# Three-Dimensional Visualization System with Spatial Information for Navigation of Tele-Operated Robots

**Seung-Hun Kim** [1,2] **, Chansung Jung** [2] **and Jaeheung Park** [1,3,*]

1   Graduate School of Convergence Science and Technology, Seoul National University, Seoul KR 08826, Korea; ksh1018@snu.ac.kr
2   Intelligent Robotics Research Center, Korea Electronics Technology Institute, Bucheon KR 14502, Korea; chansung0602@keti.re.kr
3   Advanced Institutes of Convergence Technology, Suwon KR16229, Korea
*   Correspondence: park73@snu.ac.kr

check for updates

**Abstract:** This study describes a three-dimensional visualization system with spatial information for the effective control of a tele-operated robot. The environmental visualization system for operating the robot is very important. The tele-operated robot performs tasks in a disaster area that is not accessible to humans. The visualization system should perform in real-time to cope with rapidly changing situations. The visualization system should also provide accurate and high-level information so that the tele-operator can make the right decisions. The proposed system consists of four fisheye cameras and a 360° laser scanner. When the robot moves to the unknown space, a spatial model is created using the spatial information data of the laser scanner, and a single-stitched image is created using four images from cameras and mapped in real-time. The visualized image contains the surrounding spatial information; hence, the tele-operator can not only grasp the surrounding space easily, but also knows the relative position of the robot in space. In addition, it provides various angles of view without moving the robot or sensor, thereby coping with various situations. The experimental results show that the proposed method has a more natural appearance than the conventional methods.

**Keywords:** 3D visualization; wrap around view monitoring; robot vision systems; tele-operated robots

## 1. Introduction

Tele-operated robots that can work on behalf of humans have been actively studied since the Fukushima nuclear disaster. The Defense Advanced Research Projects Agency Robotics Challenge (DRC) has also been held to evaluate the performance of these robots. However, it is still difficult to remotely control robots because many factors must be considered (e.g., narrow spaces, obstacles, road or terrain conditions, weather effects, communications, visible data about the surrounding environment). One of the important factors is visible data. The surrounding environment of the robot must be constructed, and a variety of viewpoints must be visualized in real-time.

One of the methods used to visualize the robot environment is simultaneous localization and mapping (SLAM). SLAM is an algorithm used to estimate the map and the location of a robot using the surrounding information obtained from the robot in an unknown environment. Various sensors are used to obtain the information about the surroundings of the robot for SLAM. Depth sensors [1] (e.g., laser scanners, infrared sensors, lidars, and radars) and vision sensors [2–4] (e.g., mono cameras, stereo cameras, and RGB-D cameras) are typically used for SLAM. In visualizing the environment using these sensors, the angle of view is not 360°. Hence, the person or the robot must rotate and move to accumulate the surrounding data [5–7]. In addition, accumulated error can happen in an area with featureless walls or floors.

Another visualization method is the around view monitor (AVM), which provides the driver or the tele-operator with a virtual bird's-eye view using multiple fisheye cameras. The AVM is mainly applied to vision systems (e.g., parking and tele-operation) where real-time visualization is required for safety or convenience [8,9]. The AVM presents all objects on the floor under the assumption, and hence an unnatural image is generated when obstacles are present. To solve this problem, studies are being performed to improve the user visibility using depth sensors [10].

The wrap-around view monitor (WAVM) [11] has recently been studied to provide the driver or the tele-operator with various viewpoints as well as a bird's eye view. The WAVM maps the images to a fixed three-dimensional (3D) model to visualize the surroundings of the vehicle or the robot. It provides a large field of view and various viewpoints different from the AVM, which provides only a bird's eye view. Various viewpoints are advantageous to the tele-operator because they provide appropriate points of view according to various situations. Therefore, the WAVM is also applied to safety monitoring systems for construction equipment. Existing WAVM studies have employed 3D models, such as hemisphere (Figure 1a) [12–14] or cylinder [15], based on the outdoor environment. The real world differs from these 3D models, and hence the existing WAVM practically provides tele-operators with unnatural images, which can make them feel uncomfortable because it is difficult to perceive the space around the robot.
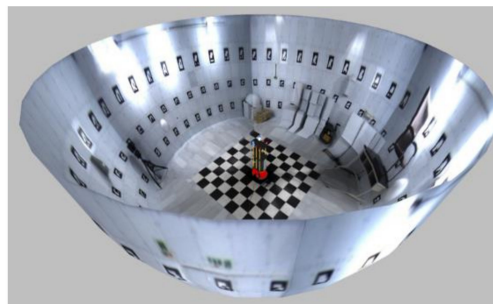


**(a)** WAVM using the hemisphere model [12–14].
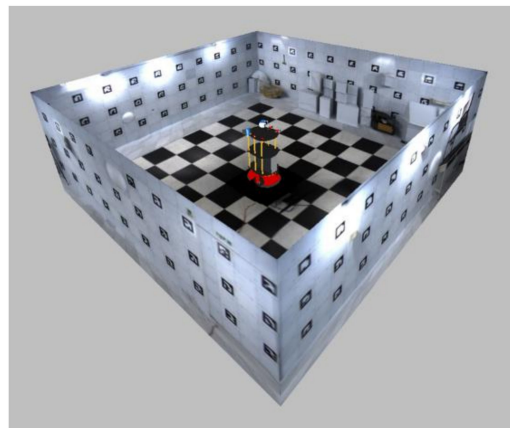


**(b)** Proposed system using the spatial model.

**Figure 1.** Demonstration of the wrap-around view monitor (WAVM).

This study proposes an improved WAVM with a spatial model using four fisheye cameras and a 360° laser scanner. The spatial model is constructed using the spatial information data obtained from a 360° laser scanner. The 360° laser scanner is a widely used sensor for gathering information about the space around a robot [16]. This provides an image of sufficient visibility for operation of the robot, and allows the tele-operator to know the relative position of the robot in space (Figure 1b). It also provides the environment of the robot at various viewpoints without moving the robot or the sensor.

## 2. Proposed Method

Figure 2 shows the schematic of the proposed system consisting of three processes, namely, image stitching, spatial modeling, and image mapping steps.
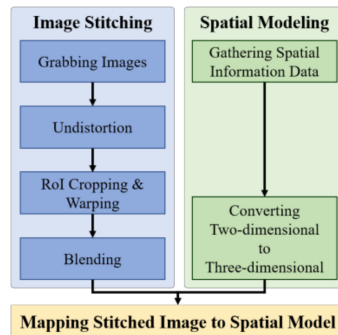


**Figure 2.** Schematic of the proposed system. RoI: region of interest.

### 2.1. Image Stitching

This process is similar to the existing WAVM [9,10], and consists of undistortion, region of interest (RoI) cropping and warping, and blending steps. Image undistortion as well as RoI cropping and warping steps are performed on each image obtained from four cameras. The blending step combines four images to create one image.

#### 2.1.1. Undistortion

The fisheye camera provides an image with a horizontally wide-angle view (over 180°), but it has radial and tangential distortion (Figure 3). The four images with distortion are corrected in this step [11]. Figure 4 shows the geometric model assuming that the lens shape is spherical. The coordinate of the image with distortion is $(x_d, y_d)$, while $F$ is the focal length and the radius of the sphere [12]. $L$ is the distance between the center point of the lens and the corrected point $(x_u, y_u)$ calculated by (2). Accordingly, (1) is obtained using the ratios of the triangles shown in Figure 4, and (3) and (4) for backward matching are obtained by substituting (2) into (1). Applying the backward mapping to the entire image, the undistorted images are obtained as shown in Figure 5.

$$(x_d, y_d) = (x_u F/L, \ y_u F/L) \tag{1}$$

$$L = \sqrt{x_u{}^2 + y_u{}^2 + F^2} \tag{2}$$

$$x_d = x_u F / \sqrt{x_u{}^2 + y_u{}^2 + F^2} \tag{3}$$

$$y_d = y_u F / \sqrt{x_u{}^2 + y_u{}^2 + F^2} \tag{4}$$



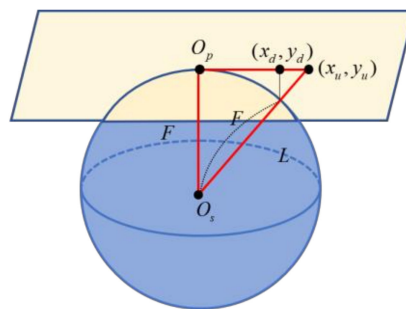**Figure 3.** Fisheye camera image with distortion.

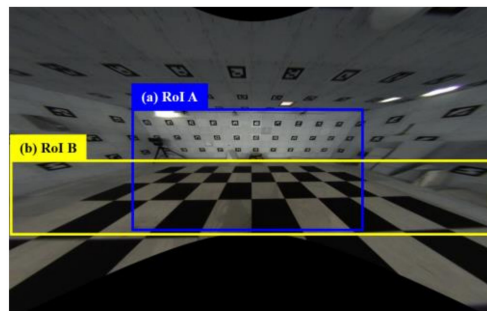**Figure 4.** The spherical geometric model.



**Figure 5.** Undistorted image.

### 2.1.2. RoI Cropping and Warping

Not all regions of the undistorted image are needed for the image stitching, because distorted regions still exist. The two RoI images with a valid meaning in the whole image are cropped out in the cropping step. The valid images are the regions in Figure 5a,b, defined as RoI A and B images, respectively. The RoI A image is mapped to both the floor and the wall because the region of the wall occupying the image differs depending on the distance between the wall and the camera. The RoI B image is always mapped to the floor.

The cropped images are warped using a homography matrix. Homography is the most common model used to describe the two-dimensional (2D) image transformation relationship of planar objects. Homography is defined in a homogeneous coordinate system. Equation (5) is a matrix of perspective transformations on a homogeneous coordinate system.

$$w \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5}$$

The scale of homography cannot be determined because $(x, y, 1)$ and $(wx' \ wy', w)$ are homogeneous coordinates. Therefore, $h_{33}$ is set to 1, and the remaining eight unknowns are calculated. We use a checkerboard and four matching pairs to calculate the homography matrix. Figure 6 shows the cropped and warped result images of Figure 5a,b. We determine the relationship between the source image and the result image in this step.
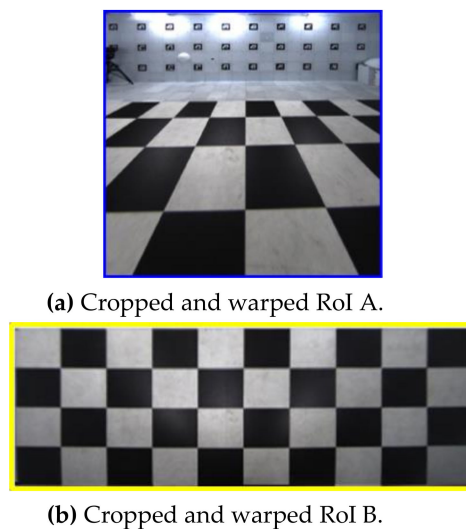
**(a)** Cropped and warped RoI A.



**(b)** Cropped and warped RoI B.

**Figure 6.** Cropped and warped images of RoIs A and B in Figure 5.

### 2.1.3. Blending

In this step, two blended and stitched images are obtained using eight cropped and warped RoI A and B images in Figure 6. At this time, the region where the two images overlap appears unnatural. We blend this region to make it look natural by assigning different weights to each pixel according to the distance in the overlapping regions [13,14]. The value of pixels is computed with (6) [15], which loops over each row in the warped image:

$$I_w(x_i, y_i) = (d_i/l) \cdot I_1(x_i, y_i) + (1 - d_i/l) \cdot I_2(x_i, y_i) \tag{6}$$

where $l$ is the length of the line that overlaps each row, and $d_i$ is the distance from the starting overlapping pixel to the current pixel $x_i$. $I_1$ and $I_2$ denote the two images to perform image blending. $I_w$ is the weighted sum. The longer $d_i$ is, the greater the weight. Figure 7 shows the stitching results for RoI A and B images.



**(a)** Four RoI A images blended and stitched.



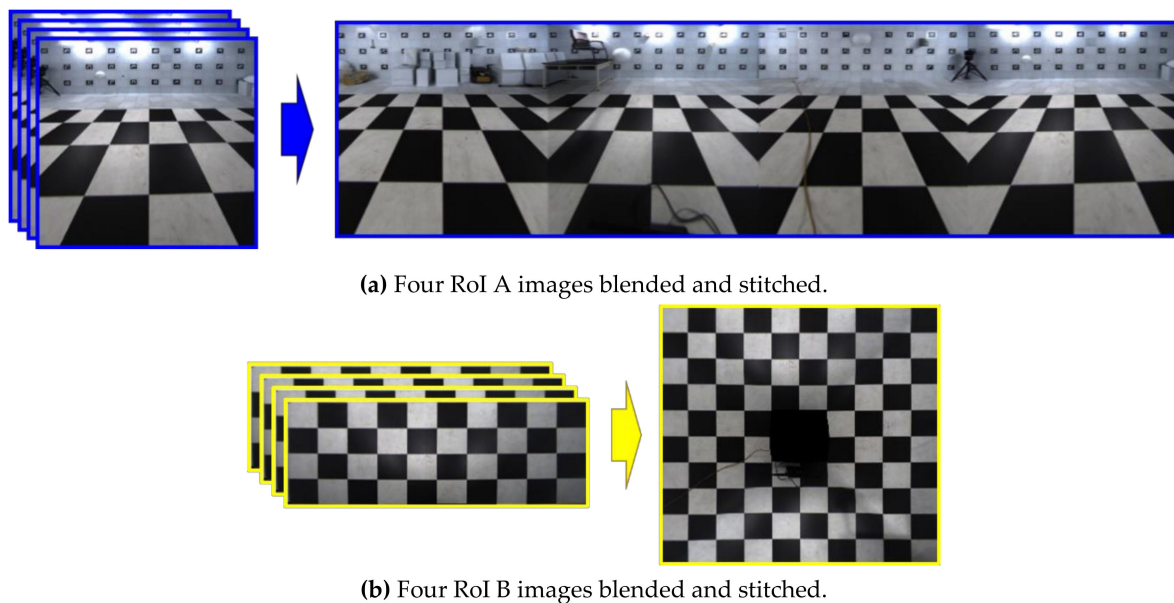**(b)** Four RoI B images blended and stitched.

**Figure 7.** Blended and stitched RoI A and B images using the four images in Figure 6.

2.1.4. Acceleration for Image Stitching

The above-mentioned steps require multiple matrix operations on the number of pixels in all source images to obtain two images (Figure 7). This is a factor that increases the computational time. To overcome this, we employ a lookup table (LUT) [16] instead of matrix operations. LUT denotes the data structures in computer science that are typically associative arrays, and are often used to replace runtime calculations with simple array indexing. Fetching values from memory is faster than through computation or I/O functions. In the proposed system, the relationship between the input and result images is fixed and stored in the LUT. Through the LUT, the pixel values of the result images are immediately known as the pixel values of the input images.

*2.2. Spatial Modeling*

The spatial modeling process is constructed using the spatial information data of the space. This process consists of gathering spatial information data and converting 2D points to 3D points.

2.2.1. Gathering Spatial Information Data

In this step, the measurement data are first obtained using a 360° laser scanner to make the 2D spatial information. Figure 8 shows the coordinate representation according to the measurement points $(p_{xn}, p_{yn})$ calculated by the spatial information data, which are distances $d_n$ and angles $\theta_n$ in (7) and (8) on the 2D coordinate. $O$ indicates the position of the center of the laser scanner.

$$p_{xn} = d_n \cdot \sin(\theta_n) \tag{7}$$

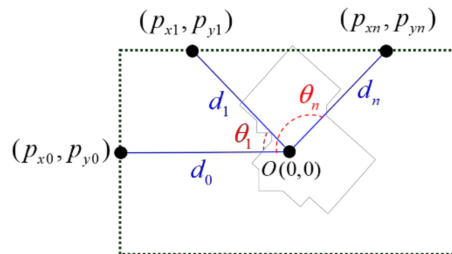$$p_{yn} = d_n \cdot \cos(\theta_n) \tag{8}$$



**Figure 8.** Coordinate representation according to spatial information.

The measurement points may have noise caused by the diffused reflection. Hence, we use a noise filter that is robust, locally weighted, and scatterplot-smoothing [17].

2.2.2. Converting 2D Points to 3D Points

In this step, two 3D points are generated at one measurement point obtained in the sampling spatial information step, assuming that the wall and the floor are vertical. The *z*-axis values of the 3D points are generated using (9) and (10) to make the 3D spatial model:

$$p_{zbn} = B \cdot D \cdot d_n / \sin(\theta_n) \tag{9}$$

$$p_{ztn} = T \cdot D \cdot d_n / \sin(\theta_n) \tag{10}$$

where $p_{zbn}$ and $p_{ztn}$ are the *z*-axis values at the bottom and top points of the spatial model, respectively. *B*, *T*, and *D* indicate the height of the boundary between the wall and the floor, the height of the boundary between the wall and the ceiling, and the distance between the robot and the walls, respectively. $P_{tn} = (p_{xn}, p_{yn}, p_{ztn})$ and $P_{bn} = (p_{xn}, p_{yn}, p_{zbn})$ on the 3D coordinates are obtained by (7)–(10) as shown in Figure 9.
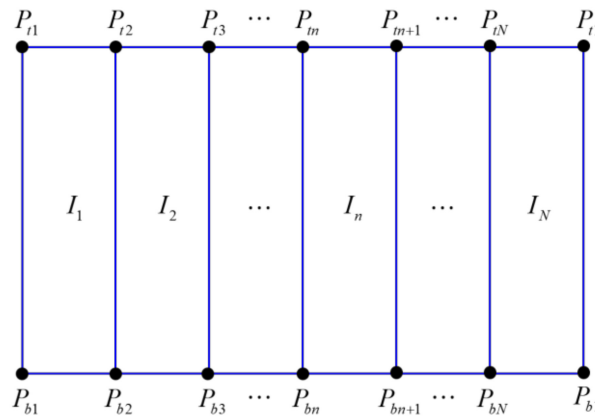
**Figure 9.** Splitting image $I$ with three-dimensional points by Equations (7)–(10).

*2.3. Mapping the Stitched Image to the Spatial Model*

This process involves mapping the resulting image obtained by the image stitching process to the spatial model obtained by the spatial modeling process. First, the stitched RoI B image in Figure 7b is mapped to the floor region (yellow quadrangle) of the spatial model, as shown in Figure 11a. The stitched RoI B image is used without further conversion. Second, the image $I$, which is the stitched RoI A image in Figure 7a, is split into $N$ images (Figure 9). Image $I$ is expressed as $I \supset \{I_1, I_2, \ldots, I_n, \ldots, I_N\}$. $N$ is the resolution of the 360° laser scanner. For example, 720 split images are obtained if $N$ is 720. The split images are sequentially mapped to the walls and floors of the spatial model. When mapping the image, it is divided into a far wall and a near wall. It is a near wall if $d_n$ is lower than $D$. It is a far wall if $d_n$ is higher than $D$. In the case of a near wall, the split image $I_n$ is mapped to the wall region (blue region) of the spatial model (Figure 11b). In the case of a far wall, the split image $I_n$ is mapped to the wall region (upper blue region) and the floor region (lower red region) divided by the two points $P_{Bn}$ and $P_{Bn+1}$ (Figure 10). The two points, $P_{Bn}$ and $P_{Bn+1}$, are the boundary points between the wall and the floor. The image $I_{n+2}$ in Figure 11c shows the mapping of the far wall and the floor to the spatial model. Figures 1b and 12 show the result images for the spatial model for the room and the corridor, respectively.
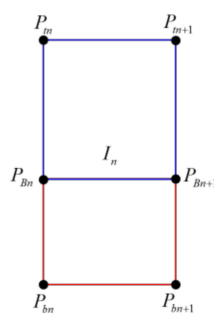


**Figure 10.** Splitting wall and floor images in $I_n$. The upper region (blue line) of the divided image is mapped to the wall, while the lower region (red line) is mapped to the blank floor.
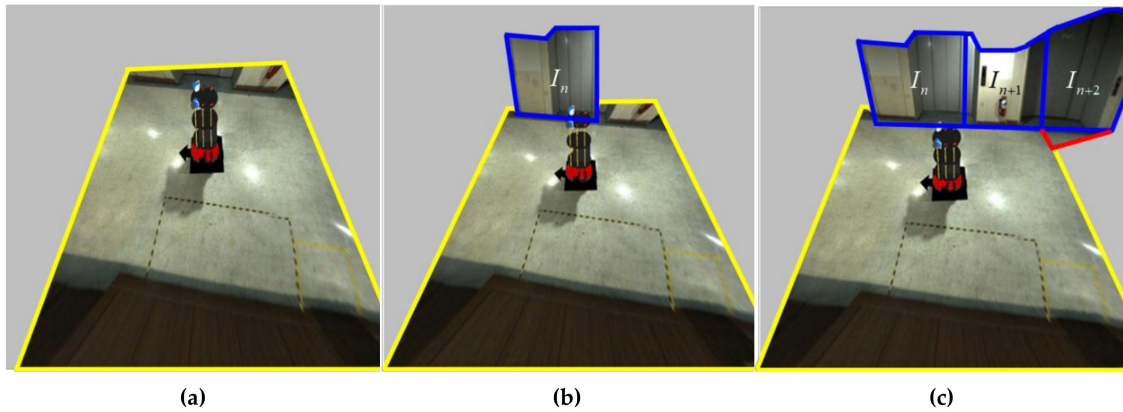
**(a)** **(b)** **(c)**

**Figure 11.** An example of mapping the stitched RoI A and B images to the spatial model. The regions of the yellow and blue lines are the RoI B and RoI A images in Figure 7. The region of the red line is the same as the region of the red line in Figure 10.
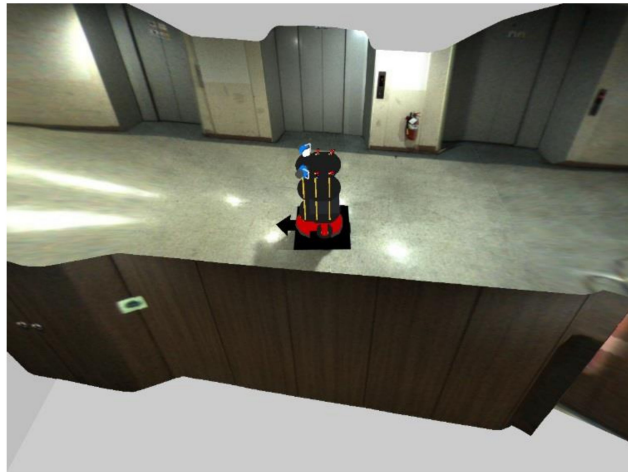


**Figure 12.** Results of mapping the stitched image to the spatial model in the corridor.

## 3. Experimental Results

In this section, subjective and objective comparative experiments were conducted to evaluate the visibility among SLAM, the existing WAVM, and the proposed system. In our system, four fisheye cameras were mounted on the upper part of the robot such that they had the same height, as shown in Figure 13. A 360° laser scanner was mounted on the center of the four cameras. In this experiment, we used a computer with CPU 3.80 GHz and 8 GB RAM. On average, the proposed system took approximately 55 ms (18 fps) to generate the result image without a GPU for the embedded system. Processing could be accelerated if we applied parallel processing on the GPU.
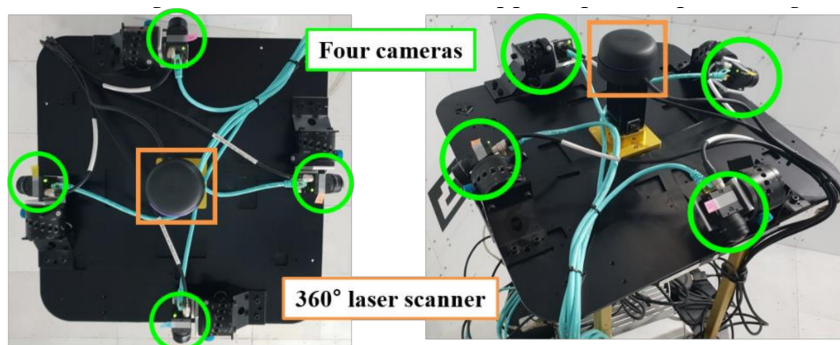


**Figure 13.** Configuration of four cameras and a 360° laser scanner on the robot.

### 3.1. Subjective Comparative Experiment

For the subjective evaluation, we configured a corridor experiment to compare the proposed system with the existing WAVM [10]. Figure 14 shows a c-shaped corridor drawing with a length of 40 m and a width of 3 m, as well as the location and orientation of the robot in the corridor. The yellow rectangles (**A**, **B**, and **C**) indicate the location of the robot. The arrow denotes the direction of the robot's movement. Location **A** is an L-shaped corner in the corridor; **B** is close to the left wall; and **C** is located in the center of the I-shaped corridor. Figure 15 compares the proposed system with the existing WAVM. Figure 15a–c show the results in locations **A**, **B**, and **C**, respectively, in Figure 14. As shown in the left image of Figure 15a, it is difficult to recognize the corridor shape in the existing WAVM result image. However, as a result of the proposed system in Figure 15a (right), it is easy to recognize that the robot is in the L-shaped corridor. Likewise, the distance between the robot and the left and right walls can be easily determined, as shown in the right-side images of Figure 15b,c.
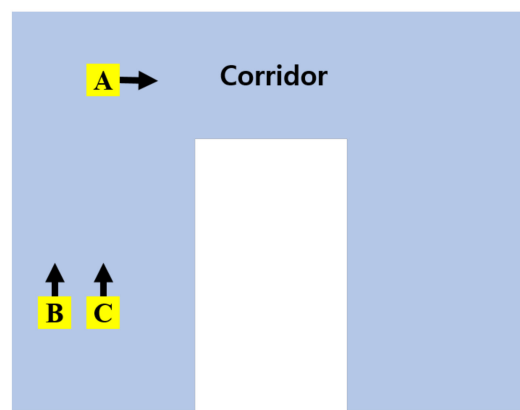


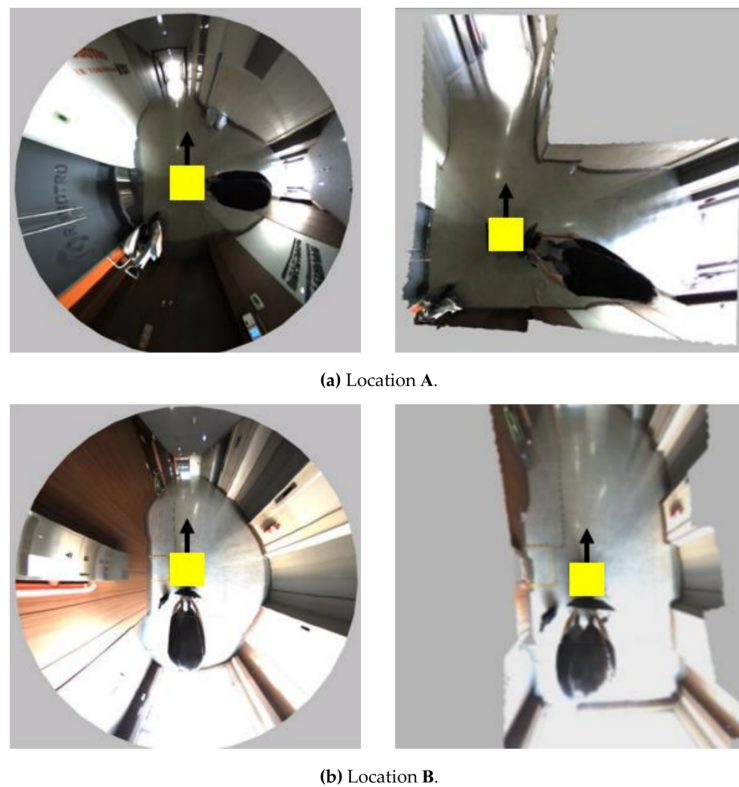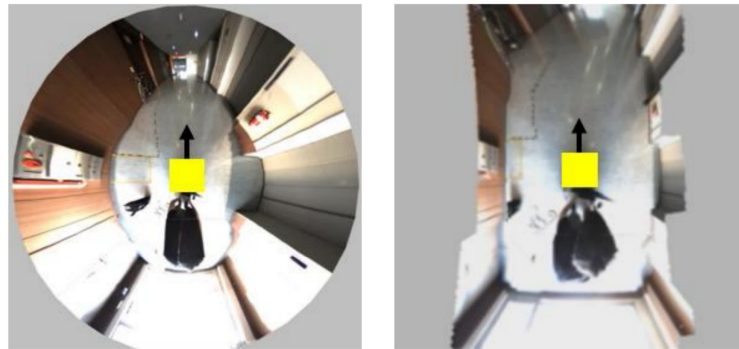**Figure 14.** Robot's location and direction (**A–C**) in the corridor drawing.



**(a)** Location **A**.



**(b)** Location **B**.

**Figure 15.** *Cont.*

(c) Location **C**.

**Figure 15.** Comparison of the experimental results between the existing WAVM (left) [10] and the proposed system (right) at locations (**A**–**C**) in Figure 14.

We also constructed the testbed in Figure 1b to evaluate the visibility of the existing WAVM, SLAM [17–19], and the proposed system. The dimension of the testbed was 6 m × 6 m × 2.1 m. Each wall had 30 markers, with a total of 120 markers. The marker measured 18 cm × 18 cm. Figure 16a shows the ground truth image taken by the camera. Figure 16 compares the result images obtained by the three methods (i.e., SLAM, existing WAVM, and the proposed system) in the same viewpoint and location. Figure 16b depicts the result image of the real-time appearance-based mapping SLAM [17–19] using a stereo camera. The result image in SLAM was not natural because the mismatched feature points appeared when the stereo camera moved to accumulate data. Figure 16c,d depict the result images of the existing WAVM and the proposed system, respectively. The red lines in each figure represent the lines connecting the bottom parts of the markers on the wall. The lines in (b) and (d) are straight lines, as shown in (a). In contrast, the line in (c) is a curve. The existing WAVM result image (c) was distorted because it did not know the spatial information of the testbed that the wall was vertical and flat.
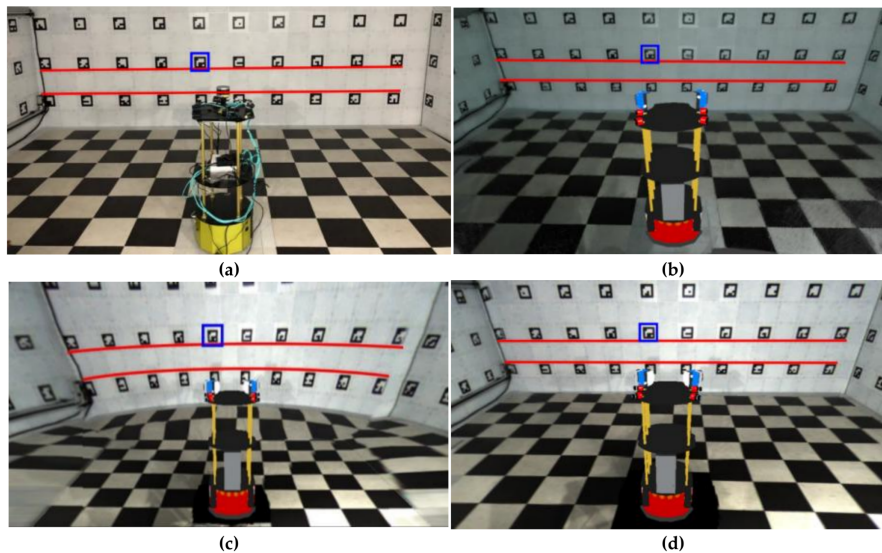


**Figure 16.** Resulting images in various and complex environments.

Figure 17 shows the resulting images in various and complex environments. Figure 17a depicts that the surrounding environment was well represented for curved wall surfaces. In the space of 12 m or more, it was expressed as shown in Figure 17d since the visibility was better when using the bowl model than the spatial model.
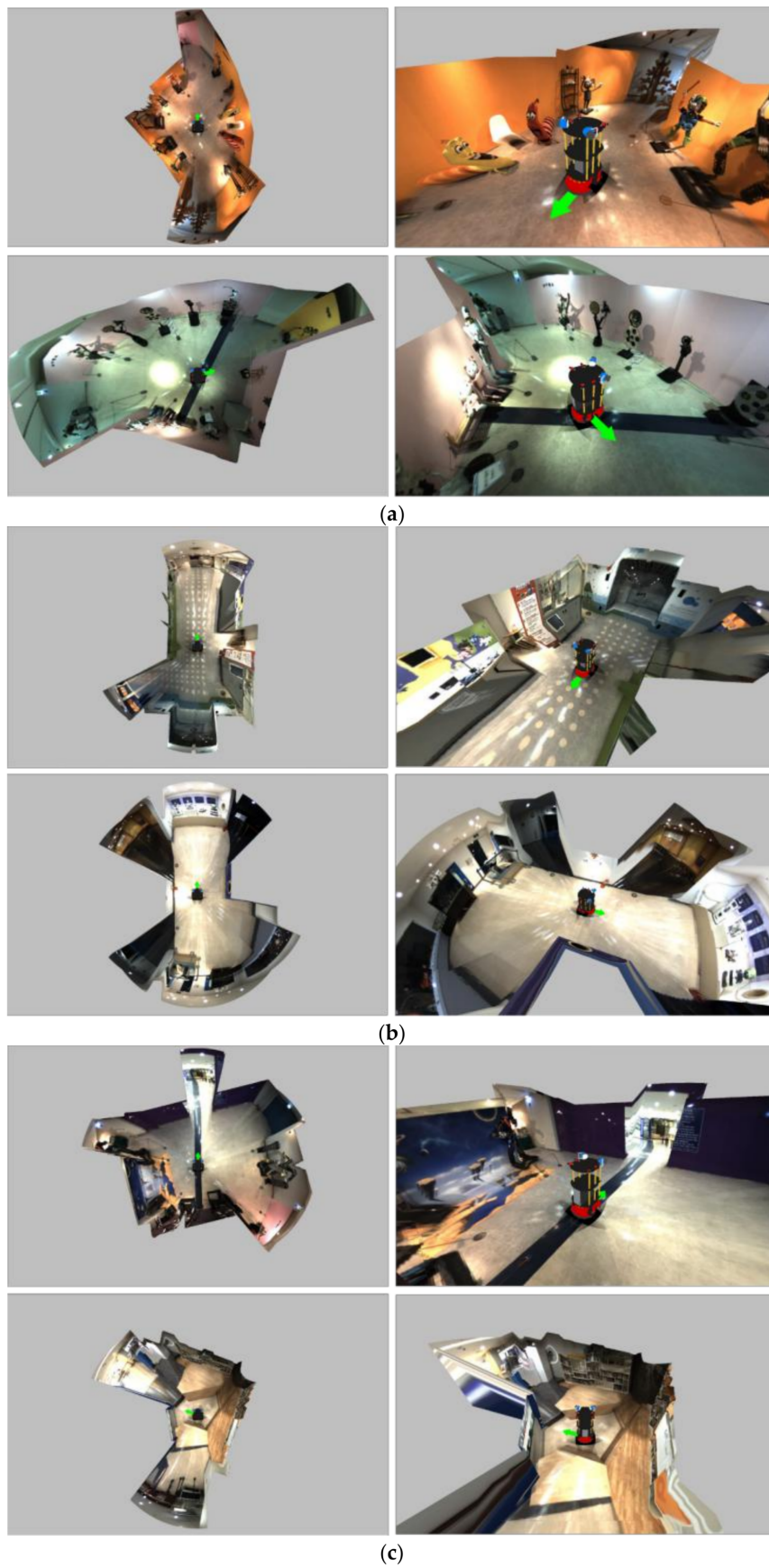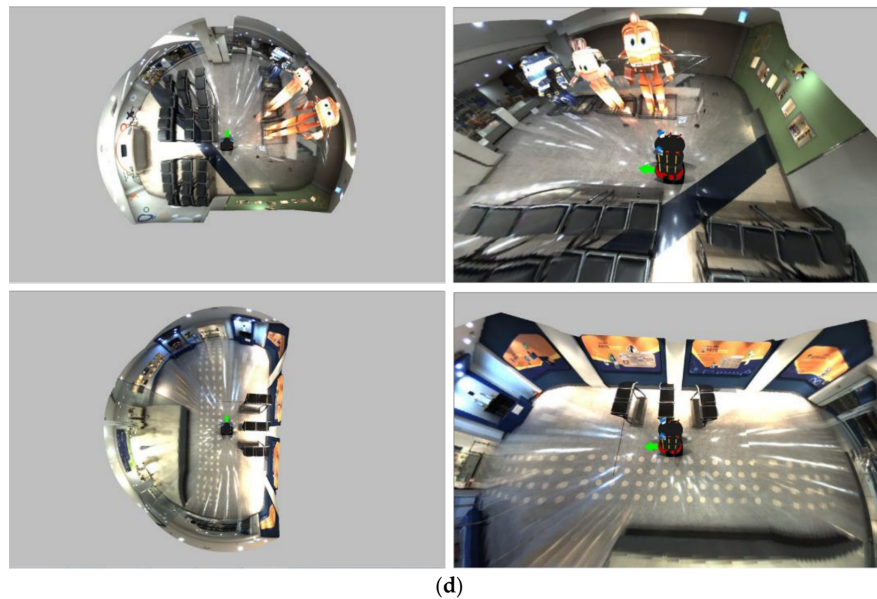
(**a**)



(**b**)



(**c**)

**Figure 17.** *Cont.*

**(d)**

**Figure 17.** Result comparisons of four methods in the same viewpoint. (**a**) Ground truth, (**b**) SLAM [17–19], (**c**) existing WAVM [10], (**d**) proposed system.

## 3.2. Objective Comparative Experiment

For the objective evaluation, we used markers (blue rectangular) in the testbed (Figure 16) and adopted the root mean square error (RMSE) to evaluate the similarity to the ground truth. A total of 120 marker images were transformed with homography to calculate the RMSE. They were also cropped and normalized to the same size. The RMSE was obtained as follows:

$$RMSE = \sqrt{\frac{1}{h \cdot w} \sum_{x=1}^{h} \sum_{y=1}^{w} \left( M_g(x, y) - M_t(x, y) \right)^2} \tag{11}$$

where $M_g$ and $M_t$ indicate the marker image of the ground truth and each test image (i.e., SLAM, existing WAVM, and proposed system). $h$ and $w$ denote the height and the width of $M_g$ and $M_t$, respectively.

**Table 1.** Comparison of the root mean square error (RMSE) for the ground truth and results with methods. SLAM: simultaneous localization and mapping.

| Methods | Distance | | |
|---|---|---|---|
| | 2 m | 2.5 m | 3 m |
| SLAM [17–19] | 7.5894 | 8.1826 | 8.3263 |
| Existing WAVM [10] | 5.4782 | 5.6378 | 5.2512 |
| Proposed system | 3.9556 | 3.6842 | 3.8215 |

Table 1 shows a comparison of the RMSE between the results of each method and the ground truth image when the distance between the robot and the wall was 2, 2.5, and 3 m, respectively. The RMSE values in Table 1 were the average of the RMSE with 120 marker images. As expected, the RMSE of SLAM was relatively large, which is due to feature matching errors. Furthermore, the RMSE of the proposed system was the lowest, indicating that it was the most similar to the ground truth. In summary, the experimental results of the subjective and objective evaluation showed that the proposed method had better visibility than the existing methods.

## 4. Conclusions

This study proposed a new visualization system that combines the WAVM and a spatial model to increase the visibility of robot tele-operation using four fisheye cameras and a 360° laser scanner. We compared the performance of the proposed system and the existing WAVM and SLAM that are widely used in robots by comparing experiments.

Our system provides tele-operators with a 360° view around the robot in real-time without accumulating data. This provides the tele-operator with various viewpoints and information about the changing environment, without any movement of the robot or sensor, in real-time. The tele-operator can determine the direction of the robot's motion and prevent collision with obstacles by intuitively recognizing the surrounding space of the robot.

## References

1. Lin, W.; Hu, J.; Xu, H.; Ye, C.; Ye, X.; Li, Z. Graph-based SLAM in indoor environment using corner feature from laser sensor. In Proceedings of the IEEE Youth Academic Annual Conference of Chinese Association of Automation, Hefei, China, 19–21 May 2017; pp. 1211–1216.
2. Cheng, Y.; Bai, J.; Xiu, C. Improved RGB-D vision SLAM algorithm for mobile robot. In Proceedings of the Chinese Control and Decision Conference, Chongqing, China, 28–30 May 2017; pp. 5419–5423.
3. Yuan, W.; Li, Z.; Su, C.-Y. RGB-D Sensor-based Visual SLAM for Localization and Navigation of Indoor Mobile Robot. In Proceedings of the IEEE International Conference on Advanced Robotics and Mechatronics, Macau, China, 18–20 August 2016; pp. 82–87.
4. Davison, A.J.; Reid, I.D.; Molton, N.D.; Stasse, O. MonoSLAM: Real-time single camera SLAM. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1052–1067. [CrossRef] [PubMed]
5. Wang, J.; Huang, S.; Zhao, L.; Ge, J.; He, S.; Zhang, C.; Wang, X. High Quality 3D Reconstruction of Indoor Environments using RGB-D Sensors. In Proceedings of the IEEE Conference on Industrial Electronics and Applications, Siem Reap, Cambodia, 18–20 June 2017.
6. Choi, S.; Zhou, Q.-Y.; Koltun, V. Robust Reconstruction of Indoor Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
7. Chen, J.; Bautembach, D.; Izadi, S. Scalable real-time volumetric surface reconstruction. *ACM Trans. Graph.* **2013**, *32*, 113. [CrossRef]
8. Lin, C.-C.; Wang, M.-S. Topview Transform Model for the Vehicle Parking Assistance System. In Proceedings of the 2010 International Computer Symposium, Tainan, Taiwan, 16–18 December 2010; pp. 306–311.
9. Jia, M.; Sun, Y.; Wang, J. Obstacle Detection in Stereo Bird's Eye View Images. In Proceedings of the 2014 Information Technology and Artificial Intelligence Conference, Chongqing, China, 20–21 December 2014; pp. 254–257.
10. Awashima, Y.; Komatsu, R.; Fujii, H.; Tamura, Y.; Yamashita, A.; Asama, H. Visualization of Obstacles on Bird's-eye View Using Depth Sensor for Remote Controlled Robot. In Proceedings of the 2017 International Workshop on Advanced Image Technology, Penang, Malaysia, 6–8 January 2017.
11. Szeliski, R. *Computer Vision: Algorithms and Applications*; Springer: Berlin/Heidelberg, Germany, 2010.
12. Shin, J.H.; Nam, D.H.; Kwon, G.J. *Non-Metric Fish-Eye Lens Distortion Correction Using Ellipsoid Model*; Human Computer Interaction Korea: Seoul, Korea, 2005; pp. 83–89.
13. Sung, K.; Lee, J.; An, J.; Chang, E. Development of Image Synthesis Algorithm with Multi-Camera. In Proceedings of the IEEE Vehicular Technology Conference, Yokohama, Japan, 6–9 May 2012; pp. 1–5.

14. Triggs, B.; Mclauchlan, P.F.; Hartley, R.I.; FitzGibbon, A.W. Bundle adjustment—A modern synthesis. In *International Workshop on Vision Algorithms*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 298–375.

15. Yebes, J.J.; Alcantarilla, P.F.; Bergasa, L.M.; Gonzalez, A.; Almazan, J. Surrounding View for Enhancing Safety on Vehicles. In Proceedings of the IEEE Intelligent Vehicles Symposium Workshops, Alcal de Henares, Madrid, Spain, 3–7 June 2012; pp. 92–95.

16. Salomon, D. *Transformations and Projections in Computer Graphics*; Springer: Berlin/Heidelberg, Germany, 2006.

17. Labbe, M.; Michaud, F. Online Global Loop Closure Detection for Large-Scale Multi-Session Graph-Based SLAM. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 2661–2666.

18. Labbe, M.; Michaud, F. Appearance-Based Loop Closure Detection for Online Large-Scale and Long-Term Operation. *IEEE Trans. Robot.* **2013**, *29*, 734–745. [CrossRef]

19. Labbe, M.; Michaud, F. Memory management for real-time appearance-based loop closure detection. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and System, San Francisco, CA, USA, 25–30 September 2011; pp. 1271–1276.