コンピュータビジョンと
イ メ ー ジ メ デ ィ ア 131－15
（２００２．１．１８）

Reconstructing Environmental Structure by Virtual Omnidirectional Image Generation

Moldovan Daniel, Takahiro Miyashita and Hiroshi Ishiguro
Faculty of Systems Engineering, Wakayama University,
Sakaedani, Wakayama 640, Japan
{s010142, miyasita, ishiguro}@sys.wakayama-u.ac.jp

Abstract

This paper describes a new method for reconstructing the 2D structure of an environment from image sequences taken by an omnidirectional camera. The process starts by capturing the original omni-images with the omnidirectional camera on a mobile platform that is moving on a straight path. By exploiting the characteristics of the virtual omni-images generated at arbitrary viewing points in the environment, we are able to synthesize a 3-D visual representation of the environment. The 2D structure that is parallel to the ground plane will emerge through analyzing the 3-D visual representation. That is, this method directly reconstructs the 2D structure from the omnidirectional visual sequence without matching process needed for multiple camera stereo. In the experimentation, we have applied this method to both of indoor and outdoor environments and verified the performance.

1. Introduction

In the previous works, various researchers have explored the use of OD (Omni-Directional) camera systems, in the context of robotic applications, for reconstructing environments from video imagery. By combining the measurements obtained from the video imagery with odometry measurements from the robot, Yagi, Kawato, Tsuji and Ishiguro [4, 5, 14] constructed maps of the robot environment.

In order to retrieve the 3D information from the environment Kawasaki et al [7] have been proposed a spatio-temporal analysis of omni images. They proposed a hybrid method using the EPI and the model-based analysis, performing a matching between video data and models. 3D information was retrieved from video data by using the matching results.

In contrast with this approach, this paper describes a new method for detecting the geometrical structure of the environments by exploiting the characteristics of the virtual OD images generated in an immersive manner into the environment. Our method does not require the matching process. It, rather, needs to generate many virtual OD images.

More similar approaches to ours are methods for approximately realizing *plenoptic functions* [1, 9], such as *lumigraph* [3] and *light field rendering* [10]. Recently, Camillo [13] presented an approach for capturing the appearance of immersive scenes by combining techniques from structure from motion with ideas from image-based rendering. The limitation of this approach in the context of robot navigation is that it actually doesn't offer information about the structure of objects surrounding the robot. By being able to create any view from any position to any direction on the ground, Takahashi's work [12] is the most closely related with ours. However, their work is to reconstruct normal views with a limited visual field. In our work, we have improved and pushed forward this idea to generate virtual OD views and to find directly the environment structure from many virtual OD views.

This paper is organized as follows. In Section 2 we review two of the main stereo methods used for detecting the structure of the environment and present the originality brought by our method. In Section 3 we describe the method for reconstructing structure from virtual OD images. Section 4 presents the results of the experiments using both indoor and outdoor scenes. Section 5 includes conclusions regarding this method, along with discussions of future work.

2. From binocular to multiple camera stereo

From the beginning of computer vision research, there are many researchers tackled to solve problems of the binocular stereo. The most important and difficult problem is feature matching. Feature matching method proposed so far, such as template matching is not stable especially for long base lines between cameras.

Suppose we have two cameras arranged so that their optical axes are parallel and separated by a distance B (See Fig.1). The disparity d is related to the distance z by:

$$d = \frac{f \cdot B}{z}$$

where B and f are baseline and focal length respectively.

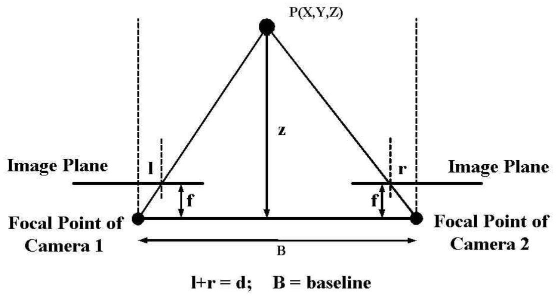


Figure 1: **Binocular Stereo**

This equation indicates that for a constant distance z, the disparity d is proportional to the baseline B. In order to have a better-estimated distance z, we need a longer disparity, which can be translated into a longer baseline. Therefore, we need to set the two cameras further apart from each other.

However, because a larger disparity range must be searched, there is a greater possibility of a false match. Figure 2 illustrates this fact. For each pixel in the reference image and for each possible disparity, the error is calculated between the *reference* and a *match* image.

That is, in binocular stereo, in order to select the baseline length there exist a trade-off between precision of measurement and accuracy of matching.

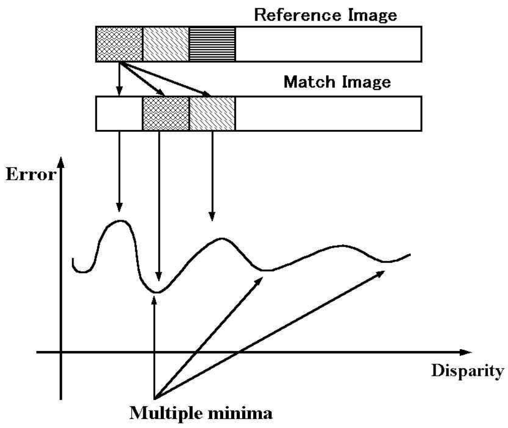


Figure 2: **Matching two images in binocular stereo**

Kanade - Okutomi's *multi-baseline stereo* [11] gave a better solution to the matching problem. They have used multiple stereo pairs with different baselines generated by a lateral displacement of cameras (see Fig. 3), and performed a simple matching by computing the sum of squared-difference (SSD) values.

By representing the SSD functions with respect to the inverse depth (1/z) and then by simply adding, to produce the sum of SSDs, the false matches are canceling each other out. The resulting function exhibited a unique minimum at the correct matching

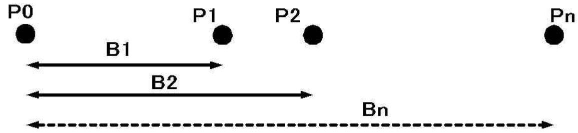


Figure 3: **Camera positions for multi-baseline stereo**

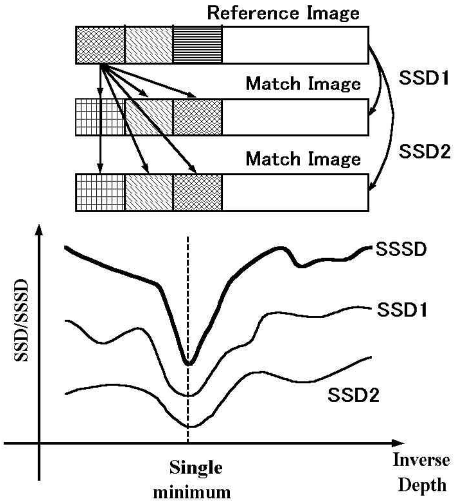


Figure 4: **SSD and SSSD functions**

Our method can be considered as an extension of the multiple camera stereo. To increase the number of cameras, our method uses a view sequence from OD camera that is moving along a straight path. From the recorded view sequence, we generate virtual omni-images on both sides of the robot path and exploit changes that appear in each virtual image that encounters with an object (see Fig. 5). By generating virtual images in a dense manner we can detect also the geometrical shape of the object.

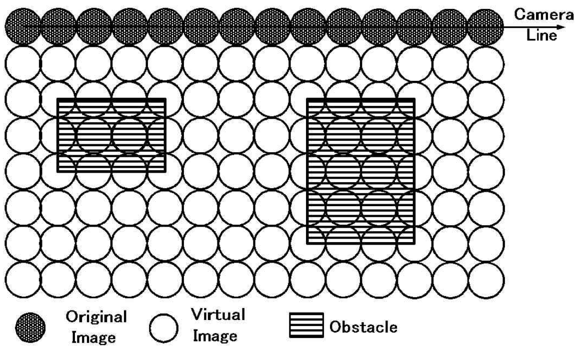


Figure 5: **Virtual images generated into the environment**

That is, in our method we don't have to solve the correspondence problem. In other words, we propose an alternative stereo method that uses rich visual information in order to avoid correspondence problem.

3. Structure from Virtual OD images

Synthesis of Virtual OD images

The omnidirectional camera that we used for image acquisition is composed of a CCD camera upward looking at a hyperboloidal mirror. For the outdoor experiments, the camera is mounted on a support on the roof of a vehicle and for the indoor experiments the camera is mounted on a mobile platform. After recording a video sequence along a straight path, we apply a quantization process and obtain a number of original OD images (Fig. 6).

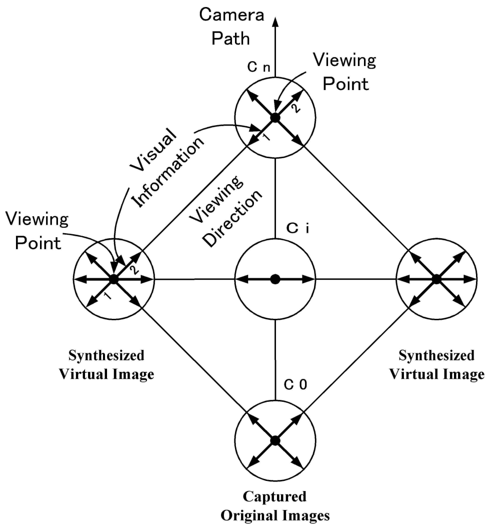


Figure 6: **Synthesis of virtual OD images**

By using the information from these original images, we generate virtual OD images on both sides of the robot path. The purpose of generating virtual images is to get a clue of the obstacle's shape by exploiting the transformation that could appear in the newly generated images when they encounter an obstacle as described in the next section.

For generating the virtual images, we use the principle of the viewing direction's continuity. In other words, if the viewing direction of two different viewing points coincides, the visual information for the corresponding rays is identical. By finding and collecting the rays from the stored original images that correspond to the rays from the virtual images, the processing system can synthesize the virtual view as shown in Fig. 6. As the location of the virtual viewing point is moving away from the camera path, the reconstructable area in the virtual image is more restricted.

Structure from Virtual OD images

Fig. 7 shows an OD image taken in the image sequence. The extracted circle represents a horizontal plane that passes the focal point of the hyperboloidal mirror. For reconstructing 2D structure on the horizontal plane, we take the image data along the circle. That is, in the following process, the OD image is represented as a line image.

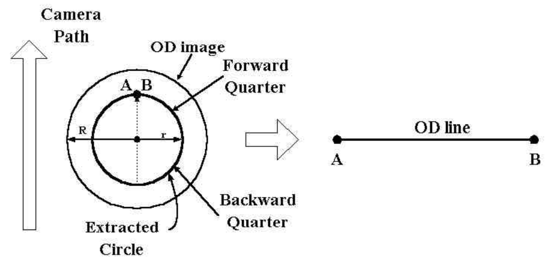


Figure 7: **OD line**

From the line image sequence taken along the straight path, we generate virtual OD lines that will be gathered at the end in an virtual OD line image as shown in Fig. 8. Each virtual circle is divided in four quarters: two on each side of the path, corresponding to a forward and backward view.

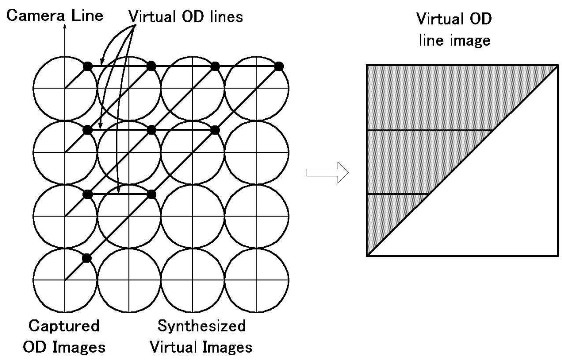


Figure 8: **Virtual OD line image**

First we apply a process of quantization and divide each of the extracted circles into a sequence of intermediate pixels. Corresponding to each quarter, we will render a number of nineteen virtual OD line images of the environment by importing one pixel at a time from each synthesized virtual image. The pixels are arranged in the same order as the location of virtual images in the surrounding environment.

By arranging of all virtual OD line images one on top of the other we can build the spatio-temporal volume (see Fig. 9). Virtual OD line images corresponding to each quarter, are then overlapped in order to we get a final image for both forward and backward views.

In the spatio-temporal volume, I, X, Y, axes represent the virtual section, distance from the camera path and the camera movement, respectively. The next step is to acquire the 2D projection of this spatio-temporal volume on to the X-Y plane.

What happens if we generate a virtual view on an object? A virtual view in a free space shows an image that should be taken at the point. However, the virtual view on an object shows a monotone image filled with the object color. That is, the virtual image loses the texture. We have focused on this characteristic of the virtual views for detecting the 2D structure of the environment.

The projection procedure is to simply compute the variance of RGB pixel values of each OD image. If the variance is smaller than a certain threshold, than the output is valid, meaning that there is an object. Otherwise, the result is invalid (no object). By the simple projection, we can get an XY image that represents environmental structure along the camera path. Note that this process for acquiring the 2-D environmental map does not require a feature matching process.

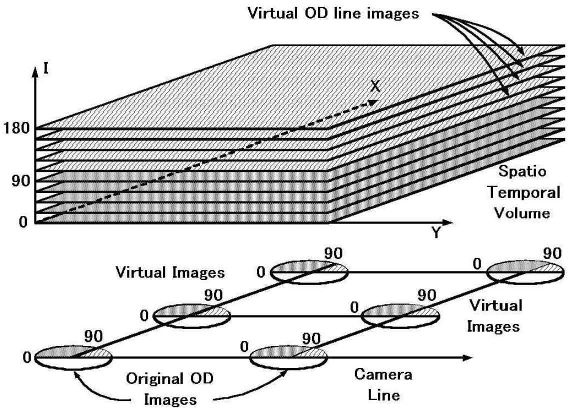


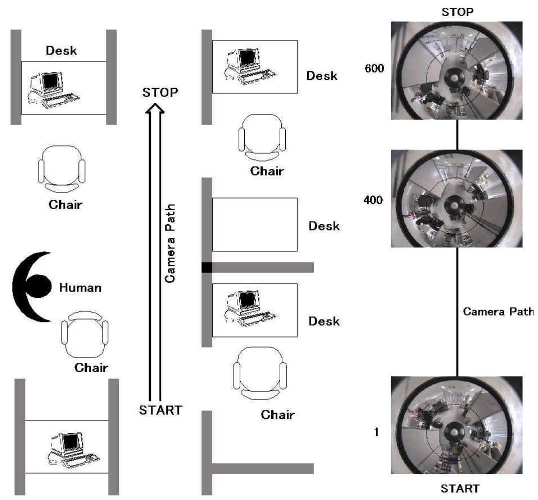
Figure 9: **Building the spatio-temporal volume**

4. Experimental results

The indoor experiment has been done using a static environment from our laboratory. Fig. 10 (b) shows the sequence of the original images that were recorded with a frequency of one image / 0.30 cm. For the extracted image data we selected a 70 pixels length of the radius from the OD image center. The thick, gray lines from Fig. 10 (a) represents the walls of the cubicles. By rendering the left side of the camera path we obtain a sequence of 180 virtual OD line images (see Fig. 11).

The structure seen from backward (Fig. 12.a) emerges from overlapping the first 90 slices and by overlapping the next 90 slices we get the structure seen from forward (Fig. 12.b). The final structure on the left side of the path is shown in Fig. 12.c. The meanings of the objects encircled and numbered is: (1) the corner of the wall; (2) the chair; (3) the standing person and the chair; (4) the corner of the wall.

By following the same procedure we have determined the structure on the right side of the path (Fig.13). The meanings of the encircled areas are as follows: (1) represents half of the upper wall; (2) represents the lower part; and (3) is the edge of the desk. The two distinct walls that were detected correspond in fact to the reality (in the original images they are separated by an edge of a different color).



(a) (b)

Figure 10: **Indoor environment: (a) simplified representation of the area; (b) omni images along the path**

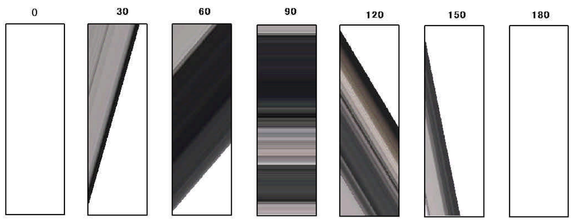
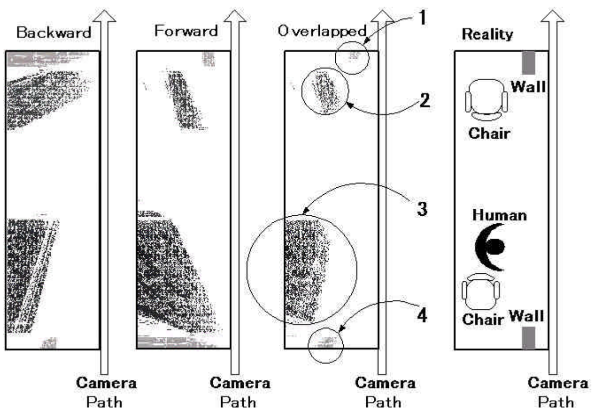


Figure 11: **Rendered virtual sections on the left side**



(a) (b) (c) (d)

Figure 12: **Left side structure: (a) seen from backward; (b) seen from forward; (c) overlapped results; (d) reality.**

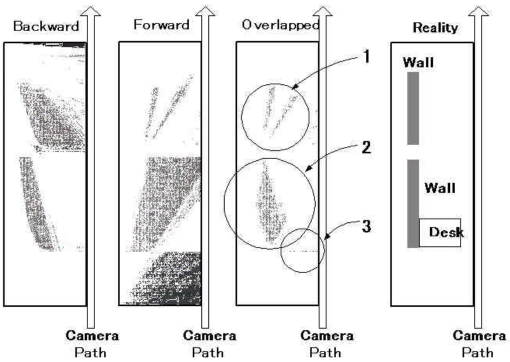


Figure 13: **Right side structure: (a) seen from backward; (b) seen from forward; (c) overlapped results; (d) reality.**

Next are the results for an outdoor scene, located in our university campus. The camera ran along a straight path for 50 m while recording a sequence of OD images (Fig. 14.b) with a frequency of one image per 25cm. The two buildings located on each side of the path represent the structure that we want to detect (Fig. 14.a). The length of the radius for extracted image data was selected 100 pixels from the OD image center (the circle was crossing both buildings located on each side of camera).

The backward and forward virtual sections corresponding to the left side of the path are shown in Fig 15.a. The right side structure is shown in Fig. 15.b. The two encircled objects (1 and 2) from the left side correspond to the front part of building 1 that includes a balcony and two pillars. The presence of the building 2 is detected on the right side (3), but because of the limitation of this method in dealing with concave shapes, the black shaded area is also assimilated in the building's shape.

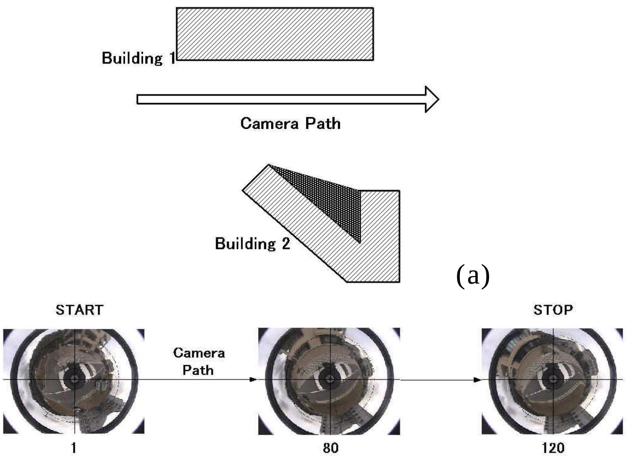
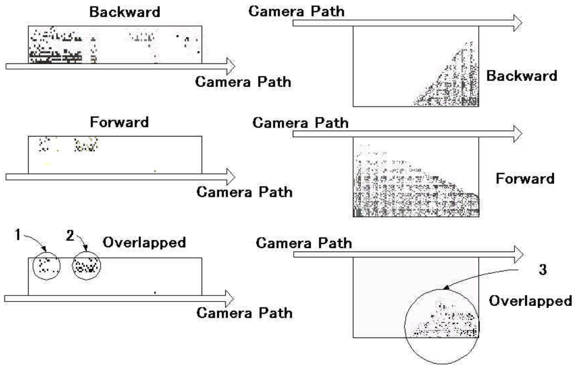


Figure 14: **Outdoors environment: (a) structure; (b) OD images along the path**



(a) (b)

Figure 15: **Structures on both side of the road: (a) left side; (b) right side.**

5. Conclusion

This paper described a new method for reconstructing the 2D structure of the surrounding environment from image sequences taken by an OD camera. We generate virtual omni-mages on both sides of the path and exploit the changes that appear in each virtual image that encounter with an object. That is, we are proposing an alternative stereo method that uses rich visual information in order to solve the correspondence problem.

The strength of our method is that it requires a single omni-camera and the processing is done in real time being well suited for robot navigation. A weak point is represented by the low accuracy in detecting the objects shape. Future work will focus on overcoming this limitation and on extending our method to 3D structure.

References

[1] E.H. Adelson and E.H. Bergen, "The plenoptic function and the elements of early vision", *MIT Press*, 1991.

[2] N.Ayache and O. Faugeras. "Maintaining representations of the environment of a mobile robot", *IEEE Transactions on Robotics and Automation*, pages 804-819, December 1989.

[3] S. J. Gortler, R. Grzeszczuk, R. Szeliski and M. F. Cohen, "The lumigraph", *Proc. SIGGRAPH*, pp. 43-54, 1996.

[4] Hiroshi Ishiguro, Takeshi Maeda, Takahiro Miyashita, and Saburo Tsuji, "A strategy for acquiring an environmental model with panoramic sensing by a mobile robot", *IEEE Int. Conf on Robotics and Automation*, pages 724-729, 1994.

[5] Hiroshi Ishiguro, Kenji Ueda and Saburo Tsuji, "Omnidirectional visual information for navigating a mobile robot", *IEEE Int. Conf. On Robotics and Automation*, pages 799-804, 1993.

[6] Hiroshi Ishiguro, Masashi Yamamoto, and Saburo Tsuji, "Omni-directional stereo", *IEEE Trans. Pattern Anal. Machine Intell.*, 14(2):257-262, February 1992.

[7] Hiroshi Kawasaki, Katsushi Ikeuchi, and Masao Sakauchi, "Spatio-Temporal Analysis of Omni Image",

Computer Vision and Pattern Recognition, pages 577-584, 2000
[8] D.Kriegman, E.Triendl, and T.Binford. "Stereo vision and navigation in buildings for mobile robots", *IEEE Transactions on Robotics and Automation*, pages 792-802, December 1989.
[9] M. Landy and J.A. Movshon eds, "Computation models of visual processing", *MIT Press*, 1991.
[10] M. Levoy and P. Hanrahan, "Light field rendering", *Proc. SIGGRAPH*, pp. 31-42, 1996.
[11] Okutomi, M., and Kanade, T.A, "Multiple Baseline Stereo", *Proceedings of the 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 63-69, 1991.
[12] Takuji Takahashi, Hiroshi Kawasaki, Katsushi Ikeuchi, and Masao Sakauchi, "Arbitrary View Position and Direction Rendering for Large-Scale Scenes", *Computer Vision and Pattern Recognition*, pages 296-303, June 2000.
[13] Camillo J. Taylor, "VideoPlus", *IEEE Workshop on Omnidirectional* Vision, pages 3-10, June 2000
[14] Yasushi Yagi, Shinjiro Kawato, and Saburo Tsuji, "Real-time omnidirectional image sensor for vision guided navigation", *IEEE Journal of Robotics and Automation*, 10(1): 11-21, February 1994.