A Comparative Evaluation of Well-known Feature Detectors and Descriptors[#]

Şahin Işık*[1], Kemal Özkan

Abstract: Comparison of feature detectors and descriptors and assessing their performance is very important in computer vision. In this study, we evaluate the performance of seven combination of well-known detectors and descriptors which are SIFT with SIFT, SURF with SURF, MSER with SIFT, BRISK with FREAK, BRISK with BRISK, ORB with ORB and FAST with BRIEF. The popular Oxford dataset is used in test stage. To compare the performance of each combination objectively, the effects of JPEG compression, zoom and rotation, blur, viewpoint and illumination variation have investigated in terms of precision and recall values. Upon inspecting the obtained results, it is observed that the combination of ORB with ORB and MSER with SIFT can be preferable almost in all possible situations when the precision and recall results are considered. Moreover, the speed of FAST with BRIEF is superior to others.

Keywords: Image matching, feature detector, feature descriptors, SIFT, SURF, BRIEF, FAST, BRISK, ORB, MSER.

1. Introduction

In parallel with developing technology, the number of smart devices have increased drastically. In old times, while computer can only compute four operations; division, multiplication, subtraction and addition, fortunately, in nowadays, they are used to identify the persons by looking their biological characteristics such as the color of eye, the tone of voice, fingerprint and features around the face. Moreover, in this age where the communication with internet become popular, the detection smuggle cars, plates and other stolen devices can be done only within a few hours by using a simple program. Undoubtedly, the contribution of feature detection methods and algorithms have considerable rate for implementation of such programs.

To match images that evolved by some transformations and distortions such as scale, rotation, illumination, noise and compression, researchers have proposed a lot of robust feature detection methods until 21st. To illustrate the well-known detectors and descriptors that receiving most citations are ORB, SIFT, SURF, BRISK, BRIEF, HARRIS, FAST and MSER. Despite the advantages of existing methods, but there is still a great demand for such algorithms in order to close to lacks of proposed methods. Since there is a tradeoff between robust feature detection and execution-time, yet a fastest one that yield best results in all conditions has not been developed. From the observation results of this study, it can be seen objectively. The invention of such algorithms look like to an implementation of security algorithm. On the other word, it is not possible obtaining the combination of best accuracy as well as best security and minimum computation-time at the same time. Therefore, we have to make concessions for the sake of selection an optimal feature detection method with respect to the task performed.

Fortunately, most recently some feature detection methods have been compared in several studies. In [1], the six feature descriptors have chosen to make a comparison: SURF, ORB, BRIEF, BRISK, SIFT and SU-BRISK (a variant of BRISK). Also, a comparative analysis of three binary descriptors (ORB, BRIEF and BRISK) by concentrating on well-known detectors (ORB, MSER, SIFT, SURF, FAST and BRISK) is carried out in terms of effects of various geometric and photometric transformations [2]. Also, in the study on comparison of low level feature extraction algorithms [3], the performance of FAST-SIFT (F-SIFT) feature detection methods have compared in case of blur, illumination and scale changes, rotation and affine transformations. In another study the comparison analysis between SIFT and traditional photogrammetric feature extraction methods and matching metrics in Photogrammetry [4] is carried out by performing experimental tests on images acquired by Unmanned Aerial Vehicles (UAV) and Mobile Mapping Technologies (MMT) with geometric distortions. Again, the performance of keypoint detectors (FREAK vs. SURF vs. BRISK) are examined in the context of pedestrian detection [5].

In this study, we have made a comparison between well-known detectors and descriptors. Seven combination of detectors and descriptors are included: SIFT with SIFT, SURF with SURF, MSER with SIFT, BRISK with FREAK, BRISK with BRISK, ORB with ORB and FAST with BRIEF. For evaluation, the precision and recall metrics are used by considering the relation between correct matches obtained after RANSAC with number of keypoints in the reference image that are also visible (after transformation) on the second image and number of keypoints in the reference image that have been matched. Our experiments are conducted on the popular Oxford datasets [6] which include images that have in different form: JPEG compression, zoom and rotation, blur, viewpoint and illumination. From the performance evaluation, we have clearly seen that ORB with ORB descriptors and MSER with SIFT descriptors are useful in case of all possible conditions.

In section 2, we introduce an overview of each detector and descriptor. In the following section 3, the experimented datasets, evaluation metric and performance of all methods in case of different type transformations and deformations have presented. Finally, the conclusion is given and future work is discussed in

[1] *Eskisehir Osmangazi University, Turkey*
* Corresponding Author: Email: sahini@ogu.edu.tr

last section.

2. Overview of Methods

2.1. SIFT

SIFT detector consists of four major stages: (1) Scale space extreme detection; (2) keypoint localization; (3) orientation assignment; (4) keypoint descriptor. In the first stage, the image is scanned over location and scale in order to determine potential interest points that are invariant to scale and orientation. These are the local scale-space maxima of the Difference of Gaussian (DoG) which is obtained by subtracting the different Gaussian scales. In the keypoint localization step, the insignificant points are rejected and edge response is eliminated. While the points that have a low contrast are rejected with respect to a predefined threshold, the non-edge points are eliminated based on the idea under the Harris method in which it is assumed that the distribution on an edge region should give larger eigenvalues and the distribution on a non-edge region should give small eigenvalues. For this purpose, Hessian matrix was used to compute the principal curvatures and eliminate the non-edge points. To obtain descriptors that invariant to rotations, an orientation histogram was formed from the gradient orientations of each local maximum of the DoG function within a region around the keypoint. The final stage of SIFT constructs a feature vector by considering the direction of a keypoint which is gradient strength is maximal. Typically, an adjacent 16x16 region is determined by put the keypoint in the center. After the region is chosen, SIFT divides this region into 4×4 sub-regions with 8 orientation bins in each. Since there are 4 x 4 = 16 histograms each with 8 bins the vector has 128 elements. Thus, the meaningful descriptors are extracted from the image that are compact, highly distinctive and yet robust to change in illumination and camera viewpoint, [7, 8].

2.2. SURF

Due to the large amount of data in a pattern recognition task (e.g. Face Recognition) and the time consumption of SIFT is significantly high, Herbert Bay have proposed the SURF [9] detector inspiring by the SIFT descriptor. It is able to generate scale and rotation invariant interest points and descriptors. SURF have been used as a feature selector in many studies because of the some reasons such as descriptors generated by SURF are invariant to rotation and scaling changes and computational time of SURF is small and fast in compare to other feature extraction algorithms in case of interest point localization and matching. Systematically, SURF uses 2-D Haar wavelet and integral images. For keypoint detection, it uses the sum of the 2D Haar wavelet response around the point of interest. A 2D Haar wavelet is obtained by an integer approximation to the determinant of Hessian matrix that extracts blob-like structures at locations where the determinant is maximum. Therefore, the performance of SURF can be attributed to non-maximal-suppression of the determinants of the hessian matrices. In description phase, firstly the neighborhood region of each keypoint is divided into a number of 4x4 sub-square regions. Then, it computes the response of a 2D Haar wavelet response each sub-region. Again, this procedure can be computed with aid of the integral image. Each response contributes four values to a descriptor, so each keypoint is described with a 64-dimensional (4x4x4) feature description of all sub-regions. Although the SURF method runs faster than the SIFT, but in some situations like viewpoint and intensity change it does not give good results as SIFT produced.
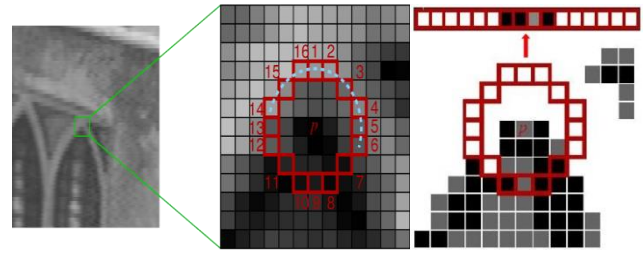

Figure 1. (a) A processed interest point and 16 pixels surrounding on it, (b) the demonstration of storing 16 values surrounding pixels in a vector form.

2.3. FAST

FAST corner detector is partly based on the SUSAN (Smallest Univalue Segment Assimilating Nucleus) corner criterion [10, 11]. Similar to the SUSAN, FAST corner detector uses a circle of 16 pixels (this is the Bresenham circle of radius 3) to classify whether a candidate point p is actually a corner or not. As plotted in Fig. 1 (a), assume the processed pixel p with intensity I_P is selected. Each pixel in the circle is labeled from integer number 1 to 16 as clockwise. To make the algorithm fast, first compare the intensity of pixels 1, 5, 9 and 13 of the circle with I_P. If at least three of these four pixels satisfies the threshold criterion so that p is chosen as an interest point. On the other hand, if at least three of the four pixel values (I1, I5, I9 and I13) are not above or below $I_P + T$, then P is not an interest point (corner). Else if at least three of the pixels are above or below $I_P + T$, then check for all 16 pixels and in this case 12 contiguous pixels should fall in given criterion. Likewise, repeat the procedure for the all others remaining pixels in the image. Because of the some limitations such as for n<12 the algorithm does not work well, the choice of pixels is not optimal and multiple features are detected adjacent to one another, a machine learning approach has been employed to the algorithm to deal with these issues. In this case, a training set is constructed as for every feature point "p", store the 16 pixels around it as a vector, as demonstrated in Fig. 1 (b). Each pixel in these 16 pixels can have one of the following three states: darker, similar and brighter. Depending on the rule in given below, the feature vector V is divided into 3 subsets, P_S (similar points), P_D (darker points) and P_B (brighter points). Then, the ID3 (a decision tree classifier) is performed to select the point which yields the most information about whether the candidate pixel is a corner with respect to an entropy minimization criteria. So, the first problem is achieved with aid of a classification algorithm. Also, the second problem, called multiple features are detected adjacent to one another, can be dealt with by applying non maximal suppression after detecting the candidate corner points. This is done by obtaining the sum of the absolute difference between the pixels in the contiguous arc and the center pixel, then the values of two adjacent interest points are compared and the lower one is discarded. Noting that Fig. 1 (a-b) is taken from website in [12].

2.4. BRISK

Although the local features obtained by vector-based descriptors such as SURF, SIFT and similar methods gives successful results in terms of an image representation while being invariant to many transformations, such as scale, rotation and viewpoint changes, however, using the descriptors of them is not an efficient way, especially for machines with a scarce amount of resources in terms of computation power and mobile wireless devices which has a limited uplink bandwidth channel and low power requirements. To address this challenge, several binary

descriptors computed directly on image patches and BRISK is one of them. BRISK is based on FAST detector. In general, BRISK [13] consists of three parts: a sampling pattern, orientation compensation and sampling pairs. In here, taking a sampling pattern around the keypoint refers to points spread on a set of concentric circles, which are used to determine a point is whether a corner or not in FAST detector. Then these pairs are separated two subsets, short-distance pairs and long-distance pairs. To achieve rotation invariance, the direction of each keypoint is determined by taking the sum of computed local gradient between long-distance pairs and short-distance pairs are rotated based on obtained orientations. Finally for all the pairs, the intensity values of the first and second points in the pair are compared, i.e., if the value of first point is larger than the second then output is "1", else "0". Hence, after going all 512 pairs, leading to a descriptor with 512 bits in length. In matching case, the Hamming distance is used instead of Euclidean distance. BRISK detector uses Hamming distance instead of Euclidean distance due to its short execution time. To achieve this, only the sum of XOR operation between two binary descriptors is sufficient to compare them.

2.5. MSER

An MSER [14-16] region is a set of all connected pixels above all thresholds and also virtually unchanged over a range threshold. In the other words, the selected regions are unchanged shapes where local binarization is stable over a large range of thresholds. According to the paper [14], the MSER detection is similar to a watershedding process that can be described in the way that the gray-scale image is represented by function $I : \Omega \rightarrow [0...255]$ where $\Omega = [1...W] \times [1...H]$ is the set of all image coordinates. They select and an intensity threshold and divide the set of pixels into two groups B (black) and W (white). It is observed that the cardinality of the two sets changes with respect to the changing the threshold from maximum (255) to minimum (0) intensity. The area of each connected component is stored as a function. Among the extremal regions, the "maximally stable" ones are chosen by analyzing this function for each potential region to find ones that maintain its state with similar function value over multiple thresholds. The selected "maximally stable" regions are called MSER regions that have changed in size only a little across at least several intensity threshold levels.

2.6. ORB

ORB detector [17] (Oriented FAST rotated BRISK) is a combination of FAST and BRISK. To extract keypoints, it modifies the FAST detector as scale invariant by constructing a scale pyramid of the image. At each scale, keypoints are detected by illustrating the FAST detector. Once the keypoints detected, the Harris corner measure is employed to sort them and only top N points are chosen based on a threshold. To obtain rotation invariant features, first-order moments is used to compute the local orientation through an intensity centroid which refers to the weighted averaging of pixel magnitudes in the local patch. BRIEF descriptors further are computed on rotated patch and keeps the binary string as ORB descriptor

2.7. FREAK

FREAK [18] is also a binary descriptor and borrows the procedures of sampling pattern and pair selection from BRISK. It uses a circular pattern where the density of points exponentially drops when moving away from the center and called as retinal sampling grid that inspired by the retinal pattern in the eye. To provide rotation invariance property, an orientation for the

selected patch is computed by summing the local gradients over chosen pairs which are symmetric to each other with when center is considered as base. Also, in descriptor creation stage, a similar approach that was used in ORB is performed, simply the less correlated pattern is selected. Generally, the 512 binary tests are used in order to obtain maximum performance.

3. Experimental Results

3.1. Dataset

Although there are a vast of datasets to evaluate the performance of feature detectors and descriptors , we have preferred the well-known dataset that proposed by [19]. The test dataset consists from 8 classes; *bark, bikes, boat, graf, leuven, trees, ubc* and *wall*. Each dataset includes 6 images and 5 homography matrices. So, we have used the given homography matrices to match keypoints. The reason to why we select this database can be attributed to include some general deformations such as rotation and zoom, image blur, illumination (light) changes, viewpoint changes or JPEG compression, which have applied to each dataset in order to assess the performance of detectors and descriptors as benchmark. The dataset is available at the website of [6] and images existing on the dataset are presented in Fig. 2. In some figures, img2-6 shows the images that exposure to variation with respect to the original image (img1).

3.2. Evaluation Metrics

To evaluate the performance of each method, the precision and recall values of each image that situated at different dataset are computed based on metric introduced by Mikolajczyk et al in [19]. According to the our survey on comparison of feature detection and extraction methods, we have observed generally the number of matching keypoints, repeatability, correspondences, recall, efficiency, duration, speed, average distance and similar metrics have been used to determine the success of each method. However, in our opinion the speed, precision and recall values are enough for a midlevel comparison. For a classification task, while the precision shows the number true positives (i.e., the number of items correctly assigned into the positive class) divided by total number items that predicted as positive (i.e., the total number of true positives and false positives), the recall shows the number of true positives divided by the total items that already labelled as positive (the sum of true positive and false negatives).

In this study, recall represents the ratio of correct matched descriptors to the number of correspondences between two images. The highest recall value shows the better performance of feature detection method and indicates the sensitivity of method. The recall value is obtained with the following formula:

$$recall = \frac{\#correct\ matches}{\#correspondences}$$

(1)

The number of correct matches refers to the matched feature after running Random Sample Consensus (RANSAC) algorithm which is employed to eliminate inconsistent matches by selecting inliers and rejecting outliers. Also, the crosscheck filter is performed on descriptors prior to RANSAC. On the other hand, the number correspondences shows the number of keypoints in the reference image (original image) that are also visible (after transformation) on the query image (exposure to the deformation). In this work, the correspondence between two images are obtained by using Brute-Force and Flann-Based matchers with hamming and l2-norm distances. For SIFT and SURF descriptors, Flann-Based matcher with l2 norm distance is run, whereas for binary string
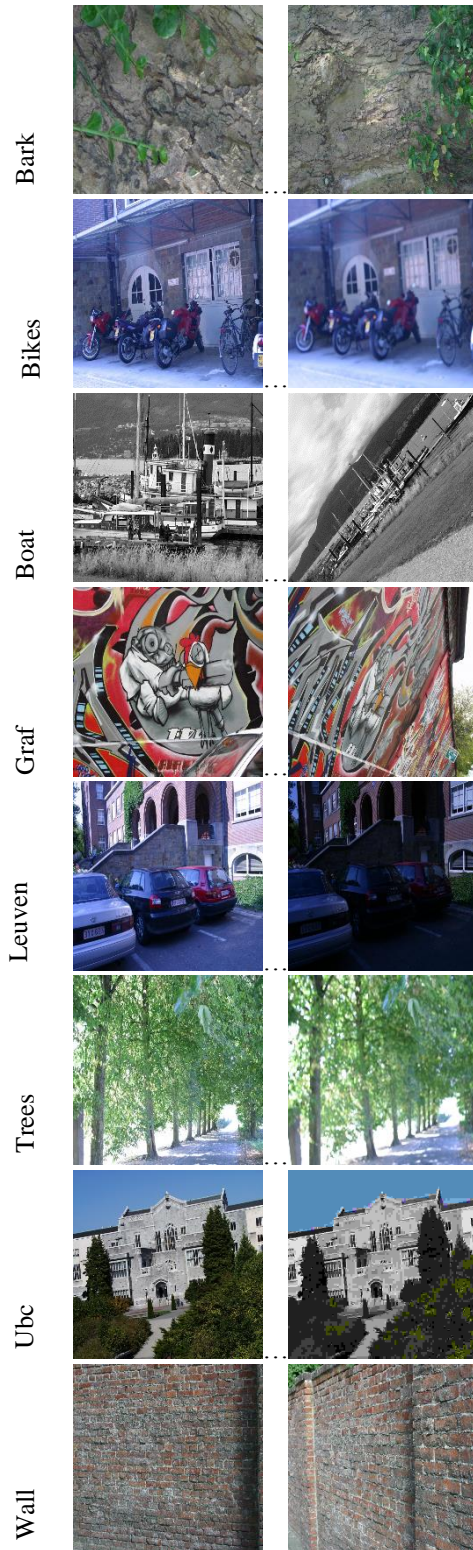
Figure 2: The existing images at Oxford dataset.

(Left column labels top to bottom: Bark, Bikes, Boat, Graf, Leuven, Trees, Ubc, Wall)

based descriptors such as ORB, BRISK and FREAK, the Brute-Force matcher with hamming distance is illustrated only for BRIEF descriptors the Brute–Force matcher with l2-norm is carried out. Since the MSER and FAST methods are only keypoint detectors, the SIFT and BRIEF descriptors are computed in case of MSER and FAST, respectively.

Additionally, the precision refers to the ratio of correct matched descriptors to the number of total positively matched descriptors which are number of keypoints in the reference image that have been matched. The highest precision value means that how

relevant the matched features to each other. The precision value is computed as:

$$precision = \frac{\#correct\ matches}{\#total\ positive\ matches}$$

(2)

Besides, the value of precision and recall varies with respect to the strictness of matching criteria and complexity of data. Therefore, the matching criteria is determined as balance as possible as mentioned above.

To discuss the experimental results of evaluation, precision, recall and execution-time are presented in Figure 3-7 and Table 1, respectively.

3.3. Effects of Zoom and Rotation Variation

(a) Precision Bark   (b) Precision Boat
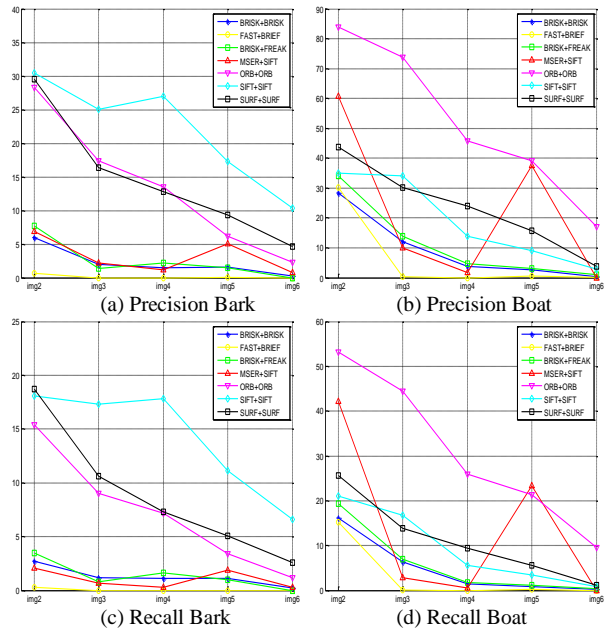
(c) Recall Bark   (d) Recall Boat

Figure 3: Precision and recall values for zoom and rotations.

To compare the performance of given methods in case zoom and rotation changes on *bark* and *boat* images, the obtained precision and recall values are exhibited in Fig. 3 (a-d). Images have rotated around its optical axes in the range from 30 to 45 degrees and zoom process has carried out by rescaling image with a factor of four. At a first glance, we can see that ORB, SIFT and SURF descriptors exhibit competitive results. Although in case of *bark* images, the SIFT outperforms the ORB, but for *boat* images the performance of ORB is better than the SIFT. Basically, the precision and recall values of ORB and SIFT are superior and other descriptors exhibit similar results. The ORB is best one whereas the FAST with BRIEF descriptors is the worst one.

3.4. Effects of Blur Variation

The second set of experiments is conducted on *bikes* and *trees* images. The different scales have obtained by blur the images with the radius in the range 2-2.5. Although images are blurred with an increasing level, *bikes* survives the main structures of objects, compared to *trees*. Comparing with *bikes*, more keypoints are caught in the evaluation on *trees*. From the Fig. 4 (a-d), it seen that the performance of ORB is better than the MSER with SIFT descriptors and FAST with BRIEF descriptors on *bikes* images in terms of precision and recall. Also for *trees* MSER with SIFT and FAST with BRIEF descriptors gives good results even if the blur level is increased. Interestingly, although SIFT descriptors are invariant to scale, rotation and illumination

changes, but it is not able to survive it's kingdom in high level blur conditions. Again other descriptors present similar performance.
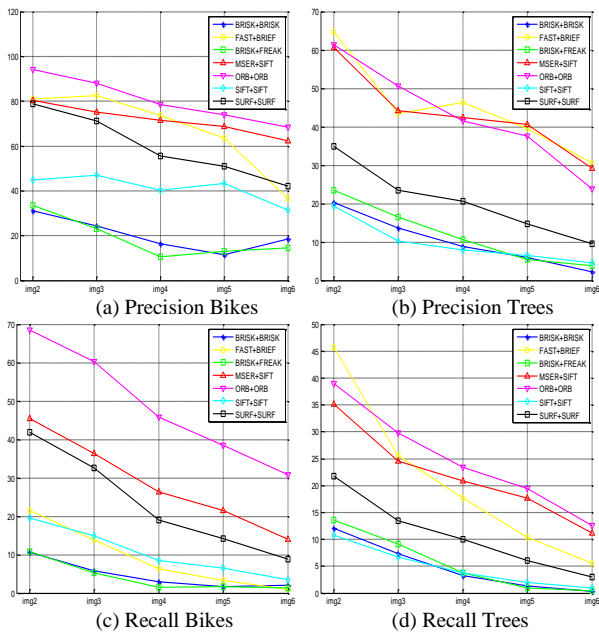

(a) Precision Bikes   (b) Precision Trees

(c) Recall Bikes   (d) Recall Trees

Figure 4: Precision and recall values for blur variation.

3.5. Effects of Viewpoint Variation

In the third experiment, the performances of detectors and descriptors are analyzed by concentrating on viewpoints change datasets which are *wall* and *graffiti* (*graf*).
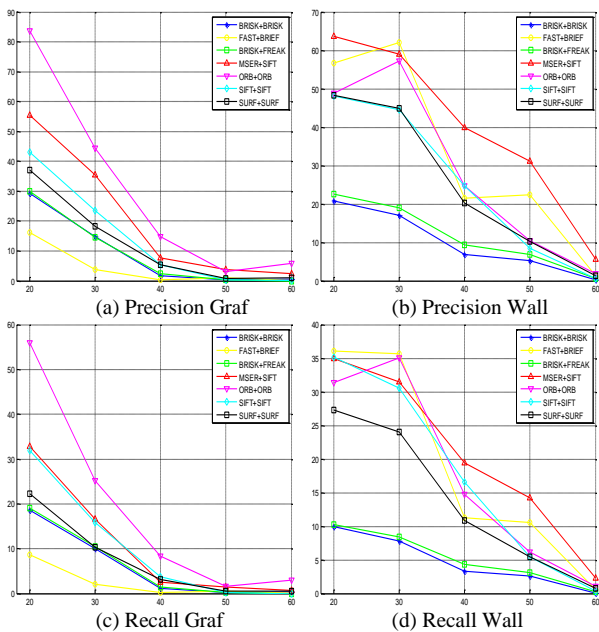

(a) Precision Graf   (b) Precision Wall

(c) Recall Graf   (d) Recall Wall

Figure 5: Precision and recall values for viewpoint variation.

To construct viewpoint change datasets, the orientation of camera have changed from a fronto-parallel view to a position with significant foreshortening at a range from 20 to 60 degrees. The contribution of graffiti and wall datasets is useful for investigating the performance of detectors and descriptors under affine invariance conditions since they contains structured scenes with distinctive edge boundaries. As demonstrated in Fig. 5 (a-d), the performances of MSER with SIFT, ORB, SIFT and FAST with BRIEF descriptors come to the forefront by considering the

precision and recall values when compared to other methods. While the ORB gives satisfactory precision and recall results in case *graffiti* images, but for *wall* images MSER with SIFT descriptors are more dominant. Again the results of SIFT, FAST with BRIEF descriptors and BRISK are mixed into each other.

3.6. Effects of Illumination Variation

To analyze the performance of methods under increasing level of illumination, an experiment is conducted on *leuven* dataset. For this purpose, the brightness of images has changed by varying the camera aperture. Fig. 6 (a-b) summarize the performance of methods at different level of brightness. By observing the results, we can say that the combination of FAST with BRIEF present superior results, compared with others when both precision and recall values are taken into account. With increasing darker conditions, the performance of SIFT and BRISK go worse than others. This is causing from the characteristics of descriptors obtained from SIFT and BRISK. Clearly, it is demonstrated that the degradation of performance under increasing illumination is not similar in terms of precision and recall for all detectors.
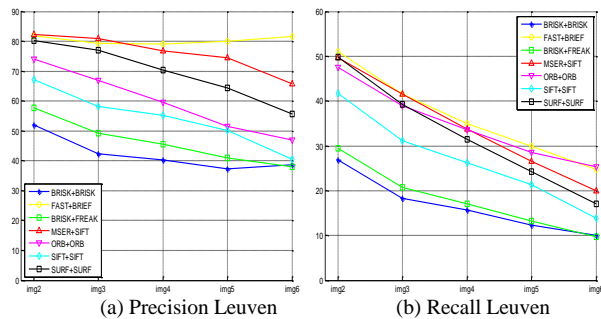

(a) Precision Leuven   (b) Recall Leuven

Figure 6: Precision and recall values for illumination variation.

3.7. Effects of JPEG Compression

In this experiment, impacts of compression is examined in terms of comparison of results obtained from each method conducting on ubc dataset. For this purpose, the artifacts have introduced in Joint Photographic Experts Group (JPEG) compression by using a standard xv image browser with the image quality parameter changing from 40% to 2%.
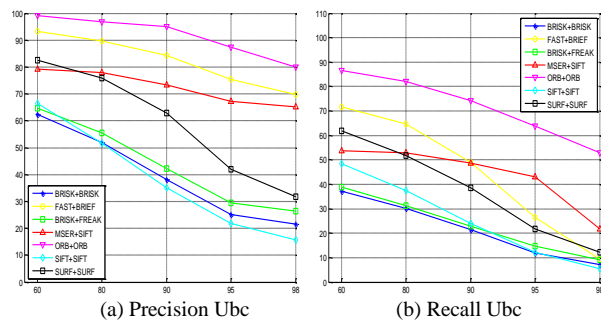

(a) Precision Ubc   (b) Recall Ubc

Figure 7: Precision and recall values for JPEG compression.

As shown in all figures of ubc dataset, the compression rate is given in the x-axis as 60 to 98. The obtained results are given in Fig. 7 (a-b). Obviously, the ORB is best one even in case of high level artifacts situated on images. Another glaring point is performance of detectors and descriptors is better than in the case of illumination, blur and viewpoint variation in terms of both precision and recall, but worse than in case of zoom and rotation of structured type images (*buildings*) having large homogeneous regions in terms only precision. When the performance of all descriptors are sorted in terms of precision and recall FAST with

BRIEF descriptors is ranked in number two through the level of compression 60 to 98.

3.8. Comparison of Computation-Time

To compare the execution time of given methods, the average time per keypoint is obtained and exhibited in Table 1. Noting that for all experiments, a software implemented on opencv 2.4.8 is worked on a computer that have 3.4 GHz and 4 GB RAM, with Windows 8 as an operating system.

Table 1: Comparison of execution time per each keypoint only for original image in bark dataset.

Method	Keypoints	Time(ms)	Time/Point
SIFT+SIFT	3665	5.989	0.0016341
SURF+SURF	3634	1.083	0.0002977
MSER+SIFT	323	0.889	0.0027533
BRISK+FREAK	466	2.531	0.0054322
BRISK+BRISK	466	0.235	0.0005046
ORB+ORB	500	0.236	0.0004715
FAST+BRIEF	11880	0.083	0.0000070

For computation the execution time, only time for the keypoint extraction is considered. Obviously, it is seen that MSER with SIFT and BRISK with FREAK are taking more execution time than the others. On the other hand, the FAST with BRIEF is fastest one and ORB is faster than BRISK. In the following table, the last column refers to the execution time in milliseconds (ms) per each keypoint. However, the evaluation should be performed on all datasets for a fair comparison, but since the time is limit, we can only give these results.

4. Conclusions and Future Work

From the quality measures, we have concluded that for all changes using ORB is admissible. Also for blur variation ORB, MSER with SIFT or FAST with BRIEF descriptors are best ones. Additionally, for viewpoint variation a one can be chosen among MSER with SIFT and ORB can be used. In case of illumination variation, again FAST with BRIEF descriptors is useful. Moreover, if we want to use a feature detector that insensitive to JPEG compression, the ORB is can be chosen. Finally, when considering the speed, FAST with BRIEF is again the best one among seven combinations.

We want to emphasize that each method is useful for different task, but a good one should be fast and at the same time robust to deformations with respect to a min error criteria. Also, there are some unrealized ideas in terms of performance comparison with different metrics such as type-1 and type-2 error, F-measure and average number of obtained descriptors per image etc. remains as a future work. In fact, different matching and filtering techniques are greatly needed for a fair comparison. However, we believe that this comparison is sufficient to investigate which method is fastest and robust to some possible forms that may occur.

References

[1] Peng Z. Efficient matching of robust features for embedded SLAM, 2012.

[2] Heinly J. Dunn E., and Frahm J.-M. Comparative evaluation of binary features, Computer Vision–ECCV, Springer, 2012, pp. 759-773.

[3] El-gayar M. and Soliman H. A comparative study of image low level feature extraction algorithms, Egyptian Informatics Journal, Vol.14, Number 2, 2013, pp. 175-181.

[4] Lingua A. Marenchino D. and Nex F. Performance analysis of the SIFT operator for automatic feature extraction and matching in photogrammetric applications, Sensors, Vol. 9, Number 5, 2009, pp. 3745-3766.

[5] Schaeffer C. A comparison of keypoint descriptors in the context of pedestrian detection: freak vs. surf vs. brisk.

[6] Oxford Dataset, robots.ox.ac.uk/~vgg/data/data-aff.html

[7] Lowe, D. G. Object recognition from local scale-invariant features. Computer vision, Vol. 2, 1999, pp. 1150-1157.

[8] Lowe D.G. Distinctive image features from scale-invariant keypoints, International journal of computer vision, Vol. 60, Number 2, 2004, pp. 91-110.

[9] Bay H. Ess A. Tuytelaars T. and Van Gool L. Speeded-up robust features (SURF), Computer vision and image understanding, Vol. 110, Number 3, 2008, pp. 346-359.

[10] Rosten E. and Drummond T. Machine learning for high-speed corner detection, Computer Vision (ECCV 2006), Springer, 2006, pp. 430-443

[11] Rosten E. Porter R. and Drummond, T. Faster and better: A machine learning approach to corner detection, Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol. 32, Number 1, 2010, pp. 105-119.

[12] http://www.edwardrosten.com/work/fast.html

[13] Leutenegger S. Chli M. and Siegwart R. Y. BRISK: Binary robust invariant scalable keypoints. Computer Vision (ICCV), IEEE, 2011, pp. 2548-2555.

[14] Matas, J., Chum, O., Urban, M., and Pajdla, T. Robust wide-baseline stereo from maximally stable extremal regions, Image and vision computing, Vol. 22, Number 10, 2004, pp. 761-767.

[15] Nistér D. and Stewénius H. Linear time maximally stable extremal regions: 'Computer Vision (ECCV 2008), Springer, 2008, pp. 183-196.

[16] Obdržálek D. Basovník S. Mach L. and Mikulík A. Detecting scene elements using maximally stable colour regions, Research and Education (Robotics-EUROBOT 2009), Springer, 2010, pp. 107-115.

[17] Rublee E. Rabaud V. Konolige K. and Bradski G. ORB: an efficient alternative to SIFT or SURF. (Computer Vision (ICCV)), IEEE, 2011, pp. 2564-2571.

[18] Alahi A. Ortiz R. and Vandergheynst P. Freak: Fast retina keypoint. Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, p. 510-517.

[19] Mikolajczyk K. and Schmid C. A performance evaluation of local descriptors, Pattern Analysis and Machine Intelligence, 2005, Vol. 27, Number 10, pp. 1615-1630.