An Assessment of Image Matching Algorithms in Depth Estimation

Ashraf Anwar *ashraaf@tu.edu.sa*
College of Computers and Information Technology
Taif University, Taif, Saudi Arabia

Ibrahim El Rube *ielrube@yahoo.com*
College of Computers and Information Technology
Taif University, Taif, Saudi Arabia

Abstract

Computer vision is often used with mobile robot for feature tracking, landmark sensing, and obstacle detection. Almost all high-end robotics systems are now equipped with pairs of cameras arranged to provide depth perception. In stereo vision application, the disparity between the stereo images allows depth estimation within a scene. Detecting conjugate pair in stereo images is a challenging problem known as the correspondence problem. The goal of this research is to assess the performance of SIFT, MSER, and SURF, the well known matching algorithms, in solving the correspondence problem and then in estimating the depth within the scene. The results of each algorithm are evaluated and presented. The conclusion and recommendations for future works, lead towards the improvement of these powerful algorithms to achieve a higher level of efficiency within the scope of their performance.

Keywords: Stereo vision, Image Matching Algorithms, SIFT, SURF, MSER, Correspondence.

1. INTRODUCTION

Stereo vision systems are used to determine depth from two images taken at the same time but from slightly different viewpoints using two cameras. The main aim is to calculate disparity which indicates the difference in locating corresponding pixels in two images. From the disparity map, we can easily calculate the correspondence of objects in 3Dspace which is known as depth map. The known algorithms for stereo matching can be classified in two basic categories: Feature-based algorithms and area based algorithms [2-14]. The algorithms of both categories often use special methods to improve the matching reliability.

Matas et al [14] find maximally stable extremely regions (MSER) correspondences between image elements from two images with different viewpoints. This method of extracting a comprehensive number of corresponding image elements contributes to the wide-baseline matching, and it has led to better stereo matching and object recognition algorithms. David. G and Lowe [15] proposed a scale invariant feature transform (SIFT) detector and descriptor, which detects a set of local feature vectors through scale space extremes and describe this feature using 3D histogram of gradient and orientation. Also Hess [16] introduced an open source SIFT library. Herbert Bay, et. al, [17] proposed SURF (Speeded-Up Robust Features) as a fast and robust algorithm for local, similarity invariant image representation and comparison. SURF selects interest points of an image from the salient features of its linear scale-space, and then builds local features based on the image gradient distribution. An open source SURF library is introduced by Evans, C. [18].

In this paper, we propose to use one of the well known image matching algorithms (SIFT, MSER, or SURF) in estimating the distance between the SVS surveyor robot, shown in figure 1, and the in front obstacles. Therefore, an evaluation and assessment of the performance of the three

image matching algorithms is conducted to determine the best applicable algorithm for the SVS surveyor robot.

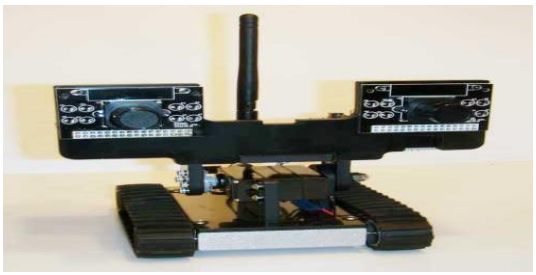

FIGURE 1: SVS Surveyor Robot.

This paper is organized as follows: section 2 presents the overview on the SVS surveyor robot. Stereo vision concept illustrated in section 3. Section 4 is dedicated to image matching algorithms. Section 5 is reserved to the methodology. Simulation results are presented in section 6. Finally section 7 concludes this paper.

2. SVS SURVEYOR ROBOT

The SVS surveyor robot [1] is designed for research, education, and exploration, Surveyor's internet-controlled robot. The robot is usually equipped with two digital video cameras with resolution from 160x128 to 1280x1024 pixels, two laser pointers, and WLAN 802.11b/g networking on a quad-motor tracked mobile robotic base.

Operating as a remotely-controlled webcam or a self-navigating autonomous robot, the robot can run onboard interpreted C programs or user-modified firmware, or be remotely managed from a Windows, Mac OS/X or Linux base station with Python or Java-based console software.

2.1 Stereo Vision System Specifications

The two SRV-1 Blackfin camera modules are separated by 10.75 cm (4.25"). Each camera module includes:

- 500MHz Analog Devices Blackfin BF537 Processor (1000 integer MIPS), 32MB SDRAM, 4MB SPI Flash, JTAG, external 32-pin i/o header w/ 2 UARTS, 4 timers (PWM/PPM), SPI, I2C, 16 GPIO

- Omnivision OV9655 1.3 megapixel sensor with AA format header and interchangeable lens - M12 P0.5 format - 3.6mm f2.0 (90-deg FOV) or optional 2.2mm f2.5 (120-deg FOV)

3. STERO VISION CONCEPT

3.1 Basics

The geometric basis key problem in stereo vision is to find corresponding points in stereo images. Corresponding points are the projections of a single 3D point in the different image spaces. The difference in the position of the corresponding points in their respective images is called disparity (see figure 2). Two cameras: Left and Right, Optical centers: OL and OR. Virtual image plane is projection of actual image plane through optical centre. Baseline, b, is the separation between the optical centers. Scene Point, P, imaged at PL and PR. Disparity, d = PR – PL.

Disparity is the amount by which the two images of P are displaced relative to each other
Depth, Z=bf/p*d

Where p: pixel width

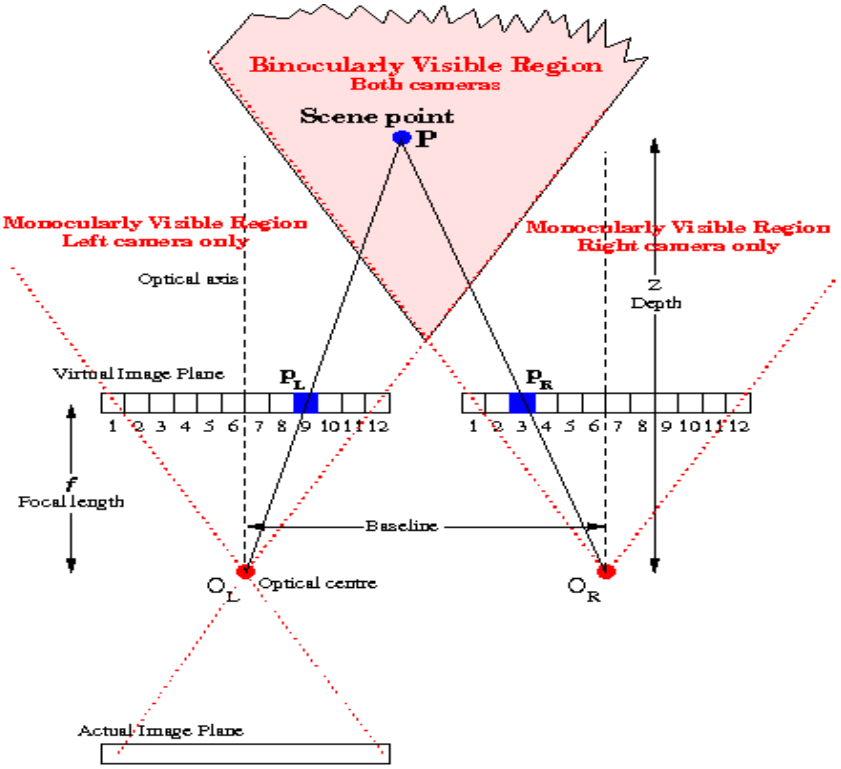

FIGURE 2: Stereo Vision Basics.

In addition to providing the function that maps pair of corresponding images points onto scene points, a camera model can be used to constraint the search for corresponding image point to one dimension. Any point in the 3D world space together with the centers of projection of two cameras systems, defines an epipolar plane. The intersection of such a plane with an image plane is called an epipolar line (see figure 3). Every point of a given epipolar line must correspond to a single point on the corresponding epipolar line. The search for a match of a point in the first image therefore is reduced to a one-dimensional neighborhood in the second image plane.

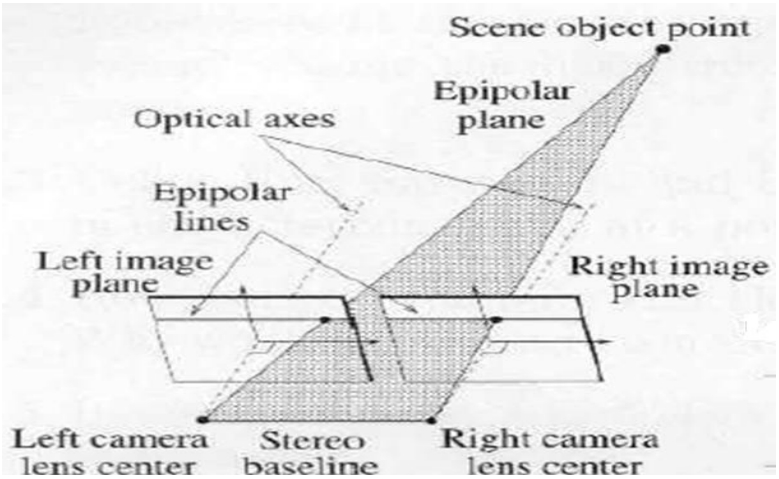

FIGURE 3: Epipolar Lines and Epipolar Planes.

3.2 Correspondence problem

There are two issues, how to select candidate matches? and how to determine the goodness of a match? Two main classes of correspondence (matching) algorithm: First, Correlation-based, attempt to establish a correspondence by matching image intensities, usually over a window of pixels in each image. Second Feature-based, attempt to establish a correspondence by matching sparse sets of image features, usually edges. Disparity map is sparse, and number of points is related to the number of image features identified. Feature-based methods, suitable when good features can be extracted from the scene, faster than correlation-based methods, provide sparse disparity maps, suitable for applications like visual navigation, and relatively insensitive to illumination changes.

4. IMAGE MATCHING ALGORITHMS

4.1 The Scale Invariant Feature Transform (SIFT) Algorithm

The SIFT algorithm operates in four major stages [16, 20] to detect and describe local features, or keypoints, in an image:

- Scale-space extrema detection. The SIFT algorithm begins by identifying the locations of candidate keypoints as the local maxima and minima of a difference-of-Gaussian pyramid that approximates the second order derivatives of the image's scale space.

- Keypoint localization and filtering. After candidate keypoints are identified, their locations in scale space are interpolated to sub-unit accuracy, and interpolated keypoints with low contrast or a high edge response computed based on the ratio of principal curvatures are rejected due to potential instability.

- Orientation assignment. The keypoints that survive filtering are assigned one or more canonical orientations based on the dominant directions of the local scale-space gradients. After orientation assignment, each keypoint's descriptor can be computed relative to the keypoint's location, scale, and orientation to provide invariance to these transformations.

- Descriptor computation. Finally, a descriptor is computed for each keypoint by partitioning the scale-space region around the keypoint into a grid, computing a histogram of local gradient directions within each grid square and concatenating those histograms into a vector. To provide invariance to illumination change, each descriptor vector is normalized to unit length, threshold to reduce the influence of large gradient values, and then renormalized.

For image matching and recognition, SIFT features are first extracted from a set of reference images and stored in a database. A new image is matched by individually comparing each feature from the new image to this previous database and finding candidate matching features based on Euclidean distance of their feature vectors..

4.2 The Maximally Stable Extremely Regions (MSER)

It is a feature detector; Like the SIFT detector, the MSER algorithm extracts from an image a number of co-variant regions, called MSERs. An MSER is a *stable* connected component of some level sets of the image. Optionally, elliptical frames are attached to the MSERs by fitting ellipses to the regions. Because the regions are defined exclusively by the intensity function in the region and the outer border, this leads to many key characteristics of the regions which make them useful. Over a large range of thresholds, the local linearization is stable in certain regions, and have the properties listed below.

- Invariance to affine transformation of image intensities

- Covariance to adjacency preserving (continuous)transformation $T : D \rightarrow D$on the image domain
- Stability: only regions whose support is nearly the same over a range of thresholds is selected.
- Multi-scale detection without any smoothing involved, both fine and large structure is detected.
  Note however that detection of MSERs in a scale pyramid improves repeatability, and number of correspondences across scale changes.

This technique was proposed by Matas et al. [14] to find correspondences between image elements from two images with different viewpoints. This method of extracting a comprehensive number of corresponding image elements contributes to the wide-baseline matching, and it has led to better stereo matching and object recognition algorithms.

4.3 Speeded Up Robust Features (SURF)
It is a robust local feature detector, first presented by Herbert Bay et al. [17, 20], it can be used in computer vision tasks like object recognition or 3D reconstruction. SURF is based on sums of 2D Haar wavelet responses and makes an efficient use of integral images.

The steps of features detection as follows:

- Interest points are selected at distinctive locations in the image, such as corners, blobs, and T-junctions. The most valuable property of an interest point detector is its repeatability, i.e. whether it reliably finds the same interest points under different viewing conditions.

- Next, the neighborhood of every interest point is represented by a feature vector. This descriptor has to be distinctive and, at the same time, robust to noise, detection errors, and geometric and photometric deformations.

- Finally, the descriptor vectors are matched between different images. The matching is often based on a distance between the vectors, e.g. the Mahalanobis or Euclidean distance. The dimension of the descriptor has a direct impact on the time this takes, and a lower number of dimensions is therefore desirable.

5. METHODOLOGY

5.1 Data source
Two SRV-1 Blackfin camera modules, illustrated in section 2

5.2 Camera Calibration
The result of camera calibration using the Camera Calibration Toolbox for Matlab [19] is obtained in Tables 1 and 2 for left and right cameras of the SVS stereo system mentioned in section 2, respectively.

Focal Length	fc_left = [390.97269 371.48472] ± [90.30192 86.25323]
Principal point:	cc_left = [176.99127 -0.14410] ± [0.00000 0.00000]
Skew	alpha_c_left = [0.00000] ± [0.00000] => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion	kc_left = [1.03881 -2.69365 -0.01420 0.03846 0.00000] ± [1.38449 5.07636 0.12276 0.02306 0.00000]

TABLE 1: Intrinsic Parameters of Left Camera.

Focal Length	fc_right = [490.50860 470.70292] ± [94.29747 97.23895]
Principal point:	cc_right = [159.50000 119.50000] ± [0.00000 0.00000]
Skew	alpha_c_right = [0.00000] ± [0.00000] => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion	kc_right = [-0.76164 9.78187 0.18255 0.00095 0.00000] ± [1.30013 11.26048 0.09762 0.01903 0.00000]

TABLE 2: Intrinsic Parameters of Right Camera.

5.3 Test Algorithm

The framework of the proposed algorithm is summarized in the following steps:

- Step 1: Read stereo image pair.
- Step 2: Compute interest points for each image using SURF/SIFT/MSER algorithm.
- Step 3: Find point correspondences between the stereo image pair.
- Step 4: Remove outliers using geometric constraint.
- Step 5: Remove further outliers using Epipolar constraint.
- Step 6: Rectify images such that the corresponding points will appear on the same rows.
- Step 7: Obtain the disparity map and calculate the depth.

5.4 Processing Steps.

In order to assess the performance of the three image matching algorithms, SIFT, MSER, and SURF, we applied the proposed algorithm on a set of images captured by stereo vision system of SVS surveyor robot. The image-pairs are captured at different distances, 50cm, 100cm, 150cm, 200cm, 250cm, and 300cm, as shown in figures 4 to 9 (a-l, a-r; ;b-l, b-r; c-l, c-r; d-l, d-r; e-l, e-r; f-l, f-r).

5.5 Performance Evaluation.

The effectiveness of the algorithm is calculated using the following formula:

$$E\% = \frac{\#\ \text{correct matches}}{\min\ (\#\text{left features},\ \#\ \text{rightFeatures})} 100 \quad (1)$$

The depth accuracy is calculated according to the following formula:

$$\text{Depth Acc } \% = \frac{\text{Real Depth} - \text{Min. Depth}}{\text{Real Depth}} 100 \quad (2)$$

FIGURE 4: Image pair at 50cm from stereo camera

FIGURE 5: Image pair at 100cm from stereo camera

c-l

c-r

Figure 6: Image Pair at 150cm From Stereo Camera.

d-l

d-r

FIGURE 7: Image Pair at 200cm From Stereo Camera.

e-l

e-r

FIGURE 8: Image Pair at 250cm From Stereo Camera.

f-l

f-r

Figure 9: Image Pair at 300cm From Stereo Camera.

6. SIMULATION RESULTS

6.1 Platform

The simulation is performed using matlab software (R2012a). The computer processor is Intel ® core TM, i5, M430, 2.27 GHz. The matching algorithms, SURF, SIFT, and MSER are tested individually by every image-pair, illustrated in figure 4 to figure 9 according to the steps of processing explained in section 5.4. The image results of every step are shown in figure 10. The performance results of SIFT, MSER, and SURF, are tabulated in Table 3, Table 4, and Table 5 respectively. The bar plots for features detected, effectiveness, and depth accuracy, for the three matching algorithms, are illustrated in figure 11, figure 12, and figure 13 respectively.

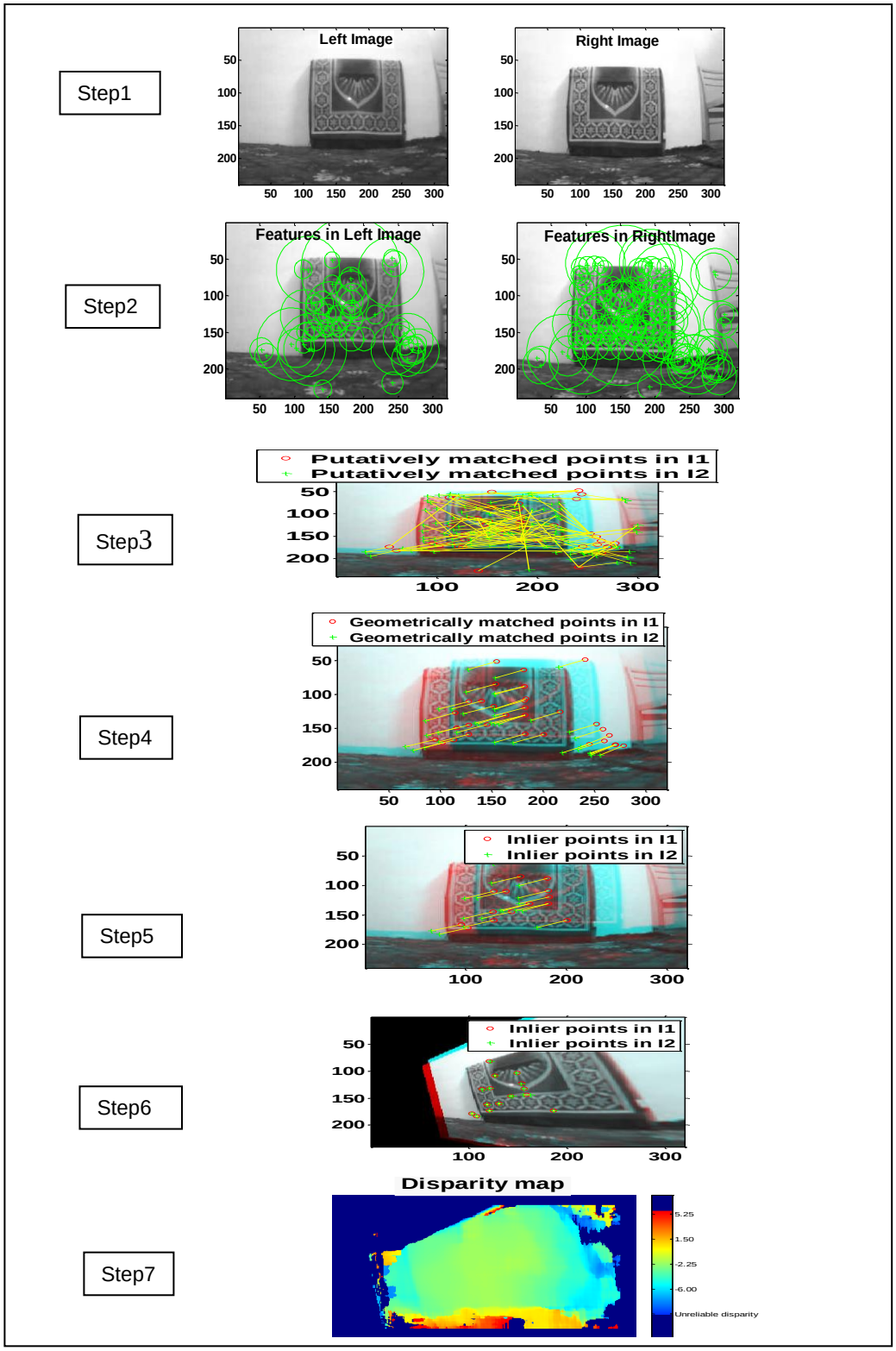

FIGURE10: Image Results.

Image pair	Real depth (cm)	Features		Match features	#Correct matches	E%	Min. depth(cm)	Depth acc%
		Left	Right					
a	50	713	753	178	68	9.5	63.1	73.8
b	100	638	646	198	106	16.6	116.8	83.2
c	150	664	513	136	54	10.5	165.3	89.8
d	200	640	644	101	28	4.3	172.8	86.4
e	250	594	609	62	15	2.5	156.7	62.7
f	300	582	429	85	17	3.9	263.7	87.9

TABLE 3: SIFT Matching Results.

Image pair	Real depth (cm)	Features		Match features	# Correct matches	E%	Min. depth(cm)	Depth acc%
		Left	Right					
a	50	186	220	342	18	9.6	51.1	97.8
b	100	164	193	277	42	25.6	93.5	93.5
c	150	65	127	157	20	30.7	133.3	88.8
d	200	70	74	118	14	20	168.2	84.0
e	250	34	34	58	10	29.4	130	52.0
f	300	25	41	53	12	48	135	45.0

TABLE 4: MSER Matching Results.

Image pair	Real depth (cm)	Features		Match features	# Correct matches	E%	Min. depth (cm)	Depth acc%
		Left	Right					
a	50	299	335	482	27	9.0	50.2	99.6
b	100	153	267	314	26	16.9	94.1	94.1
c	150	48	128	139	17	35.4	140.8	93.8
d	200	27	32	45	12	44.4	171.8	85.9
e	250	21	29	37	9	42.8	213.9	85.5
f	300	22	27	34	9	40.9	234.1	78.0

TABLE 5: SURF Matching Results.

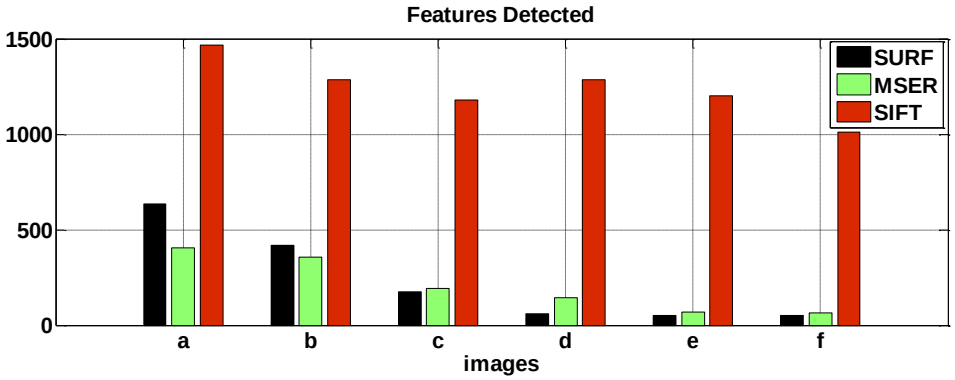

Figure 11: Bar Plot of the Detected Features.

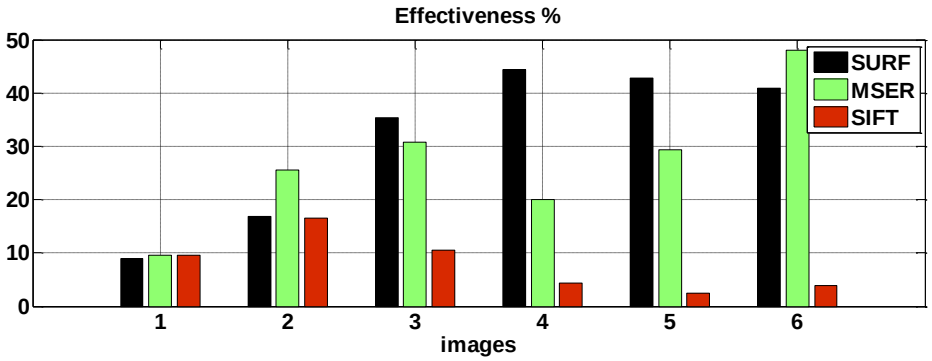

FIGURE 12: Bar Plot of the Algorithms Effectiveness.

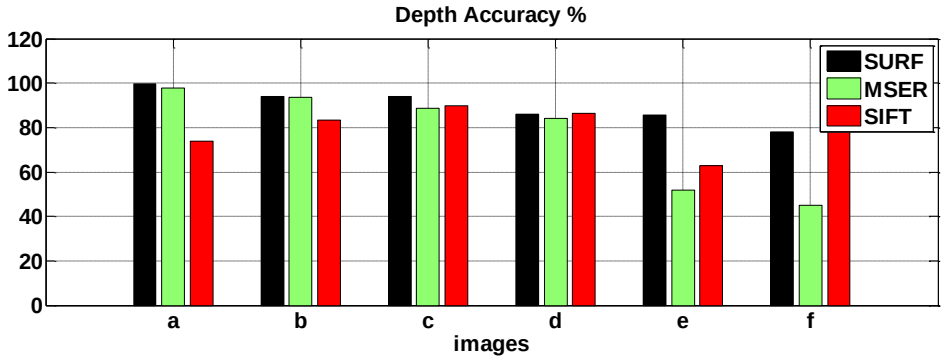

FIGURE13: Bar Plot of the Depth Accuracy.

6.2 Analysis of the Results

After reviewing the results obtained, we have the following notes:

a) The amount of features detected by SIFT, MSER and SURF is dependent on the depth between the stereo camera and the object.

b) In case of SURF, we decrease the metric threshold to 500 instead of 1000 (default value), when the distance greater than 200 cm in order to get appropriate features.

c) In case of MSER, we decrease the threshold delta to min value (0.8) instead of 2 (default value), when the distance greater than 200 cm in order to get appropriate features.

d) The amount of features is not a measure of success by itself but the "quality" of these features

e) The amount of features detected is proportional to the amount of matches.

f) Although the SIFT detect more matches, but the SURF gives the best result in estimating the depth in all images. We can deduce that, the amount of matches detected is not a good indication of the performance of the algorithm.

g) Matches detected by SURF, although fewer, are more robust than those detected by SIFT and MSER.

6.3 Comparative Evaluation

To evaluate the performance of the our test algorithm in the depth estimation, we get also the maximum depth in case of using SURF as a matching algorithm (which gives the best results), and then we get the average depth between the minimum depth and the maximum depth. The results are given in Table 6.

Real depth (cm)	Estimated depth (cm)		Mean depth (cm)	Average absolute error (cm)
	Min.	Max.		
50	50.2	53.4	51.8	1.8
100	94.1	110.1	102.1	2.1
150	140.8	168.5	154.65	3.6
200	171.8	235.8	203.8	3.8
250	213.9	302.7	258.3	8.3
300	234.1	391.1	312.6	12.6

TABLE 6: Estimated Depth Results.

The comparison between the results obtained by Young [21] method in distance estimation and our test algorithm is illustrated in Table 7.

Real depth (cm)	Young [21] Method		Our test algorithm	
	Average absolute error(cm)	% Average absolute error	Average absolute error(cm)	% Average absolute error
50	2.95	5.9	1.8	3.6
100	2.95	2.9	2.1	2.1
150	3.97	2.6	3.6	2.4
200	3.96	2.9	3.8	1.9
250	9.65	3.9	8.3	3.3
300	17.33	5.8	12.6	4.2
% Total average		4		2.9

TABLE 7: Comparative Results.

7. CONCLUSION

In this paper, we assess the performance of SIFT, MSER, and SURF, the well known matching algorithms, in solving the correspondence problem and then in estimating the depth within the scene. Furthermore we proposed a framework for estimating the distance between the robot and in front obstacles using the robot stero camera setup. The results show that the amount of features is not a measure of success by itself but the “quality” of these features. Although the SIFT, detect more matches, but the SURF gives the best result in estimating the depth in all images. We deduce that, the amount of matches detected is not a good indication of the performance of the algorithm. Matches detected by SURF, although fewer, are more robust than those detected by SIFT and MSER. It is concluded that SURF has the best overall performance against SIFT and MSER algorithms. The proposed framework using SURF algorithm performed sginifintly better than a a recent algorithm published, by other researchers, at the same depthes. Future work related to this research will be directed to implement SURF algorithm in real time stereo vision navigation and obstacle avoidance for autonomous mobile robot.

8. REFERENCES

1. http://www.surveyor.com/stereo/stereo_info.html

2. Nalpantidis, Lazaros; Gasteratos, A.; Sirakoulis, G.C., "Review of stereo vision algorithms : From software to hardware'', International Journal of Optomechatronics, Vol. 2, No. 4, p. 435-462, 2008

3. Dilip K. Prasad, "Survey of ths problem of object detection in Real images", International journal of image processing, V (6), issue (6), 2012

4. Manjusha, etal., "A survey of image registration", International journal of image processing, V (5), issue (3), 2011

5. Di Stefano, et al., "A fast area-based stereo matching algorithm", Image and vision computing, 22(12), pp. 983-1005, 2004

6. Harkanwal, et al. ," A robust area based disparity estimation technique for stereo vision applications", proceeding of the 2011 International conference on image processing.

7. Meng Chen, et al.,"A method for mobile robot obstacle avoidance based on stereo vision", proceeding 10th IEEE International conference on components, circuits, devices, and systems, 2012

8. Zhao Yong-guo, et al. "The obstacle avoidance and navigation based on stereo vision for mobile robot" proceeding international conference on optoelectronics and image processing, 2010

9. Ibrahim El rube, et al., "Automatic selection of control points for remote sensing image registration based on multi-scale SIFT", proceeding of international conference on signal, image processing, and applications, 2011

10. Ming Bai, et al., "Stereovision based obstacle detection approach for mobile robot navigation", proceeding international conference on intelligent control and information processing, 2010

11. Patrik Kamencay, et al., "Improved depth map estimation from stereo images based on hybrid method", Radio engineering, vol. 21, NO. 1, pp. 70-78, 2012

12. Sukjune Yoon, et al., "Fast correlation-based stereo matching with the reduction of systematic errors", Pattern Recognition Letters 26, 2221-2231, Elsevier, 2005

13. Kanade, T., and M. Okutomi, " A stereo matching algorithm with and adaptive window: Theory and experiment" IEEE Transactions on Pattern Analysis and Machine Intelligence, (TPAMI) 16: 920-932, 1994

14. J. Matas, O. Chum, M. Urban, and T. Pajdla. "Robust wide baseline stereo from maximally stable extremal regions", Proc. of British Machine Vision Conference, pp 384-396, 2002.

15. Lowe, D.G., "Distinctive image feature from scale-invariant keypoints", International Journal of Computer Vision, 60(2): 91-110, 2004
.
16. Hess, R. " An open source SIFT library". Proceedings of the International Conference in Multimedia, 2010, Firenze, Italy. pp. 1493-1496, 2010

17. Herbert Bay, H., A. Ess, T. Tuytelaars, and L. Van Gool. "Speeded-up robust features (SURF)", Computer Vision ECCV 2006, Vol. 3951. Lecture Notes in Computer Science. p. 404-417, 2006

18. Evans, C." Notes on the Open SURF library". UK Publication Papers. Issue 1, Citeseer. p. 25., 2008

19. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#links

20. Luo Juan, Oubong, "A comparision of SIFT, PCA-SIFT, and SURF", International journal of image processing, V (3), issue (4), 2009

21. Young Soo, et al., " Distance estimation using inertial sensor and vision", Inernational Journal of control, Automation, and Systems, V(11), (1), 2013