Flying Eyes: Free-Space Content Creation Using Autonomous Aerial Vehicles

Keita Higuchi
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku Tokyo
113-0033 Japan
higuchi1604827@gmail.com

Jun Rekimoto
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku Tokyo
113-0033 Japan
rekimoto@acm.org

Interaction Laboratory
Sony Computer Science
Laboratories, Inc.
3-14-13 Higashigotanda,
Shinagawa-ku Tokyo
141-0022 Japan

Yoshio Ishiguro
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku Tokyo
113-0033 Japan
ishiy@acm.org

Japan Society for the Promotion
of Science
6 Ichiban-cho, Chiyoda-ku Tokyo
102-8471 Japan

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Abstract
Highly effective 3D-camerawork techniques that do not have physical limitations have been developed for creating three-dimensional (3D) computer games. Recent techniques used for real-world visual content creation, such as those used for sports broadcasting and motion pictures, also incorporate cameras moving in 3D physical space to provide viewers with a more engaging experience. For such purpose, wired cameras or mechanically controlled cameras are used, but they require huge and expensive infrastructure, and their freedom of motion is limited. To realize more flexible free-space camerawork at reasonable cost, we propose a system called "Flying Eyes" based on autonomous aerial vehicles. Flying Eyes tracks target humans based on vision processing, and computes camera paths by controlling the camera position and orientation.

Keywords
Video content, unmanned aerial vehicle, autonomous camerawork

ACM Classification Keywords
H .5.1 Multimedia Information Systems

General Terms

Experimentation

Introduction

Video production of three-dimensional computer graphics (3DCG) facilitates visual expression, without placing physical constraints on the camera position [1] (figure 1). Owing to the constraint-free camerawork, audiences of 3DCG videos can experience a highly realistic sensation. In computer games camerawork such as third-person capturing and first-person view are considered as critical factors for operability. These types of camerawork determine the camera position based on motion prediction and a player's avatar actions. Therefore, several camerawork algorithms have been presented in previous researches [2, 3]. If 3DCG-like camerawork algorithms were used in the real world, it would be possible to capture motion pictures such that viewers can experience realistic feelings.

Recent developments in motion picture capturing techniques have facilitated high-quality sports broadcasts. Wired cameras such as Spydercam [4] provide viewers with a long-shot perspective by tracking a soccer player; this method was initiated at the FIFA World Cup 2010 in South Africa. Kanade et al. introduced a method to archieve synchronous controll of 30 cameras for capturing an American football game. This method facilitated pseudo-free-motion camerawork in the stadium [5]. Inamoto et al. developed an open-ended space viewer for a soccer field by carry out image completion using video frames from four high-precision cameras [6].

However, these types of camerawork have several problems. Generally, they require large-scale infrastructure such as multiple cameras and computers for image processing. Crane cameras or wired cameras are constrained by prearranged operation ranges.

Kanade et al. created a system using multiple cameras where free-space vision is simulated by the image-completion method. However, this system produces a low-resolution video image and causes a dead zone within the camera view [5]. Moreover, these capturing techniques lack scalability and hence require advance preparation.

On the other hand, unmanned aerial vehicles (UAVs) possess the potential of capturing these types of video content, because UAVs can freely move in space. Recently, researchers and experts have captured airborne imagery using UAVs [7]. The An AR.Drone [8] and the Mikrokopter [9] are examples of reasonably priced computer-controlled small quad-helicopters.

figure 1. Camerawork of 3D computer graphics can freely move in virtual space. This picture camera position is configured only in 3DCG, because in real space, camerawork is constrained by physical limitations [1].

figure 2. Result of our Flying Eyes free-space video capturing system: The system includes an autonomous aerial vehicle mounted video capturing station, and a command-and-control PC for camera path calculation. It enables extensive flexible camerawork without spatial restrictions. Therefore, audiences experience a highly realistic sensation from the motion picture images taken by this system.

In this paper, we propose "Flying Eyes" vision: it involves free-space cameras that realize a third-person objective view. The goal of our work is to capture Flying Eyes vision in the real world using an autonomous aerial vehicle (AAV) platform (figure 2). We develop a prototype system and two basic camerawork techniques for capturing Flying Eyes vision. Finally, we discuss sports applications of our system.

Flying Eyes

We defined "Flying Eyes" vision as follows:

- The free-space camera is able to track and capture a subject person.
- The camera is not spatially constrained.
- The camera captures landscape around the subject person.

Sports broadcasting camera paths are not fixed and are context aware. In soccer game broadcasting, a camera tracks ball possession and the dynamic change of offense and defense. In a winter snow-skiing scene, the camera should capture players and avoid barriers such as trees and poles automatically.

We propose a method for enabling capturing Flying Eyes vision in the real world. We have developed its video capturing platform with an AAV. To enable capturing video with 3DCG-like camerawork, the system has to sense and recognize contextual information of the environment around an AAV. The system determines the camera path based on a subject person and barrier position.

System configuration

We developed a prototype system that realizes two types of camera path technique. This system includes an autonomous aerial vehicle and a command-and-control PC for calculating camera path (figure 3). We use AR.Drone from Parrot Inc. as an AAV [8]. The command-and-control PC calculates camera paths

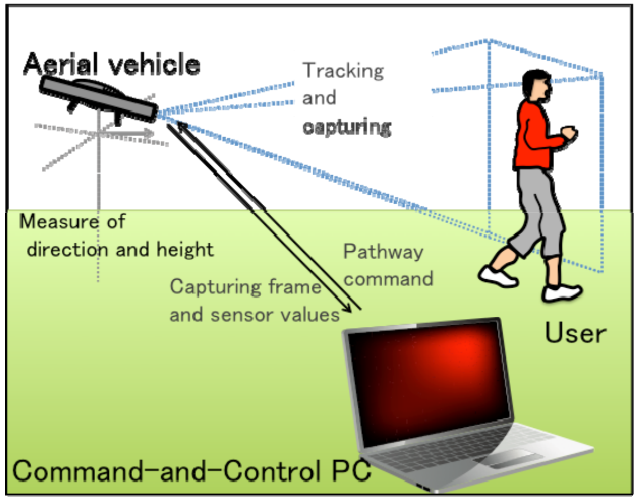

figure 3. System configuration: An autonomous aerial vehicle tracks a subject person. Calculated camera paths, as well as other sensor values such as AAV's height, its rotation angles and its video stream, are transmitted to a command-and-control PC. The base station then calculates the capturing camera path and sends it to the aerial vehicle.

based on received sensor values and video stream from the AAV. Therefore, it tracks a subject person detected on video stream and estimates the distance between the subject person and an AAV. In addition it handles AR.Drone's navigation parameters such as yaw, pitch, and roll angles.

Autonomous aerial vehicle

AR.Drone, a small UAV with 4-blade propellers, can be easily controlled through a wireless network. It sends onboard sensor information including height sensor values provided by a down-facing ultrasonic sensor and camera view to the command-and-control PC.

Person tracking

In order to perform free-angle video capture, the system recognizes and tracks a subject person. In this research, we developed a color extraction and particle filter algorithm for human tracking. The system requires the subject person wearing a distinctive color suit. It determines the person subject area by recognizing only the specific color. The captured image data contains noise because of variable color. A particle filter algorithm enables noise-robust tracking by observing time-series data; further, it estimates the current and subsequent state of the tracked object [10, 11] (figure 4).

We configured our system to initialize 1000 particles in a QVGA-size (320 × 240 pixel) image obtained by the onboard camera of the AAV. The system samples each particle in dispersion of 32 pixels. Each particle's gravity is subject's area window size bounds for 10×10 pixels.

Estimating distance between the person and the AAV

The system has to estimate the distance between a subject person and the AAV, because tracking may fail if the distance is great. Estimating distance *d* requires the following parameters: AAV's height *h1*, AAV's tilt θ, relative angle φ between person and UAV gained video frame (field angle: 93°), and pre-defined person height *h2* (configured to 120±15 cm because the system tracks the upper body of the person for the particle filter). It is calculated according to Equation (1) (figure 5).

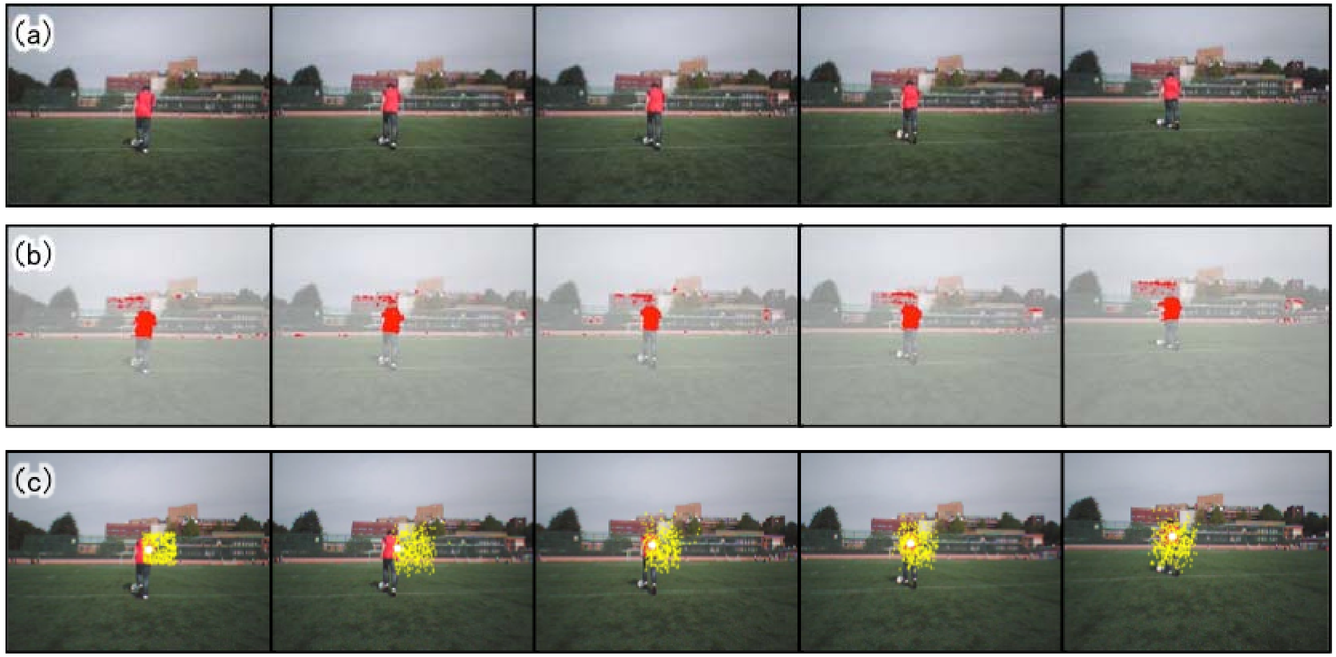

figure 4. Vision-based subject person tracking: the system tracks a person based on an AAV-mounted camera and a particle filter. The system estimates subject person's position using color extraction. The particle filter enables noise-robust person tracking. In the soccer game scenario: (a) The subject person dribbles towards the goal. (b) Our system extracts the pre-defined color. Noise is shown in extracted frames. (c) Our system enables person tracking even in noisy environments.

$$d = \frac{|h_1 - h_2|}{sin(\theta + \phi)} sin(\frac{\pi}{2} - (\theta + \phi)) \qquad (1)$$

This method does not measure an exact distance. However, the estimated value is sufficient enough for determining the AAV's navigation parameters.

Camera path algorithms

The camerawork of video contents is configured from several camera paths such as forwarding and moving circle. We defined two basic camera paths "TRACKING" and "CIRCLING" (figure 6).

- TRACKING: The camera path follows and captures a subject person from behind.
- CIRCLING: The camera path captures subject person in circular mode.

The system captures Flying Eyes vision that combines two camera paths mentioned above.

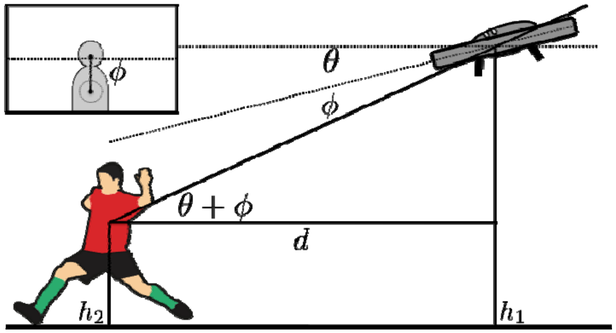

figure 5. Estimated distance: Our system estimates distance between subject person and aerial vehicle with triangulation. *h1*: aerial vehicle height between ground and itself. *h2*: pre-defined person height. θ: aerial vehicle tilt angle. φ: relative angle between person and aerial vehicle.

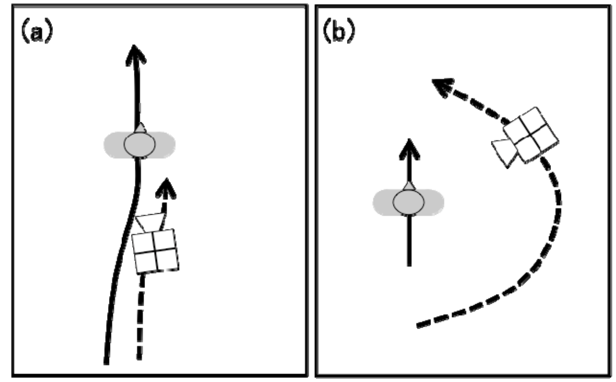

figure 6. We implemented two basic camera paths "TRACKING" and "CIRCLING." (a) TRACKING follows to track and capture subject person from behind. (b) CIRCLING follows to capturing roll on subject person circumference.

TRACKING

In order to capture The TRACKING camera path, an AAV has to follow the subject person. Therefore, it seeks a subject person and avoids losing the person inside video frame. We defined non-drop off condition Equation (2) (P_x subject person position inside video frame *Width* video frame width: *threshold* for subject person be front of the AAV).

$$|P_x - \frac{Width}{2}| < threshold \quad (2)$$

The system operates according to following algorithm:

- The AAV goes straight ahead, when the position of the subject person inside the video frame fulfills Equation (2).
- The AAV stops and rotates according to the roll angle to find the subject person, when the position of person inside the video frame does not fulfill Equation (2).
- The AAV accelerates forward, when the estimated distance *d* is above the threshold level.

CIRCLING

In case of CIRCLING, the UAV captures a subject person in a circular flying pattern. Circling is realized by rolling yaw angle and roll angle. The system operates according to following algorithm:

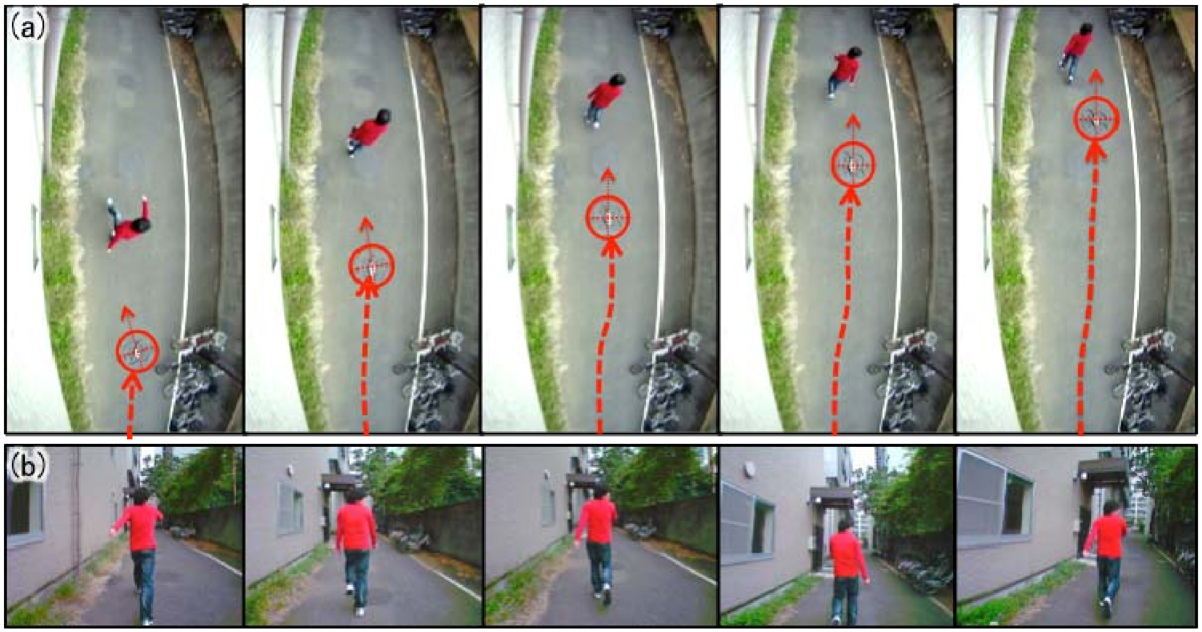

figure 7. We experimented capturing with TRACKING camerawork. (a) Upper camera view: AAV tracked a subject person linearly. (b) Aerial vehicle inside camera view: The system did not lose the subject person.

- The AAV rolls about the center of the person's location, when the position of person inside the video frame fulfills Equation (2).

- The AAV rolls according to the angle to point at the subject person, when the position of person inside the video frame does not fulfill Equation (2).

Results

We experimented with an AAV capturing a subject in the TRACKING mode (figure 7). The system was able to follow the subject person without losing from the video frame in spite of the noisy frame. However, there was a challenge with AAV's speed control because the method of estimating distance was inaccurate.

We also experimented with the AAV capturing a subject person using the CIRCLING camera path (figure 8). However, the AAV did not fully follow the subject person, because the system does not precisely estimate the distance between the subject person and the AAV.

Camera users manually shake a handheld camera by running or walking up or down stairs for capturing.

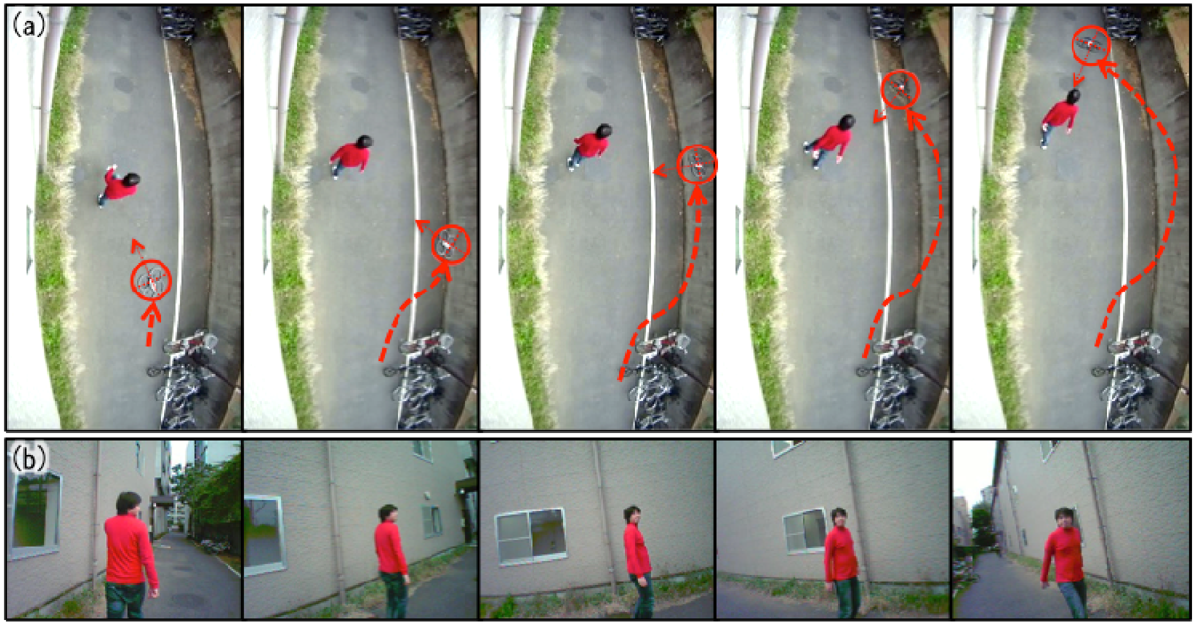

figure 8. Experiment involves capturing CIRCLING camerawork. (a) Upper camera view: an AAV rolls around the subject person. (b) AAV's on-board inside camera view. The system did not lose the subject person.

However, the AAV hovers over any types of terrain. Thus, the proposed system could capture stable images consistently (figure 9).

Discussion

The system enables capturing Flying Eyes vision using an AAV by TRACKING and CIRCLING camera paths. This section discusses sports applications with our system and future work.

Sports applications

The system enables novel sports broadcasts, because it can capture video without spatial restriction in the real world. It does not requires large-scale infrastructure or advance preparation. It can capture sports broadcasts from indoor fields and outdoor stadiums.

Camerawork editing tool

In this work we developed an autonomous aerial vehicle that can capture a subject using two camera paths. A user can create video contents by selecting combined camera paths. In the alternative approach,

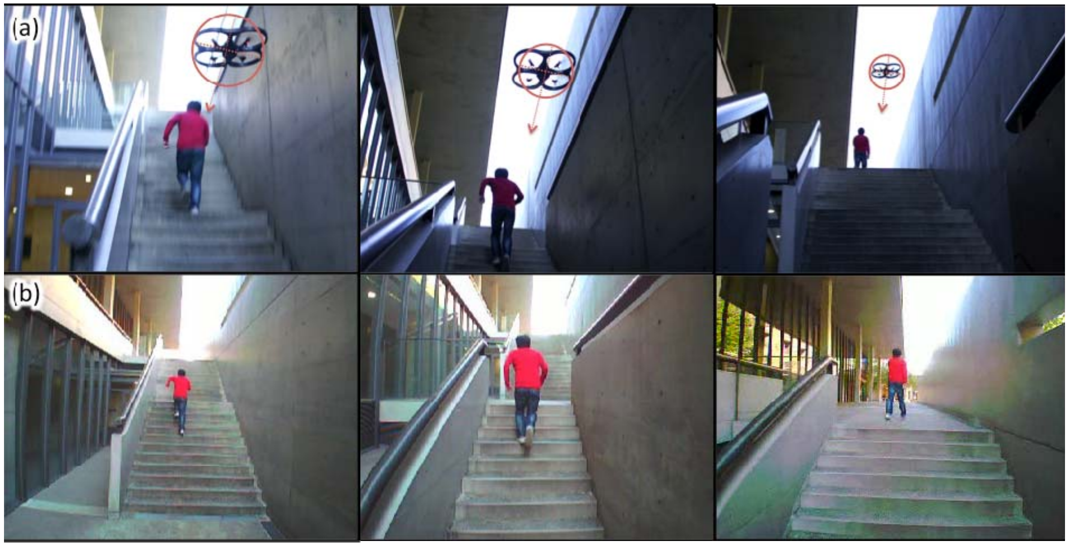

figure 9. We experimented with camerawork on a stair. (a) First picture with photographer chasing subject shows camera shake; a stationary handheld camera captured a subject person with height difference. (b) Camera on board aerial vehicle in TRACKING mode shows no camera shake.

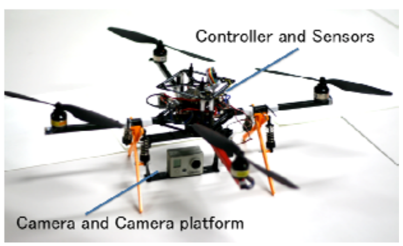

figure 10. Second-generation aerial vehicle: Mikrokopter of open source hardware can carry a payload up to 1 kg and can fly as fast as 50 km/h. We will develop a new Flying Eyes vision capturing system with Mikrokopter.

motion picture capturing requires an advanced technique.

We plan to develop creation tools for Flying Eyes video contents for any amateur or professional camera users. It comprises an autonomous aerial vehicle platform and a camerawork edit tool. By using this tool users can choose transition mixing and matching the camera paths to create their own camerawork interactively.

System Performance Requirement
For capturing sports scene with an AAV, the vehicle should carry cameras and other sensing payloads and should fly faster than sports players, whose running speed can exceed 40 km/h. Furthermore, it should avoid collision with a subject person by means of a mounted collision avoidance unit (CAU). However, our current prototype, based on AR.Drone, can fly at 18 km/h, and can carry 200 g payloads.

We thus consider a better platform that can move faster and can carry a heavier load. Mikrokopter of open source hardware can carry up to 1 kg and can fly as fast as 50 km/h. It can view sideways to the direction of its movement high-performance camera on a triaxial movable camera platform. It can mount CAU that includes several proximity sensors. We will develop a new Flying Eyes vision capturing system with Mikrokopter (Ffigure 10).

Human Tracking
The system provides tracking of a person using a color extraction and particle filter algorithm. However, these methods have several problems, e.g., color extraction is non-robust — changing with lighting environment — and cannot concurrently recognize multiple persons in the area of the video stream.

Ess et al. have developed a mobile multi-person tracking platform with image processing [12]. We will realize robust human tracking that combines several tracking methods. If players have installed some wearable devices such as an IR beacon, GPS, or Bluetooth module, the system can recognize each player's position.

Conclusion
In this paper, we proposed a system that captures real-world scenes by using flexible free-space camera motion. We developed a prototype system consisting of

an autonomous aerial vehicle and a command-and-control PC. The system provided two basic camera paths for generating highly realistic motion pictures. We discussed some sports applications of the proposed system, which required the use of a UAV performance. We are currently developing the second generation of the proposed capturing system.

Acknowledgement

This work has been partially supported by the Exploratory IT Human Resources Project 2010, IPA. This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for JSPS Fellows, 21-8596, 2009.

References

[1] Electronic Arts: FIFA Soccer 10, (2010).

[2] Hawkins, B.: Creating an Event-Driven Cinematic Camera, Game Developer Magazine, Vol.9, No.10 & 11, (2003).

[3] Hawkins, B.: Real-Time Cinematography for Games, Charles River Media, (2005).

[4] Spydercam: http://www.spydercamusa.com/.

[5] Kitahara, I., Saito, H., Akimichi, S., Ono, T., Ohta, Y., and Kanade, T.: Large-scale Virtualized Reality, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Technical Sketches, (2001).

[6] Inamoto, N. and Saito, H: Virtual Viewpoint Replay for a Soccer Match by View Interpolation from Multiple Cameras, IEEE Transactions on Multimedia, Vol.9, PP. 1155-1166, (2007).

[7] Quigley, M., Goodrich, M.A., Griffiths, S., Eldredge, A. and Beard, R.W.: Target Acquisition, Localization, and Surveillance Using a Fixed-Wing Mini-UAV and Gimbaled Camera, ICRA, pp. 2600-2605, (2005).

[8] ARDrone: http://ardrone.parrot.com/.

[9] MikroKopter: http://www.mikrokopter.de/.

[10] Isard, M. and Blake, A.: Contour Tracking by Ttochastic Propagation of Conditional Density, Lecture Notes in Computer Science, Vol.1064, pp. 345-356, (1996).

[11] Arulampalam. M.S., Maskell. S., Gordon, N.: A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian tracking, IEEE Transactions on Signal Processing, Vol.50, pp. 174-188, (2002).

[12] Ess, S., Leibe, B., Schindler, K. and Van, G.L.: A Mobile Vision System for Robust Multi-person Tracking, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1-8, 23-28, (2008).