A Performance Evaluation of Local Features for Image-Based 3D Reconstruction

Bin Fan, *Senior Member, IEEE,* Qingqun Kong, Xinchao Wang, Zhiheng Wang, Shiming Xiang, Chunhong Pan, Pascal Fua, *Fellow, IEEE,*

Abstract—This paper performs a comprehensive and comparative evaluation of the state of the art local features for the task of image based 3D reconstruction. The evaluated local features cover the recently developed ones by using powerful machine learning techniques and the elaborately designed handcrafted features. To obtain a comprehensive evaluation, we choose to include both float type features and binary ones. Meanwhile, two kinds of datasets have been used in this evaluation. One is a dataset of many different scene types with groundtruth 3D points, containing images of different scenes captured at fixed positions, for quantitative performance evaluation of different local features in the controlled image capturing situation. The other dataset contains Internet scale image sets of several landmarks with a lot of unrelated images, which is used for qualitative performance evaluation of different local features in the free image collection situation. Our experimental results show that binary features are competent to reconstruct scenes from controlled image sequences with only a fraction of processing time compared to use float type features. However, for the case of large scale image set with many distracting images, float type features show a clear advantage over binary ones. Currently, the most traditional SIFT is very stable with regard to scene types in this specific task and produces very competitive reconstruction results among all the evaluated local features. Meanwhile, although the learned binary features are not as competitive as the handcrafted ones, learning float type features with CNN is promising but still requires much effort in the future.

1 Introduction

3D vision has been a persistent research topic of computer vision, including both non-rigid and rigid 3D reconstruction. Typical non-rigid 3D reconstruction method such as morphable modeling has been widely used in 3D face modeling to improve face recognition performance [1], [2], [3]. On the other hand, as the popular solution to rigid 3D reconstruction, image-based 3D reconstruction [4], [5] relies on fundamentals of multi-view geometry to recover 3D point clouds from a number of images. The core to generate 3D points from 2D images is the triangulation of corresponding points across multiple images. How to establish reliable point correspondences is known as the problem of local feature matching, which typically contains three steps: extracting keypoints from images, constructing local descriptors for keypoints, and establishing keypoint correspondences across different images according to distances of their descriptors.

- *This work was supported in part by the National Natural Science Foundation of China (61573352, 61876180), in part by the Young Elite Scientists Sponsorship Program by CAST (2018QNRC001), in part by the Henan Science and Technology Innovation Outstanding Youth Program (184100510009), and in part by the Henan University Scientific and Technological Innovation Team Support Program (19IRTSTHN012).*
- *Bin Fan, Shiming Xiang, and Chunhong Pan are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China. Qingqun Kong is with the Institute of Automation, Chinese Academy of Sciences and the University of Chinese Academy of Sciences, China. {bfan,smxiang,chpan}@nlpr.ia.ac.cn; qingqun.kong@ia.ac.cn*
- *Xinchao Wang is with the Department of Computer Science, Stevens Institute of Technology, USA. xinchao.w@gmail.com*
- *Zhiheng Wang is with the School of Computer Science and Technique, Henan Polytechnic University, China. wzhenry@eyou.com*
- *Pascal Fua is with the CVLab, EPFL, Switzerland. pascal.fua@epfl.ch*
- *The first two authors (Bin Fan and Qingqun Kong) contribute equally. Zhiheng Wang is the corresponding author.*

Both keypoint and local descriptor could be called local feature in literature and the milestone work is SIFT [6], containing DoG keypoint and SIFT descriptor. Since then, various local features have been developed to improve SIFT's performance on establishing more point correspondences with higher precision. Some improvements were made on either keypoint extraction (e.g., FAST [7], TILDE [8]) or local image description [9] (including handcrafted descriptors like MROGH [10] and LIOP [11], as well as learning based ones such as BinBoost [12], L2Net [13], and so on [14], [15], [16]). While on the other side, there are plenty of works focusing on the whole pipeline of feature extraction and description in order to replace the whole SIFT. Typical methods of this kind include SURF [17], ORB [18], BRISK [19], FRIF [20], LIFT [21], among which ORB, BRISK and FRIF are binary features that were claimed to be faster and more compact.

Due to the large number of keypoint detectors and descriptors, many works have been conducted to evaluate the performance of their combinations on image matching, ranging from image-level benchmarks to patch-level benchmarks [22], [23], [24], [25], [26]. While almost all the recent evaluations suggest some better feature matching combinations out of the SIFT baseline, however, an interesting phenomenon is that SIFT is still the major choice for the task of image based 3D reconstruction [4], [5], [27], [28], [29], indicating that the suggested better features may not necessarily lead to better reconstruction results in practice. In other words, the performance of different feature matching methods evaluated on image matching task could not be general enough to the task of image based 3D reconstruction. Basically, given feature matches across multiple images as input to an image based 3D reconstruction system, it involves large scale optimizations

to filter out incorrect matches, obtain camera poses, and recover 3D points of the inlier matches. Besides the widely used evaluation metrics of feature matches (i.e., number of matches, matching score, precision, recall), there are more critical issues affecting the optimization procedure, such as the number of matches between similar non-overlap images, the spatial distribution of correct matches, the number of matches and matching quality between key images. Unfortunately, the current specifically designed image matching benchmarks are mainly focused on matching images under various imaging conditions, from which all these issues are hard to be considered and tested. Meanwhile, how to find key images among many images collected from one scene is still an open problem. Therefore, it is less likely to design a new image matching benchmark considering all these issues together so that the evaluated results about superiorities of local features can be directly extended to their usefulness in the image based 3D reconstruction task. Alternatively, this paper proposes an end-to-end performance comparison of different keypoints and descriptors in this specific task. Our evaluation covers recent advances on both handcrafted and learning based features, aiming to provide a practical guidance to researchers working on image based 3D reconstruction about how to choose local features.

To finish such an end-to-end comparative study, a basic but typical 3D reconstruction system is implemented[1]. The system is based on the linear time incremental structure from motion [30] (VisualSFM) and CMVS [31], [32] by taking the matching keypoints across different images as input. We evaluate different combinations of keypoint and local descriptor to generate different inputs to the system so as to obtain different reconstruction results. To make this evaluation comprehensive and up to date, we choose to evaluate on recently developed methods, containing both hand-crafted and learning based features with two different types: traditional float type ones and the emerging binary ones. Moreover, SIFT is taken as the baseline since it is still the preliminary choice for feature matching in the community of image based 3D reconstruction. More specifically, for float type descriptors, we include SIFT [6] and LIOP [11] as representative handcrafted ones and cover the learning based ones that use the traditional learning technique (VGGDesc [15]) and the recently popular CNNs (DeepDesc [33], L2Net [13], LIFT [21]) respectively. All these evaluated methods, except for SIFT and LIFT which have their own keypoint detectors, are merely feature description methods and so they have to be used with a keypoint detector. In this paper, we use SIFT keypoint for its popularity and also because that it is already used along with SIFT in the baseline. That is to say, except for LIFT, all the evaluated float type descriptors are based on the SIFT keypoint, while LIFT is based on its own keypoint. For binary descriptors, BRISK [19], FRIF [20], LDB [34], RFD [14] and BinBoost [12] are taken in our evaluation. The former two are handcrafted features while the latter three are learned ones. Among them, BRISK and FRIF contain both keypoint detector and binary descriptor. As a result, we use both of these two kinds of keypoints and combine them with all the evaluated binary descriptors respectively. As far as the considered datasets, two different types of datasets are used. The first one is a recently proposed multiview stereo dataset (DTU MVS) [35], which contains more than 100 different scenes with high resolution images captured from 49 or 64 fixed viewpoints. Meanwhile, groundtruth 3D points are available. This dataset has a large diversity in scene types with a moderate number of images for each scene, while at the meantime still providing the groundtruth 3D points to facilitate an objective evaluation of reconstruction results. The second dataset contains eight large scale structure from motion (SFM) subsets [27], each of which contains thousands of unordered images and many distracted images for one worldwide landmark. These two datasets stand for two typical image collection situations for 3D reconstruction applications. One is the controlled case where images are captured at selected viewpoints, and so is widely used for applications about reconstructing a very specific scene or object. In this case, all images cover a part of the scene and have enough overlaps due to the specially designed viewpoints. The other case does not have any constraint on the used images, and so is widely used for applications about reconstructing a very large scale place such as a landmark or a city. In this case, it resorts to collecting images from the Internet, instead of spending huge labors to capture high quality images with specially considered imaging viewpoints as in the first case. Consequently, it inevitably contains many unrelated and low quality images as well as non-overlapping images, thus is more challenging. To sum up, through this well-designed evaluation, we want to find out that for different scene types encountered in various applications of 3D reconstruction, which kind of local feature could be the best choice.

The remaining parts of this paper are organized as follows. Section 2 reviews the existing local feature evaluations. In Section 3, we briefly describe our implemented 3D reconstruction system. Then, the evaluated local features are introduced in Section 4 and Section 5. The evaluation results and analysis on the two used datasets are presented in Section 6 and Section 7 respectively. Finally, Section 8 concludes this paper.

2 Related Work

Accompany with the flourish of local features, many works have been conducted to evaluate performance of various local features. Mikolajczyk et al. [22], [36] evaluated the matching performance of different local descriptors and affine invariant interest regions on images of planar scenes. Moreels and Perona [37] extended Mikolajczyk's evaluations to images of 3D objects captured on a turntable. These evaluations demonstrated the higher distinctiveness of SIFT than its previous methods, thus promoting the development of SIFT-like local features, i.e., histogram-based handcrafted features such as SURF [17], DAISY [38], KAZE [39]. Aanæs et al. [25] revised Mikolajczyk's and Moreels's works by introducing a more comprehensive dataset with known spatial correspondence of points, while at the meantime to cover various situations for interest point matching. Although most detectors in their evaluation has been evaluated before, their evaluation was more thorough and convincing because the newly introduced dataset is more

1. www.nlpr.ia.ac.cn/fanbin/feature_evaluation_3d.htm

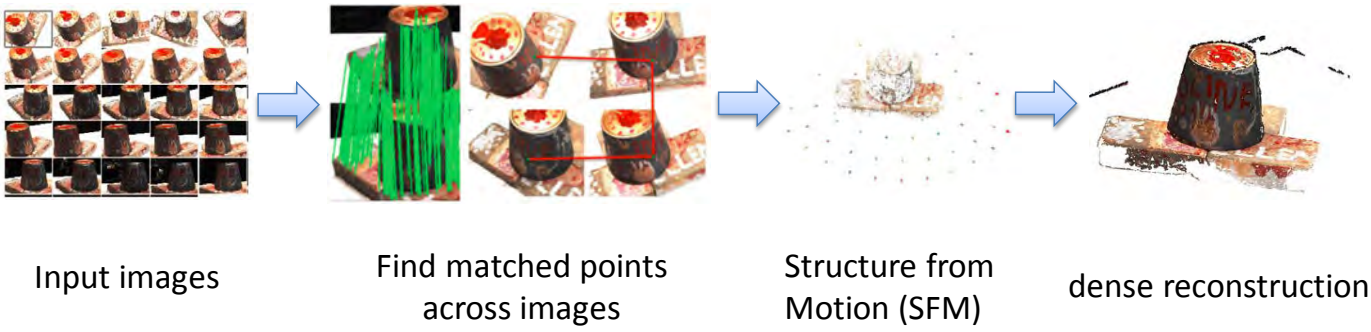


Fig. 1. Pipeline of a typical image based 3D reconstruction system. This paper is focused on the second step, i.e., finding point correspondences across multiple images, using different keypoints and local descriptors to study their practical performance on the final reconstruction results.

realistically challenging. Their evaluation re-emphasized the importance of detecting feature points in scale space and showed that the affine adaption proposed by Mikolajczyk and Schmid [40] has a little influence on feature detector itself, but is useful for the descriptor, thus is helpful in the whole pipeline of feature matching. Recently, with the development of binary descriptors, some researchers evaluated different local features under the same evaluation protocol of image matching as [22] but with an emphasize on the compactness and speed of the tested methods. For this purpose, Miksik and Mikolajczyk [24] showed that binary features such as ORB [18] and BRIEF [41] are efficient in both feature extraction and matching for image matching due to the fast computation of Hamming distance. On the other hand, the state of the art handcrafted descriptors such as LIOP [11], [42] and MROGH [10] could result in better matching performance but with much higher computational burden. Similarly, Heinly et al. [23] gave a comparative evaluation of binary features by considering more keypoints. Balntas et al. [26] introduced a patch level benchmark for evaluation of mainly deep learning based local descriptors on image matching, patch classification and patch retrieval tasks.

Beyond image matching, using local features to establish patch-level correspondences between images is a low-level building block for many computer vision applications. Therefore, some researchers were also interested in finding out which local feature performs the best on these applications, and so have conducted many local feature evaluations on various applications. Gauglitz et al. [43] evaluated different interest points and local descriptors for visual tracking. Bauml and Stiefelhagen [44] evaluated different local features for person re-identification in image sequences. Madeo and Bober [45] conducted a comparative study on using binary descriptors for mobile applications. Liu et al. [46], [47] conducted evaluations of local binary features for texture classification. Similar to this paper, Fan et al. [48] and Schonberger et al. [49] studied performance of different local features for image based 3D reconstruction systems. However, Fan et al. [48] only evaluated three binary features (ORB, BRISK and FRIF) while Schonberger et al. [49] were mainly focused on the learned float type descriptors. On the contrary, this paper extensively evaluates different combinations of existing binary descriptors and feature detectors. Besides traditional handcrafted ones, these binary descriptors also include learning based ones, e.g., BinBoost [12], LDB [34] and RFD [14], which have been shown with superior performance on standard image matching benchmarks. Moreover, a comparative study of the state of the art float type descriptors is conducted in this work too. Therefore, the evaluation of this work is more comprehensive compared to the previous works, covering the state of the arts in both binary and float type local features, and ranging from handcrafted features to the learning based ones. Many of these features are not evaluated before. What is more, about the evaluation datasets, we use both the DTU MVS dataset [35] and the large scale SFM dataset [27]. Therefore, our evaluation covers two typical cases for 3D reconstruction, i.e., 1) controlled image capturing with moderate number of images which is the case of DTU MVS dataset, and 2) free image capturing with a large number of images and many distracted images that is the case of SFM dataset. For the former case, we rely on the supplied groundtruth to study and compare performance (in terms of *accuracy* and *completeness* of the reconstruction) of different feature combinations. While for the latter, the ability of reconstructing scene as complete as possible is what we pursue, which is evaluated by the numbers of *recovered images*, *sparse points* and *dense points*.

3 Pipeline of Image Based 3D Reconstruction

To obtain the 3D points of an object or a scene by only using a number of images, the popular solutions [4], [29], [50], as shown by Fig. 1, usually include three steps: feature matching across images, structure from motion (SFM) [5], [30] and dense reconstruction [31], [51]. Feature matching aims to find the so called feature tracks by computing descriptors' distances of the detected keypoints in images. Essentially, a feature track corresponds to a 3D point, containing point correspondences across different images. For very large scale and unordered image collection, there is usually an additional preprocessing step, aiming to quickly find out possible overlapping image pairs so as to conduct feature matching only on these pairs to save matching time at the cost of image matching quality [28], [52]. Structure from motion takes a number of feature tracks as input, and

outputs a number of 3D points as well as camera parameters of some input images (i.e., the recovered images). With the recovered cameras, dense reconstruction is applied to obtain a dense 3D point cloud as the reconstruction result. In a word, a typical 3D reconstruction system outputs include a number of 3D points of the scene and the estimated camera parameters of the input images. By comparing these outputs to the groundtruth, one can evaluate how good the system is, e.g., in terms of 3D reconstruction *accuracy* and *completeness*, as well as the numbers of *recovered cameras*, *sparse points* and *dense points*.

In this paper, we focus on the step of feature matching, studying its practical influence on reconstruction quality when using different local features. As a result, we fix the last two steps with typical methods: the linear time incremental structure from motion integrated in VisualSFM [30] for SFM and the CMVS [31], [32] for dense reconstruction. Although there are many SFM algorithms that have been proposed in literature, VisualSFM is the most widely used one in the community of 3D vision for its good performance and convenience to use. CMVS extends the PMVS [31] to deal with large scale image sets by decomposing the input images into smaller manageable image clusters, while maintaining the excellent dense reconstruction performance of PMVS. The source codes of these two methods are provided too and can be downloaded from their websites. Meanwhile, no preprocessing is used, i.e., feature matching is extensively conducted for all possible image pairs, so as to reduce the risk of inferior performance induced by the preprocessing step and ensure a fair evaluation of local features only involved in feature matching. As the focus of this paper is on local features and due to space limit, for readers interested on VisualSFM and CMVS, please refer to the original papers [30], [31], [32] for more details. In the following, we first give a brief introduction to the evaluated features and then move to the evaluation results.

4 Float Type Features

Local feature has been an active and persistent topic in computer vision community. To keep this evaluation thorough and up to data, we choose recently proposed methods, including both handcrafted descriptors and the recent popular learning based ones. For reference, we also include the classical SIFT in our evaluation as baseline.

4.1 SIFT

SIFT constructs a Difference of Gaussian (DoG) scale space to detect extrema across both spatial and scale spaces as keypoints. DoG scale space is constructed by subtracting neighboring images of a Gaussian scale space of the input image. The keypoint orientation is computed by accumulating a histogram of gradient orientations from a local circular region around the keypoint to achieve rotation invariance. The orientation corresponding to the largest bin in this histogram is taken as the keypoint orientation. Meanwhile, other orientations corresponding to the peak bins which are within 80% of the largest one are also taken as the keypoint's orientations.

For feature description, SIFT divides the scale and rotation normalized local patch around a keypoint into 4×4 grids. In each grid, it computes a histogram of gradient orientations with 8 bins. All these histograms are concatenated together and normalized to get a 128 dimensional float vector as the SIFT descriptor. To improve robustness, the trilinear interpolation among spatial and orientation bins is utilized and a Gaussian weight is assigned to each pixel in the local patch.

4.2 LIOP

In SIFT and and its variants [17], [22], [39], they rely on dominant orientations to achieve rotation invariance. Fan et al. [10] observed that the dominant orientations estimated from local image context are unreliable, and thus they proposed to construct local image descriptors by intensity order pooling to achieve intrinsic rotation invariance. Under this framework, Wang et al. [11], [42] proposed the LIOP descriptor by pooling a kind of low level feature based on the local ordinal information around a pixel in the support region. The local intensity order can explore the relative relationship of intensities among all neighboring points around a pixel, not merely the relationship between two points which is often used by LBP invariants [47]. As a result, LIOP was reported with higher performance than its previous methods. For this reason, we choose to include LIOP in our evaluation as a representative handcrafted local feature.

4.3 VGGDesc

While traditional methods for local image description are handcrafted, learning good local descriptors has been extensively explored in recent years. One representative work is proposed by the Visual Geometry Group (VGG) in the Oxford University. Following Brown et al.'s work on discriminative learning of local image descriptors [53], Simonyan et al. [15] proposed to formulate the descriptor learning problem in a convex optimization framework based on the hinge loss with sparsity constraint. They used the RDA [54] to efficiently solve the involved sparse constrained optimization problem with large scale training set. They first learned a high dimensional descriptor by selecting discriminative pooling areas through sparse constraint. Then, they pursued a linear subspace of the learned high dimensional descriptor to obtain the final compact descriptor with powerful discriminative ability.

4.4 DeepDesc

With the popularity of using Convolutional Neural Networks (CNNs) in various vision tasks, it has also been used in descriptor learning. Although initial works on using CNNs to learn patch descriptors are usually combined with additional metric layers to achieve good matching performance [55], [56], [57], researchers gradually move to the more practical case, i.e., learning a patch descriptor that can be directly operated in the Euclidean space. This is because that this kind of descriptor can be used as a drop-in replacement for the widely used handcrafted descriptors, thus has wider applications. One representative work of learning patch descriptors without additional metric layers is the DeepDesc proposed by Edgar et al. [33]. They used

a Siamese Network structure and minimized a hinge-like loss when training the network. With a carefully designed network structure and a hard sample mining strategy for network training, they finally obtained a 128 dimensional float type descriptor that can be measured in the Euclidean space.

4.5 L2Net

A very recent work on learning discriminative patch descriptor in the Euclidean space by CNNs is the L2Net [13], which is specially designed for the matching task and incorporates supervision information of intermediate layers to improve its generalization ability. It takes a fully convolutional architecture with 7 convolutional layers, each of which is followed by a batch normalization layer with fixed parameters. Like DeepDesc, it finally outputs a 128 dimensional vector as the descriptor to serve as a drop-in replacement of SIFT for various applications. L2Net was the rank one method for the competition of local features held in ECCV'16 and obtained the top performance on the widely used patch matching dataset (i.e., the Brown dataset [53]). Due to its superior performance, we choose to include it in our evaluation.

4.6 LIFT

We also include LIFT [21] in this evaluation as the state of the art method for the whole pipeline of feature detection and description. Inspired by the success of deep learning and identical to the SIFT's pipeline, LIFT combines all necessary components (i.e., keypoint detector, orientation estimator, and local patch descriptor) of a local feature altogether in an end-to-end manner based on the deep convolutional architecture. Specifically, it uses TILDE [8] as the keypoint detector because TILDE is convolutional, differentiable and with good performance. After detecting keypoints, it estimates the orientations of those patches around the detected keypoints by a CNN which is trained to minimize the generated descriptors' distance of matching patches [58]. Finally, the DeepDesc is used to extract feature descriptors for the scale and rotation normalized patches. To crop, resize, and rotate the local patch around a keypoint, LIFT uses the spatial transform network [59] as connector since it is differentiable. As a result, the whole pipeline of LIFT is differentiable and so can be trained in an end-to-end manner. In practice, the authors trained LIFT one component by one component started from the descriptor part and then finetuned the whole pipeline.

4.7 Implementation Details

For SIFT, we use the implementation supplied in VLFeat [60]. For the other float type descriptors, we use the implementations provided by their authors[2]. SIFT keypoints (i.e., DoG) are used for all these descriptors except for LIFT, which has its own keypoints. The low dimensional descriptor learned on the 'Liberty' of the Pacth Dataset [53] is used for the VGG descriptor. Similarly, the evaluated L2Net is also trained on the 'Liberty'. While for the DeepDesc, we use the authors' suggested model that was trained on a subset of 'Liberty', 'Notre Dame' and 'Yosemite' of the Patch Dataset, which has better generalization ability than the one trained on the 'Liberty'. For LIFT, due to the additional supervised information of keypoints are required, the patch level dataset can not be used for training LIFT. Alternately, it was trained with a SFM dataset (Piccadilly Circus dataset [27]), and we use the public available model supplied by the authors that has been reported with superior image matching performance in their paper [21]. Please see Table 1 for a summary of all these local features. Identical to the Lowe's ratio test [6], the Nearest Neighbor Distance Ratio (NNDR) is used for matching keypoints, where the ratio threshold is set as 0.8 for all the tested descriptors. To find the nearest and the second nearest neighbors, we use the open source ANN library [61] for the fast approximate nearest neighbor search.

2. LIFT: https://github.com/cvlab-epfl/LIFT
LIOP: https://github.com/foelin/IntensityOrderFeature
L2Net: https://github.com/yuruntian/L2-Net
DeepDesc: https://github.com/etrulls/deepdesc-release
VGGDesc: http://www.robots.ox.ac.uk/ vgg/software/learn_desc/

5 Binary Features

To reduce the memory footprint of float type descriptors, binary descriptors have been widely studied in recent years. These binary descriptors have been successfully used in some light weight tasks, such as template based object detection [41] and SLAM [62], which usually involve matching only several hundreds of keypoints. However, they have not yet been used or evaluated for tasks involving extensively keypoint matching, such as the one we studied in this paper. In this work, we choose typical binary features to evaluate their performance on 3D reconstruction. For comprehensiveness, we cover both handcrafted ones and the learning based ones as summarized on Table 1.

5.1 BRISK

BRISK contains a scale and rotation invariant keypoint detector and a binary feature descriptor. For the keypoint detector, BRISK implements a scale space by using two pyramids alternately, one for the octaves and the other for the intra-octaves, to trade-off the computation and scale estimation accuracy. The keypoints are detected in each level of the scale space based on the AGAST [63], which is an effective extension of the FAST corner detector [7]. Based on the position and scale of the detected keypoint, a sampling pattern with 60 points regularly sampled from 4 concentric circles are used to compute the keypoint's orientation as well as its binary descriptor. Specifically, the point pairs generated by these sampling points are divided into long-distance pairs and short-distance ones. The long-distance pairs are used to compute an average local gradient to define the orientation of the keypoint, while the short-distance pairs are used for intensity tests to construct the binary descriptor. To deal with aliasing effects, the intensity of a sampling point is computed by filtering with a Gaussian kernel whose standard deviation is proportional to its distance to the keypoint, i.e., the central point of the sampling pattern.

TABLE 1
Summary of the evaluated local features.

keypoint	descriptor	dimension	data type	handcrafted	learned	training set
FRIF or BRISK	FRIF [20]	512	binary	√	×	×
	BRISK [19]	512	binary	√	×	×
	LDB [34]	256	binary	×	√	Liberty [53]
	RFD [14]	288	binary	×	√	Liberty [53]
	BinBoost [12]	256	binary	×	√	Liberty [53]
DoG (SIFT)	SIFT [6]	128	float	√	×	×
	LIOP [11]	144	float	√	×	×
	VGGDesc [15]	128	float	×	√	Liberty [53]
	DeepDesc [33]	77	float	×	√	subset of {Liberty,NotreDame,Yosemite} [53]
	L2Net [13]	128	float	×	√	Liberty [53]
LIFT	LIFT [21]	128	float	×	√	Piccadilly [27]

5.2 FRIF

While BRISK resorts to FAST detector for efficient keypoint detection, FRIF relies on the response of Laplacian of Gaussian (LoG). The basic idea is to approximate LoG with rectangular filters so that to compute its response very quickly by integral images. According to Mikolajczyk and Schmid's study [64], Laplacian of Gaussian is stable in characteristic scale selection and has been used in many feature detectors [6], [40]. In FRIF, it approximates a LoG template by linear combination of four rectangular filters. Therefore, computing the LoG responses on pixels of an image just requires linear combination of four rectangular filtering results, which can be done efficiently based on integral images. To detect extrema of the approximated LoG responses across both spatial and scale spaces, FRIF implements an identical scale space as BRISK does and uses a similar strategy for non-maximum suppression as well as location refinement.

As far as the binary descriptor is concerned, FRIF uses a similar sampling pattern to BRISK, but proposes a mixed binary descriptor to achieve better performance. For each sampling point, it uses its neighboring points to conduct intensity tests to obtain a number of bits as part of the descriptor. It also uses some short-distance point pairs for intensity tests as the remaining part of the descriptor to capture complementary information. The long-distance point pairs are used to compute the keypoint orientation as in BRISK.

5.3 LDB

LDB [34] is a binary descriptor computed based on intensity difference and gradient difference. It first participates the local region into several cells according to the predefined spatial configurations. Then the averaged intensities and gradients are computed for each of these cells. These average values between cell pairs are compared to generate binary values so as to construct the binary descriptor. To select only a few discriminative and meaningful test pairs from all the possible cell pairs, a modified adaboost algorithm is proposed by Yang and Cheng [34].

5.4 RFD

Gradient orientation map used in SIFT and DAISY [38] has shown its effectiveness in constructing discriminative local descriptors. Fan et al. [14] extended it for binary feature description. They proposed to construct a bit of a binary descriptor by thresholding the oriented gradient responses accumulated from a certain region, which is either a rectangular or a Gaussian shaped region. The best threshold value for each region is determined by the Bayesian criteria according to the labeled training data. Such regions constructing the so call RFD descriptor are greedy selected from a large pool of candidates according to their discriminative ability and correlation.

5.5 BinBoost

Similar to RFD which uses the thresholded gradient orientation map as the basic element, Trzcinski et al. [12] applied boosting to learn high compact binary descriptor. The learned descriptor, named as BinBoost, takes a linear combination of several thresholded gradient orientation maps and then thresholds the combination result as one bit in the descriptor. In other words, if we consider each gradient orientation map as one weak classifier, each bit in BinBoost corresponds to a strong classifier according to the boosting theory. The gradient orientation maps and their linear weights are selected based on a modified adaboost learning algorithm proposed in their paper too.

Among the above five binary descriptors, the first two have both feature detector and feature descriptor. The latter three are only binary descriptors which have to be evaluated along with a specific feature detector. Therefore, in our evaluation, we combine them with feature detectors provided by the first two methods respectively. Here, we do not evaluate ORB [18] for two reasons. First, both BRISK keypoint and ORB keypoint are based on the AGAST while BRISK uses a finer scale space, so the BRISK keypoint is better. Second, ORB has been shown with inferior performance to BRISK and FRIF in our previous work [48].

5.6 Implementation Details

All the evaluated binary features have source codes available on the Internet[3], therefore, we use the original implementations with default parameters released by their

3. RFD: http://www.nlpr.ia.ac.cn/fanbin/rfd.htm
FRIF: https://github.com/foelin/FRIF
BRISK: http://www.asl.ethz.ch/people/lestefan/personal/BRISK
BinBoost: http://cvlab.epfl.ch/research/detect/binboost
LDB: http://lbmedia.ece.ucsb.edu/research/binaryDescriptor/web_home/web_home

authors. For RFD, the one trained on the 'Liberty' of the Patch Dataset with rectangle receptive field is used (denoted as RFDR). For BinBoost, the one with 256 bits is used, which is also trained on the 'Liberty' and reported with the best generalization ability.

To match keypoints of these binary features, we use the multi-table and multi-probe LSH implemented in the FLANN library [65] to approximately find the first two nearest neighbors in an efficient manner. Then the distance ratio of the first and the second nearest neighbors is used to decide whether two keypoints are matched or not. The same as the case of float descriptor, the ratio threshold is set as 0.8. Note that although computing the Hamming distance of two binary descriptors is significantly faster than computing the Euclidean distance of two float type descriptors, it is still impractical to conduct bruteforce nearest neighbor search in Hamming space because of the large number of image matching operations involved in 3D reconstruction task. Due to this reason, the fast approximate nearest neighbor search method, i.e. multi-table, multi-probe LSH, is used. Specifically, we set the number of hash tables as 4, the multi-probe level as 1, the LSH code length as 24 in all our evaluations.

6 Evaluation on Multiview Stereo Dataset

6.1 Dataset

We first choose to evaluate the 3D reconstruction performance of different features on a recently published multi-view stereo dataset, known as the DTU MVS dataset [35]. It contains a total number of 124 different scenes, covering a wide range of objects and surface materials. For each scene, it collects images of 1600×1200 resolution from 49 or 64 different viewpoints, with 8 different illumination conditions. Among these scenes, 80 scenes contain necessary information that is required for the evaluation of reconstruction results as Jensen et al. did [35]. In this paper, we use the scenes with 49 views, which occupy 58 out of all 80 scenes. We do not study effects of different lighting conditions, so we just use the subset with all lights on.

Due to the fact that our implemented 3D reconstruction system is fully automatic and uses the self-calibration to decide the camera parameters, the coordinate system of the reconstructed 3D points can be any of those recovered cameras. In this case, the reconstructed coordinate system and the supplied reference coordinate system are related by a 3D similarity transformation (scaling, rotation and translation). Therefore, we have to firstly register the reconstructed 3D points to the reference scans (groundtruth) obtained by a structure light scanner which are supplied in the dataset. To this end, we manually select corresponding 3D points between the reconstruction and the groundtruth so as to estimate the similarity transformation for registering the reconstructed 3D points.

6.2 Evaluation Protocol

After registering the reconstructed 3D points to the reference coordinate system, we use the supplied code in the dataset for performance evaluation. The evaluation protocol designed for the DTU MVS dataset is based on that of [66], with some modifications to make it unbiased and better at handling missing data and outliers. Basically, it adopts an observability mask so that the evaluation is only focused on the visible part of the scene. Please refer to [35] for more details.

As in [35], [66], *accuracy* and *completeness* are used as the quality measures of a reconstruction. *Accuracy* measures how close a 3D point in the reconstruction is to the groundtruth 3D scan. Given a 3D point of the reconstruction, its *accuracy* is defined as the nearest distance from this point to all 3D points in the reference scan. Therefore, the lower value of *accuracy* is, the better the reconstruction fits the groundtruth. Similarly, *completeness* measures how close a 3D point in the groundtruth scan is to the reconstruction. For any 3D point in the groundtruth, its *completeness* is defined as its nearest distance to all 3D points in the reconstruction. A higher value of *completeness* usually means that there is no corresponding point in the reconstruction for the considered groundtruth point, indicating a missing point. To give comparable statistics for different reconstructions, the mean accuracy (mean completeness) are computed for all points in the reconstruction (groundtruth scan) and are recorded as indicators to evaluate the quality of different reconstructions. Other statistics such as median was also used in our experiments and we did not find any difference in conclusion, so due to space limit, we only report results of mean accuracy and mean completeness in this paper. For neatness, we use accuracy and completeness instead of mean accuracy and mean completeness in the remaining of this paper.

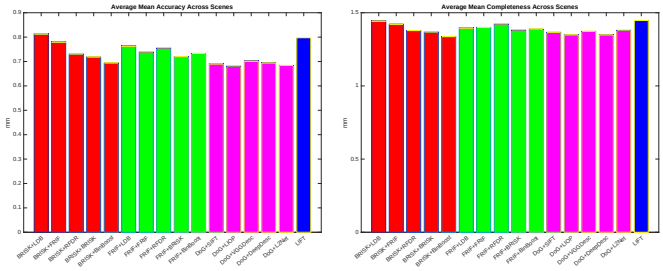
There are two situations that are commonly encountered in 3D reconstruction which could induce bias if they are not treated properly. One is that there are usually more 3D points in the textured regions, while the other one is outliers. We use the same strategy as in [35] to deal with these problems. The first issue is addressed by subsampling, i.e., the reconstructed 3D points are subsampled so that any two points have a distance larger than 0.2mm. For the second issue, those points with large errors which could be outliers are simply removed. Specifically, the points whose distances are larger than 20mm are removed when computing accuracy and completeness.

All experiments are conducted in a laptop with Intel 2.5GHz CPU and 8GB memory. The evaluation code and the dataset can be downloaded on: http://roboimagedata.compute.dtu.dk

6.3 Results and Analysis

Among the 58 tested scenes, for the convenience of analysis, we further divide them into 3 groups according to whether all the evaluated methods perform similar or not. We first compute the variance of accuracy and the variance of completeness for each scene among all the evaluated methods. For a specific scene, a low variance in both accuracy and completeness means that all the evaluated methods perform similarly. Consequently, we set a threshold (0.05mm) to select out this kind of scenes as the first group, which actually contains 49 scenes. For the remaining 9 scenes, they have a large variance in either accuracy or completeness, i.e., at least one method performs significantly different from

Fig. 2. Some example scenes that have small performance difference for the evaluated methods.



(a) accuracy (b) completeness

Fig. 3. The average reconstruction (a) accuracy and (b) completeness over all the scenes that have small performance variance for the evaluated methods. See text for details.

other ones. Among these 9 scenes, we further find that there are 3 scenes for which at least one method fails to obtain the reconstruction result. As a result, we divide the 9 scenes into two groups, one containing 6 scenes that all the evaluated methods successfully obtain the reconstruction results but with large performance variance, while the other containing 3 scenes for which some methods fail. To sum up, the first group contains 49 scenes with small performance variance, the second group contains 6 scenes with large performance variance, and the third group contains 3 scenes that at least one method fails. They should correspond to different challenging levels of 3D reconstruction. We will analysis the performance of the evaluated methods for these three groups of scenes respectively.

Scenes with small performance variance. In this case, it should refer to the easiest scenes for 3D reconstruction since all the evaluated local features lead to similar reconstruction accuracy and completeness. Some examples of these scenes are shown in Fig. 2. As can be seen, these scenes all contain rich textures and are thus easy for feature point matching. For these scenes, it is not necessary to analysis the results of different scenes one by one as there is only minor difference among them. The average mean accuracy of different methods across all scenes of this kind (i.e., with small performance variance) is shown in Fig. 3(a), while the average mean completeness is shown in Fig. 3(b). Among the binary features, the combination of BRISK keypoint with BinBoost descriptor performs the best, whose performance is comparable or even better than some float features. For all the tested combinations, BRISK keypoint with BinBoost descriptor and DoG keypoint with LIOP descriptor perform similar, both of which are with the top performance. In general, DoG with float descriptors lead to a better reconstruction accuracy than using binary features, except for the best combination of BRISK + BinBoost. An interesting observation is that the entire feature learning solution, LIFT, does not perform as well as other float features. In fact, it performs the worst among all the evaluated features, including the binary ones. Obviously, using LIFT leads to larger reconstruction errors both in terms of accuracy and completeness. Such an inferior performance of LIFT indicates that there might be larger localization error between corresponding LIFT keypoints since it indeed produces comparable or more matching points than SIFT in our experiments. Except for LIFT, LIOP produces slightly better results than other float type descriptors and the remaining ones perform similarly. Among all the binary descriptors, LDB is not as good as others no matter which keypoint is used. Meanwhile, when using FRIF keypoint, the results of different binary descriptors are more flat than using BRISK keypoint. This means that FRIF keypoint is more robust and less sensitive to descriptors. For BRISK keypoint, it has to be careful when choosing the combined descriptor in order to achieve good performance. From Fig. 3, we can conclude that it is not necessary to learn sophisticated descriptors for reconstructing scenes with rich textures under the controlled settings for image capture. In this case, using binary features is good enough to obtain satisfactory reconstruction accuracy as using float features while have the computational advantage on feature extraction and matching.

Scenes with large performance variance. In this case, it refers to the challenging scene types for image based 3D reconstruction. The results are shown in Fig. 4. In these figures, the 1st column displays the scenes, the 2nd column shows the mean accuracy of different methods, the 3rd column shows the mean completeness of different methods, and the 4th column gives the running times of different methods. Compared to the scenes with small performance variance, images of these scenes do not have rich textures to be locally distinguished, i.e., most part of these scenes has uniform appearance. Some local features are not capable of dealing with such kind of images to obtain enough high quality point correspondences across images, therefore, leading to inferior 3D reconstruction results. From Fig. 4, we have the following observations:

(1) Consistent to the observation in easy scenes, using FRIF keypoint is relatively less sensitive to the used descriptors than using the BRISK keypoint. In many scenes, it produces similar results for different binary descriptors when using FRIF keypoint. This property of FRIF is similar to DoG. To further show this point, for each kind of keypoint, we record the number of scenes that have large performance variance for different descriptors. These numbers for BRISK, FRIF and DoG are 7, 3 and 2 respectively.

(2) Different from the easy scenes, BRISK with BinBoost does not perform the best for these challenging scenes, and it is hard to say which combination is better because it tends to be scene related. In addition, LIFT no longer performs the worst like in the case of easy scenes. However, although LIFT performs better than most combinations, it is the most time consuming. In general, using float features is a better

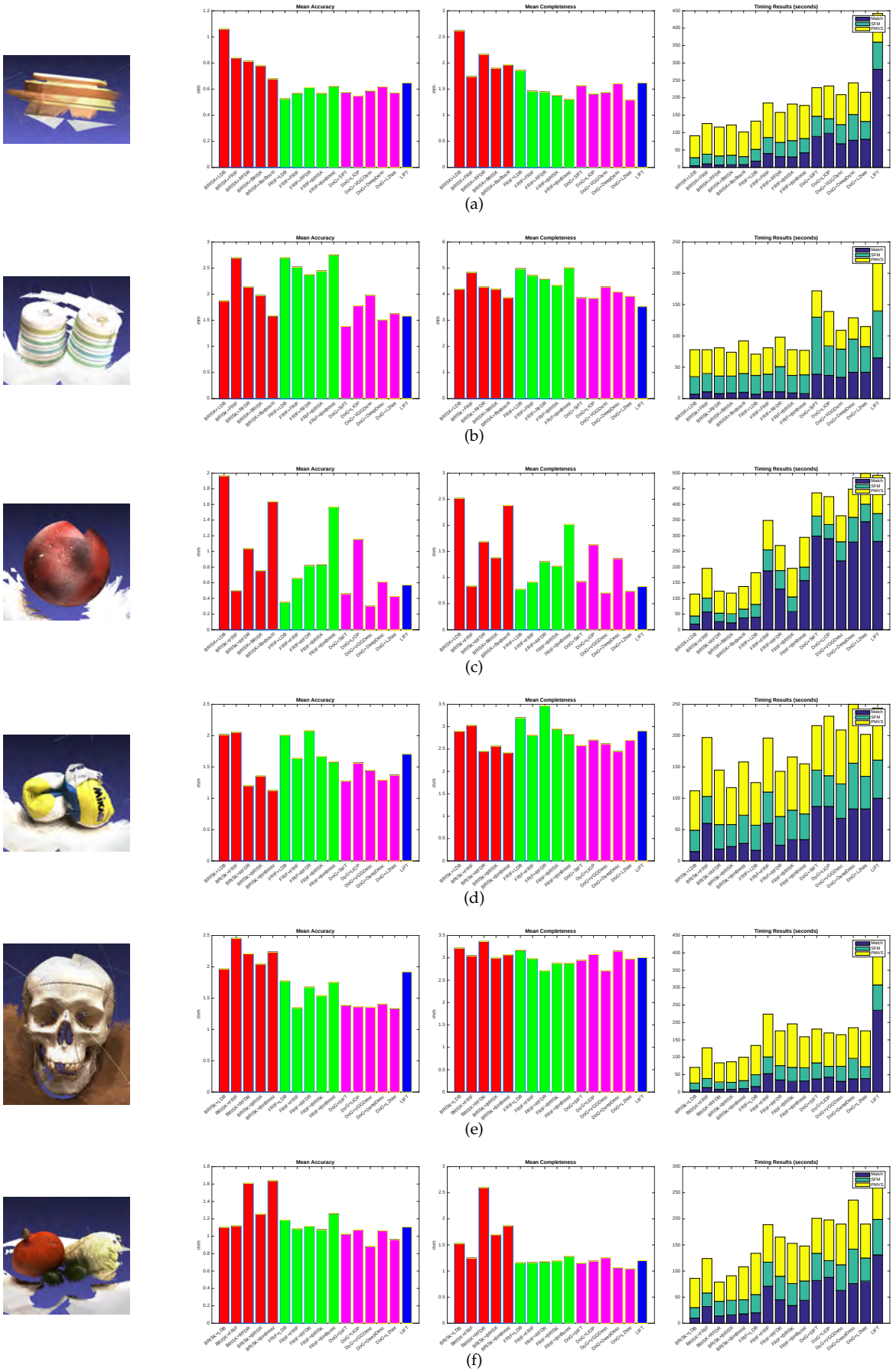


Fig. 4. Performance of scenes that have large accuracy and completeness variances among different evaluated methods. From left to right are: the scene, mean accuracy of different methods, mean completeness of different methods, and the timing results in different stages of different methods.

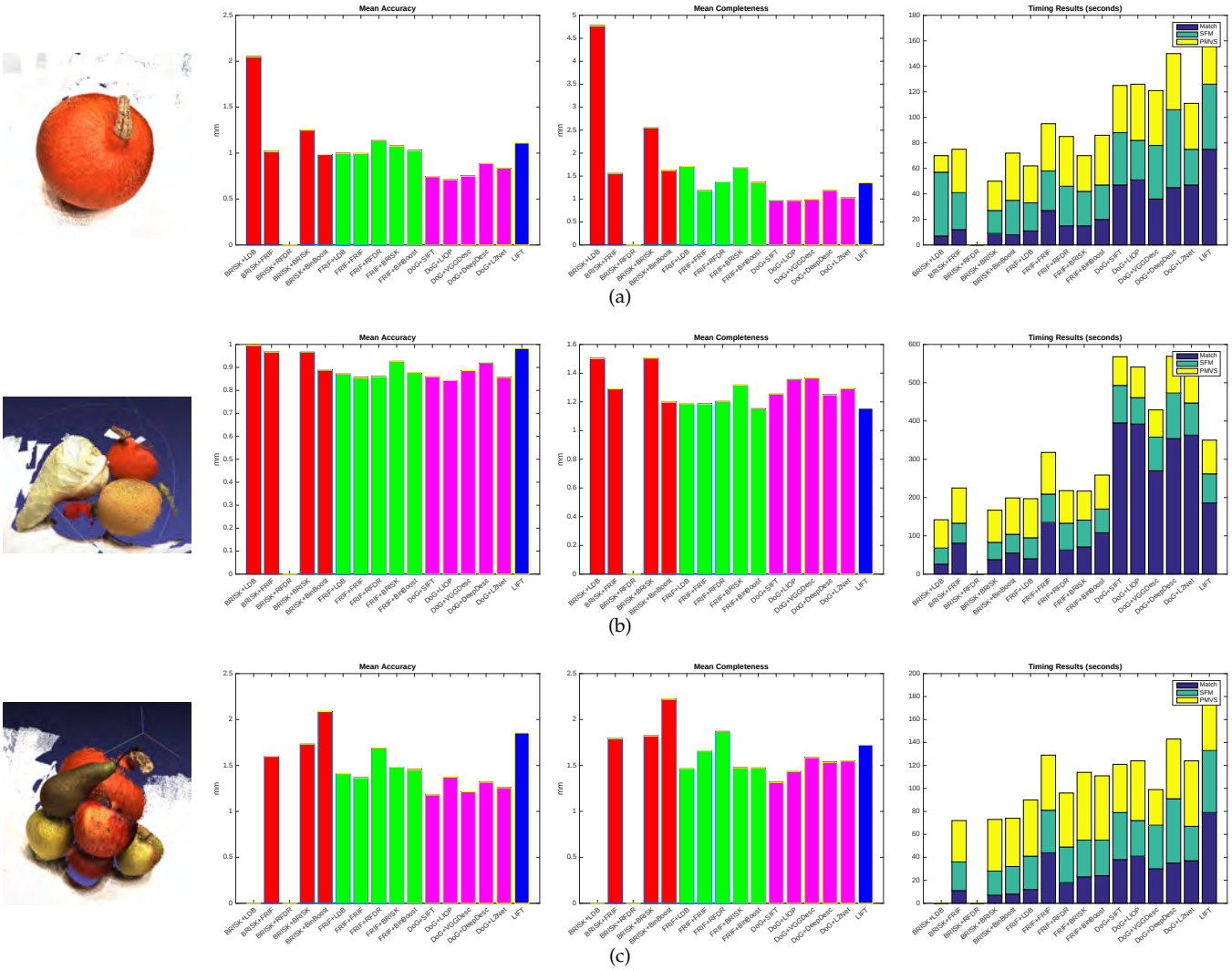


Fig. 5. Performance of scenes that at least one method fails to obtain the reconstruction result. If one method fails, there is no bar shown in the related figures.

choice than using binary features if one does not consider the running time.

(3) For the float type features, the learning based descriptors do not necessarily outperform the handcrafted ones. The baseline SIFT performs rather well. Similar results can be observed for the binary features, among which the handcrafted ones are better than many learned ones in most cases.

(4) In most cases, the running times of SFM and PMVS for all evaluated methods are similar, the main difference of total running time lies in the matching time. In general, using BRISK keypoint requires less running time than using other keypoints. For either BRISK or FRIF keypoints, using FRIF descriptor requires more matching time than other binary descriptors, thus needs more time to do the reconstruction task. Among all the evaluated methods, using float features is more time consuming since matching binary features is more efficient. Due to the smaller descriptor length, using VGGDesc requires the least running time among all the evaluated float features. L2Net usually requires less time than SIFT and DeepDesc although all of them have the same descriptor length. This implicitly indicates that L2Net could generate better matching results (i.e., similar number of matches but with higher precision), thus requiring less time for SFM.

Scenes that at least one method fails. In this case, it refers to the most challenging scene types for 3D reconstruction since one may fail if the local feature used for image matching is not chosen appropriately. The results are shown in Fig. 5. Similar to scenes in Fig. 4, here the scenes are also textureless or contain repeatable textures which are quite challenging for image matching based on local features. We can find that all the failures are from the combinations with BRISK as keypoint detector. More specifically, using LDB descriptor leads to failure for one scene, while using RFD is responsible for 3 failed cases. Even in cases that using BRISK keypoints can be survived to get a reconstruction result, it is usually less accurate and complete than using other keypoints. Considering together with the performance of BRISK keypoint for scenes with large performance variance (cf. Fig. 4), it is clear that BRISK keypoint is less suitable for reconstructing scene types where texture lacks. However, we have to acknowledge that it is a good choice for easy scene types with rich textures because it requires less time

to obtain better accuracy. While for the other keypoints, DoG is slightly better. Taking Fig. 3 to Fig. 5 altogether, it is interestingly to see that when the scene type becomes more and more challenging, using float type features gradually shows its superiority over binary features. Even though, using FRIF keypoint with one binary descriptor is still a good choice for 3D reconstruction with moderate number of images captured from controlled conditions (e.g., fixed viewpoints) as it requires less running time than using float type features and obtains comparable accuracy and completeness. While among the float type features, the reconstruction results of LIFT is less accurate due to the larger localization errors of LIFT than that of DoG.

7 Evaluation on Large Scale Structure from Motion Dataset

7.1 Dataset and Evaluation Metrics

Apart from the controlled case of image capturing, we also evaluate all these local features on 3D reconstruction from a large collection of unordered Internet images, which is the case of most large scale applications of 3D reconstruction, i.e., reconstructing landmarks or cities. For this experiment, we choose the large scale structure from motion dataset [27]. This dataset contains images of several landmarks across the world. For each landmark, it has several thousands of images obtained from the Internet. Different from the previous tested DTU MVS dataset, each image set of one landmark contains a large portion of unrelated images as distractors. On the contrary, the DTU MVS dataset only contains images of one scene from different viewpoints with considerable overlaps. Meanwhile, since there is no constraint on these collected images, they inevitably contain many low quality and non-overlapping images. For these reasons, this dataset is more challenging for feature matching across many images, and so for 3D reconstruction. In our experiment, we use eight subsets of this dataset that have images ranging from one to three thousands and are distributed over a large area. These subsets include: *Gendarmenmarket*, *Madrid Metropolis*, *Pizza del Popolo*, *Alamo*, *Roman Forum*, *NYC Library*, *Montreal Notre Dame*, and *Yorkminster*. The numbers of images contained in these subsets are listed in Table 2.

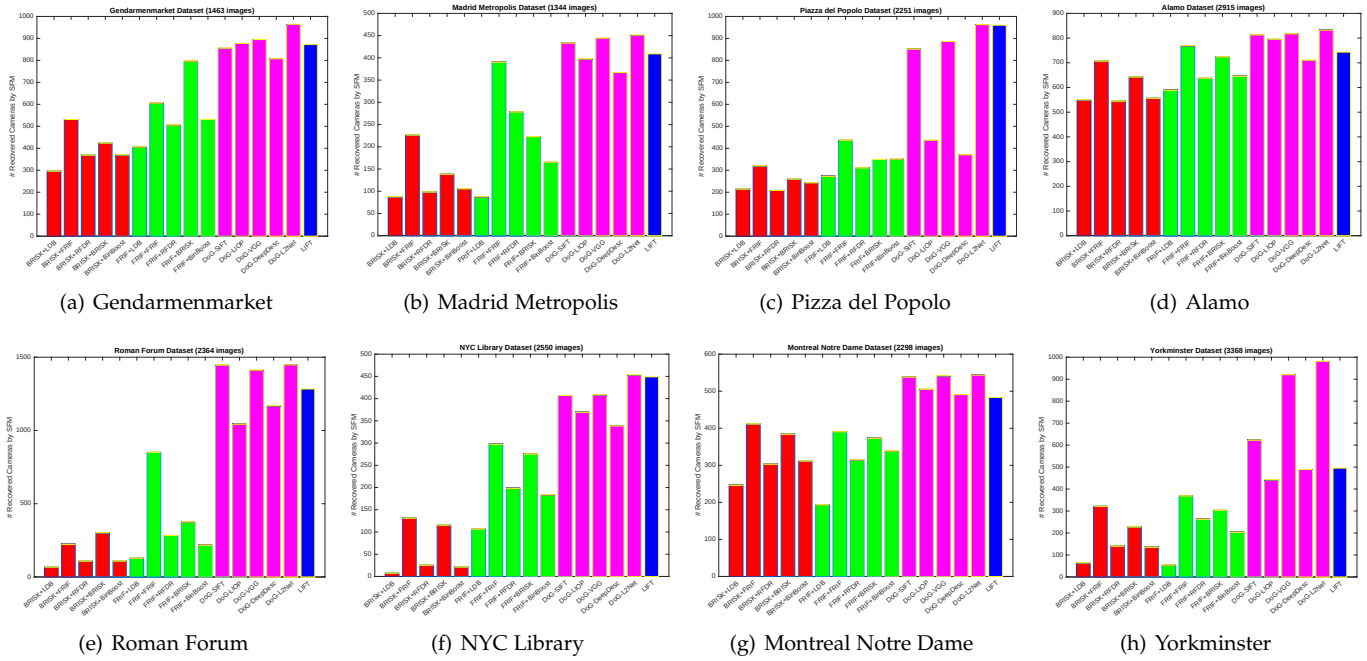
Since there is no groundtruth 3D model available for this dataset, we use the *numbers of recovered images*, *sparse points* and *dense points* as performance indicators for different methods. In general, the more number of cameras is recovered, the more parts of the scene that the reconstruction contains, i.e., a higher completeness of the reconstruction will be. In addition, the following PMVS procedure is highly related to the number of recovered cameras and more cameras often lead to more number of 3D points generated by PMVS. In other words, if one could recover as many cameras as possible, a better reconstruction is expected. As a result, the number of recovered cameras could be a reasonable measurement for a practical image based 3D reconstruction system. In addition, both the numbers of sparse points outputted by SFM and dense points outputted by PMVS quantify the completeness of the reconstruction. More number of sparse points implies that there are more correct matching points established by feature matching. More number of dense points reflects that the reconstruction is better in terms of a higher overall completeness. However, as pointed out before, they are highly related to the number of recovered images and benefited from more recovered images. Thus, they are treated as auxiliary metrics in addition to the number of recovered images. These three metrics are directly related to the reconstruction quality. Moreover, we also include the *track length* and *reprojection error* of bundle adjustment in SFM as indirect metrics to discuss and compare different methods on this specific task. The track length measures the number of image projections per sparse point, where a track is defined as a keypoint that has been matched over many images. We report average track length of all sparse points to evaluate the robustness of a reconstruction. Usually, a larger track length supplies more redundancy for camera pose estimation and triangulation of sparse points in bundle adjustment. The reprojection error measures how the sparse points fit the image projections according to the recovered cameras.

7.2 Results and Analysis

The results are shown in Fig. 6 and Table 2. In Fig. 6, it shows the number of recovered images for different methods on the eight evaluated subsets, and the caption of each figure indicates the subset name as well as its number of images. For each landmark, performance on the largest reconstructed component is reported. As clearly demonstrated by Fig. 6 and Table 2, the float type features generally outperform the binary ones with a significantly large margin, especially in terms of the numbers of recovered images, sparse points and dense points. This result is different from the one observed in the previous MVS dataset, where using binary features could achieve comparable results to those of using float type features. Such a superior performance of the float type features demonstrates their good generalization ability as in this case the image based 3D reconstruction is more challenging. Considering the fact that there are many unrelated images exist in this uncontrolled image capturing situation, the inferior performance of binary features implies that they are sensitive to the distractors, i.e., the local features extracted from unrelated images.

For the binary features, using FRIF keypoint should be a first choice since it usually leads to better results than using BRISK keypoint, no matter which binary descriptor is used. In some cases, when combined with an appropriate descriptor (most time such case is using FRIF descriptor), using FRIF keypoint can even produce comparable performance to that of using float type features. The better results of using FRIF keypoint than using BRISK keypoint are also consistent to the observations found in the MVS dataset. An interesting phenomenon is that the learned binary descriptors are not as competitive as the handcrafted ones although they have been reported with better matching performance. This further validates our motivation that the performance of local features evaluated on matching benchmarks can not be directly applied to the task of image based 3D reconstruction.

For the handcrafted float type descriptors, the performance of the most traditional SIFT is still very competitive and stable across different landmarks while LIOP fails to



(a) Gendarmenmarket (b) Madrid Metropolis (c) Pizza del Popolo (d) Alamo

(e) Roman Forum (f) NYC Library (g) Montreal Notre Dame (h) Yorkminster

Fig. 6. The number of recovered cameras by using different local features for 8 different landmarks distributed over a large area. In the title of each figure, it describes the name of landmark (i.e., the used subset) and its number of images.

reconstruct a large part of the scene for the third and eighth landmarks (Figs. 6(c) and (d)). For the learned descriptors, DeepDesc performs as worse as LIOP, while other learning based methods produce more complete reconstructions than using handcrafted features. Especially, the advanced CNN based learning method, L2Net, performs the best, which is followed by the traditional learning method, i.e., convex optimization. The end to end CNN solution of feature extraction and description like LIFT only performs on par with the baseline SIFT, demonstrating a long way is still required for learning the whole stuff of feature matching.

Regarding the indirect metrics, using float type features always leads to a larger track length than using binary ones, indicating the superior ability of float features in distinguishing matching and non-matching points among many unrelated images. It is worth to note that the larger track length results in better reconstruction when the track length is relatively small (as in most cases of using binary features), and such positively relationship becomes weak after the track length reaches a large value (as in the case of using float features). The underlying reason is that although larger track length is favored by bundle adjustment in SFM, the performance could saturate. Thus there may be no practical difference when the track length is large, which is the case of using float features. This is also the reason why the method (e.g., SIFT) with the largest track length may not necessarily generate the most complete reconstruction. By looking at the reprojection error, using LIOP has the smallest error among all the float features. However, it could be observed that the reprojection error is highly related to the track length, i.e., a shorter track length usually results in lower error. Such a correlation between reprojection error and track length also holds in cases of using binary features. In addition, the reprojection error only measures how the observations fit the estimated model which is not the groundtruth. Due to these reasons, we think that its numerical values are less meaningful among different methods. Consequently, it is good enough to evaluate the ability of different local features for image based 3D reconstruction by only considering the metrics that are directly related to the reconstruction quality, i.e., the numbers of recovered images, sparse points and dense points, while the other metrics such as reprojection error and track length are indirectly related to the reconstruction quality and should only be considered for reference in evaluating different methods.

Considering all the evaluation results together, we can see that although L2Net with DoG keypoint performs the best, the classical SIFT is still very competitive and performs consistently good across various situations. Such results demonstrate that the learning based methods have potential to replace SIFT, however, the improvement is not significant enough right now such that SIFT is still the primary choice for image based 3D reconstruction in this community. Learning good features for this specific task is quite imperative.

8 Conclusion

In this paper, we provide an extensively comparative study of popular local features for the task of image based 3D reconstruction. We focus on how the matching quality of different local features affects the final reconstruction performance, either in terms of accuracy and completeness or indicated by the numbers of recovered cameras, sparse points and dense points. Our evaluation covers a wide range of the state of the art local features, ranging from the traditional handcrafted ones to the recently popular learning based ones. Meanwhile, we also include both float type feature descriptors and binary ones to have a thorough

TABLE 2
Evaluation of different local features on the large scale SFM dataset [27]. B: BRISK, F: FRIF, L: LDB, R: RFDR, BB: BinBoost, S: SIFT, LP: LIOP, V: VGG, D: DeepDesc, LN: L2Net. For each metric, the best results of each keypoint among different descriptors are highlighted in bold. Quantities of metrics are in brackets.

Dataset Name (# images)	Metrics	BRISK Keypoint					FRIF Keyoint					DoG Keypoint					LIFT
		B	F	L	R	BB	B	F	L	R	BB	S	LP	V	D	LN	
Gendarmenmarkt (1463)	# recovered images	423	**531**	298	368	369	**796**	605	405	505	531	855	877	895	808	**964**	872
	# sparse pts (K)	56	**76**	32	44	33	**129**	101	56	83	77	96	103	98	75	**116**	78
	# dense pts (K)	734	**989**	386	527	510	**1184**	1134	489	663	803	1178	1463	1252	982	**1629**	1138
	reproj. err (px)	0.63	0.71	0.73	0.60	**0.59**	**0.66**	0.72	0.67	0.73	0.68	0.90	**0.59**	0.78	0.81	0.70	0.93
	track length	3.58	**4.07**	3.22	3.37	3.44	4.41	**4.94**	3.80	4.25	4.30	**6.82**	5.25	6.62	6.05	6.19	6.44
Madrid Metropolis (1344)	# recovered images	138	**226**	87	99	106	223	**391**	87	279	165	434	397	444	367	**451**	409
	# sparse pts (K)	16	**34**	21	24	11	28	**64**	16	41	21	53	42	54	38	**60**	33
	# dense pts (K)	177	**304**	91	100	93	341	**538**	102	297	311	477	543	578	447	**657**	522
	reproj. err (px)	0.54	0.66	0.65	0.54	**0.48**	0.57	0.68	**0.56**	0.64	0.61	0.65	**0.48**	0.63	0.57	0.56	0.82
	ave. track len.	3.60	4.08	3.59	**4.29**	3.36	4.28	**5.19**	3.80	4.14	4.00	**7.12**	6.29	7.09	6.70	6.65	6.93
Piazza del Popolo (2251)	# recovered images	259	**319**	216	208	244	349	**436**	274	312	350	853	436	886	371	**963**	960
	# sparse pts (K)	31	**45**	23	20	24	47	**57**	30	44	43	68	38	77	33	**101**	78
	# dense pts (K)	400	**465**	235	179	396	481	**652**	490	630	574	1012	477	1239	440	**1468**	1400
	reproj. err (px)	0.58	0.69	0.69	**0.54**	0.58	**0.67**	0.73	0.68	0.74	0.73	1.59	**0.53**	0.65	0.66	0.55	0.78
	track length	4.20	**5.04**	3.96	3.57	3.98	5.87	**6.13**	4.89	5.01	5.38	6.54	6.72	**7.84**	7.08	6.75	7.51
Alamo (2915)	# recovered images	634	**706**	549	544	556	725	**768**	590	637	647	814	795	818	710	**833**	743
	# sparse pts (K)	190	**210**	156	149	145	232	**241**	189	221	203	155	146	148	110	**172**	96
	# dense pts (K)	750	**1052**	678	995	937	**1074**	929	873	893	785	917	950	868	837	**1420**	733
	reproj. err (px)	0.48	0.58	0.58	0.44	**0.43**	0.54	0.62	**0.53**	0.58	0.55	0.55	**0.44**	0.60	0.51	0.52	0.75
	track length	5.37	**5.77**	4.58	4.72	4.64	7.53	**8.11**	6.04	6.49	6.80	9.62	7.72	9.66	9.05	8.64	**14.34**
Roman Forum (2364)	# recovered images	**300**	225	66	107	107	374	**850**	127	283	219	1445	1045	1411	1167	**1449**	1283
	# sparse pts (K)	**26**	18	12	7	8	26	**69**	8	19	14	92	79	97	70	**126**	57
	# dense pts (K)	**145**	91	29	33	40	153	**378**	40	97	84	543	513	562	545	**726**	549
	reproj. err (px)	0.42	0.52	0.46	0.36	**0.35**	0.45	0.54	**0.42**	0.49	0.49	0.75	**0.37**	2.19	0.49	0.45	0.67
	track length	4.08	4.15	3.19	**4.30**	4.08	5.01	5.13	4.68	4.78	**5.32**	7.47	5.91	**7.74**	6.84	6.74	7.28
NYC Library (2550)	# recovered images	115	**132**	7	25	22	276	**298**	106	199	183	407	370	408	339	**453**	449
	# sparse pts (K)	8	**9**	0.06	0.56	0.79	23	**26**	7	15	13	17	30	20	23	**42**	25
	# dense pts (K)	35	**40**	7	12	9	133	**142**	35	67	70	207	202	190	196	**329**	263
	reproj. err (px)	0.41	0.54	**0.39**	0.42	0.40	0.48	0.54	**0.43**	0.49	0.47	0.89	**0.43**	0.69	0.51	0.49	0.69
	track length	3.84	4.31	4.2	**5.29**	4.01	4.69	**4.73**	3.82	4.33	4.10	6.26	5.94	6.50	6.19	6.72	**6.89**
Montreal Notre Dame (2298)	# recovered images	385	**411**	247	304	311	374	**391**	194	314	338	539	505	541	491	**544**	483
	# sparse pts (K)	57	**68**	35	34	37	45	**50**	19	40	37	59	50	58	47	**67**	26
	# dense pts (K)	222	**252**	97	187	189	237	**247**	73	146	181	261	252	267	233	**294**	220
	reproj. err (px)	0.51	0.64	0.60	**0.48**	0.49	0.53	0.62	**0.51**	0.59	0.59	0.63	**0.48**	0.61	0.55	0.54	0.76
	track length	5.17	**5.80**	4.05	4.28	4.37	5.85	**6.70**	4.40	4.86	5.30	**9.80**	7.64	9.55	8.04	8.45	9.52
Yorkminster (3368)	# recovered images	229	**321**	63	142	137	303	**370**	53	264	205	624	441	921	489	**981**	495
	# sparse pts (K)	31	**43**	15	21	19	32	**39**	10	23	19	69	51	66	49	**128**	34
	# dense pts (K)	116	**155**	35	61	56	164	**216**	46	122	107	351	258	455	235	**509**	261
	reproj. err (px)	0.44	0.53	**0.36**	0.41	0.41	0.47	0.54	**0.36**	0.49	0.49	0.54	**0.42**	0.76	0.67	0.48	0.66
	track length	3.45	**3.94**	2.69	3.03	3.04	4.22	**4.66**	3.18	3.87	3.71	**7.13**	5.75	5.36	5.82	6.32	7.05

and comprehensive evaluation. Not only the studied local features have a large diversity, the evaluated datasets also cover the two main application situations of image based 3D reconstruction. One is a controlled case where all images are taken from different viewpoints of the reconstructed scene so that all images have a considerable range of overlap. The other is a general case where many unrelated images exist in the image set of the reconstructed scene. For the first case, we choose to use the recently proposed DTU MVS datasets, which contain various scene types with specifically designed image capturing positions and supply the groundtruth 3D points that facilitate an objective and quantitative evaluation of the reconstruction results. While for the latter case, we choose to use the Internet scale image sets of landmarks, each of which contains a large number of related images and distractors.

Such a dedicated consideration on the evaluated methods and datasets makes our work potentially be a practical guidance for researchers on 3D reconstruction applications. Our experimental results reveal that for the controlled case where no distracting images exist, using binary features is good enough to produce the state of the art 3D reconstruction results with only a fraction of time of using float type features. However, for the large scale freely collected image set with many distractors, using binary features can not guarantee the good performance. The float type descriptors are the most competitive ones in this case even though they need more time to establish point correspondences. Among the evaluated float type descriptors, using recently learned descriptors, such as VGGDesc and L2Net, can lead

to better results than using handcrafted ones (SIFT, LIOP). However, the pioneering CNN-based descriptor learning method (DeepDesc) is not as competitive as these two learned descriptors. Meanwhile, the most traditional SIFT also produces very good results among all the evaluated features, which explains the fact that SIFT is still the primary choice for this task. This also implies that it still requires a lot of efforts to improve the general matching performance of local features. The good results of the learned descriptors further encourage the potential of learning descriptors, but it has to be significantly better and much more robust than the existing SIFT so as to replace it in this task. What is more, how to learn the whole stuff of feature detection and description together still requires lots of works to do, as shown by the results of LIFT which are even inferior to the baseline in some cases. Finally, as binary features are rather competitive in controlled image capturing situation while preserving computational and memory efficiency, it is necessary to develop more powerful binary features with high discriminative ability so as to make them suitable for more general case of image based 3D reconstruction.

ACKNOWLEDGMENT

Thanks for the associated editor and anonymous reviewers for their helpful suggestions to improve the quality of this paper.

REFERENCES

[1] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.

[2] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[3] X. Peng, X. Yu, K. Sohn, D. N. Metaxas, and M. Chandraker, "Reconstruction-based disentanglement for pose-invariant face recognition," in *International Conference on Computer Vision*, 2017, pp. 1623–1632.

[4] J. Heinly, J. L. Schonberger, E. Dunn, and J.-M. Frahm, "Reconstructing the world* in six days," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3287–3295.

[5] J. Schönberger and J.-M. Frahm, "Structure-from-Motion Revisited," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[7] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 102–119, 2010.

[8] Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit, "Tilde: A temporally invariant learned detector," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5279–5288.

[9] B. Fan, Z. Wang, and F. Wu, "Local image descriptor: Modern approaches," Springer, 2015.

[10] B. Fan, F. Wu, and Z. Hu, "Rotationally invariant descriptors using intensity order pooling," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2031–2045, 2012.

[11] Z. Wang, B. Fan, and F. Wu, "Local intensity order pattern for feature description," in *International Conference on Computer Vision*, 2011, pp. 603–610.

[12] T. Trzcinski, M. Christoudias, and V. Lepetit, "Learning image descriptors with boosting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 597–610, 2015.

[13] Y. Tian, B. Fan, and F. Wu, "L2Net: Deep learning of discriminative patch descriptor in euclidean space," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6128–6136.

[14] B. Fan, Q. Kong, T. Trzcinski, Z. Wang, C. Pan, and P. Fua, "Receptive fields selection for binary feature description," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2583–2595, 2014.

[15] K. Simonyan, A. Vedaldi, and A. Zisserman, "Learning local feature descriptors using convex optimisation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1573–1585, 2014.

[16] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Neural Information Processing Systems*, 2017, pp. 4829–4840.

[17] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[18] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *International Conference on Computer Vision*, 2011, pp. 2564–2571.

[19] S. Leutenegger, M. Chli, and R. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *International Conference on Computer Vision*, 2011, pp. 2548–2555.

[20] Z. Wang, B. Fan, and F. Wu, "FRIF: Fast robust invariant feature," in *British Machine Vision Conference*, 2013.

[21] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: Learned invariant feature transform," in *European Conference on Computer Vision*, 2016, pp. 467–483.

[22] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[23] J. Heinly, E. Dunn, and J.-M. Frahm, "Comparative evaluation of binary features," in *European Conference on Computer Vision*, 2012, pp. 759–773.

[24] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," in *International Conference on Pattern Recognition*, 2012, pp. 2681–2684.

[25] H. Aanæs, A. L. Dahl, and K. Steenstrup Pedersen, "Interesting interest points," *International Journal of Computer Vision*, vol. 97, no. 1, pp. 18–35, 2012.

[26] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, "HPatches: A benchmark and evaluation of handcrafted and learned local descriptors," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3852–3861.

[27] K. Wilson and N. Snavely, "Robust global translations with 1dsfm," in *European Conference on Computer Vision*, 2014, pp. 61–75.

[28] J. L. Schönberger, A. C. Berg, and J.-M. Frahm, "PAIGE: PAirwise Image Geometry Encoding for improved efficiency in structure-from-motion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1009–1018.

[29] S. Zhu, R. Zhang, L. Zhou, T. Shen, T. Fang, P. Tan, and L. Quan, "Very large-scale global sfm by distributed motion averaging," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4568–4577.

[30] C. Wu, "Towards linear-time incremental structure from motion," in *International Conference on 3D Vision*, 2013, pp. 127–134.

[31] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[32] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards internet-scale multi-view stereo," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1434–1441.

[33] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *International Conference on Computer Vision*, 2015, pp. 118–126.

[34] X. Yang and T. Cheng, "Local difference binary for ultra-fast and distinctive feature description," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 188–194, 2014.

[35] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs, "Large scale multi-view stereopsis evaluation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 406–413.

[36] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1/2, pp. 43–72, 2005.

[37] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 263–284, 2007.

[38] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 815–830, 2010.

[39] P. Alcantarilla, A. Bartoli, and A. Davison, "KAZE features," in *European Conference on Computer Vision*, 2012, pp. 214–227.

[40] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[41] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, "BRIEF: Computing a local binary descriptor very fast," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1281–1298, 2012.

[42] Z. Wang, B. Fan, and G. W. an Fuchao Wu, "Exploring local and overall ordinal information for robust feature description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 11, pp. 2198–2211, 2016.

[43] S. Gauglitz, T. Höllerer, and M. Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International Journal of Computer Vision*, vol. 94, no. 3, pp. 335–360, 2011.

[44] M. Bauml and R. Stiefelhagen, "Evaluation of local features for person re-identification in image sequences," in *IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2011, pp. 291–296.

[45] S. Madeo and M. Bober, "Fast, compact, and discriminative: Evaluation of binary descriptors for mobile applications," *IEEE Transactions on Multimedia*, vol. 19, no. 2, pp. 221–235, 2017.

[46] L. Liu, P. Fieguth, X. Wang, and M. Pietikainen, "Evaluation of LBP and deep texture descriptors with a new robustness benchmark," in *European Conference on Computer Vision*, 2016, pp. 69–86.

[47] L. Liu, P. Fieguth, Y. Guo, X. Wang, and M. Pietikainen, "Local binary features for texture classification: Taxonomy and experimental study," *Pattern Recognition*, vol. 62, pp. 135–160, 2017.

[48] B. Fan, Q. Kong, W. Sui, Z. Wang, X. Wang, S. Xiang, C. Pan, and P. Fua, "Do we need binary features for 3d reconstruction?" in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 1126–1135.

[49] J. L. Schönberger, H. Hardmeier, T. Sattler, and M. Pollefeys, "Comparative evaluation of hand-crafted and learned local features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6959–6968.

[50] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building Rome in a day," *Communications of the ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[51] S. Shen, "Accurate multiple view 3D reconstruction using patch-based stereo for large-scale scenes," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1901–1914, 2013.

[52] Y. Lou, N. Snavely, and J. Gehrke, "MatchMiner: Efficient spanning structure mining in large image collections," in *European Conference on Computer Vision*, 2012, pp. 45–58.

[53] M. Brown, G. Hua, and S. Winder, "Discriminative learning of local image descriptors," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 43–57, 2011.

[54] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," *Journal of Machine Learning Research*, vol. 11, pp. 2543–2596, 2010.

[55] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "MatchNet: Unifying feature and metric learning for patch-based matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3279–3286.

[56] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4353–4361.

[57] V. K. B. G, G. Carneiro, and I. Reid, "Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5385–5394.

[58] K. M. Yi, Y. Verdie, P. Fua, and V. Lepetit, "Learning to assign orientations to feature poitns," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 107–116.

[59] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Neural Information Processing Systems*, 2015, pp. 2017–2025.

[60] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

[61] D. M. Mount and S. Arya, "ANN: A library for approximate nearest neighbor searching," [Online]. https://www.cs.umd.edu/ mount/ANN/.

[62] R. Mur-Artal, J. Montiel, and J. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[63] E. Mair, G. Hager, D. Burschka, M. Suppa, and G. Hirzinger, "Adaptive and generic corner detection based on the accelerated segment test," in *European Conference on Computer Vision*, 2010, pp. 183–196.

[64] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," in *International Conference on Computer Vision*, 2001, pp. 525–531.

[65] M. Muja and D. G. Lowe, "FLANN: Fast library for approximate nearest neighbors," [Online]. http://www.cs.ubc.ca/research/flann.

[66] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 519–528.

Bin Fan received the B.S. degree in automation from the Beijing University of Chemical Technology, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2011. During 2015 to 2016, he spent one year as a visiting scholar at CVLab, EPFL. He is currently an Associate Professor with the Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision, image processing and pattern recognition. His articles have been published in major venues including TPAMI, TIP, TNNLS, TMM, CVPR, ICCV, ECCV, AAAI.

Qingqun Kong received her B.S. degree in automation from Ocean University of China, in 2008, and the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2013. Currently, she is an associate professor with the Institute of Automation, Chinese Academy of Sciences. She researches on computer vision, pattern recognition and brain inspired intelligence.

Xinchao Wang is currently a tenure-track Assistant Professor at Stevens Institute of Technology, New Jersey, United States. Before joining Stevens, he was an SNSF postdoctoral fellow at University of Illinois Urbana-Champaign (UIUC). He received a PhD from Ecole Polytechnique Federale de Lausanne (EPFL) in 2015, and a first-class honorable degree from Hong Kong Polytechnic University (HKPU) in 2010. His research interests include artificial intelligence, computer vision, machine learning, medical image analysis, and multimedia. His articles have been published in major venues including CVPR, ICCV, ECCV, NeurIPS, AAAI, MICCAI, TPAMI, TIP, TMI, and TNNLS. He serves as an associate editor of Journal of Visual Communication and Image Representation (JVCI), a senior program committee member of AAAI'19, and an area chair of ICME'19 and ICIP'19.

Zhiheng Wang received his B.S. degree in mechanical and electronic engineering from Beijing Institute of Technology, China, in 2004, and the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2009. Currently, he is a professor at Henan Polytechnic University and researches on image processing and computer vision.

Shiming Xiang received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, in 1993, the M.S. degree from Chongqing University, Chongqing, China, in 1996, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, China, in 2004. From 1996 to 2001, he was a Lecturer with the Huazhong University of Science and Technology, China. He was a Postdoctorate Researcher with the Department of Automation, Tsinghua University, until 2006. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences, China. His interests include pattern recognition, machine learning, and computer vision. He has published more than 100 papers, some of which were published in the following refereed journals or conferences: TPAMI, IJCV, TIP, TNNLS, TKDE, TMM, TSMC-B, TGRS, TCSVT, NeruIPS, CVPR, ICCV, IJCAI, AAAI, and ACM MM.

Chunhong Pan received the B.S. degree in automatic control from Tsinghua University, China, in 1987, the M.S. degree from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, China, in 1990, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, China, in 2000. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision, image processing, computer graphics, and remote sensing.

Pascal Fua received an engineering degree from Ecole Polytechnique, Paris, in 1984 and the Ph.D. degree in Computer Science from the University of Orsay in 1989. He joined EPFL (Swiss Federal Institute of Technology) in 1996 where he is now a Professor in the School of Computer and Communication Science. Before that, he worked at SRI International and at INRIA Sophia-Antipolis as a Computer Scientist. His research interests include shape modeling and motion recovery from images, analysis of microscopy images, and augmented reality. He has (co)authored over 300 publications in refereed journals and conferences. He is an IEEE Fellow and has been an Associate Editor of IEEE Transactions for Pattern Analysis and Machine Intelligence. He often serves as program committee member, area chair, and program chair of major vision conferences.