Color blending based on viewpoint and surface normal for generating images from any viewpoint using multiple cameras

Yasuhiro Mukaigawa
University of Tsukuba

Daisuke Genda
Okayama University

Ryo Yamane
Okayama University

Takeshi Shakunaga
Okayama University

5-3

Color Blending based on Viewpoint and Surface Normal for Generating Images from Any Viewpoint using Multiple Cameras

Yasuhiro Mukaigawa *, Daisuke Genda, Ryo Yamane and Takeshi Shakunaga
Department of Information Technology, Okayama University, JAPAN

Abstract

A color blending method for generating a high quality image of human motion is presented. The 3D human shape is reconstructed by volume intersection and expressed as a set of voxels. As each voxel is observed as different colors from different cameras, voxel color needs to be assigned appropriately from several colors. We present a color blending method which calculates voxel color from a linear combination of the colors observed by multiple cameras. The weightings in the linear combination are calculated based on both viewpoint and surface normal. As surface normal is taken into account, the images with clear texture can be generated. Moreover, since viewpoint is also taken into account, high quality images free of unnatural warping can be generated. To examine the effectiveness of the algorithm, a traditional dance motion was captured and new images were generated from arbitrary viewpoints. Compared to existing methods, quality at the boundaries was confirmed to be improved.

1 Introduction

Recently, much research has been conducted into preserving cultural treasures in digital archives[1, 2]. In particular, intangible cultural treasures, such as traditional dance, are difficult to preserve because a complete archiving method has not been developed. To capture human motion, a marker-based motion capture system is often used. This system can precisely measure the 3D position of markers, and the measured data can be used in many applications, including computer graphics and motion analysis.

Although costumes and ornaments are also part of intangible cultural treasures, this visual information cannot be captured by ordinary motion capture systems. If the 3D visual information of dance motion was archived, the viewpoint could be controlled arbitrarily. Control of the viewpoint would be important for both digital archiving and dance training.

To generate a new image which can be observed from an arbitrary viewpoint, many methods using multiple cameras have already been proposed[3, 4, 5]. A 3D model of the human body is reconstructed based on images, and the surface texture is mapped onto the 3D model. While each point on the human body is observed from several cameras, the colors are not same. Therefore, an appropriate surface color needs to be calculated from several colors. To reproduce a detailed texture, calculation methods based on the surface normal have been proposed[6]. However, these methods tend to be noisy because the camera from which the color is taken is different for each voxel. Other calculation methods based on viewpoint have been proposed to improve the quality of images generated[7, 8]. These methods, however, often cause unnatural warping of texture, especially at boundaries.

In this paper, we propose a new color blending method based on both viewpoint and surface normal. As this approach does not produce unnatural warping at boundaries, the quality of generated images is improved.

2 Image generation

2.1 Multiple camera system

In this paper, we assume that a target person dances at the center of a room, and that multiple cameras are installed to capture image sequences from various angles, as shown in Fig. 1. The image sequences are synchronized, with all cameras geometrically calibrated in advance.

2.2 Purpose

In this paper, we do not treat the 3D reconstruction problem. We assume that the 3D shape is reconstructed by the volume intersection method. The reconstructed shape, known as the *visual hull* [9, 10], is larger than the true shape. We regard the *visual hull* as the human body.

The 3D shape is expressed using a set of micro cubes called voxels. Since each voxel contains 3D information, it is easy to geometrically calculate the 2D position in a new image from an arbitrary viewpoint. However, it is not easy to decide the color of the voxel,

*Current affiliation is Institute of Engineering Mechanics and Systems, University of Tsukuba, JAPAN

0-7803-7987-X/03/$17.00 ©2003 IEEE.

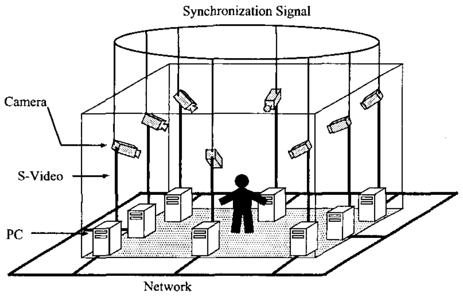


Figure 1: Multiple camera system.

because each voxel is observed from several cameras and the observed colors are usually different. Our aim is to decide an appropriate color for each surface voxel.

In this paper, the following notation is used:

- surface voxel: s_i
- unit surface normal vector of s_i: N_{s_i}
- virtual viewpoint: *eye*
- camera: C_j $(1 \leq j \leq p)$
- observed color of s_i from C_j: I_{C_j,s_i}
- unit vector from C_j to s_i: $V_{C_j \rightarrow s_i}$
- unit vector from *eye* to s_i: $V_{eye \rightarrow s_i}$

To make the problem easier, 2D conditions in which all cameras, viewpoint, and voxels are on a single plane is considered, as shown in Fig. 2. In this system, absolute angles of surface normals and the viewpoint are specified by θ_N and θ_{eye}, respectively. Of course, this can easily be extended to the 3D case.

2.3 Linear combination of observed colors

Although each surface voxel is observed by several cameras, the colors are not the same because of complex factors including specular reflections, occlusions, errors in 3D shape reconstruction, and characteristics of the cameras. Therefore, it is difficult to analyze color differences precisely.

In our method, voxel color is calculated by color blending instead of analysis by assigning a weight parameter w_j to each camera C_j, with the weight parameter given by:

$$w_j = v_{C_j \rightarrow s_i} \cdot f(N_{s_i}, V_{C_j \rightarrow s_i}, V_{eye \rightarrow s_i}), \qquad (1)$$

where $v_{C_j \rightarrow s_i}$ is a variable which specifies the visibility of the voxel as follows:

$$v_{C_j \rightarrow s_i} = \begin{cases} 1 : & s_i \textit{ is visible from } C_j. \\ 0 : & s_i \textit{ is invisible from } C_j. \end{cases} \qquad (2)$$

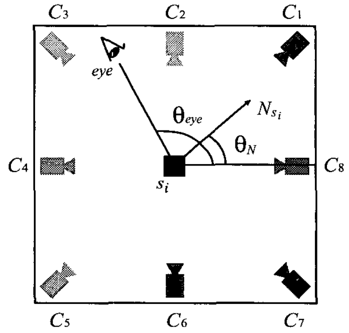


Figure 2: Top view of the system.

The f is a function which takes arguments of surface normal, camera direction, and viewpoint. Using the weightings, the color I_{s_i} of the surface voxel s_i is calculated by

$$I_{s_i} = \sum_{j=1}^{p} \frac{w_j}{\sum w} \cdot I_{C_j,s_i}, \qquad (3)$$

where $\sum w$ is a normalization factor equal to the sum of the weights. If the sum is zero, the surface cannot observed from any camera, and so this case can be ignored.

2.4 Existing methods

To decide the value of w_j, the function f needs to be appropriately designed. Existing methods can be classified into two categories, surface normal-based and viewpoint-based. Details of these algorithms are given in the following sections.

2.4.1 Calculations based on surface normal

To acquire detailed textural information about an object, an image should be taken from the surface normal, because the surface appears largest from this direction. Weightings are then calculated by the following function based on surface normal[6].

$$f(N_{s_i}, V_{C_j \rightarrow s_i}, V_{eye \rightarrow s_i}) = \begin{cases} 1 & : \textit{if } N_{s_i} \cdot V_{C_j \rightarrow s_i} \textit{ is the smallest.} \\ 0 & : \textit{otherwise} \end{cases} \qquad (4)$$

Clearly, this function does not depend on the viewpoint θ_{eye}. Although not blurred, the texture generated using this function tends to be noisy, because the camera from which the color is taken is different for each voxel.

2.4.2 Calculations based on viewpoint

To improve the quality of the generated textures, the camera nearest to the viewpoint should be selected. If the angle between the camera and the viewpoint is small, forced warping of the texture can be avoided.

To assign large weightings to the nearest cameras, interpolation methods are often used[11, 12]. The function f is thus taken to be the linear interpolation of the colors from the two cameras C_l and C_r as follows[7]:

$$f(N_{s_i}, V_{C_j\to s_i}, V_{eye\to s_i}) = \begin{cases} \alpha & : C_l \\ 1-\alpha & : C_r \\ 0 & : otherwise \end{cases} \tag{5}$$

where α $(0 \leq \alpha \leq 1)$ is a parameter representing the position of the viewpoint.

As an alternative, Matsuyama and Takai[8] have proposed a calculation method using the m-th power of the inner product of $V_{C_j\to s_i}$ and $V_{eye\to s_i}$ as follows:

$$f(N_{s_i}, V_{C_j\to s_i}, V_{eye\to s_i}) = (V_{C_j\to s_i} \cdot V_{eye\to s_i})^m, \tag{6}$$

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

Figures 3 (a) and (b) show weightings as a function of θ_{eye} as determined by linear interpolation and by the m-th power of the inner product, respectively. In the figure, different cameras are shown in different colors, as shown in Fig.2.

Clearly, functions based on viewpoint are independent of surface normal θ_N, with the same weightings assigned to all voxels even if the surface normals are different. Hence, the generated texture is smooth, and as the number of cameras increases, the texture becomes more realistic.

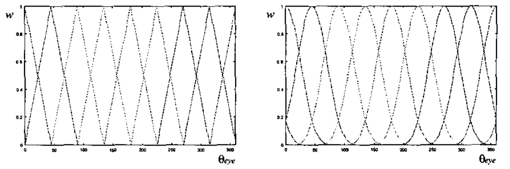
These functions, however, often cause unnatural warping of textures, especially at boundaries. If a surface is observed from the perpendicular to the surface normal, the surface is almost invisible, and this texture is forcedly warped. This problem mainly occurs at boundaries.

2.5 Calculations based on both viewpoint and surface normal

The weak point of surface normal-based methods is that the generated texture tends to be noisy, whereas the weak point of viewpoint-based method is that unnatural warping occurs at boundaries.

A new function is thus proposed based on both viewpoint and surface normal. In this method, the function f can be expressed as the product of two functions, f_{eye} and f_N as follows:

$$f(N_{s_i}, V_{C_j\to s_i}, V_{eye\to s_i}) = f_{eye}(V_{eye\to s_i}, V_{C_j\to s_i}) \times f_N(N_{s_i}, V_{C_j\to s_i}). \tag{7}$$



(a) (b)

Figure 3: Weightings as a function of θ_{eye}. (a) Linear interpolation of two cameras. (b) m-th power of the inner product $(m = 5)$.

The function f_{eye} needs to be large when the angle between $V_{eye\to s_i}$ and $V_{C_j\to s_i}$ is small, whereas the function f_N needs to be large when the angle between N_{s_i} and $V_{C_j\to s_i}$ is large.

Although a linear interpolation or an inner product could be used, it is difficult to control the degree of color blending. Only two colors are used in the linear interpolation, but too many colors are used in the inner product. Therefore, m and n-th powers of inner products are chosen, giving:

$$f_{eye} = (V_{C_j\to s_i} \cdot V_{eye\to s_i})^m, \tag{8}$$

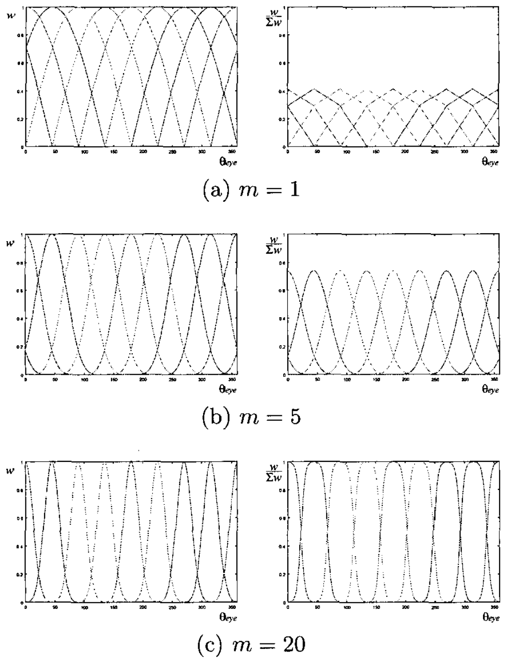
$$f_N = (-N_{s_i} \cdot V_{C_j\to s_i})^n, \tag{9}$$

where only the parameters m and n remain to be decided. Figure 4 shows the behaviors of the function for values of m of 1, 5 and 20. Values of w_j are shown in the left set of graphs, and normalized values are shown in the right. If m is small, too many colors are blended and the generated texture will be blurred, whereas if m is large, a single color is used primarily. Thus m has a large impact on texture, and both m and n are set to 5 when the angle between cameras is 45°. If there are more cameras, the m and n should be smaller.

Graphs on the right of figures 5, 6, 7 and 8 show the weightings assigned to each camera for a variety of different methods. Graphs on the left show the weighting of a single camera, w_4. The right color maps indicate the weightings of cameras by color. The relationship between color and camera is shown in Fig.2. The proposed method can be seen to take both viewpoint and surface normal into account.

3 Experimental results

To examine the effectiveness of the proposed method, a studio including eight cameras (SONY DXC-200A) and eight PCs was constructed to capture



(a) $m = 1$

(b) $m = 5$

(c) $m = 20$

Figure 4: Weightings calculated from the m-th power of the inner product.

image sequences. The observing area is a $2m \times 2m \times 2m$ cube. Image sequences are captured synchronously by the cameras at video rate (30 fps) with an image size of 640 × 480. Since the width of one pixel corresponds to $5mm$ at the center of the room, voxel size is set to $5mm \times 5mm \times 5mm$. To easily separate human region from the background, the floor and walls were covered by green cloth.

As an archive of intangible cultural treasures, a Japanese traditional dance 'ARAMAI' was captured. Because the dance is very fast, hair and costume shapes change drastically. Figure 9 shows an example of input images at $t = 10$ (where t is the frame number).

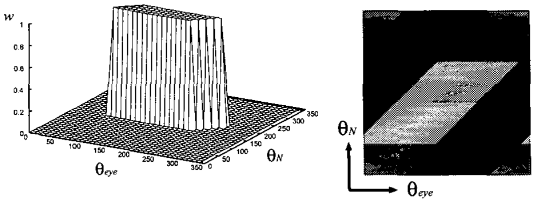


Figure 5: Weightings based on surface normal.

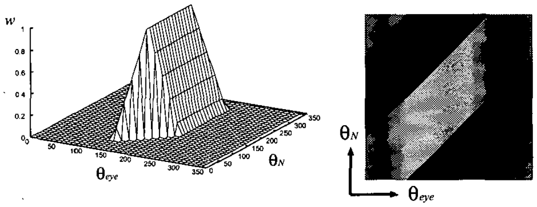


Figure 6: Weightings based on viewpoint (linear interpolation).

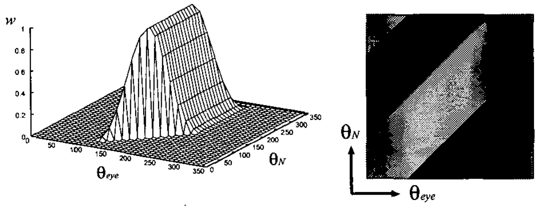


Figure 7: Weightings based on viewpoint (5th power of the inner product).

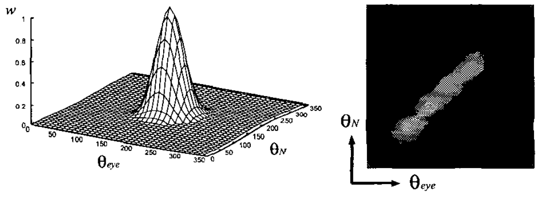


Figure 8: Weightings based on both surface normal and viewpoint (proposed method).

Figures 10 and 11 show images generated for $\theta_{eye} = 0°$ at $t = 10$ and for $\theta_{eye} = 50°$ at $t = 244$, respectively. The middle row shows enlargements of the rectangle regions in the upper row of images. The lower row shows camera weightings for each surface point by color. The left column shows results generated using the existing method based only on surface normal, the center column shows results based only on viewpoint (the 5th power of the inner product), and the right column shows results generated using the proposed method ($m = 5$, $n = 5$). The quality of the texture can be seen to be improved at the boundaries. Figure 12 shows examples of generated movie with changing virtual viewpoint. Finally, a generated image was compared to a real image for numerical evaluation. An additional camera was installed and a

Figure 9: Example of input images ($t = 10$).

real image was captured, as shown in Fig.13(a). The virtual viewpoint was then set to be the same as the additional camera. Images captured by the additional camera were not used for image generation. Figures 13(b), (c) and (d) show results of the surface normal-based method, the viewpoint-based method and the proposed method, respectively. For numerical evaluation, differences between the real image and the generated images were calculated, as shown in (e), (f) and (g). RGB root mean square errors of each method are shown next to the images. Comparison of (e) and (g) reveals that errors are reduced by the proposed method. Further comparison of (f) and (g) reveals that the proposed method can improve the quality of texture at boundaries.

4 Conclusions

In this paper, a method for generating high quality images by blending the colors observed from multiple cameras was proposed in which weightings of the cameras were calculated based on both viewpoint and surface normal. By changing the exponent parameters m and n, the degree of dependence on viewpoint and surface normal can be controlled. That is, the proposed method is based on existing methods.

To examine the effectiveness of the proposed method, 3D motion and visual information of the Japanese traditional dance 'ARAMAI' was captured using eight cameras. The quality of images generated from arbitrary viewpoints are obviously improved over existing methods.

In this paper, only the color blending problem was examined with neither color calibration between cameras nor 3D reconstruction examined. However, these problems need to be solved exhaustively. In the future, we intend to analyze differences in colors, and improve the quality of output images.

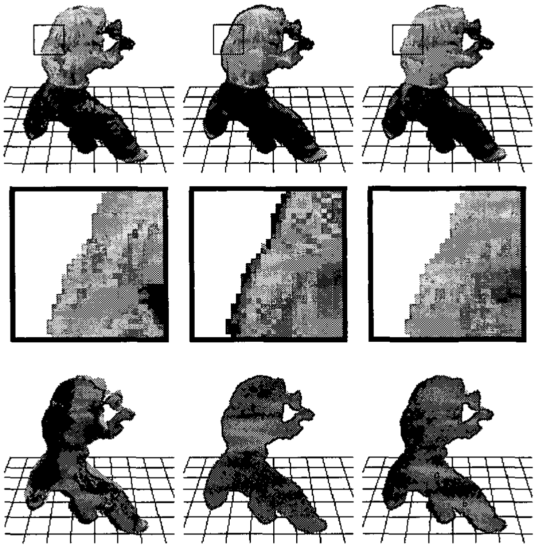


Figure 10: Comparison of generated images ($\theta_{eye} = 0°$, $t = 10$).

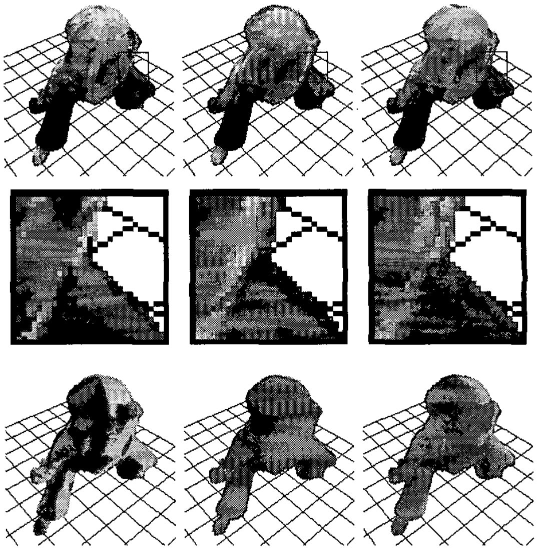
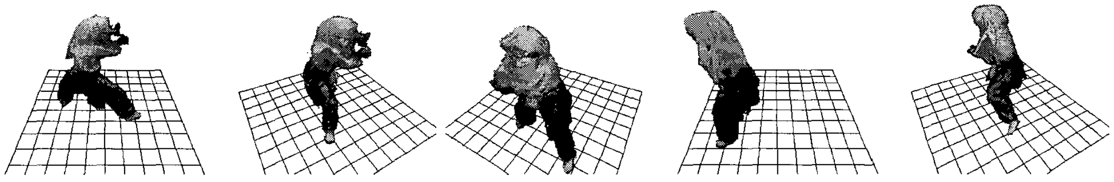
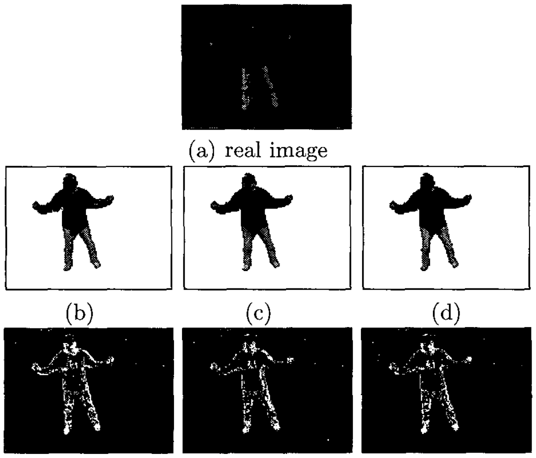


Figure 11: Comparison of generated images ($\theta_{eye} = 50°$, $t = 244$).



$t = 0$ $t = 20$ $t = 40$ $t = 60$ $t = 80$

Figure 12: Generated movie by the proposed method.



(a) real image

(b) (c) (d)

(e) 50.3 (f) 40.6 (g) 40.2

Figure 13: Evaluation. (a) real image. (b),(c) and (d) reconstructed images. (e),(f) and (g) differences between reconstructions and the real images, with RGB RMS differences also shown.

Acknowledgments

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 research has been supported by Japan Science and Technology Corporation under Ikeuchi CREST project.

References

[1] D. Miyazaki et al.: "The Great Buddha Project: Modeling Cultural Heritage through Observation", Proc. the Sixth International Conference on Virtual Systems and MultiMedia (VSMM 2000), pp.138-145, 2000.

[2] M. Levoy et al.: "The Digital Michelangelo Project: 3D Scanning of Large Statues", Proc. SIGGRAPH 2000, pp.131-144, 2000.

[3] 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.

[4] 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.

[5] I. Kitahara, H. Saito, S. Akimichi, T. Ono, Y. Ohta and T. Kanade: "Large-scale Virtualized Reality", Proc. CVPR2001, Technical Sketches, 2001.

[6] T. Takai and T. Matsuyama: "Interactive Viewer for 3D Video", Proc. Fourth International Workshop on Cooperative Distributed Vision, pp.475-494, 2001.

[7] 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.

[8] 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.

[9] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler and L. McMillan: "Image-Based Visual Hulls", Proc. SIGGRAPH 2000, pp.369-374, 2000.

[10] A. Laurentini: "The Visual Hull Concept for Silhouette-Based Image Understanding", IEEE Trans. PAMI, Vol.16, No.2, pp.150-162, 1994.

[11] S. M. Seitz and C. R. Dyer: "View Morphing", Proc. SIGGRAPH'96, pp.21-30, 1996.

[12] S. E. Chen and L. Williams: "View Interpolation for Image Synthesis", Proc. SIGGRAPH'93, pp.279-288, 1993.