INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11
MPEG2004/M11279
October 2004, Mallorca

Title: Coding of Layered Depth Image using Coherency between Point Samples
Source: GIST and ETRI
Authors: Yo-Sung Ho, Seung-Uk Yoon, and Sung-Yeol Kim
(Gwangju Institute of Science and Technology)
Daehee Kim, Sukhee Cho, Kugjin Yun, Chunghyun Ahn, and Sooin Lee
(Electronics and Telecommunications Research Institute)
Status: Proposal

1 Introduction

Layered depth image (LDI) is an efficient approach to represent three-dimensional objects with complex geometry for image-based rendering (IBR). As we have already explained in the "Multi-view Video Coding using Layered Depth Image" [1], multi-view video sequences with depth can be efficiently coded by using the concept of LDI. In this document, we describe an LDI coding algorithm considering coherency between point samples.

2 Coding of Layered Depth Image

Core experiments on image-based rendering (AFX 8.3) introduce several coding methods for IBR data [2][3]. In addition, Duan *et al*. [4] proposed a coding algorithm for the sparse and non-rectangular supported data of LDI. They first record the number of layers (NOL) at each pixel location. LDI data are then reorganized into a more suitable layout by dividing LDI into layers, each of which contains a mask indicating the existence of pixel in the layer. Each LDI layer is then separated into individual components: Y, Cb, Cr, alpha, depth, and splat table index.

Component images of each layer are compressed separately. Data aggregation is exploited to collect data on the same layer, so that data are more compactly distributed. An arbitrary shape wavelet transform and entropy coding are used to compress the aggregated data. Finally, the compressed bitstreams of the different layers and components are concatenated to form the compressed LDI bitstream. Data aggregation is depicted in Fig. 1.

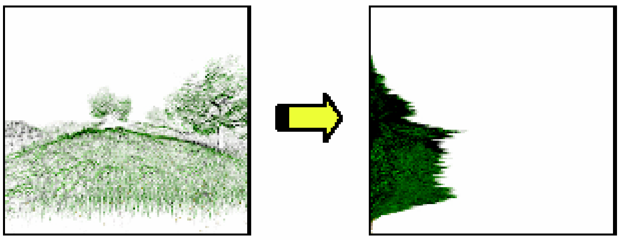


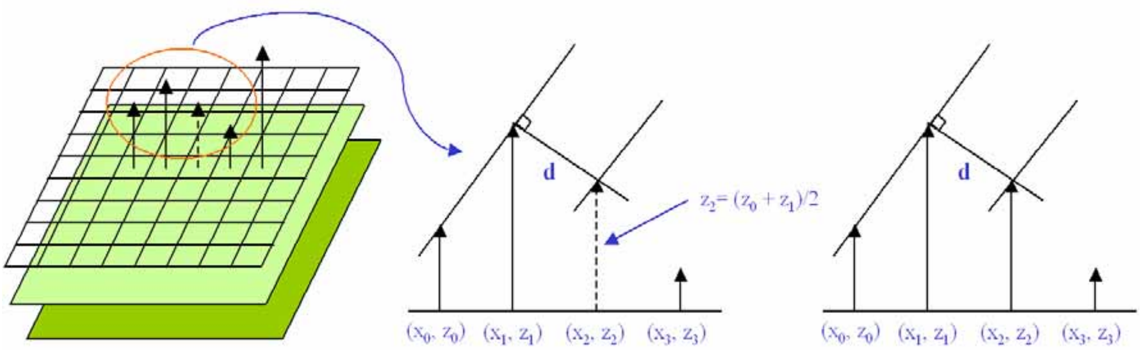
Fig. 1. Data Aggregation of LDI

Although they exploit data aggregation to apply an arbitrary shape coding method, they do not consider correlation of data. Each component image of LDI has great coherency between pixels. We can therefore use this similarity to enhance the coding efficiency of LDI data [5].

3 Coding of Layered Depth Image Using Coherency between Point Samples

Because of the special data structure of LDI, existing still image compression methods, such as JPEG, cannot be applied directly or are not very efficient. There are three key characteristics of the LDI data. It contains multiple layers at each pixel location; the distribution of pixels is sparse in the back layer; and each pixel has multiple attribute values. In the previous work [4], data aggregation is performed to use these key features of LDI. After aggregating LDI data, an arbitrary shape wavelet transform and entropy coding are applied.

In this document, we explain a new preprocessing algorithm to improve the coding efficiency. Although we focus on depth information, our algorithm can be directly applied to color data, also. Since we observe that (*x*, *z*) values are changing for the fixed Y-axis, we can consider the one-dimensional (1-D) depth value as the two-dimensional (2-D) point. Along the increasing direction of the X-axis, we draw a line passing through two points, and then calculate the Euclidean distance between the line and the current depth value. The procedure is illustrated in Fig. 2.



(a) (b) (c)

Fig. 2. Calculation of the Minimum Distance:
(a) Layers of LDI, (b) Empty pixel location, (c) Non-empty pixel location

Figure 2(a) shows the spatial relationship among layers of LDI. The proposed preprocessing method uses correlation between attribute values in the same layer for each component image. In Fig. 2(b), the dotted arrow represents a padded depth value at an empty pixel location. We insert the average value of the previous two points into the vacant position. After calculating the minimum distance, we replace the current depth value by the minimum distance. Finally, the inserted average values are removed before the data aggregation. For non-empty pixel location, we calculate the distance by using z_2 directly as shown in Fig. 2(c). The distance between the line passing through $A(x_0, z_0)$ and $B(x_1, z_1)$, and the point $C(x_2, z_2)$ is computed by

$$d = \frac{\left|(A-B)^{\perp} \cdot (C-A)\right|}{|A-B|} \tag{1}$$

where $A^{\perp}$ is the counterclockwise perpendicular vector to the given A; it means that $(x_0, y_0)^{\perp}$ is $(-y_0, x_0)$.

These procedures are similar to the differential coding method. Instead of calculating the direct difference between two values, we compute the distance from the line through the previous two points to the current point. Since the direct difference becomes greater in the back layer, the differential method is not proper for the data structure of LDI. We compare the standard deviation for the differential scheme and our method in the experimental results. Since each pixel contains depth and color at the same location, we can easily compute the minimum distance for color values of Y, Cr, and Cb components; hence, Eq. (1) can be directly reused. In our algorithm, the Euclidean distance is used as the measure for representing the coherency among neighboring depth and color values.

4 Experimental Results and Analysis

Efficiency of the proposed preprocessing algorithm is demonstrated with the following experiments. Figure 3 shows the test data set of LDI scenes. The resolution of Ball LDI is 246x246 and that of Flower is 690x476; three layers are used for each LDI.

(a) (b)

Fig. 3. Test LDI data set: (a) Ball, (b) Flower

We calculate the standard deviation for test LDIs to evaluate the performance of the proposed algorithm before and after the preprocessing. As shown in Table 1, the standard deviation of each LDI decreases over 45% after preprocessing. Especially, Table 1 shows that the standard deviation is much more reduced for Flower LDI. It means that the more pixels, the more reduction occurs because the replaced minimum distance lowers differences between depth values.

Table 1. Standard Deviations of Depth Information

	Ball		Flower	
	Original	Proposed	Original	Proposed
Layer 1	51.98	27.06	1815.07	364.46
Layer 2	118.14	49.81	3080.90	609.16
Layer 3	160.31	74.16	3749.96	769.95

Table 2 shows total bits for representing depth after the variable length coding. Original refers to the previous approach without preprocessing. The data size is decreased over 20% because the distribution of depth values is skewed.

Table 2. Total Bits for Representing Depth [kBytes]

Ball			Flower		
Original	Proposed	Reduction	Original	Proposed	Reduction
78.62	67.54	21.15 %	415.66	363.78	20.56 %

Finally, we compare our algorithm with the differential coding method in terms of the standard deviation. In table 3, Diff stands for the previous method with preprocessing using the differential method and Prop represents the proposed one. Original has the same meaning as in Table 2. Table 3 shows that the proposed scheme provides higher reduction ratio, because direct differences among depth values become greater than the minimum distance, especially in the back layer. For this reason, the proposed preprocessing algorithm further reduces the variance of depth and color data of LDI.

Table 3. Performance Comparison

	Ball			Flower		
	Original	Diff.	Prop.	Original	Diff.	Prop.
Layer 1	51.98	21.05	27.06	1815.07	399.82	364.46
Layer 2	118.14	47.24	46.28	3080.90	924.50	613.61
Layer 3	160.31	74.57	65.66	3749.96	1459.07	756.03

5 Conclusion

In this document, we have described an efficient preprocessing algorithm to code depth and color information of layered depth image. We consider each depth value as a 2-D point. After computing the minimum Euclidean distance between a line and the current point, the minimum distance replaces the current depth value. Since the previous approach does not consider coherency between neighboring pixels, we focus on using correlations of depth and color values within each component image. Experimental results demonstrate that the proposed algorithm reduces the variance of depth. Hence, the transform efficiency was improved and the size of data was reduced.

6 References

[1] ISO/IEC JTC 1/SC 29/WG 11/M11278, “Multi-view Video Coding using Layered Depth Image,” October 2004.

[2] ISO/IEC JTC 1/SC 29/WG 11/M7773, “Result of Core Experiment on Image-based Rendering (AFX A8.3),” December 2001.

[3] ISO/IEC JTC 1/SC 29/WG 11/M8037, “Result of Core Experiment on Depth Image-based Representation (AFX A8.3),” March 2002.

[4] J. Duan and J. Li, “Compression of the LDI,” IEEE Transaction on Image Processing, Vol. 12, No. 3, pp. 365-372, November 2003.

[5] S. Yoon, S. Kim, and Y. Ho, “Preprocessing of Depth and Color Information for Layered Depth Image Coding,” Pacific-Rim Conference on Multimedia, November 2004.