CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Panoramic visual system for spherical mobile robots

Muhammad Affan Arif[1,2] , Aibin Zhu[1,2], Han Mao[1,2] and Yao Tu[1,2]

[1]Institute of Robotics and Intelligent Systems, Xi'an Jiaotong University, Xi'an, 710049, China and [2]Shaanxi Key Laboratory of Intelligent Robots, Xi'an, 710049, China
Corresponding author: Aibin Zhu; Email: abzhu@mail.xjtu.edu.cn

Abstract

Aimed at the challenges of wide-angle mobile robot visual perception for diverse field applications, we present the spherical robot visual system that uses a 360° field of view (FOV) for realizing real-time object detection. The spherical robot image acquisition system model is developed with optimal parameters, including camera spacing, camera axis angle, and the distance of the target image plane. Two 180°-wide panoramic FOVs, front and rear view, are formed using four on-board cameras. The speed of the SURF algorithm is increased for feature extraction and matching. For seamless fusion of the images, an improved fade-in and fade-out algorithm is used, which not only improves the seam quality but also improves object detection performance. The speed of the dynamic image stitching is significantly enhanced by using a cache-based sequential image fusion method. On top of the acquired panoramic wide FOVs, the YOLO algorithm is used for real-time object detection. The panoramic visual system for the spherical robot is then tested in real time, which outputs panoramic views of the scene at an average frame rate of 21.69 fps and panoramic views with object detection at an average of 15.39 fps.

1. Introduction

Spherical robots represent a special nonconventional class of mobile robots that demonstrates remarkable dynamic characteristics. These robots are composed of an external spherical shell, and their drive units and electronic components are all sealed inside this spherical shell. This inherently provides protection to the key components of the robot and makes it suitable for hazardous and aquatic environments. Also, the contact between the outer spherical shell of the robot and the ground is just a single point, which makes their motion highly efficient. Moreover, there is no concern for the overturning of these robots, which renders them suitable for applications where frequent falling is inevitable. Furthermore, spherical robots can be designed as holonomic as well as non-holonomic robots depending upon the application needs. Therefore, despite having some disadvantages, such as their inability to climb stairs, the structural characteristics of the spherical robots make them robust and suitable for a multitude of applications, including child development [1], underwater research and surveillance [2, 3], security [4], and terrestrial and extraterrestrial exploration [5].

The last decade saw much work in the field of spherical robots. Several designs were presented, especially for the actuation mechanism of the robot [6]. One strategy proposed is to use a ball or wheels or a car inside the spherical robot [7, 8]. The ball or the car is then controlled and the friction between the wheel or ball and the inner surface of the spherical shell is used to transmit the force to the robot. Several researchers also used rotating flywheels or control moment gyroscopes as the actuation mechanism for spherical robots [9]–[11]. As rotated at a high speed, these flywheels would generate an angular momentum, which is balanced by the movement of the spherical robot. Another strategy is to displace the center of mass of the robot, which would disturb the robot's equilibrium and drive it. The use of sliding masses was proposed to realize spherical robot motion based on this principle [12]. More popular

spherical robots are the pendulum-driven spherical robots, which use one or two pendulums to displace their center of mass, generating higher torque compared with other types of spherical robots [13]–[16].

To fully utilize the advantages of spherical robots and deploy them for real-world field applications, visual sensing and perception abilities are necessary. Despite much work done on these robots, especially in the last two decades, not much attention has been paid to the sensing and perception of spherical robots. This is due to the fact that these robots are still in their early stages of development, and much work is needed on the design of drive units, dynamic modeling, and motion control. Also, the unconventional structure of these robots makes it difficult to mount sensors on them. A few studies are found in the literature on the spherical robot visual system. A portable amphibious robot was presented in ref. [17] for live-streaming purposes. Quan *et al.* designed a spherical robot with a binocular stereo camera for the inspection of plant stem diameter [18]. Similarly, a docking system was developed for an amphibious spherical robot based on a binocular camera [19]. A monocular camera was used on BYQ-5 spherical robot for vision-based path identification [20].

For intricate operations involved in applications such as surveillance and planetary exploration and for realizing autonomy, wide visual information of the surroundings of the robot is necessary. However, the work on visual sensing and perception of spherical robots in the literature is for very specific cases and with a small field of vision, which requires the robot to turn to collect further information. However, this is impractical for security and surveillance applications where the environment is dynamic and requires the latest information without delays. For this purpose, in this paper, we present a panoramic visual system for spherical robots, which can acquire visual information of all its surrounding environment at any given time.

To realize a panoramic view, image stitching is used. We considered only classical feature extraction and matching algorithms since these methods are suitable for resource-constrained applications, such as mobile robotics. SIFT (scale invariant feature transform) is a renowned classical method used for feature extraction and matching because of its ability to extract distinctive features that are invariant to scale and rotational changes [21]. However, the SIFT descriptors are high-dimensional, which leads to large memory requirements. Moreover, it also requires careful parameter tuning, which can be sensitive to the image data and application. ORB (oriented FAST and rotated BRIEF) is another classical algorithm that is often used in robotic applications, especially because of its higher computational speed as its descriptors are binary in nature [22]. However, this also limits its ability to capture fine-grained details and discriminate similar-looking features, which is crucial in image stitching and many other robotic visual applications. Furthermore, the camera mounting angles on the spherical robot can also lead to different viewpoints, especially for close-range objects. ORB's orientation assignment may not fully compensate for large changes in viewpoints, which might impact the stitching of the view fields. A-KAZE is another algorithm known for its higher efficiency, and distinctive and robust features [23]. It represents a compromise between the speed of ORB and the accuracy of the SIFT algorithm. In this paper, we use the SURF algorithm [24], which ensures higher robustness to viewpoint changes and higher feature-matching accuracy compared to ORB due to its more expressive feature descriptors. Moreover, compared to SIFT and A-KAZE, SURF has fewer parameters to tune and has low-dimensional descriptors leading to simplicity and lower memory usage, which is beneficial for robotic applications. However, on the downside, SURF is patented and also has a higher computational cost than that of ORB and A-KAZE.

In this paper, we present the design of a spherical robot panoramic visual system by using classical algorithms because of their lower computational cost which is suitable for real-time robotic applications. We consider a case of a surveillance spherical robot. For mounting four cameras on the robot, our designed spherical robot has side lobes with translucent domes. For the installation parameters of the onboard cameras, we derived a geometrical model, which considers the robot's geometry as well as the overlapping ratio of the image streams. For feature extraction, classical SURF is used and its computational cost is reduced by threshold optimization and reducing the region of interest. The stitching seam lines are reduced by changing the weight functions in fade-in and fade-out algorithm. Cache-based dynamic image fusion is used to increase the frame rate of the output panoramic view field. Finally, the

YOLO algorithm is used for object detection in the wide field of view (FOV) as a case scenario of a surveillance spherical robot. Although algorithmic improvements and analyses are considered, the primary goal of the paper is the design of the panoramic visual system for spherical robots rather than algorithmic novelty.

The main contributions of this paper are:

1. The first spherical robot with 360°-wide panoramicFOV is presented. A mathematical model is established for the camera installation parameters for the long-range and short-range visual acquisition, which can be used for any spherical robot that uses cameras on its sides.

2. Application-based algorithmic improvements are made specifically for the spherical robot panoramic visual system. First, the SURF algorithm is used for feature extraction and matching only in the region of interest (ROI). The ROI and the thresholds are selected on experimental basis for optimal SURF feature extraction and matching accuracy and efficiency. Improved weight gain functions are used in fade-in and fade-out algorithm, which resulted in negligible ghosting effects. For dynamic stitching, caching of the KeyFrame is used which further reduced the stitching time and improved the frame rate of the output panoramic video.

3. Finally, the visual system is tested on a spherical robot prototype for real-time performance. The output panoramic video is then subjected to an application scenario of object detection, which pushes this species of robots for real-world practical applications.

2. Spherical robot visual acquisition system model

The spherical robot is installed with four cameras, each camera having a 120°-wide FOV. Two of the cameras capture images for the front 180°-wide FOV, whereas the other two cameras for the rear 180°. The symmetry of the robot enables the use of the same algorithms twice to acquire front and rear panoramic views. Therefore, for better presentation, we only discuss the two front cameras, and all procedures are repeated for the rear two.

2.1. Short-range visual acquisition model

The short range for a spherical robot is considered when the distance of the target object plane is less than 20 m from the robot. The schematic model is shown in Fig. 1, where β is the FOV of a single camera, α is the camera axis angle, L is the distance between the robot and target object plane, d is the distance between the two cameras, also called baseline width, D′E′ is the cumulative FOV of the two cameras, and DE is the width of the overlapped region. From the quadrilateral ACBF (Fig. 2),

$$\gamma = \beta - \alpha$$

where γ is the angle of the overlapping region of the FOV of two cameras. The area CDE represents the overlapping region of the FOVs of the two cameras. The distance H of this region from the baseline AB is

$$H = \frac{d}{2 \tan\left(\frac{\gamma}{2}\right)}$$

Using similar triangles, we get the width DE of the overlapped region as

$$DE = \left(\frac{L}{H} - 1\right) AB = \left(\frac{2L \tan\left(\frac{\gamma}{2}\right)}{d} - 1\right) d = 2L \tan\left(\frac{\gamma}{2}\right) - d$$
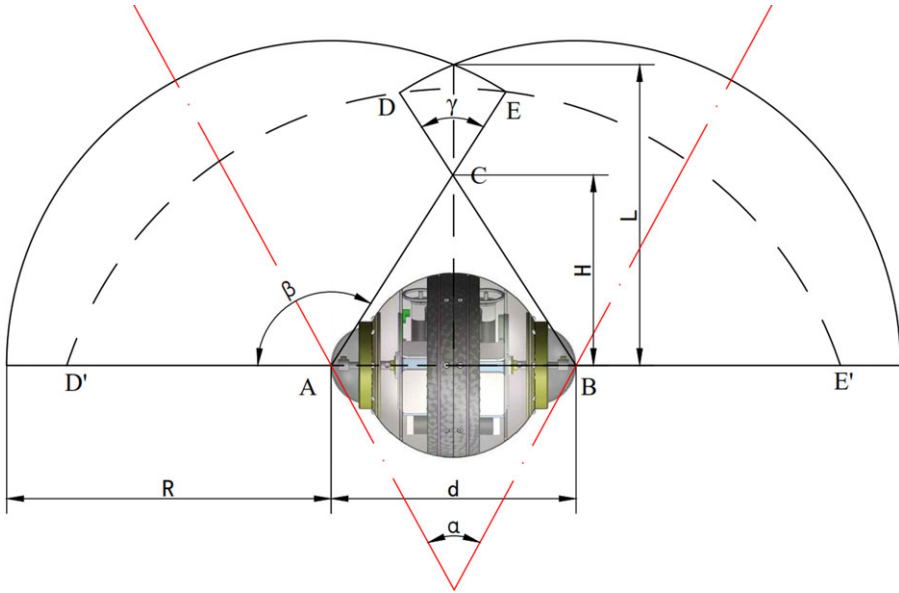
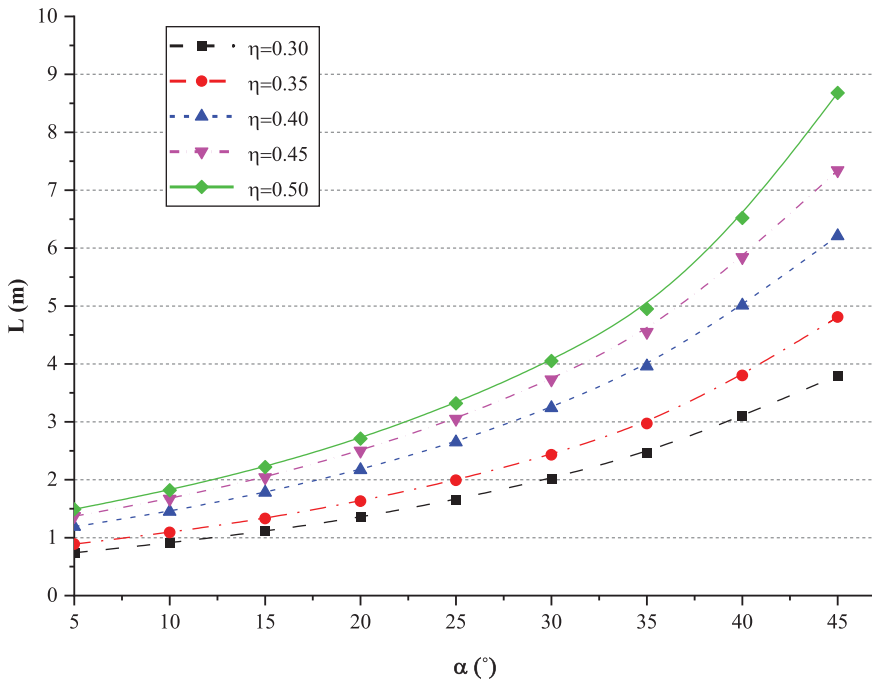**Figure 1.** *Short-range visual acquisition schematic model.*

Figure 2. *Camera axis angle for short-range visual acquisition.*

The overlapping ratio, denoted by η, is critical for image stitching, with its desired value ranging between 0.3 and 0.5. The expression of the overlapping ratio for the spherical robot is derived as

$$\eta = \frac{DE}{D'E} = \frac{2L\tan\left(\frac{\gamma}{2}\right) - d}{R\beta} = \frac{2L\tan\left(\frac{\gamma}{2}\right) - d}{\frac{L}{\sin\beta}\beta} = \sin\beta\frac{2L\tan\left(\frac{\gamma}{2}\right) - d}{L\beta}$$
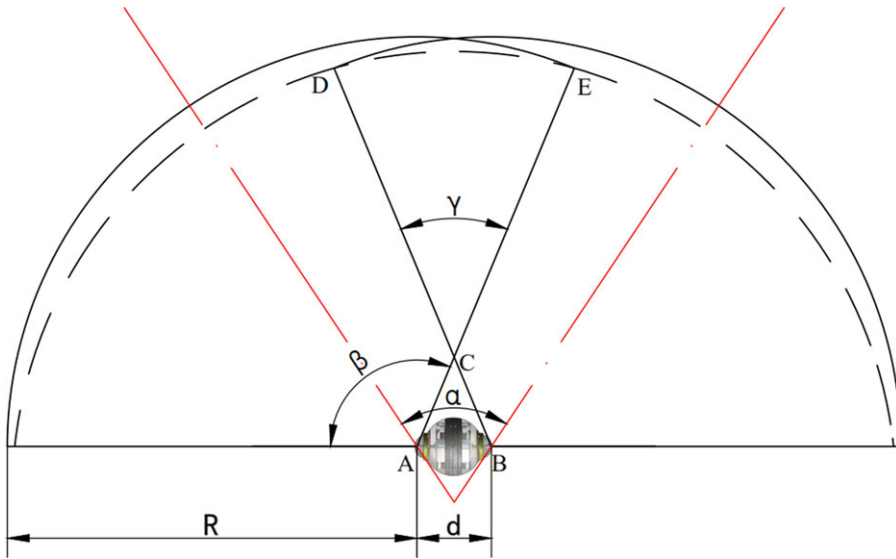
Figure 3. *Long-range visual acquisition schematic model.*

And the distance L of the target object plane is expressed by Eq. (2a) and plotted in Fig. 3:

$$L = \frac{d}{2 \tan \left(\frac{\beta - \alpha}{2} \right) - \frac{\beta \eta}{\sin \beta}} \tag{2a}$$

Using the physical parameters of our spherical robot, the above relationship is plotted as shown in Fig. 2, which shows as the distance of the target object increases, the axis angle of the two cameras should also increase. This also serves well for minimizing parallax problems.

2.2. Long-range visual acquisition model

The long-range model is considered when the distance of the target object plane is larger than 20 m. The schematic model is shown in Fig. 3. The overlapping ratio η for the long-range visual acquisition is given by

$$\eta = \frac{DE}{D'E} = \frac{\gamma (L - H)}{\beta R} = \frac{\gamma}{\beta} \left(1 - \frac{H}{L} \right) \sin \beta$$

Rearranging the above equation gives the relationship between the target object distance and the camera axis angle as

$$L = \frac{d}{2 \tan \left(\frac{\gamma}{2} \right) \left(1 - \frac{\beta \eta}{\gamma \sin \beta} \right)} = \frac{d}{2 \tan \left(\frac{\beta - \alpha}{2} \right) \left(1 - \frac{\beta \eta}{\beta - \alpha \sin \beta} \right)} \tag{2b}$$

To visualize the relationship expressed in the Eq. (2b), the plot is shown in Fig. 4. Similar to the case of short-range visual acquisition, angle α in the case of long-range visual acquisition also increases with the increase in distance L. However, the effect of the overlapping η ratio decreases at larger distances.

The overlapping ratio for this case is expressed as

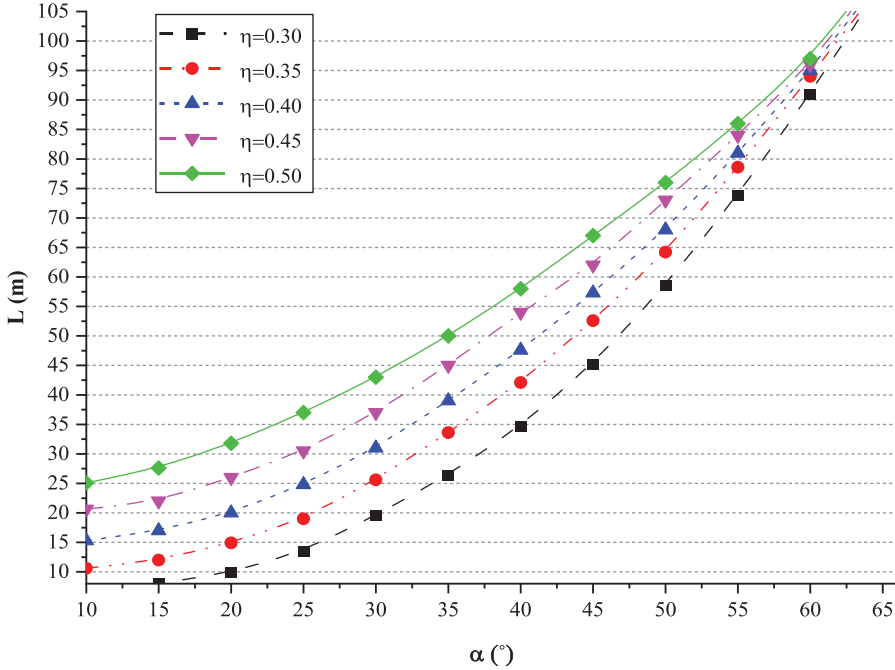$$\eta = \frac{DE}{D'E} = \frac{\gamma}{\beta} = \frac{\beta - \alpha}{\beta} \tag{2c}$$

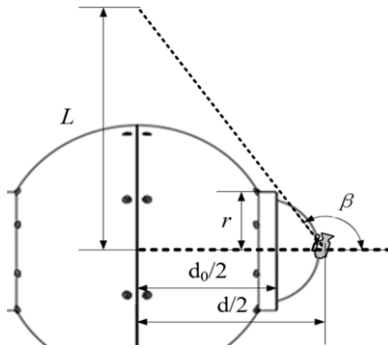**Figure 4.** *Camera axis angle for long-range visual acquisition.*

Figure 5. *Geometrical model of the spherical robot visual acquisition system.*

2.3. Spherical robot geometrical model

The shape and size of the spherical robot influence the camera axis angle as well as the overlapping ratio of the acquired video streams. Among other dimensions, the width of the robot is more critical for our case as the cameras are attached on its sides. The schematic diagram of the robot with a camera is shown in Fig. 5.

In Fig. 5, the width of the spherical robot shell is denoted by d_0, and the radius of its side face is r. By trigonometry, we get:

$$\beta = \arctan\left(-\frac{2r}{d - d_0}\right) \tag{2d}$$

From (2c) and (2d), we get the overlapping ratio:

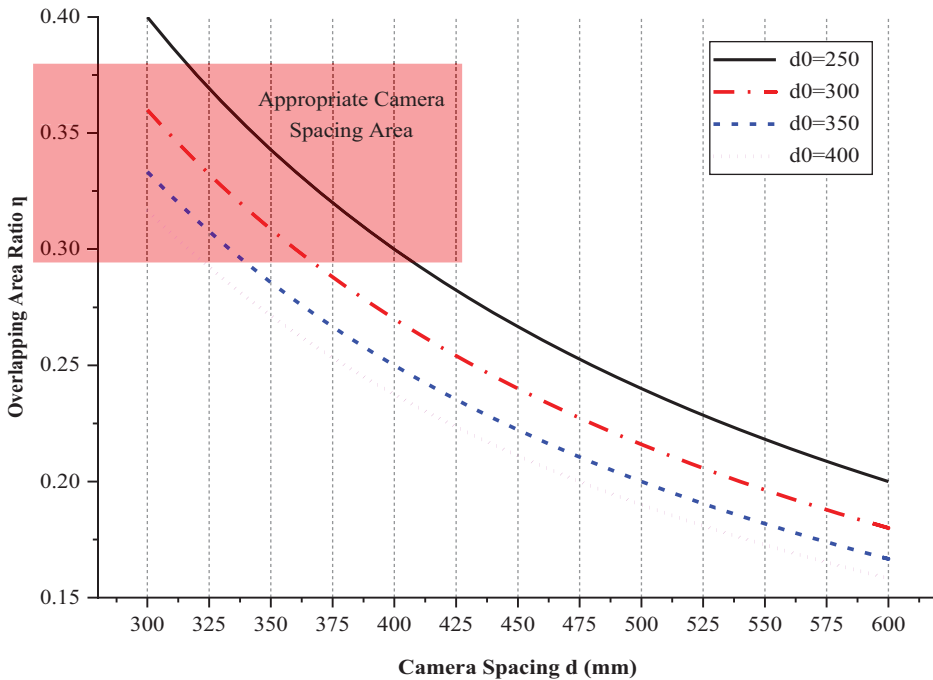$$\eta = \frac{\beta - \alpha}{\beta} = 1 - \frac{\alpha}{\arctan\left(-\frac{2r}{d - d_0}\right)}$$

Figure 6. *Overlapping ratio as a function of baseline width.*

The relationship between the overlapping ratio η and the baseline width is plotted in Fig. 6.

For a given camera axis angle α, the overlapping ratio decreases with the increase in baseline width d. To keep the overlapping ratio within the desired limit, we selected a baseline width of 400 mm for our spherical robot.

3. Image stitching

This paper uses static and dynamic image stitching of the visual streams from the two cameras to generate the real-time panoramic FOV.

3.1. Threshold for feature extraction

The Hessian matrix describes the local curvature of a function and is composed of second-order partial derivatives. The Hessian matrix threshold determines the accuracy and efficiency of the feature-matching algorithm, that is the larger the threshold, the higher the accuracy and lower the efficiency. We performed the matching tests for several images and plotted the relationships, as shown in Fig. 7.

The goodmatch represents the points that meet the threshold screening, whereas efficiency is the ratio of goodmatch points to the time spent. It is noted that setting the Hessian matrix threshold between 1500 and 2000 ascertains higher matching efficiency.

Lowe threshold is the ratio of the distance of the matching feature point from its nearest neighbor to the second nearest neighbor. Using results from data sample, we related the Hessian matrix threshold with efficiency under different Lowe thresholds, as shown in Fig. 8(a).

As the Lowe threshold increases, the efficiency of feature matching also increases. However, the feature-matching accuracy decreases. This is due to the relaxation of the screening scale with the increase in the Lowe threshold. Moreover, the highest correct matching pairs were found for a Lowe threshold
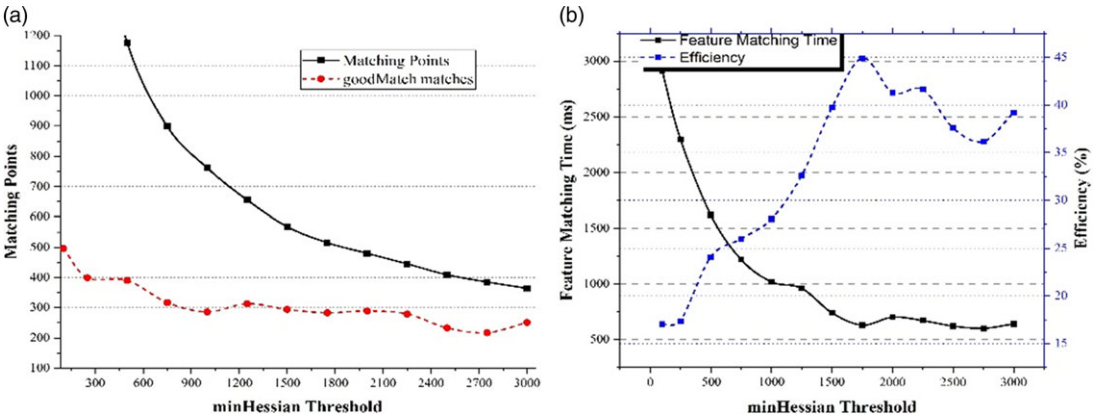
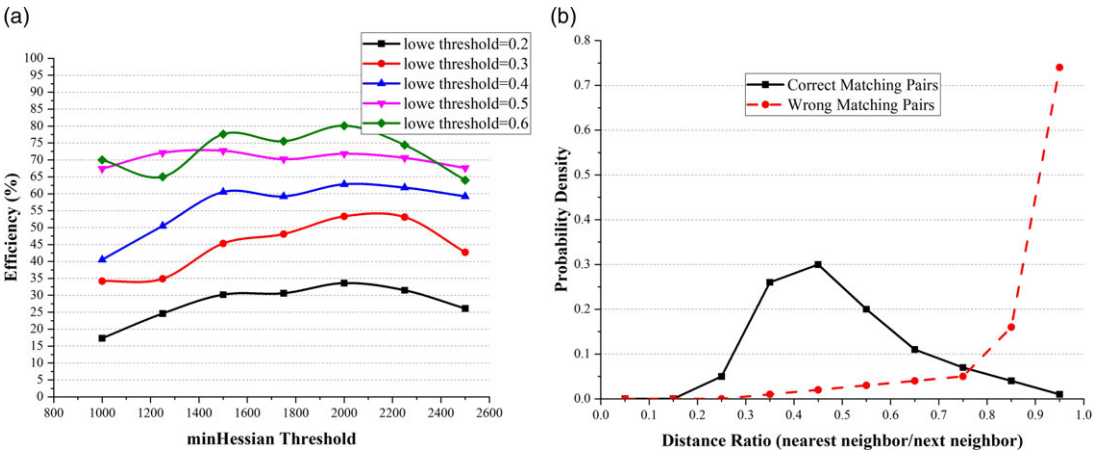**Figure 7.** *Influence of Hessian threshold on feature extraction and matching.*

Figure 8. *Influence of low threshold on feature extraction and matching.*

between 0.3 and 0.6. Therefore, for our spherical robot panoramic system, we set the Hessian threshold range between 1500 and 2000 and the Lowe threshold between 0.3 and 0.6.

3.2. Feature extraction

Our spherical robot prototype uses an onboard Raspberry Pi, which offers much lower computation power. Furthermore, we used SURF for feature extraction and matching primarily due to its expressive descriptor ensuring better feature matching, which is computationally expensive.

To reduce the computation load for ensuring high output frame rates of the panoramic video stream, we extracted the features only from the region of interest (ROI), which is the overlapped region as shown in Fig. 9. Since this region only accounts for 30-40% of the full image, the computation cost is significantly reduced.

To explore the best-suited feature extraction region ε, we conducted image stitching experiments for different overlapping ratios η, and the resulting relationship is shown in Fig. 10.

In the figure, the area on the left side of the red line, that is, $\varepsilon = 0.35$, is the region where the number of matching points is too low and leads to failed stitching. It can be seen that an increase in ε leads to an increase in stitching time and a decrease in accuracy, which is a consequence of an increase in false matches as the feature extraction region enlarges. The inflection points for each value of η, after which
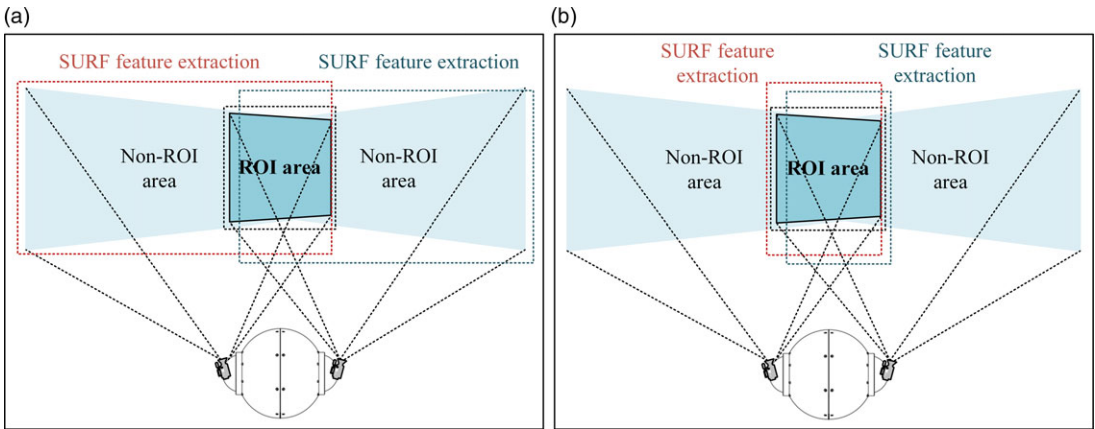
Figure 9. *SURF feature extraction (a) extraction from the complete view, (b) extraction from ROI.*
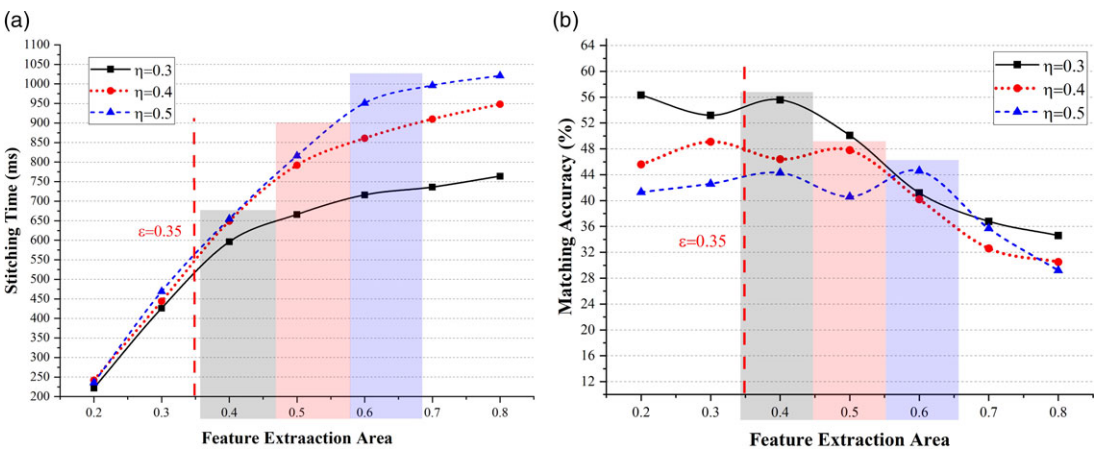

Figure 10. *Feature extraction region analysis (a) based on stitching time, (b) based on matching accuracy.*

the slope of respective curves decreases, are highlighted in the shaded region. From experiments, we observed that $\varepsilon = \eta + 1$ gives the best image stitching performance in terms of both speed and accuracy. From all the analyses discussed in the paper up to this point, we selected an overlapping ratio of $\eta = 0.4$ for our spherical robot visual system. The extracted features were further cleansed through RANSAC, and the two images are then subjected to cylindrical spherical projection.

3.3. Static image fusion

For the seamless fusion of the images acquired by the two cameras on the spherical robot, we used the fade-in and fade-out algorithm and changed its weight functions, given by

$$w_2 = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \times \frac{g_2}{\sqrt{g_1 g_2}}, w_1 = 1 - w_2$$

In the above equation, $g1$ and $g2$ represent the gray value of the left and right camera images, respectively. The difference in the illumination of pixels acquired by two cameras leads to differences in the gray values of the pixels of two images. This difference is mitigated by multiplying the ratio of the gray values with the new weights. The quality of the image fusion is evaluated in terms of standard deviation

Table I. *Evaluation of fusion algorithms.*

Fusion algorithm	Standard deviation (SD) (Group-1)	Information entropy (H) (Group-1)	Standard deviation (SD) (Group-2)	Information entropy (H) (Group-2)
Average method	37.5601	6.9842	49.6358	8.9457
Maximum method	40.1254	7.0142	48.1761	8.7634
Fade-in and fade-out	52.9427	7.0581	59.7652	9.2866
Improved fade-in and fade-out	**58.6138**	**17.1958**	**67.5843**	**10.0205**
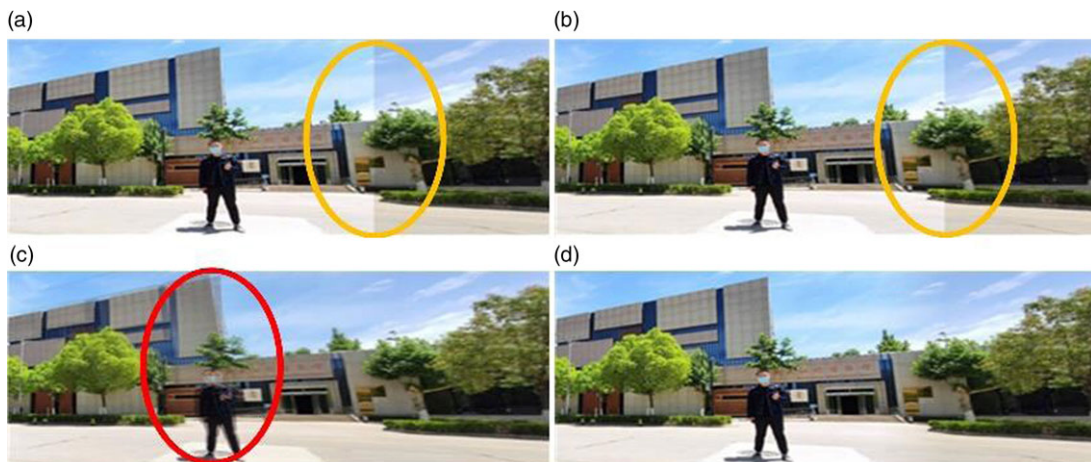
Figure 11. *Image fusion (a) average method (b), maximum method (c), fade-in and fade-out (d), improved fade-in and fade-out.*

(SD) and information entropy (H), as given in Table I. The proposed weights ensured higher value of SD, that is, dispersion of grayscale value, and *H,* that is, quantity of image information, which improves the fusion quality of images, which mitigated the stitching seams and ghosting effects as seen in Fig. 11. A slight difference in the gray values of the two fused images is still observed after using our improved fade-in and fade-out algorithm. This difference is due to the use of only the overlapping region of the two images. However, it has no drastic effects on real-time object detection performance, which is adversely affected by the ghosting effect. These results are presented in Section 5.4.

3.4. Dynamic image fusion

For the dynamic image fusion, we used the sequential image fusion method with caching of KeyFrames. The sequential method performs feature extraction and matching, feature cleansing, projection transformation, and image fusion for each image frame, which results in a panoramic video with a low frame rate.

Instead, we used the caching of the KeyFrames, that is, one of the acquired image pairs is set as a KeyFrame, and the homography is calculated for it and saved as a cache. Since the speed of the spherical robot is not very fast, the same homography can also be used for the subsequent frames for a certain time range. This saves time from feature matching and cleansing of each frame, which are the two most time-taking steps in image stitching. The algorithm is shown in Fig. 12. In practice, we set a KeyFrame after every four frames. The time consumed during each step when the caching-based sequential dynamic image stitching was performed is given in Table II.

The resulting panoramic video frame rate is plotted in Fig. 13. It can be seen that the frame rate fluctuates about an average of 22.16 fps, with a sudden drop to about 10 fps at each KeyFrame.
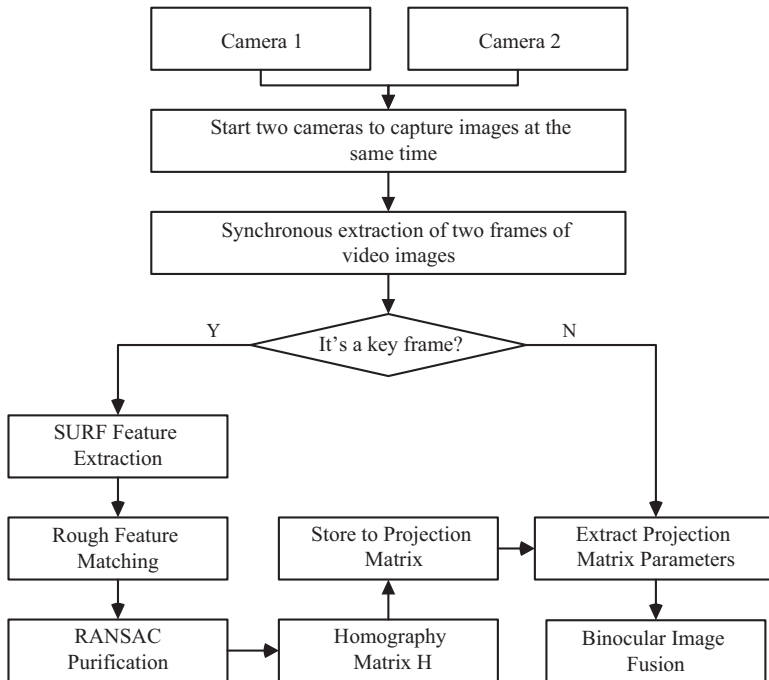
Figure 12. *Sequential dynamic image stitching based on caching of keyFrames.*

4. Real-time object detection

The 180°-wide panoramic view acquired is then used for realizing real-time object detection. The objective of the spherical robot is to carry out mobile surveillance task, which requires quick and correct identification of the objects in its 360° FOV. Deep learning guarantees both high accuracy and high detection speed. We conducted experiments to test YOLO, R-CNN, and Fast R-CNN algorithms to compare their speed, precision, and recall. The experimental results are presented in Table III. The results show YOLO outperforms both R-CNN and Fast R-CNN, especially in the number of output frames. Therefore, we used YOLO. Since Raspberry Pi could not handle this computation, we used the ground station for object detection.

5. Experimental results

Real-time experiments were performed using the spherical robot prototype to comprehensively evaluate our designed panoramic visual system for the spherical robot. The experimental platform is shown in Fig. 14.

Our spherical robot uses two pendulums to drive the robot and uses two flywheels to provide inertial stabilization. The four cameras are installed on the two sides of the robot inside translucent lobes. A bearing prevents the motion transmission between the pendulum and the shaft to which the camera bracket is attached. For computation, the robot carries a Raspberry Pi. The video resolution should neither be too high (computation cost) nor too low (stitching quality). Therefore, we selected cameras with the highest resolution of 1280x720. The overall system is shown in Fig. 15.

5.1. Camera axis angle

Although SURF is robust to viewpoint changes, the feature matching speed and accuracy can be improved by adjusting the camera axis angles. To test this, we considered two cases; (1) short-range

Table II. *Time for each step in the caching-based sequential dynamic image stitching.*

Video frames	Feature matching time (ms)	Feature cleansing time (ms)	Image fusion time (ms)	Total splicing time (ms)
1 (KeyFrame)	717.36	184.16	40.92	942.44
2	0	0	38.15	38.15
3	0	0	39.37	39.37
4	0	0	49.11	49.11
5 (KeyFrame)	698.57	168.77	40.82	908.16
6	0	0	36.94	36.94
7	0	0	42.17	4.17
8	0	0	40.28	40.28
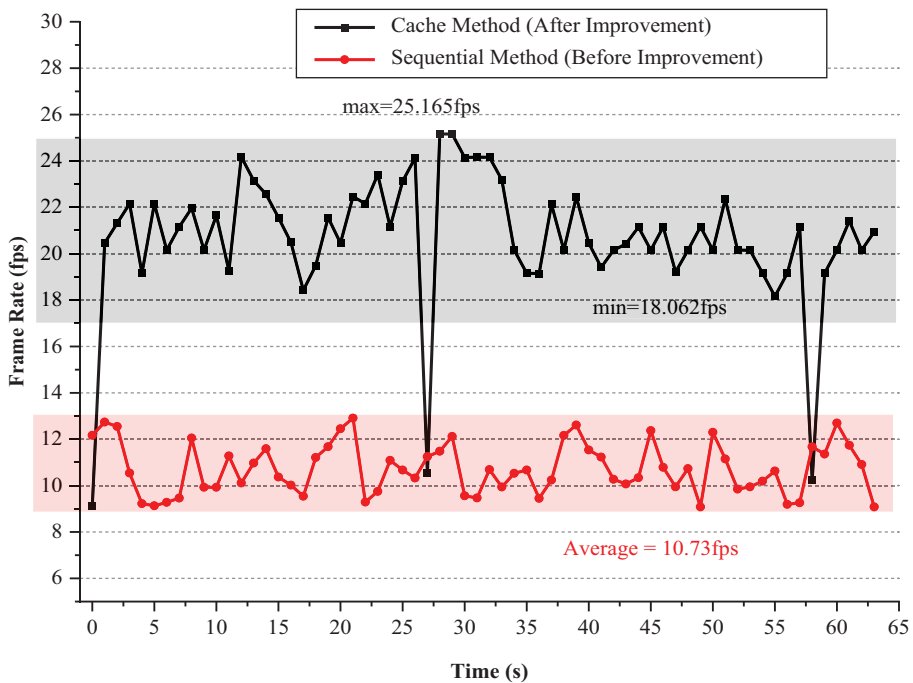
Figure 13. *Output video frame rate from cache-based sequential dynamic image stitching.*

indoor experiments and (2) long-range outdoor experiments. The performance was judged based on good match points, that is, feature points that meet the threshold after feature matching, inliers, that is, feature points that lie within the threshold after applying RANSAC, matching accuracy, and matching time. The results for the short-range indoor experiments are shown in Fig. 16. It can be seen that the number of good match points, inliers, and the correct matches decreases with the increase in the camera axis angle, whereas the matching time increases.

Similarly, in Fig. 17, it can be seen that the number of good matches, inliers, and correct matches decreases with the increase in the camera-axis angle. However, the feature matching does not get impacted by the change of axis angle for long-range visual acquisition. The matching accuracy for both short-range and long-range scenarios is found best for the camera-axis angle range of 30-40°. Also considering the feature matching speed, this camera axis angle range is the optimal choice and is used for our robot.

Table III. *Object detection test results.*

Algorithm	Precision	Recall	Number of output frames
YOLO	0.912	0.851	33
R-CNN	0.901	0.855	1/5
Fast R-CNN	0.908	0.848	3

Spherical Robot Structural Layout   Spherical Robot Camera Mounting   Robot Prototype
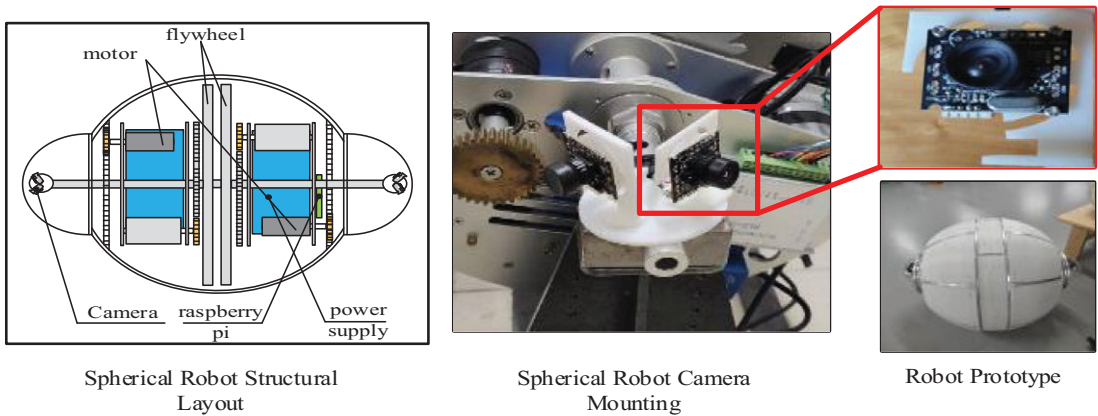
Figure 14. *Spherical robot experimental platform.*
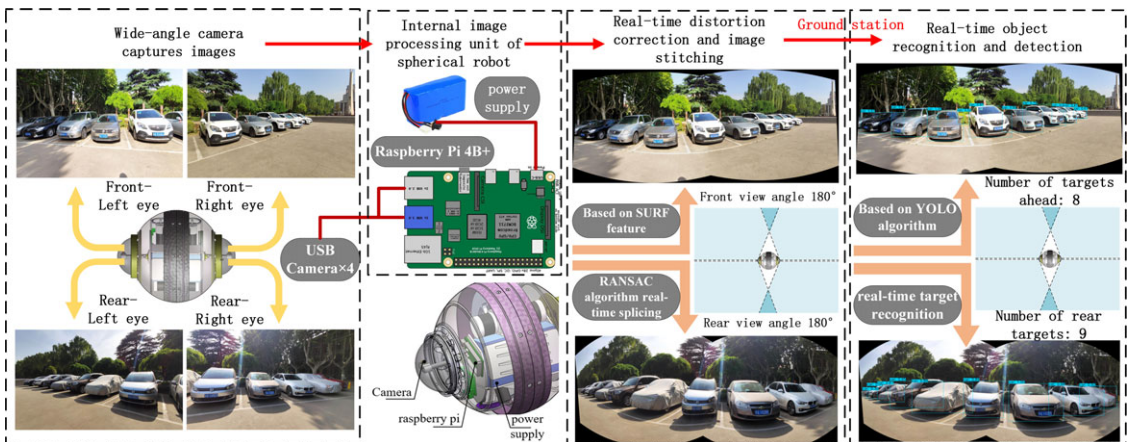
Figure 15. *Spherical robot panoramic visual system.*

5.2. Real-time dynamic image-stitching

We tested the performance of our proposed algorithms during the real-time spherical robot experiments and compared their performance against the original algorithms. During the experiments, we set the KeyFrame after every 25 frames. As discussed before, the feature extraction and projection transformations were only performed on the KeyFrames. The experimental results for the feature matching accuracy and the image stitching time for the original SURF algorithm (when features are extracted from the whole image) are compared with that of the proposed SURF algorithm (when features are only extracted from ROI) in Fig. 18. It can be seen that the matching accuracy for both algorithms stayed almost the same, since most of the inliers that may correctly be matched lie inside the ROI. However, the KeyFrame stitching time was shortened by a factor of 2.37.
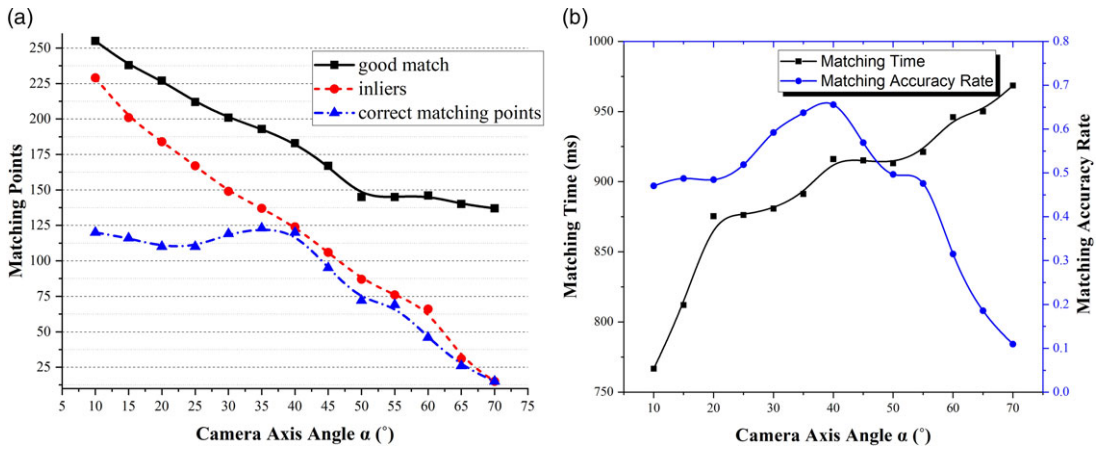
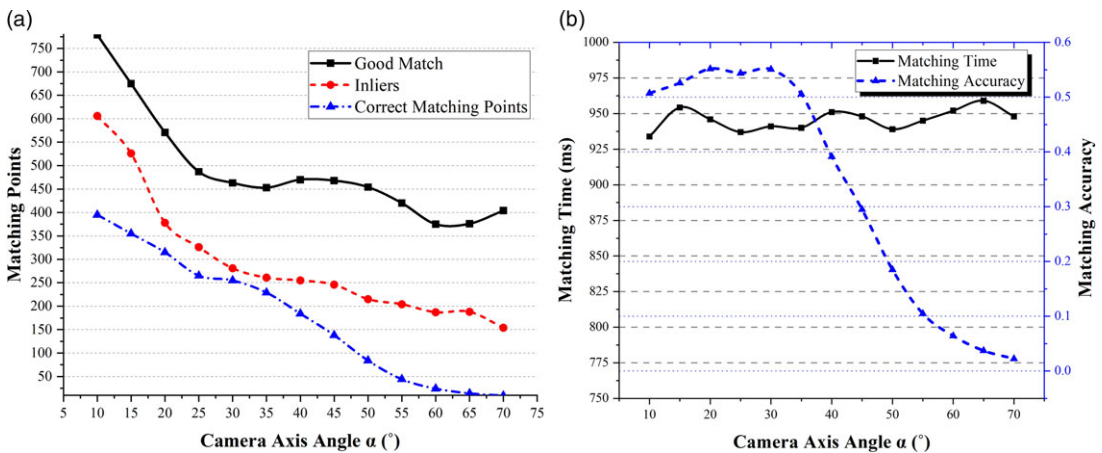**Figure 16.** *Influence of camera axis angle on feature matching for short-range image acquisition.*

Figure 17. *Influence of camera axis angle on feature matching for long-range image acquisition.*

5.3. Real-time panoramic field of view

Using the algorithms and parameters from previous sections, the real-time 360° panoramic view for the spherical robot was formed. The image streams from the front two cameras were used to create the front 180°- wide panoramic view of the scene, whereas the rear two cameras were used to generate the rear 180°-wide panoramic view. The two views were not combined as it would add computational cost, which cannot be handled by the onboard Raspberry Pi. The resulting views are shown in Fig. 19.

5.4. Real-time object detection

The spherical robot visual system was experimentally evaluated, which also constitutes the analysis of individual algorithms for feature extraction and matching, image projection transformation, feature cleansing, static image fusion, and dynamic image fusion, in addition to object detection. Different frames from real-time spherical robot object detection experiments are shown in Fig. 20.

The quantified results of object detection in the panoramic view acquired from the spherical robot are given in Table IV. The minimum accuracy of object detection was found to be 75%, whereas a maximum of 100%. Since this paper is mainly focused on developing the panoramic visual system for the spherical robot and not on developing and benchmarking new algorithms, our initial tests were conducted using
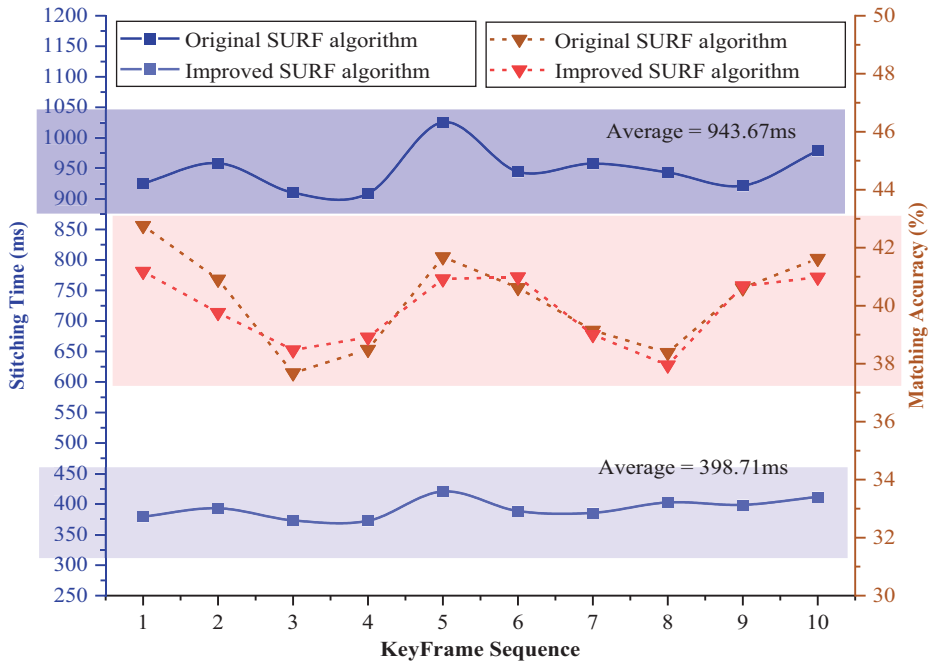
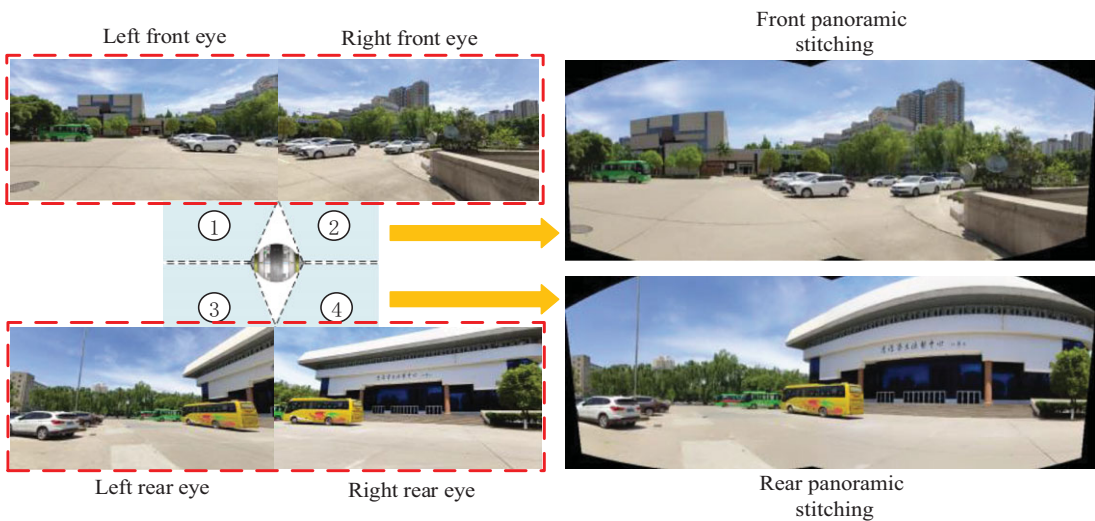**Figure 18.** *Real-time keyFrames stitching speed and accuracy.*

Figure 19. *Real-time image stitching results of the spherical robot panoramic visual system.*

only four classes. In the future, we aim to test our robot in environments with multiple object classes and further explore algorithmic possibilities.

The final frame rate of the panoramic visual system for the spherical robot was 15.39 fps, as shown in Fig. 21(a). The figure also depicts the influence of constraining the SURF feature extraction to ROI in the KeyFrames. It is also interesting to note that if caching of the KeyFrames is used for sequential dynamic image stitching, the effect of constraining the feature extraction to only ROI depends upon the frequency of selecting KeyFrames. If the scene changes frequently, this strategy will not result in a significant improvement in the output frame rate.
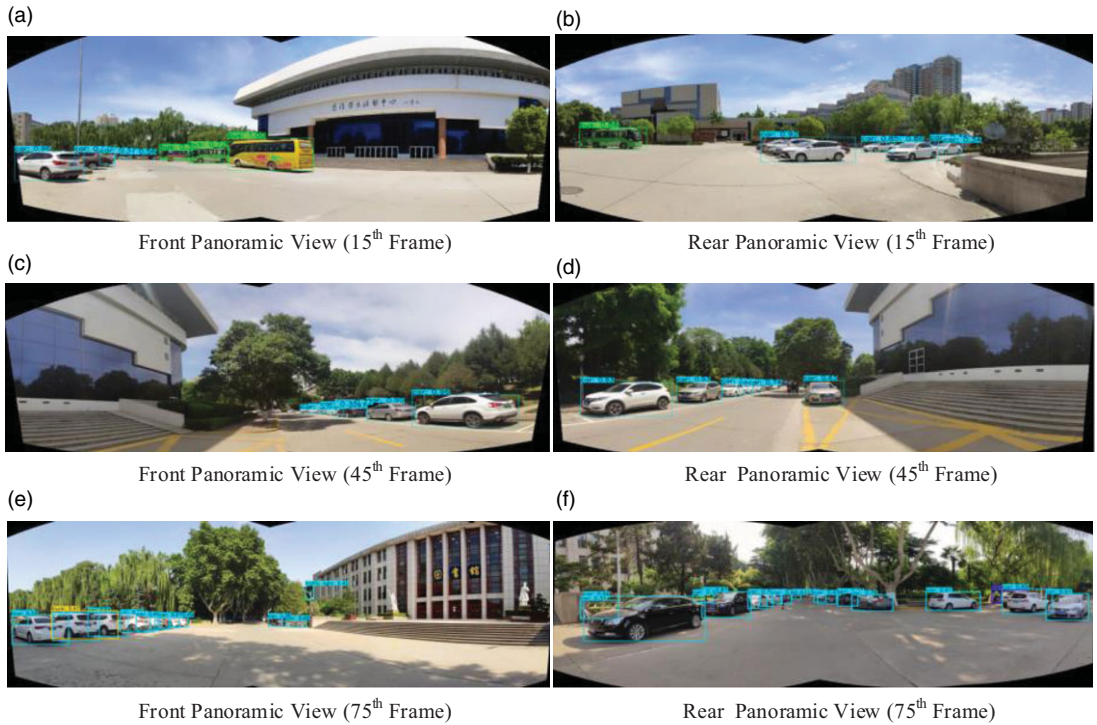
(a)

(b)

Front Panoramic View (15th Frame)   Rear Panoramic View (15th Frame)

(c)

(d)

Front Panoramic View (45th Frame)   Rear Panoramic View (45th Frame)

(e)

(f)

Front Panoramic View (75th Frame)   Rear Panoramic View (75th Frame)

Figure 20. *Real-time object detection in 360° wide panoramic field of view.*

(a)

(b)
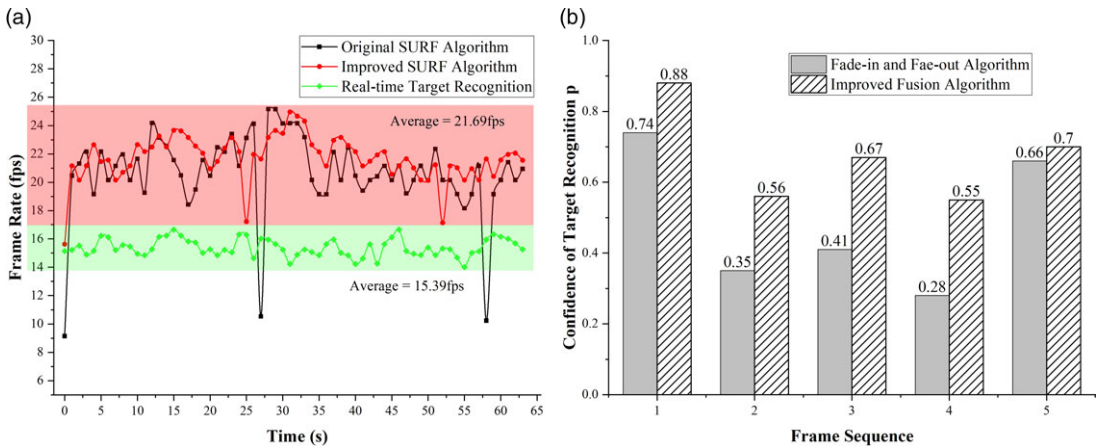
Figure 21. *Real-time spherical robot visual perception performance (a) target detection speed (b) target detection at seams for the improved image fusion algorithm.*

Furthermore, the image seams are the critical regions of the panoramic FOVs. To test the fade-in and fade-out algorithm with our proposed weights, we plotted the confidence of target recognition on the image seams, as shown in Fig. 21(b). Our proposed weights smoothen the pixel intensity changes across the seams, which not only ensure better visuals but also improve the target recognition results.

Table IV. *Object detection results in panoramic field of view of spherical robot.*

Video frames	No. of targets	Identified targets	Quantity correct rate (QCR)
10	11	9	0.82
20	9	7	0.78
30	9	9	1.00
40	10	8	0.80
50	5	5	1.00
60	12	9	0.75

Table V. *Panoramic visual systems in literature.*

Research work	Stitching time (ms)	Panoramic video frame rate (fps)	Robotic implementation	View angle	Robotic application
Li [27]	46.9	21.3	×	360°	×
Hu [28]	75.1	13.3	×	360°	×
Ullah [29]	31	32.3	×	360°	×
Guo [25]	632.8	1.6	✓	–	×
Rettkowski [26]	245	4.1	✓	110°	✓
Our work	**46.1**	**21.7**	✓	**180°+180°**	✓

5.5. Comparison with panoramic visual systems in literature

In the literature, we found several studies on the algorithms to generate panoramic video. However, their physical implementation was mostly limited to panorama-generating drones and hand-held devices. The requirements of panorama generation for robotic platforms are disparate from those of panorama-generating devices. Therefore, we cannot compare our results with such devices. Only a few robotic platforms were found in the literature that used multiple cameras to generate a wide FOV [25, 26]. For comparison, we selected studies that either presented robotic visual systems or focused on panoramic video generation algorithms for real-time applications, as shown in Table V. In the table, robotic implementation refers to the case when the panorama was generated by a mobile vehicle or a robot. Whereas, the robotic application shows the work that used this panoramic vision in some robotic applications, such as object detection or navigation.

Table V shows that only Ullah *et al.* [29] reported a higher frame rate of the panoramic video. They used six equally spaced cameras on a drone for realizing a 360°-wideFOV. However, this drone was kept stationary throughout the process. Since there was no change in the scenery, it was unnecessary to continuously carry out feature extraction, image registration, and blending. Li *et al.* [27] reported a frame rate of 21.3 fps, which is close to the frame rate of the panorama of our spherical robot visual system. However, their algorithm was not tested on a robotic platform. The only study that considered the robotic implementation and application was presented by Rettkowski *et al.* [26]. They used two cameras on a wheeled robot to generate a panorama. However, the stitching speed was very low, and the panorama was only 110° wide.

6. Conclusion and future outlook

This paper provides a firm foundation for the development of intelligent spherical robots with a 360°-wide visual perception. The designed panoramic visual system for spherical robots considers the geometrical constraints of the robot and ensures a high frame rate of output panoramic video. We used four onboard cameras, each acquiring a 120°-wide FOV. For high robustness to view-point changes and more expressive descriptors for feature matching, we used the SURF algorithm. Its computation cost was reduced by constraining the feature extraction region to the overlapping region of the two views. The

image fusion performance of the fade-in and fade-out algorithm was enhanced by using improved weight functions that allow gradual dispersion of pixel intensities. Since image registration and homography calculation add majorly to computational time, caching of the KeyFrames was used in the sequential dynamic image stitching, which increased the overall frame rate of the output video. Finally, using the real-time 360° panoramic view field, object detection was realized using the YOLO algorithm. The comprehensive experimental evaluation promised a spherical robot prototype capable of providing a 360°-wide FOV at 21.69 fps and performing real-time mobile object detection with an average frame rate of 15.39 fps. However, during our tests, only four classes for object detection were used, which is not enough for actual monitoring and mobile surveillance tasks. Furthermore, as the primary aim of the paper was toward application rather than algorithmic innovation, it is suggested to explore the use of more advanced algorithms. Moreover, for unpaved grounds, video stabilization algorithms will be necessary to compensate for the unstable motion of spherical robots, which represents the next step in the realization of a complete spherical robot visual system.

References

[1] Y. Mizumura, K. Ishibashi, S. Yamada, A. Takanishi and H. Ishii, Mechanical design of a jumping and rolling spherical robot for children with developmental disorders (2017).

[2] S. Gu, S. Guo and L. Zheng, "A highly stable and efficient spherical underwater robot with hybrid propulsion devices," *Auton. Robots* **44**(5), 759–771 (2020). doi: 10.1007/s10514-019-09895-8.

[3] C. Yue, S. Guo and L. Shi, "Hydrodynamic analysis of the spherical underwater robot SUR-II," *Int. J. Adv. Robot. Syst.* **10**(5), 247 (2013). doi: 10.5772/56524.

[4] L. Lu and M. Chu, "Shock absorber design and analysis of a catapult-spherical reconnaissance robot," *Adv. Mater. Res.* **662**, 694–697 (2013). doi: 10.4028/www.scientific.net/AMR.662.694.

[5] F. R. Hogan, J. R. Forbes and T. D. Barfoot, "Rolling stability of a power-generating tumbleweed rover," *J. Spacecr. Rockets* **51**(6), 1895–1906 (2014). doi: 10.2514/1.A32883.

[6] R. Chase and A. Pandya, "A review of active mechanical driving principles of spherical robots," *Robotics* **1**(1), 3–23 (2012). doi: 10.3390/robotics1010003.

[7] P. D. Ba, Q. D. Hoang, S.-G. Lee, T. H. Nguyen, X. Q. Duong and B. C. Tham, "Kinematic Modeling of Spherical Rolling Robots with a Three-Omnidirectional-Wheel Drive Mechanism," *2020 20th International Conference on Control, Automation and Systems (ICCAS)* (2020) pp. 463–466. doi:10.23919/ICCAS50221.2020.9268200.

[8] H. Q. Dong, S.-G. Lee, S. H. Woo and T.-A. Leb, "Back-Stepping Approach for Rolling Motion Control of an Under-Actuated Two-Wheel Spherical Robot," *2020 20th International Conference on Control, Automation and Systems (ICCAS)* (2020) pp. 233–238. doi:10.23919/ICCAS50221.2020.9268438.

[9] H. W. Kim and S. Jung, "Design and control of a sphere robot using a control moment gyroscope actuator for navigation," *Int. J. Control. Autom. Syst.* **18**(12), 3112–3120 (2020). doi: 10.1007/s12555-019-0526-2.

[10] H. W. Kim and S. Jung, "Design and Control of a Sphere Robot Actuated by a Control Moment Gyroscope," *2019 19th International Conference on Control, Automation and Systems (ICCAS)* (2019) pp. 1287–1290. doi:10.23919/ICCAS47443.2019.8971512.

[11] S. D. Lee and S. Jung, "Synchronized motion and compliant control for dual control moment gyroscopes by realizing the impedance transformation matrix," *Mechatronics* **84**, 102793 (2022). doi: 10.1016/j.mechatronics.2022.102793.

[12] M. R. Burkhardt, Dynamic Modeling and Control of Spherical Robots (2018).

[13] K. Othman, "Dynamics control of pendulums driven spherical robot," *Appl. Mech. Mater.* **315**, 192–195 (2013). doi: 10.4028/www.scientific.net/AMM.315.192.

[14] O. I. S. Michaud and S. Caron, "Roball, the rolling robot," *Auton. Robots* **12**(2), 211–222 (2002). doi: 10.1023/A:1014005728519.

[15] M. A. Arif, A. Zhu, H. Mao, X. Zhou, J. Song, Y. Tu and Peifeng, "Design of an amphibious spherical robot driven by twin eccentric pendulums with flywheel-based inertial stabilization," *IEEE/ASME Trans. Mech.* **28**(5), 1–13 (2023). doi: 10.1109/TMECH.2023.3247641.

[16] C. Li, A. Zhu, C. Zheng, H. Mao, M. A. Arif, J. Song and Y. Zhang, "Design and Analysis of a Spherical Robot Based on Reaction Wheel Stabilization," *2022 19th International Conference on Ubiquitous Robots (UR)* (2022) pp. 143–148. doi:10.1109/UR55393.2022.9826266.

[17] S. Satria, Portable Amphibious Spherical Rolling Robot with Live-Streaming Capability for Ground and Aquatic Deployment (2015).

[18] L. Z. Quan, C. Chen, Y. Li, Y. Qiao, D. Xi, T. Zhang and W. Sun, "Design and test of stem diameter inspection spherical robot," *Int. J. Agric. Biol. Eng.* **12**(2), 141–151 (2019). doi: 10.25165/j.ijabe.20191202.4163.

[19] S. Guo, Y. Liu, L. Shi, P. Guo, H. Xing, X. Hou, Z. Chen, S. Su and H. Liu, "Binocular Camera-based a Docking System for an Amphibious Spherical Robot," *Proceedings of 2018 IEEE International Conference on Mechatronics and Automation (ICMA 2018)* (2018) pp. 1621–1626. doi:10.1109/ICMA.2018.8484518.

[20] Q. Zhang, Q. Jia and H. Sun, "Monocular Vision-based Path Identification of a Spherical Robot," *2010 8th World Congress on Intelligent Control and Automation* (2010) pp. 6703–6707. doi:10.1109/WCICA.2010.5554150.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.* **60**(2), 91–110 (2004). doi: 10.1023/B:VISI.0000029664.99615.94.

[22] X. Xu, L. Tian, J. Feng and J. Zhou, "OSRI: A rotationally invariant binary descriptor," *IEEE Trans. Image Process.* **23**(7), 2983–2995 (2014). doi: 10.1109/TIP.2014.2324824.

[23] P. F. Alcantarilla, J. Nuevo and A. Bartoli, "Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces," *BMVC 2013 - Electronic Proceedings of the British Machine Vision Conference 2013* (2013).

[24] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Underst.* **110**(3), 346–359 (2008). doi: 10.1016/j.cviu.2007.09.014.

[25] S. Guo, S. Sun and J. Guo, "The Application of Image Mosaic in Information Collecting for an Amphibious Spherical Robot System," *2016 IEEE International Conference on Mechatronics and Automation* (2016) pp. 1547–1552. doi:10.1109/ICMA.2016.7558794.

[26] J. Rettkowski, D. Gburek and D. Göhringer, "Robot Navigation Based on an Efficient Combination of an Extended A∗ Algorithm, Bird's Eye View and Image Stitching," *2015 Conference on Design and Architectures for Signal and Image Processing (DASIP)* (2015) pp. 1–8. doi:10.1109/DASIP.2015.7367240.

[27] Q.-Z. Li, Y. Zhang and F.-N. Zang, "Fast multicamera video stitching for underwater wide field-of-view observation," *J. Electron. Imaging* **23**(2), 023008 (2014). doi: 10.1117/1.JEI.23.2.023008.

[28] W.-C. Hu and D.-Y. Huang, "Multicamera setup-based stitching scheme for generating wide fov video without ghost," *Int. J. Innov. Comput. Inf. Control* **5**(11), 4075–4088 (2009).

[29] H. Ullah, O. Zia, J. H. Kim, K. Han and J. W. Lee, "Automatic 360° mono-stereo panorama generation using a cost-effective multi-camera system," *Ah S Sens.* **20**(11), 3097 (2020). doi: 10.3390/s20113097.