Ubiquitous and Wearable Vision Systems

Takashi Matsuyama
Graduate School of Informatics
Kyoto University
Sakyo, Kyoto, 606-8501, Japan

Capturing multi-view images by a group of spatially distributed cameras is one of the most useful and practical methods to extend utilities and overcome limitations of a standard pinhole camera: limited size of visual field and degeneration of 3D information. This paper gives an overview of our research activities on multi-view image analysis. First we address a ubiquitous vision system, where a group of network-connected active cameras are embedded in the real world to realize 1) wide-area dynamic 3D scene understanding and 2) versatile 3D scene visualization. To demonstrate utilities of the system, we developed a cooperative distributed active object tracking system and a 3D video generation system. The latter half of the paper discusses a wearable vision system, where multiple active cameras are placed nearby around human eyes to share the visual field. To demonstrate utilities of the system, we developed systems for 1) real time accurate estimation of 3D human gaze point, 2) 3D digitization of a hand-held object, and 3) estimation of 3D human motion trajectory.

1 Introduction

Capturing multi-view images by a group of spatially distributed cameras is one of the most useful and practical methods to extend utilities and overcome limitations of a standard pinhole camera: limited size of visual field and degeneration of 3D information.

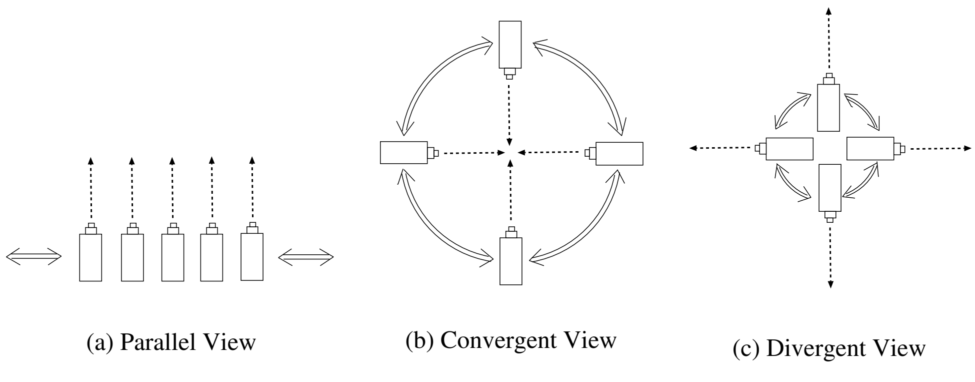
Figure 1 illustrates three typical types of multi-view camera arrangements:

(1) Parallel View : for wide area stereo vision (e.g. capturing 100m race at the Olympic game)

(2) Convergent View : for detailed 3D human action observation (e.g. digital archive of traditional dances)

(3) Divergent View : for omnidirectional panoramic scene observation

This paper gives an overview of our research activities on multi-view image analysis. Following a brief introduction of our specialized active camera, the paper addresses a convergent view multi-camera system, where a group of network-connected active cameras are embedded in the real world to realize 1) wide-area dynamic 3D scene understanding and 2) versatile 3D scene visualization[1]. We may call such system a *ubiquitous vision system*. Based on this scheme, we developed a cooperative distributed active



(a) Parallel View

(b) Convergent View

(c) Divergent View

Figure 1: Types of multi-view camera arrangements.

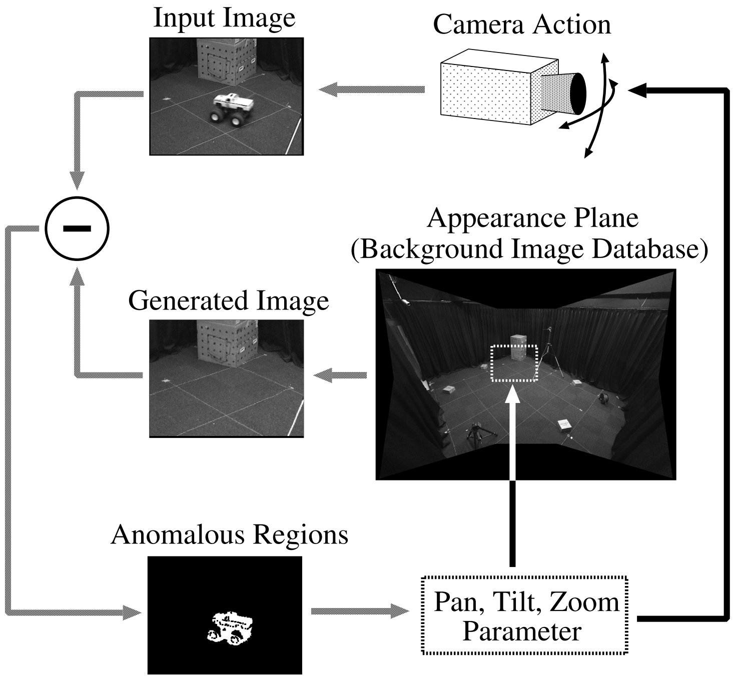


Figure 2: Active background subtraction with a fixed-viewpoint pan-tilt-zoom (FV-PTZ) camera.

object tracking system[2] and a 3D video[3] generation system[4]. Experimental results demonstrated utilities of the ubiquitous vision system.

The latter half of the paper discusses a *wearable active vision system*, where multiple active cameras are placed nearby around human eyes to share the visual field. This system employs either convergent or divergent view observations depending on required tasks: the former for 1) real time accurate estimation of 3D human gaze point and 2) 3D digitization of a hand-held object, and the latter for 3) estimation of 3D human motion trajectory[5].

Since the space is limited, the paper gives just a summary of our research attainments obtained so far. As for technical details, please refer to the references.

2 Fixed-Viewpoint Pan-Tilt-Zoom Camera for Wide Area Scene Observation and Active Object Tracking

First of all, to expand the visual field of a camera is an important issue in developing wide area scene observation and real time moving object tracking.

In [6], we developed a *fixed-viewpoint pan-tilt-zoom (FV-PTZ, in short) camera*: as its projection center stays fixed irrespectively of any camera rotations and zoomings, we can use it as a pinhole camera with a very wide visual field. All the systems described in this paper employ an off-the-shelf active video camera SONY EVI-G20 since it can be well modeled as an FV-PTZ camera.

With an FV-PTZ camera, we can easily realize an active target tracking system as well as generate an wide panoramic image by mosaicing images taken with different pan-tilt-zoom parameters. Figure 2 illustrates the basic scheme of the active background subtraction for object tracking[1]:

1. Generate the *APpearance Plane* (APP) image: a wide panoramic image of the background scene.
2. Extract a window image from the APP image according to the current pan-tilt-zoom parameters and regard it as the current background image; with the FV-PTZ camera, there exists the direct mapping between the position in the APP image and pan-tilt-zoom parameters of the camera.

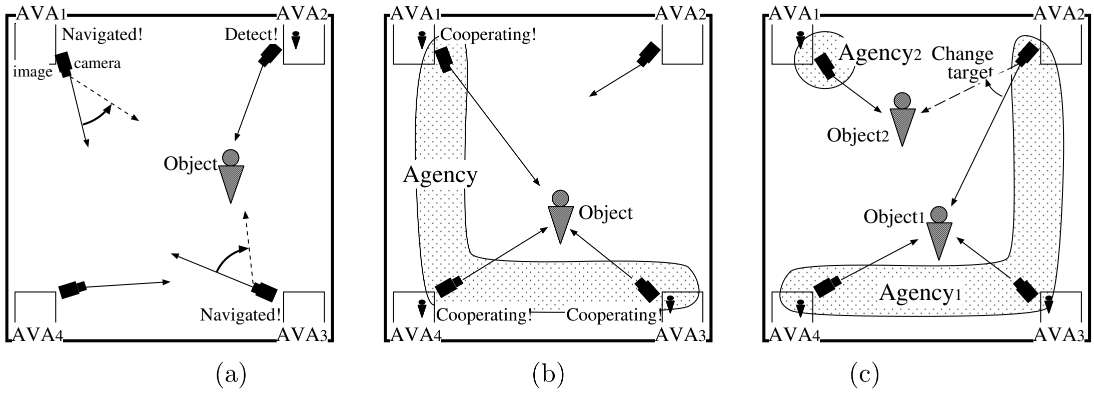


Figure 3: Basic scheme for cooperative tracking: (a) Gaze navigation, (b) Cooperative gazing, (c) Adaptive target switching.

3. Compute difference between the current background image and an observed image.
4. If anomalous regions are detected in the difference image, select one and control the camera parameters to track the selected target.

Based on this scheme, we developed a real-time active moving object tracking system, where a robust background subtraction method[7] and a sophisticated real-time camera control method[8] were employed.

3 Tracking and 3D Digitization of Objects by a Ubiquitous Vision System

3.1 Cooperative multi-target tracking by communicating active vision agents

Since the observation from a single viewpoint cannot give us explicit 3D scene information or avoid occlusion, we developed a multi-viewpoint camera system (i.e. convergent view multi-camera system), where a group of network connected FV-PTZ cameras are distributed in a wide area scene. Each camera is controlled by its corresponding PC, which exchanges observed data with each other to track objects and measure their 3D information. We call such network-connected PC with an active camera *Active Vision Agent* (AVA, in short).

Assuming that the cameras are calibrated and densely distributed over the scene so that their visual fields are well overlapping with each other, we developed a cooperative multi-target tracking system by a group of communicating AVAs[2].

Figure 3 illustrates the basic tasks conducted by the cooperations among AVAs:

1. Initially, each AVA independently searches for a target that comes into its observable area.
2. If an AVA detects a target, it navigates the gazes of the other AVAs towards that target (Figure 3 (a)).
3. A group of AVAs which gaze at the same target form what we call an *Agency* and keep measuring the 3D information of the target from multi-view images (Figure 3 (b)). Note that while some AVAs are tracking an object, others are still searching for new objects.
4. Depending on target locations in the scene, each AVA dynamically changes its target (Figure 3 (c)).

To verify the effectiveness of the proposed system, we conducted experiments of multiple human tracking in a room (about $5m \times 5m$). The system consists of ten AVAs. Each AVA is implemented on a network-connected PC (Pentium III 600MHz × 2) with an FV-PTZ camera (SONY EVI-G20).

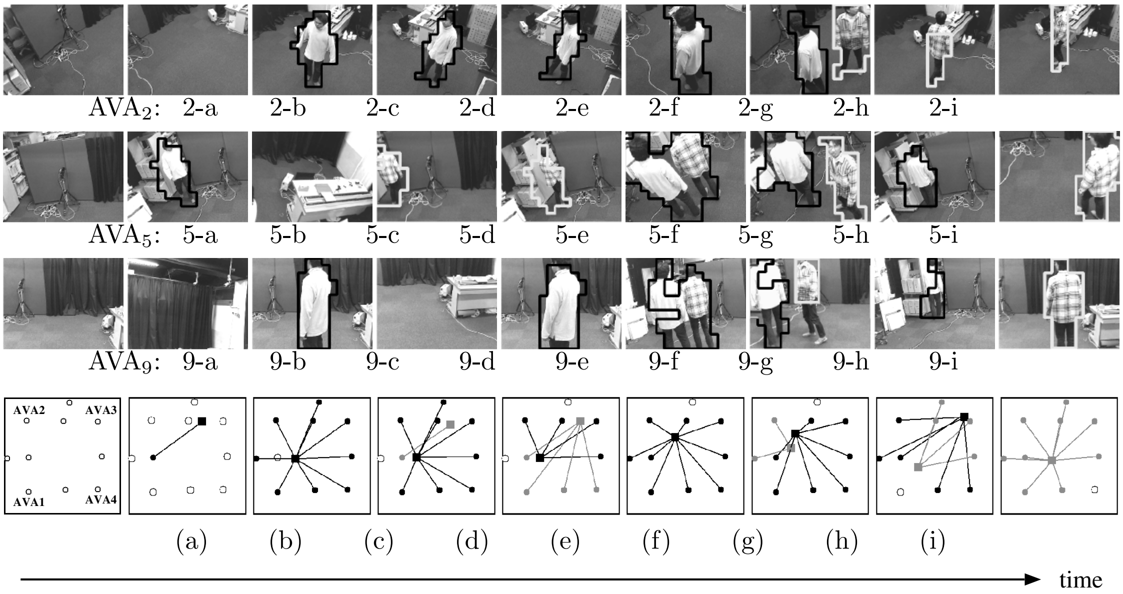

AVA$_2$: 2-a 2-b 2-c 2-d 2-e 2-f 2-g 2-h 2-i
AVA$_5$: 5-a 5-b 5-c 5-d 5-e 5-f 5-g 5-h 5-i
AVA$_9$: 9-a 9-b 9-c 9-d 9-e 9-f 9-g 9-h 9-i
AVA2
AVA3
AVA1
AVA4
(a) (b) (c) (d) (e) (f) (g) (h) (i)
time

Figure 4: Experimental results.

In the experiment shown in Figure 4, the system tracked two people. Target$_1$ first came into the scene and after a while, target$_2$ came into the scene. Both targets then moved freely. The upper three rows in Figure 4 show the partial image sequences observed by AVA$_2$, AVA$_5$ and AVA$_9$, respectively. The images on the same column were taken at almost the same time. The regions enclosed by black and gray lines in the images show the detected regions corresponding to target$_1$ and target$_2$ respectively. Note that the image sequences in Figure 4 are not recorded ones but captured real-time according to the target motions.

The bottom row in Figure 4 shows the dynamic cooperation process conducted by ten AVAs. White circles mean that AVAs are in the target search mode, while black and gray circles indicate AVAs are tracking target$_1$ or target$_2$ forming agency$_1$ or agency$_2$, respectively. Black and gray squares indicate computed locations of target$_1$ and target$_2$ respectively, toward which gazing lines from AVAs are directed.

The system worked as follows. Note that (a)-(i) below denote the situations illustrated in Figure 4.

(a) : Initially, each AVA searched for an object independently.

(b) : AVA$_5$ first detected target$_1$, and after the gaze navigation of the other AVAs, agency$_1$ was formed.

(c) : After a while, all AVAs except AVA$_5$ were tracking target$_1$, since AVA$_5$ had switched its mode from tracking to searching, depending the target motion.

(d) : Then, AVA$_5$ detected a new target, target$_2$, and generated agency$_2$.

(e) : The agency restructuring protocol (i.e. adaptive target switching) balanced the numbers of member AVAs in agency$_1$ and agency$_2$. Note that AVA$_9$ and AVA$_{10}$ were searching for still new objects.

(f) : Since two targets came very close to each other and no AVA could distinguish them, the agency unification protocol merged agency$_2$ into agency$_1$.

(g) : When the targets got apart, agency$_1$ detected a 'new' target. Then, it activated the agency spawning protocol to generate agency$_2$ again for target$_2$.

(h) : Target$_1$ was going out of the scene.

(i) : After agency$_1$ was eliminated, all the AVAs except AVA$_4$ came to track target$_2$.

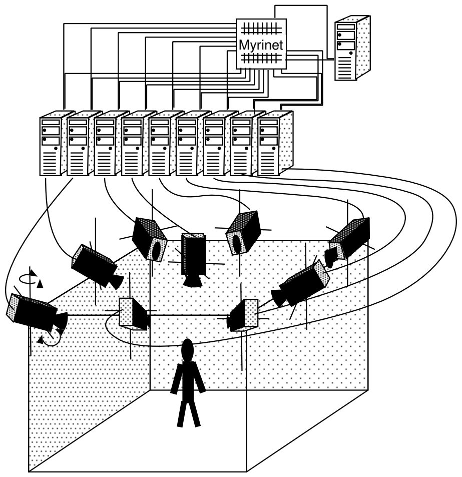


Figure 5: PC cluster for real-time active 3D object shape reconstruction.

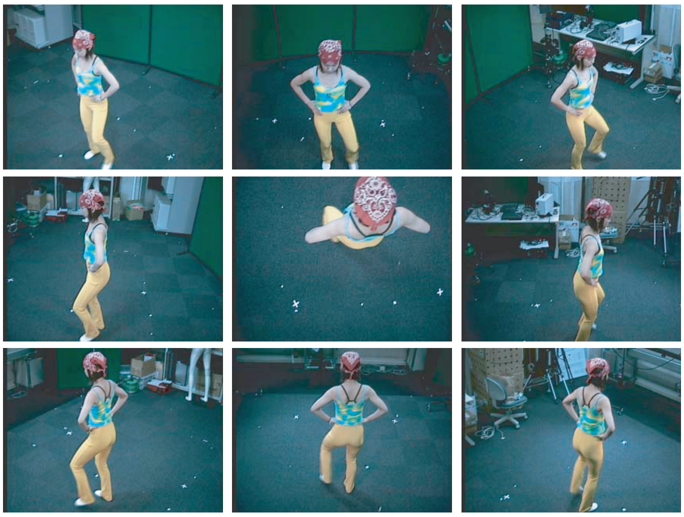


Figure 6: Captured multi-viewpoint images

These experiments proved that the cooperative target tracking by a group of multi-viewpoint active cameras is very effective to cope with unorganized dynamic object behaviors.

3.2 Generation of high fidelity 3D video

With the above mentioned tracking system, we can capture convergent multi-view video data of a moving object. To make full use of the captured video data, we developed a system for generating 3D video[4][9][10].

3D video[3] is NOT an artificial CG animation but a real 3D movie recording the full 3D shape, motion, and precise surface color & texture of real world objects. It enables us to observe real object behaviors from any viewpoints as well as to see pop-up 3D object images. Such new featured image medium will promote wide varieties of personal and social human activities: communication (e.g. 3D TV phone), entertainment (e.g. 3D game and 3D TV), education (e.g. 3D animal picture books), sports (e.g. sport performance analysis), medicine (e.g. 3D surgery monitoring), culture (e.g. 3D archive of traditional dances), and so on.

So far we developed

1. PC cluster system with distributed active cameras for real-time 3D shape reconstruction
2. Dynamic 3D mesh deformation method for obtaining accurate 3D object shape
3. Texture mapping algorithm for high fidelity visualization
4. User friendly 3D video editing system

3.2.1 System organization

Figure 5 illustrates the architecture of our real-time *active* 3D object shape reconstruction system. It consists of

- PC cluster: 30 node PCs (dual Pentium III 1GHz) are connected through Myrinet, an ultra high speed network (full duplex 1.28Gbps), which enables us to implement efficient parallel processing on the PC cluster.
- Distributed active video cameras: Among 30, 25 PCs have Fixed-Viewpoint Pan-Tilt (FV-PT) cameras, respectively, for active object tracking and imaging.

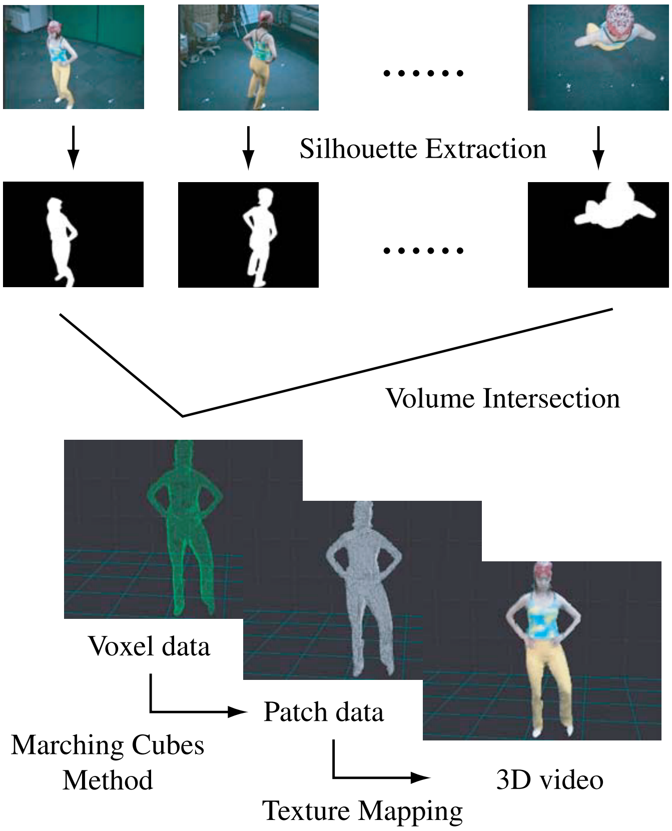


Figure 7: 3D video generation process

Figure 6 shows a snapshot of multi-view object video data captured by the system. Note that since the above mentioned PC cluster is our second generation system and has just become in operation, all test data used in this paper are those taken by the first generation system (16PCs and 12 cameras)[9]. We have verified that the second generation system can generate much more high quality 3D video in much less computation time. Experimental results by the second generation system will be published soon.

3.2.2 Processing scheme of 3D video generation

Figure 7 illustrates the basic process of generating a 3D video frame in our system:

(1) Synchronized Multi-View Image Acquisition: A set of multi-view object images are taken simultaneously (top row in Figure 7).

(2) Silhouette Extraction: Background subtraction is applied to each captured image to generate a set of multi-view object silhouettes (second top row in Figure 7).

(3) Silhouette Volume Intersection: Each silhouette is back-projected into the common 3D space to generate a visual cone encasing the 3D object. Then, such 3D cones are intersected with each other to generate the visual hull of the object (i.e. the voxel representation of the rough object shape) (third bottom in Figure 7).

To realize real-time 3D volume intersection,

- we first developed the plane-based volume intersection method, where the 3D voxel space is partitioned into a group of parallel planes and the cross-section of the 3D object volume on each plane is reconstructed.

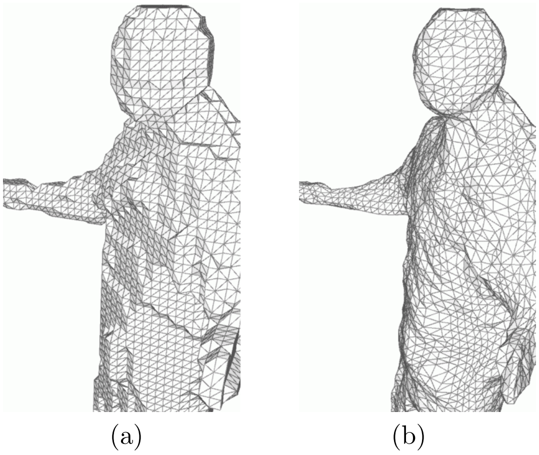


Figure 8: (a) surface mesh generated by the discrete Marching cube method and (b) surface mesh after the intra-frame mesh deformation

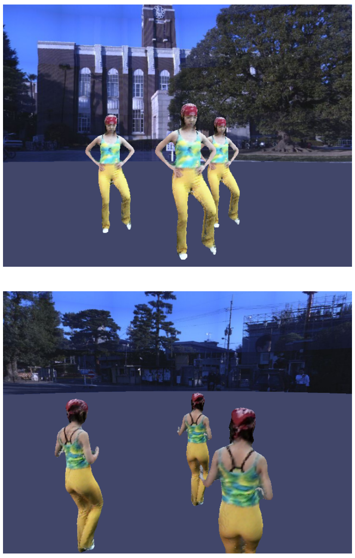


Figure 9: Visualized 3D video with an omnidirectional background

- Secondly, we devised the Plane-to-Plane Perspective Projection algorithm to realize efficient plane-to-plane projection computation.

- And thirdly, to realize real-time processing, we implemented parallel pipeline processing on a PC cluster system[9].

Experimental results showed that the proposed methods works efficiently and the PC cluster system can reconstruct 3D shape of a dancing human at about 12 volume per second in the voxel size of 2cm× 2cm× 2cm contained in a space of 2m × 2m × 2m. Note that this result is by the first generation PC cluster system.

(4) Surface Shape Computation: The discrete marching cubes method[11] is applied to convert the voxel representation to the surface mesh representation. Then the generated 3D mesh is deformed to obtain accurate 3D object shape (second bottom in Figure 7).

We developed a deformable 3D mesh model which reconstructs both the accurate 3D object shape and motion[10].

- For the initial video frame, we apply the intra-frame deformation method. Using the mesh generated from the voxel data as the initial shape, it deforms the mesh to satisfy the smoothness, silhouette, and photo-consistency constraints. The photo-consistency constraint enables us to recover concave parts of the object, which cannot be reconstructed by the volume intersection method. Figure 8 demonstrates the effectiveness of the mesh deformation.

- Using the result of the intra-frame deformation as the initial shape, we apply the inter-frame deformation method to a series of video frames. It additionally introduces the 3D motion flow and inertia constraints as well as a stiffness parameter into the mesh model to cope with non-rigid object motion.

Experimental results showed that the mesh deformation methods can significantly improve the accuracy of the reconstructed 3D shape. Moreover, we can obtain a temporal sequence of 3D meshes whose topological structures are kept constant; the complete vertex correspondence is established for all the 3D meshes. Their computation speeds, however, are far from real-time: for both the intra- and inter-frame deformations, it took about 5 minutes for 12000 vertices with 4 cameras and 10 minutes for 12000 vertices with 9 cameras by a PC (Xeon 1.7GHz). The parallel implementation to speed up the methods is one of our future works.

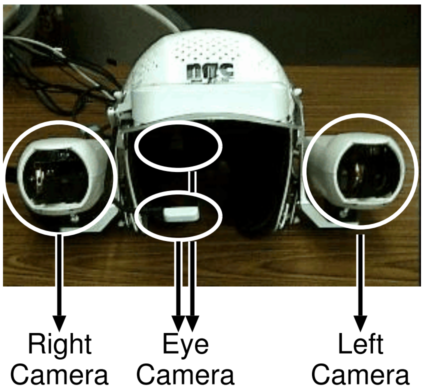


Figure 10: Active wearable vision sensor

(5) Texture Mapping: Color and texture on each patch are computed from the observed multi-view images (bottom in Figure 7).

We proposed the viewpoint dependent vertex-based texture mapping method to avoid jitters in rendered object images which are caused due to the limited accuracy of the reconstructed 3D object shape[4]. Experimental results showed that the proposed method can generate almost natural looking object images from arbitrary viewpoints. By compiling a temporal sequence of reconstructed 3D shape data and multi-view video into a temporal sequence of vertex lists, we can render arbitrary VGA views of 3D video sequence at video rate by an ordinary PC.

By repeating the above process for each video frame, we have a live 3D motion picture.

We also developed a 3D video editing system, with which we can copy and arrange a foreground 3D video object in front of a background omnidirectional video. Figure 9 illustrates a sample sequence of an edited 3D video.

4 Recognition of Human Activities and Surrounding Environments by an Active Wearable Vision System

The ubiquitous vision systems described so far observe people from outside to *objectively* analyze their behaviors. In this section, on the other hand, we introduce a wearable vision system (Figure 10)[12] to observe and analyze *subjective* view of a human; viewpoints of cameras are placed nearby around human eyes and moves with human behaviors. The system is equipped with a pair of FV-PTZ cameras and a gaze-direction detector (i.e. eye camera in Figure 10) to monitor human eye and head movements. Here we address methods to realized the following three functionalities: 1) 3D gaze point detection and focused target imaging, 2) 3D digitization of a hand-held object, and 3) 3D human motion trajectory measurement.

4.1 3D gaze point detection and focused target imaging

Here, we present a method to capture a close-up image of a human focusing object by actively controlling cameras based on 3D gaze point detection.

Since the gaze-direction detector equipped can only measure the human gaze direction, we control the FV-PTZ cameras to detect where he/she is looking at in the 3D space. Figure 11 illustrates a method to measure a 3D gaze point, which is defined by an intersection point between the gaze-direction line and an object surface. Assuming the cameras and the gaze-direction detector have been calibrated in advance, the viewing line is projected onto a pair of stereo images captured by the cameras. Then, we apply stereo matching along the pair of the projected lines.

Based on the measured 3D gaze point, we control pan, tilt, and zoom parameters of the active cameras to capture detailed target object images:

- If the human gaze direction is moving, the cameras are zoomed out to capture images of wide visual field. Pan and tilt are controlled to follow the gaze motion.
- If the human gaze direction is fixed, the camera is zoomed in to capture detailed images of the target object. The pan and tilt are controlled to converge toward the 3D gaze point.

We have implemented the above camera control strategy with a dynamic memory architecture[8], with which smooth reactive (without delay) camera control can be realized (Figure 12).

Figure 13 demonstrates the effectiveness of this active camera control. The upper and lower rows show pairs of stereo images captured without and with the gaze navigated camera control, respectively. A straight line and a dot in each image illustrate a projected view direction line and a human gazing point, respectively.

4.2 3D digitization of a hand-held object

Suppose we are in a department store and checking a coffee cup to buy. In such a situation, we manipulate an object to examine its shape, color, and surface painting from various viewpoints. With the wearable vision system, we developed a method to obtain a full 3D object image from video data captured during this human action.

From a viewpoint of human action analysis, first, we classify hand-object relationships into four classes based on the information human can acquire:
Shape Acquisition: Examine the overall object shape, where most of the object silhouette is visible (Figure 14(a)).
Surface Texture Acquisition: Examine surface painting, where some parts of the object silhouette are covered by hands (Figure 14(b)).
Haptic Texture Acquisition: Examine surface smoothness, where the object is wrapped by hands and most of the object surface is invisible (Figure 14(c)).
Total Appearance Acquisition: Examine the balance between shape and surface painting, where the object is turned around and most of object shape and surface painting can be observed (Figure 14(d)).

Then, from a viewpoint of computer vision, the problems to be studied are characterized as follows:

1. Assuming an object is rigid, the wearable vision system can capture convergent multi-view stereo images of the object; that is, representing the object manipulation process in the object centered coordinate system, a pair of stereo cameras are dynamically moved around the object to observe it from different viewpoints. Technically speaking, to determine 3D camera positions and motion in the object centered coordinate system, we have to compute 3D relative position between the cameras and object at each captured video frame as well as conduct stereo camera calibration.
2. While the object shape and position stay fixed in the object centered coordinate system, human hands change their shapes and positions dynamically to occlude the object. That is, we have to recover the 3D object shape from convergent multi-view stereo images where the shape and position of an occluding object changes depending on the viewpoint. We may call this problem *shape from multi-view stereo images with viewpoint dependent occlusion*, to which we cannot apply such conventional techniques as shape from silhouettes [13] or space carving [14]..

To solve the problem, we proposed a vacant space carving. That is, we first compute a vacant space, one that is not occupied by any object, from each viewpoint. Then, multiple vacant spaces from different viewpoints are integrated to generate a 3D object shape. The rationale behind this method is that a vacant space from one viewpoint can carve out a space occupied by hands at another viewpoint. This removes the viewpoint dependent occlusion.

The computational algorithm we developed is as follows:
1. Capture – A series of stereo images of a hand manipulated object are captured by the wearable vision sensor.
2. Feature Point Detection – From each frame of the dynamic stereo images, feature points on the

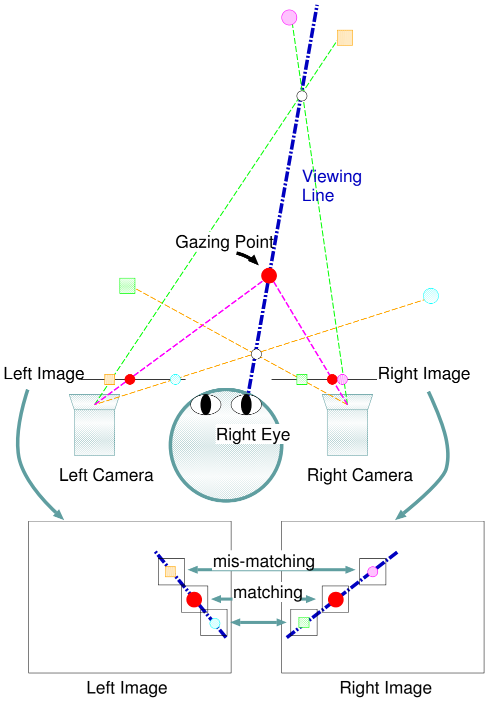


Figure 11: Stereo matching along the gaze-direction line to detect a 3D human gazing point

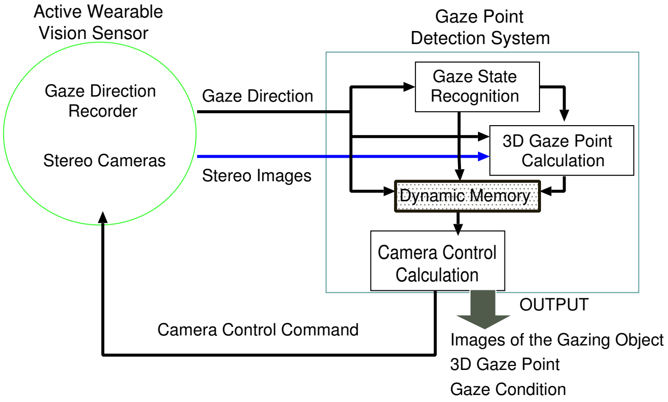
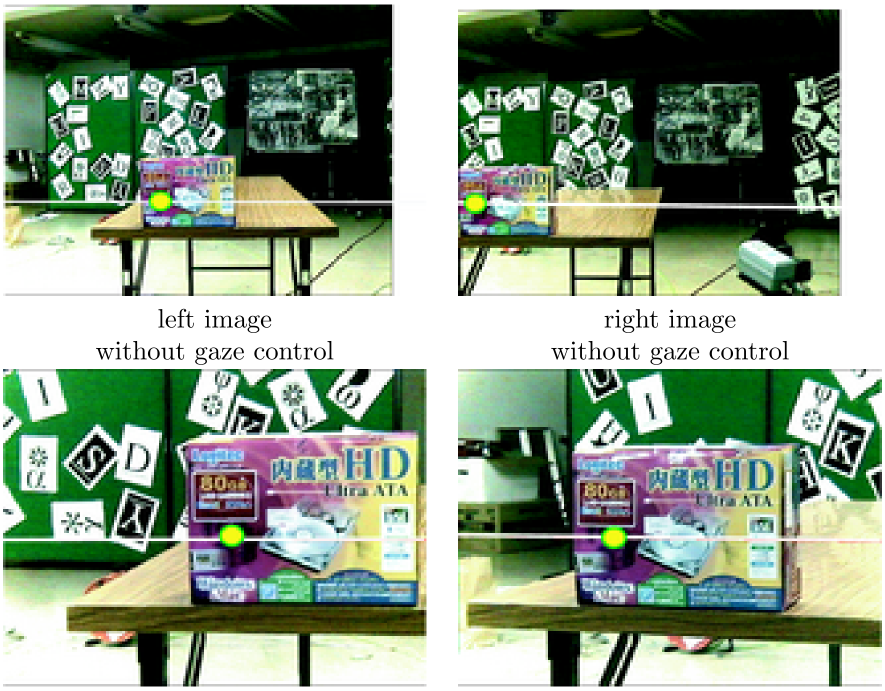


Figure 12: Control scheme of 3D gaze point detection and camera control



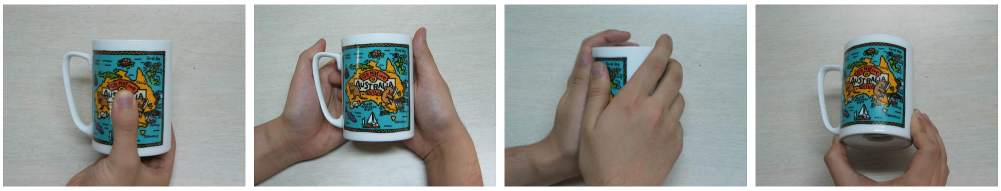
left image
without gaze control

right image
without gaze control

left image
with gaze control

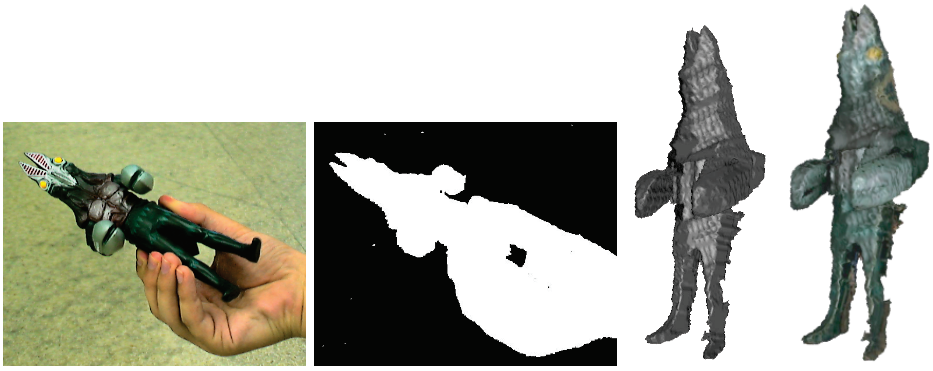
right image
with gaze control

Figure 13: Results of 3D gaze point detection and camera control



(a) (b) (c) (d)

Figure 14: Categorization of hand-object relationships: (a) Shape acquisition (b) Surface texture acquisition (c) Haptic texture acquisition (d) Total appearance acquisition



(a) (b) (c) (d)

Figure 15: 3D digitization of an alien figure: (a) Captured image (b) Silhouette image (c) 3D shape (d) Digitized object

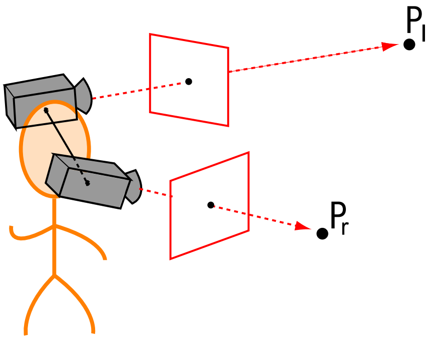


Figure 16: Binocular independent fixation camera control

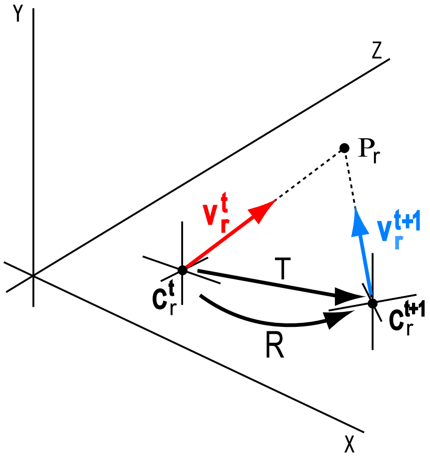


Figure 17: Geometric configuration in the right camera fixation control.

object surface are detected by Harris Corner Detector [15] and then, the 3D location of the feature points are calculated by stereo analysis.
3. Camera Motion Recovery – Based on 3D feature point data observed from multiple viewpoints, the 3D camera position and motion in the object centered coordinate system are estimated by a robust ICP algorithm [16].
4. Depth Map Acquisition – For each viewpoint, a depth map is computed by region based stereo analysis. Then, the vacant space based on the stereo vision is computed.
5. Silhouette Acquisition – For each viewpoint, an object&hand silhouette is computed by background subtraction. Then, we compute the vacant space based on the silhouette.
6. Vacant Space Carving – The 3D block space is carved by a group of vacant spaces computed from multiple viewpoint to generate a 3D object shape.

Since the wearable vision system can capture video images of the hand manipulation, we can obtain densely placed multi-view stereo images to generate an accurate 3D object shape.

We applied the method to a complex alien figure as shown in Figure 15(a). Figures 15(b) and (c) illustrate an extracted object&hand silhouette and the result of the vacant space carving, respectively. After mapping the texture, we obtained the 3D digitized object shown in Figure 15(d).

4.3 Estimation of 3D human motion trajectory by binocular independent fixation camera control

Here, we address 3D human motion trajectory estimation using the active wearable vision system. In the previous two methods, the active cameras work as stereo cameras sharing the visual field with human to understand what he/she is looking at. In other words, the cameras captured convergent multi-view images of a human interested object.

In this research, on the other hand, a pair of active cameras are used to get the 3D surrounding scene information, which enables us to estimate the 3D human motion (i.e. to be specific, camera motion) in the scene. That is, the cameras capture divergent multi-view images of the scene during human motion.

To estimate the 3D human motion trajectory with a pair of active wearable cameras, we introduced what we call the *binocular independent fixation camera control* (Figure 16): each camera automatically fixates its optical axis at a selected scene point (i.e. the *fixation point*) and keeps the fixation during human motion. This may be called *the cross-eyed vision.*

Suppose a pair of wearable cameras are calibrated and their optical axes are fixated at a pair of corresponding scene points during human motion. Let $\boldsymbol{T}$ and R denote the translation vector and rotation matrix describing the human motion between t and $t+1$, respectively.

Figure 17 shows the geometric configuration in the right camera fixation control: the projection center

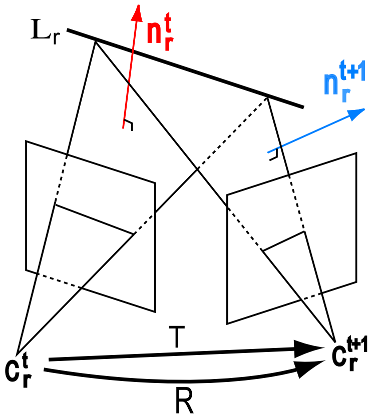


Figure 18: Geometry based on the line correspondence of the right camera.

moves from C_{r}^{t} to C_{r}^{t+1} while keeping the optical axis fixated at P_{r}. From this configuration, we can get the following constraint:

$$\lambda R_0 \boldsymbol{v}_{\mathrm{r}}^{t} \quad = \quad \lambda' R_0 R \boldsymbol{v}_{\mathrm{r}}^{t+1} + \boldsymbol{T},$$

where λ and λ' are non-zero constants, and $\boldsymbol{v}_{\mathrm{r}}^{t}$ and $\boldsymbol{v}_{\mathrm{r}}^{t+1}$ denote the viewing direction vectors at t and $t+1$, respectively. We assume that the orientation of the world coordinate system has been obtained by applying rotation matrix R_0^{-1} to the orientation of the right-camera coordinate system at time t. The above equation is rewritten by

$$\det \left[\begin{array}{ccc} R_0 \boldsymbol{v}_{\mathrm{r}}^{t} & R_0 R \boldsymbol{v}_{\mathrm{r}}^{t+1} & \boldsymbol{T} \end{array} \right] \quad = \quad 0, \qquad (1)$$

which gives the constraint on the human motion. The constraint similar to (1) is obtained from the the fixation control of the left camera.

The human motion has 6 degrees of freedom: 3 for rotation and 3 for translation. The number of constraints on the human motion derived from the fixation control of two cameras, on the other hand, is two ((1) and that computed from the left camera). We therefore need more constraints.

To derive sufficient constraints to estimate the human motion, we employ correspondences between lines located nearby around the fixation point.

We assume that we have established the image correspondence of a 3D line L_{r} at time t and $t+1$, where line L_{r} is selected from a neighborhood of the fixation point of the right camera (Figure 18). Based on this geometric configuration, we obtain the following constraint from a line correspondence between two image frames captured by the right camera:

$$\mu_{\mathrm{r}} \boldsymbol{L}_{\mathrm{r}} \quad = \quad (R_0 \boldsymbol{n}_{\mathrm{r}}^{t}) \times (R_0 R \boldsymbol{n}_{\mathrm{r}}^{t+1}), \qquad (2)$$

where $\boldsymbol{L}_{\mathrm{r}}$ denotes the unit direction vector of the focused line L_{r} in the world coordinate system, and $\boldsymbol{n}_{\mathrm{r}}^{t}$ and $\boldsymbol{n}_{\mathrm{r}}^{t+1}$ normal vectors of the planes formed by two projection centers (C_{r}^{t} and C_{r}^{t+1}) and 3D line L_{r}, respectively. μ_{r} is a non-zero constant and depends on the focused line. We see that this constraint is linear homogeneous with respect to the unknowns, i.e., R and the non-zero constant.

The constraints derived from the line correspondence depend only on the rotation of the human motion. We can thus divide the human motion estimation into two steps: the rotation estimation and the translation estimation.

The first step is the rotation estimation. Suppose that we have correspondences of n focused lines between two temporal frames. Then, we have $n+3$ unknowns (n scale factors and 3 rotation parameters) and $3n$ constraints. Consequently, we can estimate the rotation if we have correspondences of more than two focused lines. The same property holds for the left camera.

Finishing the estimation of the rotation matrix, unknowns are only the translation vector. Given the rotation matrix, the constraint derived from the camera fixation becomes homogeneous linear with respect to the unknowns. Hence, we can obtain the translation of the human motion up to scale from two independent fixation points. That is, whenever we estimate the translation of the human motion over two frames, we have one unknown scale factor. The trilinear constraints[17] on corresponding points over three frames enable us to adjust the unknown scales with only linear computation.

Comparing our binocular independent fixation camera control with ordinary stereo vision, ours has the following advantages:

- Since the image feature matching in our method is conducted between temporally separated image frames captured from almost the same viewpoint (i.e. by a moving camera), image features to be matched have enough similar appearances to facilitates the matching.

- The similar computational scheme as ours holds when we put cameras at the fixation points in the scene, which are looking at a person. Since the distance between the fixation points can be much longer than the baseline length of ordinary stereo cameras and the accuracy in the 3D measurement depends on the baseline length between the cameras, our method can realize more accurate 3D position sensing than stereo vision.

To verify the effectiveness of the proposed method, we moved a pair of cameras in a room and estimated their 3D motion trajectory (Figure 19). We marked 35 points on the trajectory and regarded them as sensing positions during the motion. We then applied the binocular independent fixation camera control at the sensing positions to estimate the right camera motion.

At the starting point of the camera motion, we manually selected a fixation point. During the estimation, we manually updated fixation points 8 times; when the camera moves largely and the surrounding scene changes much, we have to change fixation points. We used two focused lines for each camera. Figure 20 shows an example of image pairs captured at a sensing position, where the fixation point (the black circle) and two focused lines (the thick black lines) are overlaid.

Figure 21 shows the estimated trajectory of the right camera motion, obtained by concatenating the estimated motions at the sensing positions. In the figure, S denotes the starting point of the motion.

The vertical motion, which is almost zero, was almost accurately estimated. As for the horizontal motion, however, while the former part (from S to P in the figure) of the estimated trajectory is fairly close to the actual trajectory, the latter part (after P) deviates from the actual trajectory. This is because the motion at P was incorrectly estimated; since the motion was incrementally estimated, an incorrect estimation at a sensing position introduced a systematic deviation in the subsequent estimations. While not implemented, this problem can be solved by applying some global trajectory optimization to a group of local motion estimations.

5 Concluding Remarks

In this paper we discussed how we can extend visual information processing capabilities by using a group of multi-view cameras.

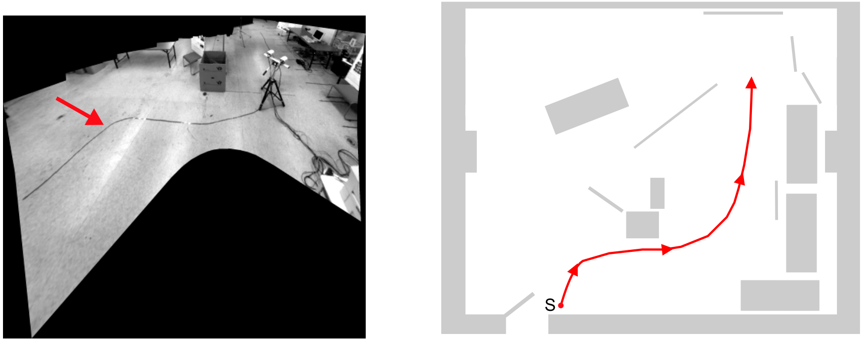
First we address a ubiquitous vision system, where a group of network-connected active cameras are embedded in the real world to observe dynamic events from various different viewpoints. We demonstrated its effectiveness with the cooperative distributed active multi-target tracking system and the high fidelity 3D video generation system.

In the latter half of the paper, we proposed a wearable active vision system, where multiple cameras are placed nearby around human eyes to share the visual field. We demonstrated its effectiveness with 1) estimation of 3D human gaze point and close-up image acquisition of a focused target object, 2) 3D digitization of a hand-held object, and 3) estimation of 3D human motion trajectory.

We believe ubiquitous and wearable visions systems enable us to improve human-computer interfaces and support our everyday life activities.

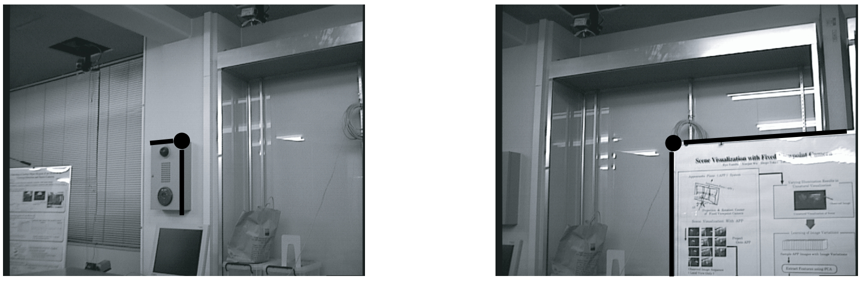
This series of researches are supported by 1) Grant-in-Aid for Scientific Research No. 13308017, No. 13224051 and No. 14380161 of the Ministry of Education, Culture, Sports, Science and Technology, Japan, 2) National research project on Development of High Fidelity Digitization Software for Large-Scale and Intangible Cultural Assets of the Ministry of Education, Culture, Sports, Science and Technology, Japan, and 3) Center of Excellence on Knowledge Society Infrastructure, Kyoto University.

Research efforts and supports to prepare the paper by all members of our laboratory and Dr. A. Sugimoto of the National Institute of Informatics, Japan are gratefully acknowledged.



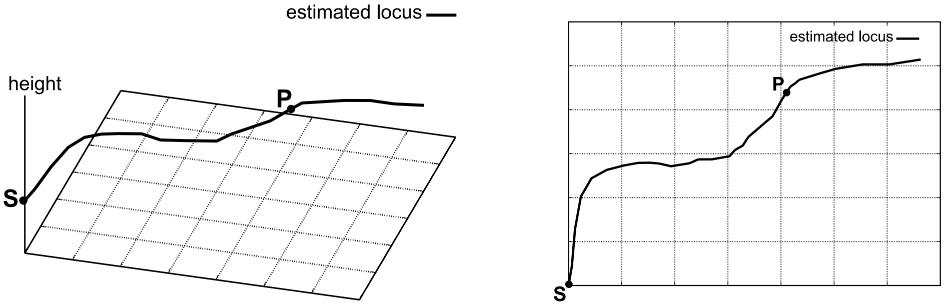
(a) wide view scene representation (b) top view of the scene

Figure 19: Camera motion trajectory.



(a) left camera image (b) right camera image

Figure 20: Example images acquired for the binocular independent fixation.



(a) 3D representation (b) top-view representation

Figure 21: Estimated trajectory of the camera motion.

References

[1] Matsuyama, T. Cooperative Distributed Vision - Dynamic Integration of Visual Perception, Action, and Communication-, *Proc. of Image Understanding Workshop*, pp. 365-384, Monterey CA, 1998.

[2] Matsuyama, T. and Ukita, N. Real-Time Multi-Target Tracking by a Cooperative Distributed Vision System, *Proc. of IEEE*, Vol.90, No.7, pp.1136-1150, 2002.

[3] Moezzi, S., Tai, L., and Gerard, P. Virtual View Generation for 3D Digital Video, IEEE Multimedia, pp.18-26, 1997.

[4] Matsuyama, T., Wu, X., Takai, T., and Wada, T. Real-Time Dynamic 3D Object Shape Reconstruction and High-Fidelity Texture Mapping for 3D Video, IEEE Trans. on Circuits and Systems for Video Technology, Vol.CSVT-14, No.3, pp.357-369, 2004.

[5] Sumi, K., Sugimoto, A., and Matsuyama, T. Active Wearable Vision Sensor: Recognition of Human Activities and Environments, Proc. of IEEE International Conference on Informatics Research for Development of Knowledge Society Infrastructure, pp.15-22, Kyoto, 2004.

[6] Wada, T. and Matsuyama, T. Appearance Sphere:Background Model for Pan-Tilt- Zoom Camera, Proc. of 13th ICPR, pp.A-718-A-722, Wienna Austria, 1996.

[7] Matsuyama, T., Ohya, T., and Habe, H. Background Subtraction for Non-Stationary Scenes, Proc. of 4th Asian Conference on Computer Vision, pp.662-667, 2000

[8] Matsuyama, T., Hiura, S., Wada, T., Murase, K., and Yoshioka, A. Dynamic Memory: Architecture for Real Time Integration of Visual Perception, Camera Action, and Network Communication, Proc. of Comuter Vision and Pattern Recognition Conference, pp.728-735, 2000

[9] Wada, T. and Wu, X. and Tokai, S. and Matsuyama, T. Homography Based Parallel Volume Intersection: Toward Real-Time Reconstruction Using Active Camera, Proc. of International Workshop on Computer Architectures for Machine Perception, pp.331–339, 2000.

[10] Matsuyama, T. and Wu, X. and Takai, T. and Nobuhara, S. Real-Time 3D Shape Reconstruction, Dynamic 3D Mesh Deformation, and High Fidelity Visualization for 3D Video, International Journal on Computer Vision and Image Understanding, 2004 (in press)

[11] Kenmochi, Y. and Kotani, K. and Imiya, A. Marching Cubes Method with Connectivity, Proc. of 1999 International Conference on Image Processing, pp. 361–365, 1999.

[12] Sugimoto, A., Nakayama, A., and Matsuyama, T. Detecting a Gazing Region by Visual Direction and Stereo Cameras, Proc. of the 16th int'l Conf. Pattern Recognition, Vol. III, pp.278-282, 2002.

[13] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Stuetzle, W. Surface reconstruction from unorganized points, SIGGRAPH '92, Vol.26, pp.71-78, 1992.

[14] Kutulakos, K.N. and Seitz, S. M. A theory of shape by space carving, IEEE Int'l Conf. on Computer Vision, pp.307-314, 1999.

[15] Harris, C. J. and Stephens, M. A combined corner and edge detector, 4th Alvey Vision Conf., pp.147–151, 1988

[16] Besl, P.J. and McKey, N.D. A method for registration of 3-D shapes, IEEE Trans. PAMI, Vol.14, No.2, pp.239-256, 1992

[17] Hartley, R. and Zisserman, A. *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, 2000.