Generation Method for Immersive Bullet-Time Video Using an Omnidirectional Camera in VR Platform

Oto Takeuchi
University of Tsukuba
Japan
takeuchi.oto@image.iit.tsukuba.ac.jp

Hidehiko Shishido
University of Tsukuba
Japan
shishido@ccs.tsukuba.ac.jp

Yoshinari Kameda
University of Tsukuba
Japan
kameda@iit.tsukuba.ac.jp

Hansung Kim
University of Surrey
United Kingdom
h.kim@surrey.ac.uk

Itaru Kitahara
University of Tsukuba
Japan
kitahara@ccs.tsukuba.ac.jp

ABSTRACT

This paper proposes a generation method of immersive bullet-time video that continuously switches the images captured by multi-viewpoint omnidirectional cameras arranged around the subject. In ordinary bullet-time processing, it is possible to observe a point of interest (POI) at the same screen position by applying projective transformation to captured multi-viewpoint images. However, the observable area is limited by the field of view of the capturing cameras. Thus, a blank region is added to the displayed image, depending on the spatial relationship between the POI and the capturing camera. This seriously harms image quality (i.e., immersiveness). We solve this problem by applying omnidirectional cameras to bullet-time video production. Furthermore, by using the virtual reality platform for calibration of multi-viewpoint omnidirectional cameras and display of bullet-time video, fast and simple processing can be realised.

KEYWORDS: Bullet-Time video; Multi-Viewpoint shooting; Omnidirectional camera; Camera calibration; VR platform

ACM Reference format:

O. Takeuchi, H. Shishido, Y. Kameda, H. Kim, and I. Kitahara. 2018. Generation Method for Immersive Bullet-Time Video Using an Omnidirectional Camera in VR Platform. *2018 Workshop on Audio-Visual Scene Understanding for Immersive Multimedia Proceedings (AVSU'18), Seoul, Republic of Korea*, *October 2018*, 8 pages.
https://doi.org/10.1145/3264869.3264872

AVSU'18, October 26, 2018, Seoul, Republic of Korea.

ACM ISBN 978-1-4503-5977-1/18/10...$15.00.
DOI: https://doi.org/10.1145/3264869.3264872

1 INTRODUCTION

Multi-Viewpoint video technology has been used for many purposes, such as sports broadcasting and movies. One of the most active multi-viewpoint technologies is bullet-time video, which portrays the movement of a viewpoint by continuously switching the captured immultiages according to the camera layout. Compared to free-viewpoint video [1, 2], bullet-time can generate higher quality video, because the captured images are presented almost as-is (i.e., devoid of the influence of errors caused by 3D reconstruction).

With bullet-time video, if the optical axis (observation direction) of the multi-viewpoint cameras intersects at one point (e.g., gazing point) in the space, a smooth viewpoint movement can be realised while the observer switches the virtual camera. If we arrange the multi-viewpoint cameras by spending enormous time and effort, it is possible to set the gazing point accurately. However, if there is a request to change or reset the gazing point, it is necessary to arrange the cameras once again. Akechi [3] enhanced bullet-time processing by applying a suitable projective transformation to each image. The transformation was calculated from the 3D positions and orientations of the multi-viewpoint cameras and those of the gazing point, so that the gazing point would be observed from a same position on the screen. As a result, it was possible to reset the gazing point without changing the camera arrangement. However, multi-viewpoint shooting is performed with perspective projection cameras having limited fields of view (FoV). Thus, if one sets the gazing point at the outer region of the image, as shown in Fig. 1, the area not captured by the camera is observed in the displayed image as a black-coloured region. This is a serious problem for quality of immersive sensation.

In this research, we propose a "immersive bullet-time" generation method that does not create a blank region. Our key-approach is shooting with an omnidirectional camera that captures the target scene without FoV limitations. Because there is no blank region, it becomes possible to generate bullet-time videos that can be set in an arbitrary gazing points in the shooting space.

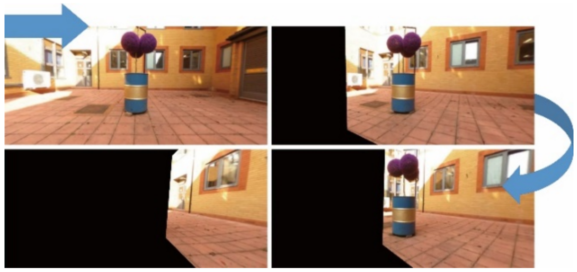

Figure 1: Bullet-Time video where a blank region is observed as the viewpoint moves.

However, omnidirectional images can create other problems. Captured omnidirectional images are generally recorded as equirectangular projection images, as shown in Fig. 2(a). Whereas one can record the target omnidirectional space as one image plane, the projective geometry of the omnidirectional image is different from the perspective projection used in general computer vision [4]. Most feature descriptors, such as scale-invariant feature transform (SIFT) [5] have been developed for perspective projection camera models. Thus, the laterally stretched distortion approaching the upper and lower ends of the image, as seen in omnidirectional images, does not correspond to most feature descriptors. Therefore, it is common to convert omnidirectional images to perspective projection images, then extract image feature points to estimate camera parameters [6]. Additionally, to display natural appearances using human-based visual projection characteristics, it is desirable to use the perspective projection-transformed image when presenting the video.

Owing to the popularisation of omnidirectional cameras, attention has been paid to virtual reality (VR) development platforms. By mapping the omnidirectional image to the CG object in the VR environment and rendering it using a virtual pinhole camera, it is possible to simply and quickly obtain the perspective projection geometric image. Additionally, by displaying the image shot by the virtual camera on a head-mounted display (HMD) equipped with the posture sensor, it is possible to provide the viewer a feeling of immersion in the shooting space. In this research, using this VR environment technology, conversion processing from omnidirectional images to perspective projection images and presentation of video are performed. As the result, it is possible to accurately estimate camera parameters including the position and orientation by applying well developed camera calibration method such as Structure from Motion (SfM) to the captured omnidirectional videos. When we integrate our bullet-time video with immersive audio to improve the presence, the position and orientation of capturing cameras, which are expressed in the 3D world coordinate system, are very important.

However, there are issues to consider for perspective projection transformation processing. Spatial information of a wide FoV can be acquired, as shown in Fig. 2(b), when the view angle of the perspective projection image is increased. However, at a pixel far from the image centre, distortion caused by perspective increases. Because SIFT is not robust against large perspective projective distortion, it is necessary to determine the appropriate FoV of the perspective projection image. Additionally, matching accuracy by image feature is influenced by the overlapping regions between images (i.e., larger overlapping regions give better matching accuracy). However, larger overlapping regions can cause issues that increase calculation costs when the number of images increase. In this paper, we consider these issues also.

(a) **(b)**

Figure 2: (a) Omnidirectional image recorded by equirectangular projection; (b) A wide-angle perspective projection image (horizontal view angle = 150°).

The main contributions of this paper are the following:

- Bullet-Time generation method that does not create a blank region by using an omnidirectional camera,
- Using VR platform, perspective projection transformation for camera calibration and display of omnidirectional image.

2 RELATED WORKS

2.1 Bullet-Time Video

When generating bullet-time video, it is necessary for the optical axis of the multi-viewpoint cameras to intersect at one point (i.e., gazing point) in space. At this time, a smooth viewpoint movement is realised when the camera is switched. The primitive method is to precisely arrange multi-viewpoint cameras using a micrometer, etc. However, shooting with this application in a large space is not feasible, because it requires too much labour and time. Additionally, when the gazing point is moved to another location in the target space, it is difficult to generate smooth view-switching video with this technique.

Kanade et al. [7] developed a bullet-time video generation method (i.e., Eye Vision) for events performed in large-scale spaces, such as with American football and baseball. The 3D position of the gazing point was estimated from the 3D position and orientation of a master camera controlled by a camera-man. Other cameras were automatically controlled so that the subject was observed at the same place on the screen. Thus, bullet-time video generation was realised, even when the subject moved around. However, to estimate the 3D position from the monocular image, the system assumes that the subject is always standing on the ground. Thus, it becomes difficult to set the gazing point to an arbitrary place in 3D space. Ikeya et al. [8] proposed a bullet-time video generation method for setting the gazing point at a subject moving in 3D space by integrating 3D object tracking technology and robotic camera control technology. However, by using robotic cameras, the gazing point was set by the camera-man at the time of shooting, making it is difficult for a viewer to set it freely.

Akechi et al. [3] proposed a bullet-time video generation method in which a viewer set the gazing point at an arbitrary 3D location. In this method, multi-viewpoint images were captured with fixed cameras, and projective transformation was performed, such that the gazing point was observed at the same position on the screen. Because it did not use robotic cameras, it was possible to shoot with general equipment, and the gazing point could be set at the time of viewing. However, when multi-viewpoint shooting was performed using a perspective projection camera with a limited FoV, the blank region was observed on the screen, depending on the point at which the gazing point was set.

2.2 Tracking for Omnidirectional Camera

Im et al. [9] and Caruso et al. [10] estimated the position and orientation of a fisheye camera capturing 180° FoV video. However, because the methods were based on the simultaneous localisation and mapping (SLAM) technique, images had to be captured continuously (i.e., very small interval).

Torii et al. [11] realised omnidirectional camera tracking by developing an image-matching method that worked well with omnidirectional images recorded by equirectangular projection. This method used limited pixels near the centre of the captured image, because pixels at the edge often had large distortions. Thus, it did not fully utilise the features of omnidirectional cameras. Taira et al. [12] rotated the omnidirectional images in a spherical coordinate system to transform the largely distorted edge portion of images to the central portion. Thus, accurate camera tracking was realised utilising a wide (i.e., omnidirectional) FoV.

Morales et al. [13] extracted the feature points after dividing the omnidirectional image and directly estimated the position and orientation of the omnidirectional cameras using a spherical projection camera model.

A perspective projection transformation method via re-projection of omnidirectional images was also proposed. Chen et al. [6] generated perspective projection images using a 60° horizontal FoV and a 50 % overlap rate, and performed image-matching. This method was the simplest approach to omnidirectional camera tracking. However, detailed examination was not given to the FoVs and the overlap rates amongst the perspective projection images. Thus, we verify it, in Section 6.

Omnidirectional camera calibration in the spherical coordinate system and reprojection of omnidirectional images require great calculation costs. Therefore, we seek a time in which we should execute such processes with multi-viewpoint video for realising bullet-time production. Otherwise, the processing cost problem will continue to occur. Thus, we utilise a rapidly developed VR environment technology to effectively realise immersive bullet-time video using an omnidirectional camera.

3 OMNIDIRECTIONAL BULLET-TIME VIDEO

Fig. 3 shows the outline of our proposed omnidirectional bullet-time video generation system. First, omnidirectional cameras are arranged at multiple positions in the shooting space to capture multi-viewpoint omnidirectional images. Then, perspective projection image sets, having different view-directions, are generated from omnidirectional images using the VR development platform. We apply SfM (VisualSFM [14]) to the generated image sets to estimate the parameters of each perspective projection camera and the information of 3D point clouds. The position and orientation of omnidirectional cameras can be calculated by the estimated perspective projection camera parameters. The spherical CG model mapped to the omnidirectional images and the reconstructed 3D point clouds are set in the VR space. A virtual camera is set at the centre of the 3D spherical model, and the orientation of a gazing point is set. By switching the position of the virtual camera one after another to the neighbouring spherical model, omnidirectional bullet-time video is generated.

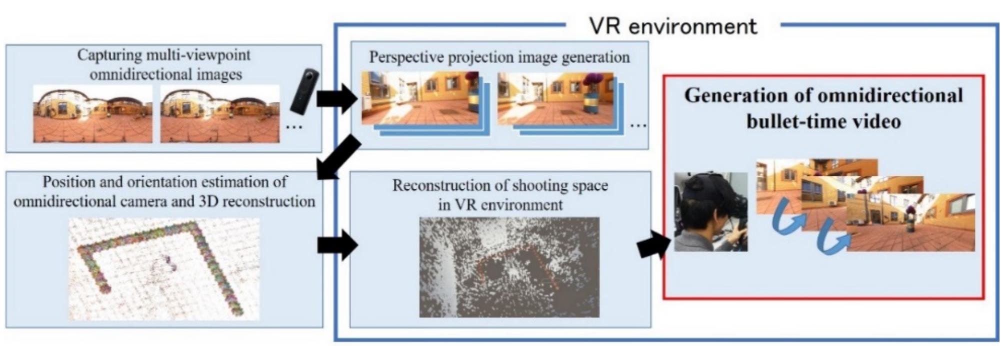

Figure 3: Overview of proposed method. First, omnidirectional cameras capture multi-viewpoint omnidirectional images. Next, the perspective projection image sets are generated in VR environment. SfM is then applied to the image sets to estimate the omnidirectional camera parameters and the 3D point clouds. Based on the estimation result, shooting space are reconstructed in VR space. Finally, we set a gazing point and generate an omnidirectional bullet-time video.

4 GENERATION METHOD OF OMNIDIRECTIONAL BULLET-TIME VIDEO

4.1 Capturing Multi-Viewpoint Omnidirectional Images

We arrange omnidirectional cameras at multiple places in the shooting space and capture multi-viewpoint omnidirectional images. The cameras' positions and numbers correspond to the viewpoint position and the number of viewpoints of bullet-time video. If all cameras are placed on a same plane, it becomes possible to generate a more smoothly switched bullet-time video. However, our method can handle any kind of camera layout, if they are well calibrated.

4.2 Perspective Projection Image Generation

Fig. 4 shows the outline of the transformation from an omnidirectional image to perspective projection images. We texture-map the captured omnidirectional image inside a prepared spherical CG model in VR space. A perspective projection image is generated by rendering the view from a virtual camera, set at the centre of the spherical model. By repeating the rendering processing while rotating the virtual camera around the centre of the sphere at an interval angle, the perspective projection images, having different view-directions, are perceived from one omnidirectional image. In Fig. 4, eight perspective projection images are generated. The view angle and the overlap rate of perspective projection images are discussed in Section 6. By performing the same processing on all multi-viewpoint omnidirectional images, multi-viewpoint perspective projection image sets are generated.

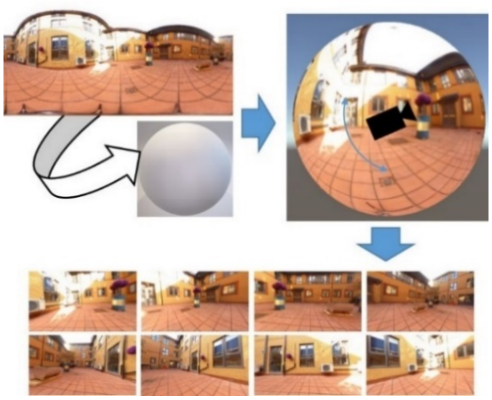

Figure 4: Perspective projection transformation processing.

4.3 Position and Orientation Estimation of Omnidirectional Camera and 3D Reconstruction

We apply SfM to the multi-viewpoint perspective projection image sets generated in the previous section. Parameters of each perspective projection camera are estimated, and the approximate shape of the shooting space (i.e., 3D point clouds) is reconstructed. The parameters (i.e., position and orientation) of the omnidirectional camera can be calculated from the estimated perspective projection camera parameters. Logically, the positions of perspective projection cameras estimated by perspective projection image sets generated from the same omnidirectional images should be the same. However, as shown in Fig. 5, owing to the influence of estimation error, they are not completely the same. Therefore, we must calculate the estimation accuracy from the re-projection error of each camera and calculate the weighted average (i.e., the centre of gravity of the camera position) using this as the weight. We let this value be the position of the omnidirectional camera. The orientation of the omnidirectional camera is then calculated as the orientation of the reference perspective projection camera. In this example, a perspective projection camera, whose optical axis passes through the image centre of the omnidirectional image, is taken as the reference camera.

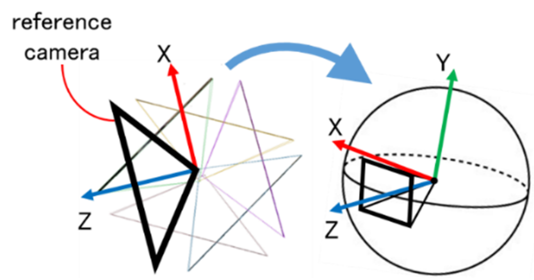

Figure 5: Position and orientation estimation of omnidirectional camera using perspective projection camera sets generated from an omnidirectional image.

4.4 Reconstruction of Shooting Space in VR Environment

We reset the texture-mapped spherical model described in Section 4.2 into the VR space, based on the position and orientation of the omnidirectional camera, as estimated in Section 4.3. The centre of the spherical model (i.e., the position of the omnidirectional camera) is the position of the virtual camera: the viewpoint position of the omnidirectional bullet-time video. Additionally, we place the 3D point clouds reconstructed in the VR space, as described in Section 4.3. This is used for resetting the gazing points, described later.

4.5 A Gazing Point Setting

We next set a gazing point required for omnidirectional bullet-time video, as shown in Fig. 6. First, we orientate a virtual camera to observe a certain direction. We search the 3D point cloud in the VR space on the optical axis of the virtual camera to locate a point cloud set as a gazing point. By referring to the 3D coordinates of the gazing point and the position of the omnidirectional camera, the orientation of each virtual camera is estimated, such that the gazing point is observed at the same point on the display. Omnidirectional bullet-time video is generated by sequentially switching the position of the virtual camera, one after another. Different from ordinary bullet-time video, generated by perspective projection cameras having limited FoVs, any blank region is unobserved, even if we set the gazing point arbitrarily.

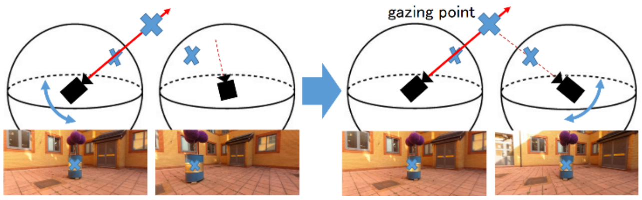

Figure 6: Calculating orientation of virtual camera for setting a gazing point.

5 DISPLAYING IMMERSIVE BULLET-TIME VIDEO

By displaying the video acquired by the virtual camera, as described in the previous section, it is possible to observe immersive bullet-time video. In this section, we introduce a display method using HMD that can display an immersive view. As shown in Fig. 7, a user places an HMD on the head and holds the controller with both hands. Omnidirectional images mapped to spheres are preferably viewed from the centre of the sphere. We set the orientation of the virtual camera as the orientation of the user's head, so that he or she can observe the video while rotating the head.

For generating bullet-time video, the observer orients the optical axis of the virtual camera using a controller held with a hand. Then, the 3D position of the gazing point is estimated. Maintaining the orientation of the optical axis to the gazing point (i.e., the gazing point is always displayed at the centre of the HMD screen), the observer switches the viewpoint of bullet-time video by pushing a controller button. An omnidirectional bullet-time video is then presented on the HMD by setting the gazing point and switching the observation viewpoint via buttons.

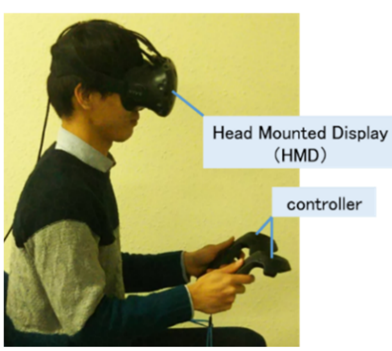

Figure 7: Observing omnidirectional bullet-time video using HMD and controllers.

6 SETTING OF VIEW ANGLE / OVERLAP RATE OF PERSPECTIVE PROJECTION IMAGE

We need to set the view angle and the overlap rate to generate perspective projection images from an omnidirectional image. Thus, we use Unity 2017.2 [15] to render perspective projection images and VisualSFM [14] for SfM to estimate camera position.

6.1 Influence of Projective Distortion

As described in Section 1, the distortion of the perspective projection increases at the edge of the image, because a wider FoV image has larger distortion. We investigate the influence of the distortion of matching with a SIFT feature descriptor. Fig. 8 shows an outline of this verification. First, we extract a part of the omnidirectional image, within the red-coloured rectangle in Fig. 8, as the target object. Then, we generate two perspective projection images, in which the target object is observed at the centre of the screen and at the left edge of the screen. The resolution of the perspective projection image is 1,232 pixels × 693 pixels. We perform a matching process on these two perspective projection images using a SIFT feature descriptor and remove outliers using random sample consensus (RANSAC) [16]. We set the horizontal view angle at intervals of 10° from 50° to 140° and examine the number of correctly matched feature points (i.e., the matching number) and the correct answer rate. From the result, we consider the matching accuracy and the influence of projection distortion caused by the change of view angle.

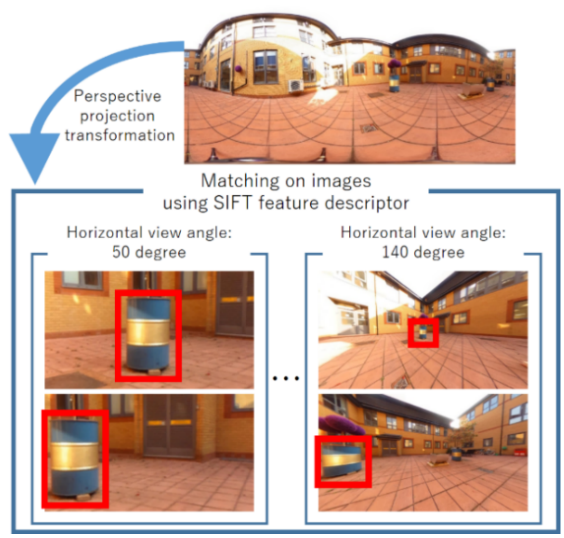

Figure 8: Outline of matching accuracy verification, owing to view angle change.

As shown in Fig. 9, as the view angle increases, the matching number decreases. Additionally, when the horizontal view angle exceeds 140°, correct correspondence is not completely obtained. However, the correct answer rate for a horizontal view angle no more than 130° is more than 70 %, indicating that there are few matching feature point errors. Thus, we can confirm that, when the horizontal view angle is very large, the influence of the projective distortion is also large. However, SIFT feature descriptor-matching effectively functions up to about 130°.

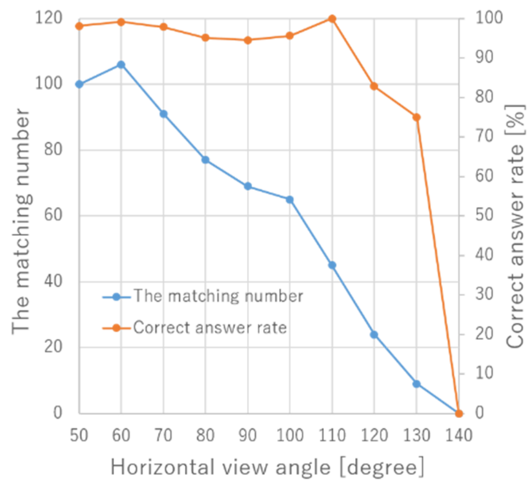

Figure 9: Matching accuracy and influence of projection distortion owing to the change of view angle.

6.2 Validation of Position Estimation Accuracy by View Angle / Overlap Rate

We verify the estimation accuracy of the camera position with respect to the view angle and the overlap rate when performing perspective projection transformation. In this verification, we perform perspective projection transformation processing and camera position estimation via SfM, using omnidirectional images taken from five positions at 1.2 m intervals. Perspective projection images with different viewing directions are generated by repeating the rendering process while rotating the virtual camera at a certain angle in the horizontal direction. Hence, the single orientation angle (e.g., pan angle) is determined by the number of perspective projection images to be generated. This is the division number. The overlap rate between adjacent perspective projection images is thus

$$100 \times \frac{\theta - \frac{360}{n}}{\theta} \ [\%] \tag{1}$$

where the horizontal view angle is θ°, and the division number is n. This overlap rate is determined by the horizontal view angle and the division number. The perspective projection image sets are generated from the given view angle and overlap rate. We apply SfM to these images and estimate the position of each perspective projection camera. Thus, we estimate the position of the omnidirectional cameras and convert it to a real-world scale using the distance between the omnidirectional cameras, measured in advance. We measure the distance between the position of each perspective projection camera and the position of the omnidirectional camera, giving us the average value as 'error distance'. Error distance represents the estimation accuracy of the estimated camera position. We evaluate view angle and overlap rate from this value. We set the horizontal view angle at 50° to 130° at 10° intervals and set the division number so that the overlap rate is about 10 % to 80 %.

Fig. 10 shows the transition of error distance with respect to the overlap rate for each horizontal view angle. As shown in Fig. 10, the error distance is suppressed to 10 cm or less and it decreases as the overlap rate increases. When the overlap rate exceeds about 50 %, the error distance stably keeps a small value, because three overlapping perspective projection cameras can observe the same point in an omnidirectional image. Focusing on the view angle, the estimation accuracy is improved as the view angle becomes wider. However, when the horizontal view angle is 130°, the estimation accuracy decreases. This is caused by the matching error caused by the projective distortion. Additionally, when the view angle is small, unless the overlap rate is increased, it is not determined that the image sets are captured in the same space. Thus, it becomes difficult to calculate the position of the omnidirectional camera.

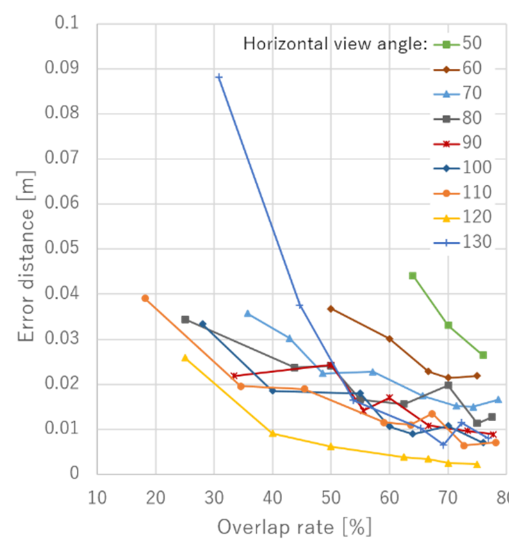

Figure 10: Transition of error distance with respect to the overlap rate for each horizontal view angle.

Since the above results depend on the complexity of the space, it is difficult to uniquely determine the view angle and overlap rate. However, rather than narrowing the view angle, it is better to include wide-view information in one perspective projection image. It is desirable that the view angle is made wide to the extent that the influence of the projective distortion does not appear large. The overlap rate is desirably 50% or more, however, it is necessary to concern about increasing the calculation cost as the overlap rate is increased. Therefore, in this method, we set the horizontal angle view the perspective projection image to 120° and the overlap rate to 62.5 % (division number: 8).

7 EXPERIMENTAL EVALUATIONS

7.1 Performance

To confirm the effectiveness of our proposed method, we conducted shooting experiments to generate omnidirectional bullet-time videos in the courtyard of University of Surrey, United Kingdom. For generating omnidirectional bullet-time video, 34 multi-viewpoint omnidirectional images were captured at 40 cm intervals, as shown in Fig. 11. For shooting, an omnidirectional camera, Ricoh THETA S [17], was used (resolution of the omnidirectional image was 5,376 pixels × 2,688 pixels). For the processing, a notebook PC equipped with an Intel Core i7-7700 HQ 2.80 GHz, GPU: NVIDIA GeForce GTX 1060, memory: 16.00 GB RAM was used. HTC Vive (HMD) [18] was used for video display. Unity 2017.2 [15] was used for rendering perspective projection images and for generating VR space. VisualSFM [14] was used for SfM to estimate camera parameters and the 3D point clouds. We set 'drum' (a drum can and a purple object) and 'door' (an entrance of the building), in Fig. 11, as gazing points.

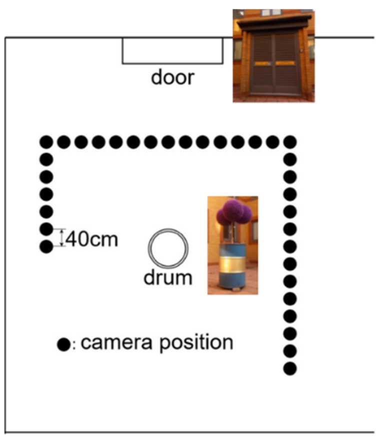

Figure 11: Arrangement of omnidirectional cameras in shooting experiments.

7.2 Result

Fig. 12 shows the perspective projection camera sets estimated by SfM and reconstructed 3D point clouds. Error distance defined in Section 6.2 was 5.97 mm, showing better accuracy than the result shown in Fig. 10. Fig. 13 shows the observer of the omnidirectional bullet-time video presentation using HMD. Imagery matching the rotating head movement was presented to the HMD, give the observer an immersive sensation.

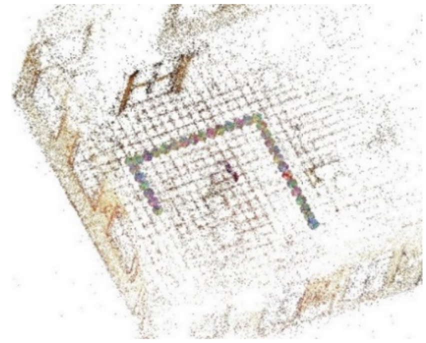

Figure 12: Perspective projection camera sets estimated by SfM and the reconstructed 3D point clouds.

Figure 13: Observer of the omnidirectional bullet-time video using HMD.

Figs. 14 and 15 show the appearance of the omnidirectional bullet-time video presented on the display. In Fig. 14, the gazing point is set to 'drum', and it is observed that the viewpoint is moving while observing the gazing point at the centre of the screen. In Fig. 15, the gazing point is set to 'door'. Similarly, we can confirm that the viewpoint moves while observing the gazing point at the centre of the screen. Without any blank region, the gazing point can be set to 'door', located on the outer side, surrounded by multi-viewpoint cameras, which is advantageous of using omnidirectional cameras.

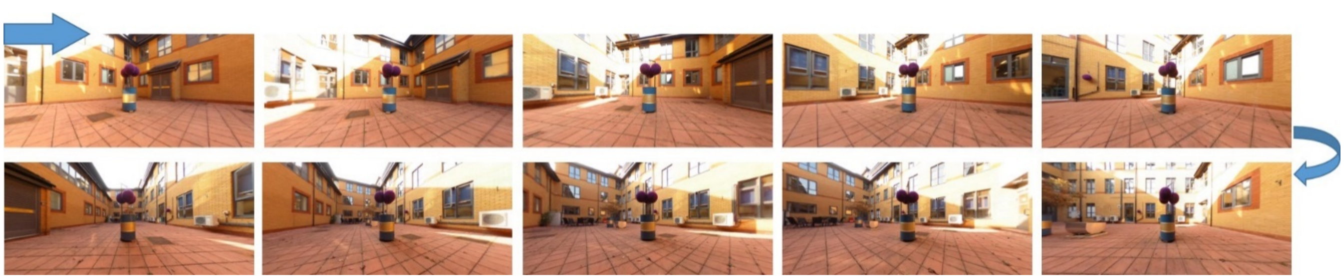

Figure 14: Generation result of omnidirectional bullet-time video (gazing point: 'drum').

Figure 15: Generation result of omnidirectional bullet-time video (gazing point: 'door').

8 CONCLUSIONS

In this research, we proposed a generation method of immersive bullet-time video, using multi-viewpoint omnidirectional images. By using an omnidirectional camera, we solved the problem of the blank region, which others occurs on the display screen, based on the gazing point. Thus, it was possible to greatly expand the observable area of the bullet-time video. Additionally, utilising VR environment technology, we easily estimated the position and orientation of the omnidirectional cameras, set gazing points using a reconstructed 3D point clouds and displayed immersive bullet-time video via HMD. We are going to integrate our bullet-time video with immersive audio to realize an immersive audio-visual display that can improve the presentation performance.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI, Grant Number 17H01772 and JST CREST, Grant Number JPMJCR14E2, Japan. This work was partially supported by the EPSRC Programme Grant S3A: Future Spatial Audio for an Immersive Listener Experience at Home (EP/L000539/1).

REFERENCES

[1] T. Shin, N. Kasuya, I. Kitahara, Y. Kameda, and Y. Ohta. 2010. A Comparison Between Two 3D Free-Viewpoint Generation Methods: Player-Billboard and 3D Reconstruction. 3DTV Conference, Tampere, Finland, June 2010. DOI: https://doi.org/10.1109/3DTV.2010.5506605

[2] Y. Liu, Q. Dai, and W Xu. 2010. A Point-Cloud-Based Multiview Stereo Algorithm for Free-Viewpoint Video. IEEE Transactions on Visualisation and Computer Graphics, 16, 3 (May-June 2010), 407-418. DOI: https://doi.org/10.1109/TVCG.2009.88

[3] N. Akechi, I. Kitahara, R.Sakamoto, and Y. Ohta. 2014. Multi-Resolution Bullet-Time Effect. ACM SIGGRAPH ASIA 2014, Shenzhen China, December 2014. DOI: https://doi.org/10.1145/2668975.2669015

[4] J. P. Snyder. 1993. *Flattening the Earth: Two Thousand Years.* University of Chicago Press.

[5] D. G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 60, 2 (November 2004), 91-110. DOI: https://doi.org/10.1023/B:VISI.0000029664.99615.94

[6] D. M. Chen, G. Beatz, K. Köser, Sam S. Tsai, R. Vedantham, T. Pylvänäinen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk. 2011. City-Scale Landmark Identification on Mobile Devices. Proceedings of the IEEE Conference on CVPR, 737-744. DOI: https://doi.org/10.1109/CVPR.2011.5995610

[7] T. Kanade et al. 'Eye Vision'. http://diva.library.cmu.edu/Kanade/kanadeeye.html,

[8] K. Ikeya and Y. Iwadate. 2016. Multi-Viewpoint Robotic Cameras and their Applications. ITE Transactions on MTA, 4, 4 (November 2016), 349-362. DOI: https://doi.org/10.3169/mta.4.349

[9] S. Im, H. Ha, F. Rameau, H. Jeon, and G. Choe. 2016. All-Around Depth from Small Motion with a Spherical Panoramic Camera. European Conference on Computer Vision, Amsterdam, The Netherlands, October 2016. 156-172.

[10] D. Caruso, J. Engel, and D. Cremers. 2015. Large-Scale Direct SLAM for Omnidirectional Cameras. IEEE/RSJ International Conference on IROS, Hamburg, Germany, September-October 2015. DOI: https://doi.org/10.1109/IROS.2015.7353366

[11] A. Torii, M. Havlena, and T. Pajdla. 2009. From google street view to 3D city models. 2009. IEEE 12th International Conference on Computer Vision Workshops, Kyoto, Japan, September-October 2010, 2188-2195. DOI: https://doi.org/10.1109/ICCVW.2009.5457551

[12] H. Taira, Y. Inoue, A Torii, and M. Okutomi. 2015. Robust Feature Matching for Distorted Projection by Spherical Cameras. IPSJ Transactions on CVA, 7 (July 2015), 84-88. DOI: https://doi.org/10.2197/ipsjtcva.7.84

[13] C. Morales, N. Chiba, and T. Oishi. 2017. Outdoor Omnidirectional View Interpolation based on Accurate Image Rectification. MIRU.

[14] C. Wu. VisualSFM: A Visual Structure from Motion System. http://ccwu.me/vsfm

[15] Unity. https://unity3d.com/jp

[16] M. A. Fischler and R. C. Bolles. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM, 24, 6 (June 1981), 381-395. DOI: https://doi.org/10.1145/358669.358692

[17] RICOH THETA S. https://theta360.com/ja/about/theta/s.html

[18] VIVE. https://www.vive.com/jp/