INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2018, VOL. 64, NO. 3, PP. 299-306
Manuscript received May 8, 2018; revised July, 2018. DOI: 10.24425/123522

Depth-based Descriptor for Matching Keypoints in 3D Scenes

Karol Matusiak, Piotr Skulimowski, and Pawel Strumillo

***Abstract*—Keypoint detection is a basic step in many computer vision algorithms aimed at recognition of objects, automatic navigation and analysis of biomedical images. Successful implementation of higher level image analysis tasks, however, is conditioned by reliable detection of characteristic image local regions termed keypoints. A large number of keypoint detection algorithms has been proposed and verified. In this paper we discuss the most important keypoint detection algorithms. The main part of this work is devoted to description of a keypoint detection algorithm we propose that incorporates depth information computed from stereovision cameras or other depth sensing devices. It is shown that filtering out keypoints that are context dependent, e.g. located at boundaries of objects can improve the matching performance of the keypoints which is the basis for object recognition tasks. This improvement is shown quantitatively by comparing the proposed algorithm to the widely accepted SIFT keypoint detector algorithm. Our study is motivated by a development of a system aimed at aiding the visually impaired in space perception and object identification.**

***Keywords*—Feature matching, keypoints detection, object recognition, depth map.**

I. Introduction

CONTINUOUS development of computer vision algorithms enables their wider and wider applications in industry, robotics, automation and medicine. One of the key problem in computer vision is working out robust solutions for automatic object recognition and 3D scene understanding. Such methods find applications in automatic navigation, human-computer interfaces, as well as in aids for the visually impaired. However, all high-level image data processing methods, e.g. implemented at a semantic level must be based on reliable low-level image processing techniques. These low level techniques are working on geometric or statistical features of the imaged scenes like lines, corners, simple shapes and textures, rather than on unstructured data at pixel level. For this reason, development of robust image feature detection techniques that would be size-, rotation- and noise-invariant attract continuous researcher interest. The building blocks of such techniques are keypoint detectors [1]. Keypoints may be defined as small regions of an image having some special features, e.g. rapid changes in image brightness domain, like corners. A set of detected keypoints and parameters describing them called descriptors accumulate most important features of an object and can create its representation in a compact form of a template. Such a concise representation of image characteristic regions facilitates and accelerates computation of high-level image recognition problems. One of the essential keypoint detectors' performance criteria is the ability to work robustly on images registered under various illumination conditions. Therefore, a vast number of keypoint detecting approaches has been developed [1], [10]. However, wider accessibility of stereovision, as well as other range imaging techniques have opened a new opportunity to improve robustness of keypoint detection algorithms by introducing the depth information of imaged 3D scenes. It is also worth mentioning that image processing algorithms based on depth find commercial applications in various consumer electronics and software [2], [3]. What is more, the SLAM (Simultaneous Localization and Mapping) techniques, indoor navigation and object recognition solutions commonly use distance and shape information of the object [5], [6], [7].

The ultimate objective of our study is developing robust and reliable solutions for recognition and identification of objects in 3D scenes. This functionality is inspired by a long term research of ours aimed at building electronic travel aids for the visually impaired that are capable of detecting and localizing obstacles in the surrounding environment [8], [9].

This paper is structured as follows. In section II keypoints detection and description algorithms that use RGB as well as depth information are presented. In section III we describe a novel approach to improving reliability of keypoint detection that incorporates depth information. Finally, Section IV presents results of an experimental evaluation of our method in comparison to other standard approaches.

This work was partially supported by the European Unions Horizon 2020 Research and Innovation Programme under grant agreement No 643636 Sound of Vision.

K. Matusiak, P. Skulimowski and P. Strumillo are with the Lodz University of Technology, Institute of Electronics, Lodz, Poland (e-mail: karol.matusiak@dokt.p.lodz.pl).

II. Related Works

A. RGB Keypoint Detectors

One of the most efficient keypoint detector termed the Scale-Invariant Feature Transform (SIFT) was proposed by David Lowe [11]. This algorithm offers good object recognition performance and was reviewed in a number of studies [1]. However, because of the computational complexity of the SIFT algorithm, simpler detectors were proposed like: SURF, ORB, BRISK algorithms [12], [13], [15]. None of these algorithms, however, including the SIFT, yields unmistaken and stable keypoint detection results. There are situations in which the same physical object registered at different conditions (illumination, observation point) generates different keypoints while using the same algorithm. This problem, has a detrimental impact on the quality of feature detection and matching algorithms [1], [10]. An interesting recent trend in keypoint detection techniques is the use of 3D information given as disparity or depth maps. These additional images offer new modality that can result in a better performance of depth based keypoint detectors [14], [7].

The SIFT algorithm detects features at different scales by applying a cascade of scale-space filters to the image under analysis. Such filters are implemented by combining Laplacian and Gaussian image filtering. First, the source image is decimated to obtain images at a given number of scales. Then, the obtained collection of images undergoes Gaussian filtering with different variances σ^2. The resulting set of images is called a Gaussian pyramid. Further, adjacent image scales are subtracted and the so called Difference of Gaussian (DoG) images are computed. Key feature points are the points for which the local extrema across the neighboring DoG images consistently exists. Such candidates are further selected and only the so called strong characteristic points are left that fulfill conditions of high contrast and large gradient values along horizontal and vertical directions. The first condition is verified by defining a threshold $|D(x)|$ and comparing it to pixel values taken from the Gaussian pyramid. The second condition is computed for Hessian matrix H for each tested pixel by using the formula:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r} \tag{1}$$

where:
$Tr(H)$ - Hessian trace,
$Det(H)$ - Hessian determinant,
r - edge threshold (adjustable algorithm parameter).

As a result, the algorithm delivers a set of features that are invariant to light conditions and object scaling. Finally, a descriptor is calculated for each of the detected keypoint. The first step of this procedure is to calculate the image gradient magnitudes and orientations in the neighborhood of each keypoint. Those values are used to form an orientation histogram, where the highest peak in the histogram is the main orientation of key points descriptor. To achieve invariance to rotation changes, the histogram values are shifted by an angle which is corresponding to the main orientation of the keypoint that was calculated in the previous step. Recognition of an object is performed with the nearest neighbour calculation between descriptors of the template image and descriptors calculated for the compared scene image.

B. Depth-based Keypoint Detectors

With the development of depth sensors many researchers proposed keypoint detection and description procedures that utilize this additional image data modality. Obtained depth images of 3D scenes are frequently termed depth maps and if combined with RBG images are referred to as RGB-D images. Normal Aligned Radial Feature (NARF) [14] is a feature detection algorithm that is based on the depth map information only. The first step of the NARF algorithm is calculation of surface changes for each point of the depth image and determination of the dominant direction of those changes. For the keypoint candidates a non-maxium suppression is applied to filter out clustered set of features. Descriptor computation procedure is divided into four steps:

- extract range image patch around the analyzed point,
- overlay a star pattern onto this patch, where each beam corresponds to a value in the descriptor, that captures pixels brightness variations under the beam changes and extract a unique orientation from the descriptor,
- extract the main orientation of the descriptor,
- shift the descriptor according to the main orientation to make it invariant to rotation (in a similar way as defined in SIFT).

However, due to the lack of RGB information the NARF detector cannot be compared to algorithms that incorporate a richer RGB-D image representation.

The authors of [7] have proposed a modification of another popular 2D detection and descriptor calculation algorithm, i.e. the BRISK [15]. The modification takes into account physical geometry of objects derived from a depth image. Similarly as in the SIFT algorithm the BRISK starts with calculation of a scale-space representation. Further, the corner detector is applied to each octave and intra-octave. The strength of the detector is calculated for all image points. The first extension of the BRISK incorporating depth information, in comparison to standard BRISK algorithm, is calculation of the local polar parametrization for each keypoint, i.e. radial and angular coordinates of each pixel are computed. The detection process begins with calculating a map of geodesic distances from a given point to other points. This step is computed for each detected keypoint and performed by applying a fast matching algorithm in the depth map.

The Binary Robust Appearance and Normals Descriptor (BRAND) algorithm focuses on descriptor calculation that is based on the RGB-D data [20]. For the feature detection step the authors use RGB-only based procedures like the ones implemented in the SURF or SIFT. However, scale estimation is based on the depth information and is not calculated directly by the algorithm. In the next step the main orientation of a keypoint is estimated by using the Haar wavelet responses in x and y direction [12]. The size (support) of the wavelet depends on the depth data. Exact descriptor calculation is performed for image regions, that are centered around each keypoint. In the next step $n = 256$ point pairs are selected inside the processed image region. Localization of those points is given by an isotropic Gaussian distribution. For points from each pair there are two functions calculated: the characteristic gradient changes in the keypoint neighborhood and geometric pattern description on its surface. The second function is based on the relation between the normal displacement and the surface's convexity. The described multilevel image processing procedure builds a descriptor for each keypoint found in the image and can be then used in a standard descriptor matching algorithm like kNN or brute force search methods.

The very recent keypoint detection and description algorithm termed Perspective-Invariant Feature Transform (PIFT) that works in the RGB-domain was proposed in [19]. To detect keypoints, the multi-scale FAST algorithm is used, that was originally used in the ORB and BRISK detectors. For each detected keypoint, an RGB image region is selected that is postprocessed to achieve perspective projection invariance. First, a background data is subtracted from the image by a segmentation method based on the depth data. Next, Singular Value Decomposition (SVD) method is used to estimate the normal vector of the tangent plane and all the pixels in the feature region are then projected onto the tangent plane. Finally, an image region is obtained by interpolation of the missing data and the region size is normalized. The next step of the procedure is filtering out ”false positive” keypoints, that are considered to be localized on image edges. Elimination of those points is performed by verification of the surface normals, the surface curvatures and the intensities of local image regions. For such a selected subset of keypoints and their image regions, color information descriptors are built. These descriptors are defined by applying a sampling mask according to the main orientation that was calculated for the region. The sampling mask selects HSV color information from a subset of pixels inside an image region. Finally, all image regions are normalized to the same size. As a result the created descriptors are perspective, scale and rotation invariant. However, the HSV-based color descriptor is not completely invariant to illumination changes that affect color representation in an image.

III. The Proposed Method for Matching Keypoints

A. Overview

In the proposed method we introduce the depth map data into standard keypoint detection and matching algorithms that are based on RGB images. The procedure enables detection and localization of the predefined object model in the analyzed scene. Our analysis is based on the SIFT algorithm, but it can be applied to any keypoint detector and descriptor that provides keypoint orientation information. For both, the template and the scene images, a greyscale image and the corresponding depth map have to be recorded. The depth map is a 2D image with floating point values of depth given in meters. During the preprocessing step, a common region of interest is defined for both greyscale and depth images. Selection of the mentioned region is based on the field of view of the camera and the depth sensor (and how they are aligned to each other). For the conducted experiments the following hardware was used:

- the ZED camera for delivering RGB images [2],
- the Structure Sensor for delivering depth images [4].

The cameras were mounted rigidly to each another. The Structure Sensor is a kind of active infrared light depth sensor for mobile applications. The Structure Sensor delivers depth map featuring spatial resolution of 640x480 pixels and a depth range of 0.4m-3.5m.

B. Detection and Selection of Keypoints in RGB-D Images

First, the SIFT keypoint detection and description algorithm is applied to the template RGB image (see Fig. 1a). The next step is to perform the Canny edge detection on the template depth map. A specific implementation of the Canny edge detector was employed that utilizes floating point values. An example depth map and edges detected in this map are shown in Fig. 1b. We assume that the position of the keypoint in relation to object edges can be used to improve the performance of keypoint detection and matching. To take this relation into account the keypoints that are located on the object boundaries are discarded from further calculations (the distance of the keypoint from a boundary must be larger than a given heuristic distance threshold equal to 3 pixels). The result of such an edge-based elimination of the keypoints for an example template image is shown in Fig. 1d.

C. Depth-based Descriptor

The next step is to compute, for each keypoint P_1, the distances along the rays protruding from the keypoint location along the four following directions that are determined in relation to the orientation of the keypoint as specified in the SIFT algorithm:

1) ray along keypoint orientation,
2) ray rotated by 90 degrees vs. keypoint orientation,
3) ray rotated by 180 degrees vs. keypoint orientation,
4) ray rotated by 270 degrees vs. keypoint orientation.

The rotations are clockwise. The candidate length value d is incremented iteratively and the end of the ray in point P_2 is checked whether it is positioned on a depth edge:

$$x_{P_2} = x_{P_1} + d \cdot cos(\theta) \tag{2}$$

$$y_{P_2} = y_{P_1} + d \cdot sin(\theta) \tag{3}$$

where:
x_{P_1}, y_{P_1} - x and y position of analyzed keypoint P_1,
x_{P_2}, y_{P_2} - x and y position of analyzed point P_2, that is verified to contain depth edge,
θ - angle of keypoint orientation P_1 [rad],
d - distance of tested ray between P_1 and P_2.

The calculations are performed for each ray separately. All coordinate values are calculated with a sub-pixel precision. However, the binary map of depth edges is stored with a pixel precision. A pixel value is equal zero if there is no depth edge at a given point, and equal to one otherwise. Therefore, a sub-pixel bilinear interpolation is used, while loading coordinate of point P_2. Visualization of this process with a line drawn between the keypoint and the nearest edge position P_2 is shown in Fig. 2. For the case for which the computed ray reaches image boundary without intersecting with the depth edge, the given ray is tagged as undefined and will not be further used in descriptor matching. The final step of the algorithm for creating the object model is storing, for each of the remaining keypoints, the depth-based descriptor that is an 8-element matrix consisting of the two following values for each ray:

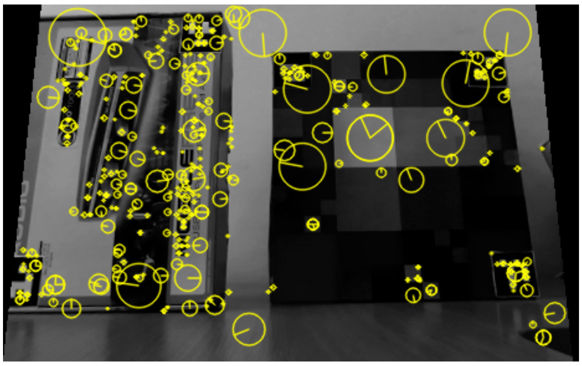
- depth value of keypoint position Z in meters,
- distance to the nearest depth edge d in pixels (explicitly -1 value represents not found edge intersection).

D. Depth Descriptor Matching Procedure

At the beginning of this procedure, we apply the object model generation algorithm to the scene image. As a result, we obtain the set of keypoints with their SIFT descriptors along with additional depth information as proposed by our method. Next, by a nearest neighbor search, pairs of the most similar SIFT keypoint descriptors between the template and the compared scene are identified. The similarity of each descriptor pair is defined according to the Euclidean distance metrics. All keypoints' matches for which the corresponding distance is larger than the minimum distance multiplied by 3 are discarded from processing. In the next step, the keypoints matches are filtered out with the use of depth and edge distance information. The following matching criterion is verified for all the keypoints and adjacent rays (relative to the main orientation of each descriptor):

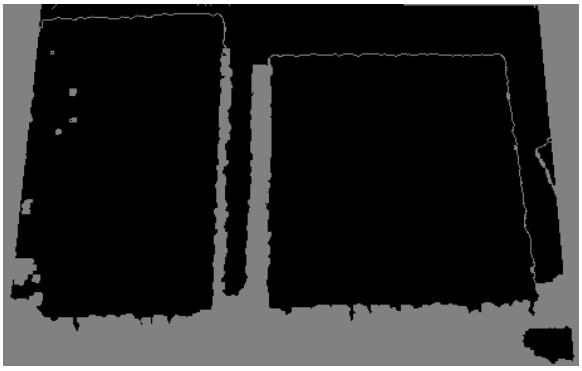
$$|d_t - d_s \cdot Z_s / Z_t| < \epsilon \tag{4}$$

where:
d_t - distance from keypoint to the nearest depth edge in the

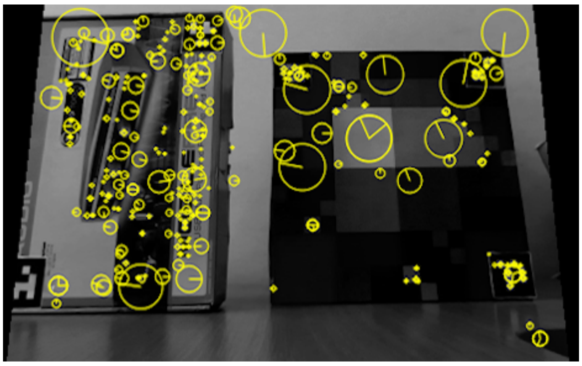


(a) SIFT keypoints

(b) example depth map



(c) edges detected in the depth map



(d) filtered SIFT keypoints

Fig. 1: Result of SIFT keypoint detection in a greyscale image (a), the corresponding depth map (b), detection result of edges in the depth map (c) and the set of SIFT keypoints after filtering out points located on depth map edges (d)

template image [px],
d_s - distance from keypoint to the nearest depth edge in the scene image [px],
Z_t - depth of keypoint in the template image [m],
Z_s - depth of keypoint in the scene image [m].

During the experiment, the optimum ϵ value was estimated at a level of 10% of d_t given in pixels. If the corresponding difference in edge distances for a pair of rays is smaller than ϵ, rays similarity counter is incremented. The counter is not incremented if the condition is not met or at least one ray from the pair is not defined. If rays similarity counter achieves a value of 50% of the total number of rays per keypoint, the keypoint's match is ranked as a good match and a bad match otherwise. So obtained set of keypoint matches is considered as the final result and can be used in further image processing tasks like object pose estimation, object tracking etc.

IV. Experimental Results

The evaluation of the proposed method was performed by applying keypoints matches for the object detection algorithm. The matches used in this evaluation were the output of standard SIFT keypoint matching procedure as well as the proposed depth based improvement.

A. Description of the Experimental Setup

Acquisition of input data was done with the use of a calibrated RGB camera and the Structure Sensor. Input RGB images with adjacent depth maps were transformed to an image representation, in which each pixel denotes values from both sources. In other words, common field of view for both devices was defined and used as a region of interest for further calculations. The result of this procedure is shown in Fig. 3.

For the evaluation purpose, 3 template and 49 test pairs were acquired for true positive tests, as well as 48 test pairs for false positive tests. The pair is defined in this paper as a pair of RGB and depth images. The test pairs represent different registration conditions of objects presented in the template pairs.

B. Evaluation Procedure

The evaluation procedure for determining keypoint matching algorithm's quality is based on the comparison of matched keypoints localization in both the template and scene images. To compare position of keypoints in reliable way we need to transform them into common coordinate systems. First, we select ground truth positions of objects in images with the use of the implemented tool. An example of the selected ground truth position of an object is shown in Fig. 4. The user of the application selects main contour points of the object in a proper pair. In the next step, object's corners ground truth positions in the template and scene images are used for homograph matrix calculation. Affine transformation of keypoints matched in the scene image into the test image coordinate system enables us to measure the quality of keypoints matches. For each pair of the matched keypoints, the Euclidean distance is calculated between keypoint position in

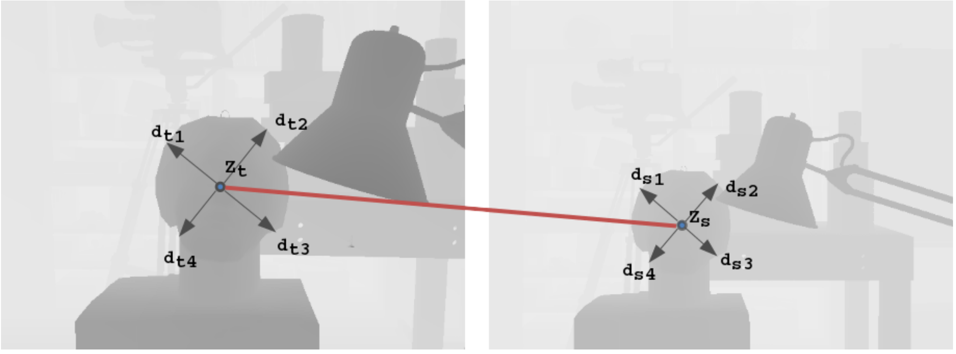
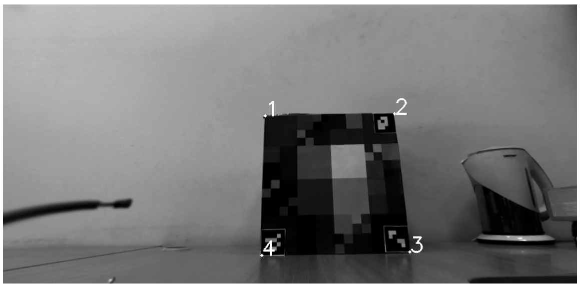


Fig. 2: Comparison of distances d_t, d_s to the nearest edges for all rays determined for keypoints in the template image and scene images correspondingly. Disparity maps come from dataset [17]

(a) Source RGB image

(b) ROI selected in the source image

Fig. 3: Result of ROI selection in the RGB image determining a common field of view for both imaging sensors



(a) Object position in the template image

(b) Object position in the scene image

Fig. 4: An example image showing the selected ground truth positions of an object with corner markers.

the test image and the transformed position from the scene image. If this distance is smaller than a given threshold ϵ in pixels, then keypoints' match is considered as an inlier. Other matches are named outliers and are regarded as false positive results of the previous algorithm step. The quality of keypoint matches is determined by the ratio of the number of inliers to the sum of the number of inliers and outliers for each detection test case. Additionally, keypoints matches are used for object detection by using a RANSAC-based homograph estimation (it is a different process than the previously mentioned). Example keypoints matches and the estimated object pose are shown in Fig. 5.

C. *Keypoint Matching Evaluation Results*

Each test pair was evaluated twice for two ϵ values, i.e. for 3 and 5 pixels as thresholds of a localization difference. Radar charts in Figure 6 and 7 illustrate the matching quality ratio of each test pair (marked with an ID number) for both the SIFT detector and the proposed method (denoted as DBFD - Depth-Based Feature Descriptor). For $\epsilon = 3$ the average quality ratio equals 0.66 for the SIFT and 0.74 for the DBFD whereas for $\epsilon = 5$ the average quality ratio equals 0.77 for the SIFT and 0.87 for the DBFD. The results for test pairs no. 13, 24 and 35 are worth a more detailed comment. For the test pair 13 the compared algorithms achieve poor quality ratio for $\epsilon = 3$ threshold. Increasing the threshold to 5 pixels allows for 70% of keypoints matches to be recognized as good matches for a given evaluation criteria for the DBFD. Significant reduction in the number of keypoint matches enables in this case to increase overall quality of the resulting matches. Visualization of keypoint matches and the detected object position is presented in Fig. 9. Similar issue can be noted for the SIFT algorithm for test pair 24 that is shown in 10. Significant number of false positive matches that are found even for completely different parts of objects (template object is partially hidden behind another object) decreases the quality ratio that is reduced below 0.3 for both ϵ values. In this test pair

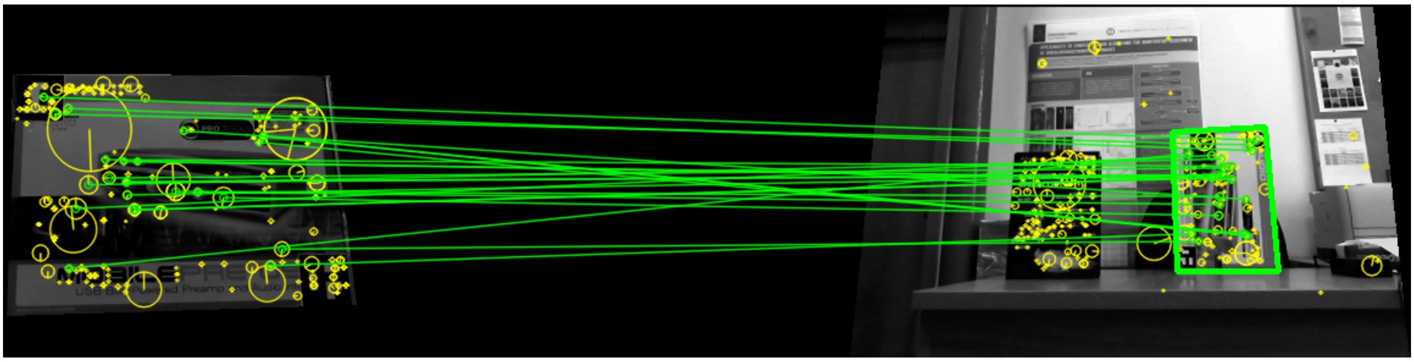


Fig. 5: Example result of keypoint matches and object detection obtained from the proposed algorithm

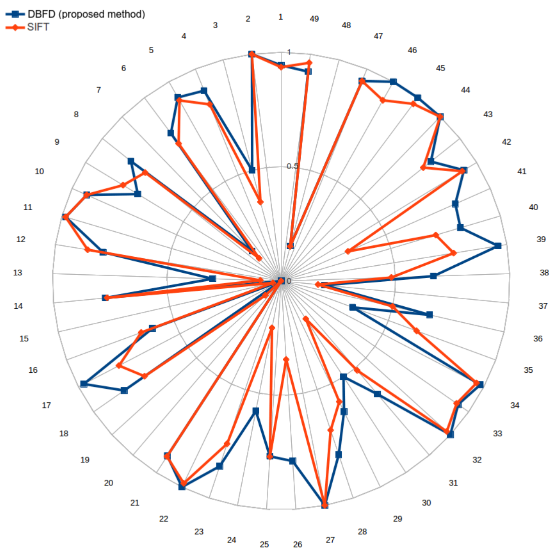


Fig. 6: Keypoints matches localization tests result for $\epsilon = 3$. Each point on the chart represents the ratio value for a test pair with a given ID

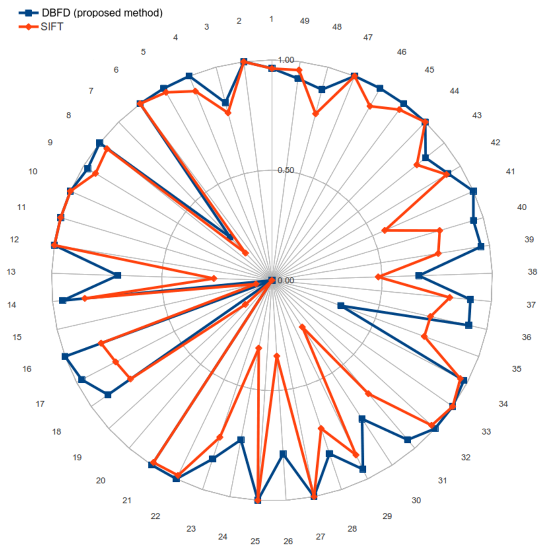


Fig. 7: Keypoints matches localization tests result for $\epsilon = 5$. Each point on the chart represents the ratio value for test a pair with a given ID

physical boundaries of the object are taken into consideration during filtering of the matches in the DBFD algorithm (see Fig. 11). Therefore, the quality ratios reached by the DBFD are 0.58 and 0.74 for $\epsilon = 3$ and $\epsilon = 5$ respectively. Test pair 35 is an example of the DBFD poor performance in comparison to the result obtained with the SIFT detector (Fig. 12). Due to the significant scale change of object representation in the scene image, the DBFD procedure filters out too many matches (even the true positive ones). This negative effect can be improved, by introducing an adaptive, distance dependent threshold of edge distances.

V. Conclusion

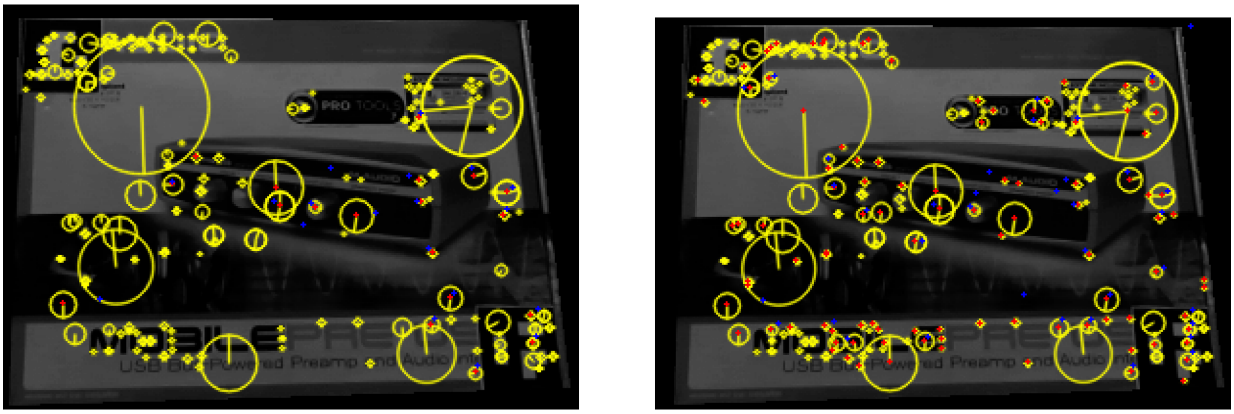
In this paper we discuss the problem of improving the matching performance of keypoints which is the basic processing step in tasks related to object recognition and identification. There is a high demand for solutions offering high matching scores. We have shown on a number of example images that our approach, we named DBFD, that is based on depth information can improve the rate of good matches. Our method is based on the assumption that keypoints detected on boundaries (termed outliers) are frequently false keypoints that are not associated with keypoints (inliers) of the object itself but rather with the image context coming form the background of the object under analysis. For the evaluated test pairs we have obtained the following improvements of the average quality ratio:

- by 0.08 for distance threshold $\epsilon = 3px$,
- by 0.10 for distance threshold $\epsilon = 5px$,

in comparison to the SIFT detector. There is, however, much room for improvement of the proposed algorithm: 1) the precision of the detected edges in the depth maps considerably influence the number of matches that are filtered out from further processing - an example approach might be to fuse the depth and RGB images for a better estimation of object edges [18], 2) we believe that prior evaluation and securing appropriate quality of 3D scene depth maps would even further enhance the method efficiency.

References

[1] T. Tuytelaars and K. Mikolajczyk, *Local Invariant Feature Detectors: A Survey*, Foundations and Trends in Computer Graphics and Vision.



(a) Comparison of keypoints localization for an example test pair of the DBFD algorithm

(b) Comparison of keypoints localization for an example test pair of the SIFT algorithm

Fig. 8: Keypoints positions results for test pair #13 and $\epsilon = 5$; original keypoints position and scale in the template image are marked with yellow color (circle size presents keypoint scale); template keypoints for which matches were found are marked with red color; matched keypoints in the scene image after reprojection to the template image coordinate system are marked with blue color

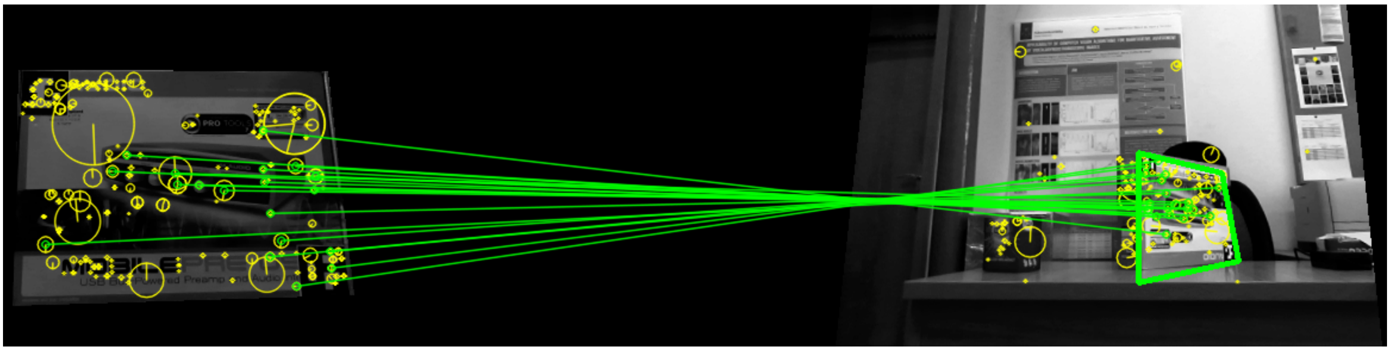


Fig. 9: Result of keypoint matches and object detection for the DBFD for test pair no 13

Volume 3, Issue 3 , pp. 177-280 (2007)
[2] ZED Stereo Camera, www.stereolabs.com, accessed 2018.03.27
[3] Google Tango Project, get.google.com/tango, accessed 2018.03.27
[4] Structure Sensor, structure.io, accessed 2018.03.27
[5] D. R. dos Santos, M. A. Basso, K. Khoshelham, E. de Oliveira, N. L. Pavan, G. Vosselman, *Mapping Indoor Spaces by Adaptive Coarse-to-Fine Registration of RGB-D Data*, IEEE Geoscience and Remote Sensing Letters, Volume 13, Issue 2 (2016)
[6] O. Wasenmller, M. Meyer, D. Stricker, *CoRBS: Comprehensive RGB-D benchmark for SLAM using Kinect v2*, IEEE Winter Conference on Applications of Computer Vision (2016)
[7] M. Karpushin, G. Valenzise and F. Dufaux, *Improving distinctivness of BRISK features using depth maps*, IEEE International Conference on Image Processing (2015)
[8] M. Bujacz, P. Skulimowski, and P. Strumillo. Naviton - a prototype mobility aid for auditory presentation of three-dimensional scenes to the visually impaired. *J. Audio Eng. Soc*, 60(9):696–708, 2012.
[9] K. Matusiak, P. Skulimowski, and P. Strumillo. Object recognition in a mobile phone application for visually impaired users. In *2013 6th International Conference on Human System Interactions (HSI)*, pages 479–484, June 2013.
[10] K. Matusiak, P. Skulimowski, P. Strumillo, *Unbiased evaluation of keypoint detectors with respect to rotation invariance*, IET Computer Vision, Volume 11, Issue 7, 507-516 (2017)
[11] D. G. Lowe, *Distinctive Image Features from Scale-Invariant Keypoints*, International Journal of Computer Vision, Volume 60, Issue 2, 91-110 (2004)
[12] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, *Speeded-Up Robust Features (SURF)*, Computer Vision and Image Understanding, Volume 110, Issue 3, 346-359 (2008)
[13] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, *ORB: an efficient alternative to SIFT or SURF*, IEEE International Conference on Computer Vision 2011
[14] B. Steder, R. Bogdan, R. Kurt and K. W. Burgard, *NARF: 3D Range Image Features for Object Recognition*, Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (2010)
[15] S. Leutenegger, M. Chliand and R. Y.Siegwart, *BRISK: Binary Robust invariant scalable keypoints*, IEEE International Conference on Computer Vision, pp. 2548-2555 (2011)
[16] E. Rosten and T. Drummond, *Machine Learning for High-Speed Corner Detection*, Computer Vision ECCV 2006, Volume 1, pp. 430-443 (2006)
[17] S. Martull, M. P. Martorell, K. Fukui, *Realistic CG Stereo Image Dataset with Ground Truth Disparity Maps*, ICPR2012 workshop TrakMark2012, pp.40-42 (2012)
[18] C. Choi, A. J. B. Trevor, H. I. Christensen, *RGB-D edge detection and edge-based registration*, 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (2013)
[19] Q. Yu, J. Liang, J. Xiao, H. Lu, Z. Zheng, *A Novel perspective invariant feature transform for RGB-D images*, Computer Vision and Image Understanding, 167, 109-120 (2018)
[20] E. R. Nascimento, G. L. Oliveira, M. F. M. Campos, A. W. Vieira *On the development of a robust, fast, lightweight keypoint descriptor*, Neurocomputing, 120, 141-155 (2013)

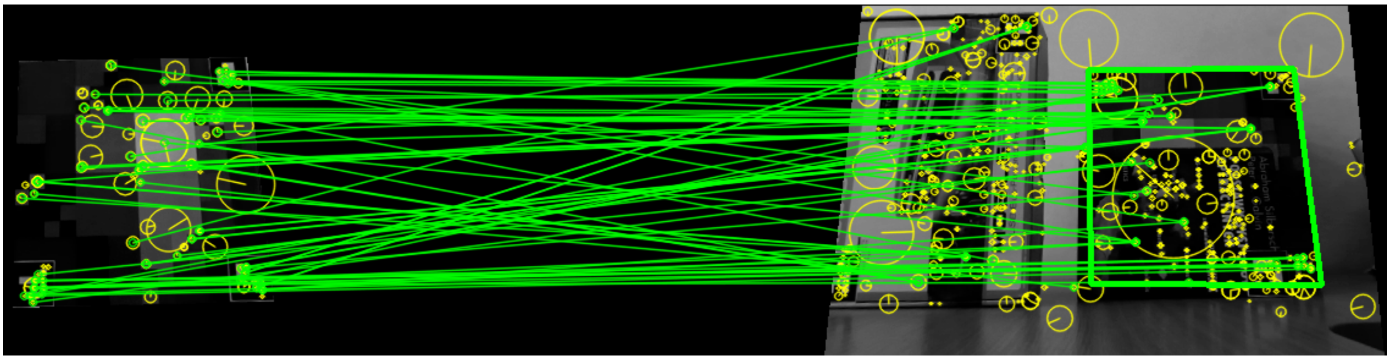


Fig. 10: Result of keypoint matches and object detection for the SIFT for test pair no 24

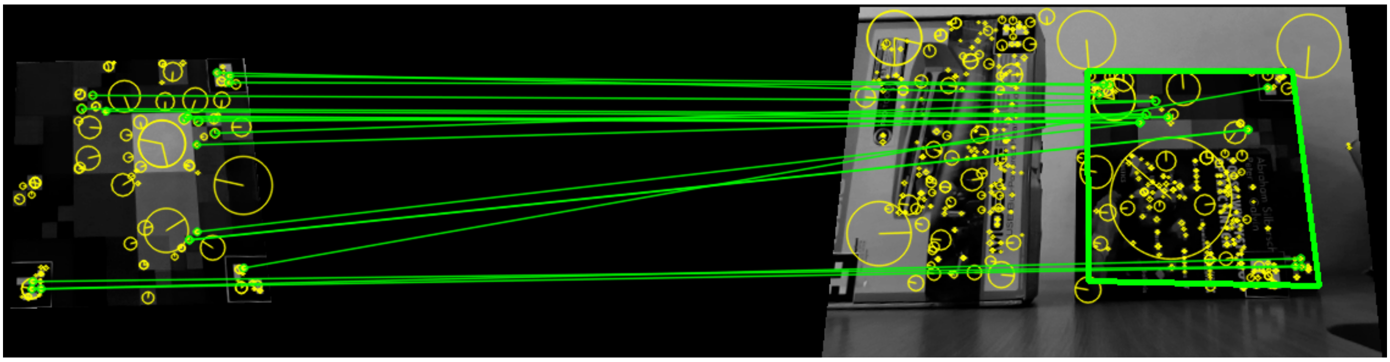


Fig. 11: Result of keypoint matches and object detection for the DBFD for test pair no 24

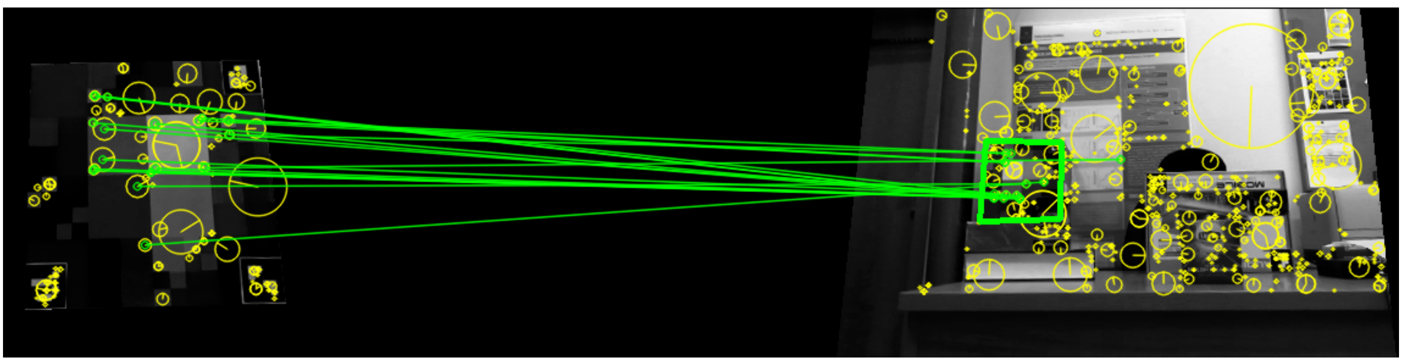


Fig. 12: Result of keypoint matches and object detection for the SIFT for test pair no 35