Evaluation of Interest Point Detectors and Feature Descriptors for Visual SLAM

Alexander Prozorov, Andrew Priorov, Vladimir Khryashchev
P.G. Demidov Yaroslavl State University
Yaroslavl, Russia
alexprozoroff@gmail.com, {andcat, vhr}@yandex.ru

***Abstract*—In this paper we present comparison of feature detectors and descriptors in the visual feature-based simultaneous localization and mapping (SLAM) context. Feature extraction concept is widely used in computer vision and image processing algorithms which are SLAM, panorama stitching, object detection, etc. Since features are used as the starting point and main primitives for subsequent algorithms, the overall algorithm will often only be as good as its feature detector. Identifying of detected features is provided with the help of descriptor that distinguish it from the rest features. In turn, descriptor should provide invariance when finding the matches between the specific points relative to the image transformation. In this study, we evaluate the performance of well-known detectors and descriptors under the effects of JPEG compression, zoom and rotation, blur, viewpoint and illumination variation. Performance parameters of the descriptors have investigated in terms of precision and recall values.**

I. Introduction

The Simultaneous Localization and Mapping (SLAM) problem has recently received a large attention. The SLAM technique consists in a mobile robot moving in an unknown environment, which attempts to estimate its own position and to realize a spatial map. The environment is described through a set of natural landmarks extracted by the robot from the surrounding during movement. The answer to the question of what type of sensor must be used depends on the application type and boundary conditions. These are the precision in estimating the trajectory and mapping, lighting changes and the geometry of surrounding space. The SLAM problem can be solved by using laser rangefinders, digital cameras (visible, infrared), sonars, etc.

During the movement in an unknown space, the robot must continually solve two independent problems. Such as computing motion trajectory and space mapping, which are not known a priori. It is also necessary to have the possibility of subsequent positioning within a resulting map. The type of calculated map is entirely dependent on the characteristics of sensors and the conditions of the surrounding area. Most often, despite this fact, the problem reduces to finding specific areas of space. These features can be stably identified within the stream of data coming from the sensors.

Recent years have seen a significant increase the interest in the complex SLAM algorithms based on the use of digital cameras as the main sensor [1]. The main reason for this is a significant growth of computing power and reduce the cost of image processing devices. The practical application of many algorithms goes to real time. Restrictions on the resolution of the original images is weakening. Although the use of high-resolution images are still associated with certain difficulties. Nevertheless, low cost of digital cameras in comparison with other SLAM sensors allows them to find a wide range of applications.

This paper presents a comparison of different methods of detection and description of key image features. The comparison is made on the basis of the requirements of visual SLAM algorithm. It uses an unscented Kalman filter to process key points. The data stream comes from a RGBD camera, which provides data on the depth of the observed space along with standard RGB images. All these conditions make it possible to identify the most suitable types of interest point detectors and feature descriptor for the given task.

II. Local features

Feature detection concept is widely used in computer vision and image processing algorithms. It aim at computing abstractions of image information and making local decisions at every image point. The resulting features will be subsets of the image areas, often in the form of isolated points, continuous curves or connected regions. The process of tracking interest points is going with the use of various description techniques. Identifying of interest point is provided with the help of descriptor that distinguish it from the rest points. In turn, descriptor should provide invariance when finding the matches between the specific points relative to the image transformation. Detected features usually used in further tracking or matching (for SLAM, panorama stitching, object detection, etc..).

In 2008 Tuytelaars and Mikolajczyk [2] defined the list of properties of good feature points:

- *Repeatability*: Given two images of the same object or scene, taken under different viewing conditions, a high percentage of the features detected on the scene part visible in both images should be found in both images.
- *Distinctiveness*: The intensity patterns underlying the detected features should show a lot of variation, such that features can be distinguished and matched.
- *Locality*: The features should be local, so as to reduce the probability of occlusion and to allow simple model approximations of the geometric and photometric deformations between two images taken under different

viewing conditions (e.g., based on a local planarity assumption).

- *Quantity*: The number of detected features should be sufficiently large, such that a reasonable number of features are detected even on small objects. However, the optimal number of features depends on the application. Ideally, the number of detected features should be adaptable over a large range by a simple and intuitive threshold. The density of features should reflect the information content of the image to provide a compact image representation.
- *Accuracy*: The detected features should be accurately localized, both in image location, as with respect to scale and possibly shape.
- *Efficiency*: Preferably, the detection of features in a new image should allow for time-critical applications.

Repeatability is the most important property of all. It is required in all application scenarios and it directly depends on the other properties like invariance, robustness, quantity etc. Depending on the application increasing or decreasing them may result in higher repeatability.

Local invariant features not only allow to find correspondences in spite of large changes in viewing conditions, occlusions, and image clutter, but also yield an interesting description of the image content and object or scene recognition tasks. Table I lists existing feature-based computer vision systems along with the algorithms that are used in each of them [3].

TABLE I. MODERN FEATURE-BASED COMPUTER VISION SYSTEMS

Reference/ Objective	Detector	Descriptor	Matching
Klein and Murray (2007) / SLAM	FAST	patch (8×8), warped	SSD, M-estimator
Engelhard et al. (2011) / SLAM	SURF	SURF	UKF
Wagner et al. (2010) / Panorama creation	FAST	patch (8×8), warped	NCC, M-estimator
Bleser and Stricker (2008) / tracking	FAST	patch, warped	SSD, Kalman filter
Carrera et al. (2007) / tracking	Harris	SURF	UKF
Lee and Höllerer (2008) / tracking	DoG	Optical flow, SIFT	RANSAC, Kalman filter
Rolo. (2013) / SLAM	GFTT	SURF	RANSAC

As seen from the compilation in Table I, different interest point detectors and feature descriptors have been used in variety of computer vision systems. Wherever explicit timings are available, they indicate that a significant part of the overall processing time is spent on feature detection, description and matching. These observations and the lack of independent comparisons in the context of real-time visual tracking motivate the evaluations in this work.

III. Feature extractors

The term feature extractors is used to define the combination of interest point detector and feature descriptor. This section provides a short characteristic of popular modern approaches to detect and describe image features to match them further.

Harris and Shi-Tomasi (GFTT - Good Features to Track) algorithms [4] detect image features based on the autocorrelation analysis. For each point on the image the structure tensor is computed. The structure tensor is the base for the computation of the corer score functions for both above mentioned algorithms. The local maximum of the corner score function with values greater than arbitrary threshold are marked as image features.

FAST (Features from Accelerated Segment Test) [5] is an algorithm for feature detection only. In this method a pixel is considered as a feature candidate when a continuous segment of n pixels which are either all darker or all brighter than the central pixel by more than the threshold value. The order in which pixels are tested is was determined using machine learning to achieve high processing speed. The candidates are further refined by using an additional corner score function and applying the non-maximum suppression.

SIFT (Scale Invariant Feature Transform) [6] is one of the most popular robust invariant feature detector and descriptor. The features are found by applying the difference of Gaussian filter at multiple scales and performing a scale space non-maximum suppression with additional filtering to reject line-like features. The location of the point in the image is interpolated. The final step is the orientation assignment based on the histogram of gradients in the neighborhood into 16 4×4 sub regions. In the next step the 8-bin orientation histograms are computed for each sub region. The partial histograms are finally concatenated to form the final 128-element feature descriptor.

SURF (Speeded Up Robust Features) algorithm [6] is a multiscale image feature detector and descriptor as well. The detection step in SURF takes advantage of the use of Haar wavelet approximation of the blob detector based on the Hessian determinant. The approximations of Haar wavelets can be efficiently computed using integral images, regardless of the scale. Accurate localization of multiscale SURF features requires interpolation process. The descriptor is also based on the Haar wavelets and encodes the distribution of pixel intensity values in the neighborhood of the detected feature. Computation of the descriptor begins with the assignment of the dominant orientation to make the descriptor rotation invariant. A square window with a side length of 20s (s being the feature scale) is placed on the feature point and oriented as it was computed in the previous step. The window is divided into 4×4 regular square sub regions that are then divided further to 5×5 uniformly distributed sample points. For each sample point, Haar wavelet responses for two principal directions are computed. Each sub region contributes to the descriptor with four components total of 64 descriptor elements.

STAR keypoint detector [7] is a derivative of the CenSurE (Center Surround Extremas) feature detector. It was developed as a multiscale detector with full spatial resolution. The detector uses a bi-level approximation of the Laplacian of

Gaussians (LoG) filter. The shape of the detector mask is aimed to minimize the directional bias to preserve rotational invariance while enabling the use of integral images for efficient computation. Scale-space is constructed without interpolation, by applying masks of different size.

MSER (Maximally Stable Extremal Regions) [8] is one of the many methods available for blob detection within images. The algorithm identifies contiguous sets of pixels whose outer boundary pixel intensities are higher than the inner boundary pixel intensities by a given threshold. Such regions are said to be maximally stable if they do not change much over a varying amount of intensities. MSER is also robust to blur and scale transformations.

BRIEF (Binary Robust Independent Elementary Features) descriptor [9] uses binary strings for feature description and subsequent matching. The use of Hamming distance as similarity measure leads to very fast computation. BRIEF method is sensitive to noise, therefore an averaging filter is applied to the image before descriptor computation. The value of each bit contributing to the descriptor corresponds to the result of a comparison between the intensity values of two points inside an image segment centered on the currently described feature.

ORB [6] (Oriented FAST and Rotated BRIEF) is one of the most modern algorithm for feature detection and description that combines and extends on the concepts of FAST detector and BRIEF descriptor. The feature detection is performed on multiple scales using FAST and augmented the with orientation data to achieve rotation invariance. Similarly to BRIEF, a binary descriptor is used, but the coordinates of the point pairs for binary tests around the described feature are rotated by the feature orientation angle. The random sampling used to select the point pairs in BRIEF has been replaced with a sampling scheme that uses machine learning for de-correlating BRIEF features under rotational invariance. This makes the nearest neighbor search during matching less error-prone.

The BRISK descriptor [10] is different from the BRIEF and ORB descriptors, by having a hand-crafted sampling pattern. The BRISK descriptor point-pair sampling shape is symmetric and circular, composed of 60 total points in 4 concentric rings. The sampling regions increase in size with distance from the center, and also proportional to the distance between sample points. Within the sampling regions, Gaussian smoothing is applied to the pixels and a local gradient is calculated over the smoothed region. Like other local binary descriptors, BRISK compares pairs of points, which are specified in two groups: (1) *long segments*, which are used together with the region gradients to determine angle and direction, (2) *short segments*, which can be pair-wise compared and composed into the 512-bit binary descriptor vector.

IV. Evaluation metrics

The most frequently noticed measure for performance characterization of local feature detectors is repeatability. Repeatability rate is defined as the ratio of the number of points repeated in the overlapping region of two images to the total number of detected points [11]. An interest point is considered repeated if its projection in the other image using planar homography lies within a neighborhood of size ε of an interest point detected in the other image.

Since feature detectors identify interest points at different scales, measuring the distance between these points detected at different scales, to decide whether they are repeatable or not, may lead to inaccurate results. That also considers the overlap of scale-dependent regions centered in the interest points. Repeatability is conventionally defined as:

$$repeatability = \frac{\#\,repeated\ features}{min(features_{img1}, features_{img2})} \tag{1}$$

Despite being popular, it has some limitations and does not guarantee high performance as well [12]. Also repeatability rate partially reflects the effect of various geometric and photometric transformations as it considers the minimum number of interest points detected in either of the two images.

The goal of feature descriptors comparison is to find the method that makes it possible to maximize the correct match rate while minimizing the incorrect match rate. We measure performance using recall and precision values. Intuitively, precision measures the percentage of correct matches out of all matches returned by the feature matching algorithm.

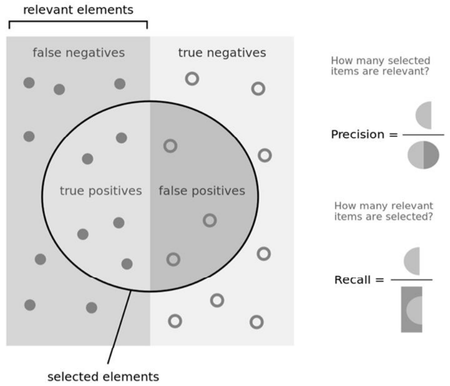

Fig. 1. Precision and recall explanation [13]

Also recall measures the approximate percentage of correct matches out of the total possible matches that exist (Fig. 1). Specifically, a correct match is a true positive, an incorrect match is a false positive, and the total number of positives in the data set is taken as the total number of possible matches. For a set of images, the numbers of correct, incorrect, and possible matches are summed over each pair of images in sequence.

$$recall = \frac{\#\,correct\ \ matches}{\#\,total\ \ possible\ \ matches} \tag{2}$$

$$1 - precision = \frac{\#\,incorrect\ \ matches}{\#\,correct\ +\ \#\,incorrect} \tag{3}$$

Since we are unable to determine if an arbitrary point in an image has a corresponding point in a second image, we approximate the number of possible matches as the minimum number of points in either image. Recall/precision curves are produced by changing matching parameters to achieve varying performance. The relevant parameters are: maximum distance ratio for nearest neighbor and correlation coefficient for normalized cross-correlation.

Plotting recall against 1−precision allows for the comparison of features extractors. Distance between curves gives an indication of feature extractors' relative performance in different regions. Optimal performance corresponds to the upper-left corner of a recall/1−precision plot, with perfect recall and zero 1−precision that corresponds to perfect precision. It is also worth noting that feature extractor performance becomes useless when precision drops below 50%, as similar results could achieved by randomly matching features. Correctness of matches is of primary importance when evaluating performance. However, we must also be concerned with the recall rate at which correct matches are produced, since perfect matching is not useful if only a few matches are made. A lower bound on recall and precision defines a rectangular region in the recall/1−precision graphs that is considered to give acceptable performance.

IV. Experimental results

Although there are a variety of datasets to evaluate the parameters of feature detectors and descriptors. In this work we use well known dataset for the feature matching evaluation [14]. The test dataset consists of 5 classes.

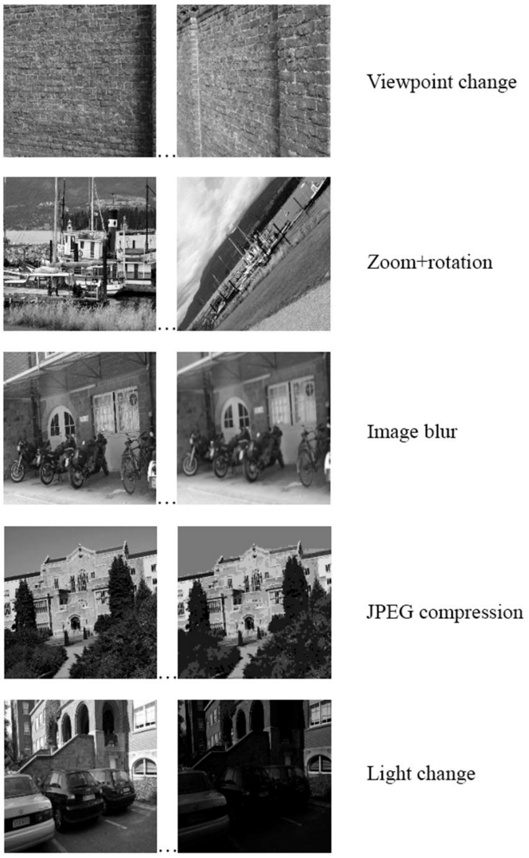

Fig. 2. Test images from Oxford dataset

Each dataset includes 6 images and homography matrices, which can be used to match keypoints. This database contains some general deformations such as rotation and zoom, image blur, light changes, viewpoint changes and JPEG compression artifacts, which have applied to each dataset in order to assess the performance of detectors and descriptors as benchmark. The dataset images are showed in Fig. 2.

Analysis of the published works in this field, as well as open source implementation of the basic algorithms for detecting local features of images allowed to choose the following types of detectors for further research: Fast, GFTT, MSER, ORB, SIFT, SURF, Star. An algorithm of simultaneous localization and mapping assumes the most important requirement for a repeatability of feature points. Requirements for the repeatability of feature points in turn depend on the number of detected values. When moving the camera the insufficiency of detected feature points distorts calculated trajectories and the final map. This may occur when the camera view does not detects sufficient number of high-textured objects which contain contrasting elements, corners and edges.

To discuss the experimental results of detectors evaluation, repeatability and correspondences count are presented in Fig. 3-12. All calculations were made using computer equipped with Intel Core i5-4210U processor, 1.7 GHz.

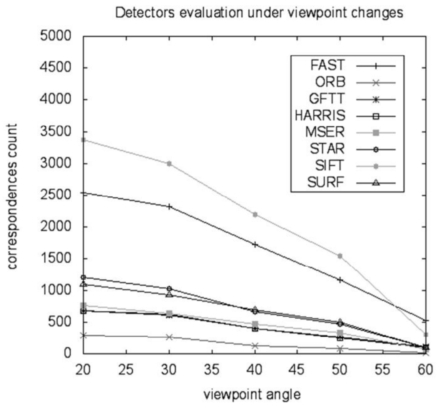

Fig. 3. Correspondences count values for viewpoint change

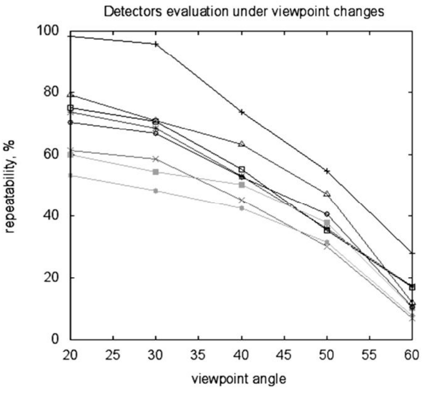

Fig. 4. Repeatability values for viewpoint change

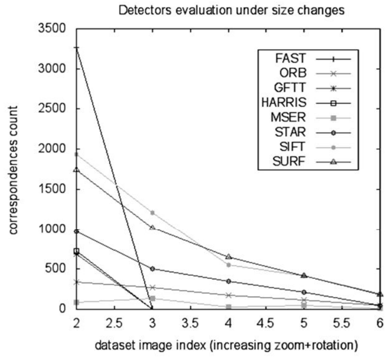

Fig. 5. Correspondences count values for zoom and rotation

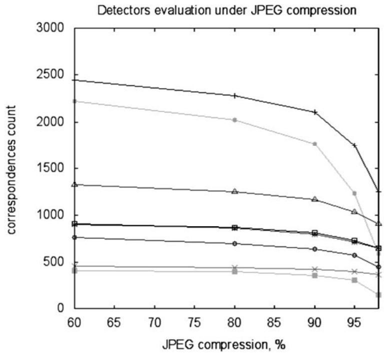

Fig. 9. Correspondences count values for JPEG compression

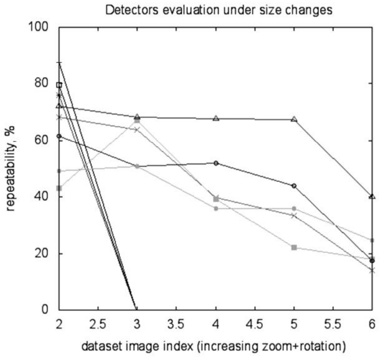

Fig. 6. Repeatability values for zoom and rotation

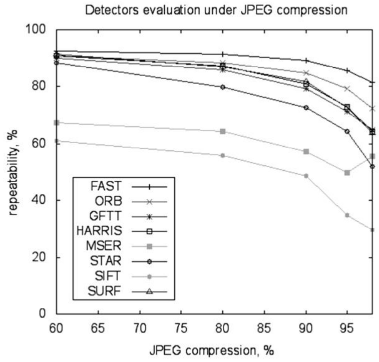

Fig. 10. Repeatability values for JPEG compression

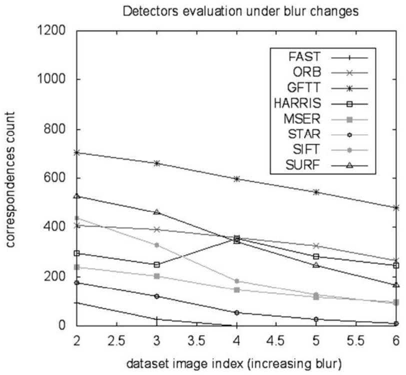

Fig. 7. Correspondences count values for Image blur

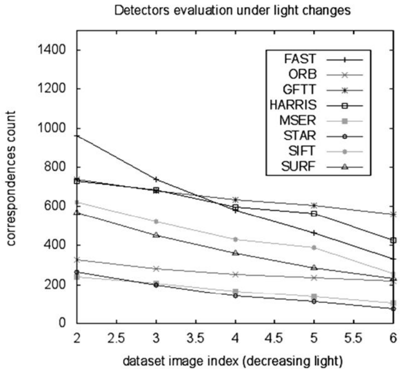

Fig. 11. Correspondences count values for light change

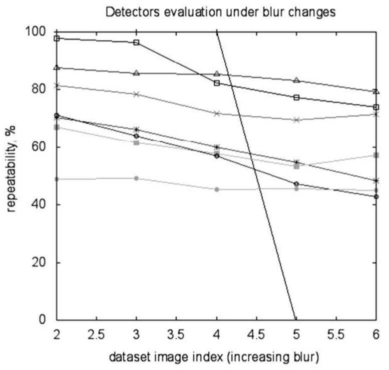

Fig. 8. Repeatability values for Image blur

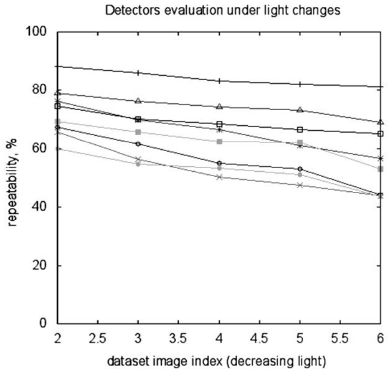

Fig. 12. Repeatability values for light change

It should also be noted that the algorithms, which use feature points detectors are often designed for use under very limited computing resources. Thus, the algorithm of simultaneous localization and mapping is widely used in mobile robotics, when autonomy and compactness of the robot significantly limit its computing capabilities. Therefore, the detector efficiency is also a critical parameter for the task. Depending on the usage scenario of the detector we can talk about the criticality of these values.

TABLE II. TIME ALLOCATION IN FEATURE POINTS TRACKING

Detector / Descriptor	Part of the total time, %			
	Detection	Descriptors extraction	Descriptors indexing	Matching
ORB/ORB	72.4	0.0	13.8	13.8
SURF/SURF	81.0	0.0	4.8	14.2
MSER/SIFT	10.0	87.1	0.4	2.5
GFTT/SIFT	33.7	38.0	2.4	25.9
Star/Brief	47.0	15.0	22.0	16.0
SIFT/SIFT	17.4	0.0	0.3	82.3
Fast/Brief	0.2	0.8	0.2	98.9

As for the task of simultaneous localization and mapping, large correspondences count value is not significant for the problem of finding a rotation matrix and transfer vector during the camera motion. Therefore, the configuration settings of the detector can be chosen based on the assumption that the two adjacent frames, obtained by camera motion, contain about 50-100 feature points, that can be stably detected [15]. Fig. 13 shows actual detection time per point for the given detectors.

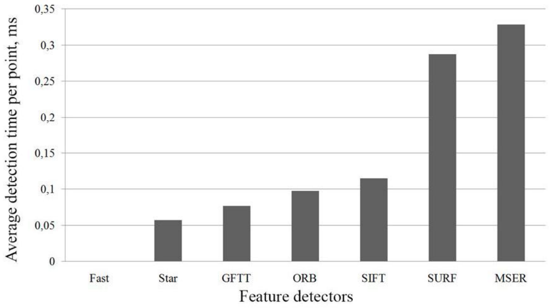

Fig. 13. Detection performance evaluation

So the performance of the detector in this task is not critical. Although, for example when detecting and tracking feature points using the descriptor SURF, detection takes about 80% of the total detection and matching time (Table II). Despite this, given detectors allow the algorithm to operate in real-time mode using hardware platforms that meet the modern level of mobile robotics devices.

For most experiments the SURF features obtain the best repeatability score. Especially, it looks better in the cases of large changes of zoom and rotation. It also good enough with increasing blur and viewpoint angle changes. SURF has great application field and it is very good for the variety of feature-based computer vision tasks. Such as visual tracking, image stitching, SLAM etc. Nevertheless, this detector is relatively demanding in terms of computing resources and also it is non free as well as SIFT detector.

Viewpoint and scale changes are the most difficult types of transformations to cope with. All detectors behave similarly under the different types of transformations with the exception of zoom and rotation case, where Fast, Harris and GFTT performs significantly worse than the others. Despite that Harris shows great results for JPEG compression and light change. Results for blur variations largely depend on the type of scene used for the experiments. In the majority of the examples ORB detector gives average results. It is good at JPEG compression and increasing blur, but it is very sensitive to illumination variations. The main advantage of this method is processing speed.

Fig. 14-18 show recall / 1-precision plots for different description techniques. All descriptors in that comparison are computed for the SURF feature points. The fastest matching results is achieved by ORB and BRIEF descriptors, which are binary features with 32 bits only.

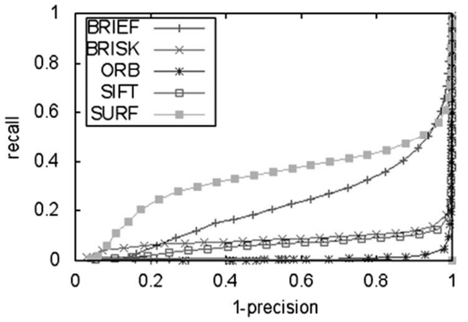

Fig. 14. Precision and recall relation for viewpoint change

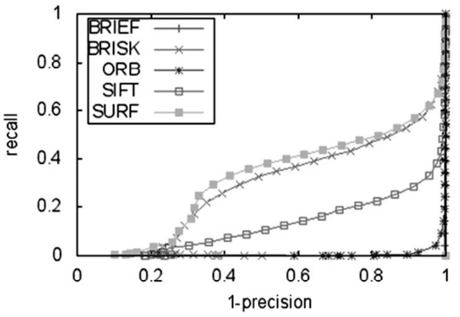

Fig. 15. Precision and recall relation for zoom + rotations

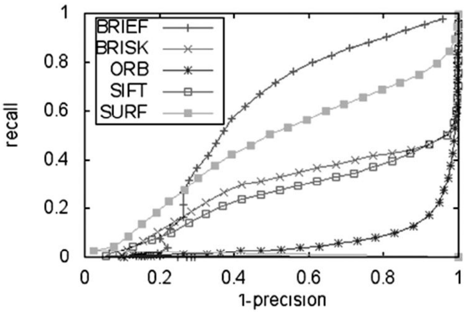

Fig. 16. Precision and recall relation for blur effect

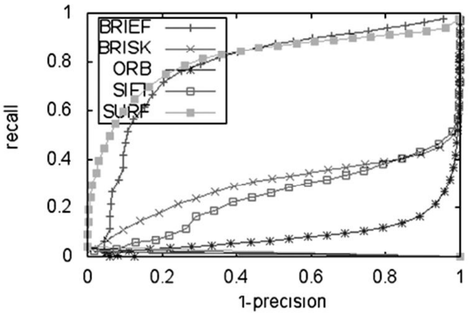

Fig. 17. Precision and recall relation for JPEG compression

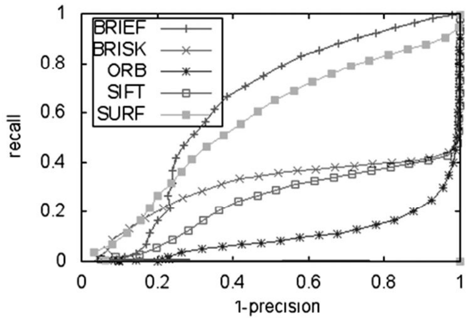

Fig. 18. Precision and recall relation for light change

In the results of scaling and rotating experiments we can see that ORB, SIFT, BRISK and SURF descriptors exhibit different results. In the case of rotated images, the SIFT outperforms the ORB, but when using alternative dataset images the performance of ORB is better than the SIFT. Basically, the precision and recall values of SURF and BRISK are superior and other descriptors exhibit worst results.

From the Fig. 16 it is seen that the performance of BRIEF is better than the SURF and BRISK descriptors in terms of precision and recall. Although SIFT descriptors are invariant to scale, rotation and illumination changes, but it is not able to show appropriate results in high level blur conditions. Affects of compression is explored in terms of comparison of results obtained from each method using compressed dataset. For this purpose, images were distorted by JPEG artifacts with the image quality parameter changing from 40% to 2%. Generally, BRIEF and SURF descriptors are best ones in the case of high level artifacts situated on images.

For analyzing the performance of methods under decreasing level of illumination, the brightness of images has changed by varying the camera aperture. Fig. 18 summarizes the performance of methods at different level of brightness. By observing the results, it can be seen that BRIEF descriptor 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 ORB go worse than others. This is causing from the characteristics of descriptors obtained from SIFT and ORB. It is showed that the decreasing of performance under illumination changes is not similar in terms of precision and recall for all detectors.

VII. Conclusion

From the quality measures, it can be concluded that for most experiments the SURF features obtain the best repeatability score. Especially, it looks better in the cases of zoom and rotation changes. Although, this detector is relatively demanding in terms of computing resources and also it is non free as well as SIFT detector. In the majority of the examples ORB detector gives average results. It is good at JPEG compression and increasing blur, but it is very sensitive to illumination variations. Among descriptors SURF and BRIEF show better results. Especially, they are good at JPEG compression as well as illumination changes. In the case of rotated and scaled images, SURF algorithm outperforms other descriptors which are very sensitive to viewpoint changes. The fastest matching results is achieved by ORB and BRIEF descriptors.

Acknowledgment

This work was supported by RFBR grants (№ 15-07-08674 and № 15-08-99639).

References

[1] A. Huletski, D. Kartashov, K. Krinkin, “Evaluation of the Modern Visual SLAM Methods”, *AINL-ISMW FRUCT Conference Proceedings*, November 2015

[2] T. Tuytelaars, K. Mikolajczyk, “Local Invariant Feature Detectors: A Survey”, *Foundations and Trends in Computer Graphics and Vision*, 3, (3), 2008, pp. 177-280

[3] S. Gauglitz, T. Höllerer, M. Turk, “Evaluation of Interest Point Detectors and Feature Descriptors for Visual Tracking”, *International Journal of Computer Vision*, vol. 4, September 2011, pp. 335--360

[4] A. Schmidt, M. Kraft, A. Kasinski: “An evaluation of image feature detectors and descriptors for robot navigation”, *LNCS*, Vol. 6375, 2010, pp. 251-259

[5] E. Rosten and T. Drummond: “Machine learning for high-speed corner detection”, *Proc. of European Conf. on Computer Vision*, 2006, pp. 430-443

[6] E. Rublee, V. Rabaud, K. Konolige, G. R. Bradski: “ORB: An efficient alternative to SIFT or SURF”, *Proc. of Int. Conv. on Computer Vision*, 2011 pp. 2564-2571

[7] M. Agrawal, K. Konolige, M.R. Blas: “CenSurE: Center surround extremas for realtime feature detection and matching”, *Proc. ECCV, LNCS* 5305, 2008, pp. 102-115

[8] D. Nister, H. Stewenius, “Linear Time Maximally Stable Extremal Regions”, *ECCV*, 2008.

[9] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, “BRIEF: Computing a Local Binary Descriptor Very Fast”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 7, 2012, pp. 1281-1298

[10] S. Krig, “Computer Vision Metrics: Survey, Taxonomy, and Analysis”, *Apress*, 2014

[11] J. Klippenstein, Z. Hong, “Quantitative Evaluation of Feature Extractors for Visual SLAM”, Computer and Robot Vision. CRV '07. Fourth Canadian Conference, 2007, pp.157-164,

[12] S. Ehsan, N. Kanwal, A. Clark, K. McDonald-Maier, “Improved repeatability measures for evaluating performance of feature detectors”, *in Electronics Letters* , vol.46, no.14, July 2010, pp.998-1000

[13] Precision and recall, https://en.wikipedia.org/wiki/Precision_and_recall

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

[15] J. Sturm, N. Engelhard, F. Endres, W. Burgard, D. Cremers, “A Benchmark for the Evaluation of RGB-D SLAM Systems”, *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, 2012