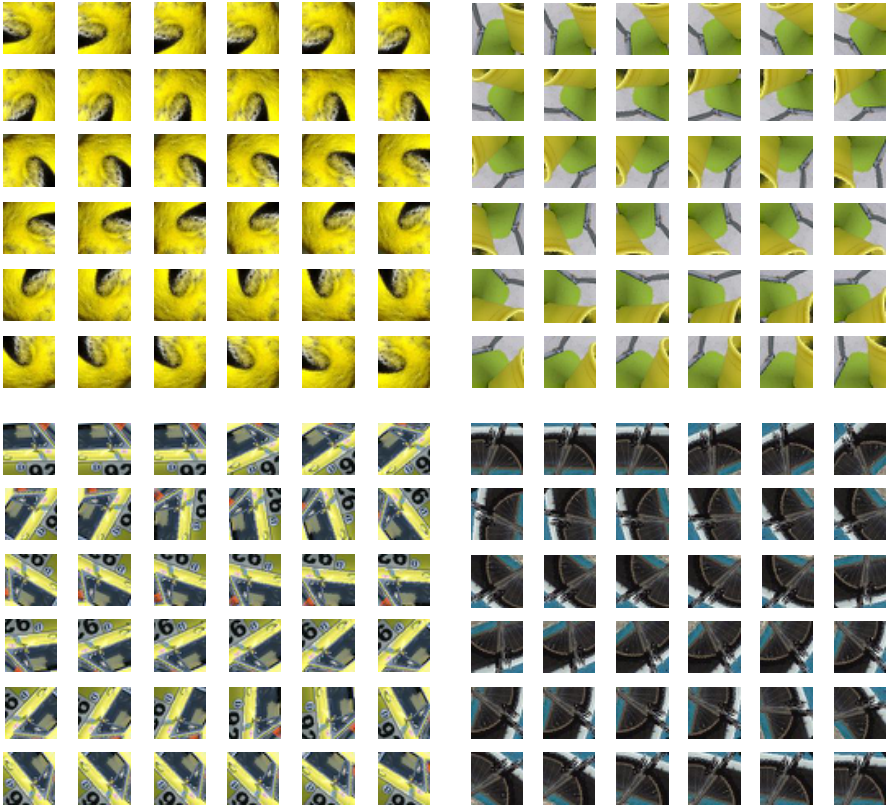


Figure 9: Visualization of orientation variation examples of the PatchPose-A (the top two 6x6 grids) and the PatchPose-B (the bottom two 6x6 grids). The top two grids show the rotated patches of PatchPose-A, at a rotation increasing anticlockwise from top left to bottom right at a $\frac{1}{18}\pi$ interval. The bottom two blocks show the rotated patches of PatchPose-B, at randomly generated rotation factors $0 \leq \Delta_o < 2\pi, \Delta_o \in \mathbb{R}$. We sort the sequences in an increasing order of degrees.

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] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key. net: Keypoint detection by handcrafted and learned cnn filters. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5836–5844, 2019.
- [3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- [4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- [5] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10364–10374, 2019.
- [6] 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.
- [7] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [8] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 27(10): 1615–1630, 2005.
- [9] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005.
- [10] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019.
- [11] 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.
- [12] 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.
- [13] 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.