3D Video Segmentation Using Point Distance Histograms

Jianfeng Xu
Dept. of Electronic Engineering
The University of Tokyo
Chiba, Japan
fenax@hal.k.u-tokyo.ac.jp

Toshihiko Yamasaki, and Kiyoharu Aizawa
Dept. of Frontier Informatics
The University of Tokyo
Chiba, Japan
{yamasaki, aizawa}@hal.k.u-tokyo.ac.jp

***Abstract*—Similar to 2D video segmentation, 3D video segmentation is to divide the 3D video in temporal domain into a set of meaningful and manageable segments (*shots*) that are used as basic elements for indexing. This paper proposes a temporal segmentation method for 3D video for the first time as far as we know. Point distance histograms are used considering the trade-off between computational cost and effectiveness. In order to reflect the real motion of 3D object, three fixed points are selected, which can avoid what we call "the same sphere problem." And simulation results on total 285 frames, which are composed of four different sequences, show our method is very effective.**

Keywords-3D video; segmentation; point distance histogram

I. INTRODUCTION

In recent years, 3D technique, which can provide much more realistic observation from any view point [1], is developed quickly and applied in many fields such as broadcast, education, medical system, entertainment. Recently, several research groups focus on generating 3D videos, which are the recordings of the real world frame by frame instead of computer graphic animations [1, 2, 3], using multiple synchronous cameras and aiming at capturing and building up a database of Japanese traditional dramas. In Ref. [1], 3D video, in which each 3D frame is encoded in VRML format, is generated by combining stereo matching and the volume intersection method with 22 synchronous cameras. In this paper, the 3D frame is the 3D object data at each capturing time in 3D video.

In order to establish a database of 3D videos, compression and shot (several successive frames with some common visual characteristics in 3D video) segmentation of 3D video are mandatory for efficient and effective archiving and retrieval. Several algorithms on 3D video compression have been developed [4, 5]. However, no 3D video segmentation technique has been reported so far. There are some related works such as 3D object retrieval [6, 7, 8] and 2D video segmentation [9, 10]. But the 3D object retrieval algorithms do not consider the strong temporal correlation existing in 3D video. Therefore, they are not suitable for 3D video segmentation. Also, there are many useful characteristics in the current 3D generation system [1], which are not fully utilized in these 3D object retrieval systems. On the other hand, 2D video are usually expressed as pixel array while VRML uses mesh model. Therefore, 2D video segmentation algorithms are also not applicable to 3D video segmentation.

The purpose of this paper is to demonstrate a simple and effective method for segmenting 3D video for the first time. In our method, three point distance histograms are utilized to detect shot transitions by revealing the motion of 3D object. Since the algorithm is based on low-level features of 3D video, computational cost is quite low. The effectiveness and robustness of our method have been demonstrated by the experimental results.

II. POINT DISTANCE HISTOGRAM

The main idea of our method is we would segment the data in temporal domain by the shape and motion of 3D object. There are three kinds of information in our 3D video. One is the vertex position in Cartesian coordinates, one is the connection information for each triangle and the last is the color information in normalized RGB. Obviously, the vertex position is a good candidate to describe the object shape and motion. To simplify the computation while keeping the performance, the point distance histogram is adopted, which reveals the point distribution of 3D object. In this paper, point distance is defined as Euclidean distance from one fixed point.

$$DP_n(i) = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} \qquad (1)$$

where (x_i, y_i, z_i) means the i-th vertex coordinates and (x_0, y_0, z_0) is for the fixed point. Fig. 1 shows the distance is rather smooth both in spatial and temporal domain. Here, it is no need to select the object center as the fixed point like most algorithms in 3D object retrieval to get the invariance feature since the generation system already guarantees the fixed point is changed little in successive 3D frames, also shown in Fig. 1, or refer to [1] in detail.

Then the histogram of $DP_n(i)$ is calculated within J bin number and normalized by total vertex number, which forms our feature vector (FV). Fixed bin length is adopted instead of fixed bin number, which can avoid great differences in the 3D object static part, as described by the bins from bin #35 to bin #60 in Fig. 2a. The latter will account the vertexes in the corresponding part into different bins due to the different bin length. Figs. 2b-2d shows our distance histogram reflects the motion well, which is the foundation of our method. Then, Euclidean distance between the histograms in two successive 3D frames is used to evaluate the 3D object shape and motion.

0-7803-9134-9/05/$20.00 ©2005 IEEE

Fig. 1. Frame #38 and frame #39 in Batter, gray value means the distance from (0, 0, 0).

$$DDH_n = \sqrt{\sum_{j=0}^{\max(J_n-1,J_{n+1}-1)} \left(DH^*_{n+1}(j) - DH^*_n(j)\right)^2}$$

$$DH^*_n(j) = \begin{cases} DH_n(j) & j < J_n \\ 0 & otherwise \end{cases} \quad (2)$$

$DH^*_{n+1}(j)$ is defined by the same way as $DH^*_n(j)$

where $DH_n(j)$ is the distance histogram of the j-th bin in the n-th frame, $DH^*_n(j)$ is the modified histogram, J_n is the bin number in the n-th frame.

III. Three Fixed Points Strategy

In our method, the moving vertexes on the same sphere with the fixed point as its center will be accounted in the same bin due to the equal distances. For example, the vertex p_i in Fig. 3a moves from frame #n to frame #(n+1) at one sphere. We call it as the same sphere problem, which may cause very different histogram distances between the similar 3D object motions. Fig. 4 shows the histogram difference between Batter #27 and Batter #28 is much smaller than that between Batter #28 and Batter #29 although they have similar motions. And this problem may happen in successive frames since the motion consistency will keep these vertexes on the sphere, which limits the effect of low pass filter. In this paper, three fixed points are selected to calculate the histogram distances separately so that a vertex can not move on all three spheres as shown in Fig. 3b. Obviously, one fixed point may cause the vertexes on a sphere are in the same distances and two fixed points may cause the vertexes on a circle are in the same distances and three fixed points not in one line will have no problem. In other words, the vertexes in one dimension (1D) will be determined by the distance from one fixed points, those in 2D will be determined by the distances from two fixed points, and those in 3D will be determined by the distances from three fixed points which are not in one line.

Lastly, the three histogram distances are averaged frame by frame. Experimental results (shown in Fig. 5) show our solution is efficient for the problem in the mean of keeping the motion information while reducing the noise greatly such as the noise at frames #28, #44, #75 and #80.

IV. Decision Strategy

The segmentation decision strategy has direct influence on the segmentation result. One efficient strategy is carefully designed to reduce the influence by the noise of histogram distance, as can be seen in Fig. 6. From Fig. 6, two different types of shot transitions are defined as follows:

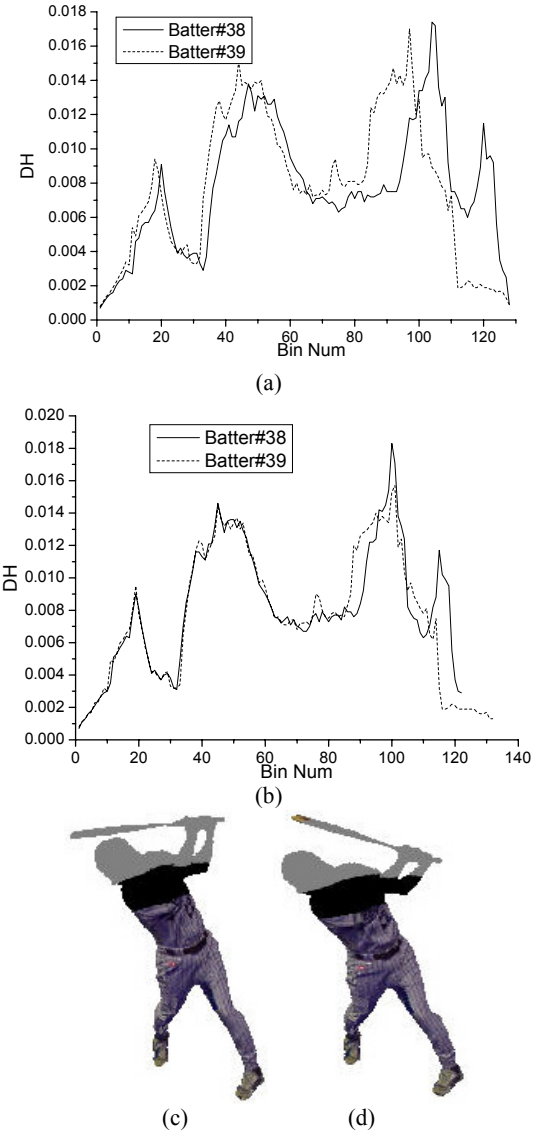

Fig. 2. Point distance histogram for frame #38 and #39 in Batter: (a) fixed bin number; (b) fixed bin length; (c) frame #38; (d) frame #39; black part in (c) and (d) is for bin73 to bin97, gray part in (c) and (d) is for bin97 to bin128..

Abrupt transition means the neighboring shots are in different scenes but in the same sequence and *gradual transition* means the neighboring shots are in different phases but in the same scene.

Then the decision strategy is as follows.

abrupt transition:

$$DDH_n - DDH_{n+1} > Th1 \quad (3)$$

gradual transition:

$$\frac{DDH_n}{DDH_{avg\_curr\_seg}} > Th2 \; and \; \frac{DDH_{n+1}}{DDH_{avg\_curr\_seg}} < Th3 \quad (4)$$

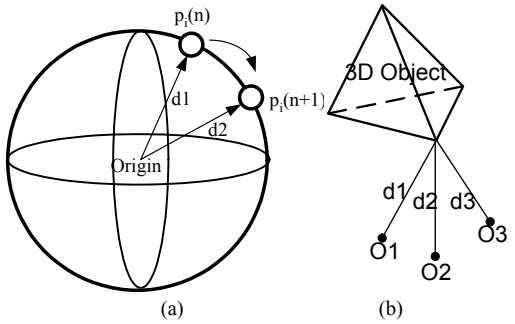

Fig. 3. The same sphere problem and its solution.

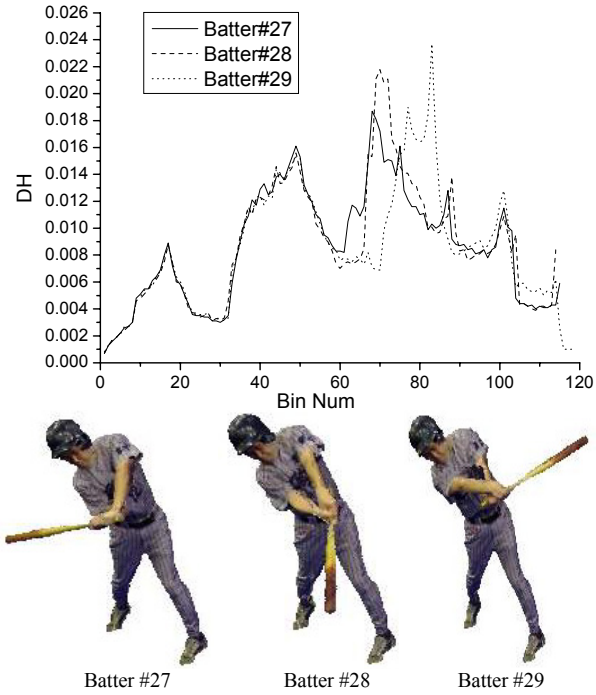

Fig. 4. Distance histograms in the similar motion frames.

where DDH_n is the histogram distance between frame #(n+1) and frame #n. $DDH_{avg\_curr\_seg}$ is the instant average histogram distance in the current shot. *Th1*, *Th2* and *Th3* are three thresholds, which are decided by the experiments. Here only if successive two frames are large or small enough, a gradual transition is decided to happen. Also one shot should include at least three frames. This strategy can achieve better performance than median filter on histogram distance.

V. Simulation Results

The test data come from NHK in VRML format [1]. Four sequences (totally 285 frames) are combined to form 3 abrupt transitions artificially, namely, Toshiko (173 frames, dancing), Fujita (10 frames, hand movement), Batter (51 frames, battering) and Pitcher (51 frames, pitching). Table 1 gives the simulation parameters. And the three fixed points are (0, 0, 0), (-20, 20, 0), (20, 20, 0).

Table 2 and Fig. 7 give the segmentation results, which show our method is effective, i.e., only three false positives are detected in total 24 segments.

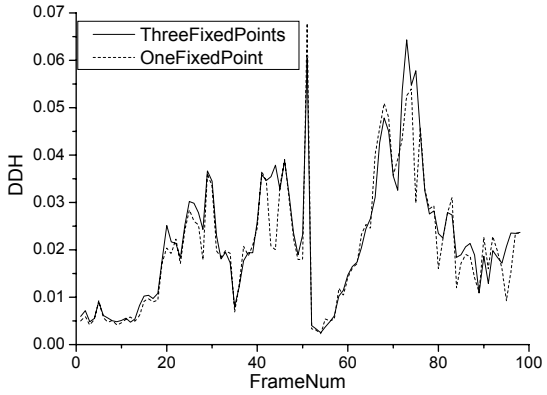

Fig. 5. Histogram distances: one fixed point is (0, 0, 0); three fixed points are (0, 0, 0), (-20, 20, 0) and (20, 20, 0).

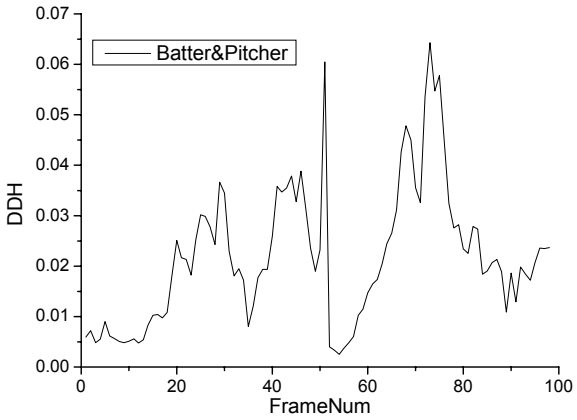

Fig. 6. Average histogram distance.

There are some parameters which are decided by the experiments. Therefore, some evaluations should be given. Obviously, *Th1* will influence the cut transition directly, refer to (3). *Th2* and *Th3* will affect the gradual transition greatly, refer to (4). Bin length has little influence on the results because the histogram shape is kept in different bin lengths. When bin length is set as 2.2, the only difference is frame #65 will be missed. Although any three points which are not in one line will solve "the same sphere problem" in theory, they should not be very close in practical system in order to discriminate the motion clearly. Fig. 8 shows the histogram distances for Batter and Pitcher by (0, 0, 0), (-20, 20, 0), (20, 20, 0) and (0, 0, 0), (0, 30, 0), (-30, 0, 0), which shows similar results.

VI. Conclusion

In this paper, one simple and effective method for segmenting 3D video is proposed for the first time, which uses point distance histogram to reflect the shape and motion of 3D object. In order to resolve "the same sphere problem" existing in point distance histogram, we propose the three fixed points strategy. After the improvement, great noise is removed while motion information is kept. Simulation results show our method is effective for 3D video segmentation with very low computation.

TABLE I. SIMULATION PARAMETERS

Parameters	*Th1*	*Th2*	*Th3*	Bin Length
Values	0.03	1.9	0.8	1.1

TABLE I. SEGMENTATION RESULTS

Sequence name (description)	Transition category	Frame No.	Description*
Toshiko (dancing)	Gradual	10	Prepare
	Gradual	17	Move hands
	Gradual	24	Pause
	Gradual	39	False
	Gradual	44	Move hands (cont.)
	Gradual	50	Pause
	Gradual	56	Move hands (cont.)
	Gradual	65	Raise both hands
	Gradual	114	Rotate body
	Gradual	138	Complex motion
	Gradual	148	False
	Gradual	164	Stop
Fujita (hand movement)	Abrupt	173	Dance
	Gradual	179	Stop
Batter (battering)	Abrupt	183	Prepare to hit
	Gradual	201	Hit
	Gradual	214	Prepare to back
	Gradual	223	Back
	Gradual	230	Prepare again
Pitcher (pitching)	Abrupt	234	Stand still
	Gradual	240	Prepare to pitch
	Gradual	249	Pitch
	Gradual	259	False
	Gradual	266	Back

* It describes the shot from last Frame No. to current Frame No.

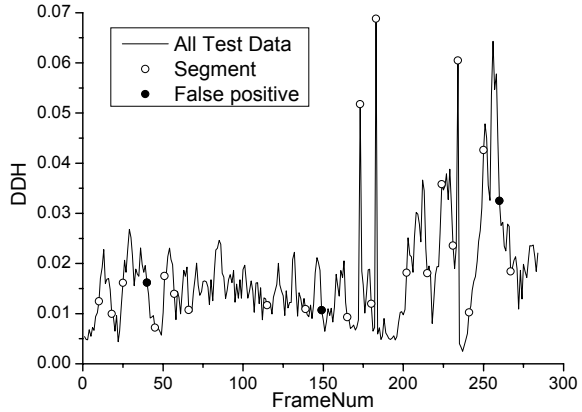

Fig. 7. Histogram distance and segment result.

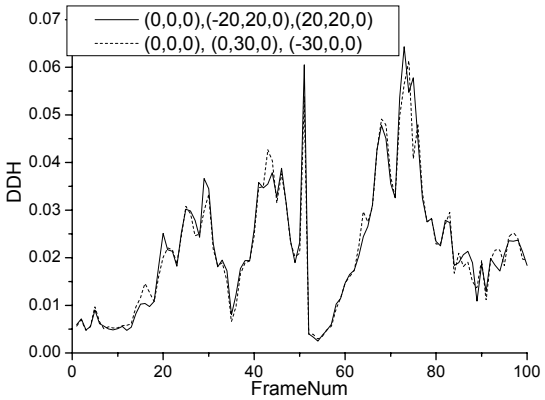

Fig. 8. Influence of fixed point's selection.

ACKNOWLEDGMENT

All test data including Toshiko, Fujita, Batter and Pitcher are provided by NHK, which is greatly appreciated by the authors. This work is supported by Ministry of Education, Culture, Sports, Science and Technology under the "Development of fundamental software technologies for digital archives" project.

REFERENCES

[1] K. Tomiyama, Y. Orihara, M. Katayama, and Y. Iwadate, "Algorithm for dynamic 3D object generation from multi-viewpoint images," Proceeding of SPIE, Vol. 5599, pp. 153-161, 2004.

[2] T. Matsuyama, X. Wu, T. Takai, and T. Wada, "Real-time dynamic 3-D object shape reconstruction and high-fidelity texture mapping for 3-D video," IEEE Trans. Circuit and System for Video Technology, Vol. 14, No. 3, pp.357-369, March 2004.

[3] T. Kanade, P. Rander, and P. Narayanan, "Virtualized reality: constructing virtual worlds from real scenes," IEEE Multimedia, Vol. 4, No. 1, pp. 34-47, Jan./March 1997.

[4] J. Lengyel, "Compression of Time Dependent Geometry," in Proc. IEEE, Vol. 86, pp.1052-1063, June 1998.

[5] J.H. Yang, C.S. Kim, and S.U. Lee "Compression of 3D Triangle Mesh Sequences," in Proc. IEEE Workshop on Multimedia Signal Processing, pp.181-186, Oct. 2001.

[6] T. Zaharia and F. Preteux, "3D shape-based retrieval within the MPEG-7 framework," In Proc. SPIE Conf. on Nonlinear Image Processing and Pattern Analysis XII, Vol. 4304, pp. 133-145, 2001.

[7] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," ACM Trans. On Graphics, Vol. 21, No. 4, pp. 807-832, Oct. 2002.

[8] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, "A Search Engine for 3D Models," ACM Transactions on Graphics, Vol. 22, No. 1, pp. 83-105, Jan. 2003.

[9] I. Koprinska and S. Carrato, "Temporal video segmentation: A Survey," Signal Processing: Image Communication, Vol. 16, No. 5, pp. 477-500, Jan. 2001.

[10] Y. Aslandogan and C. Yu, "Techniques and Systems for Image and Video Retrieval," IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, pp. 56-63, Jan./Feb. 1999.