Arabic words recognition technique for pattern matching using SIFT, SURF and ORB

Syarah Munirah Mohd Zailani[1], Malik Arman Morshidi[2] and Luqman Naim Mohd Esa[3]

[1,2,3] *Department of Electrical and Computer Engineering, Kulliyyah of Engineering*

International Islamic University Malaysia

syarazailani@gmail.com

***Abstract*— Image matching technique requires a robust and fast technique to be applicable in various application. This paper investigates which recognition technique suits better in matching an image of printed Arabic text. The recognition algorithm involves the conventional Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF) and Oriented FAST and Rotated BRIEF (ORB). A parameters estimator of models algorithm is used to weed out the outlier point of matching images. The test demonstrates on the Arabic word images with the different angles, scales, and viewpoints. We evaluate the performance through analyzing the matching accuracy rate and computational time.**

***Index Terms*—Arabic word; LMeds; MSAC; ORB; Pattern matching; RANSAC; SIFT; SURF.**

I. INTRODUCTION

Optical character recognition (OCR) has been widely investigated by researchers since decades ago. As it uses contribute tremendously in easing man activities, it also improves the interaction between human and machine [1]. Nevertheless, few research has been done for Arabic character or word recognition compared to Latin and Chinese character recognition [2].

An online Arabic character was recognized based on the digitized trace of the character on a document [2]. Arabic texts' features contain the information extracted from the printed image. It holds an essential characteristic of the Arabic words which differ it from other words [2]. The pattern matching between two images relies on that unique characteristic to conform the word recognition. As the Arabic word is joint of different Arabic character connectivity, it appears to have unique features compared to normal scene picture.

As this study focuses on pattern matching, feature detector and descriptor play a significant role in the process. The overall process could be divided into three steps; detection, description, and matching [3];

1. Detection: interest points were identified automatically regardless of the viewpoint.
2. Description: generated based on the area surrounding the interest points while possessing a unique description of the features.
3. Matching: based on the predetermined interest points of stored image, the input image feature vector will match the respective object or image.

An ideal feature detector and descriptor are believed to be robust to rotation, scale, noise and affine transformations; and distinctive [4].

The feature extraction of Arabic OCR offers enormous challenges which the characters appear in two main forms: cursive (connected) and isolated form. In general, Arabic alphabets have four different shapes according to their position in the word which are a start, middle, end and isolated [5] as shown in Table 1. These conditions lead to countless potential different forms of character connectivity that are complex, time-consuming and return a poor result for font recognition purpose [6]. As OCR offer enormous challenges cause from the complex connectivity of Arabic characters, it gives many research opportunities to the researchers.

Table 1
Arabic joining groups and group letters, defined with letter location [5].

Characters	Isolated	Start	Middle	End
ا	ا			ـا
ب - ت – ث	ب ت ث	بـ تـ ثـ نـ يـ ئـ	ـبـ ـتـ ـثـ ـنـ ـيـ ـئـ	ـب ـت ـث
ج - ح – خ	ج ح خ	جـ حـ خـ	ـجـ ـحـ ـخـ	ـج ـح ـخ
د – ذ	د ذ			ـد ـذ
ر – ز	ر ز			ـر ـز
س – ش	س ش	سـ شـ	ـسـ ـشـ	ـس ـش
ص – ض	ص ض	صـ ضـ	ـصـ ـضـ	ـص ـض
ط – ظ	ط ظ	طـ ظـ	ـطـ ـظـ	ـط ـظ
ع – غ	ع غ	عـ غـ	ـعـ ـغـ	ـع ـغ
ف	ف	فـ قـ	ـفـ ـقـ	ـف
ق	ق			ـق
ك	ك	كـ	ـكـ	ـك
ل	ل	لـ	ـلـ	ـل
م	م	مـ	ـمـ	ـم
ن	ن			ـن
ه	ه ة	هـ	ـهـ	ـه ـة
و – ؤ	و			ـو ـؤ
ى - ي – ئ	ى			ـى ـي ـئ
ء	ء			
لا	لا			ـلا

The goal of this investigation is to identify a better pattern matching technique of Arabic words without character segmentation by comparing the performance of Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF) and Oriented FAST and Rotated BRIEF (ORB) technique while adopting M-estimator Sample Consensus (MSAC) algorithm to eliminate wrong matching points. This study demonstrates the performance of this technique against rotation, scaling, and viewpoints of Arabic character image. This research is expected to contribute to the growth of research in the Arabic character recognition.

The remainder of this paper is organized as follows.

Section II reviews the prior work done on SIFT, SURF and ORB algorithm. The enhancement of these techniques on the different application is featured. In Section III we briefly describe the approach used in this paper in detail. Section IV presents our analysis results comparing original SIFT, SURF, and ORB with MSAC on image-matching experiments. This paper is concluded in Section V.

II. Prior work

SURF have been used as interest point detector and descriptor which is proven to be invariant to scale and rotation [3]. SURF was inspired by SIFT with a faster computational time using integral image and box filters [7]. SIFT proves to be robust towards scale, rotation, translation and noise occurrence as it tested for Farsi / Arabic automatic font recognition [6]. However, it consumes more computational time than SURF [6], [8]. Inspired by Feature from Accelerated Segment Test (FAST) keypoint detector, Binary Robust Independent Elementary Features (BRIEF) uses binary strings as a feature point detector [9]. The construction and matching showed that it is much faster than SURF and SIFT. In 2011, Rublee et al. proposed an alternative technique to SIFT and SURF, which is ORB. As the name stated, it used FAST interest point detector and BRIEF descriptor. However, since BRIEF is not rotation-invariant descriptor, the author used Rotation-Aware Brief (rBRIEF) [10].

Several enhancements have been made to SIFT, SURF and ORB to make it applicable to different application and condition. In 2012, inspired by Affine Scale Invariant Feature Transform (ASIFT) [11], an algorithm coined Fully Affine InvaRiant SURF (FAIR-SURF) was proposed by Pang et al. in [7]. The algorithm studied take advantage of the affine invariant benefit of ASIFT that deals with view angle problem of SURF; and the efficient merit of SURF algorithm. The experimental outputs prove that FAIR-SURF has much lower complexity than ASIFT by using modified SURF and fixed tilts. Meanwhile in 2013, [12] further improvement on SURF algorithm were able to detect moving object in dynamic scenes by limiting the number of detected feature points and adopt a fast method for calculating the feature point's dominant orientation. The tested outputs show that the improved SURF algorithm manages to increase the speed and improves the precision of the original SURF in [3]. Qin Y. in [13] developed the improved version of ORB by employs SIFT theory on scale space to detect the feature points. Then, a median filter method was used by [14] for more accurate feature points detection while adopting random sample consensus (RANSAC) algorithm and homography matrix to eliminate the wrong matching points. In 2016, Ali et al. in [15] were using SIFT and SURF method as their feature extraction approach for computing descriptor. They aimed to build a system that could convert a digitally editable text file from a printed Nastalique Urdu Script image file coined, Urdu Optical Character Recognition (UOCR). The conversion efficiency reached more than 95% accuracy as it were trained with 23204 UrduNastaleeq ligature. Since Nastalique Urdu Script is similar to Arabic words and character, we believe that SIFT and SURF could also be applied in this experiment.

III. Approach

This research study intended to use SIFT, SURF, and ORB for the pattern matching technique. Then, a parameters estimator algorithm is used to weed out the false matching point. For the parameters estimator tool, we compare the speed and average matched points with MSAC, RANSAC and least median of squares (LMeds).

A. Scale-Invariant Feature Transform (SIFT)

SIFT presents a distinctive invariant features in matching process between different vision range of object or environment [16]. This approach was proven to be able to recognize object among clutter or in the natural scene. The set of image features were generated through these steps [16]:

1. Scale-space extrema detection: Identify location and scales of keypoint in the Difference of Gaussian (DoG) functions which can be computed from the convolution of variable-scale Gaussian, G(x,y,σ) and the input image, I(x,y).

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) = L(x, \quad (1)$$

2. Keypoint localization: the interesting point candidate is distinguished from the low contrast points and poorly localized along the corner.
3. Orientation assignment: based on local image properties, each interest point was assigned a constant orientation.
4. Keypoint descriptor: around each keypoint, the local image descriptor is evaluated based on image gradient and orientation at the certain scale in the region as shown in Figure 1 below. The descriptor is build based on Hough transform.

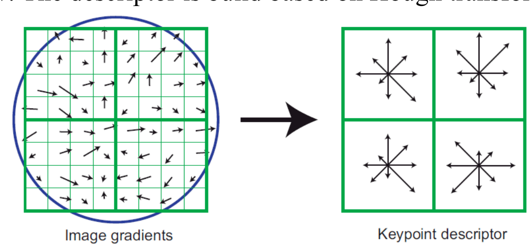


Figure 1: SIFT keypoint descriptor [16]

B. Speeded Up Robust Features (SURF)

SURF algorithm is based on Hessian matrix to find interest point since it has good performance and accuracy. The determinant of Hessian matrix was expressed by the equation below [3].

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

where $L_{xx}(x,\sigma)$ is the convolution of image with the second derivative of Gaussian. The used of integral images and approximated kernels are able to speeded-up the convolutions process and reduce the computation time. It later called 'Fast-Hessian'. Table 2 shows that Fast Hessian detector is faster than Hessian-Laplace, Harris-Laplace and Difference of Gaussian (DoG).

Table 2
Threshold, the number of detected points and calculation time for the detectors compared to Fast-Hessian [3].

Detector	Threshold	No. of Points	Comp. Time (Msec)
Fast-Hessian	600	1418	120
Hessian-Laplace	1000	1974	650
Harris-Laplace	2500	1664	1800
DoG	default	1520	400

SURF descriptor is based on Haar wavelet response and can be computed efficiently with integral images. SURF presents three main steps of the process find the correspondence between two different images of same or similar character [3];

1. Corners, blobs and T-junctions are selected as 'interest points'. The interest point detector repeatability factor plays an important role in detecting points from the distinctive location in the image.
2. Every interest points' neighborhood is signified by a feature vector or also called descriptor.
3. Using a different image, the matching is done by descriptor vector.

C. Oriented FAST and Rotated BRIEF (ORB)

ORB is a result of joining oFAST keypoint detector and rBRIEF descriptor [10]. oFAST is a FAST feature with an orientation component, while rBRIEF is a BRIEF descriptor with rotation aware. The approach used for oFAST is using intensity centroid to ascribe an orientation. The corner's intensity is assumes offset from its center. The moment of patches is defined as:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y), \tag{3}$$

And with this moment of patches we may find the centroid:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right) \tag{4}$$

We can construct a vector from the corner's center, 0, to the centroid. The orientation of the patch then simply is:

$$\theta = atan2\,(m_{01}, m_{10}), \tag{5}$$

where atan2 is the quadrant-aware version of arctan. The rotation invariance is improved by the moments are computed with x and y were stay within a circular area of radius r.

D. Parameters estimator

In an attempt to keep the good matching points of the features, MSAC, RANSAC and LMeds algorithm is used to eliminate false matching points. An investigation was run to test the most suitable algorithm to be used in this study based on the computing time and an average number of features detected in the original images as shown in Figure 2 and Figure 3 respectively.

For the computing time, the pattern matching technique with MSAC shows the fastest processing time compared to with RANSAC and LMeds. MSAC also display the most stable inlier points overall detected compared to MSAC and LMeds. Even though RANSAC and LMeds detect many inlier points especially with SIFT, around half of the points are incorrect matches as the matching accuracy rate is above 100%. Overall, MSAC shows as the most suitable parameters estimator to be used in this study.

LMeds is a robust regression method by adapting least median squares estimator; minimize $med_i\ (y_i - ax_i - b)^2$ replacing least sum of square. The method performs in good condition if at least 50% of the points are matched [17]. While RANSAC stands for random sample consensus which it selects random points over the image pairs and computes the relation between them [18]. The algorithm gets the actual sample by calculating the mathematical model of the data parameter from a set of irregular data [14]. The cluster of strong matches is determined by measuring the sum of squared differences in intensity [19]. The randomized nature of RANSAC algorithm has caused different result obtained for each run. The matching process that based on proximity and similarity only could cause incorrect matches happened [19]. MSAC is the variant of RANSAC which stands for M-estimator Sample Consensus. The M-estimator performs as the quality of the consensus set are evaluate where the sample of data fit a model and a certain set of parameters [19].

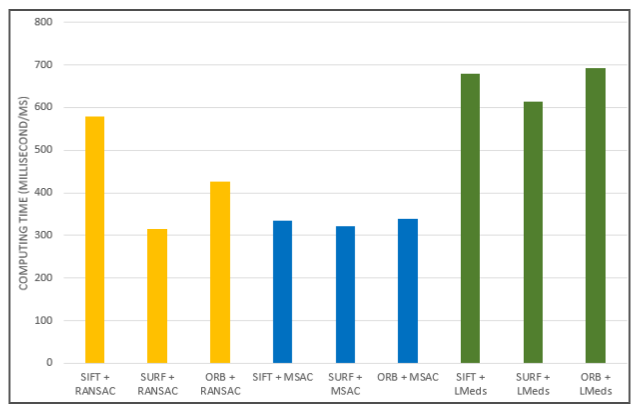


Figure 2: The computing time of SIFT, SURF, and ORB with RANSAC, MSAC, and LMeds.

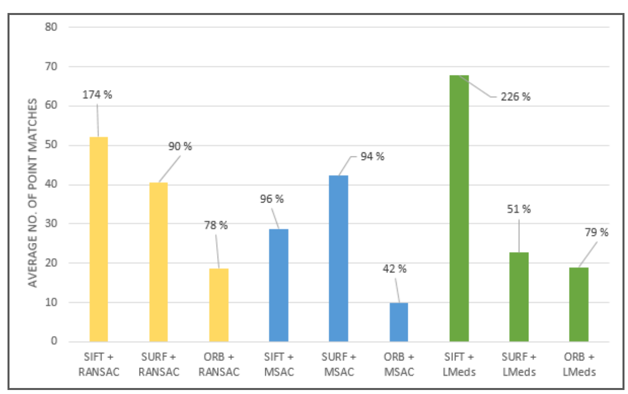
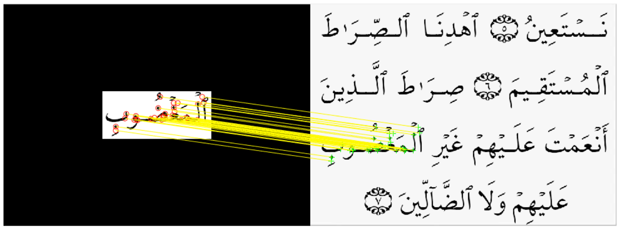
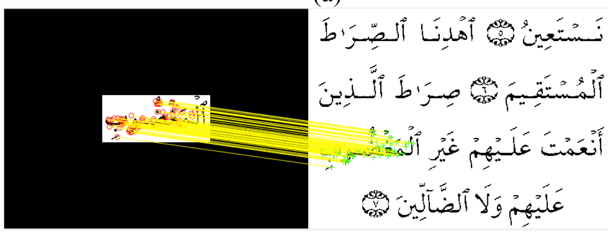


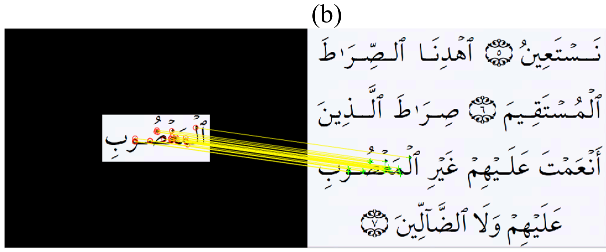
Figure 3: Average number of point matches and the accuracy matching rate for SIFT, SURF, and ORB with RANSAC, MSAC, and LMeds.



(a)



(b)



(c)

Figure 4: The sample matching between original image using (a) SIFT (b) SURF and (c) ORB with M-SAC algorithm

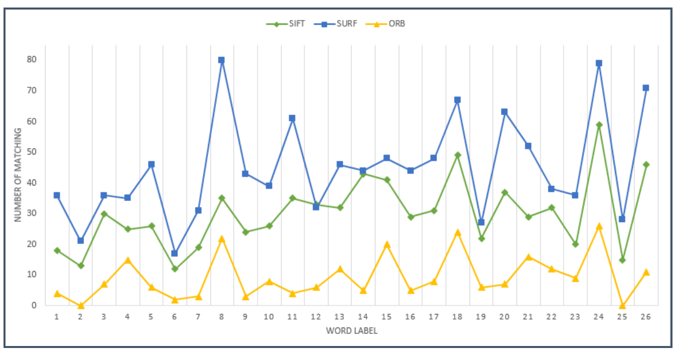


Figure 5: The original matches of word images based on SIFT, SURF and ORB algorithm.

IV. Analysis and Experiment

The three algorithm have been tested on a set of Arabic character image from different angle and scale. The performance evaluation would differentiate the sensitivity of classical SIFT, SURF, and ORB against rotation, scaling and viewpoint. The algorithm used MSAC, which is a variant of RANSAC algorithm, to weed out false matching points. The inliers between two images were found after estimating the geometric transform from matching point pair.

26 different set of Arabic words were used in the experiment as shown in the Appendix. Meanwhile, 100 trials were run for each test. Figure 5 shows the Arabic word matches without any rotation, scale or viewpoint changes. SURF shows more matches than SIFT for almost all words. While ORB presents the least number of matches. The word label 12 and 14 show similar result for SURF and SIFT as they have similar word structure, as label 12 is 'iyyaka' and label 14 is 'waiyyaka'. The performance was evaluated by determining the average matching accuracy rate and computation time regardless of changes. Noting that, for all the experiments, the test is worked on a computer that has 2.40 GHz and 8 GB RAM, with Windows 10 as an operating system.

A. Rotation

We considered 23 value of degree rotation from 15 degrees to 345 degrees to the word images. The test was conducted using SIFT and SURF algorithm only, since ORB unable to give a reliable result. The matching accuracy rate was calculated by dividing the inlier points value of rotated image with the original matches showed in Figure 5.

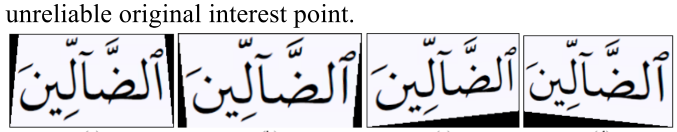
The result shown in Figure 7 is the matching rate for different rotation changes. The rotation angle at 90, 180 and 270 degree shows the highest matching rate for both algorithms with SURF leading the accuracy rate with 96% on that three angle. As 90 degree is the reflex angle of 270 degrees, they produce a similar result. SURF demonstrates a better accuracy rate compared to SIFT despite the rotation changes. In Figure 8, the average computation time for each Arabic word images was displayed. The overall computation time of SURF is faster than SIFT. For word label 6 and 19, the computing time in SIFT is too high compared to the computing time in SURF, it is due to the less of data collected for SURF caused by undetected inlier points.

B. Scaling

In scaling process, we scaled down the word images from 0.9 to 0.8 to observe the effect of scaling to the matching. For 0.9 scale changes, SIFT is shown as the highest matching rate among all three algorithms, while SURF presents the highest accuracy rates for 0.8 scale in Figure 9. Figure 10 shows the average of execution time between the three algorithms. Similar to rotation angle experiment, the scaling test shows that SURF is faster than SIFT. In spite of that, ORB taking the least execution time than the others.

C. Viewpoint

The viewpoint changes were portrayed in four different views; from the above, below, left and right of the Arabic word images as shown in Figure 6. SURF shows the highest accuracy rate for overall viewpoint, while ORB and SURF are shown as the equally lowest matching rate. For down and right viewpoints, ORB presents a better accuracy rate compared to SIFT with 41.19% and 40.63% respectively shown in Figure 11 and Table 3. For the average computing time based on Figure 12, SURF is shown as the fastest computational time compared to ORB and SIFT. Although ORB was believed to be the fastest among other algorithms as mentioned in [10] , ORB shows a poor performance in this viewpoint study. This condition happened due to the extra time is taken in finding the correct interest points for the word images caused by unreliable original interest point.



(a) (b) (c) (d)

Figure 6: The sample of viewpoint changes of Arabic word images; (a) below, (b) above, (c) right and (d) left.

Figure 7: Matching accuracy rate for images with rotation changes.

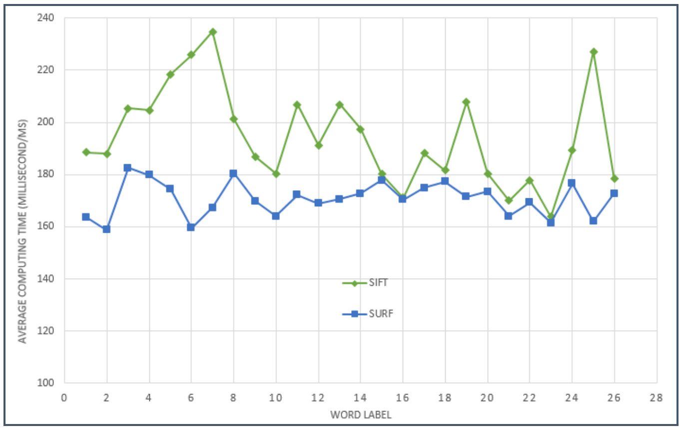


Figure 8: The contrast of average computation time for images despite the rotation changes.

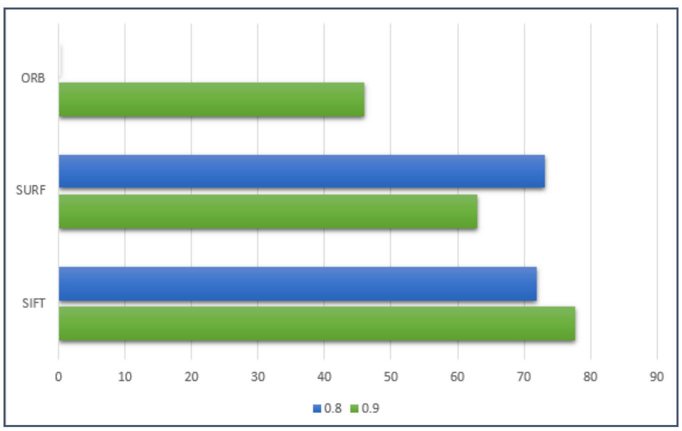


Figure 9: Matching accuracy rate (%) for images with scale changes

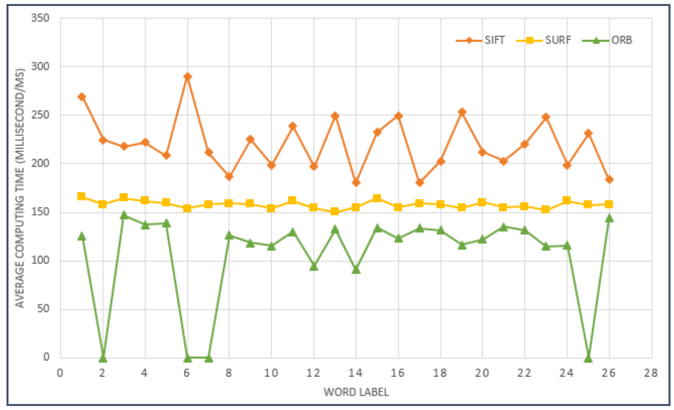


Figure 10: The contrast of average computation time for images despite the scale changes.

Table 3
Average accuracy rate (%) for viewpoint changes

	View			
Algorithm	ABOVE	DOWN	LEFT	RIGHT
SIFT	43.22	37.11	37.58	39.34
SURF	63.15	63.73	62.58	63.96
ORB	33.61	41.19	36.16	40.63

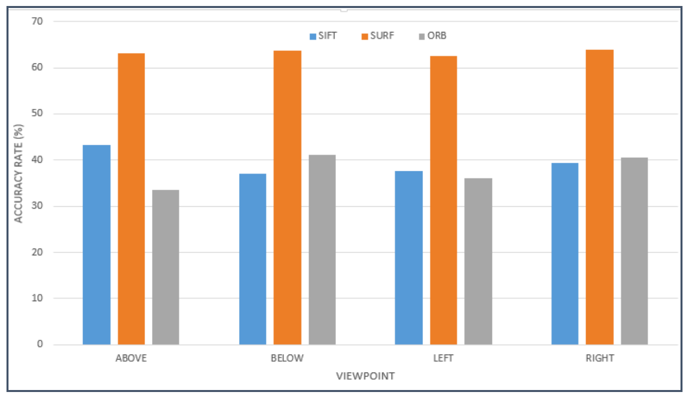


Figure 11: Accuracy rate for images with viewpoint changes.

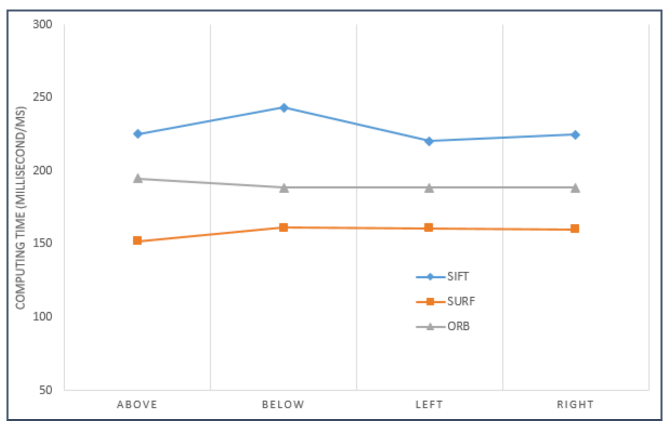


Figure 12: The contrast of average computation time for different viewpoints.

V. CONCLUSION

In this paper, we compared the performance of SIFT, SURF, and ORB with MSAC using the different angle of rotations, scales, and viewpoints on Arabic word images. MSAC shows better results after comparing it to RANSAC and LMeds. Then, for rotation and viewpoint changes study, SURF has shown the highest matching accuracy rate. Meanwhile, in scale changes, SIFT have the most stable accuracy rate. In general, SIFT detects more interest points than SURF and ORB in the original images. However, SURF shows a better accuracy rate compared to another algorithm.

For computing time analysis, ORB is taking the least execution time compared to SURF and SIFT despite the changes in scales. However, SURF appears to have the fastest computing time for different viewpoints of Arabic word images due to the unreliable interest point detected in ORB. For further improvement, the experiment could be tested on other new detector and descriptor; Accelerated KAZE (AKAZE) and Learned Arrangements of Three Patch Codes (LATCH), to observe the effect on the distorted Arabic word images.

APPENDIX

1	بِسۡمِ	10	يَوۡمِ	19	ٱلَّذِينَ
2	ٱللَّهِ	11	ٱلدِّينِ	20	أَنۡعَمۡتَ
3	ٱلرَّحۡمَٰنِ	12	إِيَّاكَ	21	عَلَيۡهِمۡ
4	ٱلرَّحِيمِ	13	وَإِيَّاكَ	22	غَيۡرِ
5	ٱلۡحَمۡدُ	14	نَسۡتَعِينُ	23	ٱلۡمَغۡضُوبِ
6	لِلَّهِ	15	ٱهۡدِنَا	24	وَلَا
7	رَبِّ	16	ٱلصِّرَٰطَ	25	نَعۡبُدُ
8	ٱلۡعَٰلَمِينَ	17	ٱلۡمُسۡتَقِيمَ	26	ٱلضَّآلِّينَ
9	مَٰلِكِ	18	صِرَٰطَ		

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the Ministry of Higher Education Malaysia for supporting this research under Research Acculturation Grant Scheme (RAGS) Ref. Code (RAGS/1/2015/ICT01/UIAM/02/1). In addition, we would like to thank the Kulliyyah of Engineering, International Islamic University Malaysia (IIUM), for giving us the opportunity to complete the research in IIUM.

REFERENCES

[1] O. B. Ali, A. Shaout, and M. Elhafiz, "Two stage classifierfor Arabic Handwritten Character Recognition," Ijarcce, vol. 4, no. 12, pp. 646–650, 2015.

[2] A. Lawgali, "A Survey on Arabic Character Recognition," vol. 8, no. 2, pp. 401–426, 2015.

[3] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3951 LNCS, pp. 404–417, 2006.

[4] E. Karami, S. Prasad, and M. Shehata, "Image Matching Using SIFT , SURF , BRIEF and ORB : Performance Comparison for Distorted Images," Newfoundl. Electr. Comput. Eng. Conf. IEEE, Newfoundl. Labrador Sect., no. February 2016, 2015.

[5] A. AbdelRaouf, C. A. Higgins, T. Pridmore, and M. I. Khalil, "Arabic character recognition using a Haar cascade classifier approach (HCC)," Pattern Anal. Appl., vol. 19, no. 2, pp. 411–426, 2016.

[6] M. Zahedi and S. Eslami, "Farsi/Arabic optical font recognition using SIFT features," Procedia Comput. Sci., vol. 3, pp. 1055–1059, 2011.

[7] Y. Pang, W. Li, Y. Yuan, and J. Pan, "Fully affine invariant SURF for image matching," Neurocomputing, vol. 85, pp. 6–10, 2012.

[8] P. M. Panchal, S. R. Panchal, and S. K. Shah, "A comparison of SIFT and SURF," Int. J. Image Process., vol. 1, no. 2, pp. 323–327, 2013.

[9] M. Calonder et al., "BRIEF: Computing a Local Binary Descriptor very Fast.," IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 7, pp. 1281–1298, 2011.

[10] E. Rublee, W. Garage, and M. Park, "ORB : an efficient alternative to SIFT or SURF," pp. 2564–2571, 2011.

[11] J.-M. Morel and G. Yu, "ASIFT: A New Framework for Fully Affine Invariant Image Comparison," SIAM J. Imaging Sci., vol. 2, no. 2, pp. 438–469, 2009.

[12] J. Pan, W. Chen, and W. Peng, "A new moving objects detection method," in 25th Chinese Control and Decision Conference (CCDC), 2013, pp. 901–906.

[13] Y. Qin, H. Xu, and H. Chen, "Image feature points matching via improved ORB," PIC 2014 - Proc. 2014 IEEE Int. Conf. Prog. Informatics Comput., no. 1, pp. 204–208, 2014.

[14] L. Yu, Z. Yu, and Y. Gong, "An Improved ORB Algorithm of Extracting and Matching Features," Int. J. Signal Process. Image Process. Pattern Recognit., vol. 8, no. 5, pp. 117–126, 2015.

[15] T. Ali, T. Ahmad, and M. Imran, "UOCR A ligature based approach for Urdu OCR system," in International Conference on Computing for Sustainable Global Development (INDIACom), 2016, pp. 388–394.

[16] D. G. Lowe, "Distinctive image features from scale invariant keypoints," Int'l J. Comput. Vis., vol. 60, pp. 91–11020042, 2004.

[17] D. L. Massart, L. Kaufman, P. J. Rousseeuw, and A. Leroy, "Least median of squares: a robust method for outlier and model error detection in regression and calibration," Anal. Chim. Acta, vol. 187, no. C, pp. 171–179, 1986.

[18] M. a Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with," Commun. ACM, vol. 24, pp. 381–395, 1981.

[19] P. H. S. Torr and A. Zisserman, "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry," Comput. Vis. Image Underst., vol. 78, no. 1, pp. 138–156, 2000