Feature-based Video Stabilization using Gabor Wavelets

Wan Nural Jawahir Hj Wan Yussof[1], Muhammad Suzuri Hitam[1], Abdul Aziz K. Abdul Hamid[1] and Ezmahamrul Afreen Awalludin[2]

[1]School of Informatics and Applied Mathematics,
[2]School of Fisheries and Aquaculture Sciences,
Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia.
wannurwy@umt.edu.my

Abstract—**This study proposes a method to stabilize jittery video using a feature-based technique. Our feature-based technique extracts local image features using Gabor wavelets. Firstly, to locate a set of interest points within a video frame, we detect some local maxima on Gabor response map image. Then, using the same Gabor response map image, we compute relational features around these interest points. The method was tested using shaky car video obtained from MATLAB version 2011b and compared with the SIFT and SURF methods. The output of using the proposed local image features is comparable to the output produced by SIFT and SURF methods and has shown good result concerning stabilization and discarded distortion from the output video.**

Index Terms—**Gabor Wavelets; Local Image Features; Relational Features; Video Stabilization.**

I. Introduction

Image capturing devices using hand-held cameras such as digital camera, camcorder, smartphone and tablets are becoming popular today thanks to low-ended price and reduced size. However, shooting videos with a hand-held camera normally will lead to unanticipated effects, which incontrovertibly reduces video quality. To compensate for the above problem, many researchers use digital video stabilization [1-9]. Digital video stabilization technique removes undesired motions due to camera shaking or jiggling.

Normally, there are pixel-based and feature-based approaches in digital video stabilization. The pixel-based approach uses pixel intensity directly, while feature-based uses local image features [1-9]. An approach of using local image features is conceivably better than using the pixel-based approach on account of their promising performance especially regarding distinctiveness; yet, they are invariant to many kinds of geometric and photometric transformation.

Roughly speaking, the generation of local image features is commonly a two-part process. The first process requires an interest point detector to select points within the image that are located at visually distinct patches. The second process generates a description of the region around the point. The data produced in these two processes consists of the location and description of the image patches around the interest points.

Recently, many different techniques of detectors and descriptors for describing local image features have been developed, and it was shown that the Scale Invariant Feature Transform (SIFT) [10] is the most appealing local image features for practical uses. Speeded Up Robust Feature (SURF) is a local image feature that has been designed by [11] as an efficient alternative to reduce computational burden in SIFT. In literature, it has been shown these two methods are widely used for video stabilization as well [6-9]. Although SIFT and SURF have made significant progress, they are not really invariant to illumination changes which limit their applicability. This paper introduces a new Gabor-based local image feature to overcome the limitation. Gabor wavelets provide multi-channel, frequency and orientation filtering that is similar to the visual image formed on the retina which is performed by the brain. A complementary between physics and biological vision has shown the Gabor wavelets successfully accounts for many of the vision applications [12-15].

The remainder of this paper is organized as follows. Section 2 describes the proposed local image features. Section 3 presents the video stabilization algorithm. Section 4 presents and discusses the result, and finally, Section 5 concludes the work.

II. The Proposed Local Image Features

As mentioned in the previous section, the proposed local image features are based on using Gabor wavelets to detect points and extract features. This section is divided into two parts. The first part describes the use of Gabor wavelets for detecting interest points and the second part shows how the descriptor is calculated.

A. Interest Points using Gabor Wavelets

Spatially, Gabor filter is a product of a Gaussian function and a complex sinusoidal given by:

$$\Psi(z,f,\theta) = \frac{||k_{f,\theta}||^2}{\sigma^2} e^{||k_{f,\theta}||^2||z||^2} [e^{jk_{f,\theta}||z||^2} e^{-\sigma^2/2}] \quad (1)$$

where z is the image location at (x,y) and σ is the spatial width of the Gaussian filter. $k_{f,\theta}$ is the filter wave-vector given by $k_{f,\theta} = fe^{j\theta}$, with f describes the frequency of sinusoidal plane wave and θ is the anti-clockwise rotation of the Gaussian envelope.

A filter response $\xi(z;f,\theta)$ can be calculated at any location z with the convolution between Gabor filter in Eq. (1) and an image $I(z)$ as follows:

$$\xi(z;f,\theta) = \Psi(z;f,\theta) * I(z) \quad (2)$$

The common method for reducing the computational cost of the above operation is to perform the convolution in Fourier space. This way, the operation is done based on simple element-wise multiplication with linear time complexity:

$$\xi(z; f, \theta) = F^{-1}\{F\big(\Psi(z; f, \theta)\big) * F\big(I(z)\big)\}, \qquad (3)$$

where F denotes the fast Fourier transform and F^{-1} is its inverse.

The interesting part of Gabor filter is that it can be represented with several filters in different orientations and frequencies which then called a bank of Gabor filters or Gabor wavelets. Gabor wavelets can be obtained by varying the orientations:

$$\theta_v = \frac{\pi v}{V} \qquad \forall v = \{0, \ldots, V-1\}, \qquad (4)$$

and determine different frequencies with

$$f_u = \frac{f_{max}}{\lambda^u} \qquad \forall u = \{0, \ldots, U-1\}, \qquad (5)$$

Using the parameter selection in Eq. (4) and Eq. (5) to cover frequencies of interest $f_0, \ldots, f_{U-1}$ and the orientations $\theta_0, \ldots, \theta_{V-1}$, Gabor features GF can be represented in matrix form $GF = \{\xi_{u,v}\}$ as follows:

$$GF = \begin{pmatrix} \xi_{0,0} & \cdots & \xi_{0,V-1} \\ \vdots & \ddots & \vdots \\ \xi_{U-1,0} & \cdots & \xi_{U,V} \end{pmatrix}. \qquad (6)$$

Gabor features in Eq. (6) are combined to produce a single response map as in the following equation:

$$\xi_I(z) = \frac{\sum_{v=0}^{V-1}\sum_{u=0}^{U-1} \Psi(z; f_u, \theta_v) * I(z)}{U * V}. \qquad (7)$$

The proposed detector obtains a set of interest points by applying a non-maximum suppression (NMS) on a response map ξ_I. To do this, ξ_I is dilated by performing a grayscale morphological dilation as expressed in the following equation:

$$[\xi_I \oplus b](z) = \max_{(s,t)\in b}\{\xi_I(x-s, y-t)\} \in [\varepsilon_1, \varepsilon_2], \qquad (8)$$

where b is the structuring element with the size $2r+1$ and r is the radius considered in NMS. Local maxima are extracted by finding the points that match the dilated image with the threshold values in the range $[\varepsilon_1, \varepsilon_2]$. The threshold value, ε_2 must be greater than threshold value ε_1 and must be assigned in the interval $\{0,1]$ whereas ε_1 must be assigned in the interval $[0,1\}$. The number of detected points will vary by a combination of threshold adjustment.

B. Image Descriptor using Relational Features

At this point, a set of interest points has been obtained. Now, the extraction of image features must be done on these interest points. Our descriptor is built based on the idea proposed by [16]. In the calculation of the descriptor, two circular neighbourhoods are used that consists of inner circular and outer circular. Let (x, y) is the interest point under consideration, the inner circular is represented as (x_1^i, y_1^i):

$$(x_1^i, y_1^i) = x + r_1 \cos\theta_i, y + r_1 \sin\theta_i, \qquad (9)$$

and the outer circular is represented as (x_2^i, y_2^i):

$$(x_2^i, y_2^i) = x + r_2 \cos\theta_i, y + r_2 \sin\theta_i. \qquad (10)$$

From the equations defined in Eq. (9) and Eq. (10), r_2 is set to $2r_1$. The θ_i value is given by $i.2\pi/N, \forall i = 1, \cdots, N$ where N is a number of neighborhood points. We calculate the descriptor on a response map as in Eq. (7) using relational features that is defined as follows:

$$RG = \frac{\sum_{i=0}^{N} rel(\xi(x_2^i, y_2^i) - \xi(x_1^i, y_1^i))}{N} \qquad (11)$$

where function rel is given by:

$$rel(x) = \frac{1}{1 + \exp(-x)} \qquad (12)$$

The illustration of the proposed features descriptor is presented in Figure 1. As shown in the figure, a red node represents the reference point (x, y). Green nodes and blue nodes represents inner circular neighborhoods and outer circular neighborhood. The relational features are formed by applying the relational function on the difference of the neighboring pixels lying on specific distance which are shown by red lines and angle to the interest point (i.e. center of the circles.) In case of points that are not lying exactly on image grid, bilinear interpolation will be performed.

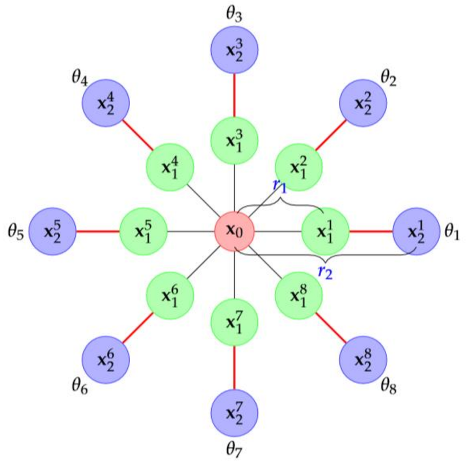


Figure 1: The two 8-neighborhoods of the proposed relational features

The result using Eq. (11) is a single value representing information on one interest point. This kind of feature is not distinctive enough as one point might have the same value to the other points. More features on one interest point could be generated when RG is extended by considering θ with the phase-shift φ. Thus, Eq. (10) becomes:

$$(x_2^i, y_2^i) = x + r_2 \cos(\theta_i + \varphi), y + \sin(\theta_i + \varphi). \qquad (13)$$

By varying the φ values, a set of RG features can be obtained. However, one can systematically set different values of φ as follows:

$$\varphi_j = \frac{j.(\theta_{i+1}-\theta_i)}{M} \quad \forall j = 0,\cdots,M-1 \qquad (14)$$

where M is the total number of φ used.

III. Video Stabilization Algorithm

In the course of demonstration video stabilization, we perform the following algorithm shown in Figure 2 to a video that is shared in MATLAB® Computer Vision System Toolbox version 2011b that named as “shaky_car.avi”. The red boxes indicate where the proposed features are generated.

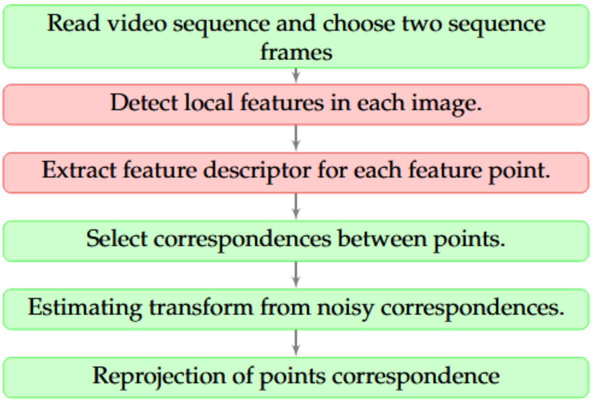
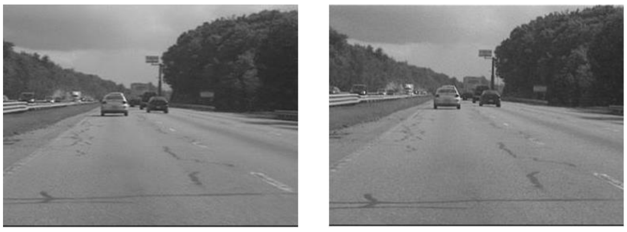

Read video sequence and choose two sequence frames
Detect local features in each image.
Extract feature descriptor for each feature point.
Select correspondences between points.
Estimating transform from noisy correspondences.
Reprojection of points correspondence

Figure 2: The pipeline of feature-based video stabilization

A. *Input frames*

As usual, to perform any computer vision tasks, the input image(s) must be provided. In this algorithm, the first two frames from "shaky_car.avi" are extracted and read them as grayscale images. The use of grayscale images is meant to improve the speed of the algorithm. In Figure 3, the two frames are shown side by side with the first frame on the left, and the other frame is on the right.



(a) Frame A (b) Frame B

Figure 3: Images from the first two frames of a video sequence.

Then, to illustrate the pixel-wise difference between them, a cyan-yellow composite image is produced as shown in Figure 4. There is obviously a large vertical and horizontal offset between the two frames.

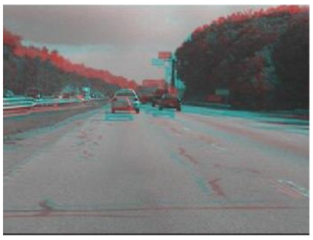
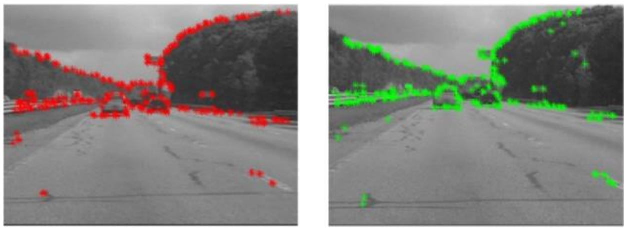


Figure 4: Colour composite between frame A (red) and frame B (cyan).

B. *Features Extraction*

The goal of this demonstration is to determine a transformation that will correct the distortion between the two frames. As input, a set of point correspondences between two frames must be provided. The correspondences are generated from both frames using the proposed features. Figure 5 shows the detected points from both frames. As observed from this figure, the proposed method detects the same image features in both frames such as points around the cars, points along the tree line and the corners of the road.



(a) Points in Frame A (b) Points in Frame B

Figure 5: Points detection using the proposed points detector.

C. *Select Correspondences between Points*

In this step, the descriptors of each point are compared between the two frames using Lowe’s method [10] for the purpose of selecting correspondences between the points derived above. The image in Figure 6 is composited image between frame A and frame B in Figure 5. The yellow line shows the correspondences obtained after applying the procedures. As noticed in Figure 6, many of these correspondences are correct, but there is also a significant number of outliers.

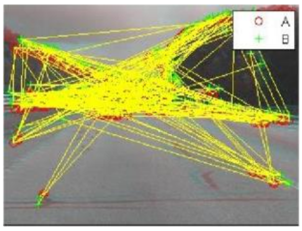


Figure 6: Matched points.

D. *Estimation transform from noisy correspondences*

In this step, we derive a robust estimate of the geometric transform between the two images using the RANSAC algorithm [17]. RANSAC searches for the valid inlier correspondences from a set of point correspondences [18]. The purpose is to derive the projective transformation that makes both inliers in the first and second set of points, a perfect match to one another. Figure 7 shows a colour composition of frame A overlaid with the reprojected frame B. The results are excellent, with the cores of the images are both well aligned.

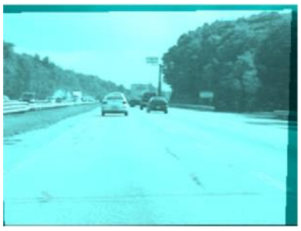
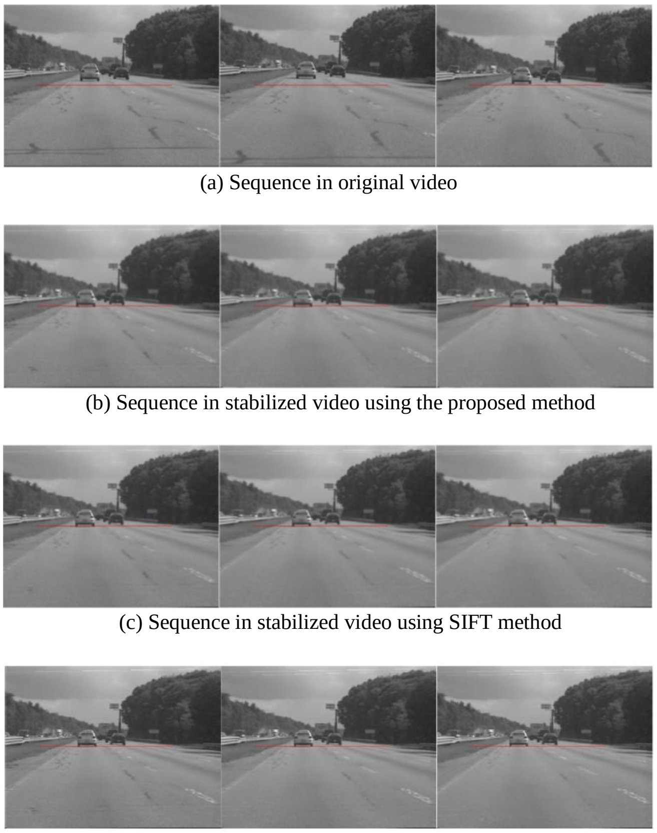


Figure 7: Transformed image.

IV. Results And Discussion

In this section, the quality of the stabilized video using the proposed method is compared with SIFT and SURF methods. The comparison is made with these two methods because they can be regarded as the most powerful methods in the literature. We conduct the experiment on the first four frames of "shaky_car.avi" video that is available in MATLAB® software. The earlier frame is used as a reference for stabilized image in the latter frame. For example, for the first two frames, the first frame acts as a reference frame for stabilizing the second one.

Figure 8 displays Frame#2, Frame#3 and Frame#4 of the original and the corresponding frames from the stabilization using the proposed method, SIFT method and SURF method, respectively. By checking the fixed reference red lines overlaid on the image in Figure 8, it is obvious that the stabilized video is now more stable than the original video. The result of the proposed stabilization video is quite similar to the results produced by SIFT and SURF methods.



(a) Sequence in original video

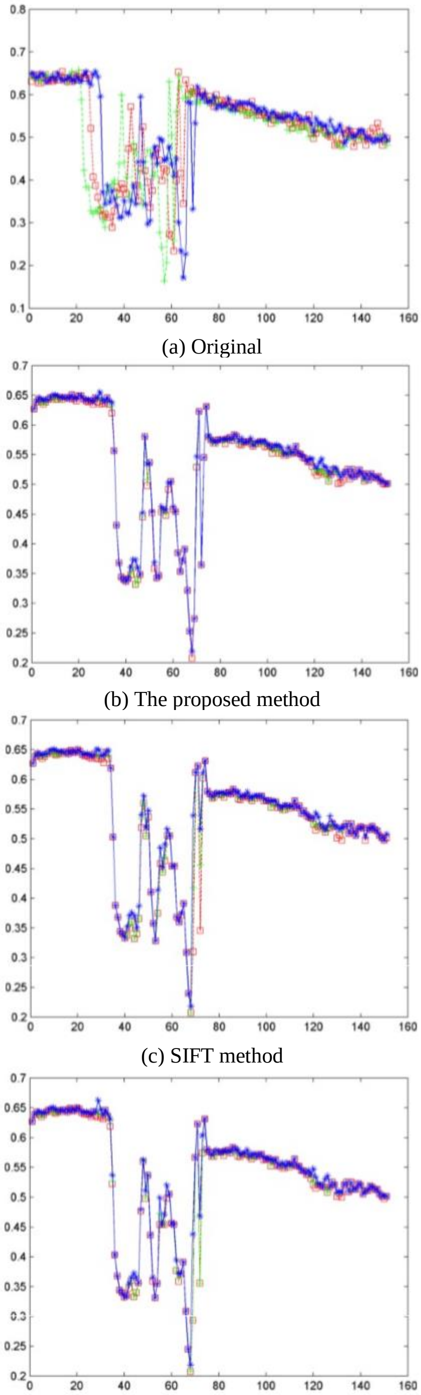
(b) Sequence in stabilized video using the proposed method

(c) Sequence in stabilized video using SIFT method

(d) Sequence in stabilized video using SURF method

Figure 5: Comparison of different stabilization results.

For further investigation, the pixel-value across sections along red line segments are computed and presented in Figure 9. The red, green and blue plots indicate profile from Frame#2, Frame#3 and Frame#4, respectively. As noticed from this figure, the plots of the original video have different profile patterns whereas the proposed method, SIFT method and SURF method produce profile patterns quite similar to each other. By looking at profile plots of the original frames, it is clear that the original video is not stable as the profile of Frame#3 shifts a little bit to the left from Frame#2 while the profile of Frame#4 suddenly shifts to the right from its previous frames. However, after performing a stabilization process, the profile of Frame#3 and Frame#4 are now similar to Frame#2. While there are some pixel value differences between position 60 and 80 along the horizontal direction of the SIFT method, this also happens to SURF method. However, the proposed method provides quite similar pixel values across frames. The consistent pixel across frames indicates that the proposed method provides the more stable result as compared to its counterparts.



(a) Original

(b) The proposed method

(c) SIFT method

(d) SURF method

Figure 9: Comparison of profile plots by different methods.

V. Conclusion

To summarize the paper, the proposed feature-based video stabilization method has been demonstrated to stabilize a jittery video. The result showed that the proposed method could align between frames very well similar to what has been obtained by SIFT and SURF methods. Thus, this study has proven that the proposed method provides a great deal of stabilization and applicable to other applications. For future work, we aim at boosting the computational performance of the method using GPU since it is known that computation using GPU can be substantially faster than using CPU.

ACKNOWLEDGMENT

We like to acknowledge Ministry of Higher Education under Fundamental Research Grants Scheme for funding this study.

REFERENCES

[1] G. Spampinato, A. R. Bruna, I. Guarneri, and V. Tomaselli, “Advanced feature based digital video stabilization,” in *6th International Conference on IEEE Consumer Electronics-Berlin (ICCE-Berlin),* 2016, pp. 54-56.

[2] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato, “SIFT features tracking for video stabilization,” in *14th International Conference on Image Analysis and Processing, ICIAP 2007,* 2007, pp. 825-830.

[3] C. Song, H. Zhao, W. Jing, and H. Zhu, “Robust video stabilization based on particle filtering with weighted feature points,” *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 570-577, May 2012.

[4] J. Xu, H. W. Chang, S. Yang, and M. Wang, “Fast feature-based video stabilization without accumulative global motion estimation,” *IEEE Transactions on Consumer Electronics*, vol. 58, no 3, pp. 993-999, 2012

[5] M. Okade and P. K. Biswas, “Video stabilization using maximally stable extremal region features,” *Multimedia Tools and Applications*, vol. 68, no. 3, pp.947-968, 2014.

[6] A. Walha, A. Wali, and A. M. Alimi, “Video stabilization for aerial video surveillance,” *AASRI Procedia*, vol. 4, pp.72-77, 2013.

[7] X. Zheng, C. Shaohui, W. Gang, and L. Jinlun, “Video stabilization system based on speeded-up robust features,” in *Proc. Int. Industrial Informatics and Computer Engineering Conf.*, 2015, pp. 1996-1998.

[8] M. M. Hossain, H. J. Lee, and J. Lee, “Fast image stitching for video stabilization using sift feature points,” *The Journal of Korea Information and Communication Society*, vol. 39, no. 10, pp.957-966, 2014.

[9] Y. H. Chen, H. Y. S. Lin, and C. W. Su, “Full-frame video stabilization via SIFT feature matching,” in *2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP),* 2014, pp. 361-364.

[10] D. G. Lowe, “Distinctive Image features from scale-invariant keypoints” *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[11] H. Bay, T. Tuytelaars, and L. Gool, “Surf: Speeded up robust features,” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008.

[12] Z. Chai, Z. Sun, H. Mendez-Vazquez, R. He, and T. Tan, “Gabor ordinal measures for face recognition,” *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp.14-26, 2014.

[13] F. Riaz, A. Hassan, S. Rehman, and U. Qamar, “Texture classification using rotation and scale-invariant Gabor texture features,” *IEEE Signal Processing Letters*, vol. 20, no. 6, pp. 607-610, 2013.

[14] Qian, Y., M. Ye, and J. Zhou, “Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 4, pp.2276-2291, 2013.

[15] S. Agarwal, A. K. Verma, and N. Dixit, “Content-based image retrieval using color edge detection and discrete wavelet transform,” in *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT),* 2014, pp. 368-372.

[16] M. Schael, “Texture defect detection using invariant textural features,” in *Joint Pattern Recognition Symposium,* Berlin Heidelberg: Springer, 2001, pp. 17-24.

[17] M.A. Fischler, and R.C.Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 1981.

[18] G. Shi, X. Xu, and Y. Dai, “SIFT feature point matching based on improved RANSAC algorithm,” in *2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC),* 2013, pp. 474-477.