

FRIF: Fast Robust Invariant Feature

Zhenhua Wang
wzh@nlpr.ia.ac.cn
Bin Fan
bfan@nlpr.ia.ac.cn
Fuchao Wu
fcwu@nlpr.ia.ac.cn

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, 100190, Beijing, China

Abstract

Establishing robust visual correspondences is a fundamental component of many computer vision applications. However, it is very challenging to obtain high quality features while maintaining a low computational cost. This paper aims to tackle this problem by adopting a novel Fast Robust Invariant Feature (FRIF) for both feature detection and description. The basic idea is to employ a fast approximated LoG detector to select scale-invariant keypoints and incorporate local pattern and inter-pattern information to construct distinctive binary descriptors. A comprehensive evaluation on standard dataset shows that FRIF achieves quite a high performance with a computation time comparable to state-of-the-art real-time features.

1 Introduction

Local image feature is a fundamental component of many computer vision applications such as 3D reconstruction [2], image retrieval [22], object recognition [16] and object categorization [10]. The main goal is to find salient image points that can be repeatedly detected under various image transformations and then construct distinctive and robust representations for them. SIFT [16] is probably the most representative method which produces high quality features at the expense of a relatively high computational cost. SURF [5] reduces the computation time without loss in performance, however, it still cannot be used in real-time applications, especially in the context of mobile device. Recently, several methods are proposed for real-time applications, *e.g.* BRIEF [6], ORB [27], BRISK [14] and FREAK [4]. They combine the FAST-based corner [17, 25, 26] and binary descriptor to achieve high speed. However, their feature detectors are based on the FAST methodology which uses FAST scores to perform scale-space maxima search. The unreliable Fast score makes them less robust than the slower methods and thus get a poor matching score under large image transformations. In practice, it is very challenging to obtain a high quality feature whilst maintaining a low computational cost.

This paper aims to tackle this problem by developing a novel Fast Robust Invariant Feature (FRIF). For feature detection, scale invariant and stable keypoints are selected in the scale space according to our Fast Approximated LoG (FALoG) filter responses. For feature description, distinctive binary descriptors are constructed by encoding both local pattern and

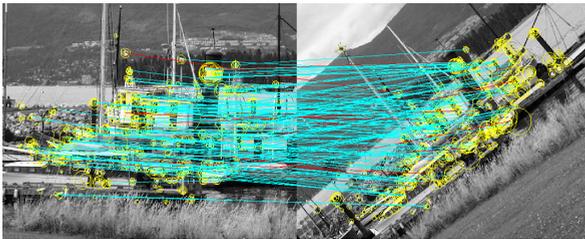


Figure 1: A typical matching result using FRIF for two images with in-plane rotation and scale change. The yellow circles represent detected keypoints whose scales are indicated by the size. The cyan lines are the correct matches which localized in 2 pixels, while the red lines are false matches.

inter-pattern information. By employing factorization and integral image, FALoG can be computed very fast. Meanwhile, the binary descriptor is constructed directly by intensity comparisons. Thus, both the detection and description of FRIF can be done very efficiently, making it suitable for real-time applications.

To validate FRIF, we have evaluated the repeatability of our FALoG detector, the matching quality of our binary descriptor and the overall performance of FRIF respectively. The experiments show that FRIF achieves quite a high performance with a computation time comparable to state of the art, *e.g.* BRISK and FREAK. A typical matching result using FRIF for two images with in-plane rotation and scale change is shown in Figure 1.

The rest of this paper is organized as follows: First, an overview of related work is given in Section 2. Section 3 and Section 4 describe the proposed Fast Approximated LoG detector and mixed binary descriptor respectively. The experimental evaluation is carried out in Section 5, and finally we conclude the paper in Section 6.

2 Related Work

2.1 Feature Detection

Moravec [21] defines corners as points where large intensity variation exists in every direction. Harris and Stephens [12] develop this idea by computing a second-moment matrix H and selecting corners where both eigenvalues of H are large. Since the Harris corners are not scale-invariant, Lindeberg [15] proposes a systematic methodology for automatic scale selection by detecting local extremas over scales of different combinations of γ -normalized derivatives. He develops the Laplacian-of-Gaussian (LoG) and the Determinant of the Hessian matrix to detect blob-like features. Mikolajczyk and Schmid [18] find that the Laplacian function is a particular stable scale-space kernel. They propose the Harris-Laplacian and Hessian-Laplacian detectors which use the Harris measure and the determinant of Hessian matrix to localize features respectively and use the Laplacian to select their characteristic scales. Lowe [16] obtains scale-invariant features by detecting local extremas of the Difference-of-Gaussian over scale space. The DoG is a good approximation of LoG and much faster to compute. Bay *et al.* [5] develop the Fast-Hessian detector which approximates the second order Gaussian derivatives of Hessian matrix with rectangle filters. Since the rectangle filters can be computed in time independent of their sizes using integral images,

the Fast-Hessian detector is about 3 times faster than DoG. Since LoG can also be approximated by the Bi-level Laplacian-of-Gaussian (BLoG) [24], Agrawal *et al.* [3] replace the two circles of BLoG by two boxes or two octagons to get the CenSurE-DOB or CenSurE-OCT detector respectively. To compute these filters efficiently, they introduce slanted integral images which can be used to compute the sum of any trapezoidal area in constant time. Recently, the FAST corner [25] becomes popular in real-time applications for its efficiency. FAST-ER [26] generalizes FAST by allowing it to be optimized for repeatability with little loss of efficiency. AGAST [17] improves its performance by combining specialized decision trees. Since these FAST-based detectors do not deal with scale change, BRISK [14] takes AGAST to detect feature candidates and searches for the FAST score maxima over scale space to achieve scale invariance.

2.2 Feature Description

The most popular feature descriptors are those based on histograms [5, 13, 16, 20, 31, 32, 34], which are shown to outperform other kinds of descriptors such as differential derivatives [29], steerable filters [9], complex filters [28] and moments invariants [11]. SIFT [16] creates a histogram of local gradient orientations and locations, where the gradient orientations are quantized into 8 orientation bins and the space locations are quantized into a 4×4 grid. DAISY [32] extends SIFT by using circular regions defined by Gaussians with increasing variances as the regions go away from the point center, while SURF [5] uses the Haar wavelet responses as the local features. Since the above gradient-based descriptors can only deal with linear illumination changes, some other methods have been proposed to tackle more general illumination changes by using relative intensity orders of pixels rather than the original intensities. Tang *et al.* [31] create a histogram encoding both the ordinal distribution and the spatial distribution. Heikkilä *et al.* [13] combine the strength of SIFT and LBP texture operator [23] to create a histogram of Center-Symmetric Local Binary Pattern (CS-LBP). Fan *et al.* [7, 8] propose MROGH and MRRID by accumulating histograms within intensity order segments. Wang *et al.* [34] propose a novel Local Intensity Order Pattern (LIOP) to encode the local ordinal information and create a histogram of LIOP for each ordinal subregion. In recent years, binary-string descriptors have been developed because of less storage and faster computation. BRIEF [6] uses a relatively small number of binary tests between pixels to represent the local patch as a binary string. ORB [27] extends BRIEF to be rotation invariant by an intensity centroid based orientation estimation, and be more discriminative by learning a good subset of binary tests. BRISK [14] introduces a Gaussian weighted pattern for sampling the neighborhood of keypoints. The long-distance pairs are used to estimate the local dominant orientation and the short-distance pairs are used to build binary descriptor. FREAK [4] proposes a retinal sampling pattern based on the human visual system, and computes the binary descriptor by comparing image intensities over the retinal patterns. D-BRIEF [33] boosts the performance by projecting the image patches into a more discriminative subspace and thresholding their coordinates to build the binary descriptor.

3 Fast Approximated LoG Detector

We develop a scale-invariant feature detector based on the LoG function since it is found to be more stable than other derivative based functions in characteristic scale selection [18]. To the best of our knowledge, CenSurE is the fastest approximation of LoG among existing

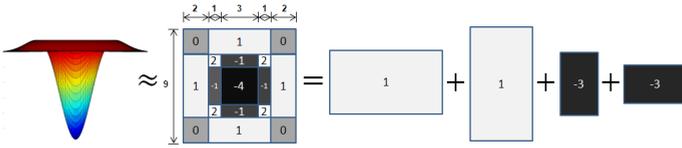


Figure 2: The approximation of LoG kernel.

approaches. However, it treats the surrounding areas equally, losing the main characteristic of LoG. Moreover, it does not perform a scale-space refinement, resulting in a lower accuracy. Hence, CenSurE obtains the advancement of speed at the cost of accuracy. To alleviate above problems, we propose a better approximation of LoG which can be computed very fast by factorization and integral image while preserving most properties of LoG.

The proposed Fast Approximation of LoG (FALoG) filter assigns the surrounding areas with different weights according to their distances to the central point and keeps the DC response to be zero. As shown in Figure 2, a 9×9 FALoG filter is used to approximate LoG with $\sigma = 1.2$. To efficiently compute the response of FALoG filter, it is factorized into four rectangles with different weights. The response of each rectangle can be computed rapidly using integral image. By summing over all the responses of these four rectangle filters, we obtain the response of FALoG filter.

The scale space is implemented in a way similar to BRISK [14]. Specifically, it consists of n octaves c_i and n intra-octaves d_i for $i = \{0, 1, \dots, n-1\}$ where c_0 is the original image and d_0 is a two-thirds downsampling of c_0 . The octaves and intra-octaves are built by progressively half-sampling c_0 and d_0 respectively. The above 9×9 FALoG filter is used in each octave and intra-octave, and therefore $scale(c_i) = 1.2 \times 2^i$ and $scale(d_i) = 1.2 \times 1.5 \times 2^i$.

To obtain scale invariant keypoints, we first compute the integral image for each octave and intra-octave. This can be done very fast since only one scan over the scale space is required. Next, we compute the 9×9 FALoG filter response for each pixel in the scale space separately, and a threshold T_F is applied to filter out weak responses. Then, for the remaining points, non-maximum suppression is performed so that only the extremas of FALoG responses in a local neighborhood over locations and scales are kept as potential keypoints. Finally, we refine the scale and location of each keypoint via quadratic function fitting.

Similar to LoG, FALoG filter has strong responses along edges, which are poorly localized and sensitive to noise. To filter out these edge responses, we compute the ratio of principal curvatures by the trace and determinant of Harris matrix. Keypoint candidates with ratios larger than T_L are considered as edges and are removed. Note that the Harris matrix is computed on the scale-space image. Experiments show that it is better than computing the Harris matrix on the filter response function as CenSurE [3] does.

4 Mixed Binary Descriptor

Both BRISK and FREAK descriptors make use of a specific pattern in the neighborhood of keypoint. For BRISK, the pattern defines locations equally spaced on circles concentric with the keypoint. A Gaussian smoothing with standard deviation σ_i proportional to the distance between the pattern location and central keypoint is applied to avoid aliasing effect. For FREAK, the pattern is similar to the distribution of the retinal ganglion cells. It has more

overlapping areas and the density of pattern locations drops exponentially with the distance to the center increases.

We observed that both of them only consider the inter-pattern intensity relationships, *i.e.* pair-wise intensity comparisons between pattern locations. The loss of local pattern information, which is more consistent under brightness variation, makes them less reliable. Thus, the basic idea of our method is to incorporate both the local pattern and inter-pattern information to improve matching performance. More specifically, the descriptor is constructed based on a modified BRISK pattern which has more overlapping areas than the original one. The local gradients of pattern locations and the inter-pattern intensity comparisons are combined to create the mixed binary descriptor.

Let \mathcal{P} be the set of all the N pattern locations, for each pattern location $p_i = (x_i, y_i) \in \mathcal{P}$, a set of four points $\mathcal{S}(p_i) = \{s_{i,k}, k = 1, 2, 3, 4\}$ are equally sampled on a circle of radius R centered at p_i . As presented by LBP texture operator [23], the local information can be encoded by the intensity relationships between the pattern location p_i and each sampling point $s_{i,k}$. However, this encoding scheme is sensitive to p_i , and thus is not appropriate for binary description. To encode the local information robustly, we make use of the intensity relationships between sampling points $s_{i,k}$.

Let $I(x, \sigma)$ be the smoothed intensity of point x with sigma σ and θ be the local dominant orientation estimated by the average local gradients [4, 14]. For each rotated pattern location p_i^θ , we compare the pair-wise intensities of its sampling points $s_{i,k}^\theta \in \mathcal{S}(p_i^\theta)$. Then, the descriptor is constructed by assembling all the test results into a binary string, each bit b of which corresponds to:

$$b = \text{sign}(I(s_{i,k}^\theta, \sigma_i) - I(s_{i,t}^\theta, \sigma_i)), \quad (1)$$

$$\forall p_i^\theta \in \mathcal{P} \wedge s_{i,k}^\theta, s_{i,t}^\theta \in \mathcal{S}(p_i^\theta) \wedge k, t = 1, 2, 3, 4 \wedge k \neq t.$$

As four points are sampled for each pattern location, the dimension of the descriptor is $N \times C_4^2 = 6N$ bits. It is worth noting that the intensity comparisons between sampling points $s_{i,k}$ is closely related to the local gradient operator, both of which consider the intensity differences between pairs of local samplings.

The above descriptor encodes the local pattern information into binary strings. We complement it with more global information, which is encoded by the inter-pattern intensity comparisons. Let the set \mathcal{A} be all pairs of pattern locations:

$$\mathcal{A} = \{(p_i, p_j) \mid p_i, p_j \in \mathcal{P} \wedge i \neq j\}. \quad (2)$$

A binary string can be computed by comparing the intensities of all the rotated pairs (p_i^θ, p_j^θ) in a subset \mathcal{B} of \mathcal{A} , each bit of which corresponds to $\text{sign}(I(p_i^\theta) - I(p_j^\theta))$. To form the subset \mathcal{B} , we tested three different selection criteria: 1) randomly select M pairs from \mathcal{A} . 2) select the shortest M pairs of \mathcal{A} . 3) select the longest M pairs of \mathcal{A} . Experiments show that the shortest pairs are more stable than others. This result is consistent with BRISK, which uses the short-distance pairs to build the descriptor. The difference is that, in our method only the shortest M pairs are used as a complementary part of the previous local gradient based binary descriptor.

By concatenating the two kinds of complementary binary strings together, we obtain our mixed binary descriptor. As the BRISK and FREAK descriptor are both 512 bits, we select $M = 512 - 6N$ to make the mixed binary descriptor the same dimension to them.

5 Experiments

We have evaluated our method on the standard Oxford dataset [1] which contains images with different geometric and photometric transformations of structured and textured scenes. There are several parameters in our method: 1) the size of FALoG filter S used in each octave, 2) the FALoG filter response threshold T_F , 3) the edge suppression threshold T_L and 4) the local sampling radius R . The 'Graf' sequence were used to investigate the effect of different parameters settings. It is found that the performance of FRIF is insensitive to a regular range of parameter settings. A typical parameters setting is given in Table 1, which is used in the later experiments.

	detector			descriptor
Parameters	S	T_F	T_L	R
Values	9	50	20	3

Table 1: A typical parameters setting of FRIF

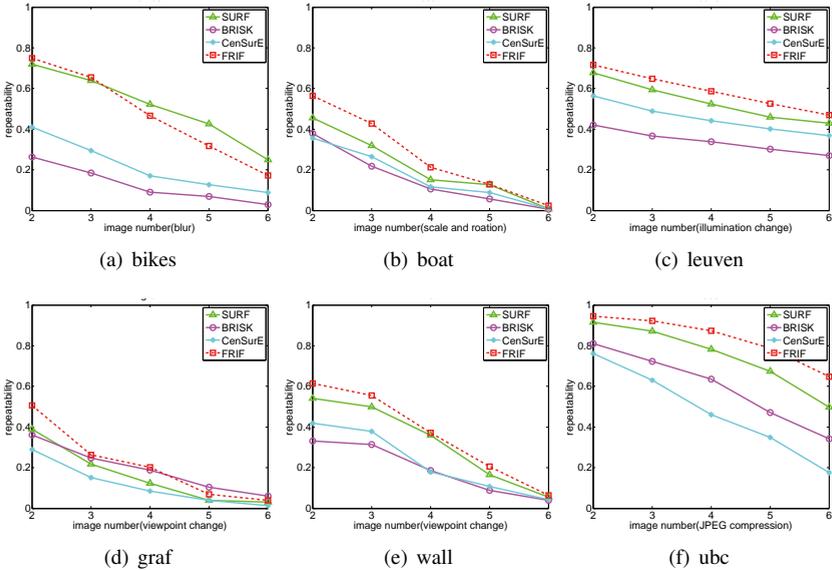


Figure 3: Experimental results of repeatability scores. The effect of image transformation increases with the image number.

5.1 Detector Evaluation

The detectors are evaluated using the repeatability criterion introduced in [30]. We compare FRIF detector with SURF, CenSurE, and BRISK detectors as they are state-of-the-art fast scale invariant detectors. To make a fair comparison, we keep the number of interest points similar for all the detectors by choosing different thresholds.

We first test the repeatability scores of different detectors for gradually increasing transformation. The results are presented in Figure 3. FRIF detector consistently outperforms

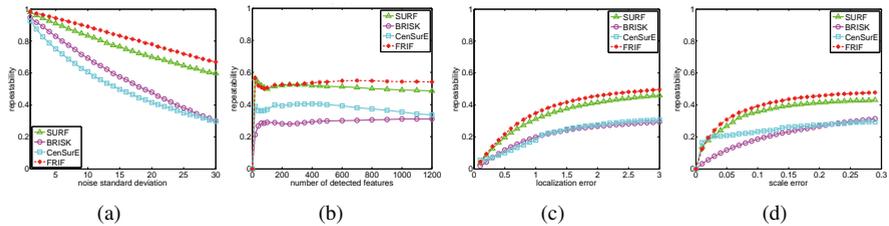


Figure 4: Experimental results of repeatability scores with (a) the amount of Gaussian noise, (b) the number of detected keypoints, (c) the localization error and (d) the scale error varied. The results are averaged over all the images in the dataset.

CenSurE in all cases, which can be attributed to the more suitable weighting strategy and better scale space interpolation. CenSurE approximates LoG by a uniform weighting and detects keypoints without scale-space interpolation, which makes it suffer a low repeatability. FRIF is also better (Bikes, Boat, Leuven, Wall, Ubc) than or comparable (Graf) to BRISK detectors. This can be explained by the fact that BRISK uses the FAST score as the scale-space maxima search criterion which is less reliable than FALOG filter response used by FRIF. Compared to SURF, FRIF also obtains a better repeatability, although a little worse for large image blur (Figure 3(a)).

We also test the robustness of FRIF detector on images corrupted by increasing amount of additive Gaussian noise. The average results over all the images in the dataset are shown in Figure 4(a). It can be seen that all the tested detectors suffer a performance degradation as the noise level increases. However, FRIF detector shows the best robustness and outperforms others. To further validate FRIF detector, we perform a more detailed analysis on the experimental setup based on the results averaged over all the images pairs in the dataset. Figure 4(b) shows how the repeatability is affected by the number of keypoints detected in each image. It is clear that the rank of different detectors is not changed when the number varies from 0 to 1200, therefore the previous experimental results are not affected by this setup. Figure 4(c) and (d) show the repeatability as a function of localization error and scale error respectively. As expected, more keypoints are qualified as correspondences when the threshold relaxes, and the repeatability scores increase. The rank of detectors remains the same, indicating that our experimental setup is not sensitive to the choice of thresholds.

5.2 Descriptor Evaluation

The descriptors are evaluated using *recall* versus *1-precision* curves proposed in [20]. We compare FRIF descriptor with SURF, ORB, BRISK and FREAK descriptors, since SURF is a classical fast descriptor while ORB, BRISK and FREAK are recently proposed binary descriptors. To make a fair comparison, we set the size of local patch the same for all the tested descriptors (31×31). We use BRISK detector to select keypoints since it is the most popular real-time scale invariant feature detector. The results of two image pairs (the 1st vs. the 2nd and the 1st vs. the 4th) for each case are presented in Figure 5.

For all cases except 'Bikes 1-4', FRIF is better than or at least comparable to all the other tested descriptors. This could be explained by the fact that the combination of local pattern and inter-pattern information improves the discriminative power of the descriptors. The performance degradation in 'Bikes 1-4' can be explained by that the local pattern information explored by FRIF is more likely to be affected by the image blur than the pair-wise

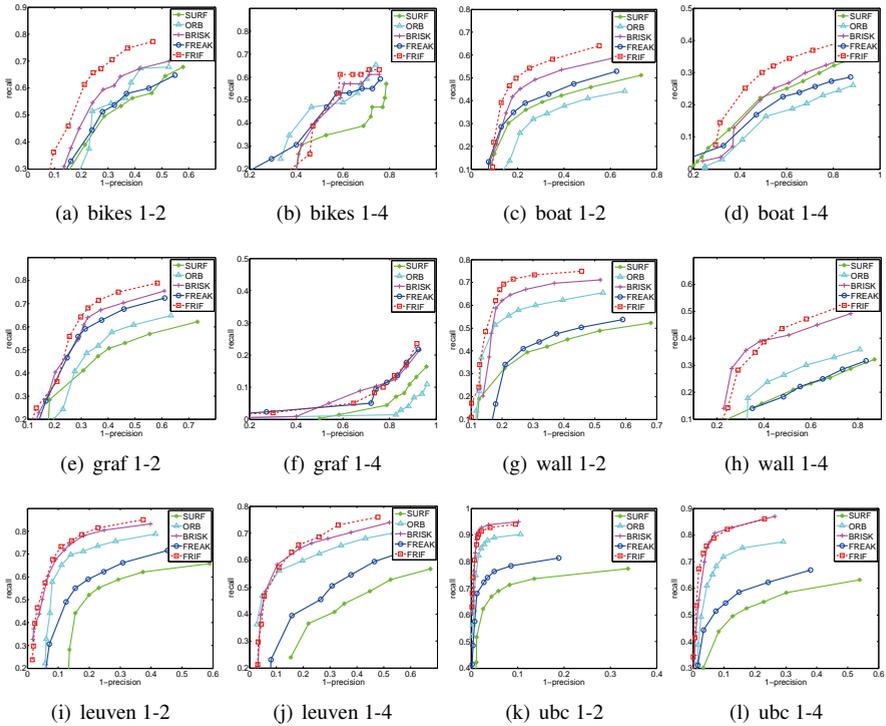


Figure 5: Experimental results for: (a)-(b) image blur, (c)-(d) rotation and scale change, (e)-(h) viewpoint change, (i)-(j) illumination change and (k)-(l) JPEG compression. Note that the scales are different for different figures to improve the clarity of the plots.

information used by ORB, BRISK and FREAK.

5.3 Joint Detection and Description Evaluation

The overall performance of our joint feature detection and description algorithms are evaluated using the matching score criterion introduced in [19]. Since the algorithm which produces highly repeatable interest points with distinctive descriptors will obtain a relatively high matching score, this criterion can be a suitable measurement of the joint performance of detection and description. We compare FRIF features to SURF, ORB, BRISK and FREAK features, and all the features use their default parameters settings to make a fair comparison. As can be observed in Figure 6, FRIF performs the best in all tested cases. This result demonstrates the effectiveness of FALoG detector and mixed binary descriptor used by FRIF.

5.4 Timing Results

In this section, we compare the computation times of FRIF to SURF, ORB, BRISK and FREAK. The experiments are carried out on a desktop with an Intel Core2 Quad 2.83GHz CPU, using the first image of the 'Bike' sequence, and the results are averaged over 100 runs. The total time of joint detection and description as a function of feature numbers is shown in Figure 7, while Table 2 gives a more detailed result when each algorithm produces

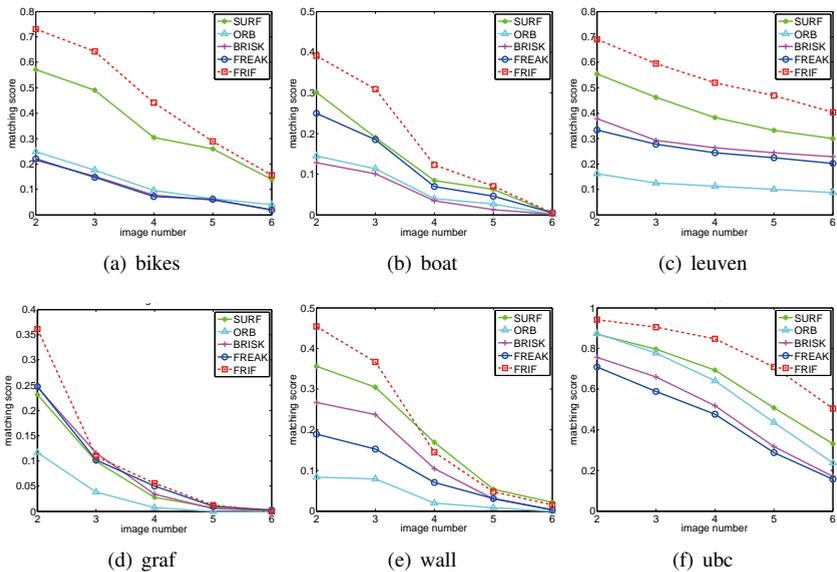


Figure 6: Experimental results of matching scores. The effect of image transformation increases with the image number. Note that the scales are different for different figures to improve the clarity of the plots.

Time per feature	SURF	ORB	BRISK	FREAK	FRIF
Detection time (ms)	0.223	0.015	0.016	0.022	0.024
Description time (ms)	0.181	0.011	0.022	0.010	0.025
Overall time (ms)	0.404	0.026	0.038	0.032	0.049

Table 2: The average time required for processing one feature. Thresholds are adapted in order to detect similar number of keypoints (about 1500) for different algorithms.

about 1500 features. It is clear that FRIF is comparable to ORB, BRISK and FREAK, and all of them are an order of magnitude faster than SURF. In a word, FRIF obtains higher performance without loss of speed, making it suitable for real-time applications.

6 Conclusion

In this paper, we have proposed a Fast Robust Invariant Feature (FRIF) for both feature detection and description. The key advantage of FRIF over existing ones is that it achieves quite a high repeatability and matching performance at a very low computational cost. This makes it a good choice for real-time computer vision applications, especially in the case of mobile device with limited resources. The advantage is achieved by employing a fast approximated LoG filter to detect scale-invariant keypoints in real time and incorporating both local pattern and inter-pattern information to construct a distinctive binary descriptor. Extensive experiments on various image transformations have shown that FRIF feature outperforms the state-of-the-art methods in general.

Acknowledgements: This work is supported by the National Science Foundation of China(91120012, 61075038, 61203277)

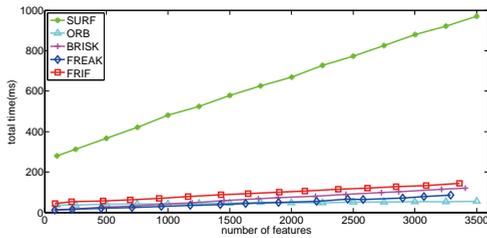


Figure 7: Experimental results of the total time as a function of feature numbers.

References

- [1] <http://www.robots.ox.ac.uk/vgg/research/affine/>.
- [2] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, and R. Szeliski. Building rome in a day. In *International Conference on Computer Vision*, pages 72–79, 2009.
- [3] Motilal Agrawal, Kurt Konolige, and Morten Blas. Censur: Center surround extremas for realtime feature detection and matching. In *European Conference on Computer Vision*, volume 5305, pages 102–115, 2008.
- [4] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. FREAK: Fast retina keypoint. In *Computer Vision and Pattern Recognition*, pages 510–517, 2012.
- [5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: speeded up robust features. In *European Conference on Computer Vision*, pages 404–417, 2006.
- [6] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: binary robust independent elementary features. In *European conference on Computer vision*, pages 778–792, 2010.
- [7] Bin Fan, Fuchao Wu, and Zhanyi Hu. Aggregating gradient distributions into intensity orders: A novel local image descriptor. In *Computer Vision and Pattern Recognition*, pages 2377–2384, 2011.
- [8] Bin Fan, Fuchao Wu, and Zhanyi Hu. Rotationally invariant descriptors using intensity order pooling. *Pattern Analysis and Machine Intelligence*, 34(10):2031–2045, 2012.
- [9] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *Pattern Analysis and Machine Intelligence*, 13:891–906, 1991.
- [10] Peter Gehler and Sebastian Nowozin. On feature combination for multiclass object classification. In *International Conference on Computer Vision*, pages 221–228, 2009.
- [11] Luc J. Van Gool, Theo Moons, and Dorin Ungureanu. Affine/ photometric invariants for planar intensity patterns. In *European Conference on Computer Vision*, pages 642–651, 1996.
- [12] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.

- [13] Marko Heikkilä, Matti Pietikäinen, and Cordelia Schmid. Description of interest regions with local binary patterns. *Pattern Recognition*, 42(3):425–436, 2009.
- [14] S. Leutenegger, M. Chli, and R.Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *International Conference on Computer Vision*, pages 2548–2555, 2011.
- [15] Tony Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30:79–116, 1998.
- [16] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.
- [17] Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In *European conference on Computer vision*, pages 183–196, 2010.
- [18] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *International Conference on Computer Vision*, volume 1, pages 525–531, 2001.
- [19] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65:43–72, 2005.
- [20] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.
- [21] Hans P. Morevec. Towards automatic visual obstacle avoidance. In *International Joint Conference on Artificial Intelligence*, pages 584–584, 1977.
- [22] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Computer Vision and Pattern Recognition*, pages 2161–2168, 2006.
- [23] Timo Ojala, Matti Pietikainen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996.
- [24] Soo-Chang Pei and Ji-Hwei Horng. Design of fir bilevel laplacian-of-gaussian filter. *Signal Process.*, 82(4):677–691, 2002.
- [25] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on Computer Vision*, pages 430–443, 2006.
- [26] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2010.
- [27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *International Conference on Computer Vision*, pages 2564–2571, 2011.
- [28] Frederik Schaffalitzky and Andrew Zisserman. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *European Conference on Computer Vision*, pages 414–431, 2002.

-
- [29] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.
- [30] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000.
- [31] Feng Tang, Suk Hwan Lim, N.L. Chang, and Hai Tao. A novel feature descriptor invariant to complex brightness changes. In *Computer Vision and Pattern Recognition*, pages 2631–2638, 2009.
- [32] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *Pattern Analysis and Machine Intelligence*, 32(5): 815–830, 2010.
- [33] T. Trzcinski and V. Lepetit. Efficient Discriminative Projections for Compact Binary Descriptors. In *European Conference on Computer Vision*, 2012.
- [34] Zhenhua Wang, Bin Fan, and Fuchao Wu. Local intensity order pattern for feature description. In *International Conference on Computer Vision*, pages 603–610, 2011.