From handcrafted to deep local invariant features

Gabriela Csurka and Martin Humenberger
Naver Labs Europe, 6 chemin de Maupertuis, 38240 Meylan, France
firstname.lastname@naverlabs.com
www.europe.naverlabs.com

Abstract The aim of this paper is to present a comprehensive overview of the evolution of local features from handcrafted to deep learning based methods, followed by a discussion of several benchmark and evaluation papers about this topic. Our investigations are motivated by 3D reconstruction problems, where the precise location of the features are important. During the description of the methods, we highlight and explain challenges of feature extraction and potential ways to overcome them. We first present traditional handcrafted methods, followed by data driven learning based approaches, and finally detail deep learning based methods. We are convinced that this evolutionary presentation will help the reader to fully understand the topic of image and region description in order to make best use of it in modern computer vision applications. In other words, understanding traditional methods and their motivation will help understanding modern approaches and how machine learning is used to improve the results. We also provide comprehensive references to most relevant literature and code.

1 Introduction

In the computer vision literature we often meet local features in two main contexts. On one hand, *local feature* can simply represent the description of a local region in the image by the characteristics of the region. Such local features, referred to as *local descriptors* could be obtained by concatenating the pixel intensities or computing a color histogram of the region. Such local descriptors can be assigned to any random local image region. On the other hand, local features or more often called *local invariant features* refer to distinctive points or areas, called *keypoints*, interest points or anchor points in an image that differs from its immediate neighborhood. In this case both their precise location in the image, called *detection* and their *description* plays an important role. The keypoint detection consists of determining the region to be described often accompanied with a set of transformations to be applied to the region in order to make the descriptor invariant to some geometric and/or photometric transformations. In this paper we mainly focus on the second case and discuss keypoint detection and invariant local descriptor methods. For clarification, in the remainder of this paper a *feature* consists of a *keypoint/interest point* and its *descriptor*.

Preprint for our journal submission

Such, local invariant features have been applied successfully in a wide range of systems and applications, including object as well as face detection and recognition, image retrieval, motion detection and tracking, depth map generation, panorama stitching, camera calibration, 3D reconstruction, structure from motion (SfM), visual odometry and VSLAM. Furthermore, the importance of local features has also been demonstrated in the context of object recognition by the human visual system [Biederman, 1987].

We distinguish three broad categories of local descriptors mainly depending on the targeted applications for which these features were designed [Tuytelaars and Mikolajczyk, 2007]. First, local features that may have a specific semantic interpretation in the limited context of a certain application, such as edges detected in aerial images often corresponding to roads or blobs representing impurities in certain inspection tasks. The second category consists of individually identifiable anchor points that can be localized accurately and consistent over time. Such features are key elements for most matching (e.g. 3D reconstruction) or tracking (e.g. visual odometry) applications. Finally, a set of local features extracted sparsely or densely from an image can be used as a robust image representation, which allows to recognize objects or scenes. In this latter case, the local features do not need to have a certain semantic meaning or sub-pixel accurate localization, as the corresponding applications such as image retrieval, scene and video understanding exploit mainly the statistics of the feature set.

When accurate geometric recovery matters, such as camera calibration, 3D reconstruction or SfM, both aspects detection and description are equally important, for other applications such as image classification or retrieval, the accent is on the description and less on the precise position of the local region for which the descriptor was derived. Formally speaking, the local feature is assigned to a so-called keypoint or interest point which defines the exact position of the local feature within the image and furthermore defines the region (often a circular or a square region) which is used to compute the corresponding descriptor. In this paper we are mainly interested in the second category, where the aim is to accurately localize and match local keypoints between images corresponding to the same 3D anchor point in the 3D scene.

To find such keypoints, the local feature ideally should have the following properties: The detected keypoints can be accurately localized in the image independent of view point and lighting conditions and it is stable under local and global perturbations in the image domain such that the interest points can be reliably computed with high degree of repeatability. The local descriptor describing two keypoints representing the same 3D anchor point in two different images should be, a.o., robust to image noise, discretization effects, compression artifacts and blur. The local descriptors should furthermore be rich in terms of local information contents, such that different features can be distinguished and the corresponding ones can be easily matched. The density of keypoints in the image should reflect the information content necessary to the application.

Since generally a projective transformation describes the motion between two images of the same scene, the local feature descriptors should be invariant to such transformations. Alternatively, assuming a canonical view of the patch, the geometric transformation between the assumed canonical view and current view is first estimated and second, the local region is transformed into the desired *canonical view*, and third, the descriptor is estimated on this normalized patch (see Figure 1). Locally, the projective transformation is well estimated by an affine transformation, but many approaches simplify this and estimate a dominant orientation and a characteristic scale to obtain the normalized patch.

To answer these large amount of requirement needs, where several properties compete with each other, in the last two decades a large set of different interest point detection and local feature descriptor methods were proposed. In addition to our description, the reader should refer to the comprehensive surveys [Tuytelaars and Mikolajczyk, 2007, Krig, 2014, Fan et al., 2014b] and benchmarks [Mikolajczyk et al., 2005, Mikolajczyk and Schmid, 2003, Heinly et al., 2012, Balntas et al., 2017, Schönberger et al., 2017]. Many of these

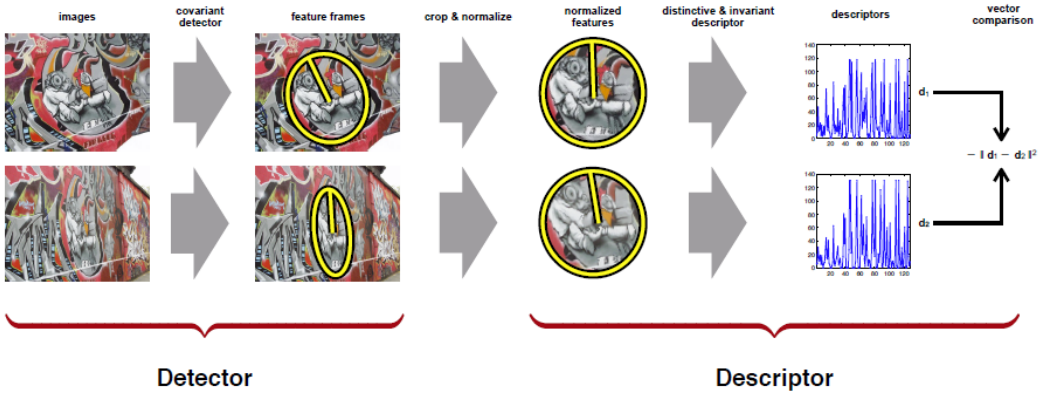


Fig. 1 Local feature detection and description pipeline. (Courtesy to Andrea Vedaldi)

feature detectors and descriptors are integrated in OpenCV[1] or VLFeat[2].

Note also, that for a timespan of more than one decade and before the deep learning era, Bag of Visual-Words [Sivic et al., 2003, Csurka et al., 2004], and its extensions such as Fisher Vectors [Perronnin and Dance, 2007] or VLAD [Jégou et al., 2010] were the most used image representations for image classification and retrieval. These representations are aggregations of local descriptors, and therefore a tremendous amount of work on methods learning, combining, and improving descriptors in the context of object recognition, image classification, and retrieval was published. However, since the focus of this paper is a survey on local invariant features, detailed presentation of that literature is out of the scope (some work is mentioned anyways).

The rest of this paper is structured as follows: Section 2 presents relevant work on so-called handcrafted features which try to describe the region in a predefined, handcrafted, way. Section 3 continues with methods which cast region description as a data driven learning task with the goal to better cope with regions which cannot be described with handcrafted methods. Section 4 presents the most recent advances in region and image description using deep learning where intermediate layers (e.g. activation layers) of a DNN can be used as image or region representation. In Section 5 we summarize the main findings of several benchmark papers and provide some guidance concerning the choice of the models and methods to be used depending on the problem. The paper is concluded in Section 6.

[1] See Section "Feature Detection and Description" of the OpenCV tutorial on `https://docs.opencv.org/3.1.0/d6/d00/tutorial_py_root.html`

[2] The matlab VLfeat library is available on `http://www.vlfeat.org/`

2 Handcrafted local invariant features

Before the end-to-end deep learning revolution, hand crafted local image descriptors were one of the key elements of almost all computer vision applications. The idea was to describe the local regions as good as possible using elements from pattern recognition such as edges, corners, gradients, circles, and known shapes in general. Their role and the associated requirements obviously were not always the same and therefore many different keypoint detectors and local descriptors were proposed in the last 3-4 decades.

2.1 Keypoint detectors

Early methods are corner detectors based on methods trying to find maxima of curvature or angular changes along edges [Rosenfeld and Kak, 1982, Wang and Brady, 1995]. However, intensity based detectors were among the most popular methods. They rely on analyzing local differential geometry or intensity patterns to find points or regions that satisfy certain uniqueness and stability criteria. Noteably the Hessian based [Zuniga and Haralick, 1983] and the Harris-based [Harris and Stephens, 1988] methods and their extensions. The Hessian Matrix method, also referred to as Determinant of Hessian (DoH) method, is used in the popular SURF algorithm [Bay et al., 2006], both methods were extended in [Mikolajczyk and Schmid, 2004] to handle affine invariance. One of the most popular keypoint detection methods is SIFT [Lowe, 2004] which uses Difference of Gaussians (DoG) to detect local extrema features.

[Gauglitz et al., 2011] propose two methods, the Center-of-Mass based Orientation Assignment and the Histogram-of-Intensities-based Orientation Assignment to improve the orientation assigned to the keypoint allowing to render the process invariant to in-plane rotation.

Salient regions-based edge detectors exploit the notion that interest points over a range of scales should exhibit local attributes or entropy that are unpredictable compared to the surrounding region. As such [Kadir et al., 2004] proposes to measure the change in entropy of a gray-value histogram computed in a set of neighborhoods varying the position, scale, and affine shape of the region. Wavelet transformation was also investigated in the context of feature point extraction based on multi-resolution analysis [Sebe et al., 2003].

Intensity variation based approaches apply mathematical morphology to extract high curvature points. As such, SUSAN [Smith and Brady, 2002] computes the fraction of pixels within a neighborhood which have similar intensity to the center pixel. Corners can then be localized by applying a threshold on this measure and selecting local minima. FAST [Rosten and Drummond, 2006] is an extension of SUSAN, where the keypoint extraction process was significantly sped up. The method relies on a connected set of pixels in a circular pattern to determine a corner, where a set of comparisons between the intensity values of the pixels on the circle and the central pixel is considered. If a number of consecutive comparisons are consistent, the central pixel is considered to be a good candidate. The process is concluded with a non-maximum suppression stage[3]. As originally, FAST is not a scale-space detector, an extension to a multi-scale detector by scale selection with the Laplacian function was proposed by [Lepetit and Fua, 2006]. This is done by estimating the Laplacian using gray-level differences between pixels on the circle and the center pixel and only the

[3] The detector can be improved using a trained decision tree.

locations where this estimate is largest is retained.

Segmentation techniques have also been employed in the context of feature extraction. These methods were either applied to find homogeneous regions to localize junctions on their boundaries [Liu and Tsai, 1990] or to directly use these regions as local features [Corso and Hager, 2005]. Similarly, MSER [Matas et al., 2002] uses watershed like segmentation to extracts homogeneous intensity regions which are stable over a wide range of thresholds. The regions are then replaced by ellipses with the same shape moments up to the second-order.

2.2 Local feature descriptors

While several methods mentioned above consider both aspects of the problem, namely the interest point detector and the local region descriptor, sometimes it is advantageous to treat them separately. This means, once an interest point was found, an additional descriptor for the given region can be computed. In addition, as we mentioned above, there are applications such as image categorization or retrieval where mainly the statistics of the local features are exploited, such as in Bag of visual-words (BOV) [Sivic et al., 2003, Csurka et al., 2004], or Fisher Vectors (FV) [Perronnin and Dance, 2007] and the precise location of the considered regions in generally less critical. What is more important is to consider features that are as informative as possible, discriminative enough depending on the task and can be computed from the local region independently from the rest of the image.

A large amount of such local descriptors were proposed in the literature, some of them being extremely task specific. Our aim is not to provide an exhaustive list of them, instead we will mention the most popular ones with a special focus on descriptors used to describe local regions around located in keypoints.

One of the most used local feature descriptor is SIFT [Lowe, 2004]. Note that in the literature we find the reference to SIFT as the whole pipeline shown in Figure 2(top line), including the interest point detection and localization with Difference of Gaussians (DoG) or as the feature descriptor based on local gradient histograms computed on a the 4x4 grid only. SIFT has several extensions including GLOH [Mikolajczyk and Schmid, 2003] considering a log-polar location grid on which the gradients are averaged to compute the histograms, CSIFT [Abdel-Hakim and Farag, 2006] which exploits color invariant characteristics, DSP-SIFT [Dong and Soatto, 2015] pooling SIFT descriptors across scales, and Scale-Less SIFT [Hassner et al., 2017] which is a subspace representation of SIFT across multiple scales.

The underlying idea of DSP-SIFT [Dong and Soatto, 2015] is that instead of considering the scale-space used by DoG referring to a continuum of images obtained by smoothing and down-sampling the original image, a size-space is used where instead of maintaining the same scale of the image, sub-images of different sizes are considered. Furthermore, while in SIFT the descriptor is constructed from the image at a selected scale and gradient orientations are pooled in its spatial neighborhood, DSP-SIFT considers patches of different sizes that are re-scaled and the gradient orientation are pooled across locations and scales (see Figure 2 bottom line). Note that the domain size pooling (DSP) can be applied to other features than SIFT as well.

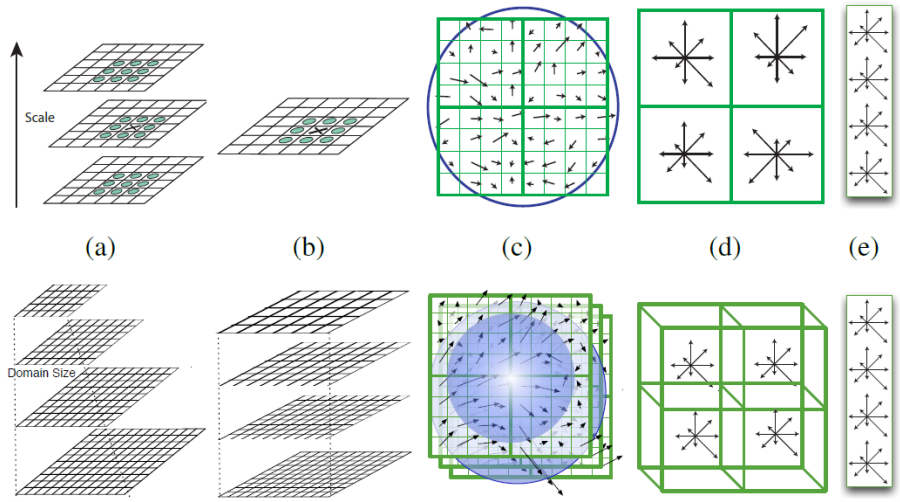


Fig. 2 Top: The SIFT [Lowe, 2004] feature detection and description steps. Bottom: The DSP-SIFT [Dong and Soatto, 2015] feature detection and description steps. (Courtesy to Jingming Dong)

Scale-Less SIFT [Hassner et al., 2017] show that representing each pixel as a set of SIFTs, extracted at multiple scales, allows for far better matches than descriptors with a selected single-scale. As this comes with significant computational cost, the authors propose to represent each set by a low-dimensional, linear subspace and a subspace-to-point mapping is used to get the final descriptors.

DAISY[4] [Tola et al., 2010] uses a log-polar grid arrangement and considers Gaussian pooling of histograms of gradient orientations. However, while DAISY features are efficient for dense computation and hence can be efficiently used to build Bag of Visual Words or Fisher Vectors representations, they are not well adapted to be used to detect sparse keypoints.

Another popular keypoint detector is MSER [Matas et al., 2002], which was developed for solving disparity correspondence in a wide baseline stereo system, uses watershed like segmentation to extracts regions of homogeneous intensity and sort the pixels based on intensity thresholds. Regions with similar pixel values over a range of threshold values in a connected component pattern are considered maximally stable. Then the so called Maximally Stable Extremal Regions (MSER) can be described using different shape metrics (shape moments, Fourier descriptors, shape context [Belongie et al., 2000]).

SURF [Bay et al., 2006] is a scale-invariant feature detector based on the Hessian-Laplace detector, where for efficiency the Hessian matrix is roughly approximated, using a set of box type filters and no smoothing is applied when going from one scale to the next. The descriptor computes Haar filter responses using integral images, which allows the features to be estimated much faster than SIFT while still maintaining high

[4] DAISY code available at `https://github.com/etola/libdaisy`

robustness.

KAZE feature detectors [Alcantarilla et al., 2012] have a similar detection pipeline to SURF, except that 2D features are detected and described in a nonlinear scale space by means of nonlinear diffusion filtering. The nonlinear scale space is built using efficient additive operator splitting techniques and variable conductance diffusion. The Accelerated KAZE[5] [Alcantarilla and Nuevo, 2013] uses Fast Explicit Diffusion embedded in a pyramidal framework to dramatically speed-up the KAZE feature detection in nonlinear scale spaces. In addition, the authors also introduced a binary version of the descriptors called Modified-Local Difference Binary (M-LDB), which is highly efficient, scale and rotation invariant, and exploits gradient information from the nonlinear scale space.

Intensity order and relationship was exploited to define MROGH [Fan et al., 2011] and LIOP [Wang et al., 2016]. MROGH[6] combines intensity orders and gradient distributions in multiple support regions. The rotation invariant gradients are adaptively pooled based on intensity orders to encode spatial information and multiple support regions are used for constructing the descriptor which further improves its discriminative ability.

LIOP (Local Intensity Order Pattern[7]) captures ordinal information by using the intensity relationships among all the neighboring sampling points around a pixel and an overall intensity order pattern (OIOP) which exploits the coarsely quantized overall intensity order of these sampling points.

In contrast to the previous approaches, EdgeFoci[8] [Zitnick and Ramnath, 2011] uses normalized intensity edges and their orientations to detect points that are roughly equidistant from edges with orientations perpendicular to that point. The scale of the interest point is defined by the distance between these interest points and the edges, called edge foci.

Occlusion-Aware Template Matching (OATM) [Korman et al., 2018] addresses the problem of how to handle partial occlusions when matching local regions. To do this they model occlusions explicitly as part of a robust template matching process where the co-visible region is assumed to undergo affine deformations of the domain and range, up to additive noise. To do this they formulate the problem as Consensus Set Maximization by searching for a transformation under which a maximal number of pixels are co-visible, *i.e.*, mapped with a residual which is within a given threshold.

2.3 Local binary features

In applications such as tracking or visual simultaneous localization and mapping (VSLAM), the keypoint detection and feature matching has to be done in real time. Therefore, many methods were proposed for which the priority was to increase computational speed and to decrease the memory footprint while maintaining high accuracy. Among them there are the so called local binary features such as LBP, Census features,

[5] (A)KAZE is available on `http://www.robesafe.com/personal/pablo.alcantarilla/kaze.html`

[6] MROGH code available at `https://github.com/bfan/MROGH-feature-descriptor`

[7] Code for LIOP available on `http://zhwang.me/publication/liop/index.html`

[8] EdgeFoci page on `http://research.microsoft.com/en-us/um/people/larryz/edgefoci/edge_foci.htm`

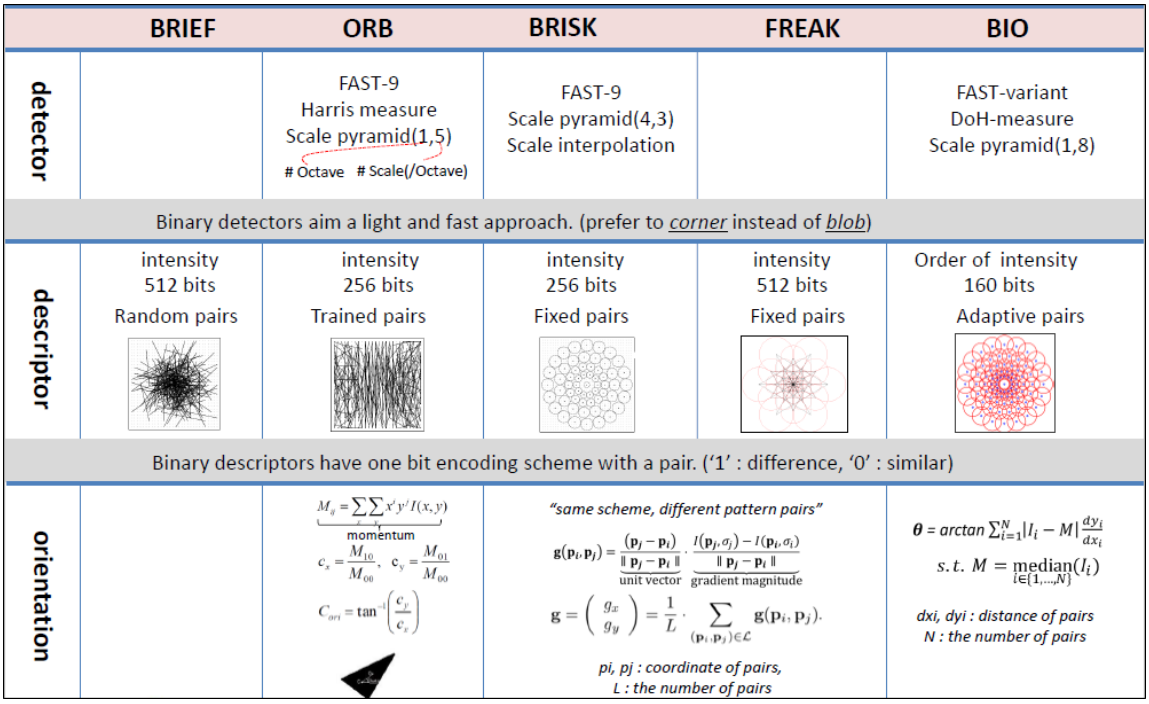


Fig. 3 An illustrative summary of some of the popular binary descriptors. (Courtesy to Yukyung Choi)

BRIEF, ORB, BRISK, FREAK or BIO where Hamming distance can be used to efficiently compute similarity between features (see also Figure 3).

Local binary patterns (LBP) [Pietikäinen et al., 2011] creates a descriptor or texture model using a set of histograms of the local texture neighborhood surrounding each pixel. This is done by comparing the intensity of each pixel's value against its neighbors and encode 1 if the value is greater and 0 otherwise. This metric in spite of its simplicity has shown to be quite powerful. LPB has a large set of variants and extensions. One of the most popular one is the Census transform [Zabih and Woodfill, 1994] that records pixel comparison results made between the center pixel in the kernel and the other pixels in the kernel region. The Census transform also uses a feature called the rank value scalar, which is the number of pixel values less than the center pixel.

The Binary Robust Independent Elementary Feature (BRIEF) descriptor [Calonder et al., 2010] uses a random distribution pattern of point-pairs in a local region for the binary comparison to create the descriptor. Oriented BRIEF, also known as ORB features [Rublee et al., 2011], adds rotational invariance to BRIEF by determining corner orientation using FAST [Rosten and Drummond, 2006] followed by a Harris corner metric to sort the keypoints.

Binary Robust Invariant Scalable Keypoint (BRISK) [Leutenegger et al., 2011] is another popular local binary method that uses a circular-symmetric pattern region shape and point-pairs as line segments arranged in four concentric rings. Using both short and long segments allows them to measure scale invariance. It uses gradient orientation to adjust and rotate short pairs.

FREAK[9] [Alahi et al., 2012] uses a novel foveal-inspired multi-resolution pixel pair sampling shape with trained pixel-pairs to mimic the design of the human eye as a coarse-to-fine descriptor with resolution highest in the center and decreasing further into the periphery. The feature descriptor is designed in a coarse-to-fine cascade of four groups containing pixel-pair binary comparisons stored in a vector.

BIO [Choi et al., 2014] gets inspiration from LIOP [Wang et al., 2016] which encodes the local ordinal information and creates a histogram of local intensity order patterns for each ordinal sub-region. They use a FAST-like binary comparison test and propose a keypoint measure as an approximation of the Determinant of Hessian (DoH) to balance computational efficiency and repeatability performance. They estimate the dominant orientation of a keypoint as the weighted average of the local directional vectors and obtain a measurement vector by tracing the pattern intensities aligned with the dominant orientation that is transformed into a rank-order vector. As ordinal descriptions are insensitive to moderate rank-order errors, they can be quantized into binary descriptors without a noticeable degradation of performance.

2.4 Improving keypoint detection and matching

A large set of methods was proposed which instead of focusing on the features themselves, tries to improve either the accuracy or the speed of the keypoint extraction and/or the matching process. Using binary features presented in Section 2.3 mainly acts on the matching speed, thanks to the very efficient computation of the Hamming distances. Further methods use geometrical or statistical constraints to improve the matching process. As such [Shah and Narayanan, 2015] propose a two-stage, geometry-aware approach for matching SIFT-like features in a fast and reliable manner. First, a small sample of SIFT features matched with KD-tree is used to estimate the epipolar geometry between the images. Second, it is used to guide the matching of the remaining features. [Torki and Elgammal, 2010] embed all features within an Euclidean space where their locations reflect both, the descriptor similarity and the spatial arrangement. They match multiple feature sets by solving an Eigen-value problem. [Chen et al., 2013] propose to represent the homographies in alternate Hough space where voting is performed. They show, that for establishing feature correspondences, alternating Hough transform and inverted Hough transform improves both, precision and recall.

A very popular way to improve the matches is geometric verification within a RANSAC [Fischler and Bolles, 1981] (Random Sample Consensus) scheme. To do this, the matches are used to estimate, *e.g.*, the fundamental matrix of an image pair. Using RANSAC, the inliers (matches which fulfill the best model estimation) are determined and kept as reliable matches (a threshold can be applied to decide about the overall quality of the estimation). Many extensions to RANSAC, such as MLESAC [Torr and Zisserman, 2000], Locally Optimized RANSAC [Chum et al., 2003], PROSAC [Chum and Matas, 2005], Optimal Randomized

[9] Code available at `https://github.com/kikohs/freak`

RANSAC [Chum and Matas, 2008] or GroupSAC [Ni et al., 2009] have been proposed.

The Vector Field Consensus (VFC) [Ma et al., 2014] relies on the calculation of an interpolated flow field based on correspondences generated by any kind of feature matcher modeled by a mixture model with the assumption that the noise is Gaussian for inliers and uniform for outliers. The method uses a smoothness criterion on the vector field for stabilization which leads to higher robustness but also rejects true positives at flow discontinuities originating from, *e.g.*, borders. They also show that in the special case where there is an underlying parametric geometrical model such as the epipolar line constraint, and a large number of outliers is present the method is more robust than the standard alternatives such as RANSAC.

Geometric or even semantic properties proved to be very helpful in order to find reliable features and matching pairs. Guided Matching based on Statistical Optical Flow [Maier et al., 2016] speeds up the matching process by constraining the search space by first estimating spatial statistics from a small subset of matched and filtered correspondences. The main motivation is that depending on the scene, keypoints rejected due to high local but low global response might be more interesting to be kept for matching than the ones selected with a global threshold. Instead, in both images local non-maxima suppression is applied to reduce the number of keypoints and ensure a good distribution over the whole image. Then, Statistical Optical Flow is used to guide the matching process by reducing the search range to a small area.

GMS [Bian et al., 2017] (Grid-based Motion Statistics) is a filter approach without using RANSAC. It incorporates motion smoothness constraint using several support points within the keypoint's neighborhood. Even when using very efficient feature types, such as ORB features, a large set of reliable and robust matches can be obtained.

To improve the keypoint detection process, a series of methods was proposed to guide some sort of learning process of keypoint detection using certain applications (such as SfM). The idea is to utilize already successfully applied keypoints to train new ones. [Winder, 2007] uses a training set consisting of patches from a multi-image 3D reconstruction with accurate ground-truth matches to improve the detection accuracy by learning parameter choices such as smoothing factor, pooling and histogram dimension of existing descriptors such as SIFT or GLOH. [Perd'och et al., 2009] propose to learn discretized local geometry representation based on minimization of the average reprojection error in the space of ellipses. The proposed representation is designed, besides compactness, to seamlessly support the assumption that the gravity vector – a vector in an image pointing towards the gravity direction which is preserved – is useful for fixing the orientation ambiguity of affine covariant points and for geometric verification [Philbin et al., 2007]. [Hinterstoisser et al., 2008] propose a learning-based approach for perspective rectification of patches followed by a patch identity classification and its sub-pixel precise position estimation. [Strecha et al., 2009] learn to identify keypoints with high repeatability using a pre-aligned training set which allows them to filter keypoints based on the classifier score. [Hartmann et al., 2014] collect matchable keypoints by observing which keypoints are retained throughout the SfM pipeline and learn these keypoints. [Richardson and Olson, 2013] learn convolutional filters through random sampling and looking for the filter which gives the smallest pose estimation error when applied to stereo visual odometry.

3 Learning local invariant features

In the previous sections, we presented a significant amount of work on manually describing interest points and image regions (*i.e*. handcrafted approaches). Even optimization and filtering approaches leverage real-world constraints which can be somehow described by a human (neighborhoods, relations, etc.). In this sections, however, we will focus on methods which try to avoid manual descriptions of the world and try to train algorithms to understand keypoints and descriptors by themselves.

Results on different benchmark datasets have shown that while SIFT [Lowe, 2004] and its extensions are still dominating the field, depending on the dataset some features perform better or worse than others. As shown in [Balntas et al., 2017], there are also contradicting conclusions reported in literature concerning feature comparison even if evaluating the same descriptors on the same benchmark. This is mainly due to different implementations and variations of the implicit parameters of the feature detectors. These clearly shows that the selection of feature types heavily depends on the target application.

As an alternative to selecting the best handcrafted features, data driven descriptors obtained with learning methods were proposed. They allow to thoroughly optimize descriptors for the given dataset and, hence to sometimes significantly outperform the handcrafted features. These learning-based methods can be unsupervised or supervised. For unsupervised methods, on the one hand, there is no need for labeled data, *i.e*. matched local keypoints, the features are learned directly from a large set of local patches extracted from the current dataset. Often, the main idea is to adapt handcrafted features, *e.g*. by projecting them in some low dimensional space, to be better suited to the new dataset. Supervised learning methods, on the other hand, require labeled data, *i.e*. matched keypoints or local image patches, and exploit the correspondences in order to learn new representation. Such correspondences can easily be generated automatically by leveraging geometric consistency, *e.g*. considering projections of the same 3D data (of reconstructed dense surface models or synthetic 3D objects) into several images.

In Table 1 we list some of the most popular datasets used to train or evaluate local feature descriptors: Oxford-Affine[10], Photo-Tourism[11], Fountain&Herzjesu[12], Cornell BigSfM[13], 1DSfM[14], RomePatches[15] and HPatches[16] In some cases only the images, camera poses, and 3D models are available from which corresponding keypoints can easily be deduced by projecting the 3D points onto the image planes. Using such datasets, learning-based methods can be built to improve the keypoint detection or to learn better local descriptors. In this section we only review methods that do not use deep learning, because Section 4.3 will present them in detail.

[10] Oxford-Affine available on `http://www.robots.ox.ac.uk/~vgg/data/data-aff.html`

[11] Photo-Tourism dataset page `http://matthewalunbrown.com/patchdata/patchdata.html`

[12] Fountain&Herzjesu can be downloaded from `https://cvlab.epfl.ch/data/keypoint`

[13] Cornell BigSfM available on `http://www.cs.cornell.edu/projects/bigsfm/`

[14] 1DSfM dataset page `http://www.cs.cornell.edu/projects/1dsfm/`

[15] RomePatches available on `http://lear.inrialpes.fr/people/paulin/projects/RomePatches/`

[16] HPatches data and benchmark on `https://github.com/hpatches`

Table 1 Popular datasets proposed for local feature detector and descriptor training/evaluation.

Dataset	Provided ground truth	Reference
Oxford-Affine	homographies	[Mikolajczyk and Schmid, 2003]
Photo-Tourism	corresponding patches	[Winder, 2007]
Fountain&Herzjesu	visibility and optical flow	[Strecha et al., 2008]
Cornell BigSfM	3D points , tracks, camera info	[Crandall et al., 2013]
RomePatches	corresponding patches, image labels	[Li et al., 2014, Paulin et al., 2015]
1DSfM	3D points and camera info	[Wilson and Snavely, 2014]
HPatches	corresponding patches, homographies	[Balntas et al., 2017]

3.1 Learning local descriptors

The main idea behind using learning local descriptors is to optimize the descriptors for the current dataset. This can be done in an unsupervised or supervised manner, where the former in general acts on the data distribution, while the latter exploits known descriptor correspondences to learn a better representation. These methods use handcrafted keypoint detection and directly act on the descriptors by learning a transformation from the handcrafted descriptors to a new space in which the discriminative power of the features and the matching accuracy is improved.

Among the early descriptor learning methods we find the unsupervised PCA-SIFT [Ke and Sukthankar, 2004] which uses principal component analysis (PCA) to embed a gradient image of a patch into a new space, or supervised methods that use randomized trees [Lepetit and Fua, 2006], and boosting [Babenko et al., 2007] in order to learn feature representations from matching and non-matching local patch pairs.

[Brown et al., 2010] proposed a model which builds on top of handcrafted low level features, such as steerable filter banks or gradient orientation map with different spatial pooling, *e.g*. used by SIFT-like or DAISY-like descriptors. Considering both linear and non-linear transforms for dimensionality reduction, they make use of discriminant learning techniques such as Linear Discriminant Analysis (LDA) and Powell minimization in order to select the pooling parameters and to obtain low dimensional representations (see Figure 4).

[Philbin et al., 2010] learn both linear and non-linear discriminative projections into lower dimensional spaces based on a margin-based cost function, which aims to separate matching descriptors from non-matching ones. Training data is generated automatically by leveraging geometric consistency.

ConvOpt[17] features proposed in [Simonyan et al., 2014] encode non-linearity into the procedure for mapping intensity patches to descriptors with the goal of learning descriptors whose similarity with respect to a chosen distance metric match the ground truth. They obtain sparse feature representation with Regularized Dual Averaging [Xiao, 2010] well suited to non-smooth sparsity-inducing cost functions. The authors further propose a weakly supervised extension, where additionally unlabeled data is exploited using an objective to be optimized inspired by the large margin nearest neighbor (LMNN) approach [Weinberger and Saul, 2009].

[17] Code available for ConvOpt on `http://www.robots.ox.ac.uk/~vgg/research/learn_desc/`

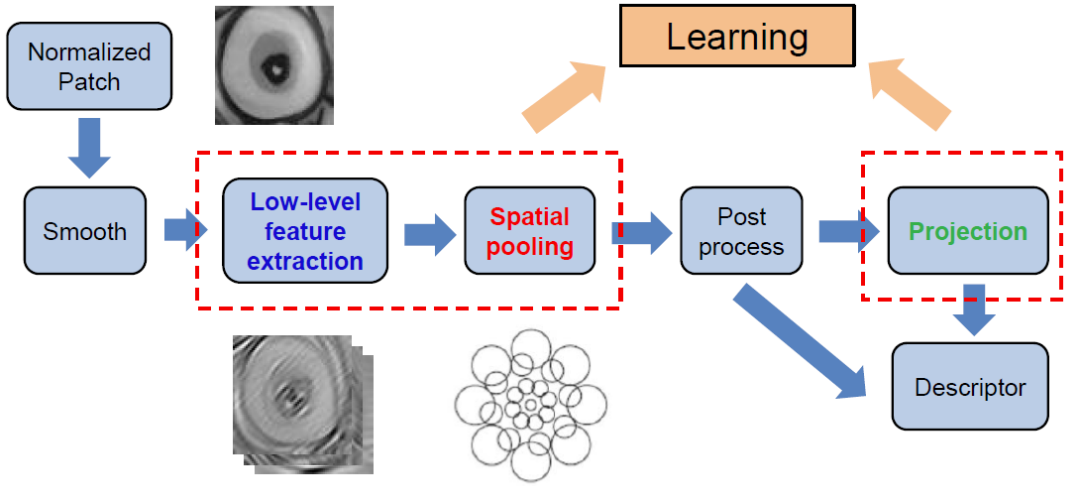


Fig. 4 Discriminative descriptor learning by optimizing spatial pooling and feature embedding in [Brown et al., 2010]. (Courtesy to Bin Fan)

[Wang et al., 2014] inherently encodes local information of multi-view patches, making it robust to affine distortions while maintaining a high discriminative ability. To this end, PCA is used to represent affine warped patches, assuming that the projection of vectors of various affine-warped patches of the same keypoint are represented by the same point in the low-dimensional linear subspace. The descriptors are obtained by a subspace-to-point mapping.

Only a few papers addressed the problem of local descriptors that are robust to more general deformations, than affine. The limitations of the affine-invariant descriptors when solving correspondences between images of objects that have undergone non-rigid deformations can be compensated twofold: On the one hand by enforcing global consistency [Cheng et al., 2008, Sanchez-Riera et al., 2010, Torresani et al., 2008] or on the other hand by introducing segmentation information within the descriptor itself [Trulls et al., 2013] and then solving complex optimization functions to establish matches. A better option is to build deformation invariant local descriptors. As such, DaLI[18] (Deformation and Light Invariant) [Simo-Serra et al., 2015a] uses diffusion geometry to build a descriptor for 2D image patches which is invariant to non-rigid deformations and photometric changes. The patches are described in terms of the heat they dissipate onto their neighborhood over time. To obtain compact descriptors, PCA dimensionality reduction is used. The DaLI descriptor can simultaneously handle relatively complex photometric and spatial warps and hence is interesting in the case of keypoint detection on deformable objects.

[18] Code for DaLI available on `http://www.iri.upc.edu/people/esimo/research/dali/`

3.2 Learning binary features

As we have seen, main concerns with regard to invariant features are localization accuracy, invariance to geometric and photometric deformations, and distinctiveness to be correctly matched against a large number of features. However, with the spread of mobile and embedded vision systems, the demand for efficient detection and matching of image features has increased. Moreover, for mobile applications, the amount of data sent over the network needs to be as small as possible so as to reduce latency and lower costs. Binary descriptors are of particular interest since they provide numerous advantages over their floating-point competitors, namely low memory footprint and faster matching.

One possible solution to build binary descriptors is to rely on existing float-valued descriptors which are then binarized through quantization [Gong and Lazebnik, 2011] or hashing techniques [Strecha et al., 2012]. An other alternative is to start from the raw image patch and learn directly the binary encoding.

Interesting work on binary region representations learning was proposed in the context of face recognition. For example, [Lu et al., 2015] introduce a compact binary feature descriptor (CBFD) which learns binary face descriptors for face representation and recognition. The authors first extract pixel difference vectors and then learn how to map them into a low dimensional space. In [Lu et al., 2017], this projection into low dimensional space as well as the dictionary for the encoding are learned jointly. This is further extended to coupled simultaneous local binary feature learning and encoding (C-SLBFLE), where coupled pixel difference vectors are extracted from a pair of aligned face images captured from different modalities (*e.g*. RGB and Near Infrared) such that each feature mapping is composed of a common projection matrix and a modal-specific projection matrix. The final codebook and projection matrices are then used to extract the face representations during testing stage. [Duan et al., 2017b] exploit contextual information of binary codes as strong prior knowledge to enhance the robustness of the features. In contrast to these unsupervised approaches, [Lei et al., 2014] proposed a discriminant face descriptor (DFD) method by learning an image filter using labeled data and LDA criterion to obtain LBP-like features.

More relevant for applications such as SfM or 3D reconstruction are LDAHash, D-BRIEF, RI-LBD, BinBoost, BOLD or RMGD: LDAHash [Strecha et al., 2012] compute a projection matrix for handcrafted features, *e.g*. SIFT or SURF, using Linear Discriminant Analysis (LDA) by jointly minimizing the in-class covariance and maximizing the covariance across classes. Then, optimal thresholds are computed that turn the projections into binary vectors. In the case of D-BRIEF[19] [Trzcinski and Lepetit, 2012] the training data is used to learn linear projections that map image patches to a more discriminative subspace. In order to obtain binary descriptors, the projected patches are simply thresholded. To obtain Rotation-Invariant Local Binary Descriptors (RI-LBD), [Duan et al., 2017c] propose to jointly learn a rotational function for each RBP and a feature mapping to project each image patch into binary codes.

BBoost[20] features [Trzcinski et al., 2015] are low-dimensional but highly discriminative descriptors computed with a boosted binary hash function. They use weak learners that were inspired from existing, prevalent keypoint descriptors such as BRIEF [Calonder et al., 2010] and orientations of intensity gradients. Weighting and combining multiple weak learners builds a highly non-linear mapping from gradients to robust descriptors. Similarly, the Receptive Fields Descriptor (RFD) [Fan et al., 2014a] uses gradient orientation maps

[19] D-BRIEF code available on `https://cvlab.epfl.ch/research/detect/dbrief`

[20] Source code for BBoost is available on `https://cvlab.epfl.ch/research/detect/binboost`

summed over rectangular shaped receptive fields, but instead of selecting them by a boosting process, RFD selects the receptive fields by a greedy approach from a large number of candidates according to their distinctiveness and correlations. In addition to rectangular pooling area they also consider Gaussian pooling area which allows them to obtain some improvement at a slightly increased cost.

BOLD[21] (Binary Online Learned Descriptors) [Balntas et al., 2015] are features adapted online for each patch using a set of synthesized views. The global selection of discriminative dimensions and the correlation between intra-class distances is done offline from a large set of possible binary tests and random patches.

RMGD (Ring-based Multi-Grouped Descriptor) [Gao et al., 2015] consists of a new pooling configuration based on spatial ring-region sampling, allowing to involve binary tests on the full set of pairwise regions with different shapes, scales, and distances. A circle integral image is used for fast calculation of the binary descriptor. An efficient bit selection by emphasizing high variance and low correlation is achieved with an extended Adaboost yielding highly compact representation. To increasing discriminativeness and robustness, the RMGD is built with multiple image properties, such as intensity, x-partial, y-partial, gradient magnitude, orientation, soft assigned gradient orientations, that are binarized and combined with a RankSVM or sparse SVM to efficiently leverage the complementary properties among those various feature groups.

4 Deep learning based local invariant features

Deep learning refers to training and inference of convolutional neural networks (CNN) which consist of multiple layers (deep). Automatically learning features at multiple levels of abstraction, as done in the human brain, allows a system to learn complex functions mapping the input to the output directly from data without depending completely on human-crafted features. Therefore, deep learning models are a succession of convolution layers consisting of neurons with learnable weights and intermediate layers introducing non-linearities, pooling to downsample, batch normalization or drop out layers setting activations randomly to zero. The role of the intermediate layers is to introduce redundancy through training, compact or average information and to avoid overfitting. The system is trained iteratively by back-propagation, with samples of training data called batches. During inference (or at test-time), the CNN processes the input, in this paper usually an image, and produces the output it was trained for.

Such deep learning models have revolutionized image-level tasks such as classification or retrieval, and obviously pushed the community to revisit patch-level tasks that were still relying on handcrafted features and detectors. Some exceptions are mentioned in Section 3. Therefore in the last few years numerous deep learning based approaches were proposed to either learn local descriptors (floating point or binary) or to learn the local keypoint detection and description pipeline end-to-end.

[21] Open source implementation of BOLD is available at `http://vbalnt.io/projects/bold/`

4.1 Learning local descriptors with CNNs

Inspired by [Donahue et al., 2014] which have shown that activation layers of CNNs can be used successfully as image level descriptors for various task, several papers considered to derive patch-level descriptors from such architectures [Fischer et al., 2014, Long et al., 2014, Paulin et al., 2015]. In [Fischer et al., 2014], the network is trained with surrogate data generated from unlabeled images, where a class is considered for every randomly extracted image patch. Each class is virtually augmented by random transformations applied to the extracted patch. [Long et al., 2014] observed that while the activations of the penultimate layer are the ones most used as image-level descriptor, the output of earlier layers actually encode more task-independent information and, thus, are more suitable to be used as keypoint descriptors. The Patch-CKN based descriptor [Paulin et al., 2015] is an unsupervised method based on the kernel feature map of a convolutional match kernel. A fast and simple stochastic procedure is proposed to compute a finite-dimensional explicit feature embedding used to approximate the kernel feature map.

The tendency of recent methods is to use Siamese CNN networks to learn discriminant patch representations from a large set of known pairs of corresponding and non-corresponding patches (see Figure 5). As such, [Jahrer et al., 2008] inspired by [Chopra et al., 2005], is among the first papers which proposed to use a Siamese CNN to learn keypoint descriptors, years before the explosive success of CNNs in computer vision. To be able to adjust the amount of desired invariance and distinctiveness, the model was trained on reliable keypoints, extracted from a synthetically generated data set.

DDesc[22] [Simo-Serra et al., 2015b] learns 128 dimensional descriptors whose L2 normalized Euclidean distances reflect patch similarity, and which can be used as a drop-in replacement for any task involving SIFT (see also Figure 5). Based on the observation that after a certain point of learning most pairs are correctly classified and they do not bring improvement anymore, they propose a strategy of aggressive mining of hard positives and negatives. This is done by selecting, after each forward-propagation, non-corresponding pairs that are hard to discriminate and corresponding pairs that match poorly and back-propagate them through the network in order to update the weights.

Similarly, L2-Net[23] [Tian et al., 2017] uses CNNs to learn high performance descriptors in Euclidean space. The L2-Net transforms a batch of patches into a batch of descriptors. The aim for each descriptor is that the nearest neighbor in a batch is the correct matching descriptor. The training of L2-Net is built on a progressive sampling strategy and a loss function consisting of three error terms: one accounts for the relative distance among descriptors, one controls descriptor compactness, and one is an extra supervision imposed on the intermediate feature maps. It is shown that incorporating supervision information from the first and last convolutional layers into the learning objective allows for a better generalization and the compactness constrain reduces the risk of overfitting. As the output of L2-Net approximates a Gaussian distribution, by applying the sign function to it they obtain the Binary L2-Net producing binary descriptors.

---

[22] Torch7 code and pretrained models for DDesc are available on `https://github.com/etrulls/deepdesc-release`

[23] Matlab implementation of L2-Net with pre-trained models are available on `https://github.com/yuruntian/L2-Net`

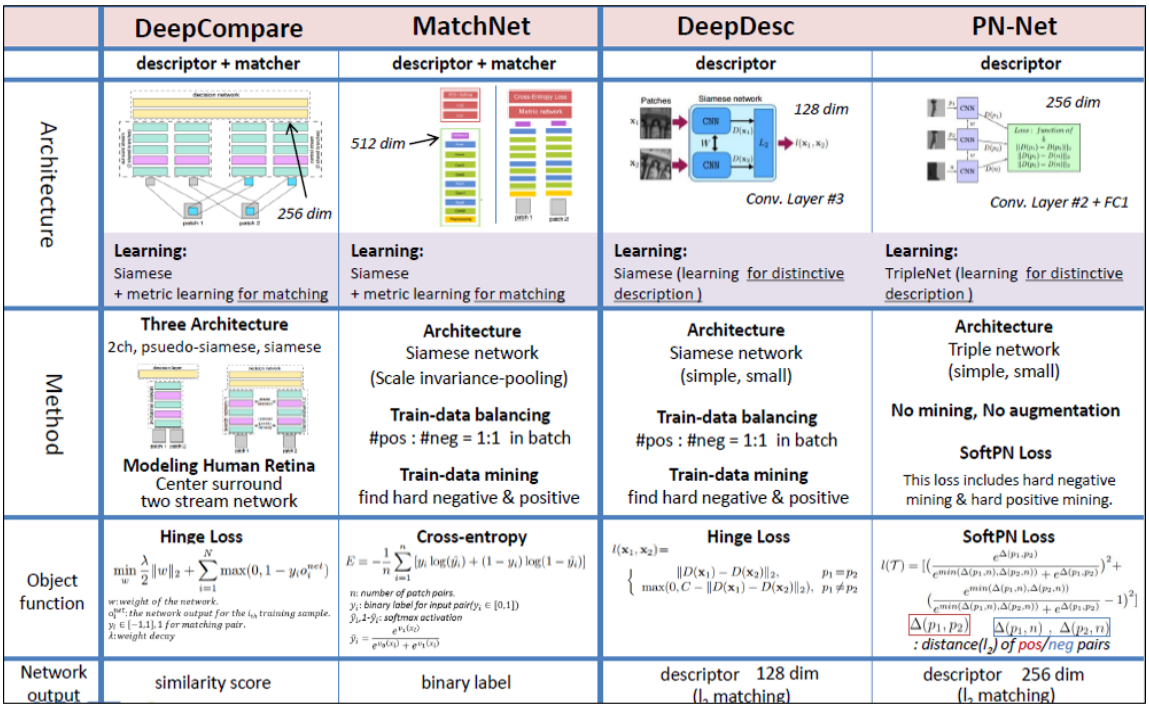


Fig. 5 An overview of some of the first Siamese CNN-based local features learning methods. From left to right, DeepCompare [Zagoruyko and Komodakis, 2015], MatchNet [Han et al., 2015], DDesc [Simo-Serra et al., 2015b] and PN-Net [Balntas et al., 2016a]. (Courtesy to Yukyung Choi)

HardNet[24] [Mishchuk et al., 2017] has a L2-Net architecture, but it has a new loss for metric learning which relies on triplets instead of pairs. In each batch the distance matrix (distances between all patches) is computed using the current descriptors. A triplet is formed by taking a pair of positive patches and adding a negative patch which is closest to one of the two patches. This can be seen as a local hard negative mining strategy. Note also that the way the triplets are formed does not require the network to be three-streamed in contrast to other triplet based approaches such as TFeat [Balntas et al., 2016b] or TGLoss [Kumar B. G. et al., 2016] which are described below.

Similarly, the structured LSSS (Lifted Structured Similarity Softmax) loss [Song et al., 2016] considers all the possible matching and non-matching pairs in one batch of samples and focuses on the hard pairs in training a batch, while the N-pair loss [Sohn, 2016] considers batches of one positive and $N-1$ negative pairs and considers them jointly in an $(N+1)$-tuplet loss.

While DDesc and L2-Net use simple L2 distance to compare patches, MatchNet[25] [Han et al., 2015] methods learn jointly the descriptor and the metric by combining a CNN with multiple convolutional and

[24] HardNet code and pretrained model are available at `https://github.com/DagnyT/hardnet`

[25] MatchNet code and pre-trained model available at `http://www.cs.unc.edu/~xufeng/matchnet`

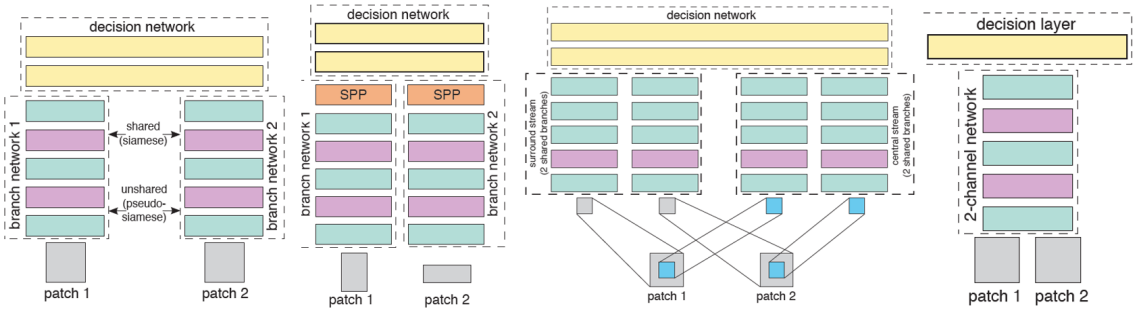


Fig. 6 Various network architectures explored in [Zagoruyko and Komodakis, 2015]. From left to right: Siamese networks, Pseudo-Siamese networks, SPP-Siamese network, central-surround two-stream network, and the 2-channel network. (Courtesy to Sergey Zagoruyko)

spatial pooling layers plus an optional bottle neck layer to obtain feature vectors followed by a metric network to learn the similarity (see details in Figure 5) The pooling layers increase the robustness of the network caused by variations due to scale changes in wide-baseline matching. The network is trained with a cross-entropy loss that transforms the matching problem into a classification problem. Note that this architecture has similarities with the one in [Zbontar and LeCun, 2016], which was design to learn the similarity between patches along epipolar lines in rectified pairs of images and is used to generate the matching cost for stereo algorithms. A drawback of MatchNet is that while it shows improved matching accuracy, it is not obvious how to combine it with fast approximate nearest neighbor algorithms, like kd-trees, for search.

Similarly, DeepCompare[26] [Zagoruyko and Komodakis, 2015] uses a CNN-based model that is trained to account for a wide variety of changes in image appearance on top of which adds a metric network to learn similarity function for a pair of patches. An interesting aspect of the paper is that it explores a variety of different neural network models adapted for representing such a function and it highlights network architectures that offer improved performance. They experimented with Siamese networks, SPP-Siamese networks, Pseudo-Siamese networks, 2-channel networks as well as Central-Surround Two-Stream networks (see Figure 6). The Pseudo-Siamese network, in contrast to Siamese network where the weights are shared between the two streams, has two branches with uncoupled streams with non-shared weights allowing for more flexibility. In the case of the 2-channel network, the patch pair is directly fed to the network as a 2-channel image and hence they are processed jointly. The Spatial Pyramid Pooling (SPP) Siamese network inserts a spatial pyramid pooling layer between the convolutional layers and the fully-connected layers of the network. Such a layer allows aggregation of the features of the last convolutional layer through spatial pooling, and therefore allows to handle patches of different sizes as input. Finally, the Central-Surround Two-Stream network consists of two separate streams, central and surround, which enable processing in the spatial domain which takes place over two different resolutions. The 2-channel-based network clearly outperformed the others in the conducted experiments on patch matching and wide baseline stereo. Nevertheless, its main drawback is that, compared to the others, it is computationally more expensive at test time where all combinations of patches have to be tested against each other in a brute-force manner.

[26] Source code and trained models are available at `http://imagine.enpc.fr/~zagoruys/deepcompare.html`

PN-Net[27] [Balntas et al., 2016a] proposes to train the network with positive and negative pairs formed by triplets of patches and introduce a new loss function, called SoftPN, which simultaneously exploit the constraints given by the positive and negative pairs (see Figure 5). This idea was continued by TFeat[28] [Balntas et al., 2016b] trained either with a triplet margin ranking loss or a triplet ratio loss, with random sampling of the patch triplets and in-triplet hard negative mining using anchor swap.

Inspired by PN-NET, the DeepCD[29] [Yang et al., 2017] framework takes a descriptor learning architecture and augments it with an additional network stream for learning a complementary descriptor. To enforce the complementary property, a so called data dependent modulation (DDM) layer which dynamically learns the attenuation factors of the learning rate in the augmented stream conditioned on training data is used. It is involved only in the back-propagation and allows the two streams to be learned adaptively and jointly so that they can better collaborate to make predictions through late fusion.

TGLoss[30] [Kumar B. G. et al., 2016] is another triplet network, but its main characteristic is that it combines the traditional triplet loss with a global loss that aims to separate two empirical means of the matching pair, respectively non-matching pair, distances by a margin and to minimize their variances. Note however that the extra margin in global loss adds extra complexity for training.

[Wei et al., 2018] propose a Subspace Pooling (SP), which is invariant to a range of geometric deformations such as circular shift, flipping, in-plane rotation, *etc*. and a marginal triplet loss combined with the projection Gaussian kernel which helps to focus on hard examples during training. The basic idea of the SP is to model the convolutional feature maps with the linear subspace spanned by its principal components. They show that proposed SP is invariant to all the geometric changes that can be expressed as column permutations of the matrix formed by the feature maps from the last convolution layer stacked as one dimensional features similarly to the bilinear pooling function in [Lin et al., 2015], which is equivalent to spatial reordering. Integrating the proposed pooling with different architectures, such as HardNet or TFeat allows to improve the original model.

UCN[31] (Universal Correspondence Network) [Choy et al., 2016] is a deep learning framework for accurate visual correspondences which learns a metric space for accurate visual correspondences. Its particularity compared to previous approaches is that it can efficiently handle geometric correspondences, dense trajectories or semantic correspondences. Furthermore, the input in the network is a whole image, not only an extracted patch and hence all considered patches are processed simultaneously. The UCN consists of a series of convolutions, pooling, non-linearities, and a convolutional spatial transformer, followed by channel-wise L2 normalization and uses the correspondence contrastive loss as well as hard-negative mining strategy for training (see Figure 7). Inspired by the GPU implementation in [Garcia et al., 2010], they implement K-NN search as a Caffe [Jia et al., 2014] layer to actively mine hard negatives on-the-fly. The convolutional spatial transformer, proposed in [Jaderberg et al., 2015], was considered to make the features invariant to particular families of transformations. By learning an optimal feature space which compensates for affine transformations, the convolutional spatial transformer aims to mimic patch normalization of descriptors such as SIFT.

[27] Code for PN-Net available at `https://github.com/vbalnt/pnnet`

[28] Code for TFeat available at `https://github.com/vbalnt/tfeat`

[29] DeepCD code available at `https://github.com/shamangary/DeepCD`

[30] Matlab code and pretrained models on `https://github.com/vijaykbg/deep-patchmatch`

[31] The UCN code, license agreement, and pre-trained models available at `http://cvgl.stanford.edu/projects/ucn/`

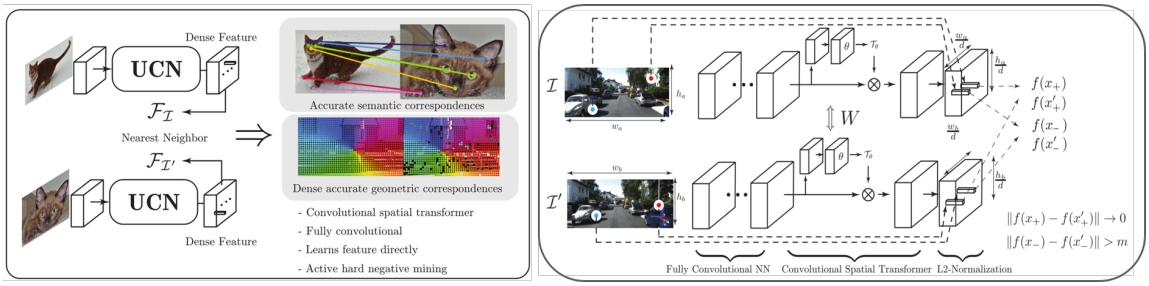


Fig. 7 The UCN architecture [Choy et al., 2016] which can accurately and efficiently learn a metric space for geometric correspondences, dense trajectories or semantic correspondences. It consists of a FCNN combined with a convolutional spatial transformer and followed by channel-wise L2 normalization. Features that correspond to the positive points (from both images) are trained to be closer to each other, while features that correspond to negative points are trained to be a certain margin apart. (Courtesy to Christopher B. Choy)

The model was evaluated on three different tasks: geometric correspondence, semantic correspondence, and accuracy of correspondences for camera localization. In each case the model was trained on corresponding datasets. At test time, the features are extracted densely or on a set of sparse keypoints (for semantic correspondence) from a query image and they find the nearest neighboring feature in the second image.

The main idea of the GOR[32] [Zhang et al., 2017b] approach is to force the features to be spread-out in the descriptor space in order to fully utilize the expressive power of the space. This is done through the introduction of a Global Orthogonal Regularization (GOR) which aims at maximizing the probability that two randomly sampled non-matching descriptors are close to orthogonal by forcing the sample mean of the inner product of the descriptors of non-matching pairs in the batch to be close to 0 and the second moment close to one over the dimension of the descriptor space. They have shown that adding this loss to the above models such as DDesc or TFeat yields a better patch matching and combined with LSSS [Song et al., 2016] allows learning of improved image-level embedding for clustering and retrieval tasks.

In contrast to pair-based or triplet-based approaches, DOAP (Descriptors Optimized for Average Precision) [He et al., 2018b] uses a deep neural network to optimize the Average Precision, which can also be seen as a list-wise learning to rank approach. The proposed framework can be used to optimize the average precision for both binary and real-valued local feature descriptors. While the baseline DOAP aims at improving the task-independent nearest neighbor matching, an augmented formulation (DOAP-ST) is also proposed where a Spatial Transformer module [Jaderberg et al., 2015] is added to effectively handle geometric noise and improves the robustness of matching.

[32] Tensorflow implementation available on `https://github.com/ColumbiaDVMM/Spread-out_Local_Feature_Descriptor`

4.2 Learning local binary descriptors with CNNs

The main advantage of binary descriptors is that the computational speed of the matching enables real-time performance in numerous applications and the low memory footprint allows easy storage on mobile devices and embedded systems or facilitate their transmission through a network. While handcrafted binary descriptors aim to uniquely encode the region or neighborhood of the keypoint using intensity differences, learned binary descriptors try to embed certain properties into binary space. We have already seen methods in Section 3.2, that learn data-dependent binary descriptors. Not surprisingly, deep methods were also designed to tackle this problem. On the one hand, some of the above mentioned methods in Section 4.1, such as L2-Net [Tian et al., 2017] or DOAP [He et al., 2018b], also include binary variants. However, as the binarization is learned on top of the non-binary features, those methods are not the most efficient concerning the feature extraction time.

On the other hand, several deep learning based methods specifically designed to learn binary features were proposed. DeepBit[33] [Lin et al., 2016] is an unsupervised deep learning approach to learn compact binary descriptors for efficient visual object matching. The main idea is to optimize the parameters of a network using back-propagation of a combination of three losses in order to obtain the binary descriptors. The aim of the first one is to force the model to generate a binary descriptor that preserve the local data structure by having as little quantization loss as possible after projecting the activations of the last layer into binary descriptors. The second term aims to encourage the binary descriptor to be evenly distributed and hence to be more discriminative. Finally, the third term makes the descriptor invariant to rotations and noise.

DBD-MQ (Deep Binary Descriptor with Multi-Quantization) [Duan et al., 2017a] is another unsupervised feature learning method which considers the binarization as a multi-quantization task and applies a K-AutoEncoders (KAEs) network to jointly learn the parameters and the binarization functions. With the fine-grained multi-quantization, similar elements of real-valued descriptors are clustered into the same class yielding to energy-saving and evenly distributive binary descriptors. The network is trained with an iterative two-step procedure.

Note that several deep hashing models were proposed to learn compact binary codes [Xia et al., 2014, Zhang et al., 2015, Liong et al., 2015, Zhao et al., 2015, Lai et al., 2015, Li et al., 2016, He et al., 2018a]. While these methods target mainly image level representations and their aim is to improve improve image search or classification, the deep hashing methods could also be used to learn local binary features. Note that nevertheless most of these deep hashing methods are supervised requiring pairwise labels or triplet labels.

4.3 End-to-end local feature keypoint detectors and descriptors

Only a few end-to-end deep models were proposed that aim to learn the local keypoint detection and description pipeline in a way that, similarly to the handcrafted pipeline (see Figure 1), estimates the canonical scale, the main orientation of the patch and also provides the descriptor of the normalized patch. And this in spite of the fact that, similarly to keypoint detectors, the early layers of CNNs are characterized by com-

[33] Code available at `https://github.com/kevinlin311tw/cvpr16-deepbit`

binations of filtering operations. Indeed, as some of the recent papers have shown, keypoint detectors can be efficiently implemented as CNNs and trained end-to-end. Another advantage of CNN implementation is that they can be implemented on GPGPUs which are characterized by ever-increasing speed and power efficiency or on FPGA (Field Programmable Gate Arrays). Indeed, [Febbo et al., 2018] proposes a sensor-based FPGA hardware implementation of a compact three-layer CNN with separable convolutional kernels that is able to emulate multiple keypoint detection algorithms, such as KAZE or SIFT, and where the keypoints are computed in a streaming-based fashion that does not require to store the entire content of the image, intercepting the sensor data stream at readout.

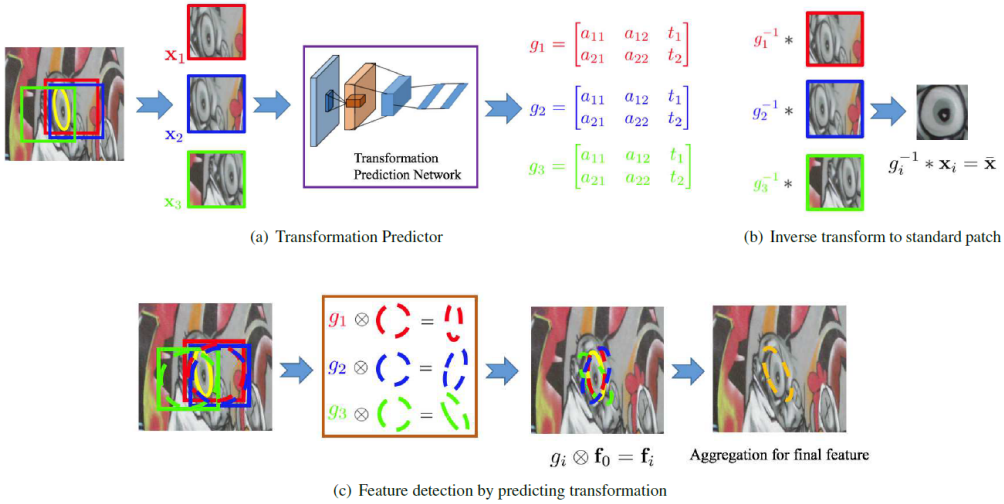
With a more complex architecture, TILDE (Temporally Invariant Learned Detector) [Verdie et al., 2015] is one of the first deep models which learns to detect repeatable keypoints under drastic imaging changes due to weather and lighting conditions to which state-of-the-art keypoint detectors are very sensitive. By identifying good keypoint candidates in multiple training images taken from the same viewpoint, a piece-wise linear regressor is trained to predict a score map whose maxima are those points so that they can be found by simple non-maximum suppression (NMS). The proposed objective function that is minimized has three terms, a classification-like loss to separate well the image locations that are close to keypoints from the ones that are far away, a shape regularizer loss enforcing the response of the regressor to have a specific shape at these locations and a temporal regularizer loss to enforce the repeatability of the regressor over time. To optimize the objective function they experimented with both boosted regression trees and a CNN using an architecture similar to LeNet-5 [LeCun et al., 1998].

CovDet (Covariant Point Detector)[34] [Lenc and Vedaldi, 2016] is a CNN architecture proposed to detect local features that are preserved under viewpoint changes. To achieve this, an objective function is proposed in terms of a covariance constraint which is anchor-agnostic and the detection is formulated as a regression problem solved using a deep networks. The geometric transformation is parameterized as a matrix, and the Frobenius norm on the matrix difference is used as the distance of two transformations. By specifying the group of transformations to which the detector should be invariant in the covariance constraint, different detector types can be derived in this framework, such as a corner detector covariant to full affine transformations or an orientation detector which is covariant with rotation and translation.

TCovDet[35] (Transformation Covariant Local Feature Detectors) [Zhang et al., 2017a] is an extension to CovDet proposed where the concepts of the "*standard patch*" and "*canonical feature*", used in the handcrafted models, are embedded into the training pipeline (see Figure 8). To find *standard patches* that are discriminative and diverse enough for learning to regress the target transformation group, the results of the detector are used as anchors, and the *canonical feature* is defined as the central point of the standard patch (for point detection) or the inscribed circle in the standard patch (for blob detection). To detect all the features in the whole image, the transformation predictor is applied at all image locations. The canonical feature in all the image patches define a dense grid in the image and the predicted transformation moves the canonical feature to the closest local feature and non-maxima suppression is applied to choose the final keypoints. Since the neural network can only process image patches with a fixed size, features are extracted on a multi-level image pyramid. The authors have shown on several large scale datasets that TConvDet allows to significantly improve the keypoint repeatability compared to other detectors as well as the matching score obtained with

[34] Matlab code for CovDet is available on `https://github.com/lenck/ddet`

[35] Tensorflow code for TCovDet is available on `\https://github.com/ColumbiaDVMM/Transform_Covariant_Detector`



(a) Transformation Predictor (b) Inverse transform to standard patch

(c) Feature detection by predicting transformation

Fig. 8 Illustrating the TCovDet [Zhang et al., 2017a] keypoint detection framework. The Transformation Prediction Network (a) predicts the geometric transformation of the patch such that the inverse of the predicted transform warps the image patch to the "*standard patch*" (b). The predicted transformations are used to get "*canonical features*" and the output from multiple patch candidates are aggregated to predict the most likely location and shape of the keypoint and the corresponding descriptor (c). (Courtesy to Xu Zhang)

SIFT descriptor computed for the detected keypoints.

LIFT[36] (Learned Invariant Feature Transform) [Yi et al., 2016a] is a network which is composed by three CNN components that feed into each other (see Figure 9): the Detector (DET), the Orientation Estimator (ORI), and the Descriptor (DESC). They are concatenated by Spatial Transformers [Jaderberg et al., 2015] to rectify the image patches between the input and output of the networks while preserving differentiability. As DET, the authors use a modified version of TILDE [Verdie et al., 2015], where instead of using non-maximum suppression, softargmax function (which computes the center of mass with the weights being the output of a standard softmax function) on the score map which acts as a differentiable version of non-maximum suppression. ORI relies on the CNN proposed in [Yi et al., 2016b] which is trained to estimate consistent and optimal orientations for matching purposes[37]. This is achieved by assigning a canonical orientation to feature points given an image patch centered on the feature point using a Siamese networks which implicitly finds the optimal orientations during training. In the paper it has been shown that the model can successfully be deployed with different descriptors. Finally, DDesc [Simo-Serra et al., 2015b] is used for the DESC component. It is a simple network, which does not require learning a metric but uses L2 pooling and local subtractive normalization.

During training, first each component network (DET, ORI, DESC) is trained independently and then the parameters are fed as initialization for the whole network which is trained end-to-end with quadruplet of

[36] Code and pre-trained models are available on `https://github.com/cvlab-epfl/LIFT`

[37] Code available on `https://github.com/cvlab-epfl/learn-orientation`

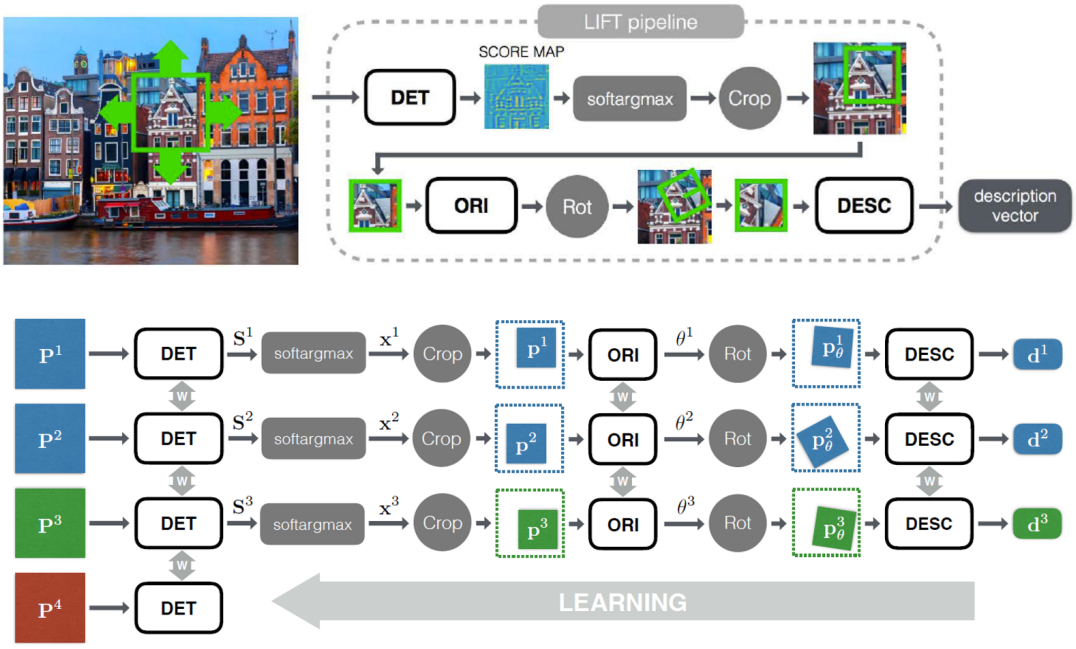


Fig. 9 Top: The LIFT [Yi et al., 2016a] pipeline. Bottom: the four branches Siamese training architecture that considers quadruplet of patches, where $\mathbf{P}^1$ and $\mathbf{P}^2$ (blue) correspond to different views of the same physical point, and are used as positive examples to train the descriptor (DESC), $\mathbf{P}^3$ (green) shows a different 3D point which serves as a negative example for DESC and finally $\mathbf{P}^4$ (red) contains no distinctive feature points and is only used as a negative example to train the detector (DET). Given a patch $\mathbf{P}$, DET followed by softargmax, and Crop provide a smaller patch $\mathbf{p}$ inside $\mathbf{P}$, that is fed to the Orientation Estimator (ORI) to provide the rotated patches as input for DESC. (Courtesy to Kwang Moo Yi)

patches (see Figure 9). During test time, the image at different scales is fed to the network, hence DET will generate score maps at each scale and considering traditional NMS at each, keypoints of different scales can be detected. ORI and DESC is then run only on patches at the detected keypoint locations.

SuperPoint [DeTone et al., 2018] is a self-supervised framework for training interest point detectors and descriptors suitable for a large number of multiple-view geometry problems in computer vision. It operates on full-sized images and jointly computes pixel-level interest point locations and associated descriptors in one forward pass. As shown in Figure 10, first a base keypoint detector is trained on examples from a synthetic dataset consisting of simple geometric shapes where the keypoint locations are well defined. Then, in order to adapt this detector to more complex images, a process called Homographic Adaptation is designed with the aim to boost the geometric consistency of the keypoint detector. It acts during the training and it consists in detecting additional keypoints in a set of randomly wrapped images generating pseudo-ground truth interest points from the wrapped images that are added to the training set. Finally, the SuperPoint network, that involves an encoder that is followed by a decoder pair, one for the interest point detection and one for the description, is trained with pair of images and with joint optimization of the detection losses and

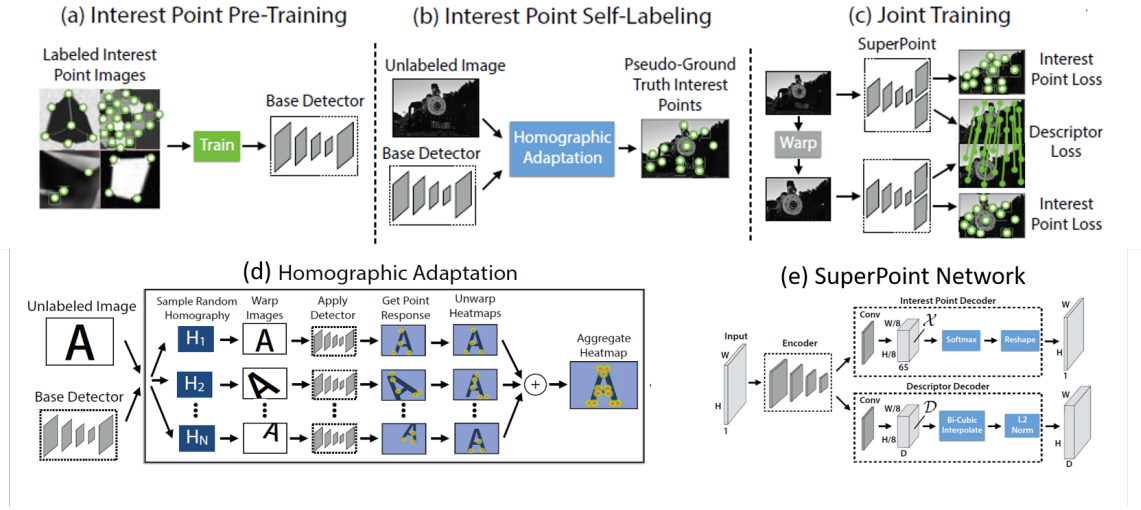


Fig. 10 Self-Supervised training overview (a,b,c) of the SuperPoint [DeTone et al., 2018] framework. An initial interest point detector is pre-trained using synthetic data (a), then the Homographic Adaptation (d) procedure automatically generates pseudo-ground truth interest points as shown in (b), which then are used to train the SuperPoint Network (e) that jointly extracts interest points and descriptors from an image, as shown in (c). (Courtesy to Daniel DeTone)

the descriptor loss. The detection loss is a cross-entropy loss, where labels are derived from the ground-truth interest point locations, and the ground truth for the descriptor loss are derived from the pseudo-ground truth interest points correspondences derived by the Homographic Adaptation process.

In contrast to the above models, DELF[38] (Deep Local Features) [Noh et al., 2017] is a network trained with image-level annotations and it was designed mainly for instance-level image recognition/retrieval. It aims to detects and describes semantic local features which can be geometrically verified between images showing the same object instance. To identify semantically useful local features an attention mechanism is designed for keypoint selection, which shares most network layers with the descriptor.

5 Discussion

Since the aim of this paper is to give a comprehensive overview of local invariant features and a suitable evaluation under fair conditions would not be suitable, we refer to existing work for discussion of the described methods. [Heinly et al., 2012, Balntas et al., 2017, Maier et al., 2017, Schönberger et al., 2017] extensively evaluate and compare many of the popular methods using various benchmark datasets as well as considering different tasks, such as image matching, retrieval, or image-based 3D reconstruction. From these evaluations, we can see that there is no method which outperforms the others independently of data

[38] DELF code available at `https://github.com/tensorflow/models/tree/master/research/delf`

or task. However, together with a good understanding of the topic, these evaluation papers (ensuring dataset and task variability and that the comparison is fairly done) provide very useful guidance to select among the best methods for a given task.

For example, [Heinly et al., 2012] analyzed the performance of different handcrafted detector and descriptor pairings on several datasets containing strong illumination changes, viewpoint changes, geometric variations, blur, compression, *etc*. focusing mainly on binary features. They have found that among the detectors FAST [Rosten and Drummond, 2006], followed by SIFT [Lowe, 2004], BRISK [Leutenegger et al., 2011] and SURF [Bay et al., 2006] had the highest entropy concerning the distribution of keypoints in an image which can be attributed to its good spread of points and the shear number of detections that occurred. Concerning the descriptors, when non-geometric transformations were applied to the test images, such as blur, compression, exposure, or illumination change, BRIEF [Calonder et al., 2010] has a better selectivity (measured by the putative match ratio) and Recall (counting how many of the possible correct matches were actually found), but lower Precision (number of correct matches) than ORB [Rublee et al., 2011] and BRISK, or sometimes SURF or SIFT. In the case of geometric transformation (scale, rotation, affine), SIFT was the best concerning all measures, but ORB and BRISK still performed well compared to BRIEF which performs poorly under large rotation or scale changes. In the case of combined scale and rotation changes, BRISK took the lead of the binary descriptors concerning selectivity and precision being close to SIFT and SURF performance, but with much lower recall. In the case of viewpoint change BRIEF seems to beat BRISK and ORB on recall and selectivity but has lower precision than BRISK. Again best robustness concerning the viewpoint was obtained with SIFT.

[Maier et al., 2017] proposes to first semi-automatically generate ground truth matches and compare the methods in addition to the traditionally used precision and recall, with matching accuracy and fall-out. The semi-automatic annotation process based on local patch matching constraints, was first validated on synthetic datasets achieving high accuracy. Using these ground truth annotations (matches), several keypoint detector and descriptor combination were evaluated and compared using FLANN [Muja and Lowe, 2014] to find the matches. The combinations were used to test the matching accuracy on flow and disparity datasets as well as on usual benchmarks testing keypoint matching between image pairs with various geometric and photometric distortions. Tests on keypoint-descriptor combinations showed that modern binary descriptors achieve comparable results in terms of quality but significantly lower processing times compared to real valued descriptors. Best average accuracy was obtained by a combination of BRISK [Leutenegger et al., 2011] detector with FREAK [Alahi et al., 2012] binary descriptor. However, the optimized ConvOpt [Simonyan et al., 2014] feature descriptor was consistently on the top independently of the keypoint detector used.

[Mishkin et al., 2015] propose WXBS dataset[39] of ground-truth *wide baseline* image pairs with a combination of different geometry, illumination, sensor, appearance or modality changes and evaluate several handcrafted keypoint detector and descriptor combinations on image pairs including matching between visual spectrum images against near infrared images or Long-Wave infrared images, different modes of magnetic resonance imaging or even between map to satellite view correspondences (see examples in Figure 11). They show that this is a very challenging benchmark and that using only one detector-descriptor model at the time does not perform well. The best results are obtained with models that incorporate multiple detectors such as [Yang et al., 2007] or MODS [Mishkin et al., 2014, 2015]. Comparing single detectors, (adaptive)

[39] The WXBS dataset `http://cmp.felk.cvut.cz/wbs/index.html`

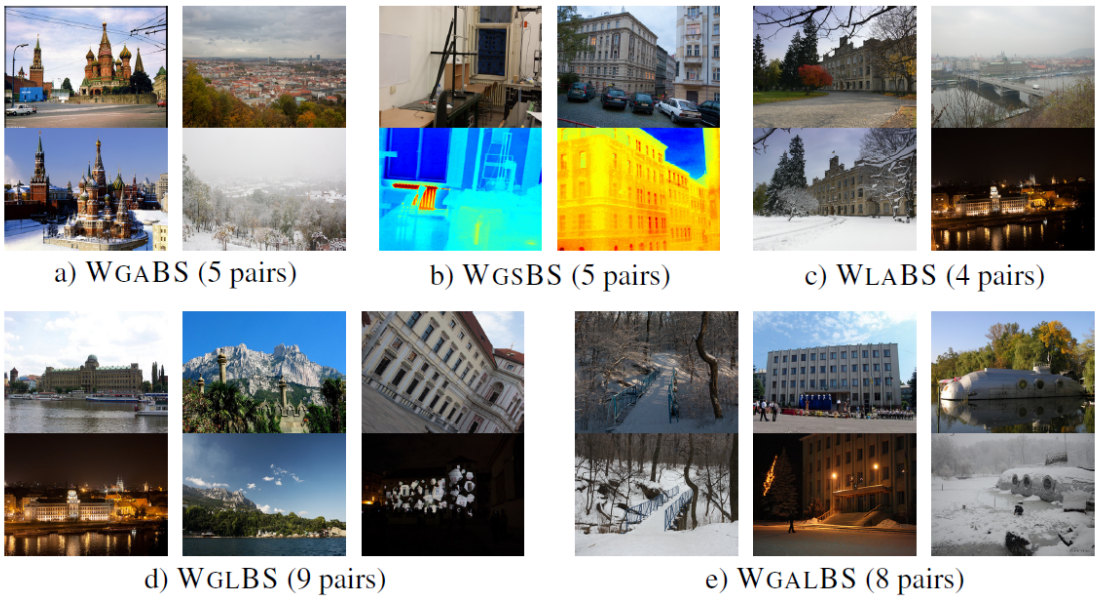


Fig. 11 Example images of the *wide baselines* WXBS dataset proposed in [Mishkin et al., 2015], where WGABS stands for viewpoint and appearance changes, WGSBS for viewpoint and modality changes, WLABS for lighting and appearance changes, WGLBS for viewpoint and lighting changes, and WGALBS for viewpoint, appearance and lighting changes. (Courtesy to Dmytro Mishkin)

Hessian-Affine showed the best performance. Comparing individual descriptors, gradient-histogram based SIFT and their variants including DAISY [Tola et al., 2010] showed the best performance. Furthermore, it was observed that most of the descriptors gain significantly from photometric normalization. One of the main problems in matching of day to night and infrared images is the low number of detected features. A possible approach addressing the problem is iiDoG [Vonikakis et al., 2013] where the difference of Gaussians is normalized by sum of Gaussians, but it cannot be easily applied for other types of detectors. Note that only handcrafted keypoint detector and descriptor combinations were considered in the paper.

[Sun et al., 2017] propose a dataset for benchmarking the image-based localization pipeline and compare the per-query registration rate obtained with BRIEF, SURF, SIFT, COV [Perd'och et al., 2009], and RSIFT local features. Best performance was obtained with RSIFT. The paper does not consider deep learning based features.

In contrary to the papers above, more recent benchmarks consider comparison of deep learning based models with handcrafted and learning based models. Deep feature detectors were compared with handcrafted detectors in [Zhang et al., 2017a]. Handcrafted keypoint descriptors (mainly SIFT) were compared with learning and deep learning based methods considering patch verification, matching and retrieval in [Balntas et al., 2017]. [Schönberger et al., 2017] also evaluate the performance of image-based reconstruc-

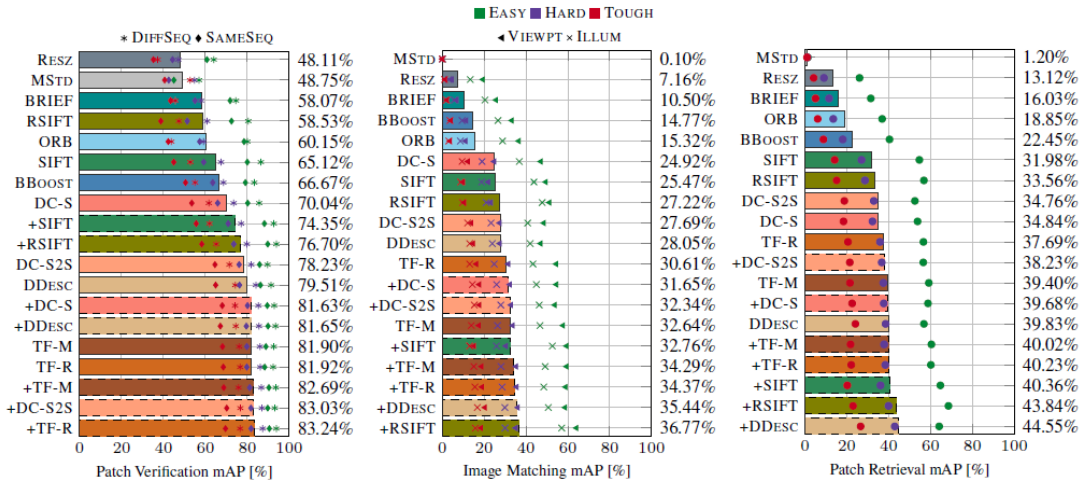


Fig. 12 Summary of the patch verification, matching, and retrieval results in the HPatches benchmark experiments. See [Balntas et al., 2017] for the details. (Courtesy to Vassileios Balntas)

tion using these features.

[Zhang et al., 2017a] compare several handcrafted feature detectors with FAST [Rosten and Drummond, 2006], TILDE [Verdie et al., 2015], CovDet [Lenc and Vedaldi, 2016] and TCovDet [Zhang et al., 2017a] on several datasets[40]. They show that the repeatability of the detected keypoints obtained with TCovDet is significantly better than the one obtained with CovDet and TILDE. The latter ones performs better than the handcrafted methods (SIFT, SURF, MSER, or Harris/Hessian Laplace/Affine detector) and FAST. Concerning the matching score, TCovDet performs best on two out of three datasets and SIFT on the third one, which contains drastic background clutter changes that seem to be less well handled by TCovDet.

[Balntas et al., 2017] first show that previous datasets (see Table 1) and evaluation protocols do not unambiguously specify all aspects of evaluation, leading to ambiguities and inconsistencies in results reported in the literature, and thus, propose HPatches, a new benchmark to overcome these weaknesses. This benchmark includes a new large dataset suitable for training and testing of modern descriptors. It follows strictly defined evaluation protocols in several tasks such as matching, retrieval and classification. They did an exhaustive evaluation comparing handcrafted features SIFT, RSIFT[41] [Arandjelović and Zisserman, 2012], BRIEF, ORB, BBoost [Trzcinski et al., 2015], and the recent deep descriptors DeepCompare (DC) [Zagoruyko and Komodakis, 2015], DDesc [Simo-Serra et al., 2015b], and TFeat (TF) [Balntas et al., 2016b]. They added two baselines, MSTD which is the mean and the standard deviation of the patch and RESZ which is a vector obtained by resizing the patch to 6x6 and normalizing it to have zero mean and the variance equal to 1. The results from [Balntas et al., 2017] are shown in Figure 12. The learning based descriptors were

[40] The datasets can be downloaded from `https://www.dropbox.com/s/l7a8zvni6ia5f9g/datasets.tar.gz`

[41] Using a square root (Hellinger) kernel instead of the standard Euclidean distance to measure the similarity between SIFT descriptors.

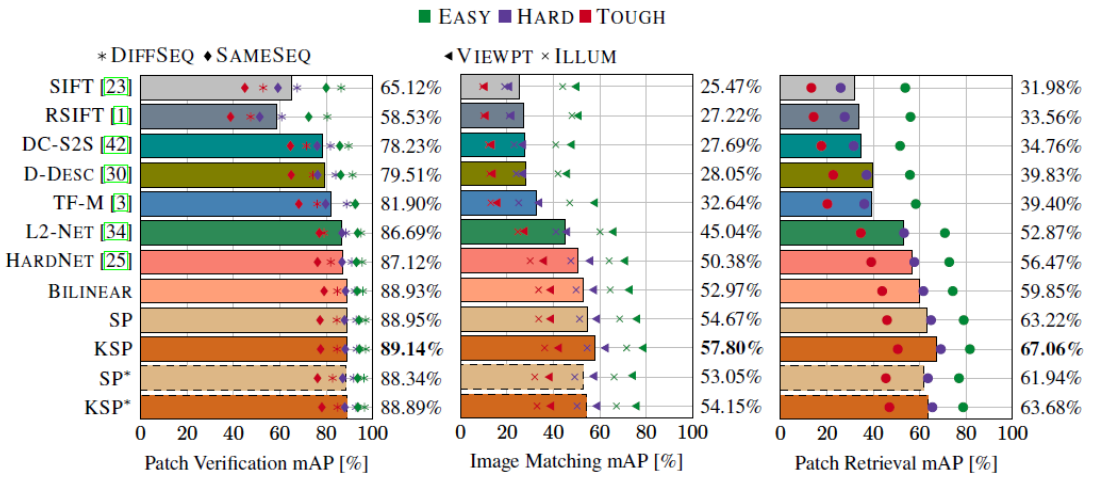


Fig. 13 Patch verification, matching, and retrieval results on the HPatches dataset from [Wei et al., 2018]. (Courtesy to Xing Wei)

trained on PhotoTourism [Winder, 2007] dataset which is completely different from HPatches. Evaluation was done on three benchmark tasks: patch verification, image matching, and patch retrieval. In most cases (see also Figure 12, where "+" indicates ZCA (Zero Components Analysis) whitening and normalization), it was observed that post-processing the descriptors by applying ZCA whitening [Bishop, 1995] followed by power law normalization [Arandjelović and Zisserman, 2012] and L2 normalization allowed to significantly improve the results on various tasks already observed in previous papers [Ke and Sukthankar, 2004, Arandjelović and Zisserman, 2012]. As expected, this gain is more important in the case of handcrafted than learned features.

Concerning patch verification, best performance was obtained with deep learning based descriptors as they were jointly optimized with their distance metric to perform well in the verification task. This is even clearer when only the tough geometric distortions are considered.

In the case of image matching, where descriptors are used to match patches from a reference image to a target image, ZCA whitened and normalized RSIFT surprisingly outperformed the descriptors obtained with deep methods. Similarly, its performance is close to the best in the patch retrieval scenario.

The binary descriptors are competitive only for the patch verification task, hence their main advantages remained compactness and speed for applications where this is a strong requirement. The learning based binary feature BBoost in general outperformed the handcrafted binary features especially on the patch verification task. Among the deep features, best matching and retrieval performance is obtained with DDesc, followed by TF, but DDesc performed worse on patch verification. Its main drawback is that it has the highest computational cost.

In order to complement the above results with a few newer methods, in Figure 13 we show results from [Wei et al., 2018] and in Figure 14 results from [He et al., 2018b]. These results were obtained on the same dataset (HPatches) following the same evaluation protocol. In Figure 13, we can see that SP is similar to L2-Net [Tian et al., 2017] and HardNet [Mishchuk et al., 2017] except for the last convolution layer, which is replaced by the subspace pooling (SP) layer. KSP means that the marginal triplet loss was combined with

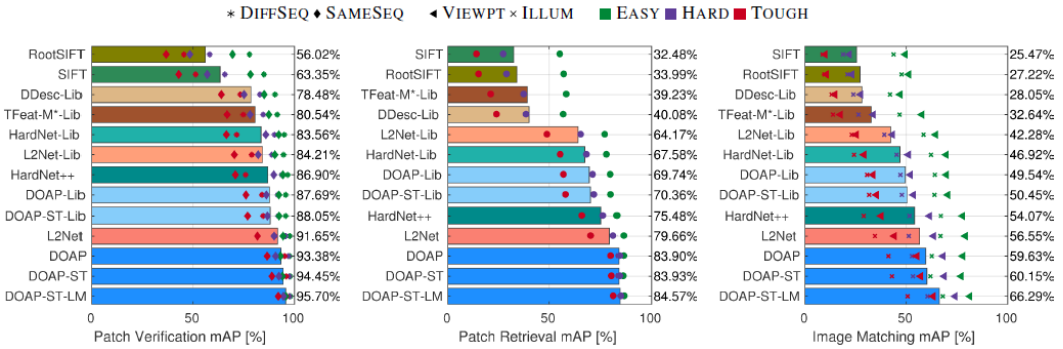


Fig. 14 Patch verification, matching, and retrieval results on the HPatches dataset from [He et al., 2018b], where -Lib means that the model was trained on the Liberty set of the Photo-Tourism dataset [Winder, 2007] instead of HPatches, -ST means that the model includes a Spatial Transformer Network to estimate the geometric transformation and -LM means that clustering-based label mining was used during training. (Courtesy to Kun He)

the projection Gaussian kernel (see Section 4.1 and [Wei et al., 2018] for more details). Bilinear is the model proposed in [Lin et al., 2015] applied for learning patch matching. The difference between SP, KSP and SP*, KSP* respectively is that in the latter case a lower dimensional subspace was considered (see details in [Wei et al., 2018]).

First, we can observe that more recent deep learning based approaches such as L2-Net [Tian et al., 2017] and HardNet [Mishchuk et al., 2017] significantly outperform the previous results obtained in the HPatches benchmark experiments (see also Figure 12). Considering on top of L2-Net, bilinear pooling or subspace pooling allows for further improvement of all tasks.

Figure 14 compares mainly DOAP variants [He et al., 2018b] with SIFT and other deep models showing that optimizing a loss which estimates the average precision, has a positive effect on patch retrieval, image matching and patch verification all measured by mAP.

The comparative evaluation proposed in [Schönberger et al., 2017], goes beyond descriptor matching, and also evaluates how well the descriptors perform in the context of image-based reconstruction using challenging small- and large-scale datasets. As part of the image-based reconstruction pipeline, structure from motion (SfM) uses descriptor matching in the first stage to produce a graph of corresponding features in multiple views. All subsequent stages strongly depend on the quantity and the quality of these correspondences. In order to give new practical insights into the performance of the evaluated descriptors, [Schönberger et al., 2017] propose to evaluate and compare the descriptors at different stages of the SfM pipeline: feature matching, geometric verification, image retrieval, and sparse and dense modeling.

They consider RSIFT as a baseline descriptor, with two advanced variants of it, RSIFT-PCA [Bursuc et al., 2015] and DSP-SIFT [Dong and Soatto, 2015], and compare them with ConvOpt [Simonyan et al., 2014], DDesc, TFeat and LIFT [Yi et al., 2016a]. Except for LIFT, which is an end-to-end detector and descriptor network, the standard Difference of Gaussians (DoG) based keypoint detector was used for all descriptors and the learning based approaches were trained on DoG keypoints.

In Table 2 we show the key properties of the evaluated descriptors from [Schönberger et al., 2017]. We can see that the extraction of the handcrafted descriptors is much faster as compared to the learned features

Table 2 Key properties of the evaluated descriptors in [Schönberger et al., 2017], with per image average timings on the Oxford5k dataset, where extraction includes the detection time as well. (Courtesy to J. L. Schönberger)

	RSIFT	RSIFT-PCA	DSP-SIFT	ConvOpt	DDesc	TFeat	LIFT
Dimensionality	128	80	128	73	128	128	128
Size [bytes]	128	320	512	292	512	512	512
Platform	CPU	CPU	CPU	GPU	GPU	GPU	GPU
Extraction [s]	9.3	10.5	23.7	49.9	24.3	11.8	212.3
Matching [s]	0.14	0.11	0.14	0.10	0.14	0.14	0.14

despite running on the CPU. As such, the learned features are not yet a practical alternative for processing millions of images in a reasonable time-frame, especially on desktop computers. Among the learning based features, LIFT is the slowest by a large margin, while TFeat is comparably fast. The matching depends highly on the feature size, therefore the matching speed is higher for ConvOpt than in the case of Tfeat, DDesc and LIFT.

Concerning the different evaluation results of image matching and measured at different stages of the SfM pipeline, here we only briefly summarize the findings of [Schönberger et al., 2017], for more detailed analyses please refer to the original paper. Similarly to [Heinly et al., 2012], the paper shows that for image matching, blur, day-night, and large viewpoint change seem to be most challenging for all descriptors. The learned descriptors typically outperformed RSIFT in terms of recall, while RSIFT performed better in terms of precision. Both RSIFT-PCA and DSP-SIFT outperformed the learned features for almost all metrics and matching scenarios tested. Among the learned descriptors, ConvOpt was found to produce overall the best results and had the lowest variance across the different datasets.

To evaluate the completeness and accuracy of the reconstruction results, the key metrics used in [Schönberger et al., 2017] were the number of registered images, the number of 3D points in the sparse SfM map obtained with COLMAP [Schönberger and Frahm, 2016], the number of verified image projections of sparse points and their track lengths, the overall reprojection error, the pose accuracy of the camera locations, as well as the number of reconstructed dense points after Multi-View Stereo (MVS) reconstruction obtained with CMVS [Furukawa and Ponce, 2010].

The experiments on several datasets, yielded the following observations: On small or easy datasets, the learned descriptors generally performed on par with or better than RSIFT in terms of the number of sparse points, the number of image observations, and the mean track length, and worse than RSIFT-PCA and DSP-SIFT on these metrics. However, considering the number of registered images, and the final dense modeling performance and accuracy metrics, all methods produce roughly the same reconstruction quality.

On larger and more challenging datasets, more variation was found when ranking the features using different metrics and datasets. In spite of the superiority of the learned descriptors over RSIFT observed in raw matching evaluation, in the reconstruction evaluation RSIFT sometimes performed better and sometimes worse than the learned descriptors. DSP-SIFT performed the best among all the methods, both in terms of sparse and dense reconstruction results. It consistently produced the most complete sparse reconstruction in terms of the number of registered images and reconstructed sparse points, and the dense models had the most points as a result of accurate camera registration. It had a slightly higher reprojection error potentially caused by the descriptor pooling across multiple scales which leads to more robustness but less accurate keypoint localization. LIFT produced the largest reprojection error and relatively short tracks for all datasets, indicating inferior keypoint localization performance as compared to the handcrafted DoG method.

Concerning camera pose estimation, the ground truth was only available for 3 datasets, two small ones Fountain&Herzjesu [Strecha et al., 2008] and the Quad6K from the Cornell BigSfM dataset [Crandall et al., 2013]. On the small datasets all methods performed similar. On Quad6K, RSIFT performed best, followed by TFeat and DCP-SIFT, RSIFT-PCA performed worst. Summarizing, even if learning approaches advanced, handcrafted features still perform on par or better than recent learned features in the practical context of image-based reconstruction.

6 Conclusion

In this paper, our aim was to give a comprehensive overview of most popular methods for local keypoint detection and region description. We started with classic handcrafted approaches and finished with more recent learning and deep learning based methods. We intentionally did not provide any experimental comparisons because, in spite of the fact that most algorithms are available online, it remains hard and requires a lot of work to properly evaluate and fairly compare all these methods. The literature already shows controversial findings, so we decided to carefully analyze existing benchmark papers, which compare most of the discussed methods, instead of providing another benchmark. Together with a sufficient understanding of the methods, which we tried to provide in this paper as well, we are convinced that the presented discussion is very useful for selection of the best feature type for a given task and data.

As a conclusion, we want to highlight the following findings[42]:
- Learned methods are a good choice when image content matters, especially for applications like image matching.
- Post-processing the descriptors with whitening, power, and L2-normalization improves matching and hence the results on various tasks.
- Among the non-deep learning based models, ConvOpt is one of the best, showing good performance across different datasets and tasks, but was outperformed by several recent deep models. TFeat, L2Net and HardNet are good deep based models that can be improved by (kernel) subspace (SP, KSP) or bilinear pooling as well as with adding global loss (TGLoss) or global orthogonal regularization (GOR).
- By directly optimizing the average precision instead of pair or triplet loss, DOAP obtains the best performance on patch verification, matching, and retrieval.
- TConvDet was shown to provide the best keypoint repeatability compared to other detectors and combined with SIFT descriptor provides good matching performance.
- LIFT and SuperPoint are among the few networks that learn keypoint detection, geometric transformation and keypoint description in a single network trained end-to-end.
- High robustness against viewpoint changes can be obtained with SIFT and its variants. All together SIFT remains to be a very good option for various applications.
- In addition to the advantages of low memory footprint and matching time, deep leaned binary features, *e.g.* binary DOAP provide competitive results on recent benchmarks.
- Extraction of handcrafted descriptors is much faster compared to learned features. Thus, learned features are not yet a practical alternative for processing millions of images in reasonable time on desktop computers.

[42] Note that for some of the findings, especially regarding the more recent methods not included in the above mentioned benchmarks, we had to rely on the results provided in the paper describing the methods.

However deep features remain promising when they come with efficient hardware implementation.
- Even if learning approaches advanced, achieving best matching performance, the handcrafted features still perform similar or better than recent deep learned features in the context of the whole image-based 3D reconstruction and localization pipeline.

However, since the applications of computer vision, especially when concerning real world problem, are very diverse and the data they have to deal with is to a large extent unpredictable, we are convinced that trying to learn how to detect and describe relevant regions is preferable against trying to manually define all possible configurations. Even the decision whether or not a region is relevant for a certain application can be handed over to the algorithm. In this way, future computer vision algorithms can better cope with different camera parameters, noise characteristics, changing environments, and different user behavior.

References

Alaa E. Abdel-Hakim and Aly A. Farag. CSIFT: A SIFT descriptor with color invariant characteristics. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Pablo F. Alcantarilla and Adrien Nuevo, Jesús Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In *BMVA British Machine Vision Conference (BMVC)*, 2013.

Pablo F. Alcantarilla, Adrien Bartoli, and Andrew J. Davison. Kaze features. In *European Conference on Computer Vision (ECCV)*, 2012.

Relja Arandjelović and Andrew Zisserman. Three things everyone should know to improve object retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Boris Babenko, Piotr Dollár, and Serge Belongie. Task specific local region matching. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

Vassileios Balntas, Lilian Tang, and Krystian Mikolajczyk. Bold - binary online learned descriptor for efficient image matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Vassileios Balntas, Edward Johns, Lilian Tang, and Krystian Mikolajczyk. Pn-net: Conjoined triple deep network for learning local image descriptors. *CoRR*, arXiv:1601.05030, 2016a.

Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *BMVA British Machine Vision Conference (BMVC)*, 2016b.

Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, 2006.

Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2000.

JiaWang Bian, Wen-Yan Lin, Yasuyuki Matsushita, Sai-Kit Yeung, Tan-Dat Nguyen, and Ming-Ming Cheng. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Irving Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 2(29):115–147, 1987.

Christopher M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.

Matthew Brown, Gang Hua, and Simon Winder. Discriminative learning of local image descriptors. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(1):43–57, 2010.

Andrei Bursuc, Giorgos Tolias, and Hervé Jégou. Kernel local descriptors with implicit rotation matching. In *ACM International Conference on Multimedia Retrieval (ICMR)*, 2015.

Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European Conference on Computer Vision (ECCV)*, 2010.

Hsin-Yi Chen, Yen-Yu Lin, and Bing-Yu Chen. Robust feature matching with alternate hough and inverted hough transforms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

Hong Cheng, Zicheng Liu, Nanning Zheng, and Jie Yang. A deformable local image descriptor. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

Yukyung Choi, Chaehoon Park, Joon-Young Lee, and In So Kweon. Robust binary feature using the intensity order. In *Asian Conference on Computer Vision (ACCV)*, 2014.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

Christopher B. Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2016.

Ondřej Chum and Jiří Matas. Matching with prosacprogressive sample consensus. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

Ondřej Chum and Jiří Matas. Optimal randomized ransac. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(8):1472–1482, 2008.

Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)*, 2003.

Jason J. Corso and Gregory D. Hager. Coherent regions for concise and stable image description. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

David Crandall, Andrew Owens, Noah Snavely, and Dan Huttenlocher. SfM with MRFs: Discrete-continuous optimization for large-scale structure from motion. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(12):2841–2853, 2013.

Gabriela Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical learning in computer vision (SLCV)*, 2004.

Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Workshop on Deep Learning for Visual SLAM*, 2018.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning (ICML)*, 2014.

Jingming Dong and Stefano Soatto. Domain-size pooling in local descriptors: Dsp-sift. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Yueqi Duan, Jiwei Lu, Ziwei Wang, Jianjiang Feng, and Jie Zhou. Learning deep binary descriptor with multi-quantization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a.

Yueqi Duan, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Context-aware local binary feature learning for face recognition. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 40(5):1139–1153, 2017b.

Yueqi Duan, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Learning rotation-invariant local binary descriptor. *Transactions on Image Processing*, 26(8):3636–3651, 2017c.

Bin Fan, Fuchao Wu, and Zhanyi Hu. Aggregating gradient distributions into intensity orders: A novel local image descriptor. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

Bin Fan, Qingqun Kong, Tomasz Trzcinski, Zhiheng Wang, Chunhong Pan, and Pascal Fua. Receptive fields selection for binary feature description. *Transactions on Image Processing*, 23(6):2583–2595, 2014a.

Bin Fan, Zhenhua Wang, and Fuchao Wu. *Local Image Descriptor: Modern Approaches*, volume SpringerBriefs in Computer Science. Springer, 2014b.

Paolo Di Febbo, Carlo Dal Mutto, Kinh Tieu, and Stefano Mattoccia. Kcnn: Extremely-efficient hardware keypoint detection with a compact convolutional neural network. In *CVPR Workshop on Embedded Vision (EVW)*, 2018.

Philipp Fischer, Alexey Dosovitskiy, and Thomas Brox. Descriptor matching with convolutional neural networks: a comparison to sift. *CoRR*, arXiv:405.5769, 2014.

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.

Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(8):1362–1376, 2010.

Yongqiang Gao, Weilin Huang, and Yu Qiao. Local multi-grouped binary descriptor with ring-based pooling configuration and optimization. *Transactions on Image Processing*, 24(12):4820–4833, 2015.

Vincent Garcia, Éric Debreuve, Frank Nielsen, and Michel Barlaud. K-nearest neighbor search: Fast gpu-based implementations and application to high-dimensional feature matching. In *International Conference on Image Processing (ICIP)*, 2010.

Steffen Gauglitz, Matthew Turk, and Tobias Höllerer. Improving keypoint orientation assignment. In *BMVA British Machine Vision Conference (BMVC)*, 2011.

Yunchao Gong and Svetlana Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, 1988.

Wilfried Hartmann, Michal Havlena, and Konrad Schindler. Predicting matchability. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Tal Hassner, Shay Filosof, Viki Mayzels, and Lihi Zelnik-Manor. Sifting through scales. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 39(7):1431–1442, 2017.

Kun He, Fatih Cakir, Sarah Adel Bargal, and Stan Sclaroff. Hashing as tie-aware learning to rank. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018a.

Kun He, Yan Lu, and Scan Sclaroff. Local descriptors optimized for average precision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018b.

Jared Heinly, Enrique Dunn, and Jan-Michael Frahm. Comparative evaluation of binary features. In *European Conference on Computer Vision (ECCV)*, 2012.

Stefan Hinterstoisser, Selim Benhimane, Nassir Navab, Pascal Fua, and Vincent Lepetit. Online learning of patch perspective rectification for efficient object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.

Michael Jahrer, Michael Grabner, and Horst Bischof. Learned local descriptors for recognition and matching. In *Computer Vision Winter Workshop*, 2008.

Hervé Jégou, Matthijs Douze, Cordelia Schmidt, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *CoRR*, arXiv:1408.5093, 2014.

Timor Kadir, Andrew Zisserman, and Michael Brady. An affine invariant salient region detector. In *European Conference on Computer Vision (ECCV)*, 2004.

Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

Simon Korman, Mark Milam, and Stefano Soatto. Oatm: Occlusion aware template matching by consensus set maximization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Scott Krig. *Computer Vision Metrics: Survey, Taxonomy, and Analysis*. Springer, 2014.

Vijay Kumar B. G., Gustavo Carneiro, and Ian Reid. Learning local image descriptors with deep siamese and triplet convolutional networks by minimizing global loss functions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Hanjiang Lai, Yan Pan, Ye Liu, and Shuicheng Yan. Simultaneous feature learning and hash coding with deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Zhen Lei, Matti Pietikaïnen, and Stan Z. Li. Learning discriminant face descriptor. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(2):289–302, 2014.

Karel Lenc and Andrea Vedaldi. Learning covariant feature detectors. *CoRR*, arXiv:1605.01224, 2016.

Vincent Lepetit and Pascal Fua. Keypoint recognition using randomized trees. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(9):1465–1479, 2006.

Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

Wu-Jun Li, Sheng Wang, and Wang-Cheng Kang. Feature learning based deep supervised hashing with pairwise labels. In *AAAI International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

Yunpeng Li, Noah Snavely, and Dan Huttenlocher. Location recognition using prioritized feature matching. In *European Conference on Computer Vision (ECCV)*, 2014.

Kevin Lin, Jiwen Lu, Chu-Song Chen, and Jie Zhou. Learning compact binary descriptors with unsupervised deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

Venice Erin Liong, Jiwen Lu, Gang Wang, v Pierre, and Jie Zhou. Deep hashing for compact binary codes learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Song-Tyang Liu and Wen-Hsiang Tsai. Moment preserving corner detection. *Pattern Recognition*, 23(5): 441–460, 1990.

Jonathan L. Long, Ning Zhang, and Trevor Darrell. Do convnets learn correspondence? In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2014.

David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

Jiwen Lu, Xiuzhuang Liong, Venice Erin Zhou, and Jie Zhou. Learning compact binary face descriptor for face recognition. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 37(10):2041–2056, 2015.

Jiwen Lu, Venice Erin Liong, and Jie Zhou. Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Early Access, 2017.

Jiayi Ma, Ji Zhao, Jinwen Tian, Alan L. Yuille, and Zhuowen Tu. Robust point matching via vector field consensus. *Transactions on Image Processing*, 23(4):1706–1721, 2014.

Josef Maier, Martin Humenberger, Markus Murschitz, Oliver Zendel1, and Markus Vincze. Guided matching based on statistical optical flow for fast and robust correspondence analysis. In *European Conference on Computer Vision (ECCV)*, 2016.

Josef Maier, Martin Humenberger, Oliver Zendel1, and Markus Vincze. Ground truth accuracy and performance of the matching pipeline. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Jiří Matas, Ondřej Chum, Martin Urban, and Tomas Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. In *BMVA British Machine Vision Conference (BMVC)*, 2002.

Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(10):993–1008, 2003.

Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiří Matas, Timor Schaffalitzky, Frederik Kadir, and Luc Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1/2):43–72, 2005.

Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenović, and Jiří Matas. Working hard to know your neighbors margins: Local descriptor learning loss. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2017.

Dmytro Mishkin, Michal Perdoch, and Jiří Matas. MODS fast and robust method for two-view matching. *CoRR*, arXiv:1503.02619, 2014.

Dmytro Mishkin, Jiří Matas, Michal Perdoch, and Karel Lenc. Wxbs: Wide baseline stereo generalizations. In *BMVA British Machine Vision Conference (BMVC)*, 2015.

Marius Muja and David G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(11):2227–2240, 2014.

Kai Ni, Hailin Jin, and Frank Dellaert. Efficient consensus in the presence of groupings. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

Hyeonwoo Noh, André Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. 2017.

Mattis Paulin, Matthijs Douze, Zaid Harchaoui, Julien Mairal, Florent Perronin, and Cordelia Schmid. Local convolutional features with unsupervised training for image retrieval. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

Michal Perd'och, Ondřej Chum, and Jiří Matas. Efficient representation of local geometry for large scale object retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Florent Perronnin and Chris Dance. Fisher Kernels on visual vocabularies for image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

James Philbin, Ondřej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

James Philbin, Michael Isard, Josef Sivic, and Andrew Zisserman. Descriptor learning for efficient retrieval. In *European Conference on Computer Vision (ECCV)*, 2010.

Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, and Timo Ahonen. *Computer Vision Using Binary Patterns*, volume CVIS Series, Vol:40. Springer, 2011.

Andrew Richardson and Edwin Olson. Learning convolutional filters for interest point detection. In *International Conference on Robotics and Automation (ICRA)*, 2013.

Azriel Rosenfeld and Avinash Kak. *Digital Picture Processing*. Academic Press, 1982.

Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision (ECCV)*, 2006.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

Jordi Sanchez-Riera, Jonas Östlund, and Francesc Fua, Pascal Moreno-Noguer. Simultaneous pose, correspondence and non-rigid shape. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Johannes L. Schönberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Nicu Sebe, Qi Tian, Etienne Loupias, Michael S. Lew, and Thomas S. Huang. Evaluation of salient point techniques. *Image and Vision Computing*, 21(13-14):1087–1095, 2003.

Vanshika Shah, Rajvi Srivastava and P.J. Narayanan. Geometry-aware feature matching for structure from motion applications. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2015.

Edgar Simo-Serra, Carme Torras, and Francesc Moreno-Noguer. Dali: Deformation and light invariant descriptor. *International Journal of Computer Vision*, 115(2):136–154, 2015a.

Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *IEEE International Conference on Computer Vision (ICCV)*, 2015b.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Learning local feature descriptors using convex optimisation. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(8):1573–1585, 2014.

Josef Sivic, , and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision (ICCV)*, 2003.

Stephen M. Smith and J. Michael Brady. SUSAN a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 2002.

Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Annual Conference on Neural Information Processing Systems (NIPS)*, 2016.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Christoph Strecha, Wolfgang von Hansen, Luc Van Gool, Pascal Fua, and Ulrich Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

Christoph Strecha, Albrecht Lindner, Karim Ali, and Pascal Fua. Training for task specific keypoint detection. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)*, 2009.

Christoph Strecha, Bronstein Alex, Michael Bronstein, and Pascal Fua. Ldahash: Improved matching with smaller descriptors. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(1):66–78, 2012.

Xun Sun, Yuanfan Xie, Pei Luo, and Liang Wang. A dataset for benchmarking image-based localization. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(5):815–830, 2010.

Marwan Torki and Ahmed Elgammal. One-shot multi-set non-rigid feature-spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

Philip H. S. Torr and Andrew Zisserman. A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000.

Lorenzo Torresani, Vladimir Kolmogorov, and Carsten Rother. Feature correspondence via graph matching: Models and global optimization. In *European Conference on Computer Vision (ECCV)*, 2008.

Eduard Trulls, Iasonas Kokkinos, Alberto Sanfeliu, and Francesc Moreno-Noguer. Dense segmentation-aware descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

Tomasz Trzcinski and Vincent Lepetit. Efficient discriminative projections for compact binary descriptors. In *European Conference on Computer Vision (ECCV)*, 2012.

Tomasz Trzcinski, Mario Christoudias, and Vincent Lepetit. Learning image descriptors with boosting. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 37(3):597–610, 2015.

Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2007.

Yannick Verdie, Kwang Moo Yi, Pascal Fua, and Vincent LePetit. Tilde: A temporally invariant learned detector. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Vasillios Vonikakis, Dimitrios Chrysostomou, Rigas Kouskouridas, and Antonios Gasteratos. A biologically inspired scale-space for illumination invariant feature detection. 24(7), 2013.

Han Wang and Michael Brady. Real-time corner detection algorithm for motion estimation. *Image and Vision Computing*, 13(9):695–703, 1995.

Zhenhua Wang, Bin Fan, and Fuchao Wu. Affine subspace representation for feature description. In *European Conference on Computer Vision (ECCV)*, 2014.

Zhenhua Wang, Bin Fan, Gang Wang, and Fuchao Wu. Exploring local and overall ordinal information for robust feature description. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 38(11): 2198–2211, 2016.

Xing Wei, Yue Zhang, Yihong Gong, and Nanning Zheng. Kernelized subspace pooling for deep local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009.

Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *European Conference on Computer Vision (ECCV)*, 2014.

Matthew Winder, Simon or Brown. Learning local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

Rongkai Xia, Yan Pan, Hanjiang Lai, Cong Liu, and Shuicheng Yan. Supervised hashing for image retrieval via image representation learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2014.

Liu Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:25432596, 2010.

Gehua Yang, Charles V. Stewart, Michal Sofka, and Chia-Ling Tsai. Registration of challenging image pairs: Initialization, estimation, and decision. *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(11):1973–1989, 2007.

Tsun-Yi Yang, Jo-Han Hsu, Yen-Yu Lin, and Yung-Yu Chuang. Deepcd: Learning deep complementary descriptors for patch representations. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

Kwang Moo Yi, Eduard Trulls, Vincent V. Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision (ECCV)*, 2016a.

Kwang Moo Yi, Yannick Verdie, Pascal Fua, and Vincent LePetit. Learning to assign orientations to feature points. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016b.

Ramin Zabih and John Woodfill. Nonparametric local transforms for computing visual correspondence. In *European Conference on Computer Vision (ECCV)*, 1994.

Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Jure Zbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 2016.

Ruimao Zhang, Liang Lin, Rui Zhang, Wangmeng Zuo, and Lei Zhang. Bit-scalable deep hashing with regularized similarity learning for image retrieval and person re-identification. *Transactions on Image Processing*, 24(12):4766–4779, 2015.

Xu Zhang, Felix X. Yu, Svebor Karaman, and Shih-Fu Chang. Learning discriminative and transformation covariant local feature detectors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a.

Xu Zhang, Felix X. Yu, Sanjiv Kumar, and Shih-Fu Chang. Learning spread-out local feature descriptors. In *IEEE International Conference on Computer Vision (ICCV)*, 2017b.

Fang Zhao, Huang Yongzhen, Liang Wang, and Tieniu Tan. Deep semantic ranking based hashing for multi-label image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

C. Lawrence Zitnick and Krishnan Ramnath. Edge foci interest points. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

Oscar A. Zuniga and Robert M. Haralick. Corner detection using the facet model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1983.