Research Article

Ring Fusion of Fisheye Images Based on Corner Detection Algorithm for Around View Monitoring System of Intelligent Driving

Jianhui Zhao,^{1,2} Hongbo Gao ,³ Xinyu Zhang ,⁴ Yinglin Zhang,⁵ and Yuchao Liu¹

¹Department of Computer Science and Technology, Tsinghua University, Beijing 100083, China

²Department of Basic Courses, Army Military Transportation University, Tianjin 300161, China

³State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100083, China

⁴Information Technology Center, Tsinghua University, Beijing 100083, China

⁵State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410000, China

Correspondence should be addressed to Hongbo Gao; ghb48@mail.tsinghua.edu.cn

Received 16 September 2017; Revised 16 December 2017; Accepted 3 January 2018; Published 1 February 2018

Academic Editor: Chenguang Yang

Copyright © 2018 Jianhui Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to improve the visual effect of the around view monitor (AVM), we propose a novel ring fusion method to reduce the brightness difference among fisheye images and achieve a smooth transition around stitching seam. Firstly, an integrated corner detection is proposed to automatically detect corner points for image registration. Then, we use equalization processing to reduce the brightness among images. And we match the color of images according to the ring fusion method. Finally, we use distance weight to blend images around stitching seam. Through this algorithm, we have made a Matlab toolbox for image blending. 100% of the required corner is accurately and fully automatically detected. The transition around the stitching seam is very smooth, with no obvious stitching trace.

1. Introduction

In the past decades, because of the rapid growth of road transportation and private cars, road traffic safety has become an important problem in society [1–3]. The statistics results of the national traffic accidents show that the proportion of accidents caused by drivers' vision limitation, the reflection of delay, judgment error, and improper operation accounted for up to 40% of the total accidents [3–5]. In order to solve the above problem, the advanced driving assistance system (ADAS) received more and more attention, such as lane departure warning system (LDWS), pro-collision warning system (FCWS), blind spot monitoring (BSD), and around view monitor (AVM) [6, 7]. Among them, AVM is used to provide the driver with the 360-degree video image information around the body, in the parking, and in crowded city traffic conditions to reduce the user's visual blind area, to help users to better judge the road traffic conditions around

the vehicle, to avoid collision with pedestrians and vehicles around, and to make the driving process safer and more convenient [8–10].

The key technology of AVM is fisheye image correction and image fusion. In this paper we focus on image fusion, which includes image registration and blending. There are three main types of registration methods, which are region matching method, transform domain based method, and feature matching method. Among them, feature based registration method is fast and accurate, and some features are robust to image deformation, illumination change, and noise. It is a common method of image registration. In the document, the overlap points of two correction images are extracted by matching the SFIT features with scale and direction invariance [11, 12]. The feature operators usually used to extract overlapping corner points include Harris [13, 14], Canny [15, 16], and Moravec [17, 18]. These feature operators are mainly used in the registration process of

general image mosaics. However, in AVM system, we match images by scene calibration. Documents [19, 20] detect the corner points of checkerboard patterns by quadrilateral join rules. In documents [21, 22], the initial corner set is obtained by the improved Hessian corner detector, and the initial concentration of false points is eliminated by the intensity and geometrical features of the chessboard pattern. But the above two methods are mainly designed for the chessboard pattern used in camera calibration, which do not meet the requirement of calibration pattern used in scene calibration. Due to the influence of inherent difference of camera and the installation position, there is a difference of exposure between the images, which leads to the obvious stitching trace. Therefore, the image blending is necessary after registration. In the documents, the optimal seam is searched by minimizing the mean square variance of pixel in the region, and the adjacent interpolation method is used to smooth the stitching effect, but this method is not suitable for the scene with too large difference of exposure [23, 24]. The documents fuse several images captured by the same camera with different parameters from the same angle of view by weighting method, but this method can only adjust the brightness and cannot reduce the color difference [25, 26]. In the documents, seamless stitching is achieved by tone compensation near the optimal seam, but the stitching seam of around view system is fixed, so the AVM image cannot be fully fused by this method [27, 28].

Therefore, in order to fully consider the needs of scene calibration, we propose an integrated corner detection method to automatically detect all corners in the process of image registration. In order to fully consider the influence of inherent difference of camera and the installation position, we propose ring fusion method to blend images from 4 fisheye cameras. The main contribution of this paper lies in the following aspects: (1) limitation condition of minimum area and shape are used to remove redundant contours for corner detection successfully. We also improve the corner positions extraction accuracy, by detecting corners in fisheye images at first and calculating the corresponding position in corrected image. (2) The color matching method and ring shape scheme are used in image blending for a more smooth transition successfully, which makes it possible to seamlessly fuse images with large differences in exposure. (3) A Matlab toolbox for image blending in AVM system is designed.

The rest of this paper is organized as follows: Section 2 introduces AVM architecture. Section 3 describes the methodology of image registration and blending in detail. Section 4 describes the experiment result of our method. Conclusions are offered in Section 5.

2. AVM Architecture

The algorithm flow of the AVM is shown in Figure 1. Firstly, we input fisheye images of the calibration scene and detect corner points. Then, positions of these corner points in corrected images are calculated by a correction model. Meanwhile, we use Look Up Table (LUT) to correct fisheye images and obtain the corrected images. Secondly, the target positions of corner points in output image are calculated

FIGURE 1: Flow chart of image fusion.

by size data in calibration scene. Then, positions of corner points in corrected images and their target positions are used to compute homography matrix H . Finally, we project corrected images into the coordinate of output image by using homography matrix H . Then we use ring fusion method to blend them, which is the emphasis of this paper.

In our experiment, the Volkswagen Magotan has been used. The length of the vehicle is 4.8 m, and the width is 1.8 m. We use fisheye camera with a 180-degree large view angle, with a focal length of 2.32 mm. 4 fisheye cameras are mounted on the front, back, left, and right sides of the vehicle separately. The size of the image captured by fisheye camera is 720×576 . The size of AVM output image is 1280×720 . This paper develops the proposed method on a PC. The adopted simulation processor is the Intel(R) Core(TM) i7-6700HQ CPU at 2.60 GHz and the simulation software is MATLAB.

3. Methodology

3.1. Scene Calibration. Calibration scene is set up for image registration in next step. The distance between vehicle body and calibration pattern in the front and rear positions is 30 cm and the right and left is 0 cm. The reference point of front pattern is A, and the rear is F. We made the point A collinear with the left vehicle body and F collinear with the right vehicle body. And there are 12 point positions we need in every view angle, as shown in Figure 2.

The size data which need to be measured include the following:

- (1) Car length: the line length of AE.
- (2) Car width: the line length of AB.
- (3) Offset: the line length of AC or the line length of BD.

FIGURE 2: The illustration of calibration scene.

After the measurement of size data, the target positions of corner points in the coordinate of output image are calculated by the following parameters: the size data measured above, the size of output image defined by users, and the size of calibration pattern and vehicle. The calculation process of the target position of all points is the same. We take the target position of point 1 (as shown in Figure 1) as an example. Firstly, we calculate the position of point 1 in the calibration scene, as shown in

$$\begin{aligned} x_1 &= -\frac{1}{2}W - w_w - w_{b1} \\ y_1 &= -\frac{1}{2}L - w_w - w_{b1}, \end{aligned} \quad (1)$$

where the origin of calibration scene is located at the center of the calibration scene, as shown in Figure 1. (x_1, y_1) denotes the position of point 1, W denotes the vehicle width, L denotes the vehicle length, w_w is the white edge width, and w_{b1} is the width of big black box.

Secondly, we use the position in calibration scene to calculate the position in the coordinate of output image, as shown in

$$\begin{aligned} \text{scale} &= \frac{W_{\text{img}}}{W_{\text{real}}} \\ u_1 &= \text{scale} * x_1 + \frac{W_{\text{img}}}{2} \\ v_1 &= \text{scale} * y_1 + \frac{L_{\text{img}}}{2}, \end{aligned} \quad (2)$$

where scale denotes the scaling factor from calibration scene to coordinate of output image. W_{img} denotes the width of output image. W_{real} denotes the width of calibration scene.

(u_1, v_1) denotes the position in calibration of output image. (x_1, y_1) denotes the position in calibration scene.

3.2. Image Registration Based on Corner Detection

3.2.1. Detect and Calculate the Corner Point Positions. Firstly, the corners are detected automatically in the fisheye image by the integrated corner detection method. Secondly, the corresponding positions of these corners in the corrected image are calculated using the correction model. Finally, we save the positions in the corrected image for the next computation of homography matrix.

Algorithm steps of integrated corner detection method are as follows:

- (1) Input fisheye images of calibration scene from all 4 cameras.
- (2) Use the Ruffli corner detection method to detect the corners in the chessboard array.
- (3) Based on the relative position between the black box and the chessboard array, use the detected corners from step (2) to obtain the Region Of Interest (ROI) of the big black box.
- (4) Preprocess the ROI by adaptive binarization using “adaptiveThreshold” function in Opencv and a morphological closing operation to denoise.
- (5) Obtain the contour of big black box from ROI and the positions of contour vertex by “findContours” function in Opencv. Then we use the following method to remove redundant contours.

- (1) Limit the minimum area of the contour: according to the size ratio of the chessboard array to the big black box and their relative positions,

FIGURE 3: The illustration of image registration and coordinate unification.

the threshold of minimum area is calculated, as shown in

$$b_area_{min} = \begin{cases} 1.5 * cb_{areaavg} & \text{(front and rear view)} \\ 0.1 * cb_{areaavg} & \text{(left and right view)}, \end{cases} \quad (3)$$

where b_area_{min} denotes the threshold of big black box area and $cb_{areaavg}$ denotes the average area of small box in chessboard array.

- (2) Limit contour shape: according to the location of the big black box and the imaging features of fisheye camera, the big black box should be in a fixed shape. The shape restrictions are shown in

$$d_1 \geq 0.15 * p \ \&\& \ d_2 \geq 0.15 * p$$

$$8 * d_3 \geq d_4 \ \&\& \ 8 * d_4 \geq d_3 \quad (4)$$

$$area_{contour} > 0.3 * area_{rect},$$

where d_1 and d_2 denote the diagonal length of the contour, d_3 and d_4 denote the length of adjacent side of contour, p denotes perimeter of contour, $area_{contour}$ denotes the area of contour, and $area_{rect}$ denotes the area of envelope of contour.

- (6) Use the SUSAN method to locate the exact positions of the contour vertex around positions obtained from step (5).

3.2.2. Image Registration and Coordinate Unification. After the calibration of Figure 3 and corner detection, the corner positions in coordinates of corrected images and their target positions in coordinates of output images are obtained. Then

we need to unify the coordinate of 4 corrected images into the coordinate of output image, as shown in Figure 3. The specific process is as follows. Firstly, we calculate the homography transform matrix, as shown in (5). The form of this matrix is shown in (6).

$$P_b = H * P_u \quad (5)$$

$$H = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad (6)$$

where $P_u = [x_u \ y_u \ 1]^T$ denotes the corner position in the coordinate of corrected image. $P_b = [x_b \ y_b \ 1]^T$ denotes the target position in the coordinate of output image.

Secondly, we project every pixel of 4 corrected images into the coordinate of output image, as shown in

$$\begin{aligned} w &= a_{31}x_c + a_{32}y_c + a_{33} \\ x_o &= \frac{a_{11}x_c + a_{12}y_c + a_{13}}{w} \\ y_o &= \frac{a_{21}x_c + a_{22}y_c + a_{23}}{w}, \end{aligned} \quad (7)$$

where (x_c, y_c) denotes the pixel position in the coordinate of corrected image. (x_o, y_o) denotes the pixel position in the coordinate of output image.

3.3. Image Blending. As the corrected images from 4 cameras are different from each other in brightness, saturation, and color, we blend them to improve the visual effect of output image by using ring fusion method.

The detailed process is shown as follows.

- (1) *Equalization Preprocessing.* The “imadjust” function in Matlab is used for equalization preprocessing to reduce

FIGURE 4: The image blending result.

the brightness difference among images. For example, the original image of left view angle is shown in Figure 4(a) and the processing result is Figure 4(b).

(2) Ring Color Matching

Step 1 (spatial transformation). As RGB space has a strong correlation, it is not suitable for image color processing. So we transform RGB space to the $\alpha\beta$ space where the correlation between three channels is the smallest. The space conversion process includes three transformations, namely, $RGB \rightarrow CIE\ XYZ \rightarrow LMS \rightarrow \alpha\beta$.

Firstly, from RGB space to CIE XYZ space, one has

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.5141 & 0.3239 & 0.1604 \\ 0.2651 & 0.6702 & 0.0641 \\ 0.0241 & 0.1228 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \quad (8)$$

Secondly, from CIE XYZ space to LMS space, one has

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3897 & 0.6890 & -0.0787 \\ -0.2298 & 1.1834 & 0.0464 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}. \quad (9)$$

Since the data are scattered in the LMS space, it is further converted to a logarithmic space with a base of 10, as shown in (10). This makes the data distribution not only more converging but also in line with the results of the psychological and physical research of human feeling for color.

$$\begin{aligned} L &= \log L \\ M &= \log M \\ S &= \log S. \end{aligned} \quad (10)$$

Finally, from LMS space to $\alpha\beta$ space, one has (11). This transformation is based on the principal component analysis (PCA) of the data, where l is the first principal component, α is the second principal component, and β is the third principal component.

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ S \\ M \end{bmatrix}. \quad (11)$$

After the above three steps, the conversion from RGB to $\alpha\beta$ space is completed.

Step 2 (color registration). Firstly, the mean and standard deviations of every channel in $\alpha\beta$ space are calculated according to

$$\begin{aligned} \mu &= \frac{1}{N} \sum_{i=1}^N v_i \\ \sigma &= \sqrt{\frac{1}{N} \sum_{i=1}^N (v_i - \mu)^2}, \end{aligned} \quad (12)$$

where μ denotes the mean value, N denotes the total number of pixels, v_i denotes the value of the pixel i , and σ indicates the standard deviation.

Secondly, the color matching factors are calculated according to

$$f_l = \frac{\sigma_{l,v1}}{\sigma_{l,v2}}$$

$$f_\alpha = \frac{\sigma_{\alpha,v1}}{\sigma_{\alpha,v2}}$$

$$f_\beta = \frac{\sigma_{\beta,v1}}{\sigma_{\beta,v2}}, \quad (13)$$

where f_l denotes the factor that matches the color of $v2$ image to $v1$ in channel l . $\sigma_{l,v1}$ denotes the variance of $v1$ image in channel l . $\sigma_{l,v2}$ denotes the variance of $v2$ image in channel l . And the rest is similar.

Finally, we match the color of images, as shown in

$$l'_{v2} = f_l * (l_{v1} - \overline{l_{v1}}) + \overline{l_{v2}}$$

$$\alpha'_{v2} = f_\alpha * (\alpha_{v1} - \overline{\alpha_{v1}}) + \overline{\alpha_{v2}} \quad (14)$$

$$\beta'_{v2} = f_\beta * (\beta_{v1} - \overline{\beta_{v1}}) + \overline{\beta_{v2}},$$

where l'_{v2} denotes pixel value of image $v2$ after color matching in channel l . f_l denotes the factor of color matching in channel l . l_{v1} denotes pixel value of image $v1$ in channel l . $\overline{l_{v1}}$ denotes average pixel value of image $v1$ in channel l . $\overline{l_{v2}}$ denotes average pixel value of image $v2$ in channel l . And the rest is similar.

Step 3 (global optimization). Then, we match the color of images from 4 cameras anticlockwise as follows to reach a global optimization result. Firstly, we match the colors of V_4 to V_3 , then V_3 to V_2 , then V_2 to V_1 , and finally V_4 to V_1 , which forms a ring shape, as shown in Figure 5. The processing result of left view is shown in Figure 4(c).

(3) *Weighted Blending.* After color matching, the visual effects of output image have been greatly improved. But around the stitching seam between different corrected images, the visual effect is still not enough. Therefore, we use (15) to ensure smooth transition. The interpolation result of left view angle image is shown in Figure 4(d).

$$O(i, j) = V_1(i, j) \times \frac{d}{d_{\max}} + V_2(i, j) \times \left(1 - \frac{d}{d_{\max}}\right), \quad (15)$$

$$0 < d < d_{\max},$$

where $O(i, j)$ denotes the pixel value in output image and (i, j) is the position index of pixel. $V_1(i, j)$ and $V_2(i, j)$ denote the corresponding pixel value in corrected images V_1 and V_2 . d denotes the distance from pixel to the seam. $d_{\max}$ denotes the width of transition field, as shown in Figure 5.

4. Experiment Result

Some details of the experiment have been provided in part 2 of this paper. So, in this part we only introduce the result. The fisheye images captured from 4 cameras are shown in Figure 6. And their corresponding corrected images are shown in Figure 7. The corner detection and calculation result are shown in Figure 8, where Figure 8(a) shows

FIGURE 5: The illustration of ring color matching method.

TABLE 1: Comparison of different corner detection algorithms.

Method	$\frac{C_c}{N}$	$\frac{C_b}{N}$	$\frac{C_c + C_b}{N}$
Rufli [19]	75%	0%	75%
Harris [13]	6.93%	1.65%	8.59%
Tian [7]	16.83%	2.63%	19.46%
Integrated corner detection method	75%	25%	100%

the corner positions detected in the distortion image and Figure 8(b) shows the corresponding positions calculated in the corrected image. The integrated corner detection algorithm is compared with several other corner detection algorithms, in Table 1.

In Table 1, C_c denotes the number of corner points detected correctly in chessboard. C_b denotes the number of corner points detected correctly in big black box. N denotes all the number of corner points detected in the calibration scene. The Rufli method cannot detect vertices of the big black box. The Harris and Shi-Tomasi methods cannot detect all target vertices and generate a lot of corner redundancy. And the integrated corner detection algorithm can accurately extract all the target corner points of calibration pattern in the scene. As a result, the integrated corner detection algorithm proposed by us is effective.

The output image result is shown in Figure 9. Figure 9(a) is the result before image blending, and Figure 9(b) is the result after image blending. The experimental results show that the proposed algorithm has visual effect around the stitching seam, which proves that our ring fusion method is effective.

5. Conclusion

This paper has proposed a ring fusion method to obtain a better visual effect of AVMS system for intelligent driving. To achieve this condition, an integrated corner detection method of image registration and a ring shape scheme for image blending have been presented. Experiment results prove that this designed approach is satisfactory. 100% of the

FIGURE 6: Fisheye images from each camera.

FIGURE 7: Corresponding corrected images.

FIGURE 8: The corner detection and calculation result.

FIGURE 9: Stitched bird view image of AVM.

required corner is accurately and fully automatically detected. The transition around the fusion seam is smooth, with no obvious stitching trace. However, the images we processed in this experiment are static. So, in the future work, we will transplant this algorithm to development board for dynamic real-time testing and try to apply the ring fusion method to more other occasions.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the National High Technology Research and Development Program ("973" Program) of China under Grant no. 2016YFB0100903, Beijing Municipal Science and Technology Commission special major under

Grant nos. D171100005017002 and D171100005117002, the National Natural Science Foundation of China under Grant no. U1664263, Junior Fellowships for Advanced Innovation Think-Tank Program of China Association for Science and Technology under Grant no. DXB-ZKQN-2017-035, and the project funded by China Postdoctoral Science Foundation under Grant no. 2017M620765.

References

- [1] C. Guo, J. Meguro, Y. Kojima, and T. Naito, "A Multimodal ADAS System for Unmarked Urban Scenarios Based on Road Context Understanding," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1690–1704, 2015.
- [2] A. Pandey and U. C. Pati, "Development of saliency-based seamless image compositing using hybrid blending (SSICHB)," *IET Image Processing*, vol. 11, no. 6, pp. 433–442, 2017.
- [3] Ministry of Public Security Traffic Administration People's Republic of China Road Traffic accident statistic annual report, Jiangsu

- Province Wuxi: Ministry of Public Security Traffic Management Science Research Institute, 2011.
- [4] S. Lee, S. J. Lee, J. Park, and H. J. Kim, "Exposure correction and image blending for planar panorama stitching," in *Proceedings of the 16th International Conference on Control, Automation and Systems, ICCAS 2016*, pp. 128–131, kor, October 2016.
 - [5] H. Ma, M. Wang, M. Fu, and C. Yang, "A New Discrete-time Guidance Law Base on Trajectory Learning and Prediction," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Minneapolis, Minnesota.
 - [6] C.-L. Su, C.-J. Lee, M.-S. Li, and K.-P. Chen, "3D AVM system for automotive applications," in *Proceedings of the 10th International Conference on Information, Communications and Signal Processing, ICICS 2015*, Singapore, December 2015.
 - [7] F. Tian and P. Shi, "Image Mosaic using ORB descriptor and improved blending algorithm," in *Proceedings of the 2014 7th International Congress on Image and Signal Processing, CISP 2014*, pp. 693–698, China, October 2014.
 - [8] S. M. Santhanam, V. Balisavira, S. H. Roh, and V. K. Pandey, "Lens distortion correction and geometrical alignment for Around View Monitoring system," in *Proceedings of the 18th IEEE International Symposium on Consumer Electronics, ISCE 2014*, Republic of Korea, June 2014.
 - [9] D. Suru and S. Karamchandani, "Image fusion in variable raster media for enhancement of graphic device interface," in *Proceedings of the 1st International Conference on Computing, Communication, Control and Automation, ICCUBEA 2015*, pp. 733–736, India, February 2015.
 - [10] C. Yang, H. Ma, B. Xu, and M. Fu, "Adaptive control with nearest-neighbor previous instant compensation for discrete-time nonlinear strict-feedback systems," in *Proceedings of the 2012 American Control Conference, ACC 2012*, pp. 1913–1918, can, June 2012.
 - [11] Z. Jiang, J. Wu, D. Cui et al., "Stitching Method for Distorted Image Based on SIFT Feature Matching," in *Proceedings of the International Conference on Computing and Networking Technology*, pp. 107–110, 2013.
 - [12] E. M. Upadhyay and N. K. Rana, "Exposure fusion for concealed weapon detection," in *Proceedings of the 2014 2nd International Conference on Devices, Circuits and Systems, ICDCS 2014*, India, March 2014.
 - [13] I. Sipiran and B. Bustos, "Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes," *The Visual Computer*, vol. 27, no. 11, pp. 963–976, 2011.
 - [14] Y. Zhao and D. Xu, "Fast image blending using seeded region growing," *Communications in Computer and Information Science*, vol. 525, pp. 408–415, 2015.
 - [15] Y.-K. Huo, G. Wei, Y.-D. Zhang, and L.-N. Wu, "An adaptive threshold for the Canny Operator of edge detection," in *Proceedings of the 2nd International Conference on Image Analysis and Signal Processing, IASP'2010*, pp. 371–374, China, April 2010.
 - [16] G. Peljor and T. Kondo, "A saturation-based image fusion method for static scenes," in *Proceedings of the 6th International Conference on Information and Communication Technology for Embedded Systems, IC-ICTES 2015*, Thailand, March 2015.
 - [17] L. Jiang, J. Liu, D. Li, and Z. Zhu, "3D point sets matching method based on moravec vertical interest operator," *Advances in Intelligent and Soft Computing*, vol. 144, no. 1, pp. 53–59, 2012.
 - [18] J. Lang, "Color image encryption based on color blend and chaos permutation in the reality-preserving multiple-parameter fractional Fourier transform domain," *Optics Communications*, vol. 338, pp. 181–192, 2015.
 - [19] M. Rufli, D. Scaramuzza, and R. Siegwart, "Automatic detection of checkerboards on blurred and distorted images," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 3121–3126, France, September 2008.
 - [20] J.-E. Scholtz, K. Hüßers, M. Kaup et al., "Non-linear image blending improves visualization of head and neck primary squamous cell carcinoma compared to linear blending in dual-energy CT," *Clinical Radiology*, vol. 70, no. 2, pp. 168–175, 2015.
 - [21] Y. Liu, S. Liu, Y. Cao, and Z. Wang, "Automatic chessboard corner detection method," *IET Image Processing*, vol. 10, no. 1, pp. 16–23, 2016.
 - [22] Y. Zhang, S. Deng, Z. Liu, and Y. Wang, "Aesthetic QR Codes Based on Two-Stage Image Blending," in *MultiMedia Modeling*, vol. 8936 of *Lecture Notes in Computer Science*, pp. 183–194, Springer International Publishing, Cham, 2015.
 - [23] K. Pulli, M. Tico, and Y. Xiong, "Mobile panoramic imaging system," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010*, pp. 108–115, USA, June 2010.
 - [24] X. Zhang, H. Gao, M. Guo, G. Li, Y. Liu, and D. Li, "A study on key technologies of unmanned driving," *CAAI Transactions on Intelligence Technology*, vol. 1, no. 1, pp. 4–13, 2016.
 - [25] Y. Tang and J. Shin, "Image Stitching with Efficient Brightness Fusion and Automatic Content Awareness," in *Proceedings of the International Conference on Signal Processing and Multimedia Applications*, pp. 60–66, Vienna, Austria, August 2014.
 - [26] H. B. Gao, X. Y. Zhang, T. L. Zhang, Y. C. Liu, and D. Y. Li, "Research of intelligent vehicle variable granularity evaluation based on cloud model," *Acta Electronica Sinica*, vol. 44, no. 2, pp. 365–374, 2016.
 - [27] J.-H. Cha, Y.-S. Jeon, Y.-S. Moon, and S.-H. Lee, "Seamless and fast panoramic image stitching," in *Proceedings of the 2012 IEEE International Conference on Consumer Electronics, ICCE 2012*, pp. 29–30, USA, January 2012.
 - [28] J. Liu, H. Ma, X. Ren, and M. Fu, "Optimal formation of robots by convex hull and particle swarm optimization," in *Proceedings of the 2013 3rd IEEE Symposium on Computational Intelligence in Control and Automation, CICA 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013*, pp. 104–111, Singapore, April 2013.