Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2009, Article ID 346425, 9 pages
doi:10.1155/2009/346425

Research Article

Motion Segmentation for Time-Varying Mesh Sequences Based on Spherical Registration

Toshihiko Yamasaki and Kiyoharu Aizawa

Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo, Engineering Building no. 2, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113 8656, Japan

Correspondence should be addressed to Toshihiko Yamasaki, yamasaki@hal.t.u-tokyo.ac.jp

Received 30 September 2007; Accepted 7 March 2008

Recommended by Thomas Sikora

A highly accurate motion segmentation technique for time-varying mesh (TVM) is presented. In conventional approaches, motion of the objects was analyzed using shape feature vectors extracted from TVM frames. This was because it was very difficult to locate and track feature points in the objects in the 3D space due to the fact that the number of vertices and connection varies each frame. In this study, we developed an algorithm to analyze the objects' motion in the 3D space using the spherical registration based on the iterative closest-point algorithm. Rough motion tracking is conducted and the degree of motion is robustly calculated by this method. Although the approach is straightforward, much better motion segmentation results than the conventional approaches are obtained by yielding such high precision and recall rates as 95% and 92% on average.

1. Introduction

Three-dimensional (3D) geometric modeling of human appearance and motion based on computer vision techniques (i.e., using only multiple cameras) [1–7] is getting much more attention as ultimate interactive multimedia. Although 3D scene generation based on image-based rendering (IBR) [8–16] is also very popular because a scene from imaginary cameras can be obtained very fast without estimating the 3D shape of the objects, 3D geometric modeling has some attractive features: (1) the number of cameras is much smaller than that in IBR, (2) 3D models can be seen from any view points and provide us "more" free view-point video than IBR, (3) it is compatible with augmented reality (AR) technology, and so on.

The idea of 3D modeling of real-world objects using silhouettes in multiple images was first introduced in 1974 by Baumgart [17]. Then, capturing dynamics and motion of human in the form of 3D mesh was popularized by Kanade et al. [1]. Since then, some more systems have been developed aiming at real-time modeling [2, 3], high-resolution, and high-quality modeling using deformable mesh [4] or stereo matching [5, 6]. The consecutive sequences of 3D models (frames) are often called "3D video." There are some variations in 3D video data structure. 3D video discussed in this paper is defined as sequential 3D mesh models composed of three kinds of data such as position of vertices, their connection, and color of each vertex. Hereafter, we call such data as time-varying mesh (TVM). In contrast with computer-graphics based 3D mesh animation called dynamic mesh or dynamic geometry, one of the most important features in TVM is that the number of vertices and topology changes every frame due to the nonrigid nature of human body and clothes. Namely, each frame is generated independently regardless of its neighboring frames. This makes data processing for TVM much more challenging.

Since TVM is still an emerging technology, most of the papers reported so far other than capturing systems are on compression to remove temporal redundancy [18–20]. However, as the amount of TVM data increases, the development of efficient and effective content management of the database will be required such as indexing, summarization, retrieval, and editing. In this regard, the authors have been developing key techniques for those purposes such as motion segmentation [21, 22], key frame extraction [23], content-based retrieval [24, 25], and editing [26]. Other applications from other groups can also be found in [27, 28].

Motion segmentation, in particular, is one of the important preprocessing for efficient content management [29–35]. Motion segmentation, which is also called temporal segmentation, is a process to divide the whole sequence into small but meaningful and manageable clips based on the object's motion. The segmented TVM clips are handled as minimum units for indexing, retrieval, and editing. One of the challenging problems in motion segmentation for TVM is that feature points are difficult to locate and track due to the unregularized number of vertices and connection as discussed above. Therefore, in [21, 22], some vectors representing some shape features were generated and the motion was analyzed in the feature vector spaces. In [21], distances among vertices of a 3D model and predefined three reference points were calculated to form a distance histogram. However, the three reference points were defined by an empirical study and how to set the proper reference points is still an open question. In [22], another feature representation called modified shape distribution was developed and segmentation was conducted by searching for local minima in the degree of motion. Searching for local minima in kinematical parameters was a reasonable approach because motion speed decreases for a moment when the motion type or the motion direction changes. This idea has also been employed in temporal segmentation for 2D video [29] and motion capture data [30, 31]. Although motion analysis using shape feature vectors extracted from 3D mesh models was computationally efficient, it was prone to miss- and over-segmentation. As discussed in [25], high-level and detailed motion analysis is required for more accurate processing.

The purpose of this paper is to present a technique to analyze the objects' motion not in the feature vector space, but in the 3D space for more accurate motion segmentation. In our approach, the iterative closest-point (ICP) algorithm [36] is employed for spherical registration between neighboring TVM frames, and rough motion tracking is achieved for calculating the degree of motion. The motion segmentation strategy was employed from our previous approach [22]. Experimental results using five TVM sequences of dances demonstrated that the precision and recall rates were improved up to 95% and 92%, respectively. In addition, some preliminary results for motion retrieval using the same technology are also presented in this paper. Although the algorithms for motion segmentation and motion retrieval are very similar to the authors' previous works, the contribution of this paper is the similarity evaluation method among the TVM frames for more accurate processing.

The rest of the paper is organized as follows. In Section 2, the detailed data description of TVM is given. In Section 3, the algorithms for dissimilarity measure among frames, motion segmentation, and similar motion retrieval are explained. Section 4 demonstrates the experimental results and concluding remarks are given in Section 5.

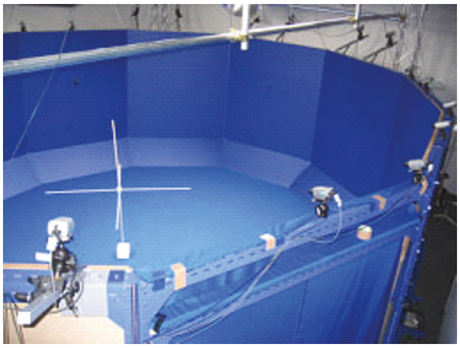


Figure 1: Studio for TVM generation.

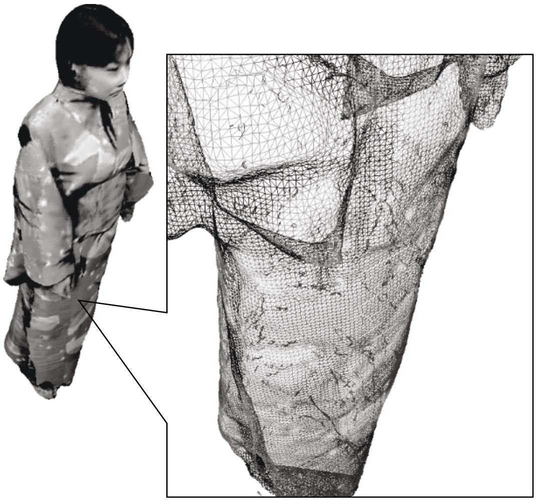


Figure 2: Example frame of our TVM data. Each frame is described in a VRML format and consists of coordinates of vertices, their connection, and color.

2. Data Description

The TVM data in the present work were obtained by courtesy of Tomiyama et al. [5]. They were generated from multiple-view images taken with 22 synchronous cameras installed in a dedicated blue-back studio with 8 m in diameter and 2.5 m in height. The studio is shown in Figure 1. The 3D object modeling is based on the combination of the volume intersection and the stereo matching [5].

Similar to 2D video, TVM is composed of a consecutive sequence of "frames." Each frame of TVM is represented as a 3D polygon mesh model. Namely, each frame is expressed by three kinds of data as shown in Figure 2: coordinates of vertices, their connection (topology), and color. The spatial resolution of the models is 5–10 mm; and, the number of vertices is from 17,000 to 50,000 depending on the spatial resolution. The number of connection data is about double the number of vertices as is the case with other 3D mesh models.

The most significant feature in TVM is that each frame is generated regardless of its neighboring frames. This is because of the nonrigid nature of human body and clothes.

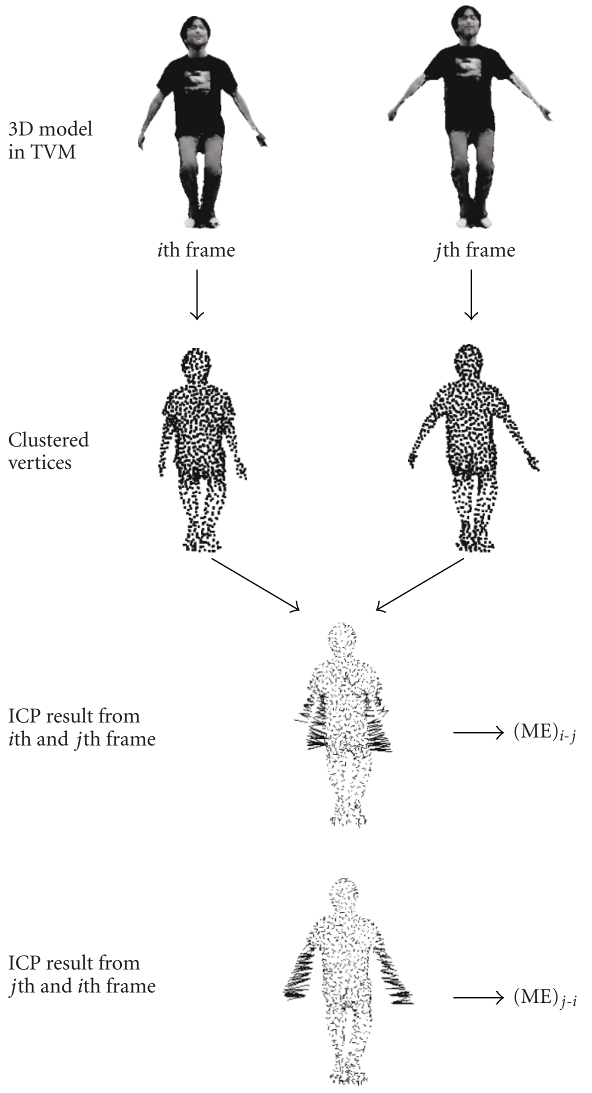
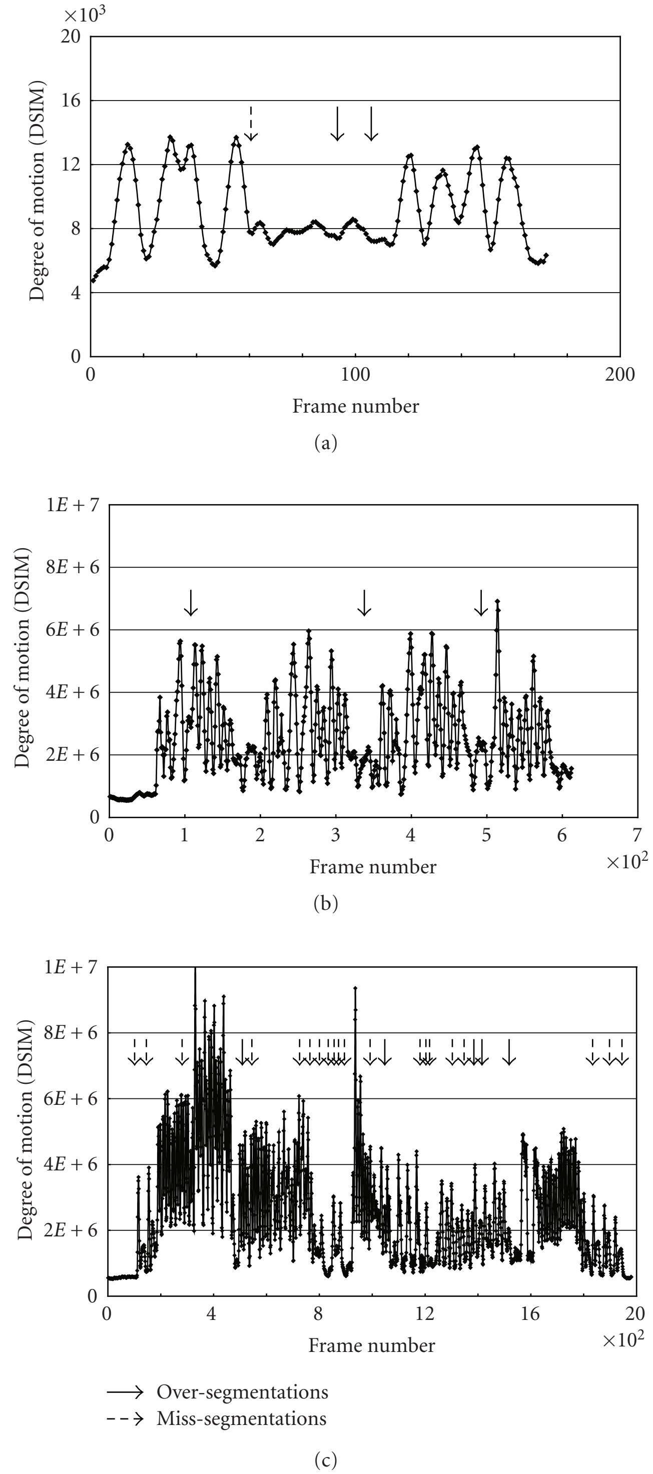


Figure 3: Flowchart to calculate dissimilarity between frames based on ICP algorithm.

Therefore, the number of vertices and topology differ frame by frame, which makes it very difficult to search the correspondent vertices or patches among frames. Although Matsuyama et al. have been developing a deformation algorithm for dynamic 3D model generation [4], the number of vertices and topology needs to be refreshed every few frames.

3. Algorithms

3.1. Dissimilarity Measure by Spherical Registration. In the previous approaches, motion segmentation for TVM was conducted in the feature vector space domains [21, 22]. Although these approaches had advantages in computational efficiency, it is pointed out that motion tracking and analysis in the 3D space is preferable for more accurate processing [25].



(a)

(b)

(c)

Figure 4: Degree of motion: (a) #1, (b) #2-1, (c) #3.

In this paper, we propose a similarity measure based on the mesh surface matching between frames using the ICP algorithm [36]. The ICP algorithm is widely used for geometric alignment between two point clouds for registration. In this work, two frames in TVM are registered with each other using their geometrical information (coordinates of vertices), and the matching error, which is the sum of the distances between correspondent vertices, is used to represent the dissimilarity between the frames. Since the ICP algorithm is asymmetric: correspondent vertices from the ith frame to the jth frame and those from the jth frame to the ith frame are not always the same as shown in Figure 3, we define the dissimilarity between the ith frame and the jth frame ($(\mathrm{DSIM})_{i\text{-}j}$) as in the following equation

$$(\mathrm{DSIM})_{i\text{-}j} = (\mathrm{ME})_{i\text{-}j} + (\mathrm{ME})_{j\text{-}i}, \tag{1}$$

where $(\mathrm{ME})_{i\text{-}j}$ is the matching error from the ith frame to the jth frame and vice versa. For motion segmentation, in particular, only the dissimilarities between neighboring frames are calculated to estimate the degree of motion. In this case, regions whose distances to correspondent vertices are large are regarded as moving parts of human body.

The ICP algorithm assumes that the two point clouds are already roughly aligned with each other. In motion segmentation, only neighboring frames are used to analyze the degree of motion. Therefore, it can be assumed that the above condition is already satisfied. For motion retrieval, dissimilarity between arbitral two frames, where rotation and translation can be different from each other frame, needs to be calculated. In this situation, the two frames are firstly aligned by applying principal component analysis (PCA), and then ICP is conducted. In this manner, the assumption mentioned above can be met.

In our database, each TVM frame contains about 20,000–50,000 vertices depending on the spatial resolution of the model (5 mm–10 mm), which would consume a lot of computational power because the cost for the ICP is proportional to the square number of vertices. Therefore, in our approach, vertices on a 3D model are clustered into 1,024 regions in advance using vector quantization [22, 25] to reduce the computational complexity. The idea of scattering the reduced number of vertices onto the surface is similar to [4]. However, our clustering results can also be used for the modified shape distribution algorithm [22, 25], providing us flexibility in choosing TVM processing algorithms.

The overall process flow for the dissimilarity calculation between frames is summarized in Figure 3.

3.2. Motion Segmentation. Motion segmentation candidates are extracted by searching for the timing when the degree of motion calculated in Section 3.1 becomes the local minimum. This idea is already employed successfully in various kinds of data [22, 25, 29–31] such as in 2D video, motion capture data, and TVM. In dance sequences, in particular, a dancer stops or decreases motion when the meaning of the motion changes to make the dance look lively and elegant.

Searching for local minima for motion segmentation is very good at extracting most of the candidates. On the other hand, such an approach includes a lot of over-segmentation. Therefore, verification is essential. Having over-segmentation is much better than having miss-segmentation. This is because it is difficult to revive the miss-segmentation while over-segmentation can be removed by the verification process.

In conventional approaches, thresholding using empirically predefined values were utilized [21, 29, 30]. However, there is a wide range of variations in the degree of motion depending on motion types. For instance, a hip hop dance and a break dance are acrobatic and contain large motion. On the other hand, *Noh*, which is a Japanese traditional dance, is very slow and elegant. Therefore, it is difficult to set appropriate fixed values for thresholding for any type of motion.

Therefore, we employ relative comparison we have developed in [22, 25] for the verification. In this scheme, each local minimum is compared with the local maxima occurring right before and after the local minimum. Only when both of the local maxima are α times larger than the local minimum, the segmentation point is defined:

$$\begin{cases} \text{if } (l_{\max})_{\text{before}} > \alpha \times l_{\min}, \quad (l_{\max})_{\text{after}} > \alpha \times l_{\min}, \\ \quad \text{the local minimum point is a segmentation point,} \\ \text{otherwise,} \\ \quad \text{the local minimum is regarded as noise.} \end{cases} \tag{2}$$

Here, $l_{\min}$, $(l_{\max})_{\text{before}}$, and $(l_{\max})_{\text{after}}$ represent the local minimum value, the local maximum value occurring right before the local minimum, and that after the local minimum, respectively. In this paper, α is set at 1.1, which was also used in [22].

3.3. Matching between Motion Clips. After the motion segmentation, similar motion retrieval is conducted using the segmented motion clips as minimum units for efficient computation. Since the algorithm is almost the same as [25], only the abstract is presented in this paper.

In our approach, example-based queries are employed. A clip from a certain TVM is given as a query and similar motion is searched from the other clips in the database.

DP matching [37, 38] is utilized to calculate the similarity between the query and candidate clips. DP matching is a well-known matching method between time-inconsistent sequences, which has been successfully used in speech [39, 40], computer vision [41], and so forth.

A TVM sequence in a database (Y) is divided into segments properly in advance according to Section 3.2. Assume that the frames in the query (Q) and the ith clip in Y, $Y^{(i)}$, are denoted as follows:

$$\begin{gathered} Q = \{q_1, q_2, \ldots, q_s, \ldots, q_l\}, \\ Y^{(i)} = \{y_1^{(i)}, y_2^{(i)}, \ldots, y_t^{(i)}, \ldots, y_m^{(i)}\}, \end{gathered} \tag{3}$$

where q_s and $y_t^{(i)}$ are the frames of the sth and tth frame in Q and $Y^{(i)}$, respectively. Besides, l and m represent the number of frames in Q and $Y^{(i)}$.

Table 1: Summary of TVM utilized in experiments. Sequence #1 and sequences #2-1–#2-3 are Japanese traditional dances called *bon-odori* and sequence #3 is a Japanese exercise dance. Sequences #2-1–#2-3 are identical but performed by different persons.

Sequence	#1	#2-1	#2-2	#2-3	#3
# of frames	173	613	612	616	1,981
# of vertices (average)	83 k	17 k	17 k	17 k	17 k
# of patches (average)	168 k	34 k	34 k	34 k	34 k
Resolution	5 mm	10 mm	10 mm	10 mm	10 mm
Frame rate			10 frames/s		

Table 2: Performance summarization of motion segmentation. Results in [25] are also shown for comparison.

Sequence	# 1	# 2-1	# 2-2	# 2-3	# 3	Total	Total [25]
A: # of relevant records retrieved	10	44	46	41	127	268	251
B: # of irrelevant records retrieved	2	3	3	2	5	15	23
C: # of relevant records not retrieved	1	0	0	1	20	22	39
Precision: A/(A+B)	83.3	93.6	93.9	95.3	96.2	94.7	91.6
Recall: A/(A+C)	90.9	100	100	97.6	86.4	92.4	86.6
F value	87.0	96.7	96.8	96.5	91.0	93.5	89.0

Let us define *d(s,t)* as the dissimilarity between $q_{s.}$ and $y_t^{(i)}$ calculated by (1):

$$d(s,t) = (\mathrm{DSIM})_{q_s - y_t^{(i)}}. \tag{4}$$

How to calculate the dissimilarity between frames differs from [25], which is our contribution of this paper. Then, the dissimilarity (D) between the sequences Q and $Y^{(i)}$ is calculated as

$$D\left(Q, Y^{(i)}\right) = \frac{\mathrm{cost}(l,m)}{\sqrt{l^2 + m^2}}, \tag{5}$$

where the cost function $\mathrm{cost}(s,t)$ is defined as in the following equation:

$$\mathrm{cost}(s,t) = \begin{cases} d(1,1) & \text{for } l = m = 1 \\ d(s,t) + \min\{\mathrm{cost}(s,t-1), \mathrm{cost}(s-1,t), \\ \qquad \mathrm{cost}(s-1,t-1)\}, & \text{otherwise.} \end{cases} \tag{6}$$

Here, symbols of Q and $Y^{(i)}$ are omitted in $d(s,t)$ and $\mathrm{cost}(l,m)$ for simplicity. Since the cost is a function of the sequence lengths, $\mathrm{cost}(l,m)$ is normalized by $\mathrm{sqrt}(l^2 + m^2)$, where sqrt is a square root function. The lower the D is, the more similar the sequences are.

4. Experimental Results

In our experiments, five TVM sequences generated using the system in [5] were utilized. The parameters of the data are summarized in Table 1. The sequences #1 and #2-1–#2-3 are Japanese traditional dances called *bon-odori* and the sequence #3 is a physical exercise. The sequences #2-1–#2-3 are identical but performed by different persons. The ground truth of motion segmentation was decided by eight volunteers as described in [22, 25]. The eight subjects were separately asked to divide the sequences into segments without any instruction how to define the segments nor knowing other participants' results. After that, ground truth was defined by analyzing the statistical distribution of the definition by the eight subjects. The α value in (2) was set at 1.1 as long as mentioned otherwise.

The processing time for calculating (1) was about a second on average using MATLAB with MEX function (some critical functions were accelerated by the C language) on a personal computer with Pentium 4 3.2 GHz. Since there are many acceleration algorithms for ICP aiming at real-time operation [39], it is not a significant problem.

Figure 4 demonstrates the degree of motion calculated for the sequences #1, #2-1, and #3. Over-segmented and miss-segmented points are represented with black and white arrows, respectively. The other candidate points after the verification process coincide with the ground truth.

The motion segmentation results for the whole sequence of #1 and for the first 21 seconds of the sequence #2-1 are illustrated in Figure 5. TVM sequence is divided into small but meaningful segments. The images with a cross represent over-segmentations and that with a triangle is a miss-segmentation. It is observed that over-segmentations occur during motion transition such as changing the pivoting foot while the dancer was rotating (Figure 5(a)) and changing direction of motion (Figure 5(b)).

Table 2 summarizes the motion segmentation performance. The results using the modified shape distribution algorithm [22, 25] is also shown for comparison. We can see that miss-segmentation is very much reduced as compared

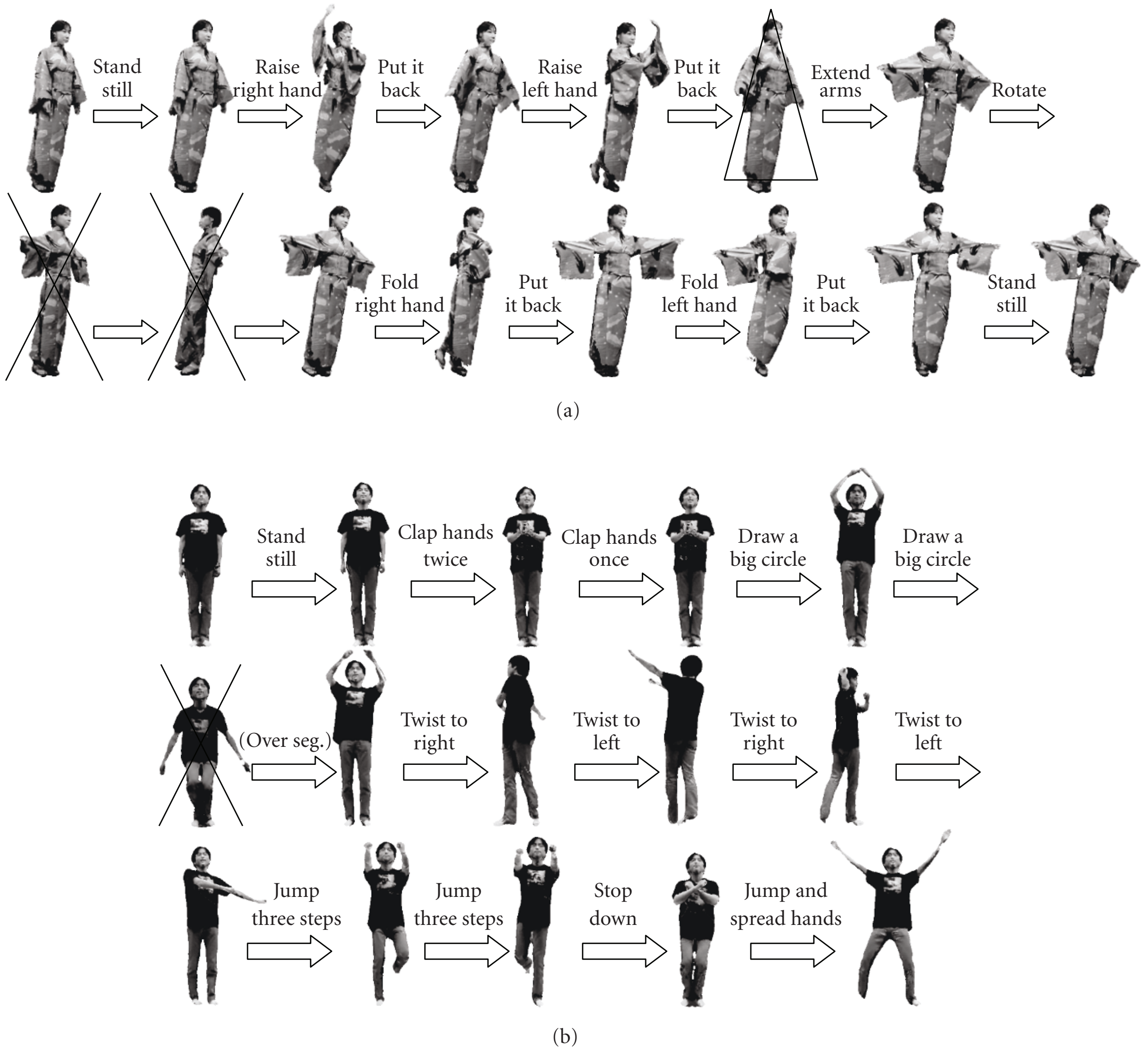


Figure 5: Motion segmentation results: (a) #1, (b) #2-1. Images with a cross represent over-segmentations and that with a triangle is a miss-segmentation.

to [22, 25]. Decreasing miss-segmentation is significantly important because it is difficult to recover miss-segmentation while over-segmentation can be removed by the verification process. The mean precision rate, recall rate, and F value are 95%, 92%, and 94%, respectively. The reason for the larger number of miss-segmentation for the sequence #3 is that the dancer did not decrease the motion speed properly between motions. It is observed that the dissimilarity measure proposed in this paper can extract subtle motion as compared to our previous approaches [22, 25]. This is because the feature vector-based approaches such as shape distributions [42] are not suitable for detecting a small motion though they are eligible for low-cost computation. They are originally developed for comparing similarity between totally different 3D models like cars, planes, coffee cups, and so forth. For similar objects such as cups and glasses, feature vector-based algorithms are designed to yield similar vectors.

The performance comparison with previous works [21, 22] using the sequence #2-1 is shown in Figure 6. The precision-recall relationship is obtained by changing the parameters for verification (α was changed from 1.05 (left top) to 1.4 (right bottom) in our case). It is shown that

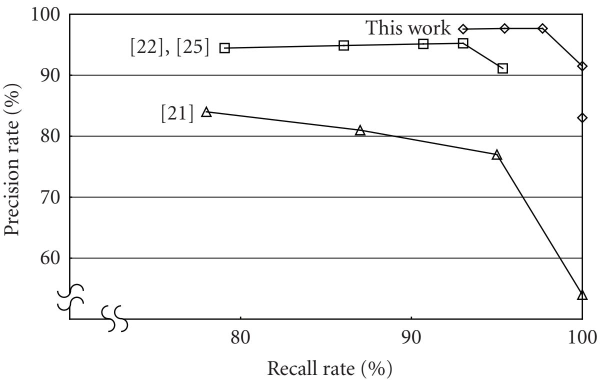
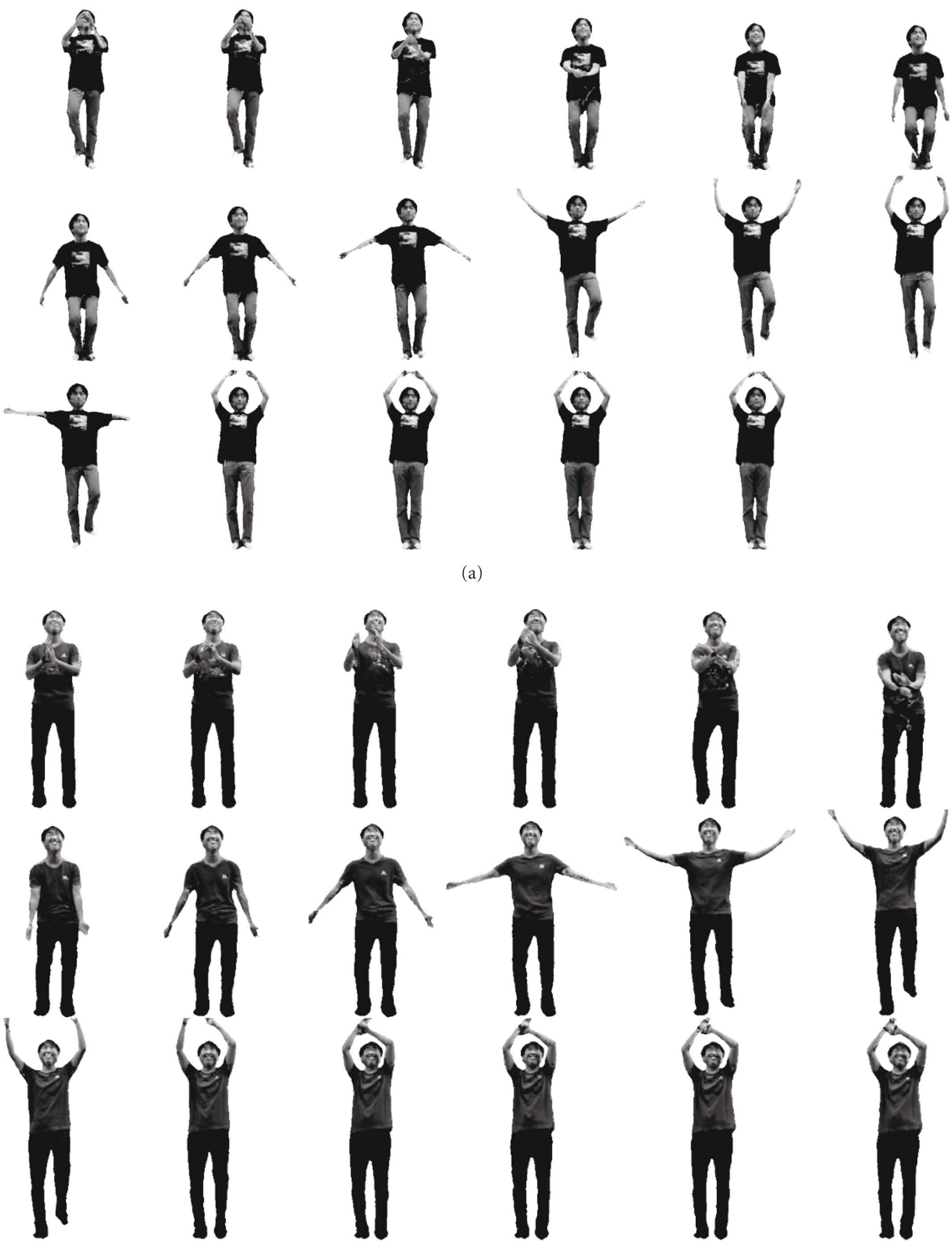


Figure 6: Performance comparison with previous approaches.



(a)

(b)

Figure 7: Clips of “draw a big circle”: (a) from #2-1, (b) from #2-2.

the algorithm developed in this paper is much better than the others. In addition, it is also demonstrated that the motion segmentation performance is the best when α value is from 1.1 to 1.3, thus demonstrating the generality and validity of the relative comparison method.

In the retrieval experiment, 10 clips from five different kinds of motion (5 kinds of motion × 2 clips) were selected from both sequences #2-1 and #2-2. The selected motions are “clap hand,” “draw a big circle,” “twist to right,” “jump three steps,” and “jump and spread hands” shown in Figure 5(b). Example clips are shown in Figure 7.

The similarity matrix among the motion clips is shown in Figure 8. The darker the color is, the closer the sequences are. We can see that similar motion yields higher similarity score, showing the possibility of similar motion retrieval. Although accurate motion analysis in the 3D space is made possible,

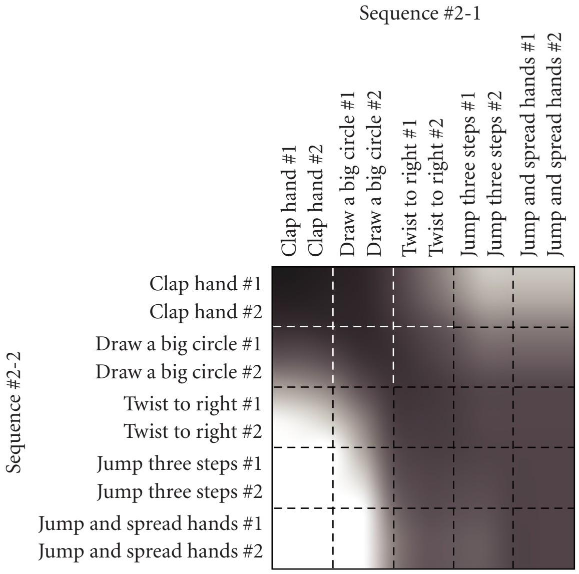


FIGURE 8: Similarity among motion clips.

high computational cost is a problem to be solved in the future work.

5. Conclusions

In this paper, a very robust motion segmentation and motion retrieval for TVM using ICP were developed. Motion segmentation is essential as a preprocessing for indexing, retrieval, editing, and so on. The degree of motion was represented by the matching error of the ICP-based surface point registration. The computational time was reduced by clustering the vertices into groups and using only about 1,000 representative points for the registration. Then, motion segmentation was accomplished by searching the local minima in the degree of motion with a simple but robust verification process employing relative comparison with the local maxima values occurring right before and after the local minima. The superiority of our algorithm to previous works, most of which are histogram-based, was demonstrated by yielding such high precision and recall rates as 95% and 92%, respectively. The high recall rate is especially important because the over-segmentation can be eliminated in the verification process while miss-segmentation cannot be recovered. Over-segmentations were found when the dancer decreased the motion speed to change the direction of the motion and so forth. Higher-level motion understanding and recognition would be required to eliminate these errors. On the other hand, miss-segmentations occurred when the subjects did not dance properly.

In addition, preliminary experimental results for similar motion retrieval presented some promising results. How to reduce the processing time for the DP matching using ICP should be discussed in the future work because the DP matching is more computationally demanding than the motion segmentation, which requires to conduct ICP with only a few neighboring frames.

Although the methods for motion segmentation and motion retrieval are very similar to the authors' previous works using the modified shape distribution algorithm, the contribution of this paper is a similarity evaluation method among the TVM frames for more accurate processing. Since the proposed algorithm calculates the similarity among frames not in the feature vector space but in the 3D space, a more accurate motion analysis has been made possible.

Acknowledgments

This work is supported by Ministry of Education, Culture, Sports, Science and Technology of Japan under the "Development of fundamental software technologies for digital archives" Project.

References

[1] 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.

[2] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan, "Image-based visual hulls," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, pp. 369–374, New Orleans, La, USA, July 2000.

[3] S. Wurmlin, E. Lamboray, O. G. Staadt, and M. H. Gross, "3D video recorder," in *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications (PG '02)*, pp. 325–334, Beijing, China, October 2002.

[4] T. Matsuyama, X. Wu, T. Takai, and T. Wada, "Real-time dynamic 3D object shape reconstruction and high-fidelity texture mapping for 3D video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 3, pp. 357–369, 2004.

[5] K. Tomiyama, Y. Orihara, M. Katayama, and Y. Iwadate, "Algorithm for dynamic 3D object generation from multiviewpoint images," in *Three-Dimensional TV, Video, and Display III*, vol. 5599 of *Proceedings of SPIE*, pp. 153–161, Philadelphia, Pa, USA, October 2004.

[6] J. Starck and A. Hilton, "Virtual view synthesis of people from multiple view video sequences," *Graphical Models*, vol. 67, no. 6, pp. 600–620, 2005.

[7] J. Starck and A. Hilton, "Surface capture for performance-based animation," *IEEE Computer Graphics and Applications*, vol. 27, no. 3, pp. 21–31, 2007.

[8] R. Skerjanc and J. Liu, "A three camera approach for calculating disparity and synthesizing intermediate pictures," *Signal Processing: Image Communication*, vol. 4, no. 1, pp. 55–64, 1991.

[9] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '93)*, pp. 279–285, Anaheim, Calif, USA, August 1993.

[10] S. E. Chen, "Quicktime VR: an image-based approach to virtual environment navigation," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, pp. 29–38, Los Angeles, Calif, USA, August 1995.

[11] N. L. Chang and A. Zakhor, "Arbitrary view generation for three-dimensional scenes from uncalibrated video cameras," in *Proceedings of the 20th International Conference on Acoustics,*

Speech, and Signal Processing (ICASSP '95), vol. 4, pp. 2455–2458, Detroit, Mich, USA, May 1995.

[12] S. M. Seitz and C. R. Dyer, "Physically-valid view synthesis by image interpolation," in *Proceedings of IEEE Workshop on Representation of Visual Scenes (VSR '95)*, pp. 18–25, Cambridge, Mass, USA, June 1995.

[13] L. McMillan and G. Bishop, "Plenoptic modeling: an image-based rendering system," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, pp. 39–46, Los Angeles, Calif, USA, August 1995.

[14] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*, pp. 31–42, New Orleans, La, USA, August 1996.

[15] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*, pp. 43–54, New Orleans, La, USA, August 1996.

[16] M. Tanimoto and T. Fujii, "FTV—free viewpoint television," ISO/IEC JTC1/SC29/WG11 M8595, July 2002.

[17] B. G. Baumgart, *Geometric modeling for computer vision*, Ph.D. thesis, Stanford University, Stanford, Calif, USA, 1974.

[18] H. Habe, Y. Katsura, and T. Matsuyama, "Skin-off: representation and compression scheme for 3D video," in *Proceedings of the Picture Coding Symposium (PCS '04)*, pp. 301–306, San Francisco, Calif, USA, December 2004.

[19] K. Müller, A. Smolic, M. Kautzner, P. Eisert, and T. Wiegand, "Predictive compression of dynamic 3D meshes," in *Proceedings of IEEE International Conference on Image Processing (ICIP '05)*, vol. 1, pp. 621–624, Genova, Italy, September 2005.

[20] S. Han, T. Yamasaki, and K. Aizawa, "3D video compression based on extended block matching algorithm," in *Proceedings of IEEE International Conference on Image Processing (ICIP '06)*, pp. 525–528, Atlanta, Ga, USA, October 2006.

[21] J. Xu, T. Yamasaki, and K. Aizawa, "3D video segmentation using point distance histograms," in *Proceedings of IEEE International Conference on Image Processing (ICIP '05)*, vol. 1, pp. 701–704, Genova, Italy, September 2005.

[22] T. Yamasaki and K. Aizawa, "Motion 3D video segmentation using modified shape distribution," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 1909–1912, Toronto, Canada, July 2006.

[23] J. Xu, T. Yamasaki, and K. Aizawa, "Key frame extraction in 3D video by rate-distortion optimization," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 1–4, Toronto, Canada, July 2006.

[24] T. Yamasaki and K. Aizawa, "Similar motion retrieval of dynamic 3D mesh based on modified shape distribution," in *Proceedings of the Annual Conference of the European Association for Computer Graphics (Eurographics '06)*, pp. 9–12, Vienna, Austria, September 2006.

[25] T. Yamasaki and K. Aizawa, "Motion segmentation and retrieval for 3D video based on modified shape distribution," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, Article ID 59535, 11 pages, 2007.

[26] J. Xu, T. Yamasaki, and K. Aizawa, "Motion editing in 3D video database," in *Proceedings of the 3rd International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT '06)*, pp. 472–479, Chapel Hill, NC, USA, June 2006.

[27] J. Starck and A. Hilton, "Spherical matching for temporal correspondence of non-rigid surfaces," in *Proceedings of the 10th International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1387–1394, Beijing, China, October 2005.

[28] G. Miller, A. Hilton, and J. Starck, "Interactive free-viewpoint video," in *Proceedings of the 2nd IEE European Conference on Visual Media Production (CVMP '05)*, pp. 52–61, London, UK, November-December 2005.

[29] T. S. Wang, H. Y. Shum, Y. Q. Xu, and N. N. Zheng, "Unsupervised analysis of human gestures," in *Proceedings of the 2nd IEEE Pacific Rim Conference on Multimedia (PCM '01)*, pp. 174–181, Bejing, China, October 2001.

[30] T. Shiratori, A. Nakazawa, and K. Ikeuchi, "Rhythmic motion analysis using motion capture and musical information," in *Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI '03)*, pp. 89–92, Tokyo, Japan, July-August 2003.

[31] K. Kahol, P. Tripathi, and S. Panchanathan, "Automated gesture segmentation from dance sequences," in *Proceedings of the 6th International Conference on Automatic Face and Gesture Recognition (FGR '04)*, pp. 883–888, Seoul, Korea, May 2004.

[32] Y. Rui and P. Anandan, "Segmenting visual actions based on spatio-temporal motion patterns," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 1, pp. 111–118, Hilton Head Island, SC, USA, June 2000.

[33] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," in *Proceedings of Graphics Interface*, pp. 185–194, London, Canada, May 2004.

[34] C. Lu and N. J. Ferrier, "Repetitive motion analysis: segmentation and event classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 258–263, 2004.

[35] W. Takano and Y. Nakamura, "Segmentation of human behavior patterns based on the probabilistic correlation," in *Proceedings of the 19th Annual Conference of the Japanese Society for Artificial Intelligence (JSAI '05)*, Kitakyushu, Japan, June 2005, 3F1-01.

[36] P. J. Besl and N. D. McKay, "A method for registration of 3D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[37] R. Bellman and S. Dreyfus, *Applied Dynamic Programming*, Princeton University Press, Princeton, NJ, USA, 1962.

[38] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, Athena Scientific, Belmont, Mass, USA, 1995.

[39] H. J. Ney and S. Ortmanns, "Dynamic programming search for continuous speech recognition," *IEEE Signal Processing Magazine*, vol. 16, no. 5, pp. 64–83, 1999.

[40] H. Ney and S. Ortmanns, "Progress in dynamic programming search for LVCSR," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1224–1240, 2000.

[41] A. A. Amini, T. E. Weymouth, and R. C. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 855–867, 1990.

[42] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Transactions on Graphics*, vol. 21, no. 4, pp. 807–832, 2002.