World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:8, No:2, 2014

Multiplayer RC-Car Driving System in a Collaborative Augmented Reality Environment

Kikuo Asai, Yuji Sugimoto

Abstract**—**We developed a prototype system for multiplayer RC-car driving in a collaborative augmented reality (AR) environment. The tele-existence environment is constructed by superimposing digital data onto images captured by a camera on an RC-car, enabling players to experience an augmented coexistence of the digital content and the real world. Marker-based tracking was used for estimating position and orientation of the camera. The plural RC-cars can be operated in a field where square markers are arranged. The video images captured by the camera are transmitted to a PC for visual tracking. The RC-cars are also tracked by using an infrared camera attached to the ceiling, so that the instability is reduced in the visual tracking. Multimedia data such as texts and graphics are visualized to be overlaid onto the video images in the geometrically correct manner. The prototype system allows a tele-existence sensation to be augmented in a collaborative AR environment.

Keywords**—**Multiplayer, RC-car, Collaborative Environment, Augmented Reality.

I. Introduction

AUGMENTED Reality (AR) technologies have been widely used for creating various applications that enhance the real world with virtual objects [1]. Typical AR applications are games, which are made more interactive and attractive for players by having virtual objects superimposed onto the real world. Although computer games also give players an interactive environment and enjoyable experiences, they are limited to the virtual world composed of images and sounds. Even though online games provide players with a collaborative environment, their experiences are still separated from the surrounding physical environment and neighboring people, which becomes the main reason many parents are afraid that computer games may adversely affect their children [2]. AR creates a collaborative game environment that multiplayers can enjoy together, sharing their spatial experiences in the real world.

One of the remarkable features in AR games is the display of images and sounds from the first-person point of view, which gives players high immersion during the game. In general, the first person corresponds to a player in AR games, and scenes of the real world are captured by a visual camera attached on the player's head. The information additional to the real world is superimposed in a player's point of view. Actually, most AR games have been designed to be played from a player's point of view. In car driving games, unfortunately, a bird's-eye view of a real floor or table has been provided with an HMD [3] and a mobile device [4], because it is easy to capture the scene images from the point of view from a camera and to render a synthesized car driven by the player. However, when a player is supposed to get in a vehicle, the point of view should exist in the vehicle. Although an AR based RC-car application has been made from the car's point of view [5], it is available only in a single–player format. So far, no multiplayer RC-car game that gives a player scene images from the first-person point of view has been applied to an AR environment.

We developed an RC-car driving system played by plural players having the first-person point of view in a collaborative AR environment. An RC-car is controlled by a player in the physical space, capturing scenes with a visual camera attached on the RC-car. The player can see the scene images from his/her point of view. The marker-based tracking system is implemented to estimate the position and orientation of the camera on the RC car, complementally increasing the accuracy using an infrared camera. The client-server architecture is installed to distribute the processing load to several PCs on the network. We believe that the RC-car driving system provides players with immersive car-driving experiences.

K. Asai is with the Open University of Japan, Chiba, 2618586 Japan (corresponding author to provide e-mail: asai@ ouj.ac.jp).
Y. Sugimoto is with the Doshisha University, Kyotanabe, 6100394 Japan.

II. Related Work

Games are regarded as a promising AR application [6]. Games are typically classified into entertainment games and serious games [7]. This categorization would work well in terms of purposes of AR games but not a manner of interaction. Hardware devices such as displays and input interfaces still dominate the manner of interaction between players or a player and an environment in AR games. Therefore, we describe the related work with respect to interaction techniques mainly based on the hardware devices: a head mounted display (HMD), a handheld device, and projection mapping.

A. HMD

The early work on AR games has often used an HMD as a display device. The kinds of video see-through and optical see-through HMDs are changed to suit AR applications [8]. In general, the video see-though HMD is relatively easier to implement to AR applications than the optical see-through HMD, since the geometric registration of virtual objects on the real scene images is eased in visual tracking.

AR2Hockey [9] is played by two players hitting a virtual puck towards each other on a physical table. The player swings a tracked physical mallet toward the motion of a puck. MIND-WARPING [10] is played by a couple of players who interact with each other through a network. One puts on an HMD with a camera and fights off virtual monsters with his/her hand gestures. The other player remotely controls the virtual monsters on the integrated workbench. AR Quake [11] is played by a player wearing an HMD who tries to slay monsters in indoor and outdoor environments. This game is an extension of the desktop 3D game, Quake, categorized as a first-person shooter (FPS). Touch-Space [12] is played by two players collaboratively finding two maps, finding a castle, and defeating a witch to save a princess. Each task is performed at a different stage: the first stage includes tangible interaction with players in the AR experience; the second stage gives players seamless transition between AR and virtual reality (VR) experiences, and the third stage features collaborative navigation in the immersive VR experiences. AR Battle Commander [13] is played by a player with an army of military units to control. The player tries to destroy the opposition forces by giving commands to his/her units. It is a kind of real time strategy (RTS) game and is played from a first-person point of view in an outdoor AR environment.

B. Handheld Device

Recently, handheld devices are often used for AR games, especially commercially based ones, since they display the scene images captured by a camera attached on the device. Handheld devices such as a smartphone and a mobile PC have the merit of being easier to handle than HMDs. These devices have been widely used in daily lives and have become low in price and running costs. The handheld devices have advantages in terms of safety and vision health, because the player’s field of view is not covered by the display. However, the small display of the handheld devices makes the players’ experiences less immersive. Moreover, the camera view does not always match the user’s view according to the geometric relationship between the handheld device and the user’s viewpoint [14], [15]. This inconsistency results in discontinuity of a scene from the user’s viewpoint.

Invisible Train [16] is played by multiple players steering virtual trains over a real miniature railroad track. Each holds a PDA device with a camera to switch the track at intersections to avoid train collisions. AR Tennis [17] is played by two players hitting a virtual ball by using mobile phones as virtual racquets. The game play has been enhanced with multimodal interaction involving audio and vibration feedback for hitting the ball. AR Domino Knockdown [18] is played by two players knocking virtual dominos off a real table by shooting virtual balls. Each player holds a mobile PC to shoot virtual balls by tapping his/her fingers on the screen. AR Defender [19] is played by a player with the iPhone or another smartphone. A pattern marker is captured by the phone’s camera to place a virtual tower. A player defends the tower from various enemies trying to take the tower by directing the phone’s camera to shoot the enemies. Invizimals [20] is played by a couple of players who hunt and capture creatures. This is a commercial game run on the PlayStation Portable with a camera. Square-shaped markers laid in the real world work as traps to capture the creatures in the virtual world.

C. Projection Mapping

A projection mapping method has also been used as presentation for AR applications. The projection images need to be arranged with the projection targets. When the surface where images are projected has a structure other than a plane, they are registered to align with surface features. When multiple projectors are used in the display screen, the projected images need to be blended correctly [21]. When the surface has some colors, the images need to be modified in color, saturation, and contrast.

The projection mapping overlays the real world with visual information, creating a tangible user interface where the user interacts intuitively with virtual objects by manipulating the corresponding physical objects [22], [23]. A fantasy dice game [24] is played by two players fighting a battle between two wizards who can call creatures up for the battle. The dices are placed on the tabletop space that is captured by a camera, and the tabletop space is displayed on the wall with a projector. Augmented Coliseum [25] is played with multiple players controlling small physical robots to attack each other with virtual weapons. Markers are placed on the robots to measure the position and orientation, and the projected game play space includes virtual objects such as barriers, explosions, and gunfire. The seamless interaction of the virtual objects and the real world is provided in the AR game environment. A multimodal tabletop game [26] is played by two players with multimodal interaction using gesture and speech recognition. Two commercial games (RTS and simulation games) have been implemented into the platform, achieving multimodal co-located tabletop interaction. IncreTable [27] is played by a couple of players solving puzzles by placing virtual domino blocks, moving a virtual car, or controlling a physical robot. The game play interfaces are composed of projection displays, digital pens, a depth-sensing camera, real domino blocks, and physical objects. The virtual domino blocks interact with the real domino blocks through portals. The projection-based AR is also used to enhance guests’ experiences in Disney theme parks [28]. The primary merit is that the projected images can make dynamic environments and breathe life into static sets, which is difficult to achieve by conventional methods.

III. System

The RC-car driving system consists of RC-cars with wireless cameras, PCs connected via the LAN, and an infrared camera set on the ceiling. The player operates an RC-car with an RC controller, viewing the synthesized images at an HMD. The synthesized images are generated by overlaying the computer graphics of virtual objects onto the real scene images captured by a camera on the RC-car. Our prototype system achieves visual stability and occlusion for the overlay by using the geometry data of the RC-cars derived from the additional infrared camera. Fig. 1 shows a screenshot of the scene images displayed at the HMD.

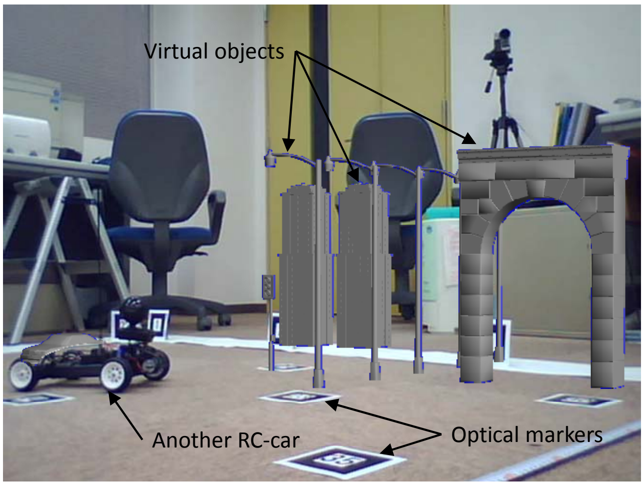


Fig. 1 Example of scene images viewed by player

A. Design

The RC-car driving system was designed to support interaction among multiple players who control their cars with the wireless RC controllers. To make a collaborative AR environment, the equal operation functions are given to all the players, sharing the same physical world with their own viewpoints. This manner of interaction in the environment is similar to that in the real world, though the viewpoint of each player is set on the RC-car in the RC-car driving system.

The field of game play was designed for a player to drive the RC-car anywhere players themselves can set on the ground plane. Although the range of the field does not basically depend on the physical limitation, the artificial boundary of the game play is necessary to maintain the position tracking of the RC-cars and to avoid hard collisions with some real objects around the game play field. The field is formed with a planner array of optical markers that are used for estimating the position and orientation of a camera attached on the RC-car.

On the basis of the transform data of the position and orientation, virtual objects are superimposed onto the scene images of the field in the geometrically correct way. However, the marker-based tracking is considered to be destabilized due to a fast movement of the RC-car, a failure to capture the optical markers, and jitter noise in the video images. Moreover, it is difficult to solve an occlusion problem in which the background virtual objects occlude the foreground real objects in the scene images. An additional camera was prepared to capture the bird's eye view of the game play field, estimating the geometric relationship among the RC-cars. The infrared camera was used to distinguish these markers on the RC-cars from the optical markers on the game play field.

The RC-car driving system was also designed to provide simple user interfaces. Each player sees scene images with an HMD instead of an LCD monitor, enhancing the visual immersion by widely covering the player's view. The scene images are captured by a visual camera attached on the RC-car. The scene changes in the proceeding line by forwarding the RC-car and in the horizontal orientation by directing the RC-car. Any RC-car can be used for the game play, though it needs to have some space for putting a wireless camera on it. An RC controller pertaining to the RC-car kit is used as an interface device to control the RC-car.

B. Configuration

Fig. 2 shows the configuration of the RC-car driving system. The prototype system uses several PCs defined to a manager and clients, so that the computational loads are distributed in graphics rendering and image processing. The manager manages the clients by collecting data transferred from the clients. The manager also estimates the geometry status of each RC-car with the infrared markers. The infrared camera is placed on the ceiling of the room, and three infrared markers forming an isosceles triangle are attached on each RC-car. The isosceles triangle is detected in the infrared images, which estimate the position and orientation of the RC-car. Both the position and orientation of the RC-car derived from the visual tracking are corrected with the position and orientation of the isosceles triangle from the infrared sensing data. The geometry data of the RC-cars are transferred from the manager to the clients through the LAN. The clients render the graphics with the transform data based on the correction of the geometry data, and then display the synthesized images of the scene and graphics at the HMDs. The number of clients depends on the number of players.

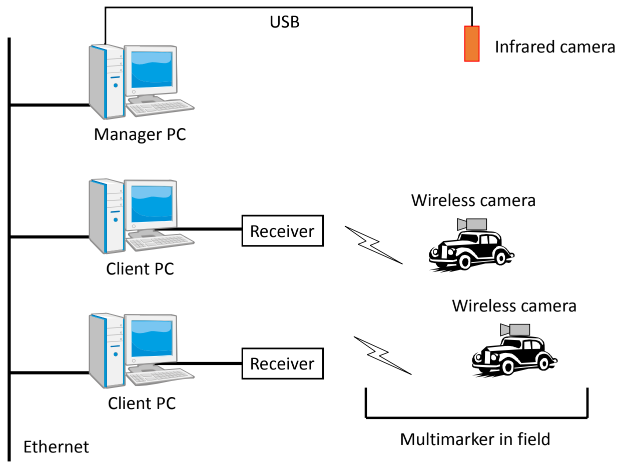


Fig. 2 Configuration of the RC-car driving system

Fig. 3 shows the processing diagram. The manager begins by capturing the infrared images from the infrared camera and displaying them on the PC monitor. The origin of the field needs to be set to define the geometry of the isosceles triangle formed with three infrared markers. A large square marker with retro-reflective materials for infrared light is temporally placed near the center of the game play field, as shown in Fig. 4 (a). When the square marker is detected correctly, the axes of the coordinates are presented as the origin at the monitor in the manager, as shown Fig. 4 (b). Once the origin is set correctly, the square marker can be cleared from the view of the infrared camera. Then, three infrared markers on each RC-car are detected in the images, and the position and orientation are estimated for the corresponding isosceles triangle.

The clients also begin by capturing the video images from the visual camera attached on each RC-car and displaying them at the corresponding HMD. When optical markers with square shapes are detected in the images, the patterns inside the square areas are recognized to identify the location within the field and estimate the position and orientation of the camera. Then, a 3DCG image of virtual objects is superimposed onto the video image at the HMD.

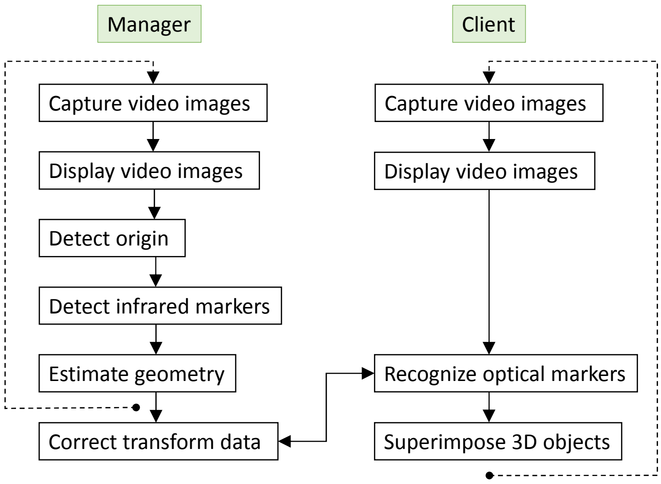


Fig. 3 Processing flow of the RC-car driving system

(a) Temporal marker for origin (b) Monitor of manager

Fig. 4 Setting of origin on images from infrared camera

C. Sensor Fusion and Occlusion

The estimation of position and orientation of a camera attached on each RC-car affects the stability for establishing the coordinates where virtual objects interact with the real world. To obtain the precise transform data for the graphics rendering in each client, the tracking system has the complementary processing for sensor fusion. The geometry data of the visual camera attached on the RC-car are corrected by fusing the geometry data of an isosceles triangle on the RC-car. The matrix $\mathbf{M}$ of the transform data for rendering 3D graphics is expressed as

$$\mathbf{M} = \mathbf{M}_m \times w + \mathbf{M}_c(1 - w) \quad (1)$$

where $\mathbf{M}_m$ and $\mathbf{M}_c$ are the matrices of the geometry data estimated from the infrared tracking at the manager and the visual tracking at the client, respectively. w is the weight that corresponds to the correction coefficient. When no geometry data are observed in the infrared tracking, the weight is set to 0. When no geometry data are observed in the visual tracking, the weight is set to 1. The weight changes with time from recovering the geometry data of both the infrared and visual tracking.

To support multiple players, multiple RC-cars must be able to exist in the game play field. However, when two RC-cars approach each other, it becomes difficult to distinguish the isosceles triangles on the RC-cars in the bird's eye view images from the infrared camera. This results in the failure to estimate the position and orientation in the infrared tracking. We solve this problem by grouping the infrared markers. Three infrared markers of each RC-car form an isosceles triangle that is regarded as a group that distinguishes between the other RC-cars. The group is determined with the center of gravity in the three infrared markers, assuming that the center of gravity does not suddenly change the location. Fig. 5 shows a game play field with optical markers and two RC-cars with infrared markers.

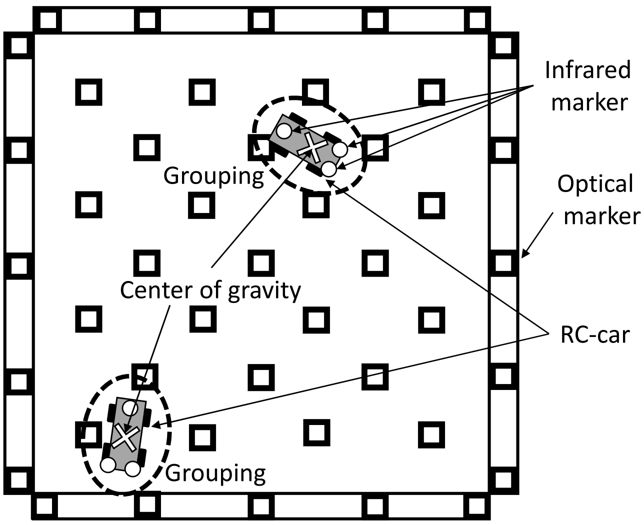


Fig. 5 Play filed with optical markers and RC-cars

The spatial accuracy between virtual objects and the real world is required for visual immersion in an AR environment. Simply overlaying virtual objects over the real world generates inconsistencies in the spatial relationship between the virtual objects and the real world, which is known as an occlusion problem [29]. The solutions proposed for the occlusion problem are mainly classified into depth-based and model-based approaches. The depth-based approach handles occlusions on the basis of depth information in the real scenes. A contour map was obtained to label each contour point of the real objects compared with the virtual objects in depth order [30]. A reliable depth map was obtained with a binocular stereo camera by extracting the object boundaries correctly [31]. The model-based approach handles occlusions on the basis of geometric models of the real objects. A3D model of the occluding object was obtained with silhouettes derived from segmentation by hand in the key-frames, and the 2D occluding boundary was determined by trimming the 3D region in the intermediate frames [32]. The 3D occluding boundary instead of the 3D model was obtained from two consecutive key-frames, and the 2D occluding boundary was recovered accurately by using region-based tracking [33].

These approaches required a complicated processing for the depth information or manual operation for the 3D model recovery. We take a simple approach to solve the occlusion problem. The bird's eye view derived from the infrared camera obtains the locations of RC-cars. Thus, the occlusions are achieved by placing transparent objects at the locations, assuming the size of the RC-cars.

IV. Implementation

The prototype system of the RC-car driving application was implemented on PCs running Windows 7, equipped with a 2.7 GHz Intel Core i7 CPU, 7 GB RAM, and nVIDIA GeForce GTX 560 Ti. A Sony HMZ-T2 was used as an HMD with 1280×720 resolution, 16:9 aspect ratio, and 45 deg. viewing angle. A Tamiya assembly kit model was used as an RC-car. Natural Point Opti Track V100 was used as an infrared camera with 640×480 resolution, 100 fps, and 45 deg. viewing angle. ARToolkit [34] was used as an image processing library for detecting markers and recognizing the predefined patterns. The multi-marker field was set on a flat floor with planer markers arranged at some intervals, as shown in Fig. 4 (a) and Fig. 5. Fig. 6 shows a snapshot of a player playing the RC-car driving.

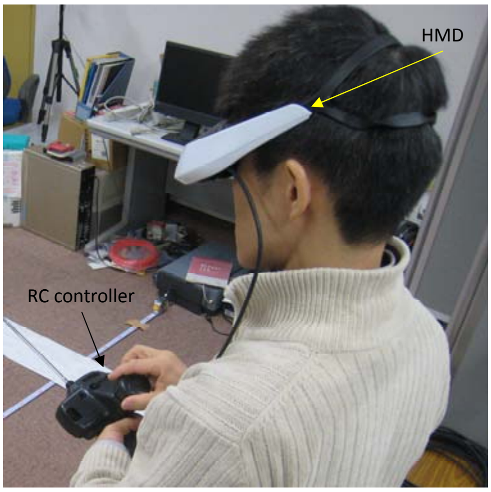


Fig. 6 A player wearing an HMD and holding an RC-car controller

Fig. 7 shows an example of the RC-cars prepared for the prototype system. The wireless visual camera was attached on the RC-car for capturing scene images in the point of view from the RC-car and estimating position and orientation of the camera in visual tracking. Three infrared markers are attached on the RC-car, forming an isosceles triangle. The infrared marker is made with retro-reflective materials. A temporal marker 20 cm × 20 cm in size was prepared to determine the origin of the coordinates, as depicted in Fig. 4 (a).

The infrared camera was placed on the ceiling of the room, 3 m above the floor, as shown in Fig. 8 (a). The viewing angle of 45 deg. covers an area on the floor 2.5 m × 2.5 m approximately in range. The RC-car controller used as a player interface is shown in Fig. 8 (b). The finger lever at the side controls an RC-car in driving forward and backward. The circular lever around the center controls an RC-car in turning left and right.

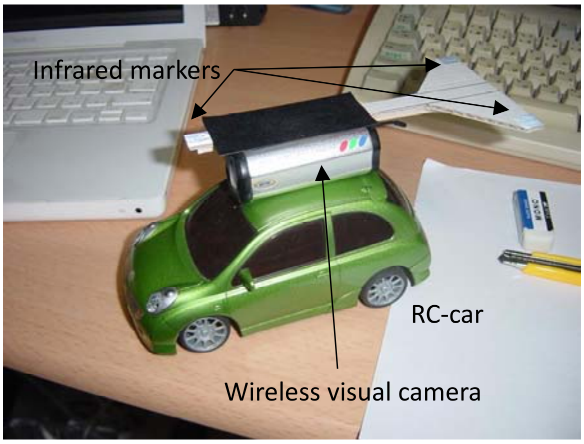
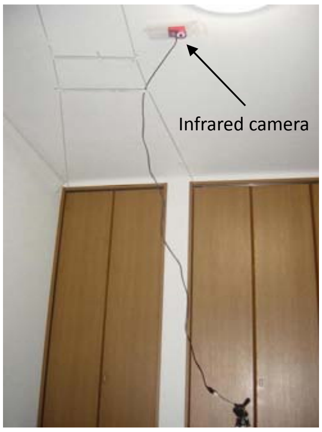


Fig. 7 RC-car with a wireless camera and infrared markers



(a) Infrared camera (b) RC-car controller

Fig. 8 Placements of infrared camera and RC-car controller

The current prototype system has some limitations. We did not play audio in the prototype system. No audio would largely reduce the immersive experiences of driving a car. Collisions between virtual and real objects are not also supported in the prototype system. The game play field is limited to a certain area in size due to the height of the room, which comes from the viewing angle of the infrared camera placed on the ceiling. The optical markers must be placed precisely at certain intervals on the flat floor to obtain the accurate transform data of the position and orientation. However, it is time-consuming to place them precisely.

We observed that two players played their RC-car driving together in the collaborative AR environment. Each player put on the HMD and held the RC-car controller. They needed to take seats or lean on a wall for their safety, since their views were covered with the HMDs. The players were told to obtain the longer driving distance safely within a time limit, which means that the RC-car must move within the game play field as fast as possible. The RC-cars easily bumped into the boundary due to players' driving mistakes. It seemed that speed preceded safety in their RC-car driving or they were not accustomed to the operation with the RC-car controllers.

V. CONCLUSION

We have described the RC-car driving system played by plural players in a collaborative AR environment. The RC-cars were controlled by players in the physical space, capturing scenes with a visual camera attached on the RC-car. The scenes superimposed with virtual objects were viewed with an HMD. The occlusions were properly processed in the AR environment, not rendering background virtual objects beyond the foreground real objects. We believe that the RC-car driving system provides players with immersive car-driving experiences. Future work includes implementing gaming functions and evaluating the RC-car driving as an AR application.

ACKNOWLEDGMENT

We thank Mr. Norio Takase for helping us technically implement the multiplayer RC-car driving system. Financial support for this research was partially provided by Grants-in-Aid for Scientific Research No. 23650517 and 90311167.

REFERENCES

[1] S. Feiner, "Augmented reality: a new way of seeing," *Scientific American*, vol.286, pp. 34–41, 2002.
[2] C. P. Barlett, C. A. Anderson, and E. L. Swing, "Video game effects—confirmed, suspected, and speculative: a review of the evidence," *Simulation & Gaming*, vol.40, pp.377–403, 2009.
[3] O. Oda, L. J. Lister, S. White, and S. Feiner, "Developing an augmented reality racing game," in *Proc. International Conference on Intelligent Technologies for Interactive Entertainment*, 8 pages, 2008.
[4] AD-DISPATCH, Augmented Reality Remote Control Car, http://www.ad-dispatch.com/our-work/toys/
[5] Sweet Tech Studio, Augmented Reality Remote Control Car, http://www.sweet-tech-studio.com/2009/01/diy-augmented-reality-remote-control.html
[6] B. H. Thomas, "A survey of visual, mixed, and augmented reality gaming," *ACM Computers in Entertainment*, vol.10, 33 pages, 2012.
[7] C. T. Tan, and D. Soh, "Augmented Reality Games: A Review," in *Proc. Asian Simulation and AI in Games Conference, GAMEON-ASIA, EUROSIS*, pp.212–218, 2011.
[8] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol.21, pp.34–47, 2001.
[9] T. Ohshima, K. Satoh, H. Yamamoto, H. Tamura, "AR2 Hockey system: A collaborative mixed reality system," *Transactions of the Virtual Reality Society of Japan*, vol.3, pp.55–60, 1998.
[10] T. Starner, B. Leibe, B. Singletary, J. Pair, "MIND-WARPING: towards creating a compelling collaborative augmented reality game," in *Proc. International Conference on Intelligent User Interfaces*, pp.256–259, 2000.
[11] B. Thomas, B. Close, J. Donoghue, J. Squires, P. D. Bondi, M. Morris, W. Piekarski, "ARQuake: An outdoor/indoor augmented reality first person application," in *Proc. International Symposium on Wearable Computers*, pp.139–146, 2000.
[12] A. D. Cheok, X. Yang, Z. Z. Ying, M. Billinghurst, H. Kato, "Touch-Space: mixed reality game space based on ubiquitous, tangible, and social computing," *Personal and Ubiquitous Computing*, vol.6, pp.430–442, 2002.
[13] K. Phillips, and W. Piekarski, "Possession techniques for interaction in real-time strategy augmented reality games," in*Proc. International Conference on Advances in Computer Entertainment Technology*, 10 pages, 2005.
[14] A. Hill, J. Schiefer, J. Wilson, B. Davidson, M. Gandy, and B. MacIntyre, "Virtual transparency: Introducing parallax view into video see-through AR," in *Proc. International Symposium on Mixed and Augmented Reality (ISMAR)*, pp.239–240, 2011.
[15] D. Baricevic, C. Lee, M. Turk, T. Hollerer, and D. A. Bowman, "A hand-held AR magic lens with user-perspective rendering," in *Proc. International Symposium on Mixed and Augmented Reality (ISMAR)*, pp.197–206, 2012.
[16] D. Wagner, T. Pintaric, F. Ledermann, D. Schmalstieg, "Towards massively multi-user augmented reality on handheld devices," in *Proc. International Conference on Pervasive Computing*, pp.208–219, 2005.
[17] A. Henrysson, M. Billinghurst, and M. Ollila, "AR tennis," in *ACM SIGGRAPH Emerging technologies*, 1 page, 2006.
[18] O. Oda, and S. Feiner, "Interference avoidance in multi-user hand-held augmented reality," in *Proc. IEEE International Symposium on Mixed and Augmented Reality*, pp.13–22, 2009.
[19] int13, AR Defender, http://www.ardefender.com/
[20] Sony Computer Entertainment Europe, Invizimals, http://www.invizimals.com/
[21] O. Bimber, and R. Rasker, *Spatial augmented reality: merging real and virtual worlds*, A K Peters, 2005.
[22] B. Piper, C. Ratti, and H. Ishii, "Illuminating clay: a 3-D tangible interface for landscape analysis," in *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp. 355–362, 2002.
[23] B. Stevens, J. Jerrams-Smith, D. Heathcote, and D. Callear, "Putting the virtual into reality: assessing object-presence with projection-augmented models," *Presence: Teleoperators and Virtual Environments*, vol.11, pp.79–92, 2002.
[24] R. Colvin, T. Hung, D. Jimison, B. Johnson, E. Myers, T. Blaine, "A dice game in third person augmented reality," in *Proc. International Augmented Reality Toolkit Workshop*, pp.3–4, 2003.
[25] M. Kojima, M. Sugimoto, A. Nakamura, M. Tomiya, M. Inami, H. Nii, "Augmented Coliseum: an augmented game environment with small vehicles," in *Proc. IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, pp.3–8, 2006.
[26] E. Tse, S. Greenberg, C. Shen, and C. Forlines, "Multimodal multiplayer tabletop gaming," *ACM Computers in Entertainment*, vol.5, 12 pages, 2007.
[27] J. Leitner, M. Haller, K. Yun, W. Woo, M. Sugimoto, and M. Inami, "IncreTable, a mixed reality tabletop game experience," in *Proc. International Conference on Advances in Computer Entertainment Technology*, pp.9–16, 2008.
[28] M. Mine, D. Rose, B. Yang, J. van Baar, and A. Grundhöfer, "Projection-based augmented reality in Disney theme parks," *Computer*, vol.45, pp.32–40, 2012.
[29] M. M. Wloka, and B. G. Anderson, "Resolving occlusion in augmented reality," in *Proc. ACM symposium on Interactive 3D graphics (I3D '95)*, pp.5–12, 1995.
[30] M. -O. Berger, "Resolving occlusion in augmented reality: a contour based approach without 3D reconstruction," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.91–96, 1997.
[31] J. Schmidt, H. Niemann, and S. Vogt, "Dense disparity maps in real-time with an application to augmented reality," in *Proc. IEEE Workshop on Applications of Computer Vision (WACV 2002)*, pp.225–230, 2002.
[32] K. C. Ong, H. C. Teh, and T. S. Tan, "Resolving occlusion in image sequence made easy," *The Visual Computer*, vol.14, pp.153–165, 1998.
[33] V. Lepetit, M. -O. Berger, "A semi-automatic method for resolving occlusion in augmented reality," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol.2, pp.225–230, 2000.
[34] HIT Lab, University of Washington: AR Tool kit. http://www.hitl.washington.edu/artoolkit/