An On-site Visual Feedback Method Using Bullet-Time Video

Takasuke Nagai
University of Tsukuba
Graduate School of System and Information Engineering
Japan
nagai.takasuke@image.iit.tsukuba.ac.jp

Hidehiko Shishido
University of Tsukuba
Center for Computational Sciences
Japan
shishido@ccs.tsukuba.ac.jp

Yoshinari Kameda
University of Tsukuba
Center for Computational Sciences
Japan
kameda@iit.tsukuba.ac.jp

Itaru Kitahara
University of Tsukuba
Center for Computational Sciences
Japan
kitahara@ccs.tsukuba.ac.jp

ABSTRACT

This paper describes an on-site visual feedback method that executes all processes from capturing of multi-view videos to generating and displaying bullet-time videos in real-time. In order to realize the on-site visual feedback in a dynamic scene where the subject moves around, such as a sports scene, it is necessary to automatically set the target point to where an observer pays attention. We combine an RGB-D camera that detects the position of the subject with our developed bullet-time video generation method in real-time, and achieve automatic setting of the target point based on the measured 3D position. Furthermore, we incorporate a function to detect a keyframe and automatically switch the viewpoint, to enable easier and more intuitive observation. [1]

KEYWORDS

Multi-view videos; Visual feedback; Bullet-time video; Motor learning; Object tracking; Automatic viewpoint switching; Real-time processing

ACM Reference format:
T. Nagai, H. Shishido, Y. Kameda and I. Kitahara. 2018. An on site visual feedback method using bullet-time video. In *1st InternationalWorkshop on Multimedia Content Analysis in Sports (MMSports'18), October 26, 2018, Seoul, Republic of Korea.* ACM, NY, NY, USA, 6 pages. https://doi.org/10.1145/3265845.3265853

MMSports'18, October 26, 2018, Seoul, Republic of Korea
 ACM ISBN 978-1-4503-5981-8/18/10…$15.00
https://doi.org/10.1145/3265845.3265853

1 INTRODUCTION

Visual feedback is attracting research attention for the development of motor learning [1–3], as it enables objective observation of the body and movement. In the common visual feedback methods, subjects observe their own body using a mirror or captured videos. Therefore, it is difficult for the subject to observe their own from various viewpoints. Observation from various viewpoints can be achieved by generating a 3D CG (Computer Graphics) model of the subject using a 3D reconstruction technique or a motion capture device [4,5]. However, the distortion in the synthesized 3D CG model may adversely affect the observation.

To address this problem, we propose a viewport visual feedback method using multiple cameras capturing the same scene from various viewpoints (i.e., multi-view videos). One of the technologies to effectively present a multi-view video is bullet-time video [6,7], which realizes a visual effect of a viewport moving around the subject by sequentially switching/displaying the multi-view videos along with the layout of the shooting cameras. Unlike free-viewpoint video [8-13], generating bullet-time video does not require reconstruction of a 3D image of the target scene; thus, it is possible to generate and present a high-quality image with a low processing cost, which is an important issue for visual feedback. Therefore, we consider bullet-time video an appropriate approach for the viewport visual feedback in motor learning.

In order to install the bullet-time video in an on-site visual feedback system, two requirements must be satisfied. The first one is "real-time generation of bullet-time video". An on-site visual feedback can be classified into two types: "real-time visual feedback" performed during action and "delayed visual feedback" performed immediately after action. Both are effective for motor learning. In the case of delayed visual feedback, the shorter the time required for feedback, the better. However, handling multi-view videos is a problem for real-time or short-time visual feedback owing to the large amount of video data. In this research, we propose a system to realize real-time/delayed visual feedback by processing all video data on a single computer to

minimize the video transmission time, thereby reducing the time required for generating a bullet-time video.

The second requirement is to simplify the operation of the viewer as much as possible. For that purpose, we introduced two approaches: (a) automatic target point setting, and (b) automatic viewpoint switching.

(a) Automatic target point setting: In bullet-time video, in order to realize smooth viewpoint switching, it is necessary to adjust the direction of the shooting camera such that the optical axis of the multi-view camera crosses at one point (target point) in the capturing space. This processing is referred to as target point setting. Akechi [14] proposed bullet-time video generation methods that allow viewers to freely reconfigure the target point. In these methods, it is necessary to input the target point information every time the subject moves. Therefore, it is not realistic to apply them to dynamic scenes such as sports where objects are constantly moving. In this research, we realize the automation of target point setting by combining a real-time 3D sensor (RGB-D camera) and a real-time bullet-time video generation method.

(b) Automatic viewpoint switching: In motor learning, when the posture of the subject at a certain moment is particularly important in the motion, we call that moment as a keyframe. By using multi-view videos, it is possible to observe keyframes from various angles [15]. However, if the number of viewpoints is massive, it takes time to switch the viewpoint to the direction in which a keyframe is most easily and intuitively observed. In this research, by detecting keyframes, we realize the automation of viewpoint switching to the most suitable observation direction.

2 ON-SITE VISUAL FEEDBACK METHOD WITH BULLET-TIME VIDEO

Fig. 1 shows the overview of our on-site visual feedback method with automatic target point setting and automatic viewpoint switching in bullet-time video presentation. A plurality of cameras (hereinafter referred to as multi-view camera) and an RGB-D camera are installed around the subject. The multi-view camera captures multi-view video for generating a bullet-time video. The RGB-D camera is used to estimate the 3D position and skeleton information of the subject for posture detection. Structure from motion (SfM) [16] is applied to the multi-view and RGB-D videos to estimate camera parameters. Automatic target point setting is realized by using the camera parameters and the 3D position of the subject estimated from the RGB-D video. At the same time, automatic viewpoint switching is realized by detecting keyframes and using a correspondence table between the keyframes and viewpoint. On-site visual feedback of bullet-time video is realized by performing these three processes, namely, bullet-time video generation, automatic target point setting, and automatic viewpoint switching, in real-time.

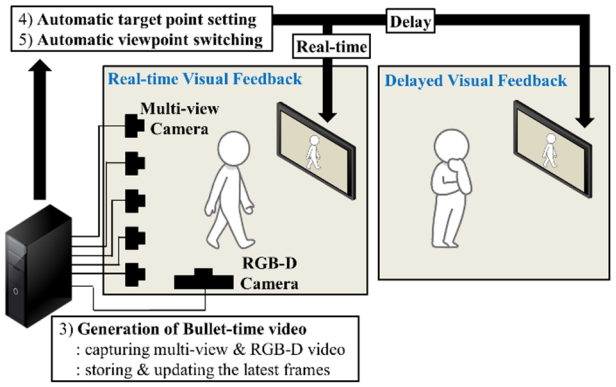

Figure 1: System overview: Multi-view video and RGB-D video are acquired with one computer. Storing and updating the latest frames for play back. Real-time visual feedback and delayed visual feedback of bullet-time video with automatic target point setting and automatic viewpoint switching is realized.

3 GENERATION OF BULLET-TIME VIDEO

3.1 Multi-view Video Capturing

As shown in Fig. 2, the multi-view camera is arranged around the subject. As the installation position of the camera influences the smoothness of the viewpoint switching of bullet-time video, it is better to arrange the cameras on almost the same horizontal plane. However, as the direction of the camera can be adjusted by the target point setting process described later, it should be positioned to capture the same area approximately. The multi-view camera is connected to a computer via network cables for generating the bullet-time video. By feeding the captured video directly to the computer, manual operation such as moving the multi-view video data is eliminated and real-time processing is realized.

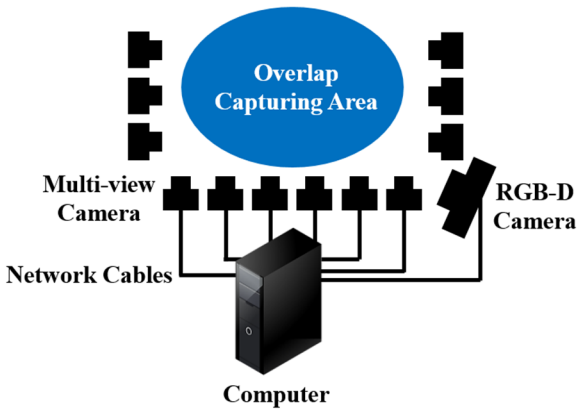

Figure 2: Bullet-time video generation method: the multi-view camera and RGB-D camera are arranged around the subject such that the overlap capturing area becomes as large as possible.

3.2 Calibration of Multi-view Camera

Camera calibration is performed using SfM to estimate the intrinsic and extrinsic parameters of the multi-view camera. In SfM, feature points are first extracted from the image using the image feature [17], and corresponding points among the multi-view images are detected. The motion (relative position and orientation information) of the camera is estimated based on the corresponding point information, and simultaneously, the 3D position of the corresponding point is estimated by stereo vision. After performing these processes for all image pairs, we apply bundle adjustment [18] to refine the estimated 3D position of the corresponding point and the camera parameters.

When calibrating N multi-view cameras, the intrinsic parameter matrix $\boldsymbol{A_{cm}}$ of the camera $C_m(m = 1, \dots, N)$ is expressed by Equation (1), where f_m is the focal length of the camera C_m estimated using SfM, and (c_{x_m}, c_{y_m}) are the image coordinates of the intersection of the optical axis of the camera and the image plane.

$$\boldsymbol{A_{cm}} = \begin{bmatrix} f_m & 0 & c_{x\_m} \\ 0 & f_m & c_{y\_m} \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

As shown in Fig. 3, the rotation matrix $\boldsymbol{R_{cm}}$ and the translation vector $\boldsymbol{t_{cm}}$ representing the position and orientation of the camera C_m in the world coordinate system can also be estimated using SfM. The above processing is performed after the multi-view camera is installed, and the estimated parameters are used at the time of execution.

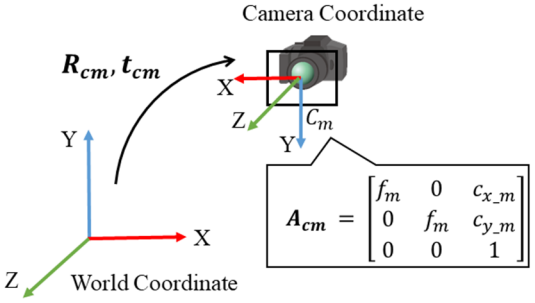

Figure 3: Camera parameter estimation using SfM.

3.3 Visual Feedback of Bullet-Time Video

When the system is starting up, the computer boots the multi-view camera that starts capturing the multi-view video. The computer sends a frame update signal to the cameras at regular time intervals and acquires frames from all the cameras. Real-time visual feedback is realized by presenting the acquired frame immediately. Alternatively, delayed visual feedback is realized by storing and updating the latest frame group.

4 AUTOMATIC TARGET POINT SETTING

4.1 3D Position Estimation of Target Point

The 3D position of the target point is estimated when the user specifies a new target point in a certain multi-view video $I_m(m = 1, \dots, N)$. First, using the camera parameter information estimated as described in Section 3.2, a fundamental matrix $\boldsymbol{F}$ between the image I_m and the image $I_n(n \neq m, n = 1, \dots, N)$ is calculated. The epipolar line on the image I_n corresponding to the target point $\boldsymbol{g_m}(u_m, v_m)$ given for the image I_m is calculated by using the fundamental matrix $\boldsymbol{F}$. The corresponding point $\boldsymbol{g_n}(u_n, v_n)$ of the target point is estimated by searching for the corresponding point of the target point given on the epipolar line. The 3D coordinates $\boldsymbol{M}(X_t, Y_t, Z_t)$ of the target point are calculated by applying stereo vision to the corresponding point between the two images.

4.2 Calculation of 2D Projective Transformation Matrix

Fig. 4 illustrates the process of applying 2D projective transformation to all the multi-view videos such that the new target point given in the previous section is observed at the same coordinates. The unit vector $\boldsymbol{e_{mz}}$ from the optical centre of the image I_m to the target point is taken as the new z axis in the camera coordinate system of the image I_m. The unit vector $\boldsymbol{e_{mx}}$ obtained by multiplying $\boldsymbol{e_{mz}}$ and the negative direction of the y axis of the world coordinate system is defined as the new x axis. Let the outer product $\boldsymbol{e_{my}}$ of $\boldsymbol{e_{mx}}$ and $\boldsymbol{e_{mz}}$ be the new y axis. $\boldsymbol{e_{mx}}$, $\boldsymbol{e_{my}}$, $\boldsymbol{e_{mz}}$ are used to obtain the rotation matrix $\boldsymbol{R'_{cm}}$ by Equation (2). The target points are observed at the centre of the image by applying $\boldsymbol{R'_{cm}}$ to the original image.

$$\boldsymbol{R'_{cm}} = \begin{bmatrix} \boldsymbol{e_{mx}} \\ \boldsymbol{e_{my}} \\ \boldsymbol{e_{mz}} \end{bmatrix} \quad (2)$$

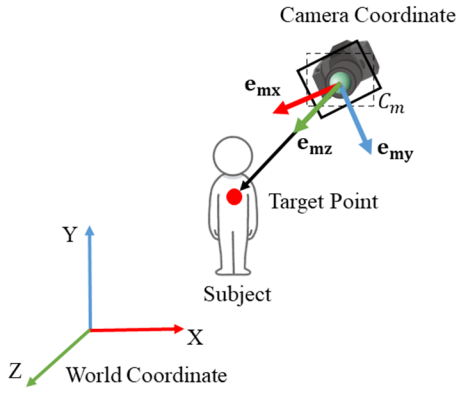

Figure 4: Conversion of a camera coordinate system.

4.3 Calculation of Intrinsic Parameter Matrix

Adjusting the intrinsic parameters prevents changes in the size of the observed object when the viewpoint is switched. The average f_{ave} of the focal length of each camera is calculated. Additionally, the distance d_i from the camera to a new target point in the scene is calculated. The distance d_m from each multi-view camera to the target point is calculated, and the new intrinsic parameter matrix $\boldsymbol{A'_{cm}}$ is calculated using Equation (3). Here, f'_m is a value obtained using Equation (4). By applying $\boldsymbol{A'_{cm}}$ to the original image, the size of the observed object is kept almost constant while changing the viewpoint.

$$\boldsymbol{A'_{cm}} = \begin{bmatrix} f'_m & 0 & c_{x\_m} \\ 0 & f'_m & c_{y\_m} \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$f'_m = f_{ave}\frac{d_i}{d_m} \tag{4}$$

The 2D projective transformation matrix $\boldsymbol{H_m}$ is calculated using Equation (5) where a scale transformation matrix $\boldsymbol{S}$ for reproducing the processing of zoom in/out to the target point is included. By applying the 2D projective transformation matrix obtained for each image, a bullet-time video is generated in which target points are projected to the same coordinates in all images and the size of the object is held constant when the viewpoint moves.

$$\boldsymbol{H_m = SA'_{cm}R'_{cm}R_{cm}^{-1}A_{cm}^{-1}} \tag{5}$$

4.4 Automatic Target Point Setting

Although the 3D position and orientation can be easily measured using a marker attached to the subject, this method restricts the motion of the subject and applying it to a sports scene is difficult. Recently, it has become possible to obtain the 2D position and orientation of the subject's body (skeleton information) by using video tracking technology such as OpenPose [19]. When two or more cameras are available for capturing the target scene, the 3D information can be estimated by applying stereo vision [20]. However, it is not feasible to estimate the skeleton information using more than two cameras in real-time while acquiring multi-view videos using computer resources that can be applied to sports scenes. Therefore, we used an RGB-D camera for real-time measurement of the position and orientation of the subject. In this section, we describe the method for automatically setting the target point in real-time on the subject.

The human skeleton detection process using the RGB-D video enables us to estimate the 3D position of the subject in real-time. The origin of the coordinate system of the estimated 3D position is the principal point of the depth camera. The camera calibration process described in Section 3.2 is also applied to the RGB-D camera to estimate the extrinsic parameters $\boldsymbol{R_k}$, $\boldsymbol{t_k}$. Using these camera parameters, 3D coordinates $\boldsymbol{M_k}(X_k, Y_k, Z_k)$ in the RGB-D camera coordinate system are converted into 3D coordinates $\boldsymbol{M}(X_t, Y_t, Z_t)$ in the world coordinate system as shown in Fig. 5.

By providing the 3D position of the subject estimated from the RGB-D video as a new target point, the target point can be tracked and updated in each frame automatically. Thus, it is not necessary to manually set the target point when the position of the target object changes frequently. By storing the estimated 3D position along with the frame information, it is possible to track the subject even with a delayed visual feedback.

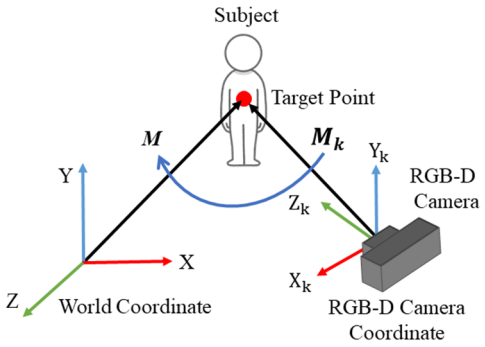

Figure 5: Conversion of RGB-D camera coordinate system to the world coordinate system.

5 AUTOMATIC VIEWPOINT SWITCHING

While the target object is moving, the important moments for motion analysis are called keyframes. When keyframes are detected, the viewpoint is automatically switched to a suitable one where the subject's posture can be observed/understood most easily. This processing is based on a preliminary setup that indicates the correspondence between a type of keyframe and the appropriate viewpoint.

Keyframe detection is realized by applying posture detection processing to RGB-D video. As preliminary processing, our method learns the correspondence between the posture of the subject and the extracted RGB-D pattern. At the time of system execution, posture detection is performed in real-time using the learned data. A plurality of keyframes can be set, and a correspondence table between the keyframes and the viewpoint is input to the system at the time of execution. When a keyframe is detected, video playback is temporarily stopped and the viewpoint is automatically moved to the viewpoint specified in the correspondence table.

6 EXPERIMENT

6.1 Experimental Environment

In this experiment, as shown in Fig. 6, 20 multi-view cameras are placed in a U-shape, and the subject freely moves around the capturing space. The convergence angle is about 5 or 6 degrees (at this angle, it is possible to smoothly switch the video in bullet-time video), and the cameras surround the subject covering approximately 110 degrees in order to capture the

left/right and front sides of the subject. Each camera is attached to a rod fixed at a certain height (approximately 120 cm) using a clamp. Using Kinect v2 by Microsoft Corporation as a 3D position sensor, we acquire the 3D position/attitude of the subject in real-time with a simple configuration (only adding one more sensor).

Figure 6: An overview of camera setting. 20 multi-view cameras are placed in a U-shape

The process from capturing a multi-view video to displaying a bullet-time video is executed on a desktop PC equipped with a CPU (Intel (R) Xeon (R) CPU E5-1620 v4 3.5 GHz), GPU (NVIDIA GeForce GT 710). As the 2D projective transformation process can be calculated independently for each pixel, parallel computation by the GPU is performed using GLSL. For capturing the video, we use 20 Basler network cameras acA1300-30gc and connect all the cameras to one computer using network cables. The resolution of the image is SXGA (1,280 pixels × 1,024 pixels) and the frame rate is 40 fps. Kinect v2 is connected to the computer by a USB cable and it tracks the subject at a frame rate of 30 fps. The posture learning and detection process described in Section 5 is executed using the Gesture Builder provided by Kinect v2 SDK. When learning and testing with the same person, it is possible to detect keyframe with reliability of about 70%. In this experiment, the time required for learning was about 10 minutes. The SfM processing described in Section 3.2 is executed using VisualSFM [21].

6.2 Results and Discussion

Processing time

Table 1 lists the steps in the processing of our system, the average time required for each process, and its standard deviation. The processing from frame acquisition to video display is completed in less than 20 ms. However, the time delay of the video is approximately 120 ms, which is consumed by the transfer of data from the camera to the computer. As a delay of approximately 120 ms is not perceived as a temporal shift, it is acceptable for real-time visual feedback.

The time required for delayed visual feedback is 5 ms, and feedback with almost no time lag is possible. As the latest 300 frames are stored for 20 cameras, even in the case of delayed visual feedback, it is possible to freely change the viewpoint.

Table 1: Processing time of the system

processing	average time (s)	standard deviation
frame acquisition	1.94×10^{-2}	5.47×10^{-3}
3D position estimation	7.14×10^{-5}	2.62×10^{-4}
target point setting	7.40×10^{-5}	2.67×10^{-4}
video output	3.70×10^{-5}	1.92×10^{-4}
total	1.96×10^{-2}	

Automatic target point setting

The subject moves along the route shown in Fig. 7. Fig. 8 shows the video presented at each position ① to ⑥ on the route shown in Fig. 7. It can be seen that the target point is automatically set to an appropriate position on the moving subject.

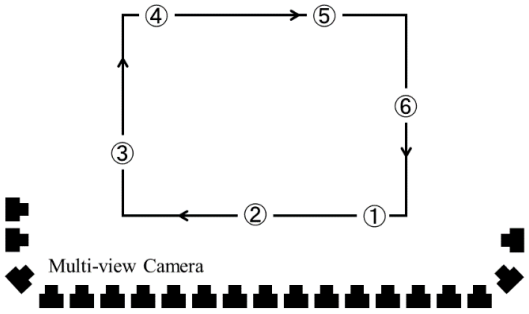

Figure 7: Movement route of the subject. The subject moves from ① to ⑥.

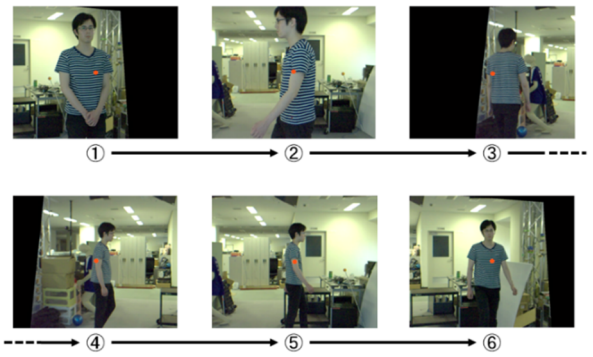

Figure 8: Subject tracking (red dots are the target points).

Automatic viewpoint switching

In this experiment, we choose pitching motion as an example of the target body motion. As shown in Fig. 9, three postures including "swing up", "acceleration", and "release" are set as keyframes. Table 2 shows a correspondence table between the keyframes and the viewpoint. The results of applying this process to the system are shown in Fig. 10 and Fig. 11. Fig. 10 shows which cameras are presented at each frame of the pitching motion. Fig. 11 shows the output images at frames (a),

(b), and (c) indicated in Fig. 10. When the keyframe is detected, it can be seen that the viewpoint is automatically moved to the designated position.

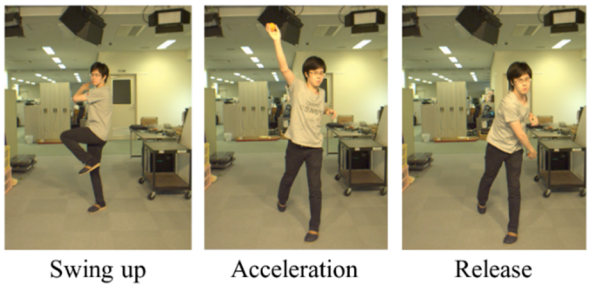

Figure 9: Postures to detect as keyframes.

Table 2: Correspondence table of keyframes and the viewpoint

keyframe	viewpoint
Swing up	1
Acceleration	20
Release	11

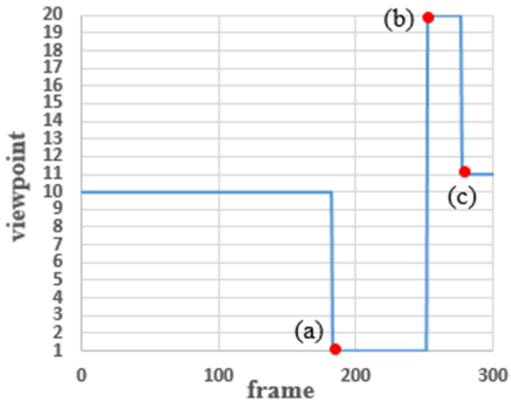

Figure 10: Viewpoint in each frame of pitching motion.

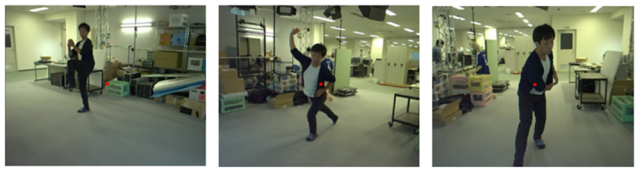

Figure 11: Output images in (a), (b) and (c) of Fig. 10.

7 CONCLUSION

This paper proposed a system that creates a bullet-time video with automatic target point setting and viewpoint switching, and achieves both real-time visual feedback and delayed visual feedback. The experiment results indicate that bullet-time videos can be generated in real-time, and delayed visual feedback is also possible with almost no time lag. In addition, it is possible to automatically reset the target point appropriately and switch the viewpoint without requiring manual operation. We would like to compare the two visual feedback methods, conventional method using a mirror and our proposed method. By using proposed method, we would like to evaluate whether the instructor can easily observe the subject, and evaluate the proficiency level of the motor learning.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number 17H01772 and JST CREST Grant Number JPMJCR14E2, Japan.

REFERENCES

[1] M.W.Kernodle and L.G.Carlton (1992) "Information Feedback and the Learning of Multiple-Degree-of-Freedom Activities." *Journal of Motor Behavior*, 24(2), 187-195
[2] M.Guadagnoli, W.Holcomb and M.Davis (2002) "The Efficacy of Video Feedback for Leaning the Golf Swing." *Journal of Sports Sciences*, 20(8), 615-622
[3] E.Boyer, R.G.Miltenberger, C.batsche and V.Fogel (2009) "Video Modelling by Experts with Video Feedback to Enhance Gymnastics Skills." *Journal of Applied Behavior Analysis*, 42(4), 855-860
[4] U.Yang and G.J.Kim (2002) "Implementation and Evaluation of "Just Follow Me": An Immersive, VR-Based, Motion-Training System." *Presence*, 11(3), 304-323
[5] J.C.P.Chan, H.Leung, J.K.T.Tang and T.Komura (2010) "A virtual reality dance training system using motion capture technology." *IEEE Transactions on Learning Technologies*, 4(2), 187-195
[6] J.G.Lou, H.Cai and J.Li (2005) "A Real-time Interactive Multi-view Video System." *MULTIMEDIA '05 Proceedings of the 13th annual ACM international conference on Multimedia*, 161-170
[7] K.Ikeya and Y.Iwadate (2016) "Multi-Viewpoint Robotic Cameras and their Applications." *ITE Transactions on Media Technology and Applications*, 4(4), 349-362
[8] T.Kanade, P.Rnader and P.J.Narayanan (1997) "Virtualized reality: constructing virtual worlds from real scenes." *IEEE Multimedia*, 4(1), 34-47
[9] H.Saito and T.Kanade (1999) "Shape Reconstruction in Projective Grid Space from Large Number of Images." *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 49-54
[10] J.Carranza, C.Theobalt, M.A.Magnor and H.P.Seidel (2003) "Free-viewpoint video of human actors." *ACM transmissions on Graphics*, 22(3), 569-577
[11] T.Koyama, I.Kitahara and Y.Ohta (2003) "Live Mixed-Reality 3D Video in Soccer Stadium." *Proceeding of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, 178-186
[12] 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." *2010 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video.*
[13] T.Matsuyama, S.Nobuhara, T.Takai and T.Tung (2012) *3D Video and Its Applications*, Springer
[14] N.Akechi, I.Kitahara, R.Sakamoto and Y.Ohta (2014) "Multi-resolution bullet-time effect." *SA'14 SIGGRAPH Asia 2014 Posters*, Article No.30
[15] F.Daniyal, M.Taj and A.Cavallaro (2010) "Content and task-based view selection from multiple video streams." *Multimedia Tools and Applications*, 46(2-3), 235-258
[16] N.Snavely, S.M.Seit and R.Szeliski (2006) "Photo Tourism: Exploring Photo Collections in 3D." *ACM Transactions on Graphics*, 25(3), 835-846
[17] D.G.Lowe (2004) "Distinctive Image Features from Scale-Invariant Keypoints." *International Journal of Computer Vision*, 60(2), 91-110
[18] C.Wu, S.Agarwal, B.Curless and S.M.Seitz (2011) "Multicore Bundle Adjustment." *CVPR '11 Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, 3057-3064
[19] Z.Cao, T.Simon, S.E.Wei and Y.Sheikh (2017) "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields." *CVPR2017*, 7291-7299
[20] https://github.com/CMU-Perceptual-ComputingLab/openpose/blob/master/doc/3d_reconstruction_demo.md
[21] C.Wu, VisualSFM: A Visual Structure from Motion System, http://ccwu.me/vsfm