

PAPER • OPEN ACCESS

## Ground target localization and recognition via descriptors fusion

Recent citations

- [Volodymyr Gorokhovatskyi et al](#)

To cite this article: Mohamed M Kamel *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **610** 012015

View the [article online](#) for updates and enhancements.



**ECS** **240th ECS Meeting**  
Digital Meeting, Oct 10-14, 2021  
**We are going fully digital!**  
Attendees register for free!  
**REGISTER NOW**

# Ground target localization and recognition via descriptors fusion

Mohamed M Kamel<sup>1,3</sup>, Hussein S Taha<sup>1</sup>, Gouda I Salama<sup>2</sup> and Yehia Z Elhalwagy<sup>2</sup>

<sup>1</sup>Egyptian Armed Forces, Egypt

<sup>2</sup>Military Technical College, Egypt

<sup>3</sup> Email: [m\\_mamdouh77@hotmail.com](mailto:m_mamdouh77@hotmail.com)

**Abstract.** Keypoint matching can be defined as locating the position of a particular point in two images precisely. Recently, keypoint descriptors have taken a great effect targeting to be powerfully invariant to rotation, scale and translation for improving target detection. The detection task is carried out by a reference-scene image matching to localize the desired target in the input scene. An innovative approach is proposed in this work to fuse the state-of-the-art feature descriptors ORB, BRISK for the sake of accurate ground target detection in two phases. Firstly, off-line phase, where the fused features are extracted from different perspective, azimuth angles of the desired target to build a comprehensive reference image representation. Secondly, on-line phase, where the fused features extraction task is carried out from the whole scene. Hence, it is matched with the stored reference one to find the keypoints correspondence. The outliers' problem is eliminated using Random Sample Consensus (RANSAC) algorithm resulting in speeding up the matching procedure. The conducted comparative analysis has revealed the discriminative power of the fused features in localization and recognition tasks while keeping the proposed system works in real-time.

## 1. Introduction

Keypoint matching is considered an essential task in several applications related to computer vision, like image stitching [1], image retrieval [2] and object detection and recognition [3]. This task can be accomplished through three successive steps; firstly, detecting the keypoints locations in both reference and scene image such as blobs or corners. Secondly, extracting a proper feature to describe each detected keypoint. Finally, finding the correspondences between the keypoints of both images via a similarity measurement. Several aspects may have applied to improve the keypoint matching task but the most significant one is leveraging the keypoint description. Feature descriptors like SIFT [3], SURF [4], ORB [5], BRISK [6] and AKAZE [7] have been introduced in the previous years for this task, with each one having its faintness and robustness. Generally, the more discriminative power the descriptor has, the more computational time it needs. Fusing numerous descriptors was presented to improve the matching capability. Fusion techniques can be classified into two main types [8], [9]; the early fusion, where the fusion task is carried out before the matching task and the late fusion, where the fusion task is carried out after the matching task. This work aims to fuse the binary descriptors to enhance the keypoint matching accuracy. In the detection task, the matched points are utilized to localize the desired target in the input scene image. The contribution of this proposal is summarized as follows:



- 1- Building a comprehensive description for the reference image to overcome the lack of descriptors invariant to the wide perspective angle variation.
- 2- Improving the discriminative power of the extracted features via descriptors fusion while keeping the detection system works in real time.
- 3- Providing the approaching direction alongside the target location in the scene image.

The organization of the paper is as follows: the related work is presented in Section 2. Section 3 introduces a background brief. The proposed detection system is explained in detail in Section 4. Section 5 includes the experimental results. Finally, Section 6 presents the conclusion.

## 2. Related works

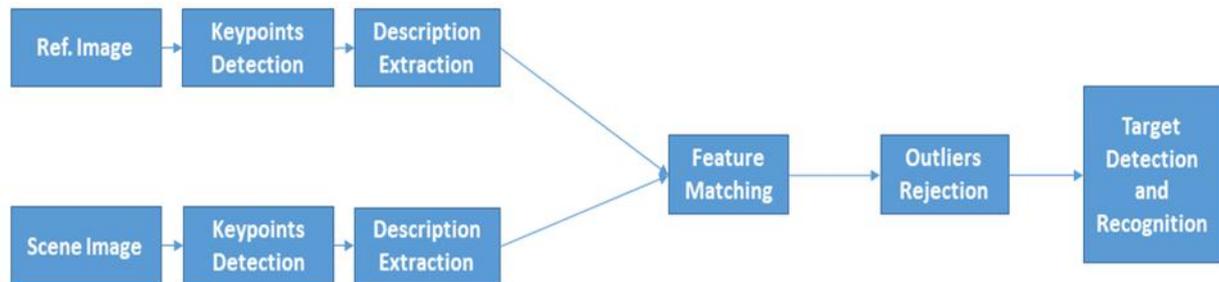
This section illustrates the related and previous work briefly in two main subsections.

### 2.1. Keypoint descriptors

In the latest few years, various keypoint descriptors were introduced. Lowe [3] proposed SIFT (Scale-Invariant Feature Transform) as a robust descriptor where the given input image is convolved with DoG functions at numerous scales to find the location of the keypoints that are scale-space extrema. Although SIFT is considered one of the most powerful descriptors as shown in [10] but its low speed has limited its utilization in many applications. Later, SURF (Speeded Up Robust Features) was proposed by Bay et al. [4] as a fast descriptor in the matching task. SURF descriptor uses the Fast-Hessian detector for the extraction phase, relying on the factor of Hessian matrix. Binary descriptors were introduced as a result of various real-time applications requesting faster feature extractors and low-memory descriptors. BRIEF (Binary Robust Independent Elementary Features) was proposed by Calonder et al. [11] as first robust binary descriptor. An image patch is binary tested and a bit string is linked to them to create the descriptor. BRISK (Binary Robust Invariant Scalable Keypoints) was proposed by Leutenegger et al. [6] depending on a round sampling pattern, as it figures various comparisons to form a binary string that is not variant to the rotation and the scale. As an improvement of the BRIEF descriptor and the FAST (Features from Accelerated Segment Test) detector, Rublee et al. [5] proposed ORB (Oriented FAST and Rotated BRIEF). Alahi et al. [12] proposed FREAK (Fast Retina Keypoint). It depends on using an algorithm to form feature descriptors by choosing a group of binary tests that feats variety among them.

### 2.2. Keypoint fusion

Significant hard work is accomplished aiming to improve keypoint descriptors. Although a specific keypoint descriptor may not be revealing for a particular dataset, it may be helpful in another situation, but its contribution can't be appraised if it is rejected. A Multiple Descriptor was proposed by He et al. in [13] that classify images via using its NN (nearest neighbors) approaching to categories and various types of given feature descriptors from input images. NN approaching to all classes were linked with various kinds of descriptors despite of applying the K-NN over a single feature descriptor only. For keypoint fusion, a Bayesian descriptors approach was proposed by Mountney et al. in [14]. Where, a set of powerful keypoint descriptors is chosen by a first training step. After that, the chosen keypoint descriptors is fused by a Naive Bayesian Network. Using a trained system that depends on the dataset assigned is considered the disadvantage of this approach. A novel approach was proposed by Bakshi et al. in [15] for recognition as to serve the IRIS people, where SIFT & SURF descriptors achieved matches are fused. The matching task is achieved individually for the two descriptors. After that, the matching task is evaluated dependent on the total matches achieved by both descriptors. A fusion of two-dimension & three-dimension features in the perspective of ground-targets detection was proposed by Perakis et. al in [16]. Firstly, ground-targets are learned from a marked dataset. Secondly, a certain template matching technique is achieved for the task of recognition. An evaluation for various fusion methods of template matching was accomplished presenting their strength and weakness points.



**Figure 1.** Typical target detection and recognition via keypoints matching

### 3. Background

Figure 1 shows typical target detection and recognition via keypoints matching. A walk through the detection system main components is introduced here.

#### 3.1. Keypoints detection

Feature detector can be defined as an algorithm that is used to detect keypoints (points of interest) in the scene-image [17]. Corners, edges and blobs are the main forms of detected features. Feature detectors are either available with their own description algorithm such as ORB, AKAZE, SIFT, KAZE, BRISK and SURF or exist individually like MSER, FAST and AGAST. As a case study, the FAST detector can be briefly explained as follows:

*3.1.1. FAST detector.* FAST is a corner detector proposed by Rosten et al. [18] for detecting the interest points in a scene-image. An interest point is a pixel which has a well-known position and can be robustly detected. FAST main idea is based on the comparison of gray values between neighborhood pixels and nucleus. If point P is a feature, then at least N contiguous pixels around P are brighter than it or darker by more than a threshold T. Although FAST detector is beatable with respect to the other detectors due to its fast performance, resulting in suitability for real time applications, it has few drawbacks such as being variant to scaling, low-memory requirements and its weak robustness to image noise [19]. As an improvement for the FAST detector, AGAST (Adaptive and Generic Accelerated Segment Test) was proposed by Elmar Mair et al. in [20] to choose a more robust method of estimating the decision tree in order to be common and not required to get adapted periodically to the new environment. AGAST detector is much faster and is not required to be adapted in the time of maintaining the similar corner response like FAST detector.

#### 3.2. Description extraction

A feature descriptor represents the locality of all interest points via a feature vector. It must be distinctive, invariant to scale, rotation and translation. An additional description about the detected features/keypoints from the detector must be provided [21]. Furthermore, it must be able to uniquely identify the corresponding feature points between two image frames to provide good feature matching accuracy. Descriptor vectors have binary or real values; a binary descriptor gives better performance in terms of time at the cost of encoding less information (e.g. BRIEF, ORB, and BRISK), while with a real values descriptor we can encode more information at the cost of increase the time required to compute it (e.g. SIFT and SURF). To figure out the difference between both types of descriptors, a comparative explanation for the ORB, BRISK, SURF and SIFT is highlighted as follows:

*3.2.1. ORB descriptor.* ORB [5] is based on the visual descriptor BRIEF [11] and the FAST [18] keypoint detector. It aims to be faster and more efficient with respect to SIFT in the detection task and deals with the BRIEF problem of being invariant to rotation. It presents a method of positioning estimation to FAST detection algorithm, thus providing independence to rotation. It is much powerful

than BRIEF as it learns an efficient subsection related to binary tests. ORB computes a local orientation through the use of an intensity centroid [22], which is a weighted averaging of pixel intensities in the local patch assumed not to be coincident with the center of the feature. Keypoints are further detected at different scales. The orientation is the vector between the feature location and the centroid. Although this might look to be unstable, it is reasonable with the only orientation assignment working in SIFT [5].

*3.2.2. BRISK descriptor.* BRISK [6] provides both rotation and scale invariance. For scale invariance, BRISK detects keypoints in a scale-space pyramid, performing non-maxima suppression and interpolation across all scales. To describe the features, BRISK turns away from the random or learned patterns of BRIEF and ORB, and instead it uses a symmetric pattern. Sample points are positioned in concentric circles surrounding the feature, with each sample point representing a Gaussian blurring of its surrounding pixels. The standard deviation of this blurring is increased with the distance from the center of the feature.

*3.2.3. SIFT descriptor.* SIFT [3] detects an amount of keypoints by searching for extrema of a Difference-of-Gaussian (DOG) function at different scales. A feature vector is extracted at each keypoint. The location of keypoints is then further refined. Using native scene-image properties, the positioning of given image is predicted to afford non-variant against rotation above a neighborhood around the point of interest. Then, a descriptor is computed for each detected point, based on local image information at the characteristic scale. The orientation of keypoints is estimated based on the local image gradient as the SIFT descriptor builds a histogram of gradient orientations of sample points in a region around the keypoint. It finds the highest orientation value and uses these orientations as the main orientation of the keypoint. The SIFT descriptor is largely invariant to scale, orientation, and illumination changes.

*3.2.4. SURF descriptor.* SURF [4] was proposed as an efficient improvement of SIFT as it aimed to enhance the computation time. It is much faster and more robust as opposed to SIFT. Features extraction is based on the integral image, using the box filter to substitute approximately the second-order Gaussian filter in an efficient way and calculating the Hessian value of the feature points and their surrounding points. The feature description is formed by calculating four kinds of Haar wavelet of the minor area around the feature point. The description vector of the feature point is used to match as SURF algorithm can complete image matching under the moderate conditions. By this way it basically achieved real-time processing.

### 3.3. Matching

Image matching is an important technology in computer vision and image processing, based mainly on keypoint matching. The main target of keypoint matching is to find pixel correspondences representing the same real point in two images. Its ideology is based on looking for the known method of image in others. Basically, the performance of matching techniques based on interest points depends on both the properties of the keypoints and the choice of their associated descriptors. The purpose of keypoints extraction is to match the same feature point in different images, and then complete the matching between images. Once the keypoints and their associated descriptors have been extracted from two or more images, the next step is to establish some feature matches between them. The Bruteforce matcher is examined due to its simplicity and efficiency in the matching task.

*3.3.1. Bruteforce matcher.* The problem of feature matching using Bruteforce can be formulated as follows: suppose that we have two images  $I_1$  and  $I_2$  and for each image we have computed respectively the interest points

$$P = \{ p_1, p_2, \dots, p_N \}, Q = \{ q_1, q_2, \dots, q_M \} \quad (1)$$

and their associated descriptors

$$\Phi(P) = \{\phi_{p1}, \phi_{p2}, \dots, \phi_{pN}\}, \quad \Phi(Q) = \{\phi_{q1}, \phi_{q2}, \dots, \phi_{qM}\} \quad (2)$$

A distance measure between the two interest points descriptors  $\phi_{pi}$  and  $\phi_{qj}$  can be defined as:

$$d_k(p_i, q_j) = \|\phi_{pi} - \phi_{qj}\| \quad \text{if we use real value descriptors} \quad (3)$$

$$d_k(p_i, q_j) = |\phi_{pi} - \phi_{qj}| \quad \text{if we use binary descriptors} \quad (4)$$

Note that other distance measures are used in matching descriptors, like Hellinger [23] and Mahalanobis [24] distance. Based on  $d_k$ , the points of Q are sorted in ascending order independently for each descriptor, creating the sets

$$\Psi(p_i, Q) = \{\psi_k(p_i, Q) \mid k = 1, 2, \dots, k\} \quad (5)$$

such that,

$$\psi_k(p_i, Q) = \{(\psi_k^1, \psi_k^2, \dots, \psi_k^M) \in Q \mid d_k(p_i, \psi_k^s) \leq d_k(p_i, \psi_k^j), \forall s > j\} \quad (6)$$

A match between the pair of interest points  $(p_i, q_j)$  is accepted only if  $p_i$  is the best match for  $q_j$  in relation to all the other points in the first image  $I_1$  and, in its variance with crosscheck, Brute-force crosscheck algorithm requires also that  $q_j$  is the best match for  $p_i$  in relation to all the other points in the second image  $I_2$ .

#### 3.4. RANSAC outliers' rejection algorithm

The Random Sample Consensus (RANSAC) is a simple algorithm for robust fitting of models in the presence of many data outliers. It is a non-deterministic algorithm as it produces a reasonable result only with a certain probability. This probability increases as more iterations are allowed. We can define the probability after  $k$  iteration that we have not picked a set of inlier as  $(1 - G^P)^k$  where  $G$  is the proportion of inlier in the matches found and  $P$  is the number of pair needed for the model, 4 in case of matrix  $H$ . Given a dataset of matches containing both inliers and outliers, RANSAC uses the voting scheme to find the optimal fitting result. Data elements in the dataset are used to vote for one or multiple models. The implementation of this voting scheme is based on two assumptions: that the noisy features will not vote consistently for any single model (few outliers) and there are enough features to agree on a good model (few missing data).

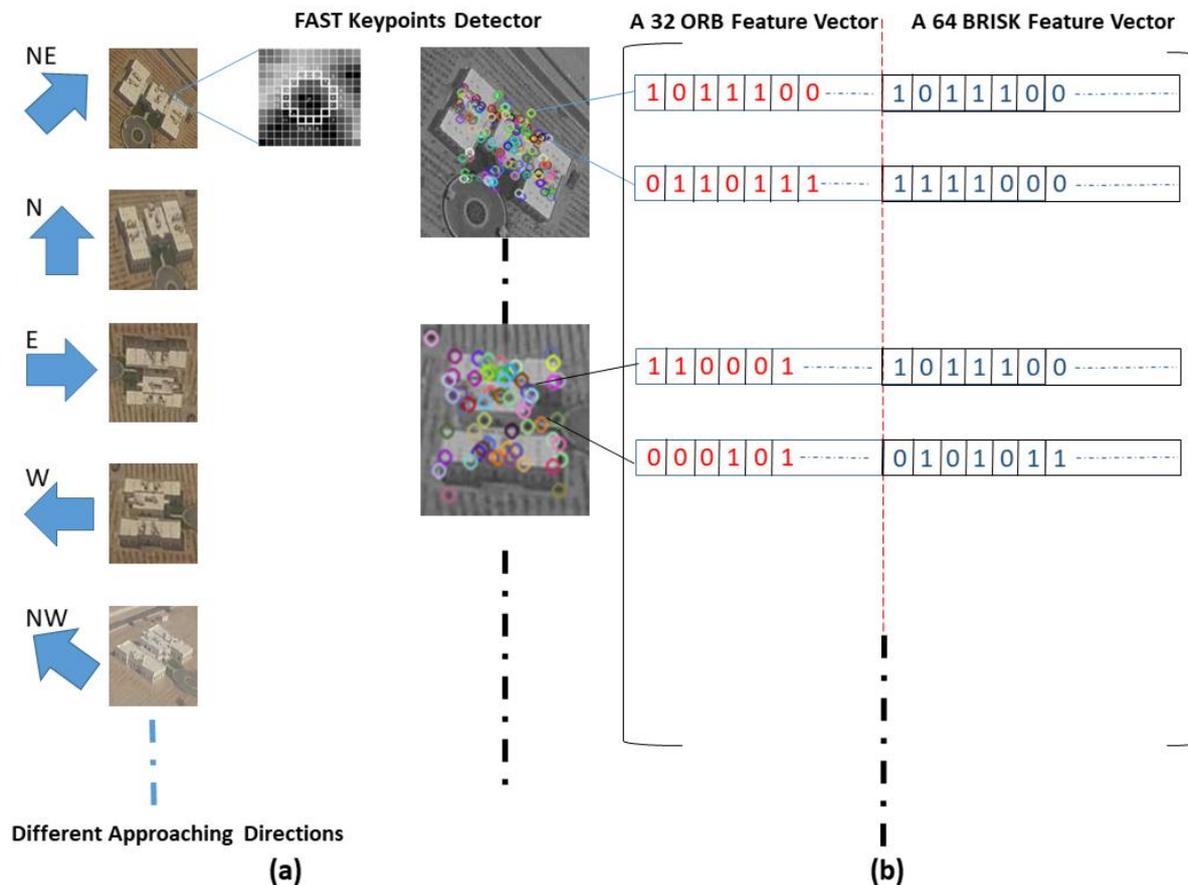
### 4. Proposed detection system

The proposed detection system accomplishes the detection task in two phases; Firstly, integrates the extracted descriptions from a pre-selected set of target reference-images in off-line phase. Secondly, matches the extracted input scene description with the pre-extracted ones in on-line phase to precisely localize the desired target and find its view-point direction. The aforementioned phases are clearly explained in the upcoming subsections.

#### 4.1. Building reference image description (off-line phase)

Detecting the ground target from an aerial scene comprises many challenges. These challenges mainly arise due to the varying in the ground target viewpoint. That's because the airborne camera may capture the desired target from any arbitrary direction surrounding it. To that end, a set of eight distinct viewpoints reference images have been used to establish a comprehensive target description. These reference images are taken from eight quarters when the airborne camera approaches the target from the

North (N) direction, the North-East (NE), and so on up to the South-East (SE) direction. The keypoints are detected using the FAST detector from each reference image to deliver them to the descriptors. Where a fused feature vector for each detected keypoint is established by concatenating both ORB and BRISK features arrays horizontally.

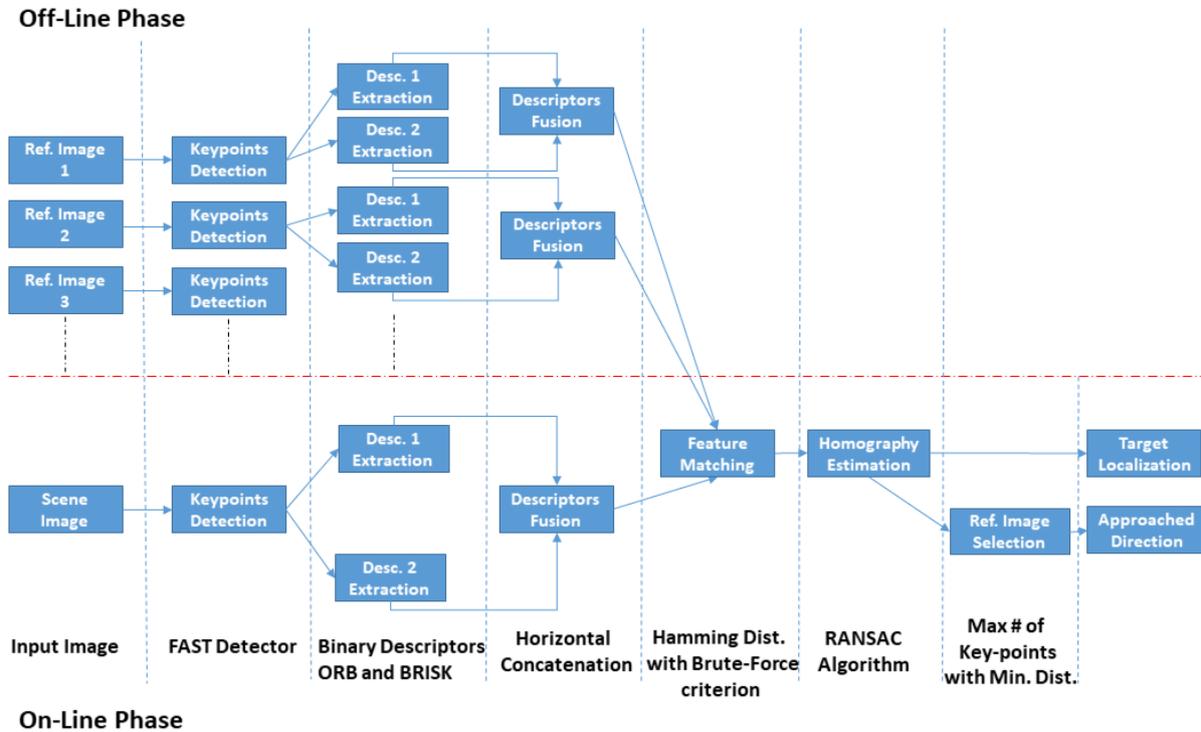


**Figure 2.** The detections of the reference keypoints are carried out from different approaching directions as depicted in figure 2(a). 2(b) Demonstrates the stacking of the fused features to build up the reference image description.

The resulting feature vector has a higher dimension and encodes more information for the detected keypoints to strengthen the matching performance. As depicted in Figure 2(a), each reference image was captured from a different approaching direction to tackle the extracted features viewpoint variant issue. Figure 2(b) demonstrates the stacking of the fused features to build up the reference image description. It is worthy to mention that, all extracted features points are indexed in a manner preserves the reference image number as follows:

$$\Phi(P) = \{\emptyset_{p1,1}, \emptyset_{p2,1}, \dots, \emptyset_{pN_1,1}, \emptyset_{p1,2}, \emptyset_{p2,2}, \dots, \emptyset_{pN_2,2}, \dots, \emptyset_{p1,8}, \emptyset_{p2,8}, \dots, \emptyset_{pN_8,8}\} \quad (7)$$

Thus, for example,  $\emptyset_{p1,2}$  is the fused extracted description for the first keypoint in the second reference image which is annotated by the NE approaching direction and  $N_k$  where  $k = 1, 2, \dots, 8$ , is total of all detected number of keypoints at each reference-image.



**Figure 3.** The proposed detection system block diagram

#### 4.2. Desired target approaching direction and localization (on-line phase)

In the online phase, the input scene keypoints are detected and their descriptions are extracted by the same way of the offline phase. Subsequently, the matching process is carried out to find the correspondence between reference and scene keypoints via the brute-force matcher based on Hamming distance. The matching task is carried out among the scene and all reference images keypoints. Later, all outliers (poor matches) are rejected using the RANSAC algorithm to find the inliers points (good matches). These inliers are utilized to decide the approaching direction alongside the target spatial location as follows:

The reference image which has the maximum number of inliers with the minimum sum of distance is picked to obtain the approaching direction according to this formula:

$$D_r = \frac{\sum_{r,k} d_{r,k} (p_i, q_j)}{N_r} \quad (8)$$

where  $D_r$  is the dissimilarity ratio for the reference image  $r$ ,  $N_r$  is the number of the inliers points for reference image  $r$ ,  $\sum_{r,k} d_{r,k} (p_i, q_j)$  is the sum of all reference-scene  $k$  inliers keypoints distances of the reference image  $r$ . Thus the reference image which achieves the minimum dissimilarity ratio  $D_r$  is chosen as the approaching direction.

The desired target center in the scene image is determined by using the median values of all inliers. Fig.3 shows the proposed detection system block diagram.

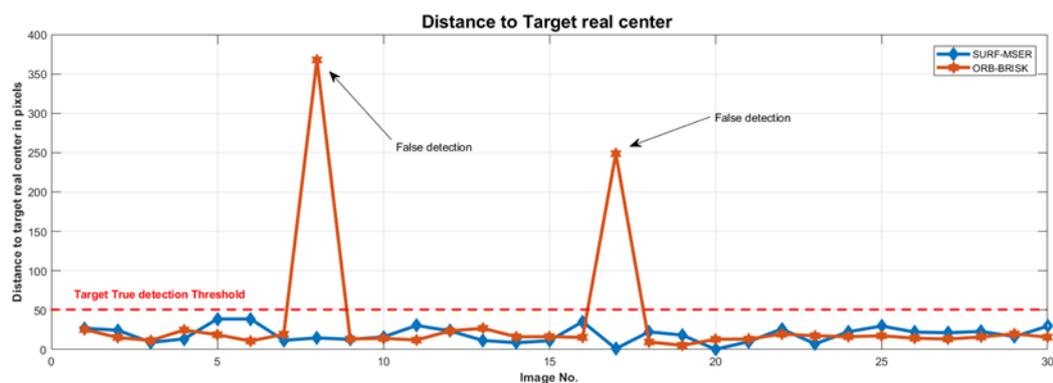
## 5. Experimental results

A method of testing the time and detection accuracy taken by the proposed ORB-BRISK fusion and the SURF-MSER fusion in [25] is illustrated in this section. These values are the total time taken to detect keypoints and to extract descriptors for a dataset of 30 test images with different perspective, rotation

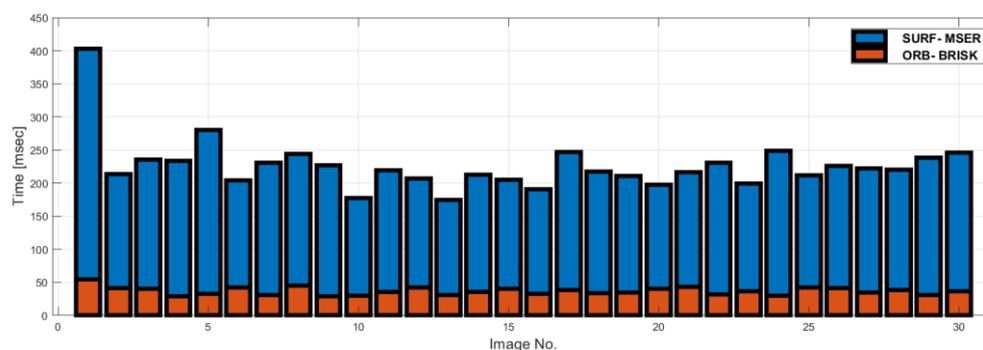
and scaling for a ground stationary target with a combination of 8 reference images taken from different sensor, each one represents an approaching direction. The speed of features detection and extraction is an important factor in real time target localization and recognition. The absolute time taken are dependent on the machine that running the code so we test both algorithms on the same machine with core i7-7700 HQ 2.8GHZ processor and 8 GB ram. Table 1 indicates the detection accuracy and the mean processing time from features detection till target localization and recognition for SURF-MSER and the proposed ORB-BRISK algorithm. Fig.4 shows as indicted in Table 1 the accuracy in object localization of SURF-MSER is a little better than that of ORB-BRISK. But in Fig.5 we can find that the time taken in each algorithm shows that the proposed ORB-BRISK takes mean processing time of 36.93 msec (27 fps) while the SURF-MSER algorithm takes 220 msec (4 fps) processing speed. Fig.6 shows the matched points between reference image and target in scene image using the proposed ORB-BRISK algorithm.

**Table 1.** ORB-BRISK and SURF-MSER accuracy and timing comparison.

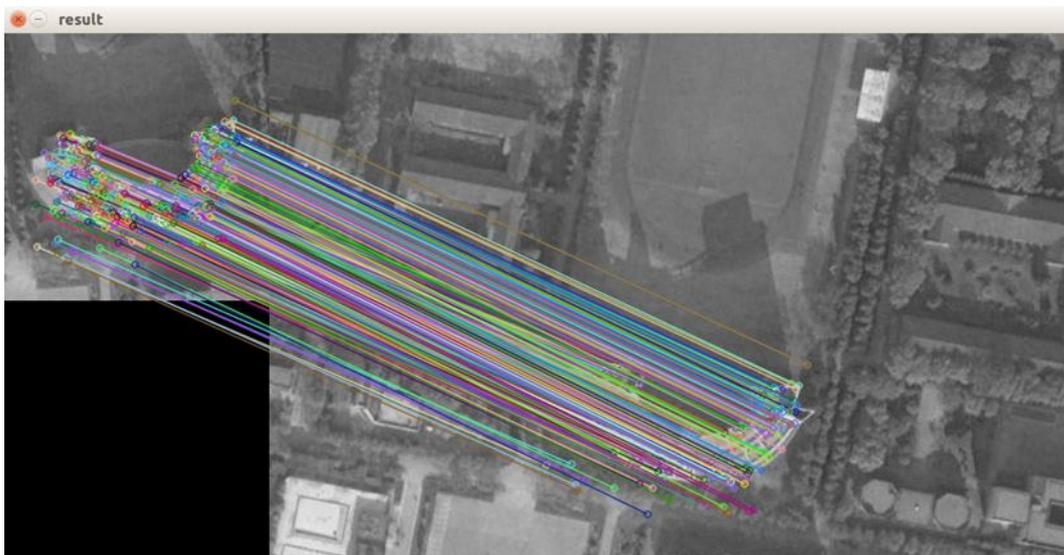
	True Detection (TD)	False Detection (FD)	Mean time Per frame (msec)
ORB-BRISK	28	2	36.93
SURF-MSER	30	0	220



**Figure 4.** ORB-BRISK and SURF-MSER target localization accuracy.



**Figure 5.** ORB-BRISK and SURF-MSER timing comparison.



(a)



(b)

**Figure 6.** ORB-BRISK matched points and inliers. (a) Matched points between reference image and test image, (b) Inliers points only after using RANSAC in test image

## 6. Conclusion

A novel approach for ground target detection via keypoints matching is presented in this paper. The approach accomplishes the ground target detection in two phases. Firstly, off-line phase, where the fused binary features are extracted from different view-point angles of the desired target to build a comprehensive reference image representation. Secondly, on-line phase, where the fused features extraction from the whole scene is accomplished then matched with the stored reference one to find the keypoints correspondence. These correspondences are filtered via RANSAC algorithm to find the good matches keypoints which are utilized to localize the target and obtain its approaching direction. The conducted comparative analysis has revealed the authentic power of the fused features in localization and recognition tasks while keeping the proposed system works in real-time.

## References

- [1] Brown M and Lowe D G 2007 *Automatic panoramic image stitching using invariant features* International Journal of Computer Vision 74 pp 59-73
- [2] Schmid C and Mohr R 1997 *Local grayvalue invariants for image retrieval* Pattern Analysis and Machine Intelligence IEEE Transactions pp 530-535
- [3] Lowe D G 2004 *Distinctive image features from scale-invariant keypoints* International Journal of Computer Vision **60** pp 91-110
- [4] Bay H, Tuytelaars T and Van Gool L 2006 *SURF: Speeded up robust features* Computer Vision ECCV vol 3951 of Lecture Notes in Computer Science Springer Berlin Heidelberg, Graz, Austria pp 404-417
- [5] Rublee E, Rabaud V, Konolige K and Bradski G 2011 *ORB: An ecient alternative to SIFT or SURF* Computer Vision (ICCV) IEEE International Conference Barcelona Spain pp 2564-2571
- [6] Leutenegger S, Chli M and Siegwart R Y 2011 *BRISK: Binary robust invariant scalable keypoints* Computer Vision (ICCV), IEEE International Conference Barcelona Spain pp 2548-2555
- [7] Alcantarilla P F, Nuevo J and Bartoli A 2013 *Fast explicit diffusion for accelerated features in nonlinear scale space* British Machine Vision Conference
- [8] Atrey P K, M. Hossain A, Saddik A E and Kankanhalli M S 2010 *Multimodal Fusion for Multimedia Analysis: A Survey* Multimedia Systems vol 16 pp 345–379
- [9] Snoek C G M, Worring M and Smeulders A W M 2005 *Early Versus Late Fusion in Semantic Video Analysis* Proceedings of the 13th Annual ACM International conference on Multimedia New York USA pp 399–402
- [10] Mikolajczyk K and Schmid C 2005 *A performance evaluation of local descriptors* Pattern Analysis and Machine Intelligence IEEE Transactions pp 1615-1630
- [11] Calonder M, Lepetit V, Strecha C and Fua P 2010 *BRIEF: binary robust independent elementary features* Computer Vision ECCV Vol 6314 of Lecture Notes in Computer Science Springer Crete Greece pp 778-792
- [12] Alahi A, Ortiz R and Vandergheynst P 2012 *FREAK: Fast retina keypoint* Computer Vision and Pattern Recognition (CVPR) IEEE Conference Rhode Island Providence USA pp 510-517
- [13] He D, Liang S and Fang Y 2010 *A multi-descriptor, multi-nearest neighbor approach for image classification* Advanced Intelligent Computing Theories and Applications International Conference on Intelligent Computing ICIC vol. 6215 of Lecture Notes in Computer Science Springer Berlin Heidelberg Changsha China pp 515-523
- [14] Mountney P, B. Lo, Thiemjarus S, Stoyanov D and Zhong-Yang G 2007 *A probabilistic framework for tracking deformable soft tissue in minimally invasive surgery* Medical Image Computing and Computer-Assisted Intervention MICCAI vol 4792 of Lecture Notes in Computer Science Springer Berlin Heidelberg Brisbane Australia pp 34-41
- [15] Bakshi S, Mehrotra H, Raman R and K Sa P 2012 *Score level fusion of SIFT and SURF for IRIS Devices, Circuits and Systems (ICDCS)* International Conference IEEE Coimbatore India pp 527-531
- [16] Perakis P, Theoharis T and Kakadiaris I A 2014 *Feature fusion for facial landmark detection* Pattern Recognition **47** pp 2783-2793
- [17] Hassaballah M et al. 2016 *Image features detection, description and matching* Image Feature Detectors and Descriptors Springer International Publishing ch. 1 pp 11-45

- [18] Rosten, Edward, Porter R and Drummond T 2010 *Faster and better: A machine learning approach to corner detection*”, Pattern Analysis and Machine Intelligence IEEE Transactions PP 105-119
- [19] Harahap H P 2016 *Performance Analysis of Corner Detector Algorithm in Image Noise: Operator Comparison Study*
- [20] Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa and Gerhard Hirzinger 2010 *Adaptive and generic corner detection based on the accelerated segment test* Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) pp 183–196
- [21] Bouchiha R and Besbes K 2013 *Automatic Remote-sensing Image Registration Using SURF* IJCTE pp 88-92
- [22] Rosin P L 1999 *Measuring Corner Properties* Comp. Vis. and Image Understanding pp 291-307
- [23] Arandjelovic Relja and Zisserman Andrew 2012 *Three things everyone should know to improve object retrieval* Proceedings IEEE Computer. Soc. Conf. Computer Vis. Pattern Recognition pp 2911-2918
- [24] Marius Muja, Lowe D G 2009 *Fast approximate nearest neighbors with automatic algorithm configuration* VISAPP International Conference on Computer Vision Theory and Applications pp 331-340
- [25] Fouad A, Taha H S and Elhalwagy Y Z 2015 *Automatic Aerial Target Recognition using a Robust SURF-MSER Feature-based Algorithm* Proceedings of the 16<sup>th</sup> International Conference on Aerospace Sciences and Aviation Technology ASAT-16.