Repeatability Is Not Enough: Learning Affine Regions via Discriminability

Dmytro Mishkin Filip Radenović Jiři Matas

Visual Recognition Group, Center for Machine Perception, FEE, CTU in Prague
{mishkdmy, filip.radenovic, matas}@cmp.felk.cvut.cz

Abstract. A method for learning local affine-covariant regions is presented. We show that maximizing geometric repeatability does not lead to local regions, a.k.a features, that are reliably matched and this necessitates descriptor-based learning. We explore factors that influence such learning and registration: the loss function, descriptor type, geometric parametrization and the trade-off between matchability and geometric accuracy and propose a novel hard negative-constant loss function for learning of affine regions. The affine shape estimator – AffNet – trained with the hard negative-constant loss outperforms the state-of-the-art in bag-of-words image retrieval and wide baseline stereo. The proposed training process does not require precisely geometrically aligned patches. The source codes and trained weights are available at https://github.com/ducha-aiki/affnet

Keywords: local features · affine shape · loss function · image retrieval

1 Introduction

Local features, forming correspondences, are exploited in state of the art pipelines for 3D reconstruction [1,2], two-view matching [3], 6DOF image localization [4]. Classical local features have also been successfully used for providing supervision for CNN-based image retrieval [5].

Affine-convariance [7] is a desirable property of local features since it allows robust matching of images separated by a wide baseline [8,3], unlike scale-covariant features like ORB [9] or difference of Gaussian (DoG) [10] that rely on tests carried out on circular neighborhoods. This is the reason why the Hessian-Affine detector [7] combined with the RootSIFT descriptor [10,11] is the gold standard for local feature in image retrieval [12,13]. Affine covariant features also provide stronger geometric constraints, e.g., for image rectification [14].

On the other hand, the classical affine adaptation procedure [15] fails in 20%-40% [8,16] cases, thus reducing the number and repeatability of detected local features. It is also not robust to significant illumination change [16]. Applications where the number of detected features is important, *e.g.*, large scale 3D reconstruction [2], therefore use the DoG detector. Alleviating the problem of the drop in the number of correspondences caused by the non-repeatability of the affine adaptation procedure, may lead to connected 3D reconstructions and improved image retrieval engines [17,20].

This paper makes four contributions towards robust estimation of the local affine shape. First, we experimentally show that geometric repeatability of a local feature is not a sufficient condition for successful matching. The learning of affine shape increases the number of corrected matches if it steers the estimators towards discriminative regions and therefore must involve optimization of a descriptor-related loss.

Second, we propose a novel loss function for descriptor-based registration and learning, named the *hard negative-constant loss*. It combines the advantages of the triplet and contrastive positive losses. Third, we propose a method for learning the affine shape, orientation and potentially other parameters related to geometric and appearance properties of local features. The learning method does not require a precise ground truth which reduces the need for manual annotation.

Last but not least, the learned AffNet itself significantly outperforms prior methods for affine shape estimation and improves the state of art in image retrieval by a large margin. Importantly, unlike the de-facto standard [15], AffNet does not significantly reduce the number of detected features, it is thus suitable even for pipelines where affine invariance is needed only occasionally.

1.1 Related work

The area of learning local features has been active recently, but the attention has focused dominantly on learning descriptors [21,22,23,24,25,26,27] and translation-covariant detectors [28,29,30,31]. The authors are not aware of any recent work on learning or improvement of local feature affine shape estimation. The most closely related work is thus the following.

Hartmann *et al.* [32] train random forest classifier for predicting feature matchability based on a local descriptor. "Bad" points are discarded, thus speeding up the matching process in a 3D reconstruction pipeline. Yi *et al.* [33] proposed to learn feature orientation by minimizing descriptor distance between positive patches, i.e. those corresponding to the same point on the 3D surface. This allows to avoid hand-picking a "canonical" orientation, thus learning the one which is the most suitable for descriptor matching. We have observed that direct application of the method [33] for affine shape estimation leads to learning degenerate shapes collapsed to single line. Yi *et al.* [34] proposed a multi-stage framework for learning the descriptor, orientation and translation-covariant detector. The detector was trained by maximizing the intersection-over-union and the reprojection error between corresponding regions.

Lenc and Vedaldi [30] introduced the “covariant constraint” for learning various types of local feature detectors. The proposed covariant loss is the Frobenius norm of the difference between the local affine frames. The disadvantage of such approach is that it could lead to features that are, while being repeatable, not necessarily suited for the matching task (see Section 2.2). On top of that, the common drawback of the Yi *et al.* [34] and Lenc and Vedaldi [30] methods is that they require to know the exact geometric relationship between patches which increases the amount of work needed to prepare the training dataset. Zhang *et al.* [29] proposed to “anchor” the detected features to some pre-defined features with known good discriminability like TILDE [28]. We remark that despite showing images of affine-covariant features, the results presented in the paper are for translation-covariant features only. Savinov *et al.* [31] proposed a ranking approach for unsupervised learning of a feature detector. While this is natural and efficient for learning the coordinates of the center of the feature, it is problematic to apply it for the affine shape estimation. The reason is that it requires sampling and scoring of many possible shapes.

Finally, Choy *et al*. [35] trained a "Universal correspondence network" (UCN) for a direct correspondence estimation with contrastive loss on a patch descriptor distance. This approach is related to the current work, yet the two methods differ in several important aspects. First, UCN used an ImageNet-pretrained network which is subsequently fine-tuned. We learn the affine shape estimation from scratch. Second, UCN uses dense feature extraction and negative examples extracted from the same image. While this could be a good setup for short baseline stereo, it does not work well for wide baseline, where affine features are usually sought. Finally, we propose the hard negative-constant loss instead of the contrastive one.

2 Learning affine shape and orientation

2.1 Affine shape parametrization

A local affine frame is defined by 6 parameters of the affine matrix. Two form a translation vector (x, y) which is given by the keypoint detector and in the rest of the paper we omit it and focus on the *affine transformation* matrix A,

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}. \tag{1}$$

Among many possible decompositions of matrix A, we use the following

$$A = \lambda R(\alpha) A' = \det A \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} a'_{11} & 0 \\ a'_{21} & a'_{22} \end{pmatrix}, \tag{2}$$

where $\lambda = \det A$ is the scale, $R(\alpha)$ the *orientation* matrix and A' [1] is the *affine shape* matrix with $\det A'$ = 1. A' is decomposed into identity matrix I and *residual shape* A'':

$$A' = I + A'' = \begin{pmatrix} a'_{11} & 0 \\ a'_{21} & a'_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} a''_{11} & 0 \\ a''_{21} & a''_{22} \end{pmatrix} \tag{3}$$

We show that the different parameterizations of the affine transformation significantly influence the performance of CNN-based estimators of local geometry, see Table 2.

2.2 The hard negative-constant loss

We propose a loss function called hard negative-constant loss (HardNegC). It is based on the hard negative triplet margin loss [25] (HardNeg), but the distance to the hardest

[1] A' has a (0,1) eigenvector, preserving the vertical direction.

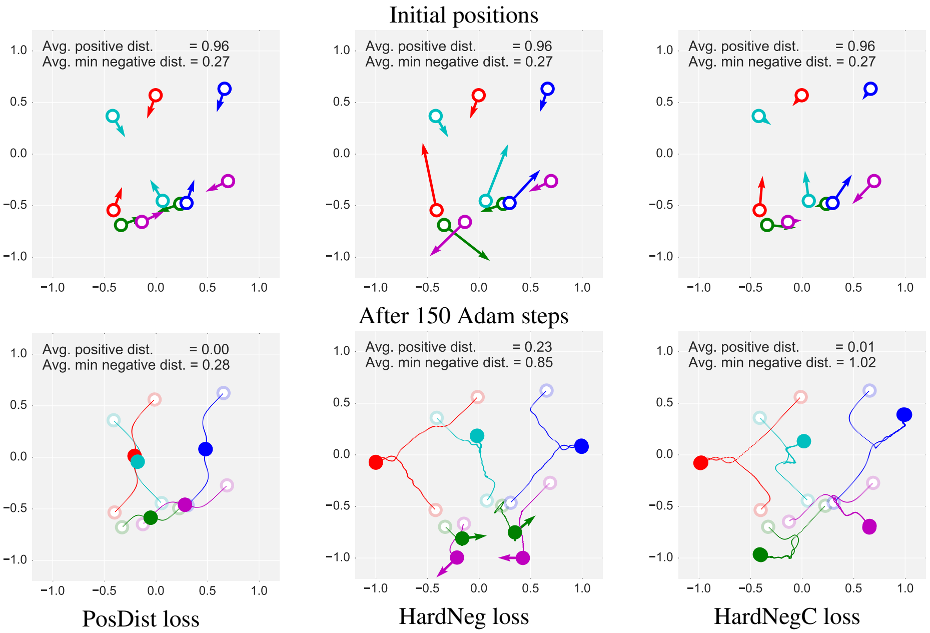


Fig. 1. A toy example optimization problem illustrating the proposed hard negative-constant (HardNegC) loss. Five pairs of points, representing 2D descriptors, are generated and the losses are minimized by Adam [36]: the positive descriptor distance (PosDist) [33] – left, the hard negative (HardNeg) margin loss [25] – center, HardNegC– right. Top row: identical initial positions of five pairs of matching points. Arrows show the gradient direction and relative magnitude. Bottom row: points after 150 steps of Adam optimization, trajectories are shown by dots. HardNeg loss has a difficulty with the green and magenta point pairs, because the negative example lies between two positives. Minimization of the positive distance only leads to a small distance to the negative examples. The proposed HardNegC loss first pushes same class points close to each other and then distributes them to increase distance to the negative pairs.

(i.e. closest) negative example is treated as constant and the respective derivative of L is set to zero:

$$L = \frac{1}{n} \sum_{i=1,n} \max\left(0, 1 + d(s_i, \dot{s}_i) - d(s_i, N)\right), \quad \frac{\partial L}{\partial N} := 0, \tag{4}$$

where $d(s_i, \dot{s}_i)$ is the distance between the matching descriptors, $d(s_i, N)$ is a distance to the hardest negative example N in the mini-batch for i^{th} pair.

$$d(s_i, N) = \min\left(\min_{j \neq i} d(s_i, \dot{s}_j), \min_{j \neq i} d(s_j, \dot{s}_i)\right)$$

The difference between the Positive descriptor distance loss (PosDist) used for learning local feature orientation in [33] and the HardNegC and HardNeg losses is shown on a toy example in Figure 1. Five pairs of points in the 2D space are generated and their positions are updated by the Adam optimizer [36] for the three loss functions. PosDist converges the first, but the different class points end up near each other, because the distance to the negative classes is not incorporated in the loss. The HardNeg margin loss has trouble when the points from different classes lie between each other. The HardNegC loss behavior first resembles the PosDist loss, bringing positive points together and then distributes them in the space, satisfying the triplet margin criterion.

2.3 Descriptor losses for shape registration

Exploring how local feature repeatability is connected with descriptor similarity, we conducted an shape registration experiment (Figure 2). Hessian features are detected in reference HSequences [37] illumination images and reprojected by (identity) homography to another image in the sequence. Thus, the repeatability is 1 and reprojection error is 0. Then, the local descriptors (HardNet [25], SIFT [10], TFeat [23] and raw pixels) are extracted and features are matched by first-to-second-nearest neighbor ratio [10] with threshold 0.8. This threshold was suggested by Lowe [10] as a good trade-off between false positives and false negatives. For SIFT, 22% of the geometrically correct correspondences are not the nearest SIFTs and they cannot be matched, regardless of the threshold. In our experiments, the 0.8 threshold worked well for all descriptors and we used it, in line with previous papers, in all experiments.

Notice that for all descriptors, the percentage of correct matches even for the *perfect* geometrical registration is only about 50%.

Adam optimizer is used to update affine region A to minimize the descriptor-based losses: PosDist, HardNeg and HardNegC. The top two rows show the results for A matrices coupled for both images, bottom – the descriptor difference optimization is allowed to deform A and $\dot{A}$ in both images independently, which leads to a pair of affine regions that are not in perfect geometric correspondence, yet they are more matchable. Note, that no training of any kind is involved.

Such descriptor-driven optimization, not maintaining perfect registration, produces a descriptor that is matched successfully up to 90% of the detections under illumination changes.

For most of the unmatched regions, the affine shapes become a degenerate lines – shown in top graphs, and the number of degenerate ellipses is high for PosDist loss; HardNeg and HardNegC perform better.

The bottom row of Figure 2 shows results for experiments where affine shapes pairs are independent in each image. Optimization of descriptor losses lead to an increase of the geometric error on the affine shape. Error E is defined as the mean square error on A matrix difference:

$$E = \sum_{i=1}^{n} \frac{2(A_i - \dot{A}_i)^2}{\det A + \det \dot{A}} \tag{5}$$

Again, PosDist loss leads to a larger error. CNN-based descriptors, HardNet and TFeat lead to relative small geometric error when reaching matchability plateau, while for SIFT and raw pixels the shapes diverge. Figure 3 shows the case when the initialized shapes include a small amount of the reprojection error.

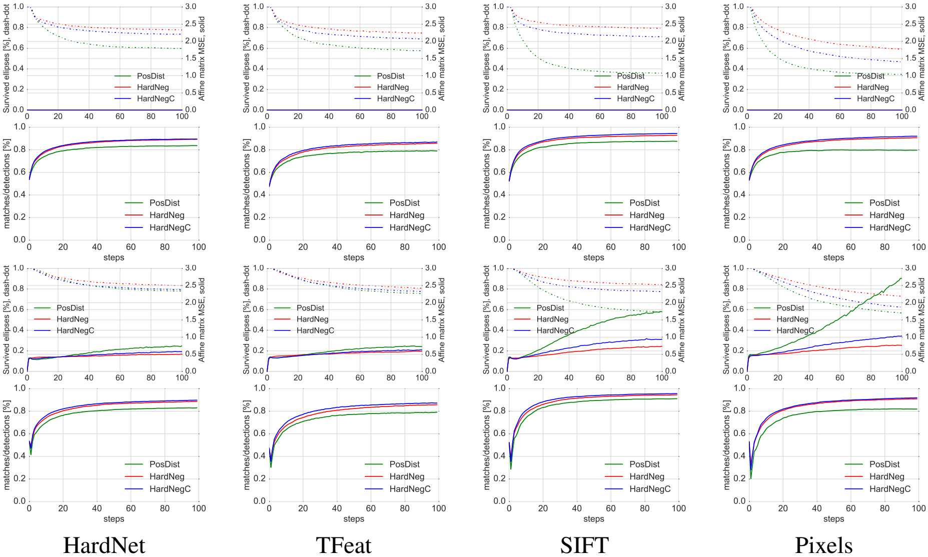
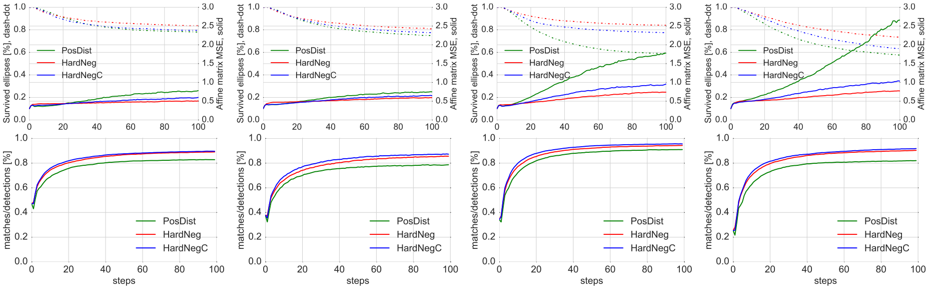


Fig. 2. Matching score versus geometric repeatability experiment. Affine shape registration by a minimization of descriptor losses of corresponding features. Descriptor losses: green – L2-descriptor distance (PosDist) [33], red – hard triplet margin HardNeg [25], blue – proposed HardNegC. Average over HSequences, illumination subset. *All features are initially perfectly registered*. First two rows: single feature geometry for both images, second two rows: feature geometries are independent in each image. Top row: geometric error of corresponding features (solid) and percentage of non-collapsed, i.e. elongation ≤ 6, features (dashed). Bottom row: the percentage of correct matches. This experiment shows that even perfectly initially registered feature might not be matched with any of descriptors – initial matching score is roughly $\approx 30..50\%$. But it is possibly to find measurement region, which offers both discriminativity and repeatability. PosDist loss squashes most of the features, leading to the largest geometrical error. HardNeg loss produces the best results in the number of survived feature and geometrical error. HardNegC performs slightly worse than HardNeg, slightly outperforming it on matching score. However, HardNegC is easier to optimize for AffNet learning – see Table 1.

2.4 AffNet training procedure

The main blocks of the proposed training procedure are shown in Figure 5. First, a batch of matching patch pairs $(P_i, \dot{P}_i)_{i=1..n}$ is generated, where P_i and $\dot{P}_i$ correspond to the same point on a 3D surface. Rotation and skew transformation matrices (T_i, T_i') are randomly and independently generated. The patches P_i and $\dot{P}_i$ are warped by $(T_i, \dot{P}_i)$ respectively into A-transformed patches. Then, a 32×32 center patch is cropped and a pair of transformed patches is fed into the convolutional neural network AffNet, which predicts a pair of affine transformations A_i, $\dot{P}_i$, that are applied to the T_i-transformed patches via spatial transformers ST [38].

Thus, geometrically normalized patches are cropped to 32×32 pixels and fed into the descriptor network, e.g. HardNet, SIFT or raw patch pixels, obtaining descriptors



HardNet TFeat SIFT Pixels

Fig. 3. Minimization of descriptor loss by optimization of affine parameters of corresponding features. Average over HPatchesSeq, illumination subset. Top row: geometric error of corresponding features (full line) and percentage of non-collapsed, *i.e.* elongation ≤ 6, features (dashed line). Bottom row: the fraction correct matches. All features initially have the same medium amount of reprojection noise. Left to right: HardNet, SIFT, TFeat, mean-normalized pixels descriptors.

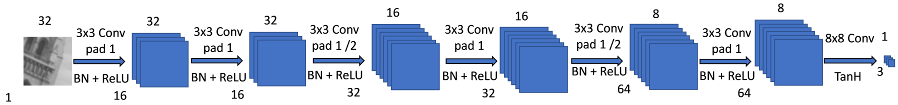


Fig. 4. AffNet. Feature map spatial size – top, # channels – bottom. /2 stands for stride 2.

$(s_i, \dot{s}_i)$. Descriptors $(s_i, \dot{s}_i)$ are then used to form triplets by the procedure proposed in [25], followed by our newly proposed hard negative-constant loss (Eq. 4).

More formally, we are finding affine transformation model parameters θ such that estimated affine transformation A minimizes descriptor HardNegC loss:

$$A(\theta|(P, \dot{P})) = \arg\min_{\theta} L(s, \dot{s}) \tag{6}$$

2.5 Training dataset and data preprocessing

UBC Phototour [39] dataset is used for training. It consists of three subsets: *Liberty*, *Notre Dame* and *Yosemite* with about 2 × 400k normalized 64x64 patches in each, detected by DoG and Harris detectors. Patches are verified by 3D reconstruction model. We randomly sample 10M pairs for training.

Although positive point corresponds to roughly the same point on the 3D surface, they are not perfectly aligned, having position, scale, rotation and affine noise. We have randomly generated affine transformations, which consist in random rotation – tied for pair of corresponding patches, and anisotropic scaling t in random direction by magnitude t_m, which is gradually increased during the training from the initial value of 3 to 5.8 at the middle of the training. The tilt is uniformly sampled from range $[0, t_m]$.

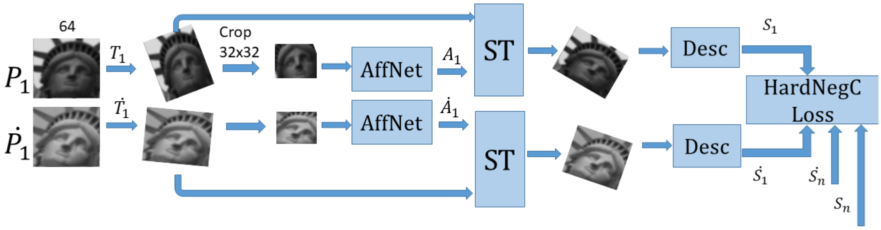


Fig. 5. AffNet training. Corresponding patches undergo random affine transformation $T_i, \dot{T}_i$, are cropped and fed into AffNet, which outputs affine transformation $A_i, \dot{A}_i$ to an unknown canonical shape. ST – the spatial transformer warps the patch into an estimated canonical shape. The patch is described by a differentiable CNN descriptor. $n \times n$ descriptor distance matrix is calculated and used to form triplets, according to the HardNegC loss.

2.6 Implementation details

The CNN architecture is adopted from HardNet[25], see Fig. 4, with the number of channels in all layers reduced 2x and the last 128D output replaced by a 3D output predicting ellipse shape. The network formula is 16C3-16C3-32C3/2-32C3-64C3/2-64C3-3C8, where 32C3/2 stands for 3x3 kernel with 32 filters and stride 2. Zero-padding is applied in all convolutional layers to preserve the size, except the last one. BatchNorm [40] layer followed by ReLU [41] is added after each convolutional layer, except the last one, which is followed by hyperbolic tangent activation. Dropout [42] with 0.25 rate is applied before the last convolution layer. Grayscale input patches 32×32 pixels are normalized by subtracting the per-patch mean and dividing by the per-patch standard deviation.

Optimization is done by SGD with learning rate 0.005, momentum 0.9, weight decay 0.0001. The learning rate decayed linearly [44] to zero within 20 epochs. The training was done with PyTorch [43] and took 24 hours on Titan X GPU; the bottleneck is the data augmentation procedure. The inference time is 0.1 ms per patch on Titan X, including patch sampling done on CPU and Baumberg iteration – 0.05 ms per patch on CPU.

3 Empirical evaluation

3.1 Loss functions and descriptors for learning measurement region

We trained different versions of the AffNet and orientation networks, with different combinations affine transformation parameterizations and descriptors with the procedure described above. The results of the comparison based on the number of correct matches (reprojection error ≤ 3 pixel) on the hardest pair for each of the 116 sequences from the HSequences [37] dataset are shown in Tables 1,2.

The proposed HardNetC loss is the only loss function with no "not converged" results. In the case of convergence, all tested descriptors and loss functions lead to comparable performance, unlike registration experiments in the previous section. We believe it is because now the CNN always outputs the same affine transformation for a patch, unlike in the previous experiment, where repeated features may end up with different shapes.

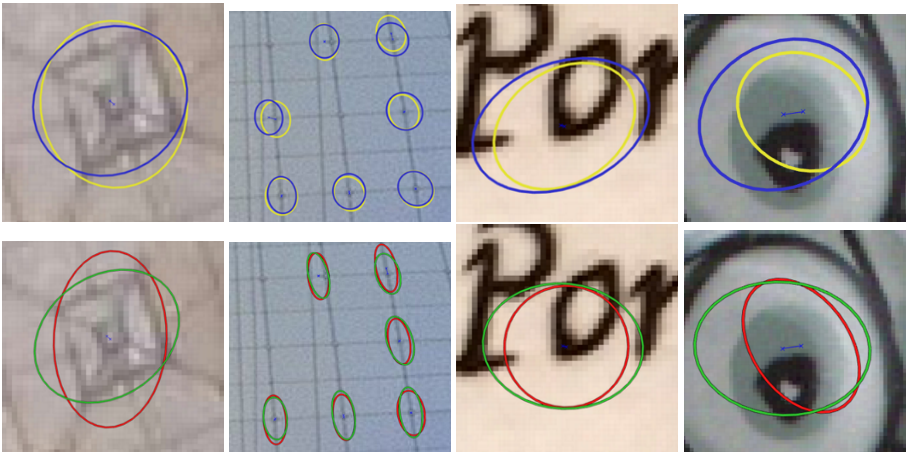


Fig. 6. AffNet (top) and Baumberg (bottom) estimated affine shape. One ellipse is detected in the reference image, the other is a reprojected closest match from the second image. Baumberg ellipses tend to be more elongated, average axis ratio is 1.99 vs. 1.63 for AffNet, median: Baumberg 1.72 vs 1.39 AffNet. The statistics are calculated over 16M features on Oxford5k.

Affine transformation parameterizations are compared in Table 2. All attempts to learn affine shape and orientation jointly in one network fail completely, or perform significantly worse than the two-stage procedure, when affine shape is learned first and orientation is estimated on an affine-shape-normalized patch. Learning residual shape A'' (Eq. 3) leads to the best results overall. Note, that such parameterization does not contain enough parameters to include feature orientation, thus "joint" learning is not possible. Slightly worse performance is obtained by using an identity matrix prior for learnable biases in the output layer.

3.2 Repeatability

Repeatability of affine detectors: Hessian detector + affine shape estimator was benchmarked, following classical work by Mikolajczyk *et al.* [8], but on recently introduced larger HSequences [37] dataset by VLBenchmarks toolbox [45].

HSequences consists of two subsets. *Illumination* part contains 57 image sixplets with illumination changes, both natural and artificial. There is no difference is viewpoint in this subset, geometrical relation between images in sixplets is identity.Second part is *Viewpoint*, where 59 image sixplets vary in scale, rotation, but mostly in horizontal tilt. The average viewpoint change is a bit smaller than in well-known *graffiti* sequence from Oxford-Affine dataset [8].

Local features are detected in pairs of images, reprojected by ground truth homography to the reference image and closest reprojected region is found for each region from reference image. The correspondence is considered correct, when overlap error of the pair is less than 40%. The repeatability score for a given pair of images is a ratio

Table 1. Learning the affine transform: loss functions and descriptor comparison. The median of average number of correct matches on the HSequences [37] hardest image pairs 1-6 for the Hessian detector and the HardNet descriptor. The match considered correct for reprojection error ≤ 3 pixels. Affine shape is parametrized as in Eq. 3. n/c – did not converge.

Training descriptor/loss	PosDist	HardNeg	HardNegC
Affine shape			
SIFT	n/c	385	386
HardNet	n/c	n/c	**388**
Baumberg [15]	298		
Orientation			
SIFT	**387**	379	382
HardNet	386	383	380
Dominant orientation [10]	339		

between number of correct correspondences and the smaller number of detected regions in common part of scene among two images.

Results are shown in Figure 7. Original affine shape estimation procedure, implemented in [12] is denoted Baum SS 19, as 19×19 patches are sampled from scale space. AffNet takes 32×32 patches, which are sampled from original image. So for fair comparison, we also tested Baum versions, where patches are sampled from original image, with 19 and 33 pixels patch size.

AffNet slightly outperforms all the variants of Baumberg procedure for images with viewpoint change in terms of repeatability and more significant – in number of correspondences. The difference is even bigger for them image with illumination change only, where AffNet performs almost the same as plain Hessian, which is upper bound here, as this part of dataset has no viewpoint changes.

We have also tested AffNet with other detectors on the Viewpoint subset of the HPatches. The repeatabilities are the following (no affine adaptation/Baumberg/AffNet): DoG: 0.46/0.51/0.52, Harris: 0.41/0.44/0.47, Hessian: 0.47/0.52/0.56 The proposed methods outperforms the standard (Baumberg) for all detectors.

One reason for such difference is the feature-rejection strategy. Baumberg iterative procedure rejects feature in one of three cases. First, elongated ellipses with long-to-short axis ratio more than six are rejected. Second, features touching boundary of the image are rejected. This is true for the AffNet post-processing procedure as well, but AffNet produces less elongated shapes: average axis ratio on Oxford5k 16M features is 1.63 vs. 1.99 for Baumberg. Both cases happen less often for AffNet, increasing the number of surviving features by 25%. We compared performance of the Baumberg vs. AffNet on the same number of features in Section 3.4. Finally, features whose shape did not converge within sixteen iteration are removed. This is quite rare, it happens in approximately 1% cases. Example of shapes estimated by AffNet and the Baumberg procedure are shown in Fig. 6.

Table 2. Learning the affine transform: parameterization comparison. The average number of correct matches on the HPatchesSeq [37] hardest image pairs 1-6 for the Hessian detector and the HardNet descriptor. Cases compared, affine shape combined with the de-facto handcrafted standard dominant orientation, affine shape and orientation learnt separately or jointly. The match considered correct for reprojection error ≤ 3 pixels. The HardNegC loss and HardNet descriptor used for learning. n/c – did not converge.

				Orientation		
		Estimated	biases	Learned		Dominant
Eq.	Matrix	parameters	init	jointly	separately	gradient [10]
(1)	A	$(a_{11}, a_{12}, a_{21}, a_{22})$	0	n/c	n/c	n/c
(1)	A	$(a_{11}, a_{12}, a_{21}, a_{22})$	1	n/c	360	320
(2)	A', $R(\alpha)$	$(a'_{11}, 0, a'_{21}, a'_{22})$, $(\sin\alpha, \cos\alpha)$	1	250	327	286
(3)	A''	$(a''_{11}, a''_{21}, a''_{22})$	1	-	370	340
(3)	A''	$(1 + a''_{11}, a''_{21}, 1 + a''_{22})$	0	-	**388**	349

Table 3. AffNet vs. Baumberg affine shape estimators on wide baseline stereo datasets, with Hessian and adaptive Hessian detectors, following the protocol [16]. The number of matched image pairs and the average number of inliers. The $\boxed{\text{numbers}}$ of image pairs in a dataset are boxed. Best results are in **bold**.

	EF [46]		EVD [3]		OxAff [8]		SymB [47]		GDB [48]		LTLL [49]	
Detector	$\boxed{33}$	inl.	$\boxed{15}$	inl.	$\boxed{40}$	inl.	$\boxed{46}$	inl.	$\boxed{22}$	inl.	$\boxed{172}$	inl.
HesAff [7]	**33**	78	**2**	38	**40**	1008	34	153	17	199	26	34
HesAffNet	**33**	**112**	**2**	**48**	**40**	**1181**	**37**	**203**	**19**	**222**	**46**	**36**
AdHesAff [16]	**33**	111	3	33	**40**	1330	35	190	19	286	28	35
AdHesAffNet	**33**	**165**	**4**	**42**	**40**	**1567**	**37**	**275**	**21**	**336**	**48**	**39**

3.3 Wide baseline stereo

We conducted an experiment on wide baseline stereo, following local feature detector benchmark protocol, defined in [16] on the set of two-view matching datasets [47,48,46,49]. The local features are detected by benchmarked detector, described by HardNet++ [25] and HalfRootSIFT [50] and geometrically verified by RANSAC [51]. Two following metrics are reported: the number of successfully matched image pairs and average number of correct inliers per matched pair. We have replaced original affine shape estimator in Hessian-Affine with AffNet in Hessian and Adaptive threshold Hessian (AdHess)

The results are shown in Table 3. AffNet outperforms Baumberg in both number of registered image pairs and/or number of correct inliers in all datasets, including painting-to-photo pairs in SymB [47] and multimodal pairs in GDB [48], despite it was not trained for that domains.

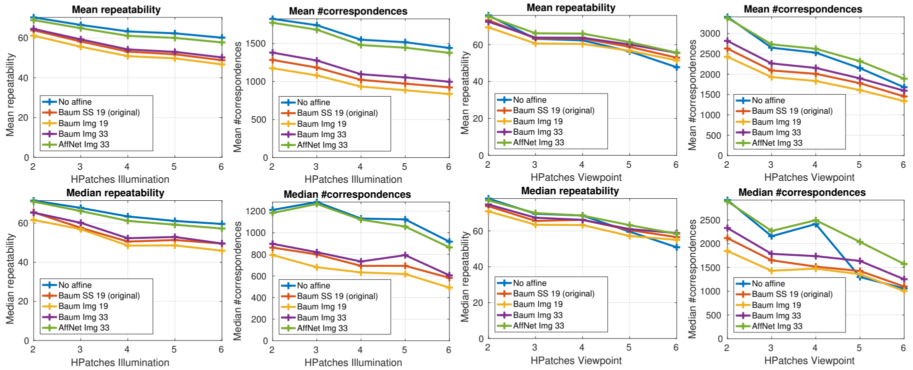


Fig. 7. Repeatability and the number of correspondences (mean top, median bottom row) on the HSequences [37]. AffNet is compared with the de facto standard Baumberg iteration [15] according to the Mikolajczyk protocol [8]. Left – images with illumination differences, right – with viewpoint and scale changes. SS – patch is sampled from the scale-space pyramid at the level of the detection, image – from the original image; 19 and 33 – patch sizes. Hessian-Affine is from[12]. For illumination subset, performance of Hessian with no adaptation is an upper bound, and AffNet performs close to it.

The total runtimes per image are the following (average for 800x600 images). Baseline HesAff + dominant gradient orientation + SIFT: no CNN components – 0.4 sec. HesAffNet (CNN) + dominant gradient orientation + SIFT – 0.8s, 3 CNN components: HesAffNet + OriNet + HardNet – 1.2 s. Now the data is naively transferred from CPU to GPU and back each of the stages, which generates the major bottleneck.

3.4 Image retrieval

We evaluate the proposed approach on standard image retrieval datasets Oxford5k [56] and Paris6k [57]. Each dataset contains images (5062 for Oxford5k and 6391 for Paris6k) depicting 11 different landmarks and distractors. The performance is reported as mean average precision (mAP) [56]. Recently, these benchmarks have been revisited, annotation errors fixed and new, more challenging sets of queries added [18]. The revisited datasets define new test protocols: *Easy*, *Medium*, and *Hard*.

We use the multi-scale Hessian-affine detector [8] with the Baumberg method for affine shape estimation. The proposed AffNet replaces Baumberg, which we denote HessAffNet. The use of HessAffNet increased the number of used feature, from 12.5M to 17.5M for Oxford5k and from 15.6M to 21.2M for Paris6k, because more features survive the affine shape adaptions, as explained in Section 3.2. We also performed additional experiment by restricting number of AffNet features to same as in Baumberg – HesAffNetLess in Table 4. We evaluated HesAffNet with both hand-crafted descriptor RootSIFT [11] and state-of-the-art learned descriptors [23,25].

First, HesAffNet is tested within the traditional bag-of-words (BoW) [58] image retrieval pipeline. A flat vocabulary with 1M centroids is created with the k-means algorithm and approximate nearest neighbor search [59]. All descriptors of an image are

Table 4. Performance (mAP) evaluation of the bag-of-words (BoW) image retrieval on the Oxford5k and Paris6k benchmarks. Vocabulary consisting of 1M visual words is learned on independent dataset: Oxford5k vocabulary for Paris6k evaluation and *vice versa*. SV: spatial verification. QE(t): query expansion with t inliers threshold. The best results are in **bold**.

	Oxford5k				Paris6k			
Detector–Descriptor	BoW	+SV	+SV+QE(15)	+SV+QE(8)	BoW	+SV	+SV+QE(15)	+SV+QE(8)
HesAff–RootSIFT [11]	55.1	63.0	78.4	80.1	59.3	63.7	76.4	77.4
HesAffNet–RootSIFT	61.6	72.8	86.5	88.0	63.5	71.2	81.7	83.5
HesAff–TFeat-M* [23]	46.7	55.6	72.2	73.8	43.8	51.8	65.3	69.7
HesAffNet–TFeat-M*	45.5	57.3	75.2	77.5	50.6	58.1	72.0	74.8
HesAff–HardNet++ [25]	60.8	69.6	84.5	85.1	65.0	70.3	79.1	79.9
HesAffNetLess–HardNet++	64.3	73.3	86.1	87.3	62.0	68.7	79.1	79.2
HesAffNet–HardNet++	**68.3**	**77.8**	**89.0**	**91.1**	**65.7**	**73.4**	**83.3**	**83.3**

Table 5. Performance (mAP) comparison with the state-of-the-art in local feature-based image retrieval. Vocabulary is learned on independent dataset: Oxford5k vocabulary for Paris6k evaluation and *vice versa*. All results are with spatial verification and query expansion. VS: vocabulary size. SA: single assignment. MA: multiple assignments. The best results are in **bold**.

		Oxford5k		Paris6k	
Method	VS	SA	MA	SA	MA
HesAff–SIFT–BoW-fVocab [52]	16M	74.0	84.9	73.6	82.4
HesAff–RootSIFT–HQE [13]	65k	85.3	88.0	81.3	82.8
HesAff–HardNet++–HQE [25]	65k	86.8	88.3	82.8	84.9
HesAffNet–HardNet++–HQE	65k	**87.9**	**89.5**	**84.2**	**85.9**

assigned to a respective centroid of the vocabulary, and then they are aggregated with a histogram of occurrences into a BoW image representation.

We also apply spatial verification (SV) [56] and standard query expansion (QE) [57]. QE is performed with images that have either 15 (typically used) or 8 inliers after the spatial verification. The results of the comparison are presented in Table 4.

AffNet achieves the best results on both Oxford5k and Paris6k datasets, in most of the cases it outperforms the second best approach by a large margin. This experiment clearly shows the benefit of using AffNet in the local feature detection pipeline.

Additionally, we compare with state-of-the-art local-feature-based image retrieval methods. A visual vocabulary of 65k words is learned, with Hamming embedding (HE) [60] technique added that further refines descriptor assignments with a 128 bits binary signature. We follow the same procedure as HesAff–RootSIFT–HQE [13] method. All parameters are set as in [13]. The performance of AffNet methods is the best reported on both Oxford5k and Paris6k for local features.

Finally, on the revisited R-Oxford and R-Paris, we compare with state-of-the-art methods in image retrieval, both local and global feature based: the best-performing fine-

Table 6. Performance (mAP, mP@10) comparison with the state-of-the-art in image retrieval on the R-Oxford and R-Paris benchmarks [18]. SV: spatial verification. HQE: hamming query expansion. αQE: α query expansion. DFS: global diffusion. The best results are in **bold**.

	Medium				Hard			
	R-Oxford		R-Paris		R-Oxford		R-Paris	
Method	mAP	mP@10	mAP	mP@10	mAP	mP@10	mAP	mP@10
ResNet101–GeM+αQE [53]	67.2	86.0	80.7	**98.9**	40.7	54.9	61.8	90.6
ResNet101–GeM[53]+DFS [54]	69.8	84.0	88.9	96.9	40.5	54.4	78.5	**94.6**
ResNet101–R-MAC[55]+DFS [54]	69.0	82.3	**89.5**	96.7	44.7	60.5	**80.0**	94.1
ResNet50–DELF[64]–HQE+SV	73.4	88.2	84.0	98.3	50.3	67.2	69.3	93.7
HesAff–RootSIFT–HQE [13]	66.3	85.6	68.9	97.3	41.3	60.0	44.7	79.9
HesAff–RootSIFT–HQE+SV [13]	71.3	88.1	70.2	98.6	49.7	69.6	45.1	83.9
HesAffNet–HardNet++–HQE	71.7	89.4	72.6	98.1	47.5	66.3	48.9	85.9
HesAffNet–HardNet++–HQE+SV	**75.2**	**90.9**	73.1	98.1	**53.3**	**72.6**	48.9	89.1

tuned networks [63], ResNet101 with generalized-mean pooling (ResNet101–GeM) [53] and ResNet101 with regional maximum activations pooling (ResNet101–R-MAC) [55]. Deep methods use re-ranking methods: α query expansion (αQE) [53], and global diffusion (DFS) [54]. Results are in Table 6.

HesAffNet performs best on the R-Oxford. It is consistently the best performing local-feature method, yet is worse than deep methods on R-Paris. A possible explanation is that deep networks (ResNet and DELF) were finetuned from ImageNet, which contains Paris-related images, e.g. Sacre-Coeur and Notre Dame Basilica in the "church" category. Therefore global deep nets are partially evaluated on the training set.

4 Conclusions

We presented a method for learning affine shape of local features in a weakly-supervised manner. The proposed HardNegC loss function might find other application domains as well. Our intuition is that the distance to the hard-negative estimates the local density of all points and provides a scale for the positive distance. The resulting AffNet regressor bridges the gap between performance of the similarity-covariant and affine-covariant detectors on images with short baseline and big illumination differences and it improves performance of affine-covariant detectors in the wide baseline setup. AffNet applied to the output of the Hessian detector improves the state-of-the art in wide baseline matching, affine detector repeatability and image retrieval.

We experimentally show that descriptor matchability, not only repeatability should be taken into account when learning a feature detector.

Acknowledgements The authors were supported by the Czech Science Foundation Project GACR P103/12/G084, the Austrian Ministry for Transport, Innovation and Technology, the Federal Ministry of Science, Research and Economy, and the Province of Upper Austria in the frame of the COMET center, the CTU student grant SGS17/185/OHK3/3T/13, and the MSMT LL1303 ERC-CZ grant.

References

1. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2016) 4104–4113 1
2. Schonberger, J.L., Hardmeier, H., Sattler, T., Pollefeys, M.: Comparative evaluation of hand-crafted and learned local features. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2017) 1
3. Mishkin, D., Matas, J., Perdoch, M.: Mods: Fast and robust method for two-view matching. Computer Vision and Image Understanding **141** (2015) 81 – 93 1, 11
4. Sattler, T., Maddern, W., Torii, A., Sivic, J., Pajdla, T., Pollefeys, M., Okutomi, M.: Benchmarking 6DOF Urban Visual Localization in Changing Conditions. ArXiv e-prints (July 2017) 1
5. Radenovic, F., Tolias, G., Chum, O.: CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In: European Conference on Computer Vision (ECCV). (2016) 3–20 1
6. Lucas, B., Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. In: International Joint Conference on Artificial Intelligence (IJCAI). (1981) 674–679
7. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. International Journal of Computer Vision (IJCV) **60**(1) (2004) 63–86 1, 11
8. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. International Journal of Computer Vision (IJCV) **65**(1) (2005) 43–72 1, 9, 11, 12
9. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. In: International Conference on Computer Vision (ICCV). (2011) 2564–2571 1
10. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) **60**(2) (2004) 91–110 1, 5, 10, 11
11. Arandjelovic, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2012) 2911–2918 1, 12, 13
12. Perdoch, M., Chum, O., Matas, J.: Efficient representation of local geometry for large scale object retrieval. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2009) 9–16 1, 10, 12
13. Tolias, G., Jegou, H.: Visual query expansion with or without geometry: refining local descriptors by feature aggregation. Pattern Recognition **47**(10) (2014) 3466–3476 1, 13, 14
14. Pritts, J., Kukelova, Z., Larsson, V., Chum, O.: Radially-distorted conjugate translations. In: CVPR. (2018) 1
15. Baumberg, A.: Reliable feature matching across widely separated views. In: CVPR, IEEE Computer Society (2000) 1774–1781 1, 2, 10, 12
16. Mishkin, D., Matas, J., Perdoch, M., Lenc, K.: Wxbs: Wide baseline stereo generalizations. Arxiv 1504.06603 (2015) 1, 11
17. Schonberger, J.L., Radenovic, F., Chum, O., Frahm, J.M.: From single image query to detailed 3D reconstruction. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 5126–5134 1
18. Radenovic, F., Iscen, A., Tolias, G., Avrithis, Y., Chum, O.: Revisiting Oxford and Paris: Large-Scale Image Retrieval Benchmarking. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2018) 12, 14
19. Hara, K., Vemulapalli, R., Chellappa, R.: Designing Deep Convolutional Neural Networks for Continuous Object Orientation Estimation. ArXiv e-prints (February 2017)

20. Radenovic, F., Schonberger, J.L., Ji, D., Frahm, J.M., Chum, O., Matas, J.: From dusk till dawn: Modeling in the dark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 5488–5496 1
21. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 2
22. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C.: Matchnet: Unifying feature and metric learning for patch-based matching. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 3279–3286 2
23. Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K.: Learning local feature descriptors with triplets and shallow convolutional neural networks. In: British Machine Vision Conference (BMVC). (2016) 2, 5, 12, 13
24. Yurun Tian, B.F., Wu, F.: L2-net: Deep learning of discriminative patch descriptor in euclidean space. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2017) 2
25. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. In: Proceedings of NIPS. (December 2017) 2, 3, 4, 5, 6, 7, 8, 11, 12, 13
26. Zhang, X., Yu, F.X., Kumar, S., Chang, S.F.: Learning Spread-out Local Feature Descriptors. ArXiv e-prints (August 2017) 2
27. Dosovitskiy, A., Fischer, P., Springenberg, J.T., Riedmiller, M.A., Brox, T.: Discriminative unsupervised feature learning with exemplar convolutional neural networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(9) (2016) 1734–1747 2
28. Verdie, Y., Yi, K., Fua, P., Lepetit, V.: Tilde: a temporally invariant learned detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 5279–5288 2
29. Zhang, X., Yu, F., Karaman, S., Chang, S.F.: Learning discriminative and transformation covariant local feature detectors. In: CVPR. (2017) 2
30. Lenc, K., Vedaldi, A. In: Learning Covariant Feature Detectors. Springer International Publishing, Cham (2016) 100–117 2
31. Savinov, N., Seki, A., Ladicky, L., Sattler, T., Pollefeys, M.: Quad-networks: unsupervised learning to rank for interest point detection. ArXiv e-prints (November 2016) 2
32. W. Hartmann, M. Havlena, and K. Schindler. Predicting matchability. In *CVPR*, pages 9–16. IEEE Computer Society, 2014. 2
33. Yi, K.M., Verdie, Y., Fua, P., Lepetit, V.: Learning to Assign Orientations to Feature Points. In: Proceedings of the Computer Vision and Pattern Recognition. (2016) 2, 4, 6
34. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: Learned invariant feature transform. In: European Conference on Computer Vision (ECCV). (2016) 467–483 2
35. Choy, C.B., Gwak, J., Savarese, S., Chandraker, M.: Universal correspondence network. In: Advances in Neural Information Processing Systems. (2016) 2414–2422 3
36. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR. (2015) 4
37. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2017) 5, 8, 9, 10, 11, 12
38. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial Transformer Networks. ArXiv e-prints (June 2015) 6
39. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. International Journal of Computer Vision (IJCV) **74**(1) (2007) 59–73 7
40. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. ArXiv 1502.03167 (2015) 8
41. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: International Conference on Machine Learning (ICML). (2010) 807–814 8

42. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research (JMLR) **15**(1) (2014) 1929–1958 8
43. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: Proceedings of NIPS Workshop. (December 2017) 8
44. Mishkin, D., Sergievskiy, N., Matas, J.: Systematic evaluation of convolution neural network advances on the Imagenet. Computer Vision and Image Understanding (2017) 11–19 8
45. Lenc, K., Gulshan, V., Vedaldi, A.: Vlbenchmarks (2012) 9
46. Zitnick, C.L., Ramnath, K.: Edge foci interest points. In: International Conference on Computer Vision (ICCV). (2011) 359–366 11
47. Hauagge, D.C., Snavely, N.: Image matching using local symmetry features. In: Computer Vision and Pattern Recognition (CVPR). (2012) 206–213 11
48. Yang, G., Stewart, C.V., Sofka, M., Tsai, C.L.: Registration of challenging image pairs: Initialization, estimation, and decision. Pattern Analysis and Machine Intelligence (PAMI) **29**(11) (2007) 1973–1989 11
49. Fernando, B., Tommasi, T., Tuytelaars, T.: Location recognition over large time lags. Computer Vision and Image Understanding **139** (2015) 21 – 28 11
50. Kelman, A., Sofka, M., Stewart, C.V.: Keypoint descriptors for matching across multiple image modalities and non-linear intensity variations. In: CVPR 2007. (2007) 11
51. Lebeda, K., Matas, J., Chum, O.: Fixing the locally optimized ransac. In: BMVC 2012. (2012) 11
52. Mikulik, A., Perdoch, M., Chum, O., Matas, J.: Learning vocabularies over a fine quantization. International Journal of Computer Vision (IJCV) **103**(1) (2013) 163–175 13
53. Radenović, F., Tolias, G., Chum, O.: Fine-tuning cnn image retrieval with no human annotation. arXiv:1711.02512 (2017) 14
54. Iscen, A., Tolias, G., Avrithis, Y., Furon, T., Chum, O.: Efficient diffusion on region manifolds: Recovering small objects with compact cnn representations. In: CVPR. (2017) 14
55. Gordo, A., Almazan, J., Revaud, J., Larlus, D.: End-to-end learning of deep visual representations for image retrieval. IJCV (2017) 14
56. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2007) 1–8 12, 13
57. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2008) 1–8 12, 13
58. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: International Conference on Computer Vision (ICCV). (2003) 1470–1477 12
59. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: International Conference on Computer Vision Theory and Application (VISSAPP). (2009) 331–340 12
60. Jegou, H., Douze, M., Schmid, C.: Improving bag-of-features for large scale image search. International Journal of Computer Vision (IJCV) **87**(3) (2010) 316–336 13
61. Jegou, H., Douze, M., Schmid, C.: On the burstiness of visual elements. In: Computer Vision and Pattern Recognition (CVPR). (2009) 1169–1176
62. Jegou, H., Schmid, C., Harzallah, H., Verbeek, J.: Accurate image search using the contextual dissimilarity measure. Pattern Analysis and Machine Intelligence (PAMI) **32**(1) (2010) 2–11
63. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 14
64. Noh, H., Araujo, A., Sim, J., Weyand, T., Han, B.: Large-Scale Image Retrieval with Attentive Deep Local Features. In: ICCV. (2017) 14