3D Scene Localization and Mapping Based on Omnidirectional SLAM

Xilin Ke\textsuperscript{1,2,a}, Fei Huang\textsuperscript{3,b}, Yi Zhang\textsuperscript{3,c}, Zhiyong Tu\textsuperscript{4} and Weiwei Song\textsuperscript{4}

\textsuperscript{1}State Key Laboratory of Geo-Information Engineering, Xi'an, Shanxi, 710054, China
\textsuperscript{2}Xi'an Research Institute of Surveying and Mapping, Xi'an, Shanxi, 710054, China
\textsuperscript{3}School of Geodesy and Geomatics, Wuhan University, Wuhan, Hubei, 430079, China
\textsuperscript{4}GNSS Research Center, Wuhan University, Wuhan, Hubei, 430079, China
\textsuperscript{a}e-mail: chauchl@163.com
\textsuperscript{b}Corresponding author’s e-mail: huangfei28@whu.edu.cn
\textsuperscript{c}Corresponding author’s e-mail: yzhang@sgg.whu.edu.cn

Abstract: The vision SLAM technology is the key technology of robot navigation and unmanned driving. In order to further study the accuracy of multi-view SLAM positioning and mapping applications, this paper combines omnidirectional imaging and SLAM technology, improves the existing technical model in the field of omnidirectional imaging, studies the pose map optimization and SLAM overall adjustment technology based on object constraints and global consistency, and carries out omnidirectional SLAM point cloud reconstruction and 3D mapping. Through the time synchronization experiment of LadyBug 5 Plus and Femtomes Mini2-D-INS, the two positioning results of the system and the GNSS measurement results are compared and analysed.

1. Introduction

SLAM (Simultaneous Localization And Mapping) is a leading technology in the field of spatial information. SLAM refers to the carrier carrying sensors to establish a description model of the environment while moving, and estimate its own motion. It can use its own positioning to obtain the surrounding space information in unknown environment, complete the accurate estimation of its own pose, and realize the mapping process of the surrounding detection area.

At present, the outdoor positioning, navigation and mapping technology has achieved fruitful results. It can provide high-precision navigation and positioning services in smart city, logistics, big data, Internet of things and other industries. In the field of remote sensing photogrammetry and computer vision, such as geographic conditions monitoring, digital city modeling, natural disaster assessment, unmanned driving and other intelligent robots are widely used, the use of global navigation and positioning system for autonomous navigation has also been widely applied and studied, and the business model has been basically mature. But in some environment, such as tree or high rise buildings sheltered area, the positioning accuracy is low. It is difficult to achieve better autonomous navigation and positioning based on GNSS and other positioning and navigation methods. The sensor cannot rely on this kind of sensor to complete the high-precision positioning and mapping task. Therefore, when carrying out visual detection, it relies on its own sensors to carry out...
autonomous positioning and construction of the surrounding environment map through the detection of the surrounding environment. Using the constructed three-dimensional scene map to realize its own position optimization, improve its own positioning accuracy and mapping quality, and realize high-precision mobile robot positioning and navigation.

With the improvement of sensor integration and the development of image processing technology, vision based SLAM technology has become a new research hotspot in the field of robot theory and application. Vision sensors can obtain the feature information and texture information of the detection environment, and the cost is low, and the collected information is rich. Visual SLAM includes collection of sensors’ data, visual odometry or visual inertial odometry in front end, optimization in back end, loop closure in back end and mapping[1]. At present, according to the number of vision sensors, vision based autonomous positioning and mapping can be divided into monocular vision SLAM(MonoSLAM, PTAM,ORB-SLAM, SVO)[2-5]and multi-view vision SLAM(LSD-SLAM, multicol-SLAM)[6-7]. Monocular vision SLAM uses a single camera for mobile robot navigation, its camera field of view is small, only through the forward and backward motion of multi frame image data to obtain the depth feature information of the surrounding environment, and cannot obtain the absolute depth information of the surrounding objects. Monocular visual SLAM has problems such as necessary initialization, scale ambiguity and scale drift[8]. Multi-view vision SLAM can obtain the information of the same scene from different perspectives, and calculate the corresponding image through the images collected at the same time In order to obtain the 3D information of the surrounding environment through triangulation, the parallax of the pixel is calculated in a large amount and the system design is complex. However, whether it is monocular vision SLAM or multi-view vision SLAM, its basic idea process is consistent, that is, through the feature information on the image to locate its own pose and realize the mapping of the surrounding environment.

The main problem of traditional monocular or multi-view vision SLAM technology in positioning and mapping is that the field of view is limited, and the matching area between frames are small, which affects the final positioning and mapping accuracy. At the same time, the scene coverage is not complete, so it is unable to complete a full range of fine mapping. In order to solve this problem, many scholars have been engaged in improve the field of view of SLAM. Kangni et al. [9] proposed a panoramic image pose calculation method. It uses paired basic matrices to restore the position of the panoramic camera in the scene, and optimizes the pose through bundle adjustment. Li et al. [10] presented a SLAM system based on a spherical model for full-view images. Seok et al. [11] presented robust omnidirectional visual odometry for wide-FOV camera systems(ROVO).PAN-SLAM implements a panoramic visual SLAM based on a multi camera system[12]. Caruso et al. [13] proposed large-scale direct SLAM for omnidirectional cameras based on LSD-SLAM. Liu et al. [14]. The articles [15-18] propose better methods for some theories of panoramic images, such as calibration, synthesis, imaging models, etc.

The vision SLAM technology is the key technology of robot navigation and unmanned driving. In the field of surveying and mapping, SLAM technology can also play its technical advantages to complete positioning and mapping. In order to further study the accuracy of multi-view SLAM positioning and mapping applications, this paper combines omnidirectional imaging and SLAM technology, improves the existing technical model in the field of omnidirectional imaging, studies the pose map optimization and SLAM overall adjustment technology based on object constraints and global consistency, and carries out omnidirectional SLAM point cloud reconstruction and 3D mapping. Omnidirectional image combined with SLAM three-dimensional mapping technology, with omnidirectional perspective, to achieve a comprehensive and uniform access to the surrounding environment information, improve the integrity of scene data, make up for the shortcomings of monocular SLAM positioning and orientation inaccurate, overcome the requirements of multi-objective SLAM imaging at the same time, will become the main technical means of environment rapid and high-precision reconstruction and mapping.
2. General SLAM Model Description

SLAM technology refers to the use of the observation data collected by the sensor to complete the positioning of the sensor itself and the construction of the surrounding environment map, which can be described as follows: in an unfamiliar environment, the mobile robot carrying the sensor uses the observation data of the sensor to determine the feature point information in the surrounding environment and its own three-dimensional coordinates.

SLAM problem can be described by two equations, namely motion equation (formula 1) and observation equation (formula 2).

\[
x_k = f(x_{k-1}, u_k, w_k) \tag{1}
\]

\[
z_{k,j} = h(y_j, x_k, v_{k,j}) \tag{2}
\]

In the equation of motion (1), the subscript \( k \) indicates the sequence number of the current time, \( k - 1 \) represents the last moment, \( u_k \) is the sensor reading, \( w_k \) is noise. \( x_k \) represents the position of the sensor at the current time and is a three-dimensional vector. \( x_{k-1} \) indicates the position of the sensor at the previous time.

In the observation equation (2), the subscript \( j \) indicates the serial number of the currently observed road mark, \( y_j \) is the road mark observed by the sensor at the position \( x_k \), which is also a three-dimensional vector. \( z_{k,j} \) is the observation data corresponding to the road marking point \( y_j \). \( v_{k,j} \) is observation noise.

These two equations are the most basic equations in SLAM problem motion equation and observation equation, which describe the motion model and observation model of sensor in SLAM problem. Therefore, this problem can be abstracted as: when we know the reading \( U \) of the motion measurement and the reading \( Z \) of the sensor, how to solve the positioning problem (estimate \( x \)) and the mapping problem (estimate \( y \)). At this time, we model the SLAM problem as a state estimation problem: how to estimate the internal and hidden state variables from the measurement data with noise, and which method can minimize the impact of noise and make the state estimation solution optimal.

This paper mainly studies the localization problem of omnidirectional SLAM, that is, in the above state estimation problem, we solve the \( X \) vector - the position and attitude of the omnidirectional camera, and make full use of the omnidirectional camera's wide range of perspectives to optimize the \( X \) vector.

3. Omnidirectional SLAM Algorithm Framework

The algorithm framework of classic vision SLAM is: First, input the data of vision sensor, including video data and image data. Secondly, feature extraction and matching are performed on the image data. According to the principle of minimizing the re projection error, the transformation matrix \( t \) (including rotation matrix \( R \) and translation vector \( t \)) is calculated to estimate the pose change of the camera. At the same time, the local map and the initial pose map are constructed. Then in the back-end optimization, considering the loop information, the nonlinear optimization method is used to optimize the transformation matrix \( T \) and the three-dimensional coordinate \( X \) of the road marking point at the same time. Finally, a sparse 3D point cloud image is generated.

Compared with the classical SLAM algorithm framework, the SLAM algorithm based on omnidirectional vision faces some problems: the distortion of spherical image will make feature extraction and matching difficult; the mapping relationship between the pixel coordinates and camera coordinates of planar image is not suitable for spherical; the polar constraint method of planar image is not suitable for spherical. Therefore, it is necessary to transform from pixel coordinates to camera coordinates. The plane image is described by internal parameter matrix, while the omnidirectional image is a sphere, and the mapping relationship between pixel coordinates and camera coordinates needs to be described by longitude and latitude expression.
In this paper, we use Ladybug 5 Plus image data for image mosaic experiment. Ladybug 5 Plus is a multi-lens omnidirectional camera, in which the transformation parameters of each lens relative to the overall coordinate system are known. See Table 1 and table 2 for details.

**Table 1. Internal parameters of each lens of Ladybug 5 Plus**

| Camera   | \( f_x \)     | \( f_y \)     | \( c_x \)     | \( c_y \)     |
|----------|----------------|----------------|----------------|----------------|
| Camera 0 | 617.781691     | 618.239261     | 508.640954     | 606.697584     |
| Camera 1 | 615.130853     | 615.464415     | 503.555405     | 609.374047     |
| Camera 2 | 613.651108     | 613.854014     | 505.300657     | 606.367852     |
| Camera 3 | 613.619999     | 613.601457     | 505.394312     | 605.285783     |
| Camera 4 | 617.111171     | 617.314511     | 509.326311     | 602.685731     |
| Camera 5 | 614.836368     | 614.164147     | 496.181509     | 608.088842     |

**Table 2. Transformation parameters of each lens of Ladybug 5 Plus**

| Camera   | \( x \)       | \( y \)       | \( z \)       | \( \phi \)   | \( \omega \)  | \( k \)     |
|----------|---------------|---------------|---------------|--------------|--------------|------------|
| Camera 01| -0.090394     | 0.054672      | -0.126703     | 2.24474      | -72.0421     | -1.90352   |
| Camera 12| -0.093454     | -0.027608     | -0.144855     | -3.37736     | -71.5384     | 2.59948    |
| Camera 23| -0.087874     | 0.036924      | -0.080582     | 0.215645     | -72.5588     | 0.42486    |
| Camera 34| 0.049623      | -0.305582     | -0.287426     | -6.74614     | -73.0177     | 6.49319    |
| Camera 40| -0.255537     | 0.230472      | -0.183380     | 0.529042     | -70.6951     | 1.28064    |

According to the parameters provided in Table 1, the internal parameter matrix of each camera can be constructed as follows:

\[
K = \begin{pmatrix}
    f_x & 0 & c_x \\
    0 & f_y & c_y \\
    0 & 0 & 0
\end{pmatrix}
\]  

(3)

Through the internal parameter matrix \( K \), the pixel coordinates of the image are converted into their respective camera coordinates. See the following table for the conversion relationship.

\[
\begin{pmatrix}
    X_c \\
    Y_c \\
    Z_c
\end{pmatrix} = s \cdot K^{-1} \begin{pmatrix}
    u \\
    v \\
    1
\end{pmatrix}
\]  

(4)

According to the transformation parameters (including three translation parameters and three Euler angles) provided in table 1, the transformation matrix can be constructed as follows:
\[ T = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} \]  

(5)

Where the dimension of the matrix \( t \) is; the matrix \( R \) is the rotation matrix, which can be calculated from the Euler angle provided in table 1; the vector \( t \) is the translation vector, which is composed of \( X \), \( y \) and \( Z \) in table 2. Then the coordinates in the overall coordinate system of the omnidirectional camera can be calculated by the transformation matrix \( T \), see the formula.

\[
\begin{pmatrix}
X_A \\
Y_A \\
Z_A
\end{pmatrix} = T^{-1} \begin{pmatrix}
X_C \\
Y_C \\
Z_C
\end{pmatrix}
\]  

(6)

At this point, the pixel coordinates of each sub camera can be converted to its own sub camera coordinates, and then sub camera coordinates can be converted to omnidirectional camera coordinates. Next, when splicing the omnidirectional image, project it onto the sphere to calculate the longitude and latitude, and then map it onto the plane image according to the longitude and latitude expression method. The flow chart is shown in Figure 2.

\[ \text{Figure 2. Coordinate Transform from sub camera to omnidirectional image} \]

\[ \text{Figure 3. Omnidirectional coordinate transform result} \]

4. Omnidirectional SLAM Experiment and Analysis

4.1. Experimental Platform

We use the Agile Robots car as a carrier, and the Ladybug omnidirectional camera and integrated navigation device are installed on the car. Among them, the integrated navigation equipment is "Femtomes Mini2-D-INS", which uses dual antennas to assist the initial alignment of the inertial navigation system. The device uses RTK ins loose combination real-time filtering algorithm to output high-precision position information, which is used to evaluate the accuracy of omnidirectional SLAM. In addition, the time synchronization device is ublox m8t, and the horizontal spacing of the thumb
antennas on both sides is 0.5m. The overall structure of the system is shown in the figure below.

![Figure 4. Structure of the experimental platform](image)

### 4.2. Experimental Data and Preprocessing

This paper uses the omnidirectional image synthesized by 6 lenses of Ladybug 5Plus for experimentation. Since the lens will inevitably capture the camera itself, but such dynamic objects can easily affect the accuracy of SLAM, so we must find a way to eliminate them. Therefore, we made a mask for the experimental image (as shown in the figure below), so that our omnidirectional SLAM system does not track the black part of the mask, which improves the accuracy of the experimental results to a certain extent.

![Figure 5. Original omnidirectional images](image)

![Figure 6. Using mask to avoid tracking the robot body](image)

![Figure 7. Omnidirectional images after mask overlayed](image)
4.3 Omnidirectional SLAM Result

Omnidirectional images have the advantages of a wide range of viewing angles and a large number of feature points. Therefore, the use of omnidirectional images for SLAM can track more feature points more stably, and obtain better positioning and mapping effects. As shown in the figure below, by superimposing the system's result image with the satellite image, it can be found that the trajectory is basically consistent.

![Figure 8. Overlay image of satellite image and omnidirectional SLAM results](image)

In order to further verify the advantages of the omnidirectional SLAM system, this paper also does an additional set of experiments: perform monocular SLAM on the image of the camera 0 corresponding to each frame of the omnidirectional image. The experimental results show that the monocular SLAM fails to track at the 896th frame, as shown in the figure below. Obviously, this is caused by the narrow viewing angle of the monocular camera. The omnidirectional SLAM has never failed to track. As shown in Table 3, the omnidirectional SLAM has a faster initialization speed, and the initialization is successful after only 136 frames, while the monocular SLAM uses 233 frames. On the other hand, the number of initial map points of omnidirectional SLAM (255 points) is also more than that of monocular SLAM (147 points).

![Figure 9. Monocular SLAM results of camera 0](image)

![Figure 10. Statistical graph of reprojection errors](image)

|                     | Camera 0 | Omnidirectional |
|---------------------|----------|-----------------|
| Initial Frames      | 233      | 136             |
| Initial Map Points  | 147      | 255             |
| Lost Frames         | 896-5727 | None            |

In order to evaluate the internal coincidence accuracy of the omnidirectional SLAM system, we reprojected the landmark points observed in each frame into the image, and calculated the difference between its pixels coordinates and the pixel coordinates of the matched feature points. Each frame has many feature points. We draw the average, standard deviation, maximum, and minimum of the reprojection error of all points in the current frame into the following figure. It can be seen from the figure that the average reprojection error is 15 pixels.
4.4. Accuracy Analysis with GNSS trajectory

The trajectory output of the omnidirectional SLAM is compared and analyzed with the trajectory of GNSS. The results are shown in Table 4 and Figure 10. According to the results in Table 4, the root mean square error (RMSE) of omnidirectional SLAM is 2.411m, which is not bad, but it is not a satisfactory result. Further analysis results, as shown in Figure 10 a) and c), the trajectory of the omnidirectional SLAM and the GNSS trajectory are very close at most of the time, and most of the errors are within 1m. Looking at Figure 10 b), it can be seen more intuitively that the time when the error distribution is about 1.4m is the largest. It can be inferred that a part of the time when the error is too large is an abnormal value, which is probably caused by the unstable GNSS signal, which makes the result difference at a small part of the time too large. Figure 10 d) shows the time-varying errors in the three directions of XYZ. It can also be seen intuitively that, except for the larger error at the initial time, the trajectories of the two are basically the same at other times.

| Index | rmse | mean | median | std  | min  | max  |
|-------|------|------|--------|------|------|------|
| Value(m) | 2.411 | 2.078 | 1.680 | 1.223 | 0.096 | 6.867 |

Figure 11. Comparison of omnidirectional SLAM trajectory and GNSS trajectory
4.5. Loop Closure Error Analysis

The data collected in the experiment constitutes a loop, and the omnidirectional SLAM system also successfully detected the loop. As shown in the figure below, the loop was detected at the 563th and 5302th frame. We compare the positioning results of these two frames to further analyze the positioning accuracy of omnidirectional SLAM.

![Loop closure detected by omnidirectional SLAM system](image)

First, we extract the SLAM positioning results of the 563th frame and the 5302th frame, as shown in Table 5, and calculate the difference in the XYZ direction and fill in the second row of Table 7. The coordinates are under the SLAM system, and the camera coordinates of the first frame are used as the reference system. Align the coordinate system with the WGS-84 coordinate system, and calculate its scale factor to be 1.3 (this coordinate system is smaller than the real world scale).

Next, according to the time synchronization device, the GNSS time at which each frame of the omnidirectional image was taken is acquired. After inquiry, the GNSS time corresponding to the 563th frame and the 5302th frame are respectively 373349.22 and 373665.15 (both in the 2152 week) of the GNSS time. The WGS-84 coordinates of the corresponding time are extracted, as shown in Table 6, and the difference in the XYZ direction is calculated and filled in the third row of Table 7.

Finally, multiply the difference in the XYZ direction of the omnidirectional SLAM result by its scale factor, and make the difference with the WGS-84 coordinate system (the calculation formula is similar to \( error = SLAM_{dx} \times SLAM_{scale} - WGS84_{dx} \times WGS84_{scale} \)), the fourth row of Table 7 is calculated. It can be seen from Table 7 that there is a certain gap between the closed-loop error calculated by the omnidirectional SLAM and the real closed-loop error (take the GNSS result as the true value), but the difference is not much, within 2m.

| Frame ID | X(m) | Y(m) | Z(m) |
|----------|------|------|------|
| 563      | 0.739| 0.074| 3.292|
| 5302     | 0.717| 0.076| 3.189|

| Time stamp | X(m)     | Y(m)     | Z(m)     |
|------------|----------|----------|----------|
| 373349.22  | -2267754.982 | 5009385.517 | 3220929.495 |
| 373665.15  | -2267755.115 | 5009385.106  | 3220930.792 |

| dx(m) | dy(m) | dz(m) | scale |
|-------|-------|-------|-------|
| 0.021 | 0.002 | 0.103 | 1.3   |
| 0.132 | 0.411 | 1.297 | 1.0   |
5. Conclusions
This paper improves the monocular SLAM system and extends SLAM to omnidirectional images. Due to the wider range of the omnidirectional image and the larger number of feature points, the omnidirectional SLAM system in this paper has higher robustness and accuracy. Compared with the traditional monocular SLAM system, this paper studies the imaging model of the omnidirectional spherical image, and also adapts the omnidirectional spherical model in the back-end optimization of SLAM. In the process of realizing simultaneous positioning and mapping, since there are more feature points than monocular images, the density of map points is also greater. Finally, through the time synchronization device, the positioning results of the system and the GNSS measurement results are compared and analyzed. Except for the unstable GNSS signal at the initial moment, the positioning results of the omnidirectional SLAM system are quite different from the GNSS results. At other times, the differences between the positioning results of omnidirectional SLAM and GNSS are within 1m.

Acknowledgments
This research is supported by State Key Laboratory of Geo-Information Engineering, grant number SKLGIE2018-M-4-2, and the Joint Foundation for Ministry of Education of China, grant number 6141A02011907.

References
[1] Gao, X., Zhang, T., Liu, Y. and Yan, Q. (2017) 14 Lectures on Visual SLAM: from Theory to Practice. Publishing House of Electronics Industry, Beijing.
[2] Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O. (2007) MonoSLAM: Real-time Single Camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6):1052-1067.
[3] Klein, G., Murray, D. (2007) Parallel Tracking and Mapping for Small AR Workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan. pp.225-234.
[4] Mur-Artal, R., Montiel, J.M.M., Tardos, J.D. (2015) ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on Robotics. 31(5):1147-1163.
[5] Forster, C., Pizzoli, M., Scaramuzza, D. (2014) SVO: Fast Semi-Direct Monocular Visual Odometry. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China. pp. 15-22.
[6] Engel, J., Schöps, T., Cremers, D. (2014) LSD-SLAM: Large-scale Direct Monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland. pp. 834-849.
[7] Urban, S., Hinz, S. (2016) MultiCol-SLAM-A Modular Real-Time Multi-Camera SLAM System. ArXiv preprint arXiv:1610.07336
[8] Hauke Strasdat, J Montiel, and Andrew J Davison. (2010) Scale Drift-Aware Large Scale Monocular SLAM. Robotics: Science and Systems VI, 2(3):7.
[9] Kangni, F., Laganiere, R. (2007) Orientation and Pose Recovery from Spherical Panoramas. In Proceedings of the 2007 IEEE 11th International Conference on Computer Vision, Rio de Janeiro, Brazil. pp. 1-8.
[10] Li, J., Wang, X., Li, S. (2018) Spherical-Model-Based SLAM on Full-View Images for Indoor Environments. Applied Sciences, 8(11): 2268.
[11] Seok, H., Lim, J. (2019) Rovo: Robust Omnidirectional Visual Odometry for Wide-Baseline Wide-Fovecamera Systems. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada. pp. 6344-6350.
[12] Ji, S., Qin, Z., Shan, J., Lu, M. (2020) Panoramic SLAM from A Multiple Fisheye Camera Rig. ISPRS J. Photogramm. Remote. Sens. pp.159, 169-183.
[13] Caruso, D., Engel, J., Cremers, D. (2015) Large-scale Direct SLAM for Omnidirectional Cameras.
In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany. pp. 141-148.

[14] Liu, P., Heng, L., Sattler, T., Geiger, A., Pollefeys, M. (2017) Direct Visual Odometry for A Fisheye-Stereo Camera. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada. pp. 1746-1752.

[15] Sato, T., Yokoya, N. (2010) Efficient Hundreds-Baseline Stereo by Counting Interest Points for Moving Omni-Directional Multi-Camera system. J. Vis. Commun. Image Represent., 21:416-426.

[16] Parian, J.A., Gruen, A. A Sensor Model for Panoramic Cameras. (2003) In Proceedings of the 6th Optical 3D Measurement Techniques, Zurich, Switzerland. pp. 22-25.

[17] Ikeda, S., Sato, T., Yokoya, N. (2003) High-resolution Panoramic Movie Generation from Video Streams Acquired by An Omnidirectional Multi-Camera System. In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Tokyo. pp. 155-160.

[18] Geyer, C., Daniilidis, K. (1999) Catadioptric Camera Calibration. In Proceedings of the 7th IEEE International Conference on Computer Vision, Corfu, Greece. pp. 398-404.