HIERARCHICAL SEGMENTATION BASED POINT CLOUD ATTRIBUTE COMPRESSION

Ke Zhang, Wenjie Zhu, Yiling Xu

Shanghai Jiao Tong University, Shanghai, China
E-mail: {zhang_ke, rebecca28, yl.xu}@sjtu.edu.cn

ABSTRACT

With the rapid development of 3D capture techniques, point cloud has attracted significant attentions in recent years. Due to the large data volume of point cloud, efficient compression algorithms are essential for reducing bandwidth and storage consumption. In this paper, we present a novel scheme for point cloud attribute compression based on hierarchical segmentation. In this case, both global segmentation in photometric space and local segmentation in geometric space are analyzed to split point cloud into clusters. An octree based traversal algorithm is introduced to obtain the attribute stream of each cluster. Then, an intra-cluster prediction method is applied to achieve lossless compression. Meanwhile, we map the attribute streams to uniform 2D grids and leverage image coding method to achieve satisfying lossy compression performance. Experimental results demonstrate that our scheme outperforms the previous MPEG scheme in terms of coding efficiency.

Index Terms— point cloud, attribute compression, segmentation, intra-cluster prediction

1. INTRODUCTION

With the increasing capacity of 3D scanning devices, it is feasible to represent an object by point cloud. In the last decade, point cloud has been used in many applications, such as virtual reality, 3D printing, and autonomous driving. It is, however, prohibitive to transmit point cloud over the current networks due to the huge data volume. Thus, the compression of point cloud is absolutely a necessary task. Much work has been done to find effective point cloud compression methods.

Octree is a widely used data structure which recursively subdivides the point cloud-aligned bounding box into eight children cells [1, 2]. Typically, the position of each cell is represented by the cell center and the attribute is set to the average of enclosed points [3-5]. Based on octree decomposition, Schnabel [4] proposed a progressive method for geometry compression by encoding the point cloud in terms of non-empty octree nodes. Later, a differential encoding technique of consecutive octree was introduced to remove temporal redundancies of point cloud streams [5]. Considering the positions and colors as graph signals, Thanou [6] used motion compensation to perform geometry and color prediction.

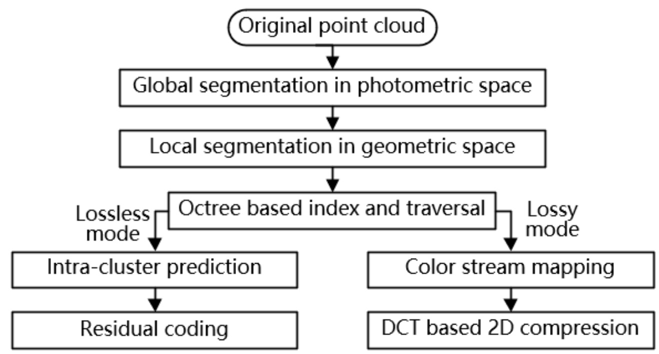


Fig.1. Overview of the proposed scheme

For point cloud attribute compression, various methods have been proposed as well. Graph transform was applied in [7] to encode point cloud attributes. Cohen [8, 9] improved this method by down sampling and block based prediction. Mekuria [10] presented a framework in MPEG-3DG which mapped the color attributes to image grids by octree based traversal. 8-by-8 image grids were constructed, via snake scan, to make good use of legacy image compression methods. This work assumed that subsequent points in depth first traversal, i.e., adjacent points in the point cloud, would share similar features. However, this assumption could be violated, particularly for point cloud with rich colors.

Effective segmentation which gathers points with similar features may improve compression performance. Zhang [11] introduced mean shift algorithm to point cloud segmentation and applied six-dimensional clustering on the point cloud. This method works well in point cloud with small color gamut. However, for point cloud with more complex textures, like typically used human body, this method couldn't separate point cloud effectively. In this paper, we propose a hierarchical segmentation scheme to address these weaknesses and facilitate the subsequent compression.

The main contribution of this paper lies in the hierarchical segmentation scheme for point cloud attribute compression (shown in Fig.1). We introduce the segmentation into photometric space and create clusters by maximizing the color similarity. On this basis, we design a hierarchical segmentation scheme which can exploit the geometry and color attributes of point cloud adequately. Moreover, we use a traversal algorithm and a mapping scheme to fit color attributes into image grids, so that well-developed image compression techniques can be used for raw point cloud data.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

The rest of this paper is organized as follows. Section 2 first introduces the mean shift clustering in high-dimensional space, and then it presents a novel hierarchical segmentation scheme. An octree based traversal method followed by two compression modes is explained in section 3. Experimental results and conclusions are given in section 4 and 5, respectively.

2. HIERARCHICAL POINT CLOUD SEGMENTATION

Point cloud segmentation splits the point cloud into blocks based on geometry, color, and other characteristics, aiming to aggregate points with similar characteristics for each division. Thus, an effective segmentation can significantly improve the intra-cluster compression performance. In this paper, we focus on typical 3D point cloud represented by {x, y, z, r, g, b}, which indicates both geometry (XYZ) and color attributes (RGB) should be taken into account in the segmentation. Due to the sparsity and irregularity of the point cloud, we propose a hierarchical segmentation scheme based on mean shift clustering to meet the requirement and facilitate the subsequent compression.

2.1. High-dimensional mean shift clustering

Mean shift clustering is a hill climbing algorithm which shifts the mean iteratively to a higher density region until convergence [12, 13]. Mean shift clustering adapts well to the irregular distribution nature of the point cloud. Given an input $\mathrm{X} = \{x_1, x_2, \dots, x_n\}, x_i \in R^d$, the kernel density function could be evaluated by

$$\mathrm{f(x)} = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{1}$$

where h is the search bandwidth, $\mathrm{K(x) = ck(\|x\|^2)}$ is the kernel function, and c is the corresponding normalization constant assuring that K(x) integrates to 1. Typically, the Gaussian kernel $\mathrm{k(x)} = \exp(-\frac{x^2}{\sigma^2})$ is adopted.

Mean shift is a gradient ascend algorithm which searches for the nearest stationary point of the density. The gradient of the density function (1) is

$$\nabla\mathrm{f(x)} = \frac{2c}{nh^{d+2}}\sum_{i=1}^{n}(x_i - x)g(\|\frac{x - x_i}{h}\|^2) \tag{2}$$

where $\mathrm{g(x)} = -\mathrm{k}'(\mathrm{x}) = c_1\mathrm{k}(x)$ in Gaussian kernel. The modes of the density function are located at the zeros of the gradient function $\nabla\mathrm{f(x)} = 0$. This function can be solved by iteratively shifting the mean $m_h(x)$ [11-13]:

$$m_h(x) = \frac{\sum_{i=1}^{n} x_i g(\|\frac{x - x_i}{h}\|^2)}{\sum_{i=1}^{n} g(\|\frac{x - x_i}{h}\|^2)} \tag{3}$$

When running the mean shift clustering algorithm, we randomly select a point as initial mean $m_h(x)$, on which an Euclidean distance based neighborhood search is performed. Then we shift the mean to a higher density region according to Eq. (3), and iterate the searching and shifting until the mean converges according to a pre-specified threshold. Repeat all the steps above until all points are clustered. Then clusters are created around the converged means. We merge the newly created clusters with existing one if the means of them are sufficiently close. Besides, another merger, described in 2.2, is designed to fit the hierarchical segmentation scheme.

2.2. Hierarchical segmentation scheme

Benefit from the high-dimensional mean shift algorithm, the point cloud can be effectively segmented. Given a point cloud representation $\{x_i, \mathrm{y}, z_i, r_i, g_i, b_i\}, \mathrm{i} = 1, \dots, \mathrm{n}$, the input X of mean shift algorithm can be $\{x_i, \mathrm{y}, z_i\}$ in 3D geometric space, $\{r_i, g_i, b_i\}$ in 3D photometric space or $\{x_i, \mathrm{y}, z_i, r_i, g_i, b_i\}$ in 6D joint space. However, being applied in different spaces, mean shift algorithm may have different performance. Clustering in geometric space (XYZ) leaves out of the color information, resulting in points with different colors be assigned to the same cluster while clustering in the six-dimensional space (XYZRGB), just like what Zhang did in [11], ignores the difference between geometry and color, resulting in poor performance in boundaries.

To address these problems, a hierarchical segmentation scheme, which exploits geometry and color information adequately, is proposed. Given a point cloud, we first implement the global segmentation which executes mean shift algorithm in photometric space and searches for neighborhoods in the whole point cloud. By shifting the means to higher color density regions, several clusters are generated around the converged means (i.e., the main colors of the point cloud). To avoid the outliers brought by sparse colors, we also test if the number of points of a cluster is less than a pre-specified threshold, if so, we merge it to the one closest to it. Then the global segmentation is completed. The result is shown in Fig.2. Next, we set the local segmentation, which runs mean shift algorithm in geometric space and searches for neighborhoods within the cluster, as the second layer to further split each cluster generated in the first layer. Fig.3 shows the result of the hierarchical segmentation. Since mean shift algorithm is a nonparametric clustering technique, we neither set the number of clusters nor constrain the shape of the clusters.

The global segmentation in the first layer guarantees the color similarity of points within each cluster. For instance, the red and black patterns on the skirt are correctly separated into different clusters in Fig.2. However, the shoes and the red patterns on the skirt in Fig.2 (d), clustered in same cluster, having similar colors, are far away in position. They are successfully separated in Fig.3 (c), which demonstrates the necessity and outstanding performance of the local segmentation in the second layer.

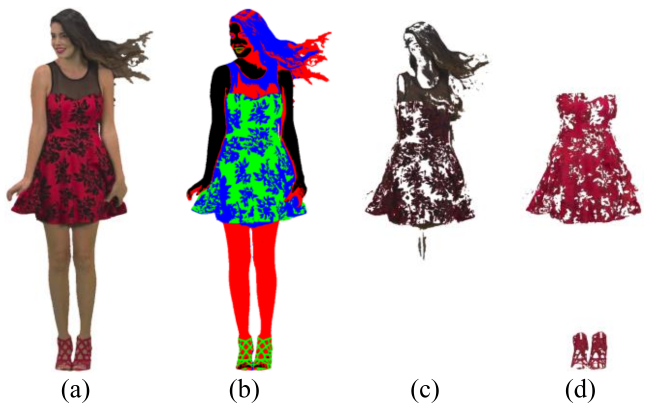


Fig.2. Example of global segmentation. (a) The input point cloud. (b)The overall result of global segmentation labeled by four colors. (c)&(d) Two clusters of (b).

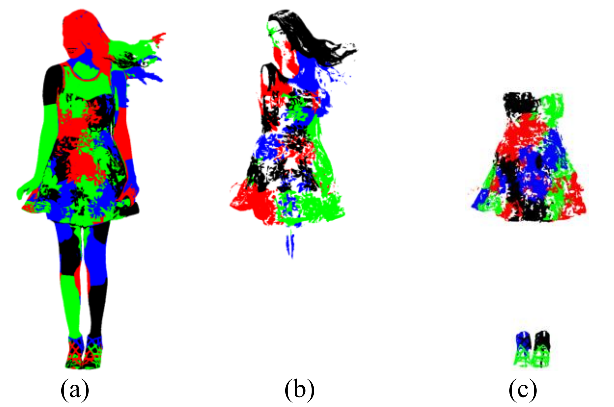


Fig.3. Example of hierarchical segmentation labeled by four colors. (a) The overall segmentation result using the proposed method. (b)&(c) Local segmentation result of Fig.2 (c)&(d).

3. INTRA PREDICTION BASED ATTRIBUTE COMPRESSION

Since the hierarchical segmentation scheme above guarantees the spatial correlation and color similarity of points within each cluster, the assumption that adjacent points share similar attributes becomes reasonable. Nevertheless, how to get the adjacency remains a demanding issue due to the irregular distribution of point cloud. In this section, an octree followed by a traversal algorithm is introduced to figure out proper index of all points, resulting in smooth color stream of each cluster which is well-fitted for the following compression.

3.1. Octree based index and traversal

Octree is a space partition method which recursively decomposes the point cloud-aligned bounding box into eight cells until certain constraint is met. Instead of replacing the points by the cell centers [3-6], we use octree in photometric space to index the point cloud.

Given a cluster generated by the aforementioned hierarchical segmentation, we set the maximum and minimum of each dimension as the corner of the cubic bounding box. The box is then divided into eight cells, and every point in the cluster is assigned into the cell it belongs to. Next, repeat the partition and assignment until every cell contains no more than one point. Then, an octree structure, in which each node represents a cubic cell, is constructed. Furthermore, every point in the cluster can be indexed by a certain non-empty node since each non-empty cell contains only one point.

By traversing the non-empty nodes of the octree, we can obtain the color stream of the cluster. Since points belonged to the same parent node (i.e. they are settled in the same parent cell) tend to have smaller distance than others, the depth-first-search algorithm is employed to traverse the octree, generating smooth color stream of each cluster. Right now, all information of the original point cloud is preserved, as both the segmentation and traversal we adopt are lossless. Therefore, both lossless and lossy compression can be done.

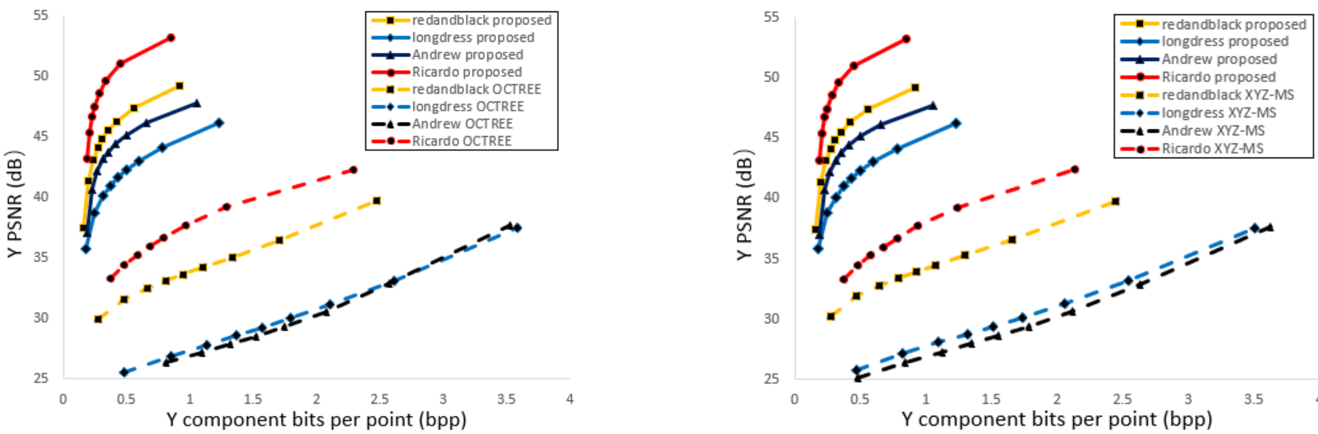
3.2. Attribute compression

For lossless compression, we implement intra-cluster prediction by computing the inter-point color difference to reduce redundancy. After the previous processing, the differences between contiguous points of the color streams are small. Therefore, we utilize the color of previous point as reference to predict the color of current point in the stream. In our experiments, the residuals (prediction error) are concentrated to {-1,0,1}, which verifies the effectiveness of our intra-cluster prediction method. Finally, a Huffman coder is adopted to encode the residuals.

In this paper, we take advantage of the well-developed image compression algorithm to improve the performance of lossy compression. Here, after segmentation and traversal, we map the color streams to uniform 2D grids via a z-scan which enables efficient coding. Then, a 2D-DCT transform is applied on the uniform grid, resulting in one DC and more AC coefficients. Since the color streams are smooth, most of the AC coefficients are small and compression can be achieved by quantizing the coefficients. Finally, run length coding and Huffman coding are carried out in sequence to encode the quantized coefficients.

4. EXPERIMENTAL RESULTS

In this section, we illustrate the performance of the proposed hierarchical segmentation based color attribute compression. The input point clouds are chosen randomly from MPEG test sets [14], i.e., the redandblack and the longdress, typical point cloud for human body containing approximately 800,000 points with geometry and color attributes. Moreover, we test other two point cloud samples, the Andrew and the Ricardo, captured according to [15], to eliminate the impact of data volume. We convert the RGB color to YUV luminance and chrominance values and use the bits per point of Y component to evaluate the coding efficiency. In lossy compression, the color distortion is evaluated by PSNR metric proposed by MPEG-3DG [16].



(a) The proposed method vs. OCTREE. (b) The proposed method vs. mean shift in geometric space

Fig.4. Comparison of experimental results.

Table 1. Experimental results of lossless compression

Input data	Number of points	OCTREE (bpp)	XYZ-MS (bpp)	Proposed (bpp)
longdress	794641	5.43	6.01	**2.43**
redandblack	729133	4.21	4.43	**1.93**
Andrew	283449	5.76	6.04	**1.85**
Ricardo	214748	3.64	3.71	**1.37**

To further verify the performance of our scheme, two contrast experiments are conducted. OCTREE denotes Mekuria's method [10] which constructs octree on the original point cloud directly without segmentation, and the subsequent steps and parameters are the same as our proposed scheme. XYZ-MS denotes applying the mean shift clustering only in geometric space on the basis of OCRTREE.

Here, we implement the segmentation scheme described in section 2 to generate clusters in which points have close positions and similar color attributes. Due to the randomly selected initial mean points and the difference between four data sets, the number of cluster varies from 200 to 600 under the same bandwidth, i.e., 20 of global segmentation, 40 of local segmentation. Then an octree structure, whose leaf nodes contain no more than one point (one or empty), is constructed to index all points of the cluster the octree represented. We employ depth first search algorithm to traverse all the non-empty nodes, and obtain a smooth color stream of each cluster.

For lossless compression, we introduce an intra-cluster prediction algorithm and treats the color of previous point as reference to predict the color of current point in the stream. A simple Huffman coder is adopted to encode the residuals. As shown in Table. 1, the proposed scheme outperforms others over all data sets, and reduces the Y component bits per point (bpp) from 8 to approximately 2, since the residuals are concentrated to {-1,0,1} in our proposed scheme.

For lossy compression, we map the color streams to uniform grids, and leverage the legacy image coding method. In Fig.4, we compared the compression performance between the proposed scheme and other two methods via R-D curves. Different data sets may have different results. For instance, the Ricardo have the best coding efficiency in all experiments. But all the R-D curves show the same tendency that the proposed method has much better compression performance.

5. CONCLUSIONS

In this paper, we proposed an efficient point cloud attribute compression scheme based on hierarchical segmentation. We successively implemented global segmentation in photometric space and local segmentation in geometric space to split point cloud into clusters, in which points share similar features. Then, we conducted octree on each cluster to figure out proper index of all points and introduced a traversal algorithm to obtain smooth attribute streams. An effective intra-cluster prediction method was applied to achieve lossless compression. Meanwhile, we mapped the attribute streams to uniform grids and introduced a DCT based 2D image compression method to achieve satisfying lossy compression performance. In the end, we reconstructed the point cloud and adopted bits per point and PSNR metric of 8-bit luminance Y to evaluate the compression performance. Experimental results showed that our scheme is distinctly superior to contrast methods in both lossless and lossy compression. Note that, even in the lossy compression, all the geometry information is preserved and can be compressed separately by other methods.

In the future, we will optimize the segmentation structure in which we take normal into consideration and design more layers. Besides, exploring an advantageous traversal or mapping scheme which fits image compression algorithm better would be another attempt.

6. ACKNOWLEDGMENT

This paper is supported in part by National Natural Science Foundation of China (61650101), Scientific Research Plan of the Science and Technology Commission of Shanghai Municipality (16511104203, 16Z111010003), and in part by the 111 Program (B07022). The corresponding author is Yiling Xu (e-mail: yl.xu@sjtu.edu.cn).

7. REFERENCES

[1] J. Peng and C. J Kuo, "Geometry-guided progressive lossless 3D mesh coding with octree (OT) decomposition," ACM Trans. Graph. Proceedings of ACM SIGGRAPH, vol. 24, no. 3, pp. 609–616, Jul. 2005.

[2] Y. Huang, J. Peng, and C.J. Kuo, "A generic scheme for progressive point cloud coding," Visualization and Computer Graphics, IEEE Transactions on, vol. 14, no. 2, pp. 440–453, Mar. 2008.

[3] A. Hornung, K.M. Wurm, and M. Bennewitz, "Octomap: An efficient probabilistic 3D mapping framework based on octrees," Autonomous Robots, pp. 189–206, Apr. 2013.

[4] R. Schnabel and R. Klein, "Octree-based point-cloud compression," in Symposium on Point-Based Graphics, Eurographics, pp. 111-120, July. 2006.

[5] J. Kammerl, N. Blodow, and R.B. Rusu, "Real-time compression of point cloud streams," in Robotics and Automation (ICRA), 2012 IEEE International Conference on, pp. 778–785, May. 2012.

[6] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3d point cloud sequences," Image Process, IEEE Transactions on, vol. 25, no. 4, pp. 1765–1778, Feb. 2016.

[7] C. Zhang, D. Florencio, and C. Loop, "Point cloud attribute compression with graph transform," in Image Processing (ICIP), 2014 IEEE International Conference on, pp. 2066–2070, Oct. 2014.

[8] R. A. Cohen, D. Tian, and A. Vetro, "Point cloud attribute compression using 3-D intra prediction and shape adaptive transforms," Data Compression Conference (DCC), IEEE, pp. 141-150, March. 2016.

[9] R. A. Cohen, D. Tian, and A. Vetro, "Attribute compression for sparse point cloud using graph transforms," in Image Processing (ICIP), 2016 IEEE International Conference on, pp. 1374-1378, Sept. 2016.

[10] R. Mekuria, K. Blom and P. Cesar, "Design, Implementation, and Evaluation of a Point Cloud Codec for Tele-Immersive Video," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 27, no. 4, pp. 828-842, Apr 2017.

[11] X. Zhang, and W. Wang, "Six dimensional clustering segmentation of color point cloud," 2016 International Conference on Audio, Language and Image Processing (ICALIP), pp. 423-427, Jul 2016.

[12] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," pattern analysis and machine intelligence, IEEE Transactions on, vol. 24, no. 5, pp. 603-619, May 2002.

[13] B. Georgescu, I. Shimshoni and P. Meer, "Mean Shift Based Clustering in High Dimensions: A Texture Classification Example," Computer Vision, IEEE International Conference on, pp. 456-463, Oct. 2003.

[14] T. Myers, B. Harrison, and P. A. Chou, "8i voxelized full bodies - a voxelized point cloud dataset," In ISO/IEC JTC1/SC29/WG11 (MPEG) input document M40059, Jan. 2017.

[15] C. Loop, C. Zhang, and Z. Zhang, "Real-time high-resolution sparse voxelization with application to image-based modeling," In Proceedings of the 5th High-Performance Graphics Conference (HPG '13). ACM, pp. 73–79, Jul. 2013.

[16] "Current Status on Point Cloud Compression," in ISO/IEC JTC1/SC29/WG11 (MPEG) output document W15869, Oct. 2015