

Engineering

Industrial & Management Engineering fields

Okayama University

Year 2005

Coordination of appearance and motion
data for virtual view generation of
traditional dances

Yuji Kamon
Sharp Corporation
Yasuhiro Mukaigawa
Osaka University

Ryo Yamane
Okayama University
Takeshi Shakunaga
Okayama University

This paper is posted at eScholarship@OUDIR : Okayama University Digital Information Repository.

<http://escholarship.lib.okayama-u.ac.jp/industrial-engineering/7>

Coordination of Appearance and Motion Data for Virtual View Generation of Traditional Dances

Yuji Kamon* Ryo Yamane Yasuhiro Mukaigawa† Takeshi Shakunaga
Department of computer science, Okayama University
{kamon, ryo, shaku}@chino.cs.okayama-u.ac.jp

Abstract

A novel method is proposed for virtual view generation of traditional dances. In the proposed framework, a traditional dance is captured separately for appearance registration and motion registration. By coordinating the appearance and motion data, we can easily control virtual camera motion within a dancer-centered coordinate system. For this purpose, a coordination problem should be solved between the appearance and motion data, since they are captured separately and the dancer moves freely in the room. The present paper shows a practical algorithm to solve it. A set of algorithms are also provided for appearance and motion registration, and virtual view generation from archived data. In the appearance registration, a 3D human shape is recovered in each time from a set of input images after suppressing their backgrounds. By combining the recovered 3D shape and a set of images for each time, we can compose archived dance data. In the motion registration, stereoscopic tracking is accomplished for color markers placed on the dancer. A virtual view generation is formalized as a color blending among multiple views, and a novel and efficient algorithm is proposed for the composition of a natural virtual view from a set of images. In the proposed method, weightings of the linear combination are calculated from both an assumed viewpoint and a surface normal.

1. Introduction

Several research activities have examined the preservation of cultural treasures in digital archiving projects[1, 2]. Among the various subjects, intangible cultural treasures, such as traditional dances, are particularly challenging because no complete archiving technique for human motion has yet been developed.

In motion data acquisition, a motion capture system is constructed using marker tracking. This system can pre-

cisely measure the 3D positions of markers at each time. The acquired motion data can be used in several applications, including computer graphics and motion analysis. In addition, appearance data can also be archived using a set of cameras.

In appearance data acquisition, a human shape should be estimated in advance at each time. There are several researches which utilize human models for 3d shape estimation [5, 6]. For traditional dances, however, these methods are not appropriate because dancer wears traditional costumes with some ornaments which make dancer shape look very different from a human model. In our system, the 3d shape is reconstructed without using any human-model. Although previously appearance data and motion data have been used separately because the applications of these data have been thought of as being different from each other. However, if these two types of data can be combined for virtual view generation, several new applications can be realized. Generally, appearance data have been considered in world-centered coordinates and camera-centered coordinates. However, if a camera position is specified in dancer-centered coordinates, a novel virtual view can be generated very easily. In order to create a set of dancer-centered coordinates, it is essential to coordinate the appearance and motion data automatically. The present paper focuses on this problem after collecting the appearance and motion data.

The present paper is organized as follows. In Section 2, we describe how appearance data can be archived using a multiple camera system. In Section 3, we describe how motion data are archived and coordinated with the appearance data. In Section 4, we discuss how a virtual image sequence is generated from the archived dance data. An efficient method is proposed for generating virtual views from archived dance data. Finally, Section 5 presents experimental results that indicate how the coordination between the appearance and motion data works for the virtual view generation.

*Present address:Sharp Corporation

†Present address:Osaka University

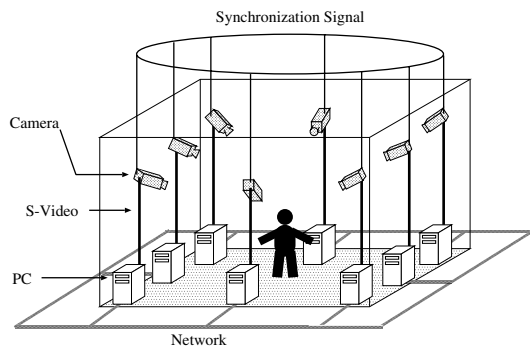


Figure 1. Multiple-camera system.

2. Appearance Registration

2.1. Multiple-camera system

In the present paper, we assume a target dancer that dances around the room center and the installation of multiple cameras to capture image sequences of the dancer from various points of view shown in Fig. 1. The image sequences are synchronized with a synchronization signal. Therefore, we can independently solve the shape reconstruction problem at each time.

All of the cameras are calibrated in advance both geometrically and photometrically. In our current system, geometric calibration is accomplished by using 384 points that are uniformly distributed throughout the dancing zone. Point-to-point correspondences between all of the calibration points and image points are manually provided for each camera by a human operator. Once the geometric calibration is accomplished, color calibration is performed for each camera. For this purpose, a standard color map created in our laboratories was used for statistic color calibration among eight cameras.

2.2. Reconstruction of 3D shape and voxel representation

In order to reconstruct a 3D shape of the dancer, a human region is extracted by each camera. In order to simplify the human region extraction, all of the background scenes are colored in green in advance, where the dancer is assumed to be wearing no green clothing or ornaments. Thus, a simple background suppression can provide a 2D figure of the dancer without mixing background scenes.

Since all of the cameras are calibrated in advance, the 3D shape of the dancer can be reconstructed from a set of input images at each time. In our first implementation, a 3D shape was approximated to a visual hull which is reconstructed from a set of binarized figures by a volume intersection method [12, 13]. While a visual hull is easy to reconstruct from input images, the reconstructed shape is slightly larger than the original shape. The shape differ-

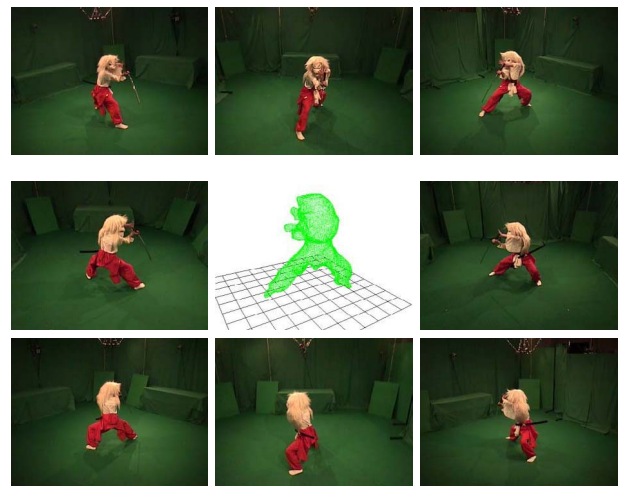


Figure 2. Archived appearance data: synchronized shapes and images ($t = 10$).

ence sometimes results in the generation of unnatural virtual views. In order to reduce the shape difference, we have refined the shape reconstruction method into a more sophisticated method that can take into account photometric consistency among visible cameras [18]. Thus, in our current implementation, the photometric consistency is confirmed to be sufficient for 3D shape generation.

The reconstructed shape is regarded as a human shape in this paper, although this is slightly different from a real shape. The reconstructed 3D shape is expressed using a set of micro cubes called voxels, and a surface representation is constructed for virtual view generation.

From the voxel representation, a surface normal is calculated for each surface voxel. In our current implementation, a surface normal is calculated by a simple plane fitting in the neighborhood of each surface voxel.

In this stage, a human shape is associated with a set of original images that are used for the 3D shape generation. Figure 2 shows how a human shape is associated with original images. For the archived data, we can treat the virtual view generation problem independently, without considering the views at other times.

3. Motion Registration and Coordination

3.1. Motion registration in dancer-centered coordinates

A motion tracker is also constructed in the multiple camera system, as shown in Fig.1. In order to stably track human motion, however, the dancer wears a special suit to which fifteen color markers are attached shown in Fig.3. Since all of the cameras have been calibrated and synchronized, the 3D position of each marker is calculated at

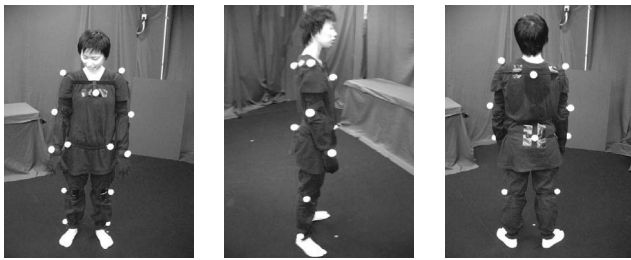


Figure 3. Special suit for motion tracking.

each time by a simple stereo recovery algorithm when color marker detection is successfully accomplished in each camera. When a color marker is observed from at least two cameras, the 3D marker position is estimated in the world-centered coordinate system. Thus, motion data are calculated for all of the markers in the world-centered coordinates.

In our current system, incomplete records are sometimes included among the motion data when some markers are observed from at most one camera due to occlusion. Such incomplete records are repaired by a human operator. After careful repair, the complete set of motion data is constructed.

Although the motion data are represented in world-centered coordinates, the coordinate system is not sufficient for attractive view generation. In the present paper, the representation is converted into dancer-centered coordinates in order to control the camera view in the coordinate system. For this purpose, a set of 3D positions at each time are converted into dancer-centered coordinates. Figure 4 shows 15 markers in the human structure. Since the dancer has a degree of freedom around a gravity vector, we define a body direction in the floor plane. In our current implementation, the body direction is simply defined by a triangle consisting of three markers, as shown by $(m_8, m_9$ and $m_{10})$ in Fig. 4. Although the dancer also has two degrees of freedom in translation on the floor, these are also estimated by the three markers. Thus, when positions of the three markers are estimated, the motion data can be represented in the dancer-centered coordinate system. Figure 5 shows two pairs of images of the dancer in the world-centered and dancer-centered coordinates, respectively. In the world-centered coordinates, the frontal direction of the dancer changes during the dance. However, the frontal direction of the dancer is maintained in front of the camera in the dancer-centered coordinates.

3.2. Coordination of appearance and motion

At this stage, the motion data are represented in the dancer-centered coordinates, while the appearance data are represented in the world-centered coordinates. If the appearance data can be converted into the dancer-centered co-

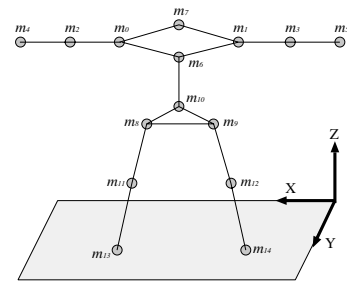


Figure 4. Fifteen markers for motion tracking.

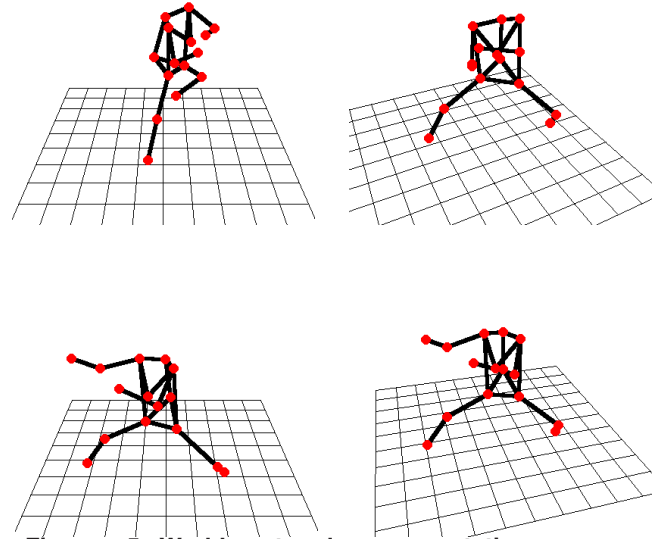


Figure 5. World-centered representation vs. dancer-centered representation.

ordinates, then camera control can be discussed in this set of coordinates. In order to solve this problem, we have to coordinate the appearance and motion data.

For the purpose, let us define an optimization problem at each time. We assume that appearance and motion data are approximately synchronized because they are taken from the same dance performed by the same professional dancer. Therefore, the coordination problem does not seem very difficult because the poses seem similar between the two data at each time. Therefore, we can assume that differences of the two data are only in rotation on the floor and translation on the floor. In order to adjust the rotation and translation, we define an objective function E between two data.

Let v denote a voxel position in the human shape V . Let m_i denote the position of the i -th marker in the motion data. Let R and t denote the rotation and translation of the motion data, respectively. Then, the coordination problem can be formalized as the minimization of the total distance defined by

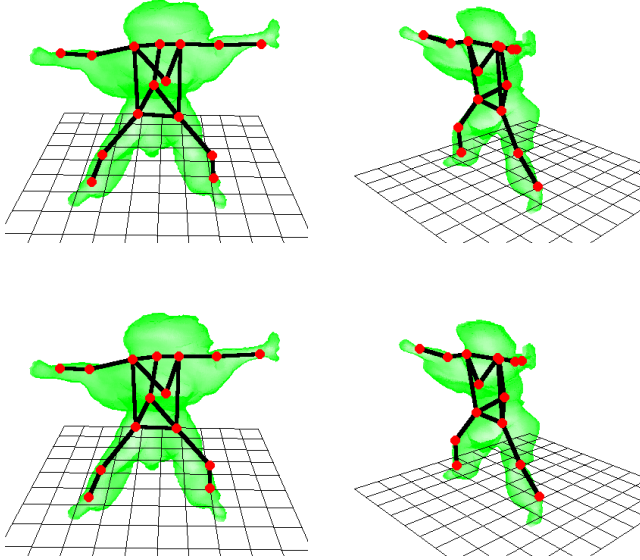


Figure 6. Coordination from the estimated pose of the previous frame.

$$E = \sum_{i=0}^{14} \rho \left(\min_{v \in V} \|\hat{m}_i - v\| \right) \quad (1)$$

where $\hat{m}_i = Rm_i + t$
and $\rho(x) = \frac{x^2}{x^2 + \sigma^2}$.

Here, $\rho(x)$ is a robust function that is introduced for suppressing the effects of noises caused by differences in dress and motion between the two data. We can coordinate the two types of data so as to minimize the objective function when an approximate pose is given for the two data. The optimization problem is solved by a hill climbing algorithm at each time. Since an estimated pose can be used as the initial pose for the next frame, a sequence of the coordination problems can also be solved easily. Figure 6 shows how the successive coordination is accomplished from a initial pose, as shown in the upper row, to a final estimated pose, as shown in the lower row.

4. Image generation from synchronized shape and images

4.1. Color blending approach using synchronized data

Generation of a virtual view from given multiple views has been widely discussed both for static and dynamic scenes. In the static domain, digital archiving of several

images of the great Buddha statue has been performed by Ikeuchi et al.[1, 3] and many 3D objects, including statues have been archived in the Michelangelo project[2]. In the dynamic domain, a number of technologies have been developed for mainly sports-related television programs[8, 10]. Similar attempts have also been discussed with respect to cooperative distributed vision[9, 11].

In this section, we discuss how to generate a virtual view of a traditional dance from archived dance data. Differences in the application domain results in differences in the qualities and sensitivities required for each application. In the sports domain, greater sensitivity is required for subject tracking in complex situations. On the other hand, a higher of naturalness is required for images of traditional dances compared to other applications. Therefore, this paper hereinafter focuses on naturalness.

Since each voxel is located in world coordinates, it is easy to calculate the position of the voxel image in an image plane when a projective transformation and a viewpoint are provided for a camera. However, The selection of a color for the voxel from a set of archived images is not easy, because voxels are often observed by several cameras simultaneously, and so appear to differ in color due to both photometric and geometric effects.

Of course, the shape reconstruction problem can be regarded in each time as a wide baseline stereo problem in the sparse camera configuration. However, the sparse stereo problem has not yet been solved; whereas the dense stereo class can provide a good approximation of 3D shapes[16, 17]. Our subjects are traditional dances that include various actions and artistic aspects. A fine shape reconstruction for such subject is still too difficult via sparse stereo. As such, an artistic quality that includes smoothness and naturalness should be required for virtual view generation.

For appropriate virtual view generation, we focus on appropriate color selection for each surface voxel. For the discussion of this problem, we define the following notation:

- surface voxel: s
- unit surface normal vector of s : N
- virtual viewpoint: eye
- camera: C_j ($1 \leq j \leq p$)
- observed color of s from C_j : I_j
- unit vector from C_j to s : V_j
- unit vector from eye to s : V_0

In order to simplify the problem, let us consider a 2D scenario in which all cameras, viewpoints, and voxels are located on a single plane. Figure 7 shows a 2D scenario in which the absolute angles of a surface normal and a viewpoint, denoted as θ_N and θ_{eye} , respectively, are specified by angles from the C_8 direction. These definitions are extended to the 3D case without modification.

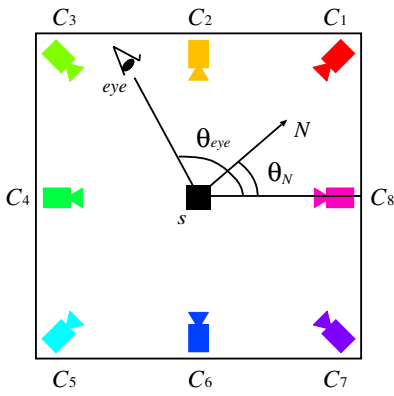


Figure 7. Top view of the system.

4.2. Linear combination of observed colors

Surface voxels often appear different when observed from multiple cameras. Observed color is dependent on several factors including specular reflections, occlusions, errors in 3D shape reconstruction, and the characteristics of the cameras. Precise pixel-by-pixel analysis of color differences seems too bulky.

A practical and effective solution is provided by color blending. In the color blending approach, voxel color is calculated by blending colors observed from multiple cameras without precise photometric/geometric analyses. In this approach, a weight parameter w_j is assigned to each camera C_j . Let us define w_j as

$$w_j = v_j f(N, V_j, V_0), \quad (2)$$

where v_j is a variable indicating the visibility of the voxel from camera C_j :

$$v_j = \begin{cases} 1: & s \text{ is visible from } C_j. \\ 0: & s \text{ is invisible from } C_j. \end{cases} \quad (3)$$

In Eq.(2), f is a function that takes three argument vectors, which represent a surface normal, a camera direction, and a viewpoint direction, respectively. A color I of a surface voxel s is calculated from the weight parameters and observed colors as

$$I = \frac{\sum_{j=1}^p w_j I_j}{\sum w}, \quad (4)$$

where $\sum w$ is a total sum of the weights, which works as a normalization factor. When the sum equals zero, the surface cannot be observed from any camera, and the surface voxel is ignored in the view generation.

In order to calculate an value of w_j , the function f must be designed appropriately. Several methods have been proposed for the function f , and these methods can be classified into three classes. In the first class, the function f is dependent on a surface normal relative to the camera position.

In the second class, the function is dependent on a virtual viewpoint relative to the camera position. In the third class, the function is dependent on both a surface normal and a virtual viewpoint.

4.3. Color blending based on surface normal

In the first class, color blending is performed due to relative camera positions with respect to a surface normal[9].

A typical function f is defined by the n -th power of the inner product of $-N$ and V_j as follows:

$$f(N, V_j, V_0) = \max((-N^T V_j)^n, 0), \quad (5)$$

where n is an odd integer that controls the degree of color blending. Since n is odd, Eq.(5) keeps f in 0 for any invisible camera C_j that satisfies $-N^T V_j < 0$.

4.4. Color blending based on virtual viewpoint

In the second class, color blending is performed due to relative camera positions with respect to a virtual viewpoint without using a surface normal.

A typical function f is proposed by Matsuyama and Takai[11] using the m -th power of the inner product of V_j and V_0 as:

$$f(N, V_j, V_0) = (V_j^T V_0)^m, \quad (6)$$

where m is a parameter that controls the degree of color blending.

Figure 8 shows the weight as a function of θ_{eye} in the eight-camera system shown in Fig.7 when f is defined by Eq.(6). In the figure, different cameras are shown in different colors, corresponding to those used in Fig.7.

Since the color blending method is independent of surface normal θ_N , the same weights are assigned to all of the cameras due to θ_{eye} , even if the surface normals are different. This results in advantages and disadvantages. The generated texture becomes smoother and more realistic as the number of effective cameras increases. The viewpoint-based method, however, often causes unnatural texture warps, especially at boundaries. If a surface is observed from a direction that is approximately perpendicular to the surface normal, the surface is almost invisible, and the texture is forcibly warped. This problem mainly occurs at the boundaries of the human body.

4.5. Color blending based on viewpoint and surface normal

A significant disadvantage of surface-normal-based methods is that the generated texture tends to be noisy, whereas the problem with viewpoint-based methods is that several unnatural warps occur at boundaries.

A new method is thus proposed by designing a function f using both a virtual viewpoint and a surface normal. In the

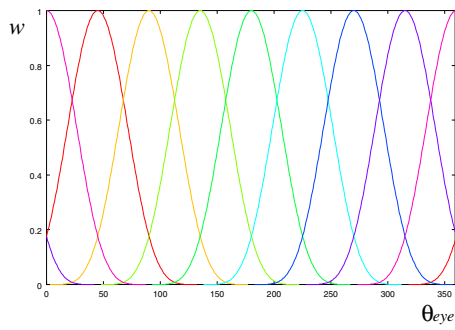


Figure 8. Weight as a function of θ_{eye} .

current method, f is defined as the product of two functions, f_{eye} and f_N , as follows:

$$f(N, V_j, V_0) = f_{eye}(V_0, V_j) f_N(N, V_j). \quad (7)$$

Here, the function f_{eye} must be large when the angle between V_0 and V_j is small, whereas the function f_N must be large when the angle between $-N$ and V_j is small.

A linear interpolation or an inner product, f_{eye} and f_N , could be used. In these cases, it is difficult to control the blending ratios among visible cameras. That is, only two colors are used in the linear interpolation, and too many colors are often used in the inner product type. Therefore, we choose a function that includes the m -th and n -th powers of inner products, as defined by:

$$f_{eye} = \max((V_j^T V_0)^m, 0), \quad (8)$$

$$f_N = \max((-N^T V_j)^n, 0), \quad (9)$$

where n and m are odd integers that control the degree of color blending. Since n is odd, Eq.(9) maintains f_N as 0 for any invisible camera C_j that satisfies $-N^T V_j < 0$. For symmetric operations between f_{eye} and f_N , a similar constraint is assumed as m . Only the parameters m and n remain to be tuned.

After some preliminary experiments, we have set both m and n to 5. If more cameras are used in the system, then m and n should be larger.

4.6. Comparison of color blending methods

Here, we compare the three types of color blending methods. Three-dimensional graphs in Figs. 9 and 10 show the weight parameters assigned to a given pair of θ_{eye} and θ_N when color blending is performed based on surface normals only, and viewpoints only, respectively. In the graphs, the graph domain specified by two angles, θ_N and θ_{eye} , are shown in the bottom plane, while the height indicates a weight assigned to a pair of angles. These graphs show that the weight parameters are decided by one of the two angles in these two methods. The color maps in the figures indicate

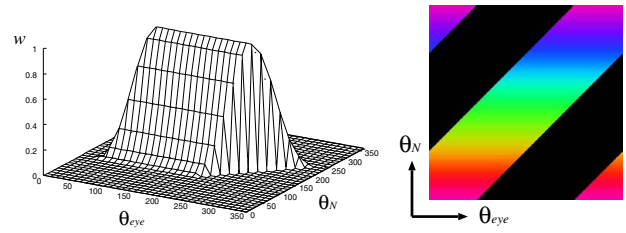


Figure 9. Weight based on surface normal.

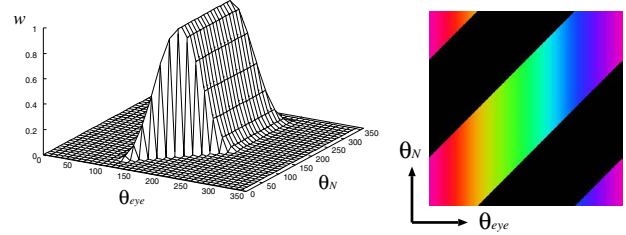


Figure 10. Weight based on viewpoint.

the dominant cameras in the (θ_{eye}, θ_N) domain using eight colors, which are specified in Fig.7 when eight cameras are configured as shown in the figure.

Figure 11 shows the weight parameters and dominant cameras in the (θ_{eye}, θ_N) domain when the function f is specified by Eqs.(7) - (9) while setting $n = m = 5$. Since the same values are selected for n and m , the dominant camera changes along a diagonal line, $\theta_N = \theta_{eye}$. This shows that the proposed method takes both a virtual viewpoint and a surface normal into account, and the cooperative use thereof effectively eliminates the disadvantages of the surface-normal-based method and the viewpoint-based method, respectively.

5. Virtual View Generation

5.1. 'ARAMAI' data

In order to evaluate the effectiveness of the proposed method, a studio including eight cameras (SONY DXC-200A) and eight PCs was constructed to capture image sequences. The observed space is roughly a $2m \times 2m \times 2m$ cube. Image sequences are captured synchronously by the cameras at a video rate of 30 fps and an image size of 640×480 . Since the width of one pixel corresponds to $5mm$ at the center of the room, the voxel size is set to $5mm \times 5mm \times 5mm$. In order to easily distinguish the human region from the background region, the floor and walls were covered by green cloth.

As an example of an intangible cultural treasure, the Japanese traditional dance 'ARAMAI' was captured. Since the dance is very fast, the shapes of the hair and the costume change drastically and rapidly. Figure 2 shows an example

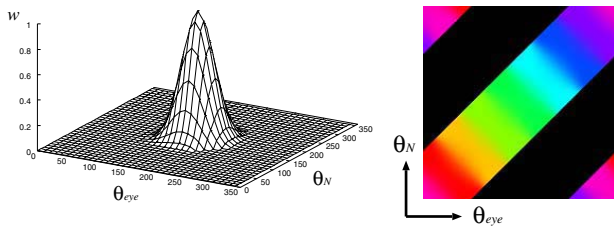


Figure 11. Weight based on both surface normal and viewpoint (proposed method).

of archived dance data at $t = 10$ (where t is the frame number).

5.2. Comparison of generated images

Figure 12 shows images generated for $\theta_{eye} = 45^\circ$ at $t = 0$. The second and third rows of the images show enlargements of the rectangular regions in the first row of images. The last row of images shows camera weightings for each surface point by color. The left-hand column shows the results generated using the color blending method which is based only on surface normal. The center column shows the results based only on viewpoint (5th power of the inner product). Finally, the right-hand column shows the results generated using the proposed method ($m = 5, n = 5$). In the left-hand images, the unnatural texture appears in the face region. In the center images, the dark texture appears in the boundaries between a human region and a background region. The quality of the texture are improved in right-hand images.

5.3. Virtual exhibition

Figure 13 shows an example of a generated movie with a changing virtual viewpoint. In this example, the camera position is moved at a constant speed around the dancer. We have confirmed that the position of the virtual camera does not affect the quality of the generated images.

Figure 14 shows an example of a generated movie in which the virtual camera position is controlled as if the camera were located in front of the dancer. Although the automatic coordination is somewhat rough, we can synthesize a very useful camera movement in the dancer-centered coordinates.

6. Conclusions

A novel method was proposed for virtual view generation of traditional dances. In the proposed framework, a traditional dance is captured separately for appearance registration and motion registration. By coordinating the appear-

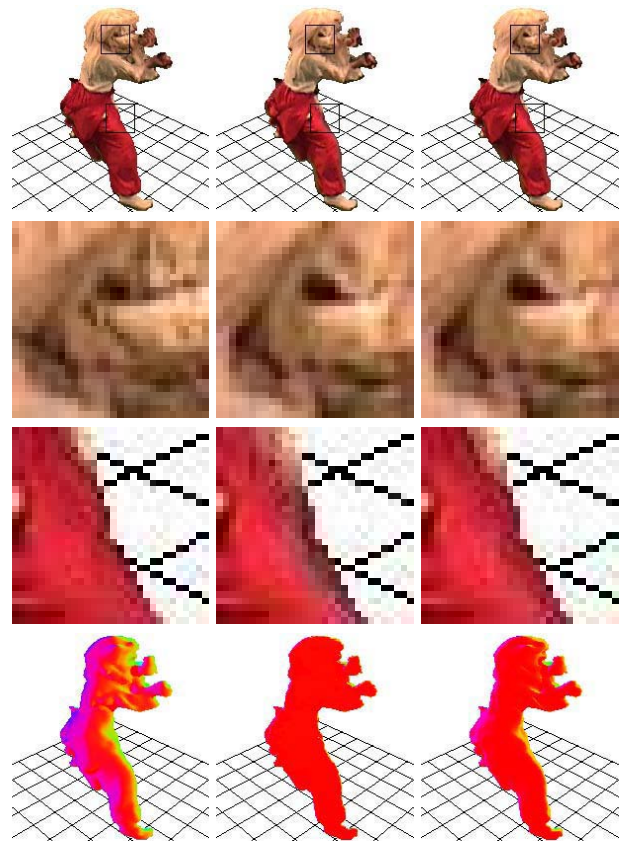


Figure 12. Comparison of generated images ($\theta_{eye} = 45^\circ, t = 0$).

ance and motion data, we can easily control camera motion in the dancer-centered coordinates. For this purpose, an automatic coordination algorithm for the appearance and motion data was proposed. In addition, we have provided a set of algorithms for appearance and motion registration, and virtual view generation from archived data.

Acknowledgment

The authors would like to thank WARABI-ZA (<http://www.warabi.jp/english/index.html>) for their kind cooperation in the capture of the 'ARAMAI' dance. This work has been supported in part by Japan Science and Technology Corporation under Ikeuchi CREST project and by Grant-In-Aid for Science Research under Grant No.15300062 from the Ministry of Education, Science, Sports, and Culture, Japanese Government.

References

- [1] D. Miyazaki, T. Ooishi, T. Nishikawa, R. Sagawa, K. Nishino, T. Tomomatsu, Y. Takase and K. Ikeuchi: "The Great Buddha Project: Modeling Cultural Heritage through Observation", Proc. the Sixth International Conference on Virtual Systems and Multi Media (VSMM 2000), pp.138-145, 2000.

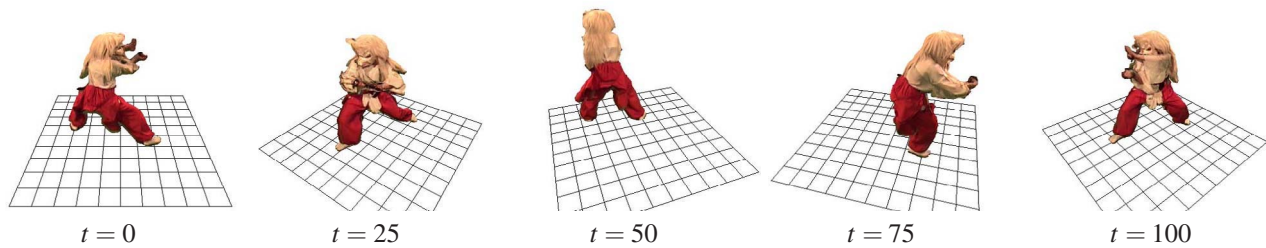


Figure 13. Movie generated by a virtual camera moving around the dancer.

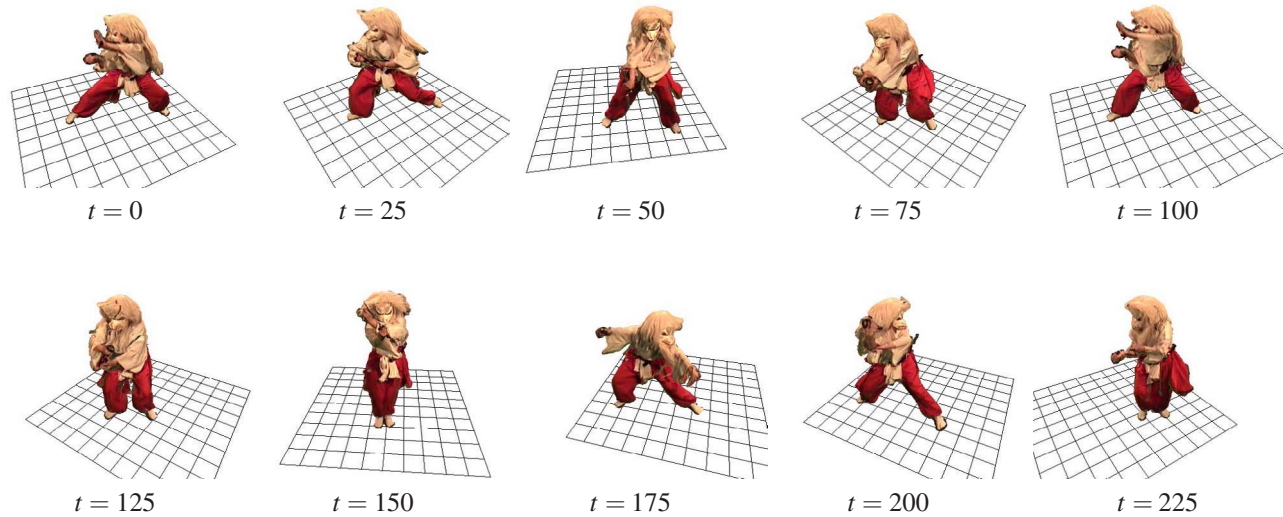


Figure 14. Movie generated by a virtual camera in front of the dancer.

- [2] M. Levoy, et al.: "The Digital Michelangelo Project: 3D Scanning of Large Statues", Proc. SIGGRAPH 2000, pp.131-144, 2000.
- [3] R. Sagawa, K. Nishino and K. Ikeuchi: "Robust and Adaptive Integration of Multiple Range Images with Photometric Attributes", Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2001, vol.II, pp.172-179, 2001.
- [4] T. Kanade, P. Rander and P. J. Narayanan: "Virtualized Reality: Constructing Virtual Worlds from Real Scenes", IEEE Multimedia, Vol.4, No.1, pp.34-47, 1997.
- [5] J. Starck and A. Hilton: "Model-based multiple view reconstruction of people", IEEE International Conference on Computer Vision, pp.915-922, 2003.
- [6] J. Carranza, C. Theobalt, M. Magnor and H.-P. Seidel: "Free-Viewpoint Video of Human Actors," ACM Trans. on Computer Graphics, vol. 22, no. 3, pp 569-577, 2003.
- [7] S. Moezzi, L. C. Tai and P. Gerard: "Virtual View Generation for 3D Digital Video", IEEE Multimedia, Vol.4, No.1, pp.18-26, 1997.
- [8] I. Kitahara, H. Saito, S. Akimichi, T. Ono, Y. Ohta and T. Kanade: "Large-scale Virtualized Reality", Proc. CVPR2001, Technical Sketches, 2001.
- [9] T. Takai and T. Matsuyama: "Interactive Viewer for 3D Video", Proc. Fourth International Workshop on Cooperative Distributed Vision, pp.475-494, 2001.
- [10] H. Saito, S. Baba, M. Kimura, S. Vedula and T. Kanade: "Appearance-Based Virtual View Generation of Temporally-Varying Events from Multi-Camera Images in the 3D Room", Proc. Second International Conference on 3-D Digital Imaging and Modeling (3DIM'99), pp.516-525, 1999.
- [11] T. Matsuyama and T. Takai: "Generation, Visualization, and Editing of 3D Video", Proc. Symposium on 3D Data Processing Visualization and Transmission, pp.234-245, 2002.
- [12] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler and L. McMillan: "Image-Based Visual Hulls", Proc. SIGGRAPH 2000, pp.369-374, 2000.
- [13] A. Laurentini: "The Visual Hull Concept for Silhouette-Based Image Understanding", IEEE Trans. PAMI, Vol.16, No.2, pp.150-162, 1994.
- [14] S. M. Seitz and C. R. Dyer: "View Morphing", Proc. SIGGRAPH'96, pp.21-30, 1996.
- [15] S. E. Chen and L. Williams: "View Interpolation for Image Synthesis", Proc. SIGGRAPH'93, pp.279-288, 1993.
- [16] M. Okutomi and T. Kanade: "A Multiple-Baseline Stereo", IEEE Trans. PAMI, Vol.15, No.4, pp.353-363, 1993.
- [17] T. Sato, M. Kanbara, N. Yokoya and H. Takemura: "Dense 3-D Reconstruction of an Outdoor Scene by Hundreds-baseline Stereo Using a Hand-held Video Camera", International Journal of Computer Vision, Vol.47, No.1-3, pp.119-129, 2002.
- [18] K. N. Kutulakos and S. M. Seitz: "A Theory of Shape by Space Carving", International Journal of Computer Vision, Vol.38, No.3, pp199-218, 2000.