

Creation of 3D Environmental Map using Omnidirectional Camera Images

Yuya Kato
Graduate School of Fundamental
Science and Engineering
Waseda University
Tokyo, Japan
yuya-ulysses.393@ruri.waseda.jp

Junichi Hara
Global Information and
Telecommunication Institute
Waseda University
Tokyo, Japan
j.hara3@kurenai.waseda.jp

Hiroshi Watanabe
Graduate School of Fundamental
Science and Engineering
Waseda University
Tokyo, Japan
hiroshi.watanabe@waseda.jp

Abstract—In this paper, we propose a method to create a 3D environmental map from images captured by omnidirectional camera. Conventional methods using Simultaneous Localization and Mapping (SLAM) and fisheye images have a problem that the obtained feature points are geometrically distorted. To solve this problem, we transform the images captured by the omnidirectional camera using cube mapping and apply SLAM to the video created by concatenating the transformed images. The feature points located 360 degrees around the camera are merged into a single map. Experimental results show that our method outperforms other methods in terms of the number and accuracy of the obtained point cloud.

Keywords—*slam, omnidirectional camera, 3d environmental map*

I. INTRODUCTION

In recent years, technologies related to automated driving have been attracting attention. Two important technologies for automatic driving are self-positioning and environment mapping. Simultaneous Localization and Mapping (SLAM) is a technology that performs both of these tasks simultaneously. SLAM using a single camera image is called monocular SLAM, and is being actively researched as cameras are becoming lighter and cheaper. SLAM is also used for inspection, and in such applications, it is required to obtain accurate 3D data over a wide area in a single measurement.

Among SLAMs, LSD-SLAM [1] is a method that uses the brightness gradient of the entire image to estimate self-position and create an environment map, and it can obtain a dense point cloud. However, when the camera is rotated to capture the image, the path of the camera cannot be acquired during operation, and the 3D reconstruction of the output environmental map breaks down. ORB-SLAM [2], on the other hand, is a method for self-positioning and environment mapping using feature points in the image. It is robust against camera rotation. However, the number of obtained feature points is small, and the output point cloud is also sparse.

In this paper, we propose a method for creating 3D environmental maps from a video captured by an omnidirectional camera using monocular ORB-SLAM. First, we show the problem of ORB-SLAM when we directly input fisheye images. Next, we propose to transform the images taken by the omnidirectional camera using cube mapping. Then, transformed images are concatenated into a single video. Finally, ORB-SLAM is applied to the video, and obtained point clouds are merged. Experimental results show that our method outperforms other methods in terms of the number and accuracy of the obtained point clouds.

II. RELATED WORK

A. ORB-SLAM

ORB-SLAM is a method proposed by Mur-Artal et al. for self-positioning and environmental mapping using feature points in an image. ORB-SLAM is a kind of monocular SLAM that uses a single camera image, and two types of camera models are available: pinhole model and fisheye model. It is a method that can be executed on a CPU and is widely used because it can acquire point clouds with low latency and the camera path is accurate. The method using feature points has a small number of feature points to be selected and the output point cloud tend to be sparse. However, it is robust to changes in camera motion, especially rotation, and can accurately obtain the movement path.

B. Omnidirectional camera

An omnidirectional camera is a device that can capture 360 degrees of its surroundings at a time. There are types that combine multiple cameras to capture all directions, and types that combine two fisheye cameras in front and rear. Since these cameras can capture 360 degree images, they are beginning to attract attention in business and academic fields, such as when they are mounted on drones for measurement. There are two available image formats for omnidirectional cameras: fisheye and equirectangular as shown in Fig. 1.

C. Cube mapping

Cube mapping is a method of transforming an image as if it were taken with an omnidirectional camera inside a cubic box. By converting the image using cube mapping, it is output as a cubic development. Equirectangular images are used for the image transformation, and each image is divided into six square images. The angle of view of an omnidirectional camera image is 360 degrees, but the angle of view of each of the cube-mapped images is 90 degrees in both the vertical and horizontal directions. Image transformation using cube mapping is illustrated in Fig. 2.

Cube mapping minimizes the distortion of the six rectangular regions, but the image quality of each region is degraded depending on the sampling and curvature.



Fig. 1. Formats available for omnidirectional cameras. (left: fisheye, right: equirectangular).

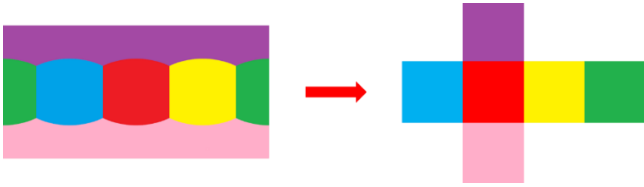


Fig. 2. Image transformation using mapping.

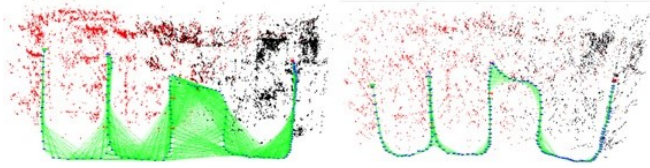


Fig. 3. Camera path and obtained point cloud. (left: pinhole model, right: fisheye model).

TABLE I. NUMBER OF OBTAINED FEATURE POINTS

Model	Angle of input image [degree]	Number of obtained points
Pinhole	90	13802
Fisheye	180	5422

III. PROBLEM SETTING

There are two types of camera models that can be used with ORB-SLAM: the pinhole model and the fisheye model. We conducted a preliminary experiment to compare the characteristics of the pinhole and fisheye models using images captured by an omnidirectional camera. For the pinhole model, we input a video that is cube-mapped to the omnidirectional camera image, and for the fisheye model, we input a video that is taken with only one lens of the omnidirectional camera. In both videos, the original video is the same and the center points of the videos are exactly the same.

The experimental results are shown in Table I. The path of the camera is shown in Fig. 3. From Table I, the pinhole camera model can obtain more feature points than the fisheye model. In this experiment, all the corners of the camera path are bent to 90 degrees. However, in Fig.3, we can see that most of the corners of the fisheye model are distorted instead of 90 degrees. This indicates that ORB-SLAM applied directly to the fisheye image does not provide sufficient quality of map generation.

Based on the results of our preliminary experiments, we will focus on the pinhole model of ORB-SLAM and devise a method to input images that match the model.

IV. PROPOSED METHOD

In this paper, we propose a method to acquire a wide range of point clouds using omnidirectional camera images.

When the same object is captured from different angles, the distribution of feature points may be different. In the experiment, we check whether the same feature points appeared by changing the camera angles of the two videos. As a result, it is confirmed that the same feature points are measured for the same object when the angle of the camera is 45 degrees or less. When the angle of the camera is 90 degrees, the same feature points are sometimes measured in the video, but it depends on the shooting conditions and the target objects.

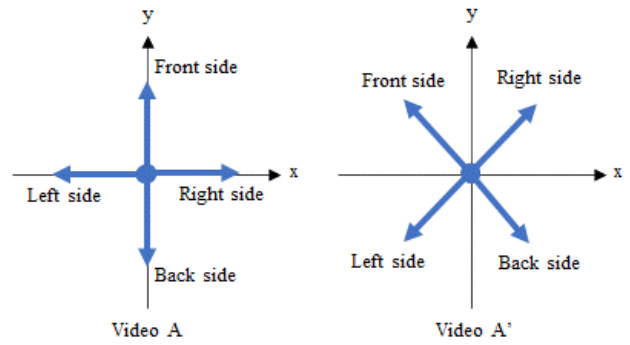


Fig. 4. Camera orientation when cube mapping is applied. (left: original, right: rotated by editing).

Therefore, a camera angle of 45 degrees or less is considered to be appropriate from the viewpoint of robustness to shooting conditions.

We convert the image taken by the omnidirectional camera to the equirectangular format, then convert the equirectangular image to an image with a shifted image center. Cube mapping is applied to the equirectangular image and the image with the shifted image center, and the necessary image is selected from the output images. The selected images are rearranged according to the direction that the center of each image is facing, and connected to form a single image. Finally, this image is input to ORB-SLAM to obtain the point cloud.

Let's assume that the video from the omnidirectional camera is "Video A" and the image with the camera orientation rotated 45 degrees by image editing is "Video A'". If the direction that the lens on the front of the omnidirectional camera faces in "Video A" is the positive y-axis direction, the camera directions of the front, right, back, and left images are as shown in Fig. 4 when "Video A" and "Video A'" are cube-mapped. The order of concatenation of images for video generation is shown in Fig.5. The whole process of the proposed method is shown in Fig. 6.

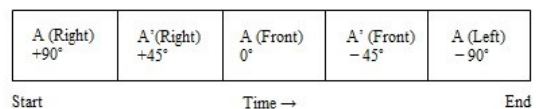


Fig. 5. Concatenation order of selected images for video generation.

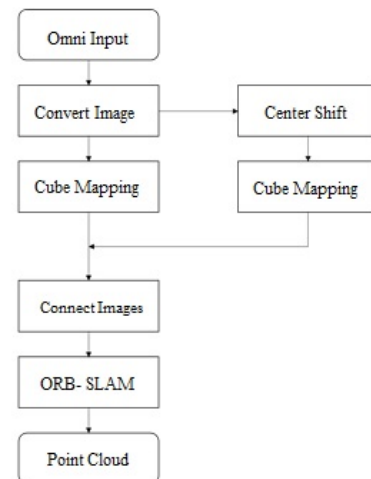


Fig. 6. Block diagram of the proposed method.

V. EXPERIMENT

A. Path with one turn (video1)

In the first evaluation experiment, the point cloud acquired by LiDAR is used as the ground truth, and the distance between the plane calculated from the ground truth and the point cloud acquired by ORB-SLAM is used for evaluation. A blackboard marker and LiDAR data is shown in Fig.7.

On the other hand, ORB-SLAM is a method that uses image features, so if a blackboard is used as a marker, no point cloud can be acquired inside the marker. Since many feature points appear in complex colors and shapes, we acquire a point cloud by placing a checkerboard on a white wall, where almost no feature points can be acquired. Fig. 8 shows the checkerboard used and the acquired point cloud.

The feature points acquired by LiDAR and ORB-SLAM exist in different places, so it is impossible to map and evaluate each point. Therefore, the point corresponding to the marker is cut out from the point acquired by LiDAR, and the evaluation is performed using the distance between the plane on which the point rides, and the point of the marker acquired by ORB-SLAM.

Fig. 9 shows the plane with $\mathbf{n} = (a, b, c)$ normal vector passing through the point $P(x_0, y_0, z_0)$. The distance D at the point $A(x_a, y_a, z_a)$ is given by

$$D = \frac{|ax_a + by_a + cz_a - ax_0 - by_0 - cz_0|}{\sqrt{a^2 + b^2 + c^2}} \quad (1)$$

The placed makers A, B, C, and D in our laboratory and measured LiDAR data are shown in Fig. 10.

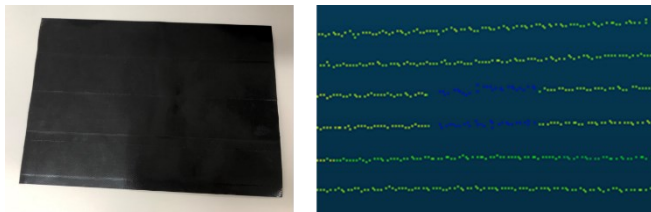


Fig. 7. Blackboard marker and LiDAR data.



Fig. 8. Checkerboard and ORB-SLAM data.

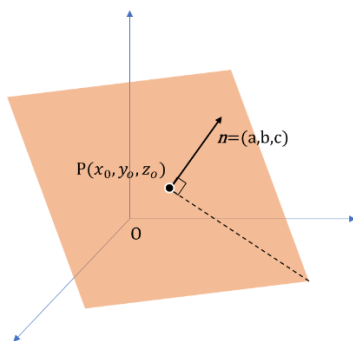


Fig. 9. Plane through point P .

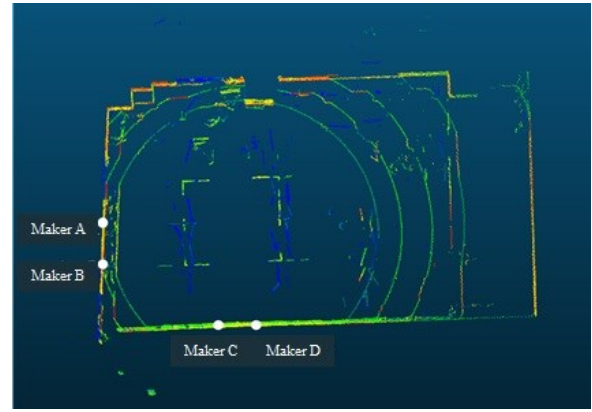


Fig. 10. Measured LiDAR data and makers in the laboratory.

In the evaluation experiment, Markers A, B, C, and D are placed at four locations in the moving path of the camera. Among the omnidirectional camera images, the image acquired from the front camera is input to the ORB-SLAM of the fisheye model (Method 1), and the image acquired by cube mapping in the front direction is input to the ORB-SLAM of the pinhole model (Method 2) are used for comparison.

The obtained distances from the plane to the makers are shown in Table II. It shows that the proposed method has smaller error values for all markers. In addition, the proposed method and Method 2 are able to measure, while Method 1 is not able to obtain the point cloud, indicating that the cube mapping method can be used in various environments. The obtained 3D environmental map by the proposed and method 2 are shown in Fig. 11 and Fig. 12. We can observe that more feature points are obtained in Fig. 11 than Fig. 12.

B. Path with two turns (video2)

In the video used in the experiment, the path was straight, then left, then left again, and then straight. The environmental maps created by the proposed method, Method 1 and Method 2 are shown in Fig. 13, Fig. 14 and Fig. 15, respectively.

TABLE II. DISTANCE BETWEEN MAKER AND PLANE FOR VIDEO1

Approach	Average distance between maker and corresponding plane [cm]			
	Maker A	Maker B ^a	Maker C	Maker D
Proposed	2.14	2.44	4.03	4.07
Method 1	-	-	-	-
Method 2	6.77	3.45	5.10	6.40

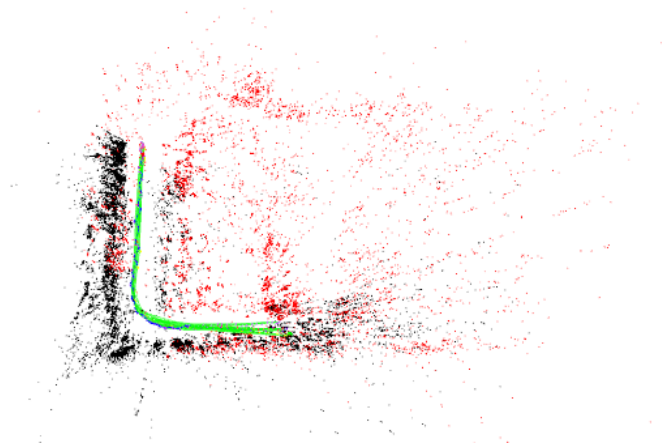


Fig. 11. 3D environmental map created by the proposed method. (video1).

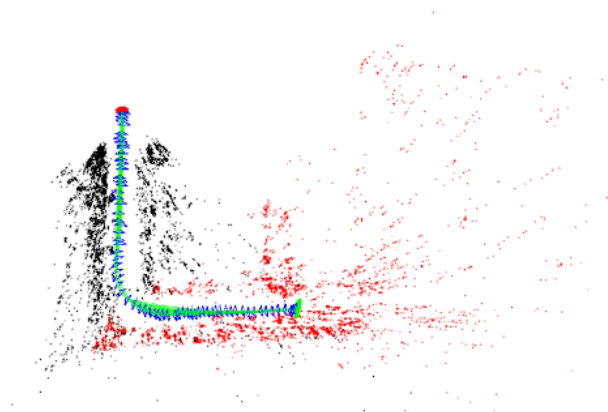


Fig. 12. 3D environmental map created by the method 2. (video1).

The moving path of the camera is represented by the blue mark, and the acquired point cloud is represented by the red and black dots. From Fig. 13 to Fig. 15, we can confirm that the number of point clouds acquired by the proposed method is large. Since the proposed method traverses the same path five times while changing the camera direction, it has a wide measurement range and can obtain a large number of points.

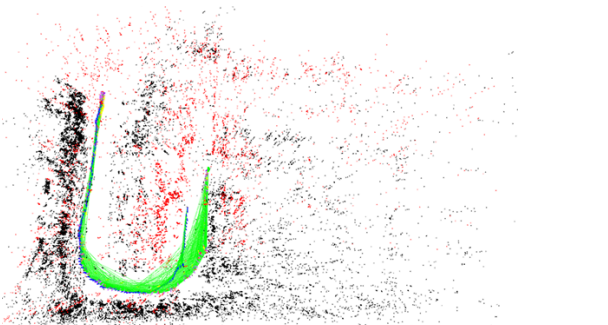


Fig. 13. 3D environmental map created by the proposed method. (video2).

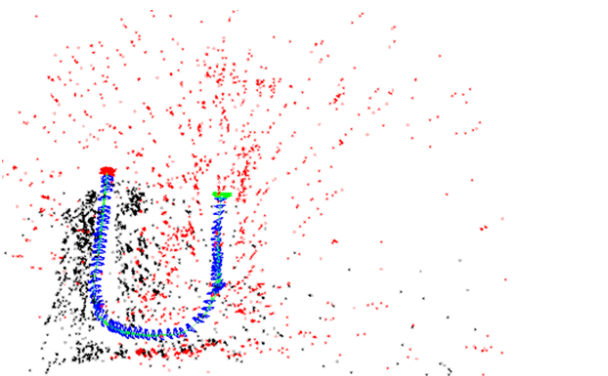


Fig. 14. 3D environmental map created by the method 1. (video2).

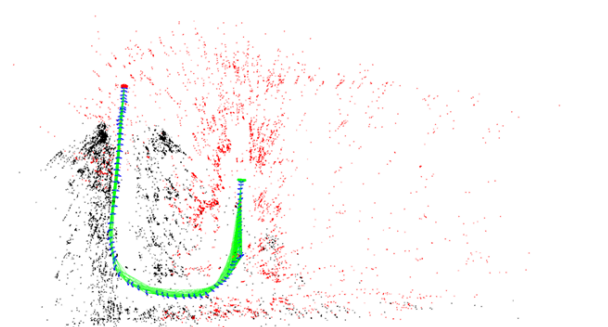


Fig. 15. 3D environmental map created by the method 2. (video2).

TABLE III. DISTANCE BETWEEN MAKER AND PLANE FOR VIDEO2

Approach	Average distance between maker and corresponding plane [cm]			
	Maker A	Maker B ^a	Maker C	Maker D
Proposed	1.93	8.32	0.96	3.58
Method 1	-	12.03	8.57	14.24
Method 2	-	13.43	-	4.50

One of the problems of the proposed method is the error of the camera path. In Fig. 14 and Fig. 15, the camera path is represented by a single line, while in Fig. 13, the path of the camera does not coincide with that of the proposed method.

The point clouds of the four marker parts located on the shooting path are cut out and the average distances to the corresponding planes are taken; the point clouds of the marker parts are selected from the point clouds acquired using ORB-SLAM, and the distances between the point clouds and the planes of the ground truth acquired by LiDAR are obtained. The distances between the point clouds and the corresponding planes acquired by the proposed method, Method 1 and Method 2 are shown in Table III. The values left blank in Table III are those for which no point cloud indicating the marker is obtained, or those for which it is not clear whether the point cloud represents a feature point on the marker due to the small number of obtained point clouds, and thus could not be evaluated.

Table III shows that the error of the proposed method is the smallest for all three markers except for marker A. As for marker A, only the proposed method can acquire the point cloud, which indicates that the proposed method is robust to changes in shooting conditions. Other methods could not get points for maker A and one for maker C.

VI. CONCLUSION

In this paper, we proposed a method to acquire point clouds by concatenating the images obtained by applying cube mapping to the omnidirectional camera images and then inputting them to ORB-SLAM. By comparing the existing methods with the proposed method, we confirmed that the proposed method increases the number of point clouds that can be acquired. In addition, evaluation experiments using markers showed that the proposed method improved the accuracy of the point cloud compared to the existing methods. Note that DSO [3] cannot be applied to fisheye images and OpenVSLAM [4] may not be able to be measured because the image contains observers.

REFERENCES

- [1] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM," European Conference on Computer Vision (ECCV2014), pp. 834-849, Sep. 2014.
- [2] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," IEEE Trans. On Robotics, vol. 31, no. 5, pp. 1147-1163, Aug. 2015.
- [3] D. Caruso, J. Engel, and D. Cremers, "Large-Scale Direct SLAM for Omnidirectional Cameras," International Conference on Intelligent Robot Systems (IROS), pp. 141-148, Oct. 2015.
- [4] S. Sumikura, M. Shibuya, and K. Sakurada, "OpenVSLAM: a Versatile Visual SLAM Framework," ACM International Conference on Multimedia (MM'19), pp. 2292-2295, Oct. 2019.