DOI: 10.19729/j.cnki.1673-5188.2018.03.002
http://kns.cnki.net/kcms/detail/34.1294.TN.20180824.1103.002.html, published online August 24, 2018

Introduction to Point Cloud Compression

XU Yiling, ZHANG Ke, HE Lanyi, JIANG Zhiqian, and ZHU Wenjie
(Cooperative MediaNet Innovation Center, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract

Characterized by geometry and photometry attributes, point cloud has been widely applied in the immersive services of various 3D objects and scenes. The development of even more precise capture devices and the increasing requirements for vivid rendering inevitably induce huge point capacity, thus making the point cloud compression a demanding issue. In this paper, we introduce several well-known compression algorithms in the research area as well as the boosting industry standardization works. Specifically, based on various applications of this 3D data, we summarize the static and dynamic point cloud compression, both including irregular geometry and photometry information that represent the spatial structure information and corresponding attributes, respectively. In the end, we conclude the point cloud compression as a promising topic and discuss trends for future works.

Keywords

immersive services; point cloud compression; geometry and photometry

1 Introduction

Emerging immersive media services are capable of providing customers with unprecedented experiences. Representing as omnidirectional videos and 3D point cloud, customers would feel being personally at the scene, personalized viewing perspective and enjoy real-time full interaction. The contents of the immersive media scene may be the shooting of a realistic scene or the synthesis of a virtual scene. Although traditional multimedia applications still play a leading role, the unique immersive presentation and consumption methods of immersive media have attracted tremendous attentions. In the near future, immersive media will form a big market in a variety of areas such as video, games, medical cares and engineering.

The technologies for immersive media have increasingly appealed to both the academic and industrial communities. Among various newly proposed content types, 3D point cloud appears to be one of the most prevalent form of media presentation thanks to the fast development of 3D scanning techniques. 3D point cloud relies on modern measurement methods to record the collection of coordinate data of the surface of the object, and obtains the effect of "stealing truth" in the form of a three-dimensional model. Furthermore, each coordinate can have multiple attributes associated to it, where the attributes may correspond to color, reflectance or other properties of the object/scene that would be associated with a single point. It forms a spatially discrete set of points by sampling point data obtained by camera arrays, laser scanners, etc. which are presented in **Fig. 1**. This media type contains complete information on the surface of the object and the image after reconstruction is the most realistic "replica" of the object. **Fig. 2** shows typical point cloud scenarios. Based on the different applications, 3D point cloud can be well-classified into three categories: static objects and scenes, dynamic objects, and dynamic acquisition.

As described above, 3D Point cloud is an efficient representation as it can be seamlessly integrated and rendered in 3D virtual worlds, establishing a convergence between real and virtual realities and enabling more sophisticated applications. For instance, advances in 3D capture and reconstruction enable real-time generation of highly realistic 3D point cloud representations for 3D tele-presence. They can also improve the visual comfort related to the viewing of content with interactive parallax. Furthermore, they may enhance the performance in geographic information systems, culture heritage and autonomous navigation [1].

2 Challenges

In order to realistically represent the reconstructed scenes, a point cloud may be made up of thousands up to billions of points. This not only results in a huge amount of data, but also causes high complexity in the scattered random distribution of the spatial distribution, which brings great challenges to the storage and transmission system. Hence, more advanced compression coding techniques are needed to significantly reduce the amount of data, combining with the unique consumption

a) Camera arrays b) Laser scanners

GPS: Global Positioning System

▲**Figure 1. Data acquisition devices.**

a) b)

▲**Figure 2. Examples of 3D point cloud.**

characteristics of immersive media. For instance, technologies are needed for lossy compression of point clouds for use in real-time communications and Six Degrees of Freedom (6 DoF) virtual reality. In addition, technology is sought for lossless point cloud compression in the context of dynamic mapping for autonomous driving, cultural heritage applications, etc.

However, the existing coding technologies such as Advanced Video Coding (AVC) and High Efficiency Video Coding (HEVC) mainly focus on two-dimensional video compression. Although a new video coding standards working group for future video coding (FVC) has also been established, which fully considers the unique attributes of immersive media with 360-degree panoramic videos included in test sequences, FVC standards cannot be applied to three-dimensional immersive media such as point cloud yet. As a new three-dimensional spatial data model, point clouds have complex properties such as scattered distribution and time-varying irregularities compared with planar videos. Therefore, point cloud compression coding technology is urgently needed.

3 Technologies

3.1 Static Point Cloud Compression

3.1.1 Octree-Based Point Cloud Compression

In order to deal with irregularly distributed points in 3D space, various decomposition algorithms have been proposed. In fact, the hierarchical tree data structure can effectively describe sparse 3D information. Octree based compression is the most widely used method in the literature. An octree is a tree data structure. Each node subdivides the space into eight nodes [2], [3]. For each octree branch node, one bit is used to represent each child node and called a voxel. This configuration can be effectively represented by one byte, which is considered as the occupancy node based encoding.

As shown in **Fig. 3**, each point is divided into octants when constructing octree on the point cloud. If a node contains more points than the threshold, it is recursively subdivided into eight nodes. Given a point cloud, the corners of the cube bounding box are set to the maximum and minimum values of the input point cloud-aligned bounding box. Then each point is assigned to the node it belongs to. Next, partitions and allocations are repeated until all leaf nodes contain no more than one point. Finally, an octree structure, in which each point is settled, is constructed.

By traversing the tree in different orders and outputting each occupied code encountered, the generated bit stream can be further encoded by an entropy encoding method such as an arithmetic encoder. In this way, the distribution of spatial points can be efficiently coded. In most cases, the points in each eight-leaf node are replaced by the corresponding centroid [4]. The decomposition level determines the accuracy of the data quantification and therefore may result in loss of the encoding. An octree based lossless coding algorithm has been introduced in [5]. In the sense that the quantization coordinates are preserved, it can be considered as lossless, using local surface approximations for compression.

In order to further improve the entropy encoding performance, various schemes are applied to adjust the ergodic sequence between octree voxels. By implementing a breadth-first or depth-first search on an axis-aligned grid, certain sequences

Figure 3. ▶
Octree decomposition.

of voxels can be guaranteed and used as the basis for proper residual calculation [5]–[7]. In addition, experiments have been conducted to increase the flexible traversal order of occupancy codes based on probability reduction order or different leaf node prediction errors on the approximate surface. A prediction tree has been proposed to encode point clouds with potentially serialized point order and reduce redundancy through certain prediction rules [6]. Lossless compression is achieved by exploiting the correlation between the correction vectors, which is the difference between the predicted position and the actual position of the points.

In addition, various schemes are applied to further improve the entropy encoding performance. By implementing the breadth-first or the depth-first search on the axis-aligned grids, certain order of the voxels could be guaranteed and acts as the basis for proper residual computation [5]–[7]. Moreover, trials have been made to promote flexible traversal order for the occupancy code according to a probability descending order or the predicted error of different leaf node to an approximation surface. A prediction tree has been proposed to encode the point cloud by potentially serialized the points orders and reduce the redundancy via certain prediction rules. The lossless compression is achieved by exploiting the correlation between corrective vectors that are the difference between the predicted and real positions of a point.

3.1.2 Binary Tree Based Point Cloud Compression

We develop an efficient point cloud geometry compression scheme via binary tree partition and intra prediction [8]. As shown in **Fig. 4**, we take advantage of the binary tree structure for analyzing the geometry characteristics of the non-uniform data while decomposing the 3D space, providing the basis for the block wise efficient coding scheme. Next, we explore the optimal permutation of the points within specified leaf nodes and realize efficient intra prediction via the extended TSP in 3D space. Further proficiency is obtained for the simple residuals between sequential points. Moreover, the preserved information is largely reduced to only a single reference point for each leaf node. **Fig. 5** illustrates a novel lossless entropy coding tool PAQ which effectively combines the prediction and compression at the same time. The input context is modified to better fit point cloud and a corresponding optimal size is evaluated, achieving preferable compression performance.

3.1.3 Graph Based Point Cloud Compression

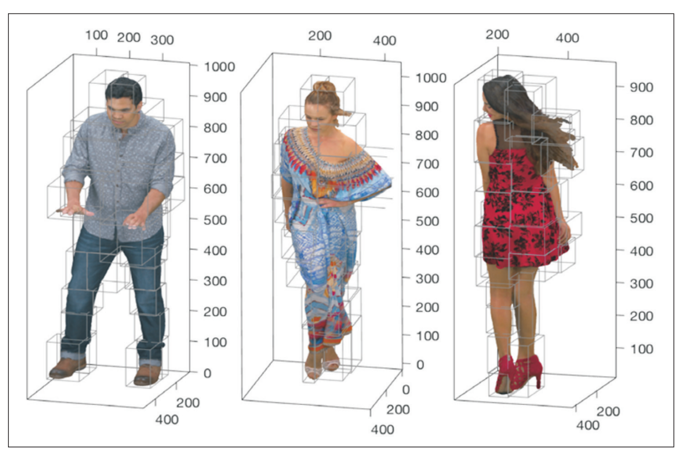
Graph is a data representation which could be used to describe signals in many applications, such as transportation and networks. Graph signal processing has obtained significant attention recently. For example, some methods using graph transform have been proposed to compress color information efficiently. As described in [9], the graph is formed by connecting adjacent points in 3D space. If the distance of two points is less than the threshold, they are defined as adjacent points. The weight of the edge is inversely proportional to the distance between the two adjacent points. Then the adjacency matrix is constructed, which consists of the weights between the adjacent points. Next, the eigenvector matrix of the Laplacian matrix is calculated to implement the attribute transformation. One DC coefficient and one or more AC coefficients are obtained after the graph transformation.

Combined with block-based prediction and shape-adaptive Discrete Cosine Transform (DCT), a compact representation of the points is introduced to tackle the sparsely populated character of point clouds described in [10] and [11]. For the point cloud sequences, the approaches used in video coding like intra-frame prediction, motion estimation and compensation have been introduced into point clouds [10], [11].

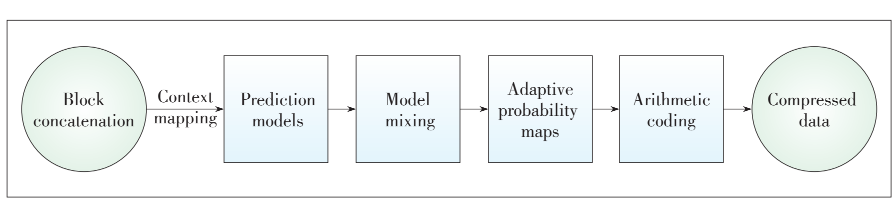
Since key points are distributed irregularly across the images, Tian et al. [12] used graph transform to represent the key point trajectories. This method makes the coding more efficient than traditional DCT based transformation and it is easier for energy compacting.

3.1.4 Clustering Based Point Cloud Compression

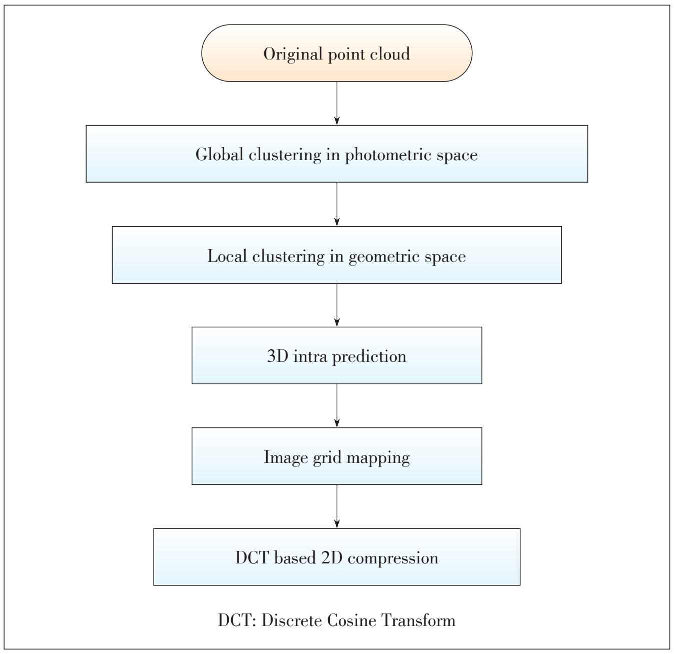
We propose a novel point cloud attribute compression scheme base on clustering [13]. **Fig. 6** shows our compression scheme. Global segmentation is successively implemented in photometric space and local segmentation is conducted in geometric space to split a point cloud into clusters, in which points share similar features [14]. Then, the genetic algorithm based 3D intra prediction is utilized to organize points of each



▲**Figure 4. Binary tree based partition and representation.**



▲**Figure 5. Adapted PAQ8 compression procedure.**



▲**Figure 6. Overview of the clustering based point cloud compression scheme.**

cluster and then all points of each cluster are traversed in the order produced by the intra prediction algorithm. Next, the color attributes of those points are mapped to uniform grids via zigzag scan, which allows us to compress the raw point cloud data without voxelization or other preprocessing methods. A DCT based 2D image compression algorithm is also introduced to achieve impressive lossy compression performance.

3.2 Dynamic Point Cloud Compression

3.2.1 Motion-Compensated Point Cloud Compression

In Philip A Chou's work [15]–[17], the 3D representation of choice is sparse voxel arrays, which they call voxelized point clouds. Neglecting the volumetric aspect of voxels, voxelized point clouds can be considered simply as point clouds whose points are restricted to lie on a regular 3D grid or lattice. For the kinds of data expected in 3D scene capture, voxelized point clouds are a more natural fit than dense voxels arrays, and they obviate the kinds of problems that polygonal meshes have with sampled data. Compared to color and depth maps, voxelized point clouds are a higher level representation, in which redundancies and inconsistencies between overlapping sensor maps have already been removed in a multi-camera sensor fusion step. Compared to arbitrary point clouds, voxelized point clouds have implementation advantages and are highly efficient for real-time processing of captured 3D data.

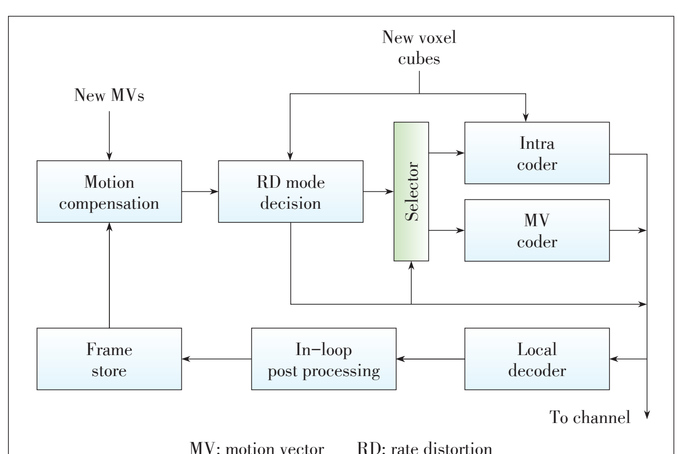
Each representation employs its own compression techniques; they believe graph-based 3D motion estimation and compensation, until recently, represented the state-of-the-art in (voxelized) point cloud color compression, with the former focusing on intra-frame color compression and the latter extending that work to inter-frame color compression. The graph transform is a natural choice for the spatial transform of the color signal due to the irregular domain of definition of the signal. Unfortunately, the graph transform requires repeated eigen-decompositions of many and/or large graph Laplacians, rendering the approach infeasible for real-time processing. They have recently submitted a work on a coder that is able to match or outperform existing intra-frame color compression methods at a reduced cost. Such a point cloud coder is based on a region-adaptive hierarchical transform (RAHT) specially developed for point clouds and is used as a fundamental building block in the present framework for our dynamic point cloud coder, which can be considered as a 3D video coder.

Our objective is to build a coder for dynamic point clouds, which can be implemented in real time with existing technology and is expected to outperform the use of RAHT and octrees to compress color and geometry, respectively. In order to do this, they decided to explore the temporal dimension to remove temporal redundancies, i.e., to explore the fact that the geometry and color of the point cloud may not change much from one frame to another and to use $\mathcal{F}(t)$ as a predictor for $\mathcal{F}(t+1)$. At every discrete time t, the frame $\mathcal{F}(t)=\{Vit\}$, which is represented as a list of voxels in (1).

$$V_{it}=\left[x_{it},y_{it},z_{it},Y_{it},U_{it},V_{it}\right]. \tag{1}$$

Furthermore, they decided to explore 3D analogs of traditional video compression techniques. Motion estimation and motion compensation were used into the compression of dynamic point clouds, in order to achieve higher compression ratios at the expense of lossy coding of the geometry.

The coder (**Fig. 7**) is similar to a traditional video coder in essence, but they are actually quite different in details. In traditional video coders, the frame is broken into blocks of $N\times N$ pixels. However, the frame in the proposed coder is broken into blocks of $N\times N\times N$ voxels, i.e., the voxel space is



▲**Figure 7. Motion-compensated compression encoder.**

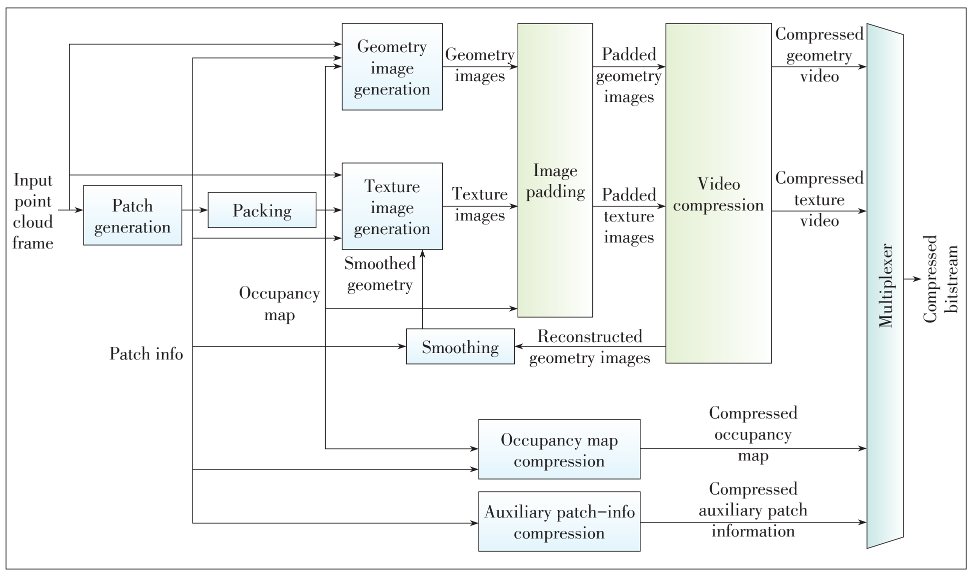
partitioned into blocks and the list of occupied voxels is likewise partitioned into occupied blocks. Therefore, the occupied block at integer position (bx, by, bz) in a frame at instant $t+1$ is composed of occupied voxels $V_{i,t}+1$ within the block boundaries. Unlike traditional video coding, where the pixel position is known and the color is to be encoded, the need to encode the geometry along with the color makes it a distinct problem. So far, the geometry information has not been able to be encoded at a rate significantly lower than 2.5–3.0 bpv, which however can be achieved by using octrees without any prediction from $\mathcal{F}(t)$ to $\mathcal{F}(t+1)$. Therefore, the proposed coder does not encode geometry residuals and operates in two modes: either a block is purely motion compensated or it is entirely encoded in intra mode.

3.2.2 Video-Based Compression

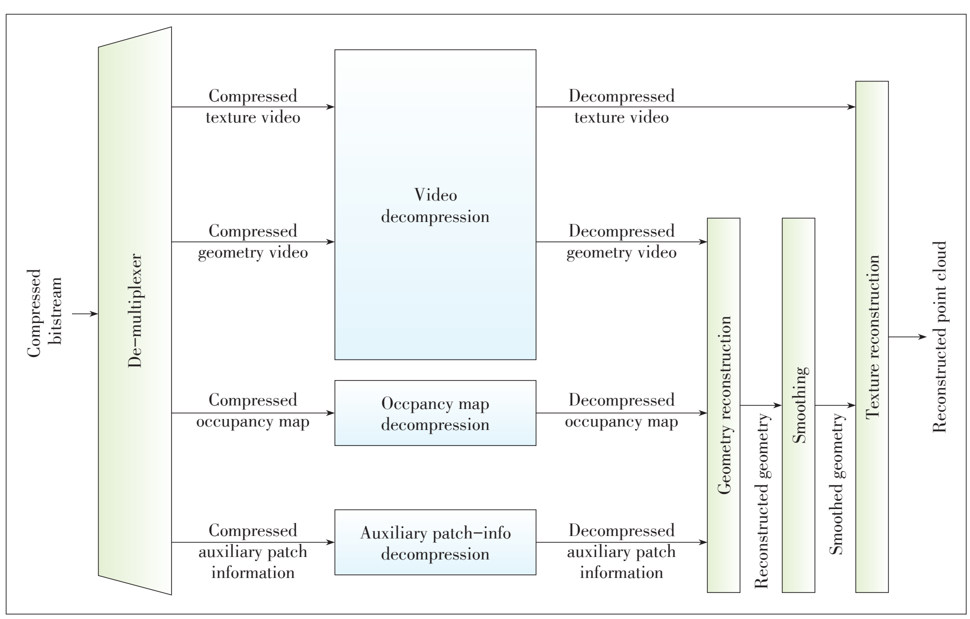
Khaled proposed a video-based point cloud codec to the MPEG PCC group, aiming at test model category 2 (TMC2) [18]. Meanwhile, some studies on point cloud compression based on projection from 3D to 2D have also been proposed [19]–[21]. The main philosophy behind video-based compression is to leverage existing video codecs to compress the geometry and texture information of a dynamic point cloud, by essentially converting the point cloud data into a set of different video sequences. In particular, two video sequences, one for capturing the geometry information of the point cloud data and another for capturing the texture information, are generated and compressed using existing video codecs, e.g. using the HEVC Main profile encoder. Additional metadata that are needed to interpret the two video sequences, i.e., an occupancy map and auxiliary patch information, are also generated and compressed separately. The video generated video bitstreams and the metadata are then multiplexed together so as to generate the final point cloud TMC2 bitstream. **Figs. 8** and **9** provide overviews of the compression and decompression processes implemented in TMC2v0.

The patch generation process aims at decomposing the point cloud into a minimum number of patches with smooth boundaries, while also minimizing the reconstruction error. First, the normal at every point is estimated from the fitting plane of the nearby points. An initial clustering of the point cloud is then obtained by associating each point with one of the following six oriented planes, defined by their normals. The packing process aims at mapping the extracted patches onto a 2D grid, while trying to minimize the unused space and guaranteeing that every TxT (e.g., 16×16) block of the grid is associated with a unique patch.

The image generation process exploits the 3D to 2D mapping computed during the packing process to store the geometry and texture of the point cloud as images. In order to better handle the case of multiple points being projected to the same



▲**Figure 8. Overview of the text model category 2 (TMC2v0) compression process.**



▲**Figure 9. Overview of the text model category 2 (TMC2v0) decompression process.**

pixel, each patch is projected onto two images, referred to as layers. The padding process aims at filling the empty space between patches in order to generate a piecewise smooth image suited for video compression. The occupancy map consists of a binary map that indicates for each cell of the grid whether it belongs to the empty space or to the point cloud. This could be encoded with a precision of a $B_0 \times B_0$ blocks and B_0 is a user-defined parameter. In order to achieve lossless encoding, B_0 should be set to 1. In practice, $B_0=2$ or $B_0=4$ will result in visually acceptable results, while significantly reducing the number of bits required to encode the occupancy map. The generated images/layers are stored as video frames and compressed using the HM16.16 video codec according to the HM configurations provided as parameters.

4 Conclusions

With the rapid development of 3D capture technologies, point clouds have been widely used in many emerging applications such as augmented reality and autonomous driving. However, a point cloud, used to represent real world objects in these applications, may contain millions of points, which results in huge data volume. Therefore, efficient point cloud compression algorithms are essential for reducing bandwidth and storage consumption.

References

[1] C. Tulvan, R. N. Mekuria, Z. Li, S. Laserre, "Use cases for point cloud compression," ISO/IEC JTC1/SC29 WG11 MPEG Output Document, 2016.
[2] J. Peng and C.-C. J. Kuo, "Geometry-guided progressive lossless 3D mesh coding with octree (OT) decomposition," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 609–616, Jul. 2005. doi: 10.1145/1073204.1073237.
[3] Y. Huang, J. Peng, C.-C. J. Kuo, and M. Gopi, "A generic scheme for progressive point cloud coding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 440–453, Mar. 2008. doi: 10.1109/TVCG.2007.70441.
[4] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: an efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, Apr. 2013. doi: 10.1007/s10514-012-9321-0.
[5] R. Schnabel and R. Klein, "Octree-based point-cloud compression," in *3rd Eurographics/IEEE VGTC Conference on Point-Based Graphics*, Boston, USA, 2006, pp. 111–120. doi: 10.2312/SPBG/SPBG06/111-120.
[6] S. Gumhold, Z. Kami, M. Isenburg, and H.-P. Seidel, "Predictive point-cloud compression," in *ACM SIGGRAPH 2005 Sketches*, New York, USA, 2005. doi: 10.1145/1187112.1187277.
[7] Y. Huang, J. Peng, C.-C. J. Kuo, and M. Gopi, "A generic scheme for progressive point cloud coding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 440–453, Jan. 2008. doi: 10.1109/TVCG.2007.70441.
[8] W. Zhu, Y. Xu, L. Li, and Z. Li, "Lossless point cloud geometry compression via binary tree partition and intra prediction," in *IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, Luton, UK, 2017. doi: 10.1109/MMSP.2017.8122226.
[9] C. Zhang, D. Florencio, and C. Loop, "Point cloud attribute compression with graph transform," in *IEEE International Conference on Image Processing (ICIP)*, Paris, France, 2014, pp. 2066–2070. doi: 10.1109/ICIP.2014.7025414.
[10] R. A. Cohen, D. Tian, and A. Vetro, "Attribute compression for sparse point clouds using graph transforms," in *IEEE International Conference on Image Processing (ICIP)*, Phoenix, USA, 2016. doi: 10.1109/ICIP.2016.7532583.
[11] R. A. Cohen, D. Tian, and A. Vetro, "Point cloud attribute compression using 3-D intra prediction and shape-adaptive transforms," in *IEEE Data Compression Conference (DCC)*, Snowbird, USA, 2016., 2016. doi: 10.1109/DCC.2016.67.
[12] D. Tian, H. Sun, and A. Vetro, "Graph transformation for keypoint trajectory coding," *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Washington, DC, USA, 2016. doi: 10.1109/GlobalSIP.2016.7905881.
[13] K. Zhang, W. Zhu, and Y. Xu, "point cloud attribute compression via clustering and intra prediction," to be published.
[14] K. Zhang, W. Zhu, and Y. Xu, "Hierarchical segmentation based point cloud attribute compression," to be published.
[15] R. L. de Queiroz and P. A. Chou, "Motion-compensated compression of dynamic voxelized point clouds," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 3886–3895, Aug. 2017. doi: 10.1109/TIP.2017.2707807.
[16] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based motion estimation and compensation for dynamic 3D point cloud compression," in *IEEE International Conference on Image Processing (ICIP)*, Quebec City, Canada, 2015, pp. 3235–3239. doi: 10.1109/ICIP.2015.7351401.
[17] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1765–1778, Apr. 2016. doi: 10.1109/TIP.2016.2529506.
[18] "PCC Test Model Category 2 v0", ISO/IEC JTC1/SC29/WG11 w17248, Macau, China, 2017, 10.
[19] L. He, W. Zhu, and Y. Xu, "Best-effort projection based attribute compression for 3D point cloud," in *Asia-Pacific Conference on Communications (APCC)*, Perth, Australia, 2017.
[20] Nokia, "Nokia's response to cfp for point cloud compression (category 2)," ISO/IEC JTC1/SC29 WG11 Doc.m41779, Macau, China, Oct. 2017.
[21] Technicolor, "Technicolor's response to the cfp for point cloud compression," ISO/IEC JTC1/SC29 WG11 Doc.m41822, Macau, China, Oct. 2017.

Manuscript received: 2018-03-29

Biographies

XU Yiling received her B.S., M.S., and Ph.D. from the University of Electronic Science and Technology of China in 1999, 2001, and 2004, respectively. She is a full researcher at Shanghai Jiaotong University, China. From 2004 to 2013, she was with the Multimedia Communication Research Institute of Samsung Electronics, Korea. Her research interests mainly include architecture design for next generation multimedia systems, cross-layer design, and dynamic adaptation for heterogeneous networks.

ZHANG Ke (zhang_ke@sjtu.edu.cn) received the B.S. degree from Nanjing University of Science and Technology, China in 2016. He is pursuing the master degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interest is about point cloud compression and video coding.

HE Lanyi (hly92711@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Xidian University, China in 2015. He is pursuing the master degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interest is about point cloud coding.

JIANG Zhiqian (zhiqjiang@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Xidian University, China in 2015. He is pursuing the Ph.D degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. His research interests include reliable transmission and immersive media transmission.

ZHU Wenjie (rebecca28@sjtu.edu.cn) received the B.S. degree in electronics and information engineering from Harbin Institute of Technology, China in 2014. She is pursuing the Ph.D degree in the Cooperative Medianet Innovation Center of Shanghai Jiao Tong University, China. Currently, she is studying in University of Missouri, Kansas City as an exchange scholar under the support of China Scholarship Council. Her research interests including point cloud compression and immersive media transportation. She has published around 6 conference papers and has 4 patents in processing with 3 MPEG proposals been accepted in international standard.