A Virtual Camera Controlling Method Using Multi-Touch Gestures for Capturing Free-viewpoint Video

Junya Kashiwakuma
University of Tsukuba
Tennoudai 1-1-1
Tsukuba, Ibaraki, Japan
+81 298 53 6470
kashiwakuma@image.iit.tsukuba.ac.jp

Itaru Kitahara
University of Tsukuba
Tennoudai 1-1-1
Tsukuba, Ibaraki, Japan
+81 298 53 6470
kitahara@iit.tsukuba.ac.jp

Yoshinari Kameda
University of Tsukuba
Tennoudai 1-1-1
Tsukuba, Ibaraki, Japan
+81 298 53 6470
kameda@iit.tsukuba.ac.jp

Yuichi Ohta
University of Tsukuba
Tennoudai 1-1-1
Tsukuba, Ibaraki, Japan
+81 298 53 6470
ohta@iit.tsukuba.ac.jp

ABSTRACT
This paper introduces a 3D free-viewpoint video-browsing interface that applies multi-touch manipulation to virtual camera control. It is difficult for general users to browse 3D free-viewpoint video because they are not accustomed to controlling a virtual camera from a free-viewpoint perspective. We focus on the multi-touch interface installed in tablet PCs as an input device, which has become a popular interface thanks to its easy and intuitive manipulation. We conduct subjective evaluations to define suitable gestures for virtual camera control. The results reveal which multi-touch gestures users tend to prefer for controlling a virtual camera.

Categories and Subject Descriptors
H.5.2 [Information interfaces and presentation]: User interfaces—Interaction styles, H.5.1 [Multimedia Information Systems] Artificial, augmented, and virtual realities.

General Terms
Performance, Design, Experimentation, Human Factors.

Keywords
3D free-viewpoint video, user study, 3D navigation, multi-touch gesture

1. INTRODUCTION
As computer vision and image media technologies are well developed, 3D free-viewpoint video generation is garnering much research attention [3][10][11][13][16]. The free-viewpoint video technique makes it possible to observe the target scene from various viewpoints including an immersive view and a bird's-eye view. In a large-scale space like a soccer stadium, the advantage becomes remarkable [10][13]. There are well-developed methods for generating a free-view using computer vision and computer graphics. On the other hand, the interfaces for freely controlling the movements of a virtual camera in a 3D environment still have room for improvement. We consider that developing an easy-to-use browsing interface is necessary for popularizing free-viewpoint video. Further, such an interface has to fully demonstrate the merits of free-viewpoint video. Watanabe et al. [22] developed a 3D free-viewpoint video browsing system that controls a virtual camera by using a 3D position sensor and an overview monitor. The user can watch free-viewpoint videos to smoothly set the viewpoint and the gazing point at same time by using both hands. The overview monitor allows the user to understand the context of the scene that is being captured.

Thus, it is possible to accurately track moving objects such as a ball or players in a scene. However, this requires a lot of equipment, such as a 3D position sensor and large displays. Such a setup might not be available in the outdoor environment or in a living room. Inamoto et al. developed an interface using a head-mounted display (HMD) [10]. Users can enjoy a free-viewpoint video of a soccer game by observing an augmented reality (AR) scene in a dioramic soccer stadium. However, the movable area of the virtual camera is constrained by human body motion.

As shown in Figure 1, we propose an interface that utilizes widely used equipment (e.g., a tablet PC) as an input device, which aims to control a virtual camera as the user chooses. These days, watching videos on a smart phone or a tablet PC has become popular; our approach is an extension of these recent video-browsing trends. In order to develop an intuitive interface, we have to solve an important problem caused by the difference in degrees of freedom between the virtual camera motion and the input positional information given by the tablet PC (multi-touch gestures). In other words, we need to control a virtual camera, which has seven motion parameters (i.e., 3D positions, 3D rotations and a zoom), by using a multi-touch interface, which can input 2D positional information and a click.

As the first step to solving the problem, in this paper we investigate intuitive and efficient multi-touch gestures for controlling a virtual camera that captures free-viewpoint videos. We collect subjective evaluations to define suitable gestures for virtual camera control.

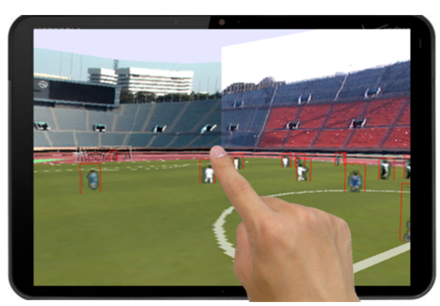

Figure 1. Virtual camera control with multi-touch gestures

EuroITV'13, June 24-26, 2013, Como, Italy.

2. RELATED WORK

In the computer graphics (CG) and virtual reality (VR) research fields, some research has been conducted about virtual camera controlling methods using a two degrees of freedom input-device such as a mouse [4][5][7][12][17][18][19][23][24].

Cohe et al. [2] proposed a method named "tBox" which has a cube-widget around a CG object and users can control its rotation, translation and scale to touch the sides or apexes of the cube.

Hachet et al. [6] proposed an interface, "Navidget", for navigating in a 3D environment using a 2D input device such as a PDA or a mouse. It provided easy 3D navigation by improving point of interest (POI), which is a 3D navigation method proposed by Makinlay et al. [14]. The widgets for the Navidget allowed users to easily control a virtual camera with only one-stroke gestures. This method also provides visual effects and a preview window to compare the displayed view and the virtual camera view after the position/orientation is changed.

Hancock et al. [8] and Martinet et al. [15] proposed a method for controlling CG objects using multi-touch gestures. This method switches the various control methods for CG objects (e.g., 2D and 3D navigation) by referring to the state of specific touch points.

These four approaches work well for controlling a virtual camera in 3D CG space; however, they deal with only static CG objects. In contrast, our target application is for capturing free-viewpoint videos of dynamic scenes such as soccer games. tBox is suitable for realizing CG controlling interface because it is based on formative study to find the most suitable multi-touch gestures. So we propose an intuitive and efficient interface by following the approach. Navidget is a well-developed interface, which has intelligible functions for easy 3D navigation, such as visual widgets and a preview window; however, it is difficult to observe/check the additional windows while capturing free-viewpoint videos in a dynamic scene. We need to realize virtual camera control that allows the user to keep watch on the captured video window. The approach of Martinet et al. [15] for realizing various types of 3D virtual camera control using a 2D input device that refers to the state of touch points is helpful for our development. We apply their proposed method, which focused on controlling CG objects, to controlling a virtual camera to capture free-viewpoint videos.

Human touch gestures and design of virtual camera control in CG space has also been actively researched. Ware et al. [21] conducted experimental evaluations on methods of operating a virtual camera in 3D space. They compared three types of operations, "scene-in-hand", "eyeball-in-hand", and "flying-vehicle", using a 3D position sensor, which has six degrees of freedom. We had to consider how to apply their results to a multi-touch input device, which cannot input as much information at the same time.

In order to realize an intuitive and efficient multi-touch-based interface, Hinrich et al. [9] conducted a study of touch gestures. In the observation, they set up multi-touch tabletop interfaces for browsing visual information in an aquarium, and analyzed the observed users' gestures (e.g., translation, rotation, pushing the buttons, etc.) Purposely, they did not provide a manual for controlling the interface so that they could observe natural human multi-touch gestures. Cohé et al. [1] also displayed 3D objects by using a video projector on a table and asking subjects to control an object without giving any instruction on multi-touch gestures. They analyzed the subjects' multi-touch gestures for controlling rotation, scaling and translation of the 3D object. Although the display device is quite different (i.e., a large projected display and a small touch pad), their approaches are helpful to us. We observed subjects' multi-touch gestures in our free-viewpoint video browsing system without providing instructions for use, and classified the results of the observations in order to identify suitable multi-touch inputs for controlling a virtual camera in each scene. In classifying our experiments, we referred to Cohé's approach that analyzes especially the three kinds of the subjects' gestures, "Form" (the state of the touching finger), "Trajectory" (the locus of motions of fingers) and "Initial Point Location" (start point of fingers). We also analyzed the subjects' touch position, the numbers of fingers they used and the direction of movement.

3. FREE-VIEWPOINT VIDEO BROWSING SYSTEM USING MULTI-TOUCH GESTURES

3.1 Generating free-viewpoint videos

As illustrated in Figure 2, our free-viewpoint video browsing system is based on the 3D live free-viewpoint video system proposed by Koyama et al. [13]. It executes all processes, from capturing multiple images to rendering free-viewpoint video in real-time by using an effective 3D modeling technique, a "player billboard", which represents a target object (e.g., a soccer player) with a single polygon and its texture.

In a scene recognition block, a target scene is captured by using two cameras set in higher places such as rooftops, and the 3D positions of a soccer ball/players are calculated using stereo vision. In a multiple video capturing block, multiple videos to extract the texture information of the players are simultaneously captured. To completely obtain all the texture information without large appearance gaps between the multiple cameras, the system should have a layout of more than eight cameras. The 3D modeling server generates a 3D model of the target objects using the estimated position and the texture information, and transmits the 3D model, which is necessary for rendering an observer's view, to the 3D free-viewpoint browser. In the 3D free-viewpoint browser block, users input a desired viewpoint to observe the soccer action by using a multi-touch interface, which is discussed in this paper. Then the system calculates the camera parameters of the virtual camera and sends them to the 3D modeling server. When the browser block receives a 3D model corresponding to the request, a 3D free-viewpoint image is rendered and displayed using the model.

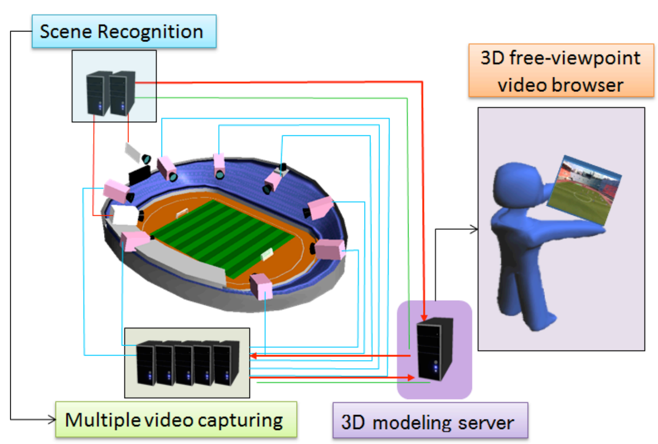

Figure 2. Our proposed free-viewpoint browsing system

3.2 Browsing Interface For Free-viewpoint with Multi-touch Gestures

In this section, we introduce some issues that have to be considered in our free-viewpoint video browsing system, and our approach to solving them.

Our proposed system aims to allow users to comfortably observe free-viewpoint video without having to prepare display equipment. So, as shown in Figure 3, we developed a free-viewpoint video browsing system using a tablet PC as a common information terminal device. Since visual widgets such as GUI controllers sometimes disturb video browsing, and the display size of a tablet PC is much smaller than a PC monitor, we decided not to use any visual widgets, but only multi-touch gestures for control. By using multi-touch gestures, users can input more pointing information at the same time than when using only single-touch gestures. To utilize this feature, we aim to compensate for the difference in degrees-of-freedom between the virtual camera motion and the input positional information given by the tablet PC (multi-touch gestures). A "pinch-in/out gesture", which is commonly used in smartphone and tablet PC interfaces, is a good example.

However, it is difficult to completely compensate for the gap between the virtual camera motion and the input positional information. So we needed to carefully consider the correspondence table between the multi-touch inputs and the virtual camera motion. In the free-viewpoint browsing system, the users control a virtual camera with multi-touch gestures while watching the displayed view. If the user is not satisfied with the view, he/she controls the virtual camera in order to capture a preferred view. In this process, the user has to image motions of the virtual camera to get the preferred view. In order to realize the interface for various types of users, the change of the capturing appearance should be intuitively estimated by multi-touch interface inputs and should be accepted by general consensus. However, there has not been any research aimed at defining multi-touch gestures for controlling a virtual camera for capturing free-viewpoint systems. So in this paper we conduct some subjective evaluation to define them.

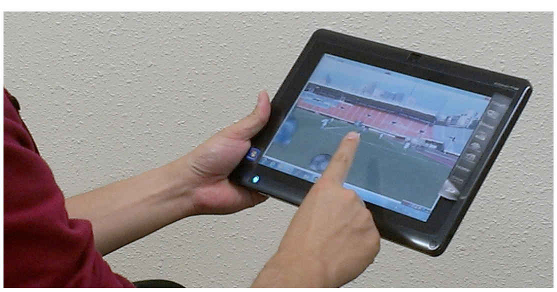

Figure 3. Free-viewpoint video browsing system using a tablet PC

4. SUBJECTIVE EVALUATION TO DEFINE SUITABLE MULTI-TOUCH GESTURES FOR CAPTURING FREE-VIEWPOINT VIDEOS

In this section, we introduce our subjective evaluations for defining intuitive and efficient multi-touch gestures for virtual camera control to capture free-viewpoint videos.

4.1 Procedure and Environments of Our Subjective Evaluation

4.1.1 Warming up and preliminary explanations

In order to familiarize the subjects with observing free-viewpoint videos and controlling the virtual camera using the multi-touch interface, 12 types of short (2 or 3 seconds) free-viewpoint videos were displayed on a tablet PC monitor (the display order was random), and for practice, the subjects were asked to input a multi-touch gesture that controlled the virtual camera that was capturing the video sequence. The free-viewpoint video was continuously playing until the subjects finished the gesture input exercise. Before this warm up, the subjects were informed that control of the virtual camera is realized by controlling the viewpoint (position), the gaze direction (orientation) and the zoom, and they were shown example movies so they could easily understand. The subjects were also informed that they could input gestures using two fingers, that the experiment did not have any correct answers so they could freely gesture as they wanted, and that there was no time limit for the experiments.

4.1.2 Video sequences for subjective evaluations

In our evaluations, the subjects observed short video sequences (5 seconds) generated at a fixed viewpoint, and then were asked to perform a specific camera movement using multi-touch gestures. Each video sequence contained 3D objects such as a ball and soccer players as cues for camera control.

As shown in Table 1, we prepared 12 video sequences. They were divided into 2 groups. Of these, 6 of them are the view from a player standing on the soccer field (field view video); the others are the view looking down the soccer field (bird's-eye view video). Each group was classified into three sub-groups, "gaze direction (orientation)", "camera position" and "zoom", by the controlled camera parameters. The gaze direction (orientation) and camera position were divided into another subclass depending on the direction of movements.

As shown in Figure 4, the field view videos reproduced a scene that a soccer player was watching, and these are expected to give immersive presence to users than ordinary telecasted video sequences. When free-viewpoint video telecasting is put to practical use, users may tend to observe the field view video so it is important to investigate virtual camera control methods using multi-touch gestures in the field view video.

Figure 4. Examples of field view videos

One of the typical camera shots in ordinary telecasted video sequences is the view overlooking a large area at a glance, as shown in Figure 5. The audiences can immediately understand the situation in a soccer game by observing the video sequence. The bird's-eye view video that enhances this overview feature is one of the advantages of free-viewpoint video. It is also important to investigate the virtual camera control methods using multi-touch gestures in the bird's-eye view video.

Figure 5. Example of bird's-eye view video

4.1.3 Instructions for subjects

In all evaluations, we asked subjects to imagine controlling the virtual camera to keep watching a ball using their favorite multi-touch gestures. It was possible to input the gestures whenever he/she wanted. In each trial, the attributes of the controllable camera parameters were fixed (e.g., gaze direction (orientation), camera position or zoom, so that we could investigate how the subjects used the different multi-touch gestures to control each attribute of the camera parameters. The subjects were given clear instructions. For example, the instruction for controlling the gaze direction (orientation) was "rotate the virtual camera as if you are shaking your head", for controlling the camera position, it was "shift the camera position" and for controlling zooming, it was "control the field of view of the virtual camera using zoom-in or zoom-out". Then, we explained the context of the target soccer scene. Table 1 shows the detailed instructions the subjects were given in each trial.

4.2 Environment for Subjective Evaluations

We conducted the subjective evaluations described in the previous section with 14 subjects (13 males and one female). They were from 22 years to 28 years old. We sent out questionnaires about their experience of 3D contents and to touch interfaces. The result is shown in Table 3. The evaluations were conducted with users who had various levels of the experience so that we could define the multi-touch gestures preferred by general users.

The equipment used in the evaluation is shown in Figure 6. The subjects sat down with a tablet PC. Although we did not tell the subjects how to hold the tablet PC, the subjects held it comfortably as they browsed free-viewpoint videos. In the experiment, we used ICONIA Tab-W500 PCs produced by Acer. The OS was Windows 7, 32bit version; the display size was 10.1 inches and the resolution was 1280 [pixels] x 800 [pixels]. The tablets had 2GB memory, 32GB HDD, AMD Dual-Core processer C-50 and AMD Radeon HD 6250 Graphic boards. The evaluated videos were generated and displayed at 30 fps.

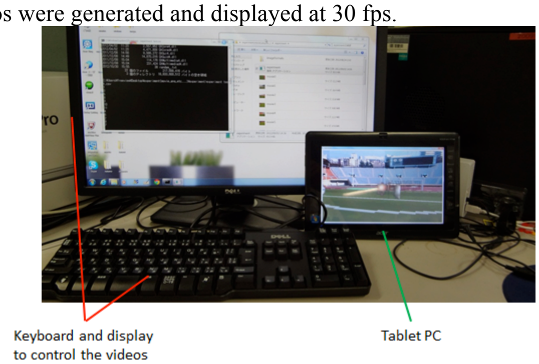

Figure 6. Environment for the experiment

4.3 Results and Discussion

We conducted subjective evaluations with 14 subjects by displaying 12 kinds of video sequences. For each video sequence, the subjects carried out their touch gestures three times. As a result, we extracted 504 multi-touch gesture data. In this section, we introduce the results of our analysis for the data set.

4.3.1 Tendency of the inputted multi-touch gestures

Table 4 summarizes the gestures inputted by the subjects. By monitoring the evaluation scenes, we confirmed that all given touch gestures were somehow meaningful for the virtual camera control in each video. We are curious that the results show almost all subjects choose stroke gestures using one or two fingers to move the virtual camera body (e.g., rotating the orientation and shifting the position), and to choose pinch-in/out gestures to control the zoom parameter.

4.3.2 Gestures for zoom-in/out

In both the field view videos and the bird's-eye view videos, 80-90% of the subjects input the pinch-in/ out gesture to control the zoom parameter (video 5, 6, 11, and 12). In some ordinary multi-touch-based applications such as photo albums, photographs are enlarged with the pitch-out gesture and the size is reduced with the pinch-in gesture. Thus it is reasonable to choose the pitch-in/out gestures to change the viewing size of the displayed videos.

For the other gestures, the subject must draw a circle. To zoom in, he or she draws a clockwise circle; to zoom out, the subject draws a counterclockwise circle. Schmalstieg et al. [20] found that the gesture of drawing a circle is suitable to observe a displayed object carefully; however, the gesture can be applied for a scene that has single object because it is difficult to set the focusing point of zoom-in/out. In our target scene, there were many objects to keep in focus. Thus, we consider that the pinch-in and pinch-out gestures are more suitable for controlling zoom-in and zoom-out functions of the virtual camera for capturing free-viewpoint videos.

4.3.3 Direction of stroke gestures

Figure 7 shows frequency of each stroke gesture inputted in the evaluation for 1-4 and 7-10, which we ask the subjects to move the virtual camera body. In the evaluation, in which the subjects were asked to input the direction of movement for rotating the virtual camera upward or shifting the position forward/right, the results were divided almost in half. One group moved their finger in the same direction as the ordered virtual camera motion. The other moved in the opposite direction (i.e., when the subjects wanted to move the virtual camera to the right, they moved their fingers from right to left on a tablet PC).

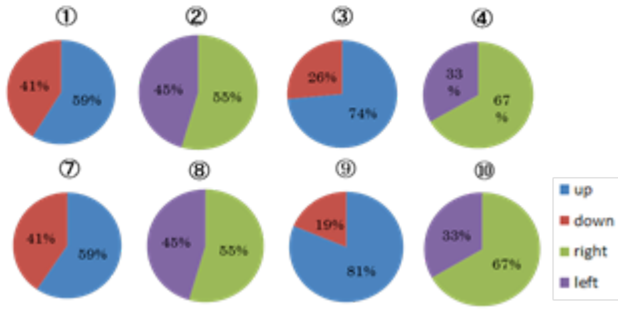

Figure 7. Frequency of stroke gesture inputs in evaluations for videos 1 - 4 and 7 - 10

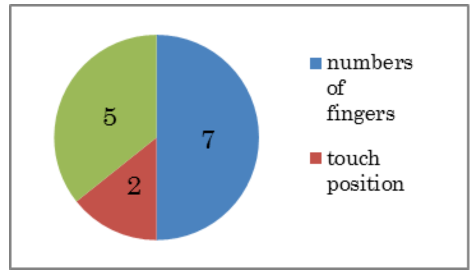

Figure 8. Frequency of each way of switching between the two types of camera control (rotating the orientation and shifting the position)

We investigated the reason by sending out short questionnaires to the subjects who chose the inverse direction stroke gesture. We found that they thought the ordered appearance (view) change could be realized by relatively moving the target space with fixing the virtual camera. For example, the interface of Google maps is designed such that when we want to see to the right of the map, we need to move our finger to left. This interface gives us the feeling given by daily activities such as flipping sheets of paper. We consider such natural behavior is important to designing intuitive multi-touch gestures. On the other hand, since the multi-touch interface should be accepted by general users, it is difficult to ignore that half of the subjects moved their finger in the same direction as the camera motion. There was no correlation between the direction of stroke gestures and the subjects' experience with using multi-touch interfaces. So, we think it is better to leave the choice to the users in a practical system.

4.3.4 Switching controlled parameters

In this evaluation, we found that many subjects switched the two types of camera control (i.e., rotating the camera to the right and shifting the camera position to right) by changing the number of the fingers they used to gesture. Almost all of them controlled the orientation of the camera with one finger and the viewpoint with two fingers. Since it is possible to more exactly point out the touching position, the subjects considered that it was possible to input precise gestures using one finger rather than two. Controlling the orientation of the virtual camera requires more precise operation than controlling the camera position. Even if we shift the view position a little while observing an object in the distance, the disparity is not so large. On the other hand, if we rotate the gaze direction a little in the same situation, the disparity becomes much larger. As a result, the subjects tended to be more careful when controlling the camera orientation and used a one-finger gesture.

Another principal way of switching the two types of camera control was changing the touching position without changing the number of touching fingers. Some subjects input gestures for controlling the camera orientation at the center of the tablet PC display, and for the camera position at the edges of the display. The reason for the difference is that they felt that controlling the camera orientation was an objective motion and controlling the camera position was a subjective one. Thus, they used input gestures like scrolling in a Web browser, which is suitable for changing the value, to control the camera position.

Figure 8 shows the frequency of each way of switching between the two types of camera control. Since half of the subjects chose the number of fingers as a switching method, it seems that switching by changing the number of touching fingers is more suitable for virtual camera control to capture the free-viewpoint videos.

4.3.5 Field View Video and Bird's-Eye View Video

In the evaluations, two types of videos (the field view videos and the bird's-eye view videos) were used. We investigated the difference in the input gestures by observing the two types of video sequences. Figure 10 shows the results of comparing the field view videos (videos 1 - 7) with the bird's-eye view videos (videos 2 - 8). Since the subjects chose almost all the same gestures, it seems that there was not a significant difference. In more detail, we calculated the similarity among the users' operations while using the two types of views. The similarity was given using the ratio of cosine similarity, which is the method to calculate the degree of similarity of the two vectors. The angle between two vectors is calculated with their inner product and magnitude. If the angle becomes small (i.e., the cosine value is close to 1), the two vectors have similar data. We generated 15 dimensional vectors for the 12 free-viewpoint videos used in the experiment, where the vector elements are the number of times the gestures were input by the subjects. Table 2 shows the results of the similarity between two free-viewpoint videos. We would like to note that the similarity between videos 1 and 7; 2 and 8; 3 and 9; 4 and 10; 5 and 11; and 6 and 12 is almost 1. In all these pairs, the subjects were asked to input multi-touch gestures for realizing a same virtual camera control when watching the two types of view (bird's-eye or field view) video.

4.4 Suitable Multi-touch Gestures for Capturing Free-viewpoint Video

To summarize our subjective evaluations, as illustrated in Figure 9, we proposed a multi-touch gesture suite for controlling a virtual camera to capture free-viewpoint videos. First of all, we utilized the extracted data indicating that users prefer to control the camera using simple strokes with one or two fingers. Since controlling the orientation of the camera requires more precise operation, we applied one-finger stroke gestures to controlling the orientation. Alternatively, a stroke gesture with two fingers was applied to the positional control of the virtual camera. For zoom-in or zoom-out control, we applied pinch-in/out gestures. The moving direction of the fingers depends on the user's preferences, so we have left that setting to users.

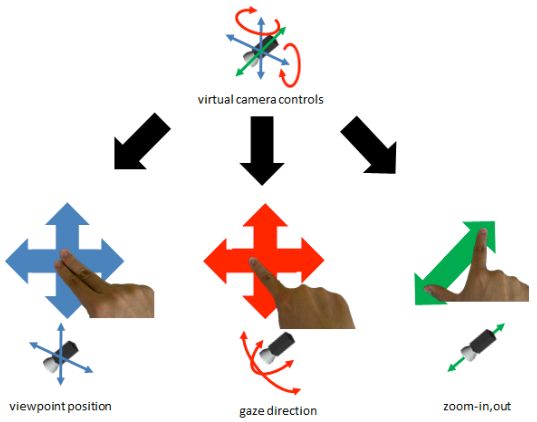

Figure 9. Our proposed multi-touch gestures for capturing free-viewpoint video

5. CONCLUSION

This paper introduced an intuitive and efficient 3D free-viewpoint video-browsing interface that applies multi-touch manipulation to virtual camera control. We focused on the multi-touch interface installed in tablet PCs as an input device. Subjective evaluations to define suitable gestures for virtual camera control were conducted. The results indicated the user's tendencies for virtual camera control and an example of the multi-touch gesture suite for controlling a virtual camera to capture free-viewpoint videos was proposed.

In future work, we will implement a few prototypes of the interfaces based on the extracted tendencies, and conduct user tests for evaluating the proposed multi-touch gestures.

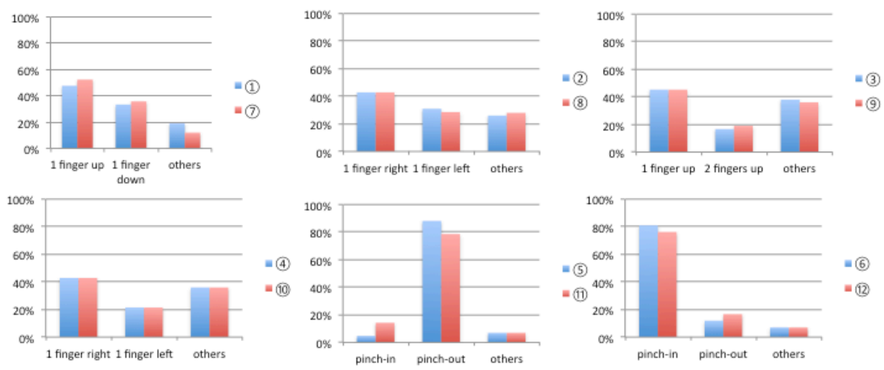

Figure 10. Results comparing the bird's-eye view to the field view videos

Table 1. Displayed videos in our evaluations

Bird's-eye view	Orientation	up	①	The ball is moving to upward. How do you control (rotate) the virtual camera capturing the ball as if you are shaking your head to change the orientation?
		right	②	The ball is moving to the right. How do you control (rotate) the virtual camera capturing the ball as if you are shaking your head to change the orientation?
	Viewpoint	forward	③	The ball is moving forward. How do you control (shift) the virtual camera capturing the ball with translation the camera position?
		right	④	The ball is moving rightward. How do you control (shift) the virtual camera capturing the ball with translation the camera position?
	Zoom	in	⑤	The ball is observed at the center. How do you control to control the field of view of the virtual camera to zoom in the ball?
		out	⑥	The ball is observed at the center. How do you control to control the field of view of the virtual camera to zoom out the ball?
Field view	Orientation	up	⑦	The ball is moving to upward. How do you control (rotate) the virtual camera capturing the ball as if you are shaking your head to change the orientation?
		right	⑧	The ball is moving to rightward. How do you control (rotate) the virtual camera capturing the ball as if you are shaking your head to change the orientation?
	Viewpoint	forward	⑨	The ball is moving forward. How do you control (shift) the virtual camera capturing the ball with translation the camera position?
		right	⑩	The ball is moving rightward. How do you control (shift) the virtual camera capturing the ball with translation the camera position?
	Zoom	in	⑪	The ball is observed at the center. How do you control to control the field of view of the virtual camera to zoom in the ball?
		out	⑫	The ball is observed at the center. How do you control to control the field of view of the virtual camera to zoom out the ball?

Table 2 Results of similarity of cosine

	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫
①	–	–	–	–	–	–	–	–	–	–	–	–
②	0.026	–	–	–	–	–	–	–	–	–	–	–
③	0.863	0	–	–	–	–	–	–	–	–	–	–
④	0	0.933	0	–	–	–	–	–	–	–	–	–
⑤	0	0	0.056	0	–	–	–	–	–	–	–	–
⑥	0	0	0.141	0	0.197	–	–	–	–	–	–	–
⑦	0.993	0.01	0.871	0	0	0	–	–	–	–	–	–
⑧	0.027	0.998	0	0.946	0	0	0.01	–	–	–	–	–
⑨	0.872	0	0.979	0.019	0.011	0.135	0.88	0	–	–	–	–
⑩	0	0.933	0	1	0	0	0	0.946	0.019	–	–	–
⑪	0	0	0.073	0	0.98	0.359	0	0	0.029	0	–	–
⑫	0	0	0.146	0	0.237	0.992	0	0	0.139	0	0.382	–

Table 3. Experience and knowledge of the subjects

Questions								
Usually use a tablet PC?	Yes	8	No	6				
Familiar with for 3D contents?	Well	8	Sometimes	4	A little	2	Not at all	0

Table 4. The input gestures in the subjective evaluations

	1 finger up	1 finger down	1 finger right	1 finger left	2 fingers up	2 fingers down	2 fingers right	2 fingers left	pinch-in	pinch-out	circle(counter clockwise)	circle(clock wise)	1 finger arc	2 fingers arc	2tap
①	48%	33%	0%	0%	7%	0%	0%	0%	0%	0%	0%	0%	7%	5%	0%
②	0%	0%	43%	31%	0%	0%	7%	5%	0%	0%	0%	0%	7%	7%	0%
③	45%	12%	0%	0%	17%	14%	0%	0%	7%	2%	0%	2%	0%	0%	0%
④	0%	0%	43%	21%	0%	0%	14%	14%	0%	0%	0%	0%	0%	0%	7%
⑤	0%	0%	0%	0%	0%	0%	0%	0%	5%	88%	0%	7%	0%	0%	0%
⑥	0%	0%	0%	0%	0%	0%	0%	0%	81%	12%	7%	0%	0%	0%	0%
⑦	52%	36%	0%	0%	7%	0%	0%	0%	0%	0%	0%	0%	0%	5%	0%
⑧	0%	0%	43%	29%	0%	0%	7%	7%	0%	0%	0%	0%	7%	7%	0%
⑨	45%	12%	0%	0%	19%	7%	0%	0%	7%	0%	0%	2%	0%	0%	7%
⑩	0%	0%	43%	21%	0%	0%	14%	14%	0%	0%	0%	0%	0%	0%	7%
⑪	0%	0%	0%	0%	0%	0%	0%	0%	17%	76%	7%	0%	0%	0%	0%
⑫	0%	0%	0%	0%	0%	0%	0%	0%	79%	14%	0%	7%	0%	0%	0%

6. REFERENCES

[1] A. Cohe, M. Hachet, Understanding User Gestures for Manipulating 3D Objects from Touchscreen Inputs, In Proceedings of the 2012 Graphics Interface Conference (GI), 157-164, 2012.

[2] A. Cohe, F, Decle, M. Hachet, tBox: a 3d transformation widget designed for touch-screens, In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI), 3005-3008,2011.

[3] J. Carranza, C. Theobalt, M. A. Mgnor, and H. P. Seidel, Free viewpoint Video of Human Actors, *ACM Transaction on Graphics*, Vol.22, No. 3, 569-577, 2003.

[4] F. Decle, M. Hachet, P. Guitton, ScrutiCam: Camera Manipulation Technique for 3D Objects Inspection, In Proceedings of the IEEE Symposium on 3D User Interfaces (3DUI), 19-22, 2009.

[5] J. Edelmann, A. Schilling, S. Fleck, THE DABR – A MULTITOUCH SYSTEM FOR INTUITIVE 3D SCENE NAVIGATION, In Proceedings of the 3DTV-Conference:The True Vision – capture, Transmission and Display of 3D Video (3DTV-CON), 1-4, 2009.

[6] M. Hachet, F. Decle, S. Knodel, P. Guitton, Navidget for Easy 3D Camera Positioning from 2D Inputs, In *Proceedings of IEEE Symposium on 3D User Interfaces* (3DUI), 83-89, 2008.

[7] M. Hancock, T. Cate, S. Carpendale, Sticky Tools: Full 6DOF Force-Based Interaction for Multitouch Tables, In Proceedings of ACM International Conference on Interactive Tabletops and Surfaces (ITS), 133-140, 2009.

[8] M. Hancock, S. Carpendale, A. Cockburn, Shallow-Depth 3D Interaction: Design and Evaluation of One-, Two-, and Three-Touch Techniques, In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI), 1147-1156, 2007.

[9] U. Hinrichs, S. Carpendale, Gestures in the Wild: Studying Multi-Touch Gesture Sequences on Interactive Tabletop Exhibits, In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI), ACM, 3023-3032, 2011.

[10] N. Inamoto, H. Saito, Immersive Observation of Virtualized Soccer Match at Real Stadium Model, In *Proceedings of International Symposium on Mixed and Augmented Reality* (ISMAR), 188-197, 2003.

[11] T. Kanade, P. Render, P. J. Narayanan, Virtualized Reality: Constructing Virtual Worlds from Real Scenes, *IEEE Multimedia*, Vol.4, No 1, 34-47, 1997.

[12] K. Kin, T. Miller, B. Bollensdorff, T. Derose, B. Hartmann, M. Agrawala, Eden: A professional Multitouch Tool for Constructing Virtual Organic Environments, In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI), ACM, 1343 – 1352, 2011.

[13] T. Koyama, I. Kitahara, Y. Ohta, Live Mixed Reality 3D Video in Soccer Stadium, In *Proceedings of the 2nd IEEE/ACM*

International Symposium on Mixed and Augmented Reality (ISMAR), 178-187,2003.

[14] J. D. Mackinlay, S. K. Card, G. G. Robertson, Rapid Controlled Movement Through a Virtual 3D Workspace, In *Proceedings of the annual conference on Computer graphics and interactive techniques* (SIGGRAPH), 171-176, 1990.

[15] A. Martinet, G. Casiez, L. Grisoni, The Design and Evaluation of 3D Positioning Techniques for Multi-touch Displays, In *Proceedings of the IEEE Symposium on 3D User Interfaces* (3DUI), 115-118, 2011.

[16] W. Matusik, C. Buehler, R. Rasker, S. J. Gortler, L. McMillan, Image-based Visual Hulls, In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (SIGGRAPH), 369-374, 2000.

[17] C. Moerman, D. Marchel, L. Frisoni, Drag'n Go: simple and fast navigation in virtual environment, In Proceedings of the IEEE Symposium on 3D User Interfaces (3DUI), 15-18, 2012.

[18] T. Ohnishi, R. Lindeman, K. Kiyokawa, Multiple Multi-Touch Touchpads for 3D Selection, In *Proceedings of the IEEE Symposium on 3D User Interfaces* (3DUI), 115-116, 2011.

[19] J. L. Reisman, P. L. Davidson, J. Y. Han, A Screen-Space Formulation for 2D and 3D Direct Manipulation, In Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST), 69-78, 2009

[20] D. Schmalstieg, L. M. Encarnacao, Z. Szalavari. Using transparent props for interaction with the virtual table, In *Proceedings of ACM Symposium on Interactive 3D Graphics* (I3D), ACM, 147-153, 1999.

[21] C. Ware, S. Osborne, Exploration and Virtual Camera Control in Virtual Three Dimensional Environments, In *Proceedings of ACM Symposium on Interactive 3D Graphics* (I3D), ACM, 175 – 183, 1990.

[22] T. Watanabe, I. Kitahara, Y. Kameda, Y. Ohta, 3D Free viewpoint Video Capturing Interface by Using Bimanual Operation, In Proceedings of the 3DTV-Conference:The True Vision – capture, Transmission and Display of 3D Video (3DTV-CON), 2010.

[23] J. O. Wobbrock, M. R. Morris, A. D. Wilson, User-defined gestures for surface computing, In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI), 1083-1092, 2009.

[24] R. C. Zeleznik, A. S. Forsberg. Unicam – 2d gestural camera controls for 3D environments, In *Proceedings of ACM Symposium on Interactive 3D Graphics* (I3D), ACM, 169-173,1999.