Multiple Active Camera Assignment for High Fidelity 3D Video

Sofiane Yous, Norimichi Ukita, and Masatsugu Kidode
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara, 6300192 Japan
{yous-s, ukita, kidode}@is.naist.jp

Abstract

We are designing a self controlling active camera system for a 3D video of a moving object (mainly human body). We made up our system of cameras with long focal length lenses for high resolution input images. However, such cameras can get only partial views of the object. We present, in this paper, a multiple active (pan-tilt) camera assignment scheme. The goal is to assign each camera to a specific part of the moving object so as to allow the best visibility of the whole object. For each camera, we evaluate the visibility to the different regions of the object, corresponding to different camera orientations and with respect to the field of view of the camera in question. Thereafter, we assign each camera to one orientation in such a way to maximize the visibility to the whole object.

1 Introduction

A 3D video is an interactive video system where the viewer has the freedom to choose and change his viewpoint. Such a system has been widely tackled by the past and is still getting an increasing interest. Consequently, numerous 3D video systems have been proposed, such as the systems presented in [5], [1], [4], [6], [7], and [8]. The focus of these systems is a human body acting within a scene around the which, a distributed fixed camera system is spread for a real-time synchronized observation. [5],[1], and [8] generate the final video in off-line, while [3],[6],[7], and [9] employ a volume intersection method on a PC cluster in order to achieve a full 3D shape reconstruction. However, the quality did not reach a practical level yet. Among the key reasons of these limits[3], are:

- Wide area observation.
- High fidelity: It rely upon 3D reconstruction details and texture mapping performances, and consequently, upon the resolution of input images.

If we want to widen the observable scene area without increasing the number of cameras, we need to widen the FOV(field of view) of our cameras by shortening their respective focal length. Consequently, the resolution of input images is reduced and the high fidelity affected. Similarly, if we want to get higher resolution input images using the same fixed camera system, we need to narrow the FOV of the cameras by lengthening their the focal length. Consequently, the observable area is narrowed. In other words, *high fidelity* and *wide observation area* are two interlinked problems such that, we cannot improve the fidelity by increase the resolution, without affecting the observable area, and vice versa.

In order to increase the resolution of input images without affecting the area observation, we have been designing an active camera system. With an active camera system, it is not required to get a continuous observation of the whole scene. Consequently, the above mentioned interlink between *high fidelity* and *wide observation area* is broken and it becomes possible to increase the resolution of input images independently. In contrast, our active camera system need to be endowed with self-control capability. Self-control allows the active camera system to: 1) follow the object movement within the scene, and 2) assign each camera to a specific part of the moving object, so as to allow the best visibility of the whole object.

Our aimed 3D video system, as shown in figure 1, consists in two subsystems. The final 3D reconstruction is processed in off-line (figure 1-b), whereas self-control represents the on-line subsystem (figure 1-a). The mission of the on-line subsystem is to control the active camera system, based on the analysis of a rough 3D surface provided by a real-time 3D reconstruction process, in order to provide high resolution input images to the off-line subsystem.

Our work can be regarded as what is referred to as camera planning or more generally, sensor planning in robotics field. The focus of camera planning is to control a camera in order to satisfy geometric, perceptual and aesthetic needs. The methods of camera planning can be classified mainly into four classes[10]. Algebraic systems[11] repre-

Proceedings of the Fourth IEEE International Conference on Computer Vision Systems (ICVS 2006)
0-7695-2506-7/06 $20.00 © 2006 IEEE

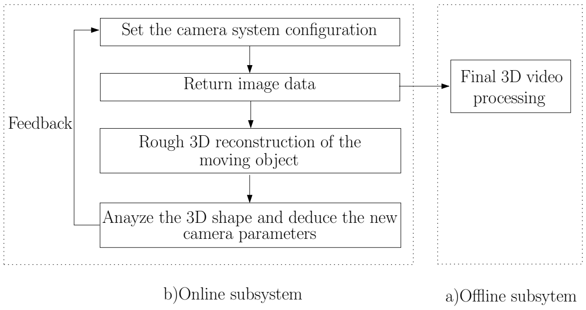
Set the camera system configuration
Return image data
Rough 3D reconstruction of the moving object
Anayze the 3D shape and deduce the new camera parameters
Feedback
Final 3D video processing
b)Online subsystem
a)Offline subsytem

Figure 1. Overview of our aimed 3D video system

sent the camera set-up as vector algebra and directly compute the solution. In the interactive systems class[12], the camera is set up based on the user directives. As for reactive real-time systems class[13], the camera is set up based on motion planning in a dynamic virtual environment. Finally, the constraint-based systems[14] model the problem as constraints and objective functions and solve it as an optimization problem. Our work fall into this latter class of systems, and can be considered as a constraint-based multiple camera planning. However, 'planning' in our system has a narrower meaning than it is for camera or sensor planning, as it concerns only the camera orientation.

2 Overview of Our Algorithm

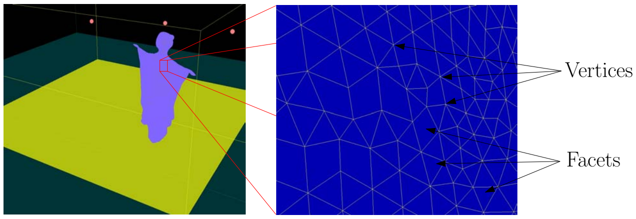

Figure 2. Input data : 3D mesh surface.

We have as input of our scheme, a 3D reconstruction of the object (figure 1) in the form of a mesh surface defined by a set of facets(figure 2). Each facet is defined by a set of vertices. Using these vertices and their order we can deduce the orientation of the facet[1] and compute its outward normal vector(Figure 5-b). As a main purpose of this paper, Camera assignment scheme consists in analyzing this 3D surface in order to deduce the orientations of our cameras that allows the best visibility to the whole surface.

The proposed algorithm can be summarized in three steps:

1. First, we project the 3D surface to the panoramic plane[2].
2. Next, for each camera, we evaluate the different orientations to the object image (corresponding to the panoramic plane), with respect to the the FOV of the camera under consideration. For a given camera orientation, the evaluation concerns the visibility of the 3D object region covered by the FOV of the camera. We refer to the object image region, corresponding to a given orientation, by a window. In section 3, we will present a windowing scheme whose goal is to split the object image into several windows with respect to the FOV of the camera in question. As for the visibility evaluation for each orientation, it will be the purpose of the fourth section.
3. Finally, we seek for the the best assignment (camera, window) configuration that maximize the global visibility. This will be the subject of the fifth section.

In the rest of this paper, we adopt the following notation:
$c_i|_{i=1..N}$: The set of N cameras.
$f_j|_{j=1..M}$: The set of M facets composing the 3D mesh surface.
$\overrightarrow{n_i}$: The unit outward normal vector of facet f_j.
c_i : The optical center of camera c_i.
t_j : The centroid of facet f_j.
We begin by introducing the main constraints to be taken into account in the rest of this paper.

2.1 Assignment Constraints

These constraints have the role to restrain and lead our assignment process.

2.1.1 Visibility Constraint

The visibility constraint tells if a given facet can be viewed from a given camera, regardless of occlusion. It can be verified if the dot product of the normal vector of the facet and the vector associated to the optical center of the camera and the centroid of the facet, is negative.

$$f_j \;\; is\, visible\, from \;\; c_i \Rightarrow (\overrightarrow{n_i} \cdot \overrightarrow{c_i t_j}) < 0 \qquad (1)$$

Furthermore, If both vectors are normalized, then the dot product can quantify the visibility (sec. 4.1). That is, the greater the product, the better the visibility.

[1] the outward face of a facet is defined by the counterclockwise order of its vertices

[2] The panoramic plane of a given camera is a reference plane corresponding to a chosen camera orientation. All the images referred to in this paper correspond to this plane

2.1.2 Accessibility Constraint

Suppose we have two facets respecting both the visibility constraint, with respect of a given camera, and having the same projection onto the panoramic plane. The farther facet can be occluded by the nearest, which make it inaccessible from the camera under consideration. Thus, we can say that the accessibility constraint can be violated by occlusion.

Given a camera c_i, we can define the set of visible and accessible facets denoted $Vf\,(i)$, by building a depth image D_i and an index image I_i. Let us denote by A_i the object image, if $A_i(x, y)$ is the projection of a surface point belonging to a facet f_j to the panoramic plane of camera c_i, then $D_i(x, y)$ is the distance of that facet from the camera, and $I_i(x, y) = j$ its index (identifier).

$Vf\,(i)$ can be defined such that:

$$j \in Vf\,(i) \Leftrightarrow \exists_{x,y} I_i(x, y) = j \qquad (2)$$

2.1.3 Connectivity Constraint

The connectivity constraint impose to a given camera to be assigned to a connected region of the 3D surface, no matter if it can get a larger view (this constraint is valid only in the assignment process). This constraint is mainly useful in the presence of self-occlusion.

2.1.4 3D Reconstruction Constraint

The purpose of this constraint is to ensure that we dispose, at least, of two views toward each surface point, as a matter of 3D reconstruction.

2.1.5 Depth Constraint

The distance from the camera should be taken into account in the assignment process. That is, if we have two regions with similar visibility, then the nearest region is given more priority in the assignment process.

2.1.6 FOV Constraint

For each camera orientation, only the region defined by the FOV of the camera in question, is taken into account.

3 Windowing Scheme

Given a 3D mesh surface, the goal of the windowing operation in to define, for a given camera, the set of possible orientations ,and for each orientation, select the set of facets to be involved in its visibility evaluation. This should be accomplished such that:

1. The set of orientations should cover the entire 3D surface region visible from the camera in question.
2. The fewer the orientation, the better.
3. The aforementioned connectivity constraint is respected.

To achieve such a goal, we proceed with gradually splitting the depth image into several windows. After a given window in set to a predefined position[3], a flood-filling is applied to the region of interest defined by the window in order to extract one connected region. We need to make sure to respect the same order in setting the first window position and the first point to apply the flood-filling. Then, the window position is repeatedly readjusted to fit the best with the connected region, and for each new position, the flood-filling is reapplied to find the new limits of the connected region. Finally, the connected region is withdrawn from the depth image and used as a mask of the index image in order to establish the set facets to be associated to the window.

Let us denote by $w_k^i|_{k=1..L}$the L resultant windows, and (x_k, y_k) their respective off-set addresses with respect to the depth image. The set of facets $Vf(i)$ is split into $Vf(i, k)$ such that:

$$\begin{cases} \bigcup_k Vf(i,k) = Vf(i) \\ \\ \wedge \\ j \in Vf(i,k) \Leftrightarrow \exists_{x,y} \begin{cases} w_k^i(x,y) > 0 \\ \wedge \\ A_i(x_k + x, y_k + y) = j \end{cases} \end{cases} \qquad (3)$$

The proposed algorithm can be summarized as follows(Figure 3):

1. Calculate the bounding rectangle of the object depth image if it is the first iteration, or the remaining regions of the object depth image if not. The bounded region will be set as the region of interest of the depth image, as shown in Figure 3-a.
2. Set a window to an expected position(Figure 3-a). To do so, we need to define, beforehand, a fixed order such as Top-Down Left-right. The offset of the first window should coincide with this of the bounding rectangle.
3. Repeat:
   - Apply a flood-filling starting from the first point with respect to the predefined order, as shown in Figure 3-b.
   - Adjust the window(in term of position) to the connected region.

[3] The position in this context refer to the off-set with respect to the depth image

in so far as the top and left borders of the window fit with the external contour of the connected component. if it is the last horizontal window, we consider the right border of the window instead of the left, and similarly the bottom border in stead of the top if it is the last vertical window (Figure 3-c).

4. Delete the area corresponding to the connected component from the depth image and save it as a mask associated to the actual window.

5. If the depth image is not empty, then goto 1.

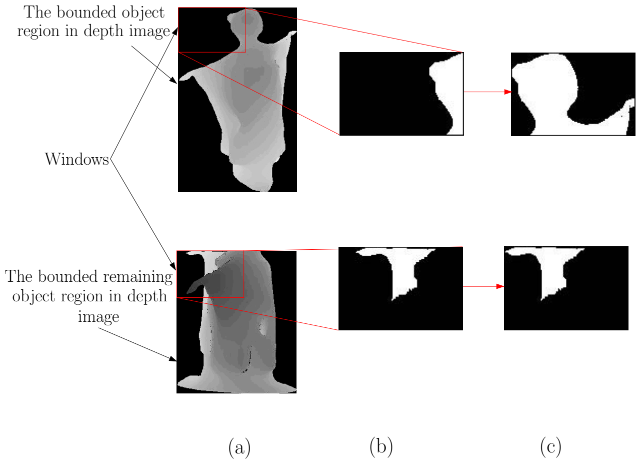
The bounded object region in depth image
Windows
The bounded remaining object region in depth image
(a)
(b)
(c)

Figure 3. Windowing scheme: From (a) to (b) a flood filling is applied, and an adjustment is applied to get (c). The top cycle corresponds to the first windowing iteration for a given camera. The missing part in the depth image of the bottom cycle was windowed and deleted in previous iterations.

4 Visibility Quantification

Our assignment scheme is based inter alia on the visibility evaluation of the 3D surface. We define three levels of visibility; facet-wise, local, and global visibility.

4.1 Facet-wise Visibility Quantification

The facet-wise visibility is the direct application of the visibility constraint. It concerns the visibility of a given facet from a given camera. It can be expressed by the absolute value of the dot product between the unit normal vector of the facet in question, and the normalized vector connecting the optical center of the camera and the centroid of the facet.

If the unit vector, corresponding to the optical ray of a camera c_i toward t_j, is:

$$\overrightarrow{r_{i,j}}= \frac{\overrightarrow{c_i t_j}}{\left\|\overrightarrow{c_i t_j}\right\|}$$

Then, the facet-wise visibility is given by:

$$F(i,j) = \left|\overrightarrow{r_{i,j}} \cdot \overrightarrow{n_i}\right| \qquad (4)$$

4.2 Local Visibility Quantification

The local visibility level concerns, for a given camera, the visibility toward each of its windows. For a given window, it involves the facet-wise visibilities of all facets concerned by the actual window. The formulation of the local visibility is very sensitive, as it can express our assignment strategy. The simplest way is to sum the facet-wise visibilities of the corresponding facets. Albeit simple, this solution is not the best for mainly two reasons:

1. A narrow region made of tiny facets is given similar evaluation as a wide region with large facets, if the two regions have a similar number of facets.

2. One region is given the same evaluation whatever its distance from the camera.

So as to make the local visibility as expressive as possible, the proposed formulation should:

1. Respect the aforementioned depth constraint.

2. Involve the facet area. That is, the contribution of each facet in the local visibility should be proportional to its surface area.

3. Be normalized.

Let us denote by:

$cpt(i,k)$: The number of facets visible from a camera c_i and corresponding to a window w_k^i.

$\bar{D}\left(i,k\right)$: The mean depth of a window w_k^i with respect to a camera c_i such that:

$$\bar{D}\left(i,k\right) = \frac{\sum_{j\in Vf(i,k)}\left(D\left(i,j\right)\right)}{cpt_{i,k}}$$

where $D\left(i,j\right)$ denotes the depth of f_j.

$\bar{S}\left(i,k\right)$, the area of the 3D surface related to window w_k^i of camera c_i such that:

$$\bar{S}\left(i,k\right) = \sum_{j\in Vf(i,k)} S\left(j\right)$$

where $S\left(j\right)$ denotes the surface area of a facet f_j in the 3D space.

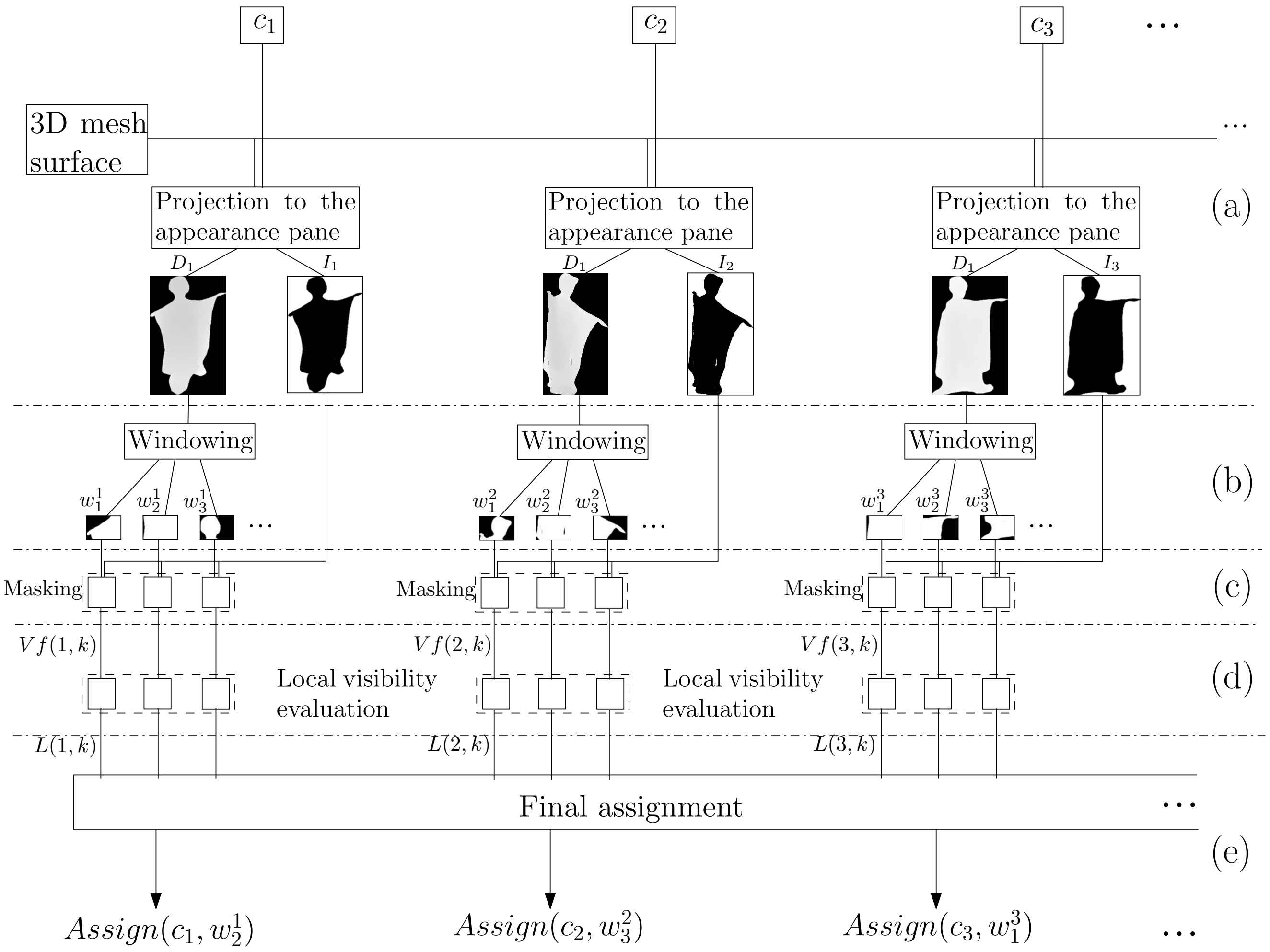

Figure 4. Assignment scheme: The different steps of our proposed scheme as explained in the the text. $Vf(i)$, $Vf(i,k)$, and $L(i,k)$ refer respectively to : the set of facet visible from camera c_i, these corresponding to a window w_k^i, and the local visibility corresponding to c_i w_k^i

The local visibility of a window w_k^i from a camera c_i is given by:

$$L(i,k) = \frac{\bar{D}\,(i,k)}{\bar{S}\,(i,k)} . \sum_{j \in Vf(i,k)} \frac{F\,(i,j)\,.S\,(j)}{D\,(i,j)} \qquad (5)$$

For a set of facets $Vf(i,k)$, the local visibility is the normalization of the sum the scaled facet-wise visibilities. The scale is the ratio between the 3D surface area of the facet and its mean depth in respect to the camera in question. As for the normalizing factor, it is the ration between the mean depth of the surface defined by the set of facets and its 3D area area in the 3D space.

4.3 Global Visibility Quantification

The purpose of the assignment mechanism is to maximize the global visibility of the 3D surface. After the system is set to a given configuration, this global visibility can be expressed simply by averaging the local visibilities of all cameras.

$$G = \frac{1}{N} \sum_{i,k} L\,(i,k) \qquad (6)$$

5 Global Assignment Scheme

After having addressed the windowing and visibility quantification issues, we are now able to establish a global assignment scheme. The purpose is to assign each camera to one of its windows so as to get the highest global visibility of the whole 3D surface. Mainly, the 3D reconstruction constraint is to be considered in the proposed algorithm that can be summarized as follows:

1. For each camera c_i, compute the set $Vf(i)$: Project the 3D mesh surface to the panoramic plane and build the depth and index images I_i and D_i respectively, as shown in Figure 4-a.
2. Split $Vf(i)$ into $Vf(i,k)$ (Figure 4-b,c): Apply the above-mentioned windowing scheme to get the windows w_k^i (Figure 4-b), thereafter, mask the index image I_i using these windows in order to get the sets $Vf(i,k)$ (Figure 4-c).
3. For each couple (c_i, w_k^i), calculate $L(i,k)$: Evaluate the local visibility between all cameras and their respective windows(Figure 4-d).
4. Repeat:
   (a) Select the pair (c_i, W_k^i) having the highest local visibility and assign the camera c_i to the window w_k^i.
   (b) Delete all facets chosen twice and accordingly, update L(the local visibility) for all windows (of all cameras) comprising the deleted facets.

   until no camera or no facet left (Figure 4-e).

6 Experimental Results

In order to evaluate the effectiveness of our presented assignment scheme, we conducted an experiment whose results will be presented in this section.

6.1 Experimental Environment

The environment of our experiment consisted in a kimono lady dancing a folkloric dance within a scene around the which, 25 cameras were spread, as shown in figure 5. At each frame, a set of 25 images were captured and employed in the 3D mesh surface reconstruction.

We applied our proposed multi camera assignment scheme to the 3D mesh surface at a given frame, while considering the same camera system but with narrower FOV (longer focal length).

6.2 Evaluations

Figure 6 shows an assignment sample. The top row images are depth images for 5 selected cameras, while these of the bottom show the respective windows designated by the assignment scheme. Figure 7 shows the local visibility evaluation of the 25 cameras, after applying our proposed assignment scheme. The global visibility is around 0.82. If the average angle of view (angle of incidence) to a surface point can be expressed as $\arccos(G)$ (G:the global visibility), then it is about 35 degrees. As for Figure 8, it shows the facet-wise visibility evaluation for all surface points. Most of facets have a visibility greater than 0.6, which witness, in addition to Figure 7, the performance of our proposed scheme. Finally, Figure 9 presents the orientation of our 25 camera.

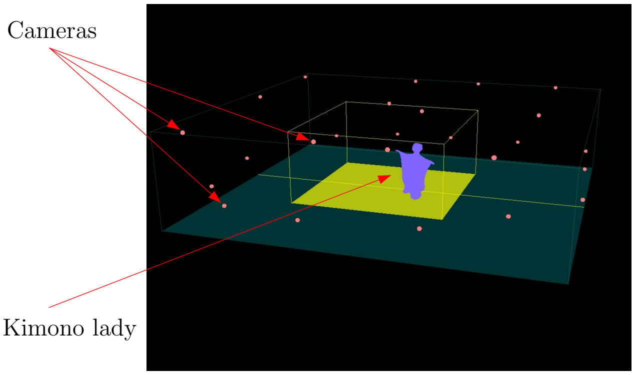

Figure 5. Experimental environment

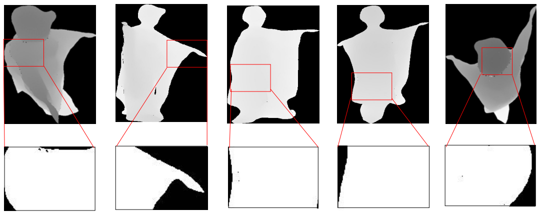

Figure 6. Assignment sample: Each depth image (top row) is associated to its associated window (lower row) designated by the assignment process. The assignment of 5 selected cameras is shown

7 Conclusion

Throughout this paper, we have presented a multiple active camera assignment scheme for high fidelity 3D video of a moving object. The active camera system allows us to get high resolution input images without affecting the wide area observation. Our active camera system, as made of long focal length cameras, can have only partial views of the mov-

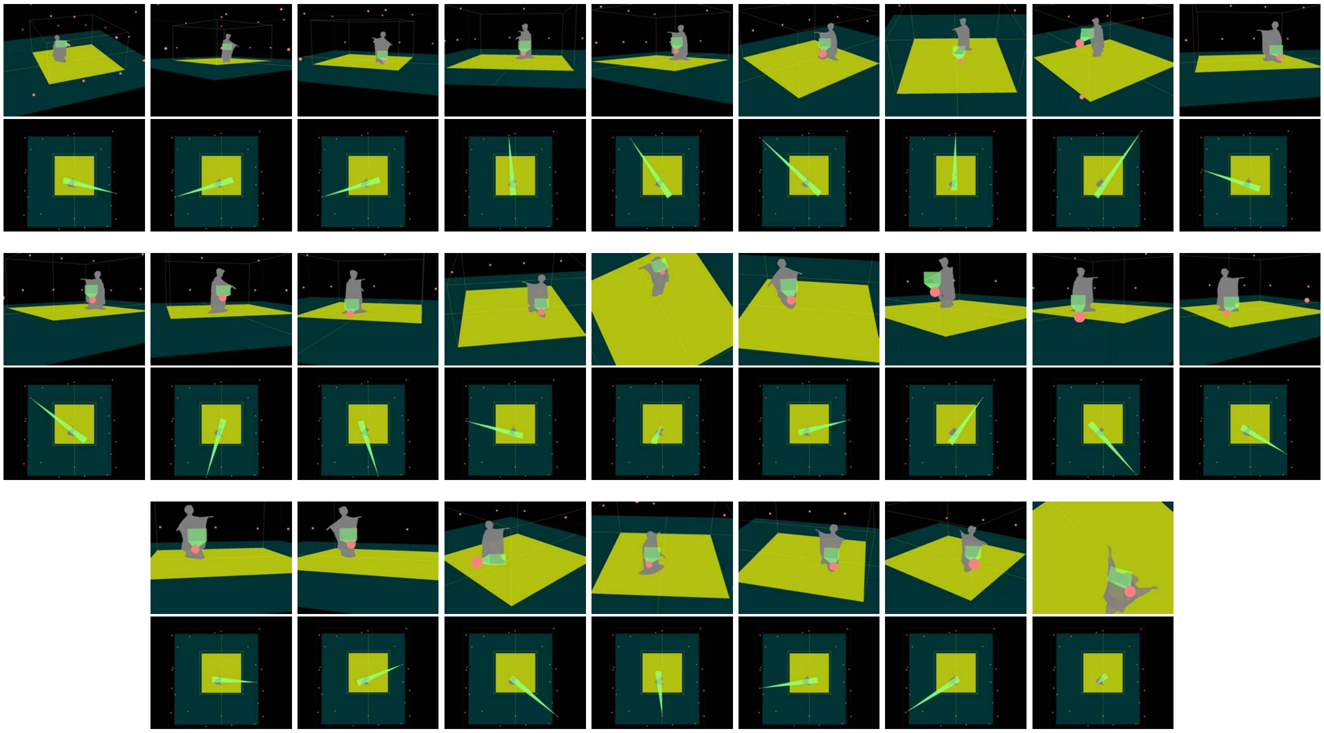

Figure 9. The result of the assignment scheme: each pair of column images refer to two views of the same camera orientation. The orientation is represented by a green cone.

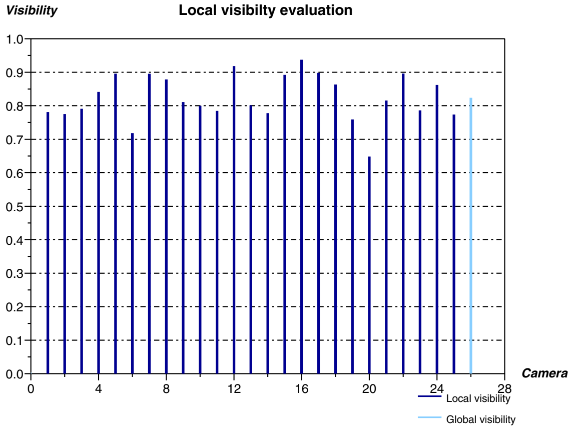

Figure 7. Local visibility evaluation

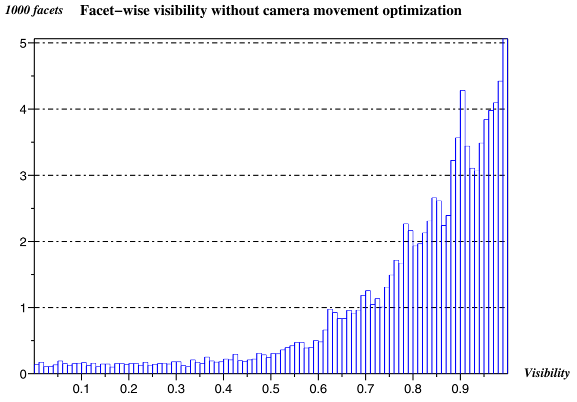

Figure 8. Facet-wise visibility histogram

ing object. therefore, The goal of our presented scheme is to assign each camera to a specific part, so us to get the best view of all part of the object. Our work is motivated by the aim to address the problem of high fidelity without being interfered by the wide area observation problem, while high fidelity is necessary for 3D video to reach a practical level.

The presented scheme needs to be processed to each frame separately. As a future work, we are planning to address the temporal generalization of our scheme with optimization of the inter-frame camera movemento.

Acknowledgments

This work is supported in part by PRESTO program of Japan Science and Technology Agency (JST), the National Project on Development of High Fidelity Digitalization Software for Large-Scale and Intangible Cultural Assets, and the 21st Century COE program ”Ubiquitous Networked Media Computing” of Nara Institute of Science and Technology (NAIST-JAPAN).

References

[1] T. Kanade, P. Rander, and P.J. Narayanan, ”Virtualized Reality: Constructing Virtual Worlds from Real Scenes”, *IEEE Multimedia, Immersive Telepresence* Vol. 4, No. 1, pp. 34-47, January, 1997.

[2] R.T. Collins, ”A Space-Sweep Approach To True Multi-Image Matching”, *IEEE Computer Vision and Pattern Recognition,* pp. 358-363, June 1996.

[3] T. Matsuyama, X. Wu, T. Takai, S. Nobuhara , ”Real-Time 3D Shape Reconstruction, Dynamic 3D Mesh Deformation, and High Fidelity Visualization for 3D Video”, *International Journal on Computer Vision and Image Understanding,* Vol. 96, pp.393-434.

[4] X. Wu and T. Matsuyama, ”Real-Time Active 3D Shape Reconstruction for 3D Video”, *In the proceeding of the 3rd International Symposium on Image and Signal Processing and Analysis,Rome, Italy,* pp. 186–191 September 18-20, 2003.

[5] S. Moezzi, L. Tai, P. Gerard, ”Virtual view generation for 3d digital video,” *IEEE Multimedia,*pp. 1826, 1997.

[6] E. Borovikov, L. Davis, ”A distributed system for real-time volume reconstruction,” *in Proceedings of International Workshop on Computer Architectures for Machine Perception, Padova, Italy,* pp. 183189, 2000.

[7] G. Cheung, T. Kanade, ”A real time system for robust 3d voxel reconstruction of human motions,” *in Proceedings of Computer Vision and Pattern Recognition, South Carolina, USA,* pp. 714720, 2000.

[8] J. Carranza, C. Theobalt, M. A. Magnor, H.-P. Seidel, ”Free-viewpoint video of human actors,” *ACM Transactions on Computer Graphics* Vol. 22(3), pp. 569-577, July 2003

[9] M. Li, M. Magnor, H.-P. Seidel, ”Hardware-accelerated visual hull reconstruction and rendering,” *In Proceedings of Graphics Interface (GI’03), Halifax, Canada,* pp. 65-71, June 2003.

[10] M. Christie, R. Machap, J. M. Normand, P. Olivier, J. Pickering, ”Virtual Camera Planning: A Survey”, *In proceedings of the 5th International Symposium on Smart Graphics, Frauenworth Cloister, Germany,* August 22-24, 2005)

[11] J. Blinn. ”Where am I? what am I looking at?” *IEEE Computer Graphics and Applications,* pp 76-81, July 1988.

[12] C. Ware and S. Osborne. ”Exploration and virtual camera control in virtual three dimensional environments,” *In proceedings of the Symposium on Interactive 3D Graphics, New York, NY, USA* ACM Press, pp. 175-183, 1990.

[13] N. Courty and E. Marchand. ”Computer animation: A new application for image-based visual servoing,” *In Proceedings of IEEE International Conference on Robotics and Automation,ICRA’2001,* Vol 1, pages 223-228, 2001.

[14] W. H. Bares, J. P. Gregoire, and J. C. Lester. ”Real-time Constraint-Based Cinematography for Complex Interactive 3D Worlds,” *In Proceedings of AAAI-98/IAAI-98,* pp. 1101-1106, 1998.

[15] K. Yachi, T. Wada, and T. Matsuyama, Kyoto University ”Human Head Tracking Using Adaptive Appearance Models with a Fixed-Viewpoint Pan-Tilt-Zoom Camera,” *In the proceeding of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition,*pp. 150, 2000