Automatic Aerial Target Recognition using A Robust SURF-MSER Feature-based Algorithm

A. FOUAD*, H. S. TAHA*, Y.Z. ELHALWAGY*

Abstract: Recognition and Identification of Targets in Aerial images play a significant role and wide applications in imaging guidance of cruise missiles, standoff missiles and during terminal phase of ballistic missiles from optics images or radar images. A proposed scene matching algorithm based on both feature detection and extraction Speeded-Up Robust Features (SURF) and Maximally Stable Extremal Regions (MSER) techniques. The extracted features are invariant to image scaling, translation, rotation, and partially invariant to illumination changes and affine or 3D projection. Image data set references of target are collected based on the criteria of azimuth, elevation and altitude of the target image region of interest. The proposed targets are assumed to be buildings. Firstly, SURF and MSER local feature vectors are detected and extracted from reference images of the target and merged to constitute the feature vectors matrix offline. Secondly, a feature detection and extraction using both SURF and MSER algorithms for the real time captured image are performed to form real time feature matrix. Finally, a matching algorithm is applied between both matrices of reference images and real time image. Moreover, Key points outlier elimination technique is used based on a Random Sample Consensus (RANSAC) algorithm to reduce false matching and speed up the matching procedure. The proposed algorithm has been tested on different real image data sets. A comparative analysis is performed between the SURF, FAST corner detector and MSER individually with the proposed SURF-MSER algorithm. The proposed algorithm process more complicated deformation between reference image and real time acquired image and attained higher matching accuracy.

Keywords: Local feature, SURF feature, MSER feature, Scene Matching, RANSAC

1. Introduction

Accurate control and guidance for cruise and ballistic missiles using scene matching in terminal phase is the main objective of this research. In the terminal flight phase, matching should be implemented between the real time image features acquired continuously and the corresponding reference images features loaded in the onboard computer before firing, which can provide the location information of the missile to correct the navigation error. The special application of scene matching has high requirements for matching accuracy and real time performance of the adopted algorithm. The different imaging time and condition lead to the difference of image resolution, and view angle and gray value feature between the reference images and real time image, which makes scene matching a difficult task [1]. The traditional technique is the normal cross-correlation method based on mask operation, that is, the normal cross-correlation between two images is computed on each location, and the location corresponding to the maximum correlation is chosen as the matched location. This technique has problems of matching due to the invariants of the reference and real time images.

*Egyptian Armed Forces

Recently, local features detection and extraction methods that are invariants to affine transformations, scale changes, rotation, blur, and illumination changes are used. They are divided into geometry-based and texture-based features. They can be corners, lines, edges, blobs, even random points, regions, etc. Many approaches are commonly used such as Merovac detector, Harris corner detector [2], SUSAN detector, Harris-Laplacian, SIFT algorithm [3]. The major stages of computation used to generate the set of image features [4]:

1. **Scale-space extrema detection:** The first stage of computation searches over all scales and image locations. It is implemented efficiently by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation.
2. **Key point localization:** At each candidate location, a detailed model is fit to determine location and scale. Key points are selected based on measures of their stability.
3. **Orientation assignment:** One or more orientations are assigned to each key point location based on local image gradient directions. All future operations are performed on image data that has been transformed relative to the assigned orientation, scale, and location for each feature, thereby providing invariance to these transformations.
4. **Key point descriptor:** The local image gradients are measured at the selected scale in the region around each key point. These are transformed into a representation that allows for significant levels of local shape distortion and change in illumination.

A variety of local invariant descriptors have made remarkable progresses, of which Scale Invariant Feature Transform (SIFT) descriptor shows excellent performance [3]. Therefore, SIFT feature matching has find wide application in many fields, such as object recognition, image registration, recognizing panoramas, robot applications, etc [5]. However, the heavy computational burden induced by the underlying complexity of the method makes it cannot meet the real time requirements for scene matching.

A scene matching method based on SURF and MSER feature matching is proposed in this paper. Firstly, SURF and MSER local feature vectors are detected and extracted from reference images of the target and merged to constitute the feature vectors matrix offline. Secondly, a feature detection and extraction using both SURF and MSER algorithms for the real time captured image are performed to form real time feature matrix. Finally, a coarse matching algorithm is applied between both matrices of reference images and real time image. Moreover, Key points outlier elimination technique is used based on a Random Sample Consensus (RANSAC) algorithm to reduce false matching and speed up the matching procedure (fine matching).

If there are more than three pairs of key points in the reference image features matrix and the real time image features matrix are matched correctly, the corresponding reference image and real time image can be regarded as matched. The location of the central of gravity (c.g) of the target can be determined through the transform model constructed by the matched key points. The calculated c.g point location with respect to center of the real time image is used with the guidance and control module for correcting the navigation sensor error. The performance of matching can be measured by repeatability, which is determined by the percentage of points simultaneously present in two images. Direct point correspondences should be established. The higher the repeatability rate between two images, the more points can potentially be matched and the better are the matching and tracking results. This paper is divided into 4 sections. Sections 2, introduces the local feature vector detection and extraction algorithms. Section 3 describes the proposed scene matching algorithm while section 4 illustrates the experimental results and performance of the proposed algorithm. Finally, section 5 presents a conclusion for the published work.

2. Local feature detection, extraction and matching algorithms

Two algorithms of local feature detection and extraction will be introduced in this work. The first method is called the SURF key points blobs feature detection and extraction which is the extension of SIFT technique [6]. The second method is the MSER that based on support regions [7]. Both algorithms are multi scale techniques. Next, an exploration of each method in details will be introduced in addition to coarse feature matching and fine matching using RANSAC outlier rejection.

2.1 SURF key points detection and feature extraction algorithm

SURF algorithm is inspired by SIFT algorithm, and uses same matching scheme as SIFT, but it has some additional improvements. Otherwise, the performance in repeatability and distinctive is comparable to SIFT algorithm.

SURF algorithm is achieved by detecting the interest points as follows [6, 8]:

1. Compute a Hessian matrix-based measure as an interest point detector.
2. Approximate Gaussian second order derivative with a box filter.
3. Rely on integral images for rapid image convolutions to construct scale space.
4. Extract maxima of determinant of the Hessian matrix as candidate interest points

SURF algorithm is achieved by interest point description and matching as follows:

1. Assign orientation to ensure invariance to rotation.
2. Determine descriptors based on sum of Haar wavelet responses 350
3. Check the sign of Laplacian indexing strategy to make rapid matching without any loss in terms of performance.
4. Match based on Euclidean distance.

A SURF feature can be explained with four parameters including location, sign and descriptor. The important speed gain of SURF is due to the use of integral images, which drastically reduce the number of operations for simple box convolutions, independent of the chosen scale. The bigger scale of box filter, the more obvious of benefit in computation. Moreover, the sign of the Laplacian allows for a quick distinction between bright features on a dark background and dark features on a bright background. This approach allows for quicker feature matching without any performance loss. The features with different signs can be avoided calculating distances in high dimension. The wavelet responses are summed up over each sub region and form a first set of entries to the feature vector. Each sub region has a four dimensional descriptor vector for its underlying intensity structure as in equation (1) that results descriptor vector of length 64 for each 4×4 blocks.

$$V = (\sum dx, \sum dy, \sum |dx|, \sum |dy|) \tag{1}$$

The detector used in SURF feature is the Fast- Hessian detector [6]. The detection can be stated such as a point x =(x, y) in an image I, its Hessian matrix at scale σ is defined as:

$$H(X, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{yx}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \tag{2}$$

Where $L_{xx}(x, \sigma)$ represents the convolution of the Gaussian second order derivative with the image $I(x, y)$ at point x, and similarly for $L_{xy}(x, \sigma)$ and $L_{yy}(x, \sigma)$.

Herbert proposed to use box filter to approximate the second order Gaussian partial derivatives and use integral images to speed up the convolution, through which the computational burden can be decreased greatly. The mask of the box filter with size 9 X 9 of the approximation for the second order Gaussian partial derivatives by box filters on the original image, image pyramid with different scale is constructed through expanding the size of box filter as shown in Fig. 1. After the approximated determinant of the Hessian matrix at

each scale is calculated, the non-maximum suppression in 3×3×3 neighborhood is applied to find the maxima. After the interpolation in both scale space and image space, the stable location of SURF key points and the value of located scale can be obtained. The convolution of each box filter and the image can be denoted by D_{xx}, D_{yy} and D_{xy} respectively, therefore

$$\det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \tag{3}$$

The dominant orientation is determined for each SURF key point in order to guarantee the invariance of feature description to image rotation. The Haar wavelet responses in both x and y direction within a circular neighborhood of radius 6s around the key point is calculated, with s the scale at which the key point was detected. The responses are represented as points in a space with the horizontal response strength along the abscissa and the vertical response strength along the ordinate. The sum of the horizontal and vertical responses within a sliding orientation window of size 60° is calculated to yield a local orientation vector, and the longest vector over all windows defines the dominant orientation of the key point. A square region with size 20s centered around the key point and oriented along the dominant orientation is constructed and divided into 4 × 4 sub-regions. In each sub-region, the horizontal and vertical Haar wavelet response dx and dy are computed and summed up. In addition, the absolute values of the responses are also calculated and summed up. Then a 4-D feature V is formed for each sub-region. Therefore, for each extracted SURF key point, a 4×(4×4) description vector can be constructed [6,8].
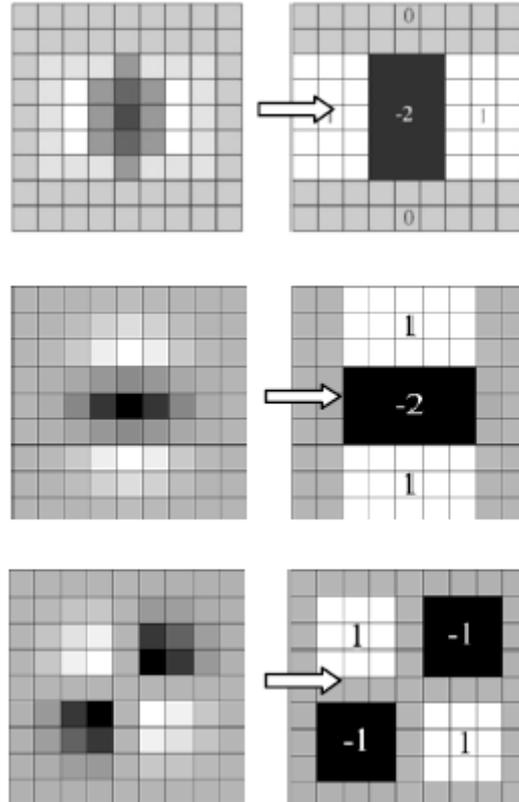

Fig. 1. Approximation for the second order Gaussian partial derivatives

2.2 MSER region detection and feature extraction algorithm

Maximally Stable Extremal Regions have been proposed by Matas et al. [7]. It is a connected component of an appropriately threshold image. The word "extremal" refers to the property that all pixels inside the MSER have either higher (bright extremal regions) or lower (dark extremal regions) intensity than all the pixels on its outer boundary. The regions are defined solely by an extremal property of the intensity function in the region and on its outer boundary [9]. In many images, local binarization is stable over a large range of thresholds in certain regions. Such regions are of interest since they possess the following properties [10]:

- Invariance to affine transformation of image intensities.
- Covariance to adjacency preserving (continuous) transformation $T : D \rightarrow D$ on the image domain.
- Stability, since only extremal regions whose support is virtually unchanged over a range of thresholds is selected.
- Multi-scale detection. Since no smoothing is involved, both very fine and very large structure is detected.
- The set of all extremal regions can be enumerated in $O(n \log \log n)$, where n is the number of pixels in the image.

Enumeration of extremal regions proceeds as follows. First, pixels are sorted by intensity. The computational complexity of this step is $O(n)$ if the range of image values S is small, e.g. the typical $[0, \ldots, 255]$, since the sort can be implemented as BINSORT [7]. After sorting, pixels are placed in the image (either in decreasing or increasing order) and the list of connected components and their areas is maintained using the efficient union find algorithm. The complexity of our union-find implementation is $O(n \log \log n)$„ i.e. almost linear. Importantly, the algorithm is very fast in practice. The process produces a data structure storing the area of each connected component as a function of intensity. A merge of two components is viewed as termination of existence of the smaller component and an insertion of all pixels of the smaller component into the larger one. Finally, intensity levels that are local minima of the rate of change of the area function are selected as thresholds producing maximally stable extremal regions. In the output, each MSER is represented by position of a local intensity minimum (or maximum) and a threshold. Many affine invariant detectors output shape are ellipses. However ellipses can be fitted based on the first and second shape moments [11].

2.3 Features matching and RANSAC outliers rejection algorithm

Blob and region features extracted using SURF and MSER are merged for both reference images constituting one matrix for reference images features and another matrix for the real time image features. A coarse-to-fine matching is implemented between both features [12]. If there are more than three pairs of key points in the reference images and the real time image are matched correctly, the corresponding reference image and real time image can be regarded as matched [8]. The coarse matching is processed by computing the sum of squared differences (SSD) metric between features of the reference images and features of the real time image. The feature matching metric used is based on approximate nearest neighbor search such that this method is recommended for large feature sets. The fine matching is based on estimation of the geometric transformation from matching point pairs. The proposed method is called RANSAC algorithm. The RANSAC algorithm relies on a distance threshold. A pair of Key points is an inlier only when the distance between Key Point of real time image and the projection of Key Point of reference image based on the transformation matrix falls within the specified threshold. The distance metric used in the RANSAC algorithm is defined as follows:

$$d = \sum_{i=1}^{Num} \min\left(D\left(P_i^{rt}, \Psi\left(P_i^{ref}:H\right)\right), t \right) \tag{4}$$

Where P_i^{ref} is a point in reference image key points set, P_i^{rt} is a point in real time image key points set, $\Psi\left(P_i^{ref}:H\right)$ is the projection of a point on reference image based on transformation matrix H, $D\left(P_i^{rt}, P_j^{rt}\right)$ is the distance between two point pairs on real time image, t is the threshold, *Num* is the total number of Key points.

In general, RANSAC method is an effective technique of approximating model parameter; it is adopted to eliminate outliers [14]. The chosen matching model in RANSAC estimation is the affine transform model, so in each process of model estimation, the necessary minimum number of points used to determine affine transform model is 3. After using RANSAC, the obtained matched point sets have relatively high matching accuracy. RANSAC is capable of interpreting or smoothing data containing a significant percentage of gross errors, and is thus ideally suited for applications in automated image recognition where interpretation is based on the data provided by error prone feature [13].

A theoretical study for a given image that depicting a set of features with known locations, new results are derived on the minimum number of features needed to obtain a solution, and RANSAC algorithm is presented for computing these minimum-feature solutions in closed form. High dimension descriptor, such as SURF can get correct match ratio about 30%~50% while other descriptors with lower dimension can only get correct match ratio about 20%~30%. RANSAC can get correct model from samples with over a half of outlier theoretically. The outlier ratio is ε and m correspondences are required in every sample, then the probability that at least one out of M samples including m inliers is defined as: $P = 1 - (1 - (1 -)^m)^M$ If P is settled, then

$$M = \frac{\log\left(1 - P\right)}{\log\left(1 - (1 -)^m\right)} \tag{5}$$

According to equation (5), the smaller of the outlier ratio, the fewer samples are required. According to Nistér's method [14], at least 5 correspondences are required to estimate fundamental matrix. If the outlier ratio is 50% ~ 70%, hundreds or thousands samples are required. Fundamental matrix contains the information about the relative transformation between two camera pose and plays an important role in computer vision [1].

3. Proposed Target Recognition Algorithm Based on SURF-MSER
The proposed algorithm is based on two main processing phases. The first phase is an offline stage as shown in Fig. 2 such that a certain number N of reference images contain the templates of the target are collected. The criteria of collection are based on selecting different viewpoints (azimuth and elevation angles) and altitudes corresponding scale change. A smoothing process is performed using 3x3 median filters for all reference images to eliminate noise. Next, Feature detection using SURF and MSER algorithms is performed. Moreover, Feature extraction vectors of length 64 using SURF and MSER are calculated. Finally, a merging process for both feature vectors of SURF and MSER is done as follows:

$$\vec{F}_{Refs} = \left\{ \vec{F}_{Refs\_SURF}, \vec{F}_{Refs\_MSER} \right\} \tag{6}$$

Where $\vec{F}_{Refs}$ represents the merged features for all reference images based on SURF and MSER algorithms while $\vec{F}_{Refs\_SURF}$ represents the features for all reference images based on

SURF algorithm and $\vec{F}_{Refs\_MSER}$ represents the features for all reference images based on MSER algorithm. The second phase is a real time stage as shown in Fig. 3. A capturing process for the real time scene image is performed. A smoothing is performed using 3x3 median filters for image noise elimination of the real time image. Next, Feature detection using SURF and MSER algorithms is performed. Moreover, Feature vectors of length 64 extraction using SURF and MSER are calculated. Feature vectors of SURF and MSER is merged follows:

$$\vec{F}_{RTSCENE} = \left\{ \vec{F}_{RTSCENEs\_SURF}, \vec{F}_{RTSCENEs\_MSER} \right\} \tag{7}$$

Where $\vec{F}_{RTSCENE}$ represents the merged features for the real time scene image based on SURF and MSER algorithms while $\vec{F}_{RTSCENE\_SURF}$ represents the features for the entire real time scene image based on SURF algorithm and $\vec{F}_{RTSCENE\_MSER}$ represents the features for the real time scene image based on MSER algorithm. Finally, a feature matching between $\vec{F}_{Refs}$ and $\vec{F}_{RTSCENE}$ is performed to provide the corresponding key points (inliers) in the real time scene. RANSAC algorithm is implemented to reject the outliers. The central of gravity (c.g) of the target is evaluated using the statistical median of all X and Y points location for the inliers on the target.
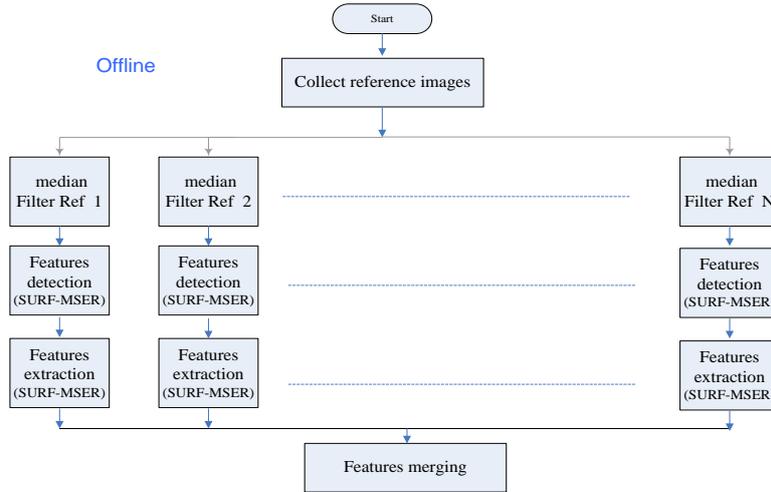

**Fig. 2. Proposed algorithm for feature detection, extraction
and merging of the reference images**

4. Experimental results and performance analysis

Experiments on several real sets of images are carried out to test the performance of the proposed algorithm, and excellent results are obtained. These real sets are described as follows: Imageset1 consists of 3 test real images and 1 reference real image from the same video. Imageset2 consists of 17 test real images and 2 reference real images. Imageset3 consists of 45 test real images and 4 reference real images from the same video and 2 real images obtained from Google earth for the target area. Imageset4 consists of 70 low resolution test real images and 6 high resolution reference real images. Imageset5 consists of 80 high resolution test real images and 6 low resolution reference real images.

```
                              ┌─────────┐
                              │  Start  │
                              └─────────┘
                                   │
                        ┌──────────────────────┐
                        │  Capture scene image  │
                        └──────────────────────┘
                                   │
                        ┌──────────────────────┐
                        │     Median filter     │
                        └──────────────────────┘
                          │                  │
          ┌──────────────────────┐   ┌──────────────────────┐
          │  Features detection   │   │  Features detection   │
          │        (SURF)         │   │        (MSER)         │
          └──────────────────────┘   └──────────────────────┘
          ┌──────────────────────┐   ┌──────────────────────┐
          │  Features extraction  │   │  Features extraction  │
          │        (SURF)         │   │        (MSER)         │
          └──────────────────────┘   └──────────────────────┘
                          │                  │
                        ┌──────────────────────┐
                        │    Feature merging    │
                        │     SURF & MSER       │
                        └──────────────────────┘
                                   │
                        ┌──────────────────────┐
                        │   Features matching   │
                        └──────────────────────┘
                                   │
                        ┌──────────────────────┐
                        │   Outliers rejection  │
                        └──────────────────────┘
                                   │
                        ┌──────────────────────┐
                        │  Computing c.g of target │
                        │   (median for inliers) │
                        └──────────────────────┘
                                   │
                   NO           ◇ Mission ◇
                ◄──────────────   End
                                   │ YES
                              ┌─────────┐
                              │   END   │
                              └─────────┘
```
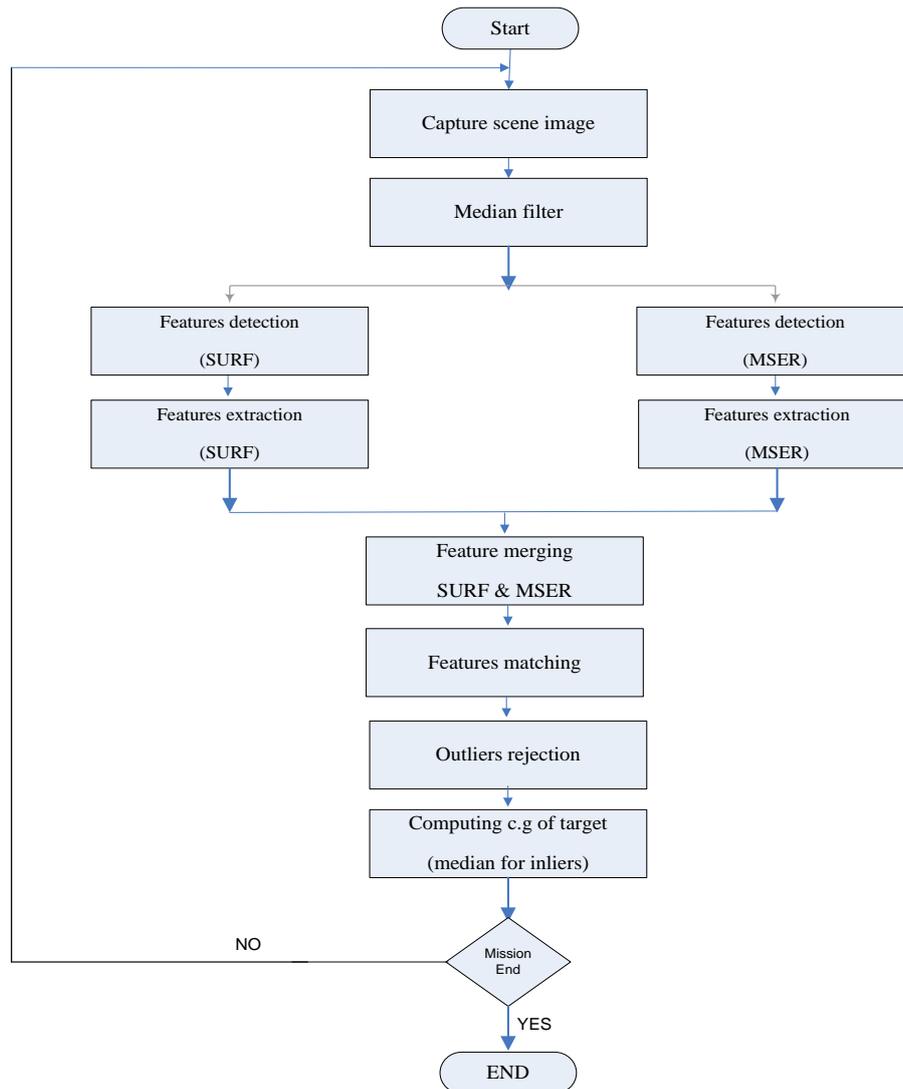
**Fig. 3. Proposed algorithm for feature detection,
extraction, merging a of the real time image and matching
with reference images features with outlier elimination**

Fig. 4.a and Fig. 4.b represent two real time images from Imageset1 with different geometric deformations exist between them. Fig. 4.c presents the results of feature detection and extraction of the proposed SURF-MSER algorithm and illustrates the matching key points between a reference image and real time image with an existing deformation. The theoretical location of central of gravity (c.g) of building targets are identified manually. The matching accuracy is the sum of square difference (SSD) distance between the theoretical location and the calculated location of the building targets based on the proposed SURF-MSER algorithm. The value of matching accuracy is considered in pixels. Fig. 5.a illustrates the matching accuracy between the three real time images target and reference image in Fig.4.c such that using the proposed algorithm is more accurate than SURF alone. Under the hardware configuration of Intel core 2 Duo processor with 3G memory, and Matlab 2014.b, a comparison of the elapsed time in seconds between the proposed SURF-MSER and standalone SURF indicating more elapsed time for matching for the proposed algorithm as shown in Fig. 5.b. Fig.6.a-c present another experiments for Imagset2 while Fig. 7.a and Fig.7.b illustrate a comparative analysis for the matching accuracy and matching elapsed time for Imageset2 using proposed algorithm against SURF standalone. Fig. 8.a illustrates a

comparison of the matching accuracy between the SURF, MSER and FAST algorithms individually indicating that the SURF is the best one using Imageset3. Fig. 8.b illustrates a comparison of the matching accuracy between the proposed SURF-MSER and SURF algorithm standalone indicating that the proposed algorithm is better using Imageset4. Fig. 8.c illustrates a comparison of the matching accuracy between the proposed SURF-MSER and SURF algorithm standalone indicating that the proposed algorithm is better using Imageset5. Fig. 9.a illustrates a comparison of the matching elapsed between the SURF, MSER and FAST algorithms individually indicating that the SURF is the fastest one using Imageset3. Fig. 9.b illustrates a comparison of the matching elapsed time between the proposed SURF-MSER and SURF algorithm standalone indicating that the proposed algorithm is taking longer time using Imageset4. Fig. 9.c illustrates a comparison of the matching elapsed time between the proposed SURF-MSER and SURF algorithm standalone indicating that the proposed algorithm is taking longer time using Imageset5. Table 1 presents a comparison between the true detection (TD) and false alarm (FA) using SURF, MSER and FAST algorithms individually for the probability of matching success. A TD case is considered if the matching accuracy is less than certain threshold value based on the target size as shown in Fig. 8.c. The calculation is based on counting the number of TD for the targets of the Imageset3 divided by the total number of images indicating that the SURF algorithm is the best with probability %100. Table 1 is tabulated based on Fig. 8.a. While Table 2 illustrates a comparison between the true detection (TD) and false alarm (FA) the proposed SURF-MSER and SURF algorithm standalone for the probability of matching success of the Imageset1, Imageset2, Imageset4 and Imageset5 indicating that the proposed algorithm is the best with highest probability. Table 2 is tabulated based on Fig. 5.a, Fig. 7.a, Fig. 8.b and Fig. 8.c.
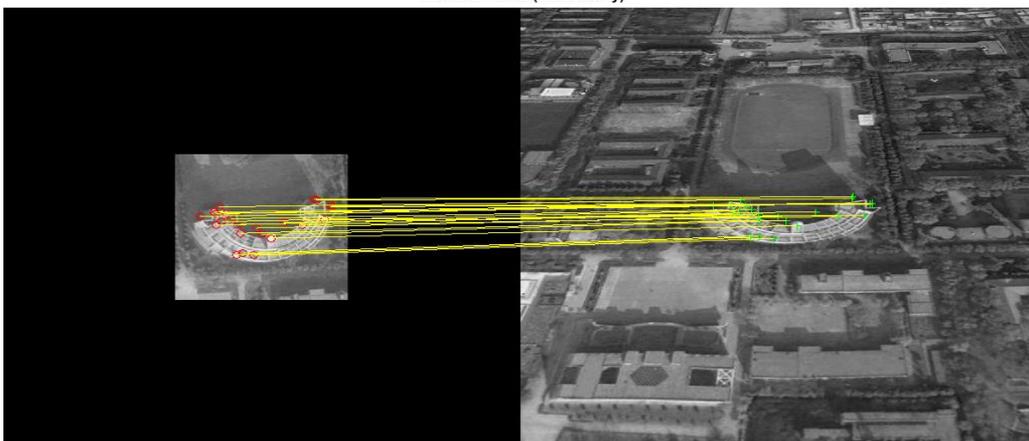

(a) Real Time Image 1

(b) Real Time Image 2

(c) Matching Key Points

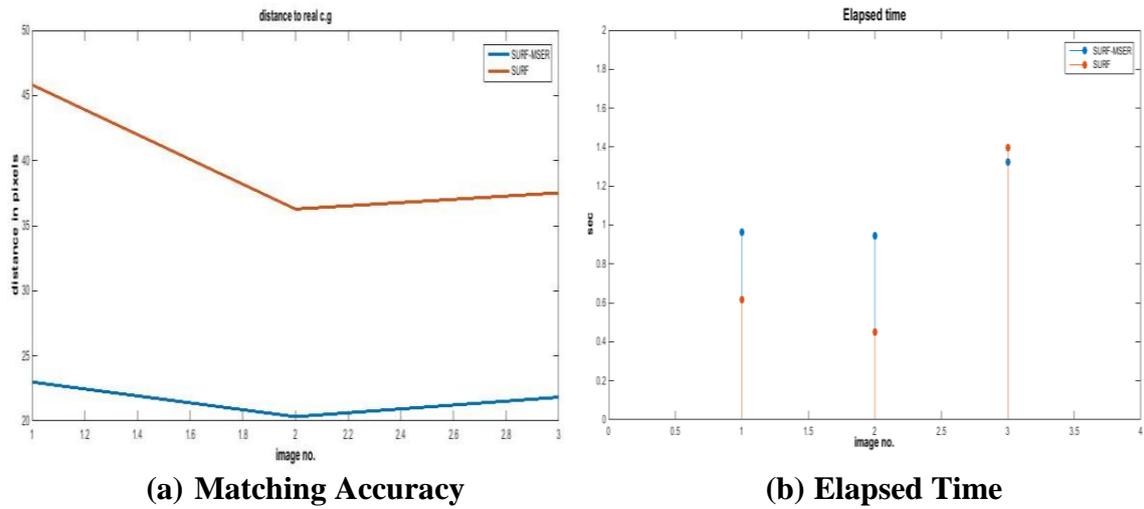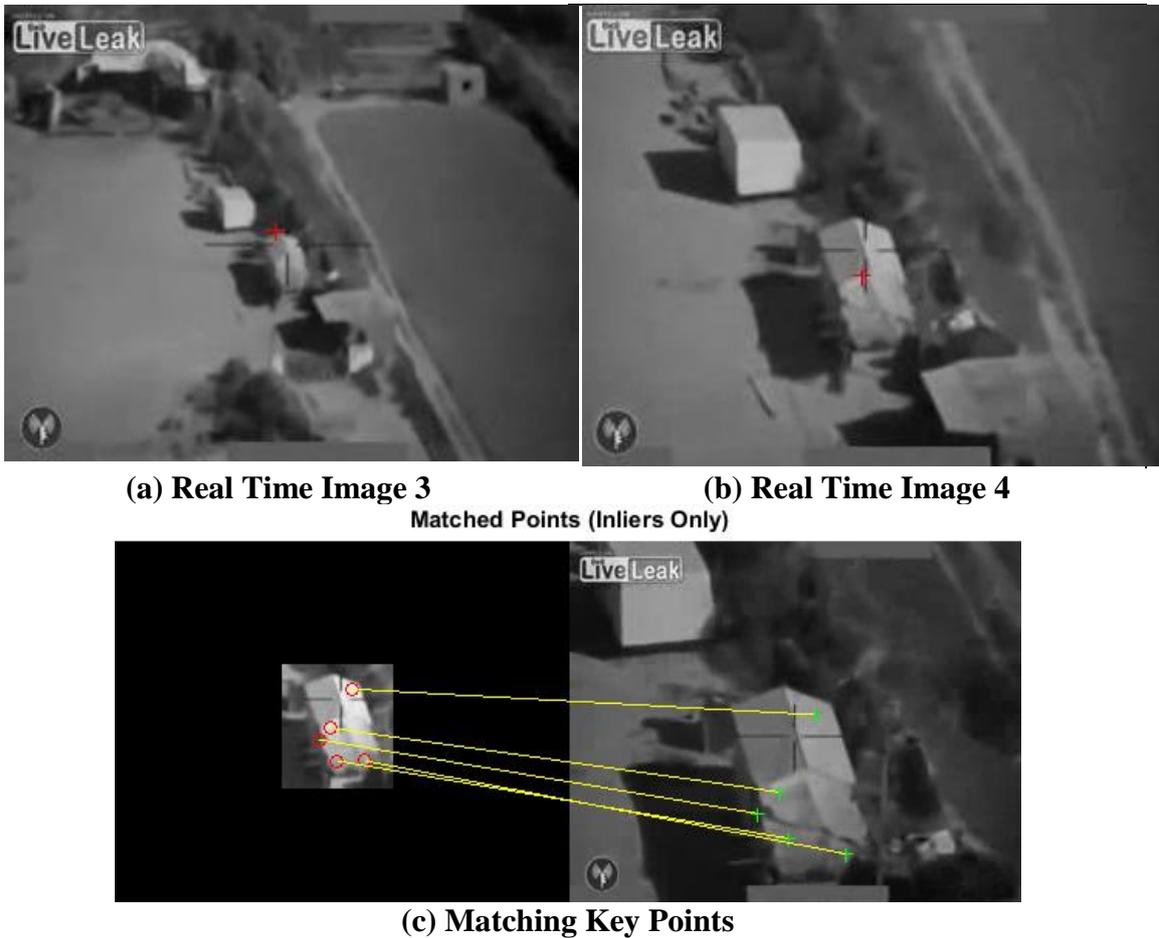**Fig. 4. Imageset1 Frames and Feature Matching based on the Proposed Algorithm**
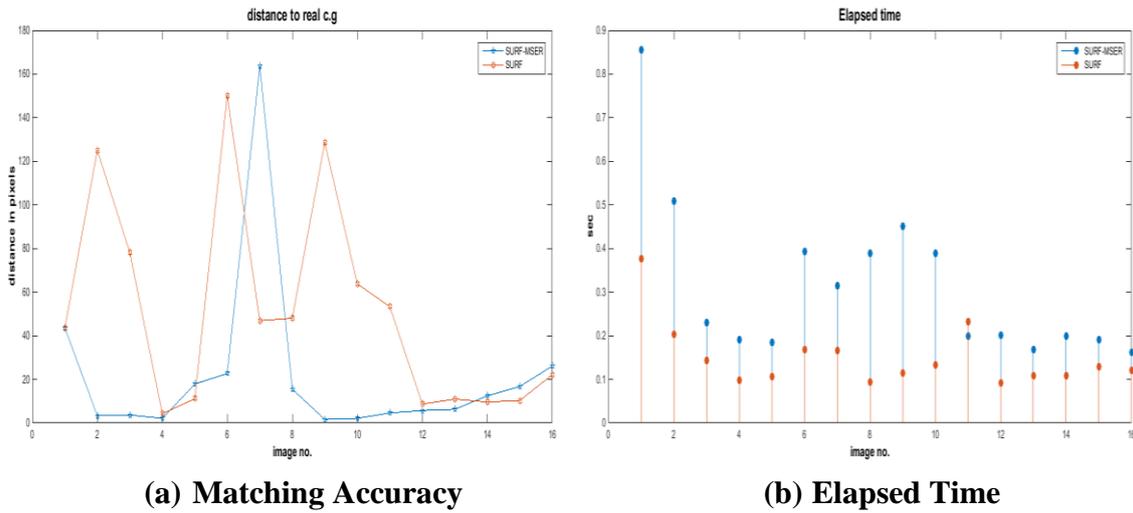

(a) Matching Accuracy

(b) Elapsed Time

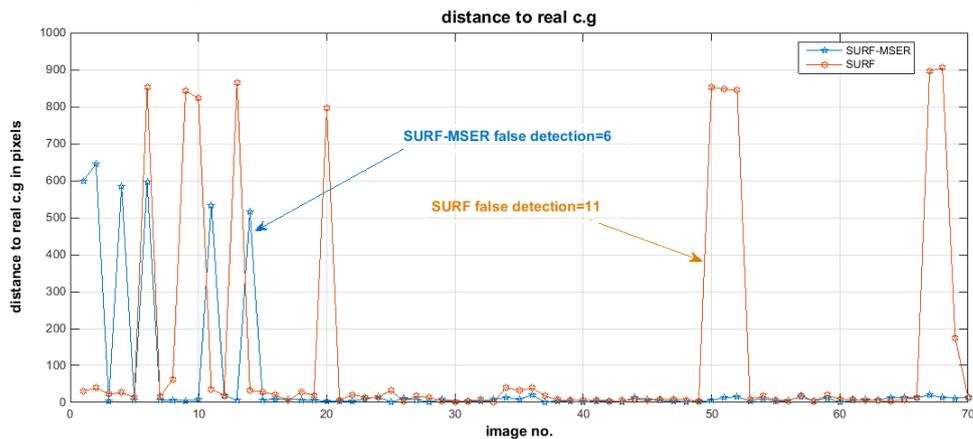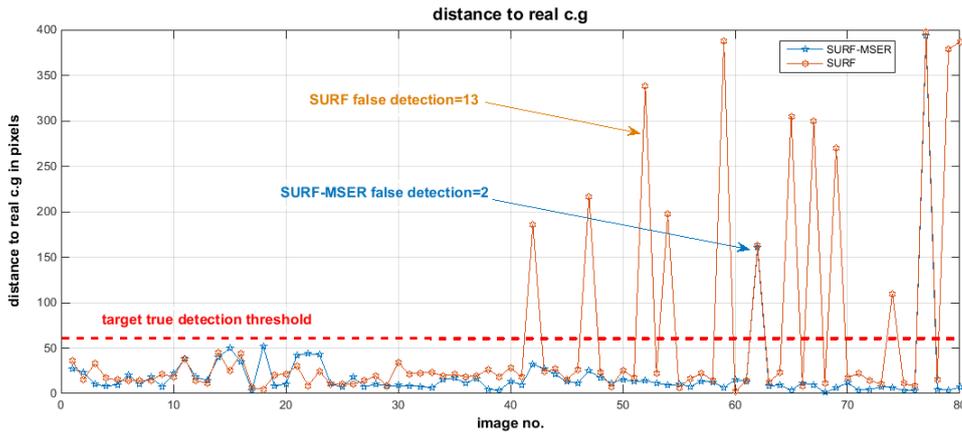**Fig. 5. Matching accuracy and elapsed time for Imageset1**

(a) Real Time Image 3

(b) Real Time Image 4

(c) Matching Key Points

Fig. 6. Imageset2 Frames and Feature Matching based on the Proposed Algorithm

(a) Matching Accuracy

(b) Elapsed Time

Fig. 7. Matching accuracy and elapsed time for Imageset2
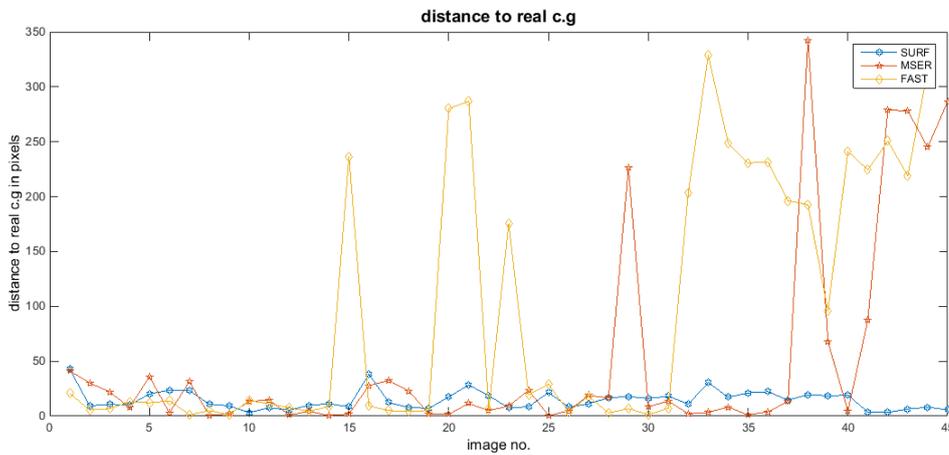
(a) Matching accuracy for Imageset3 based on SURF, MSER and FAST
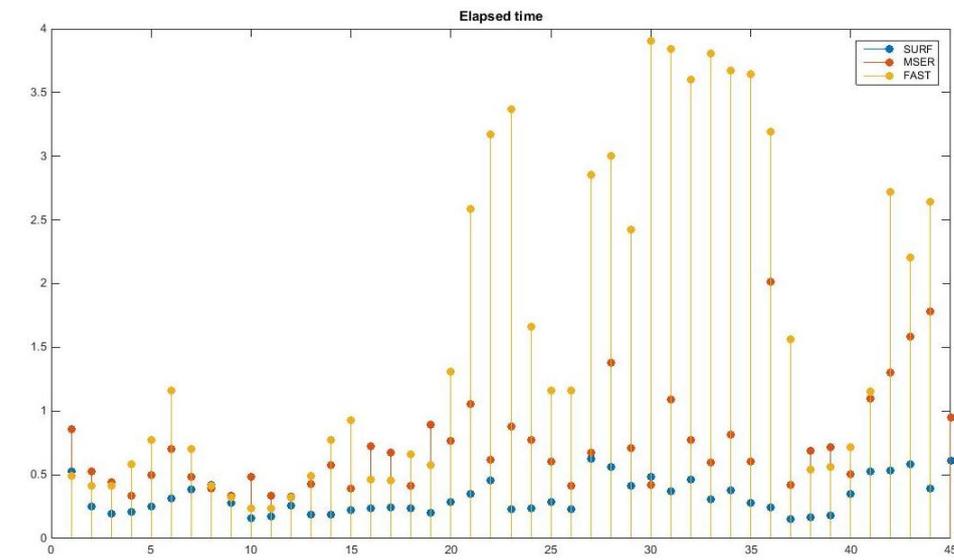

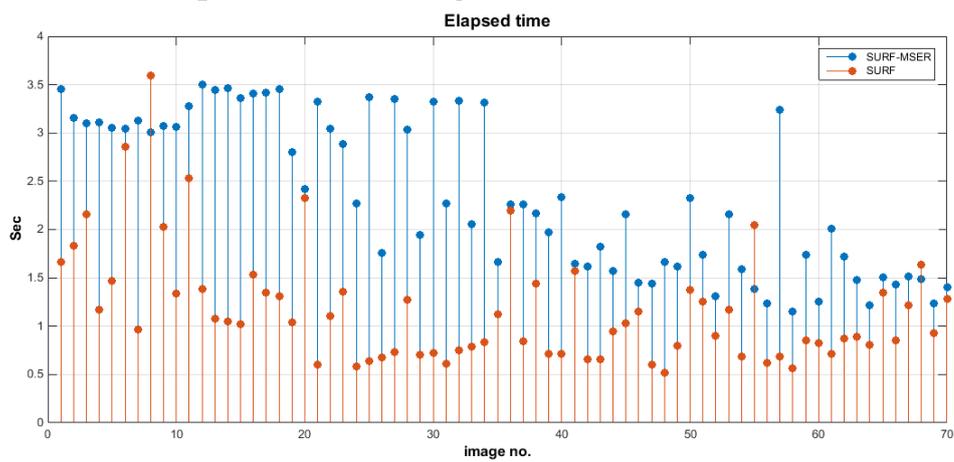**(b) Matching accuracy for Imageset4**
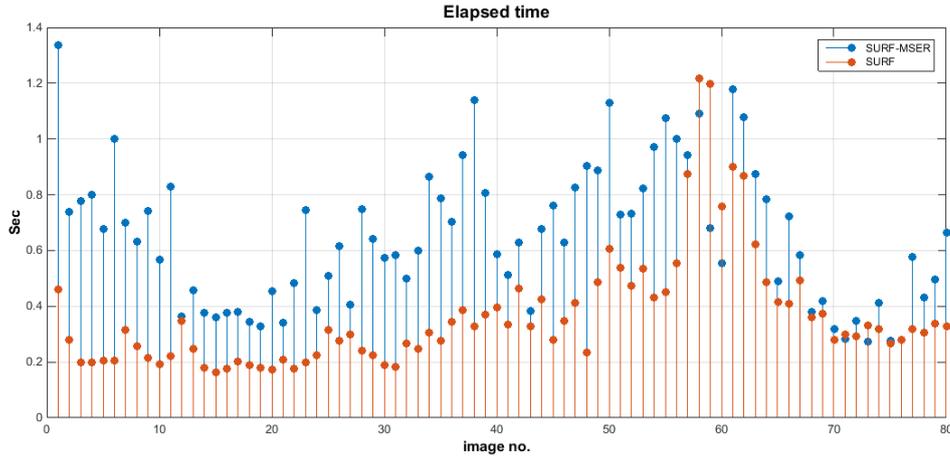
(c) Matching accuracy for Imageset5

Fig. 8. Matching accuracy for Imagesets4,5 based on the proposed algorithm

(a) Elapsed Time for Imageset3 based on SURF, MSER and FAST

(b) Elapsed Time for Imageset4

(c) Elapsed Time for Imageset5

Fig. 9. Elapsed Time for Imagesets4, 5 based on proposed algorithm

Table 1. A Comparison of the matching between the SURF, FAST and MSER feature detection and extraction using Imageset3

True Detection (TD)			False Detection (FD)			Probability of Success %		
SURF	FAST	MSER	SURF	FAST	MSER	SURF	FAST	MSER
45	28	37	0	17	8	100	62.22	82.22

Table 2. A Comparison of the matching between the SURF and proposed SURF-MSER feature detection and extraction

IMAGE SETS	True Detection (TD)		False Detection (FD)		Probability of Success %	
	SURF	Proposed	SURF	Proposed	SURF	Proposed
Imageset1	3	3	0	0	100.0	100.0
Imageset2	6	13	9	3	37.50	81.25
Imageset5	67	78	13	2	83.75	97.50
Imageset4	59	64	11	6	84.20	91.40

5. Conclusion

 In this paper, a literature surveying for some common vision matching algorithms is studied. The SURF and MSER local feature detection and extraction algorithm are considered the robustness according to invariant to image scaling, translation, rotation, and partially invariant to illumination changes and affine or 3D projection. A proposed merging algorithm for both SURF and MSER is used for a preselected reference target images according to certain criteria offline. In real time manner the proposed algorithm detects and extracts the feature vectors for the real time scene. A matrix constituting both feature vectors of SURF and MSER for the real time image is matched with the matrix feature of the reference image. A robust matching process based on computing the euclidian distance between the feature vectors in addition to, outlier rejection procedure based on RANSAC algorithm is used. The location of central of gravity of the building targets according to the correspond inliers on the target is used to improve the error of guidance and control for the application of terminal phase of cruises and ballistic missiles A comparative and performance analysis is performed on a lot of

real image data sets indicating high robustness to various image deformations therefore its feasibility and effectiveness make it more suitable for scene matching.

References

[1] Tsai, Shi-Xue. Introduction to the scene matching missile guidance technologies. No. NAIC-ID (RS) T-0379-96. NATIONAL AIR INTELLIGENCE CENTER WRIGHT-PATTERSON AFB OH, 1996.

[2] C. Harris, M.Stephens, "A combined corner and edge detector", Fourth Alvey Vision Conference, Manchester, UK, 1988, pp. 147-151.

[3] D. Lowe, "Distinctive image features from scale-invariant keypoints", IJCV, 2004, vol. 60, pp. 91–110.

[4] Lenc, Karel, Jirı Matas, and Dmytro Mishkin. "A Few Things One Should Know About Feature Extraction, Description and Matching." 19th Computer Vision Winter Czech Republic, 2014.

[5] Loncomilla, Patricio, and Javier Ruiz-del-Solar. "Improving SIFT-based object recognition for robot applications." *Image Analysis and Processing–ICIAP 2005*. Springer Berlin Heidelberg, 2005. 1084-1092.

[6] H. Bay, A. Ess, T. Tuytelaars, L. V. Gool. Speeded-Up Robust Features (SURF), Computer Vision and Image Understanding, pp.346-359, 2008, vol.110.

[7] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," in Proceedings of the British Machine Vision Conference, pp. 384–393, 2002.

[8] Su Juan, Xu Qingsong, Zhu Jinghua. A Scene Matching Algorithm Based on SURF Feature, IEEE, 2010.

[9] Kimmel, Ron, et al. "Are MSER features really interesting?." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33.11 (2011): 2316-2320.

[10] Li, Jing, and Nigel M. Allinson. "A comprehensive review of current local features for computer vision." *Neurocomputing* 71.10 (2008): 1771-1787.

[11] Tuytelaars, Tinne, and Krystian Mikolajczyk. "Local invariant feature detectors: a survey." Foundations and Trends® in Computer Graphics and Vision 3.3 (2008): 177-280.

[12] Sirmacek, Beril, and Cem Unsalan. "A probabilistic framework to detect buildings in aerial and satellite images." Geoscience and Remote Sensing, IEEE Transactions on 49.1 (2011): 211-221.

[13] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. Commun. Assoc. Comp. Mach., vol. 24:381–95, 1981.

[14] Yang, Shuqiang, and Biao Li. "Outliers Elimination Based RANSAC for Fundamental Matrix Estimation." Virtual Reality and Visualization (ICVRV), 2013 International Conference on. IEEE, 2013.