Model-Based Encoding Parameter Optimization for 3D Point Cloud Compression

Qi Liu*†, Hui Yuan (corresponding author)†*, Junhui Hou‡, Hao Liu*, Raouf Hamzaoui§
* School of Information Science and Engineering, Shandong University, Shandong, China
E-mail: sdqi.liu@gmail.com, yuanhui0325@gmail.com, liuhaoxb@gmail.com
† Shenzhen Research Institute of Shandong University, Shenzhen, China
E-mail: yuanhui0325@gmail.com, sdqi.liu@gmail.com
‡ Department of Computer Science, City University of Hong Kong, Hongkong, China
E-mail: jh.hou@cityu.edu.hk
§ School of Engineering and Sustainable Development, De Montfort University, Leicester, UK
E-mail: rhamzaoui@dmu.ac.uk

***Abstract*—Rate-distortion optimal 3D point cloud compression is very challenging due to the irregular structure of 3D point clouds. For a popular 3D point cloud codec that uses octrees for geometry compression and JPEG for color compression, we first find analytical models that describe the relationship between the encoding parameters and the bitrate and distortion, respectively. We then use our models to formulate the rate-distortion optimization problem as a constrained convex optimization problem and apply an interior point method to solve it. Experimental results for six 3D point clouds show that our technique gives similar results to exhaustive search at only about 1.57% of its computational cost.**

I. Introduction

With the increasing capability of 3D data acquisition devices, 3D point clouds have recently emerged as an effective way to represent objects. A 3D point cloud consists of a set of 3D coordinates indicating the locations of points, along with one or more attributes (e.g., normals or colors). 3D point clouds are becoming more and more popular in emerging applications such as augmented reality [1], 3D telepresence [2] and mobile robots [3]. However, their widespread use is hindered by several challenges. In particular, high-quality point clouds may contain millions of points, making their processing, storage and transmission challenging. For this reason, efficient compression algorithms have to be developed for 3D point clouds to accommodate existing network bandwidth and storage capacity.

3D point clouds exhibit redundancy in both geometry and attribute information. Initially, most of the works [4-7] focused on the compression of geometry information. Among them, the octree decomposition method [4] has been used extensively because of its efficiency and low-complexity. For the bounding cube of a 3D point cloud that is to be compressed, an octree is constructed for a given maximum octree level (corresponding to the depth of the octree and denoted by L in the remainder of the paper). The bounding cube is then partitioned into $2^L \times 2^L \times 2^L$ voxels. The content of each voxel can be determined by verifying whether there are points inside the voxel. The maximum octree level determines the precision of the geometry information, i.e, the number of voxels to be encoded. Jiang *et al.* [5] proposed an octree-based progressive 3D point cloud coder where the geometry information is efficiently compressed by optimizing the order in which the child cells are traversed. Ochotta and Saupe [6] partition the point cloud in a number of point clusters. A surface patch is associated to each cluster and parameterized as a height field, which is efficiently encoded with a shape-adaptive wavelet coder. Ahn *et al.* [7] proposed an adaptive range image coding algorithm for the geometry compression of large-scale 3D point clouds. In this method, a 3D point cloud is first partitioned into blocks of various sizes. Then, each block is encoded by selecting one prediction mode from twelve candidates.

Compression of the attribute information has recently gained more attention. Unlike a 2D image, a 3D point cloud has an irregular data structure. Therefore, to compress the attribute information (especially color), many works used special transforms that are suitable for irregular data structures, e.g., shape-adaptive discrete cosine transform [8][9], graph transform [10][11], Gaussian process transform [12][13], and Haar wavelet-based region-adaptive hierarchical transform [14]. Another approach to compress the color attributes was proposed by Hou *et al.* [15]. The main idea is to use a virtual adaptive sampling process so that the task can be expressed as an l_0-norm regularized optimization problem. Instead of compressing the irregular data directly, some methods [16][17][18] map the irregular data to regular data for convenient compression. Mekuria, Blom, and Cesar [16] applied a depth-first tree traversal to read the color attributes from the octree and used a zig-zag scan to map them to 8 x 8 blocks of a 2D grid. Correlation between the color attributes was then exploited by compressing the grid with JPEG. Similarly, Tu *et al.* [17] converted the point cloud data into range images which were then compressed using either JPEG or MPEG-4. In addition, the rotation position vectors were compressed with run-length coding. Cui, Xu and Jang [18] also grouped a point cloud into blocks that were compressed by selecting the optimal coding method from two predefined methods.

As the 3D point cloud format became widely used in prac-

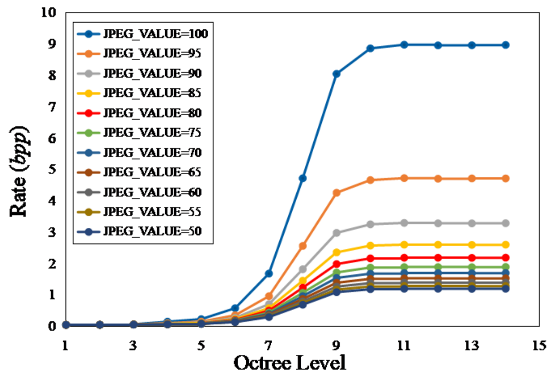


Fig. 1. Relationship between the bit rates, the maximum octree levels, and the JPEG_VALUEs for the *Alex* point cloud set.

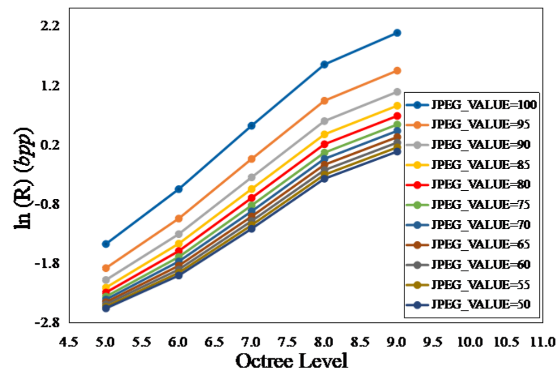


Fig. 2. Relationship between the logarithm of the bitrates, the maximum octree levels, and the JPEG_VALUEs for the *Alex* point cloud set.

tical applications, a fully functional testing platform known as the point cloud library (PCL-PCC) [19] emerged and was initially adopted by MPEG for verification experiments. For the PCL-PCC platform, the color distortion depends on both the maximum octree level which affects the number of coded voxels and the quantization parameter (called JPEG_VALUE) which affects the coding errors of voxels. Different combinations of the maximum octree level and JPEG_VALUE give different bitrates and reconstruction qualities.

In this paper, we focus on the PCL-PCC platform and address the problem of how to determine the optimal coding parameters, i.e., the maximum octree level and the JPEG_VALUE, subject to a constraint on the target bitrate. We use curve fitting to build analytical models for the rate and distortion of the PCL-PCC 3D point cloud coder. We then formulate the problem as a constrained optimization problem and use an interior point method to solve it. Experimental results show that our approach gives similar results to the optimal ones obtained with exhaustive search at a fraction of the computational cost.

The rest of the paper is organized as follows. Rate and distortion models for PCL-PCC compression are proposed in Section II. The optimal bit allocation (or coding parameter determination) problem is formulated as a convex optimization problem and solved by an interior point method in Section III. Experimental results and conclusions are given in Section IV and V, respectively.

II. RATE AND DISTORTION MODEL DERIVATION

In this section, we use statistical analysis to derive rate and distortion models for the PCL-PCC platform. Compression with this platform starts by carrying out an octree decomposition. The predefined maximum octree level determines the number of coded voxels and thus highly affects the bitrate and reconstructed quality of a 3D point cloud. Then the color values are mapped onto a 2D image and encoded by a JPEG encoder in which the quantization parameters are represented by the parameter "JPEG_VALUE" (a large JPEG_VALUE corresponds to a small quantization error). For a given target bitrate, in order to determine the optimal coding parameters, i.e., the maximum octree level and the JPEG_VALUE, the rate and distortion models must be determined.

As Fig.1 shows, the rate is nearly constant when the maximum octree level is either too big or too small. Therefore, we only considered the range 5 to 9 for the octree level. Similarly, since an unacceptable quality deterioration will occur with small JPEG_VALUEs, only those ranging from 50 to 100 were considered.

A. Rate model derivation

Fig. 2 shows the relationship between the logarithm of the bitrate (given by the average number of bits per point, *bpp*), the maximum octree level, and the JPEG_VALUE. We observe that there is an approximately linear relationship between the logarithm of the bitrate and the maximum octree level for a fixed JPEG_VALUE, that is,

$$lnR = a_0 L + b_0, \tag{1}$$

where L denotes the maximum octree level, R represents the bitrate, and a_0 and b_0 are model parameters. Table I, which was obtained by curve fitting, shows that the squared correlation coefficient (SCC) of the linear relationship (1) between lnR and L is greater than or equal to 0.98 and up to 1 in some cases. Furthermore, the parameter b_0 is almost constant for a given 3D point cloud. On the other hand, the parameter a_0 depends on the JPEG_VALUE (denoted by J). Therefore, we further analyzed the relationship between a_0 and J. As Fig. 3 shows, there is an approximate linear relationship between a_0 and J:

$$a_0 = a_1 J + b_1, \tag{2}$$

where the SCC is always greater than or equal to 0.93. Based on (1) and (2), we can express the rate model as

$$lnR = aLJ + bL + c, \tag{3}$$

where $a = a_1$, $b = b_1$, and $c = b_0$ are the model parameters. For different 3D point clouds, the model parameters (a, b, and c) and the SCC between the actual logarithm of the bitrate and

TABLE I
RATE MODEL DATA

3D Point Cloud Data	JPEG _VALUE	$lnR = a_0L+b_0$			$a_0 = a_1J+b_1$		
		a_0	b_0	SCC	a_1	b_1	SCC
Alex	50	0.69	-6.05	0.99			
	55	0.70	-6.07	0.99			
	60	0.71	-6.08	0.99			
	65	0.73	-6.11	0.99			
	70	0.74	-6.13	0.99			
	75	0.76	-6.14	0.99	0.0042	0.4590	0.93
	80	0.78	-6.16	0.99			
	85	0.80	-6.18	0.99			
	90	0.82	-6.18	0.99			
	95	0.86	-6.16	0.99			
	100	0.92	-6.02	0.99			
Andrew	50	0.75	-6.57	0.98			
	55	0.77	-6.62	0.98			
	60	0.78	-6.65	0.98			
	65	0.80	-6.69	0.98			
	70	0.82	-6.73	0.98			
	75	0.83	-6.77	0.98	0.0047	0.4980	0.94
	80	0.86	-6.80	0.98			
	85	0.88	-6.83	0.98			
	90	0.91	-6.86	0.98			
	95	0.95	-6.85	0.98			
	100	1.01	-6.71	0.98			
Phil	50	0.81	-7.13	0.99			
	55	0.82	-7.14	0.99			
	60	0.83	-7.17	0.99			
	65	0.85	-7.21	0.99			
	70	0.86	-7.25	0.99			
	75	0.88	-7.28	0.99	0.0049	0.5325	0.93
	80	0.90	-7.32	0.99			
	85	0.93	-7.37	0.99			
	90	0.97	-7.42	0.99			
	95	1.01	-7.45	0.99			
	100	1.07	-7.30	0.99			
Soldier	50	0.92	-8.86	0.99			
	55	0.93	-8.87	0.99			
	60	0.94	-8.90	0.99			
	65	0.96	-8.93	0.99			
	70	0.97	-8.95	1.00			
	75	0.99	-8.96	1.00	0.0049	0.6495	0.93
	80	1.01	-8.99	1.00			
	85	1.04	-9.04	1.00			
	90	1.08	-9.10	1.00			
	95	1.13	-9.15	1.00			
	100	1.18	-8.96	1.00			

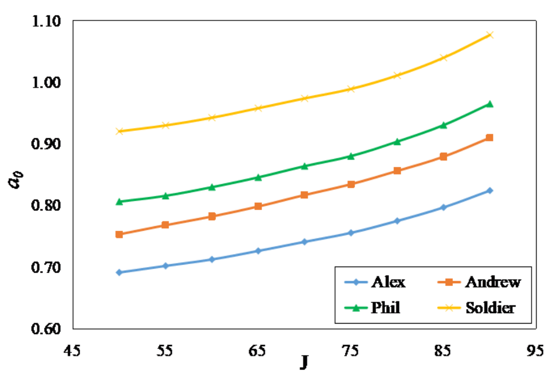
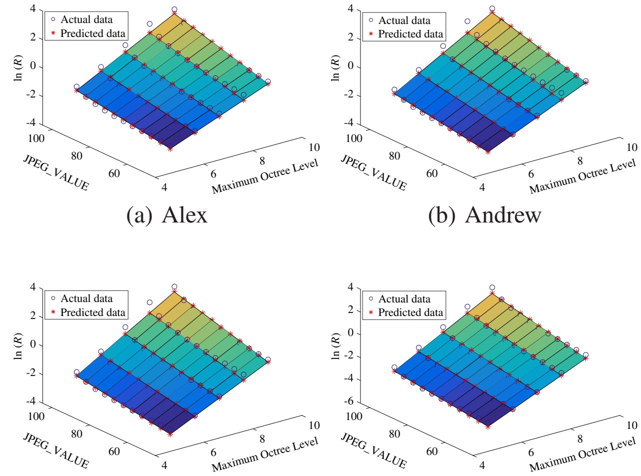


Fig. 3. Relationship between a_0 and J.



(a) Alex (b) Andrew

(c) Phil (d) Soldier

Fig. 4. Accuracy of the proposed rate model.

the fitted ones are provided in Table II. We see that the SCC of all the tested 3D point cloud sets are larger than 0.96, which indicates that the derived rate model is accurate. As the model parameter a is almost constant in four point cloud sets, we fix it to 0.0041, which corresponds to the average value for the four point cloud sets. Fig. 4 shows the actual logarithm of the bitrate and the fitted ones with respect to different maximum octree levels and JPEG_VALUEs from which we conclude that the rate model is accurate enough.

TABLE II
PARAMETERS AND SCC OF FITTED RATE AND DISTORTION MODELS

3D Point Cloud Data	Rate Model				Distortion Model			
	a	b	c	SCC	s	p	q	SCC
Alex	0.0040	0.4718	-6.1184	0.97	1781451.81	-0.6895	-3.1634	0.97
Andrew	0.0041	0.5449	-6.7334	0.96	990821.52	-0.7438	-2.8603	0.96
Phil	0.0041	0.5943	-7.2767	0.97	2088343.24	-0.7288	-3.3123	0.97
Soldier	0.0042	0.6954	-8.9743	0.98	1053707.72	-0.8018	-2.9315	0.97

B. Distortion model derivation

We measure the distortion between the original point cloud v_{or} and the reconstructed point cloud v_{re} using the square of color difference [20]

$$D\left(v_{or}, v_{re}\right) = \frac{1}{K}\sum_{v_i \in v_{or}} \left\| y\left(v_i\right) - y\left(v_{nn\_re}\right)\right\|_2^2, \quad (4)$$

where v_i is a point in the original cloud, v_{nn_re} is the nearest neighboring point of the original point in the reconstructed point cloud, $y\left(v_i\right)$ and $y\left(v_{nn\_re}\right)$ are the luminance values of the original point and the reconstructed point respectively, and K is the number of points in the original point cloud. Fig. 5 shows the relationship between the coding distortion, the maximum octree level, and the JPEG_VALUE. We observe that there exists a power function relationship between

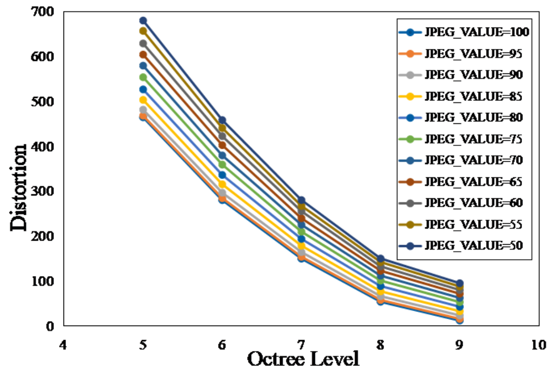


Fig. 5. Relationship between the coding distortions, the maximum octree levels and the JPEG_VALUEs for the *Alex* point cloud set.

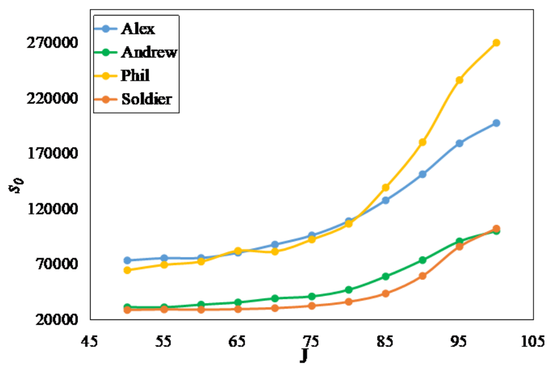


Fig. 6. Relationship between s_0 and J.

coding distortions and the maximum octree levels for a fixed JPEG_VALUE, as given in (5):

$$D = s_0 L^q, \tag{5}$$

where D is the distortion, and s_0 and q are model parameters. From Table III, the SCC of the estimated distortion and the actual distortion is greater than or equal to 0.94. Besides, we can also observe that s_0 is related to the JPEG_VALUE. Accordingly, we analyzed the relationship between s_0 and J for each point cloud set. As Fig. 6 shows, there exists a power relationship between s_0 and J

$$s_0 = sJ^p, \tag{6}$$

where p is a model parameter. Therefore, we can write the distortion model as

$$D = sJ^p L^q, \tag{7}$$

where the model parameters are obtained by data fitting. Table II shows the model parameters (s, p, and q) and the SCC between the actual coding distortion and the fitted ones for various 3D point clouds. We can see that the SCC of all the tested 3D point clouds sets are greater than or equal to 0.96, which indicates that the derived distortion model is accurate. The actual distortion and the fitted ones with respect to different maximum octree levels and JPEG_VALUEs are shown in Fig. 7 from which we can conclude that the proposed distortion model is accurate.

TABLE III
DISTORTION MODEL DATA

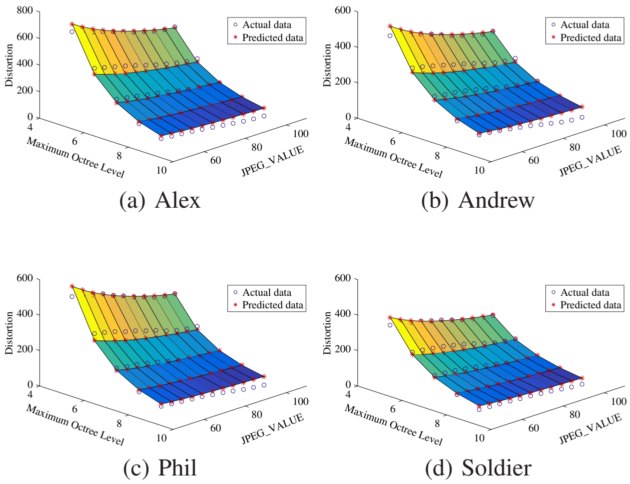
3D Point Cloud Data	JPEG _VALUE	$D = s_0 L^q$			$s_0 = sJ^p$		
		s_0	q	SCC	s	p	SCC
	50	73507.7770	-2.89	0.98			
	55	75560.1424	-2.93	0.98			
	60	75706.5524	-2.96	0.98			
	65	80469.8034	-3.02	0.98			
	70	87815.1311	-3.10	0.98			
Alex	75	96107.1040	-3.19	0.98	178.4400	1.1897	0.91
	80	108961.0318	-3.29	0.98			
	85	127746.7394	-3.42	0.98			
	90	151275.4089	-3.55	0.97			
	95	179191.7992	-3.68	0.97			
	100	197569.8356	-3.74	0.97			
	50	31237.2783	-2.57	0.98			
	55	31242.6784	-2.59	0.98			
	60	33574.7083	-2.65	0.98			
	65	35615.6504	-2.71	0.97			
	70	39180.2130	-2.80	0.97			
Andrew	75	40982.8189	-2.86	0.97	26.6820	1.7477	0.90
	80	47017.7191	-2.98	0.96			
	85	59029.9400	-3.15	0.96			
	90	73791.4810	-3.32	0.95			
	95	90734.3211	-3.47	0.95			
	100	100090.0556	-3.54	0.94			
	50	64680.4298	-2.97	0.98			
	55	69540.9491	-3.03	0.98			
	60	72483.6502	-3.08	0.98			
	65	82190.6743	-3.17	0.98			
	70	81499.5603	-3.20	0.98			
Phil	75	92370.9626	-3.31	0.98	16.1160	2.0598	0.89
	80	106546.1887	-3.44	0.98			
	85	139332.9319	-3.63	0.98			
	90	180240.9460	-3.82	0.97			
	95	236313.6790	-4.01	0.97			
	100	270054.1450	-4.10	0.97			
	50	28779.9676	-2.69	0.98			
	55	29277.4323	-2.72	0.98			
	60	29032.1008	-2.74	0.97			
	65	29523.4915	-2.77	0.97			
	70	30439.6636	-2.82	0.97			
Soldier	75	32529.1560	-2.89	0.98	24.9650	1.7256	0.77
	80	36188.2805	-3.00	0.98			
	85	43634.0801	-3.16	0.98			
	90	59534.3206	-3.39	0.98			
	95	85975.6748	-3.65	0.97			
	100	102240.5069	-3.76	0.97			

III. OPTIMAL CODING PARAMETER DETERMINATION

The goal of 3D point cloud compression is to maximize the reconstruction quality of the 3D point cloud subject to a constraint on the bitrate. The reconstruction quality of a 3D point cloud is determined by both the number of coded voxels and the quantization errors. For the PCL-PCC platform, the number of coded voxels depends on the maximum octree level L, while the quantization errors depend on the JPEG_VALUE J. Therefore, the problem can be formulated as the constrained optimization problem

$$\begin{aligned} &\min_{L,J} \; D(L,J) \\ &s.t. \quad R(L,J) \le R_t, \end{aligned} \tag{8}$$

where L ranges from 5 to 9, J ranges from 50 to 100, and R_t is the target bitrate. Based on the derived rate model and distortion model, the optimization problem (8) can be



(a) Alex (b) Andrew

(c) Phil (d) Soldier

Fig. 7. Accuracy of the proposed distortion model.

reformulated as

$$\min_{L,J} \; sL^qJ^p$$
$$s.t. \begin{cases} 5 \leq L \leq 9 \\ 50 \leq J \leq 100 \\ exp\{aLJ + bL + c\} \leq R_t. \end{cases} \tag{9}$$

To solve (9), we need first to determine the model parameters. As mentioned in Section II.A, we fix the value of a to 0.0041. The other model parameters, i.e., b, c, s, p, and q, are obtained by pre-encoding the given 3D point cloud with the four pairs of coding parameters $(L, J) \in \{(5, 90), (7, 50), (7, 70), (8, 80)\}$. Then, the parameters s, p and q are computed by solving the equations:

$$\begin{cases} ln(D(5, 90)) = ln(s) + q \times ln(5) + p \times ln(90) \\ ln(D(7, 50)) = ln(s) + q \times ln(7) + p \times ln(50) \\ ln(D(8, 80)) = ln(s) + q \times ln(8) + p \times ln(80). \end{cases} \tag{10}$$

Similarly, the parameters b and c can be obtained by solving the equations:

$$\begin{cases} ln(R(5, 90)) = 0.0041 \times 5 \times 90 + b \times \; 5 + c \\ ln(R(7, 70)) = 0.0041 \times 7 \times 70 + b \times \; 7 + c. \end{cases} \tag{11}$$

Table IV shows the SCC between the actual logarithm of the bitrate and the estimated ones that are calculated by the estimated model parameters in terms of the proposed rate model. In addition, the SCC between the actual logarithms of the distortion and the estimated ones are also provided. We can see that all SCCs are larger than 0.81, indicating accuracy of the estimated model parameters. Based on the estimated model parameters, the optimization problem is solved by an interior point method [20] [21] in which the convex optimization problem with inequality constraints is first converted to a convex optimization problem with no constraints by a barrier function and then solved with Newton's method.

TABLE IV
SCC BETWEEN THE LOGARITHM OF THE ACTUAL RATE (RESP. DISTORTION) AND THE ESTIMATED ONES CALCULATED FROM (3) AND (7) USING (10) AND (11).

3D Point Cloud Data	SCC between actual $\ln(R)$ and estimated $\ln(R)$	SCC between actual $\ln(D)$ and estimated $\ln(D)$
Alex	0.9767	0.9227
Andrew	0.9677	0.8138
Phil	0.9775	0.8816
Soldier	0.9860	0.9061

IV. EXPERIMENTAL RESULTS

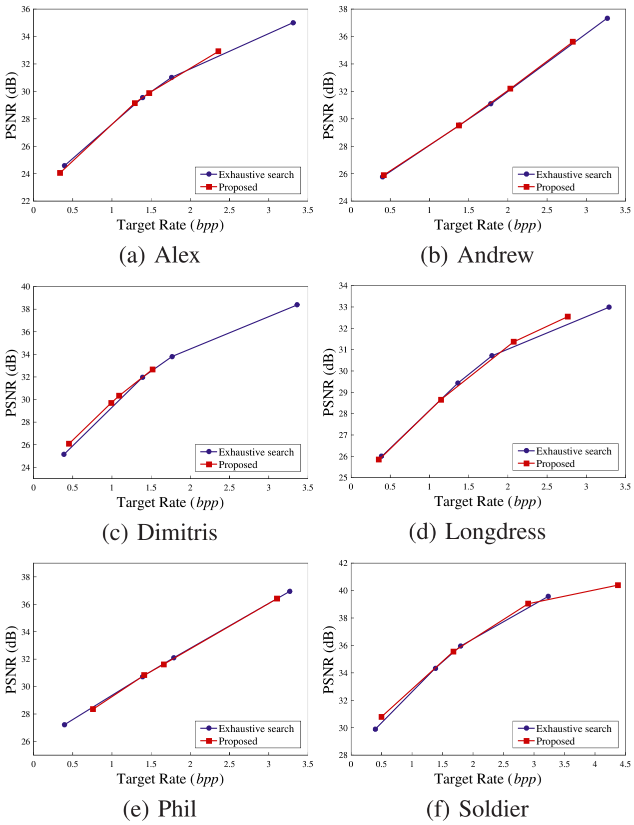
In this section, we verify the performance of the proposed algorithm. Six 3D point clouds [22], namely, *Alex*, *Andrew*, *Dimitris*, *Longdress*, *Phil*, and *Soldier* were used for the experiments. The target bitrates R_t were 0.4 bpp, 1.4 bpp, 1.8 bpp, and 3.4 bpp. Exhaustive search was used as the benchmark. In exhaustive search, a 3D point cloud was first encoded by all the possible combinations of maximum octree levels L and JPEG_VALUEs J. Then, the set $\mathbb{S} = \{(L, J)|R(L, J) \leqslant R_t\}$ was determined. Finally, the combination (L^{s_opt}, J^{s_opt}) that gives the minimum distortion was selected from the set $\mathbb{S}$. To derive the rate and distortion models, point clouds are encoded with maximum octree levels L and JPEG_VALUEs J pairs (5,90), (7,50), (7,70), and (8,80) by the PCL-PCC platform. The distortion model parameters s, p and q are computed by solving equations (10) and the rate model parameters b and c are obtained by solving equations (11). Given a target bitrate, we solve (9) to obtain the optimal maximum octree level L and JPEG_VALUE J. To evaluate the performance of the proposed algorithm, we compared the rate-PSNR curve of the proposed algorithm and the exhaustive search algorithm (Fig. 8). The PSNR is computed by (12):

$$PSNR = 10 \times log_{10}\left(\frac{255^2}{D}\right), \tag{12}$$

where the distortion D is derived from (4). We can observe that the performance of the model-based algorithm is very close to that of exhaustive search. In the experiment, since the maximum octree level ranged from 5 to 9, and the JPEG_VALUE ranged from 50 to 100, a 3D point cloud was encoded $5 \times 51 = 255$ times to find the optimal L and J with exhaustive search. In contrast, only four pre-encodings were required by our method to compute the model parameters and obtain the optimal L and J by the interior point method. Thus, the time complexity of the proposed algorithm was only about 1.57% of that of exhaustive search. Moreover, it should be noted that the time complexity of the proposed method mainly depends on the encoding procedure, not the interior point method. Take *Alex* as an example, the time spent by the interior point method is only 0.5% of that required by the encoding procedure.

V. CONCLUSION

We proposed a model-based technique to efficiently determine the optimal coding parameters for PCL-PCC 3D point



(a) Alex (b) Andrew (c) Dimitris (d) Longdress (e) Phil (f) Soldier

Fig. 8. PSNR vs. target bitrate for the proposed algorithm and exhaustive search.

cloud compression. Rate and distortion models with respect to the maximum octree level and JPEG_VALUE were first derived and verified by statistical analysis. Then, based on the rate and distortion models, the bit allocation problem was converted to a convex optimization problem that was solved by an interior point method. Model parameters were derived with a small number of pre-encodings. In order to evaluate the performance of the proposed algorithm, we compared it to exhaustive search. Experimental results showed that the rate-distortion performance of the proposed method is very close to that of exhaustive search, while its time complexity is about 63 times lower.

VI. ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China under Grants 61571274 and 61871342, in part by the Shandong Natural Science Funds for Distinguished Young Scholar under Grant JQ201614, in part by the Shandong Provincial Key Research and Development Plan under Grant 2017CXGC1504, in part by Shenzhen Science and Technology Research and Development Funds under Grant JCYJ20170818103244664, and in part by the Young Scholars Program of Shandong University (YSPSDU) under Grant 2015WLJH39.

REFERENCES

[1] W. Guan, S. You and U. Neumann, "Recognition-driven 3D navigation in large-scale virtual environments," in *Proc. IEEE Virtual Reality Conference*, Singapore, 2011, pp. 71-74.

[2] H. Fuchs, A. State and J. C. Bazin, "Immersive 3D Telepresence," *Computer*, vol. 47, no. 7, pp. 46-52, Jul. 2014.

[3] M. Ruhnke, R. Kummerle, G. Grisetti, and W. Burgard, "Highly accurate 3d surface models by sparse surface adjustment," in *Proc. IEEE. Int. Conf. Robotics and Automation*, Saint Paul, MN, 2012, pp. 751-757.

[4] R. Schnabel and R. Klein, "Octree-based point-cloud compression," in *Proc. Euro graphics Symposium on Point-Based Graphics*, Zurich, Switzerland, 2006, pp. 111-120.

[5] W. Jiang, J. Tian, K. Cai, F. Zhang and T. Luo, "Tangent-plane-continuity maximization based 3D point compression," in *Proc. IEEE. Int. Conf. Image Processing*, Orlando, FL, 2012, pp. 1277-1280.

[6] T. Ochotta and D. Saupe, "Compression of point-based 3d models by shape-adaptive wavelet coding of multi-height fiels," in *Proc. Symposium on Point-Based Graphics*, Zurich, Switzerland, Jun. 2004.

[7] J. K. Ahn, K. Y. Lee, J. Y. Sim and C. S. Kim, "Large-scale 3D point cloud compression using adaptive radial distance prediction in hybrid coordinate domains," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 3, pp. 422-434, Apr. 2015.

[8] R. A. Cohen, M. Krivokuca, C. Feng, Y. Taguchi, H. Ochimizu, D. Tian, and A. Vetro, "Compression of 3-D point clouds using hierarchical patch fitting," *Technical Report of MERL*, 2017. [Online]. Available: https://www.merl.com/publications/docs/TR2017-115.pdf.

[9] R. A. Cohen, D. Tian and A. Vetro, "Point cloud attribute compression using 3-D intra prediction and shape-adaptive transforms," in *Proc. Data Compression Conference (DCC)*, Snowbird, UT, 2016, pp. 141-150.

[10] C. Zhang, D. Florncio and C. Loop, "Point cloud attribute compression with graph transform," in *Proc. IEEE. Int. Conf. Image Processing (ICIP)*, Paris, 2014, pp. 2066-2070.

[11] R. A. Cohen, D. Tian and A. Vetro, "Attribute compression for sparse point clouds using graph transforms," in *Proc. IEEE. Int. Conf. Image Processing (ICIP)*, Phoenix, AZ, 2016, pp. 1374-1378.

[12] P. A. Chou and R. L. de Queiroz, "Gaussian process transforms," in *Proc. IEEE. Int. Conf. Image Processing (ICIP)*, Phoenix, AZ, 2016, pp. 1524-1528.

[13] R. L. de Queiroz and P. A. Chou, "Transform coding for point clouds using a gaussian process model," *IEEE Trans. Image Processing*, vol. 26, no. 7, pp. 3507-3517, Jul. 2017.

[14] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Processing*, vol. 25, no. 8, pp. 3947-3956, Aug. 2016.

[15] J. Hou, L. P. Chau, Y. He and P. A. Chou, "Sparse representation for colors of 3D point cloud via virtual adaptive sampling," in *Proc. IEEE. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, 2017, pp. 2926-2930.

[16] R. Mekuria, K. Blom and P. Cesar, "Design, implementation, and evaluation of a point cloud codec for tele-immersive video," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 828-842, Apr. 2017.

[17] C. Tu, E. Takeuchi, C. Miyajima and K. Takeda, "Compressing continuous point cloud data using image compression methods," in *Proc. IEEE Intelligent Transportation Systems (ITSC)*, Rio de Janeiro, 2016, pp. 1712-1719.

[18] L. Cui, H. Xu and E. S. Jang, "Hybrid color attribute compression for point cloud data," in *Proc. IEEE. Int. Conf. Multimedia and Expo (ICME)*, Hong Kong, 2017, pp. 1273-1278.

[19] Test Code PCC in 3DG. Accessed on Aug.17, 2017. [Online]. Available: https://github.com/RufaelDev/pcc-mp3dg.

[20] B. Stephen and L. Vandenberghe, "Convex optimization," *Cambridge University Press*, pp.561-596, Mar. 2004.

[21] Y. Zhang. "Solving large-scale linear programs by interior-point methods under the MATLAB environment," *Optimization Methods and Software*, vol. 10, no. 1, pp. 1-31, Jan. 1998.

[22] PCC DataSets: MPEG Content repository. Accessed on Aug. 18, 2017. [Online]. Available: http://157.159.160.118/MPEG/PCC/DataSets/pointCloud/.