

# Original Paper

## Quantization Parameter Cascading for Lossy Point Cloud Attribute Compression in G-PCC

Lei Wei<sup>1,2</sup>, Zhiwei Zhu<sup>1\*</sup>, Zhecheng Wang<sup>1,3</sup> and Shuai Wan<sup>1,4</sup>

<sup>1</sup>*School of Electronics and Information, Northwestern Polytechnical University, China*

<sup>2</sup>*College of Engineering, Xi'an International University, China*

<sup>3</sup>*School of Computer Technology and Application, Qinghai University, China*

<sup>4</sup>*School of Engineering, Royal Melbourne Institute of Technology, Melbourne, Australia*

---

### ABSTRACT

Region adaptive hierarchical transform (RAHT) is employed in G-PCC to make attribute compression more efficient. The performance of RAHT is closely related to the quantization parameter (QP), where applying different QPs to different transform depths is beneficial for coding efficiency. In this paper, QP cascading (QPC) is designed based on rate-distortion modelling. Firstly, the single-layer rate-quantization and distortion-quantization models are built by investigating the distribution of residuals. Later, the dependency of adjacent layers is studied to establish the rate-distortion model with dependency. Based on the proposed model, a rate-distortion optimization (RDO) guided QPC (O-QPC) and a fast implementation (F-QPC) are proposed. The experimental results verify the efficiency of the proposed methods. Compared with the G-PCC anchor, under the lossless geometry compression, O-QPC achieves an average of 1.5% performance gain in luma and nearly

---

\*Corresponding author: Zhu Zhiwei, zhiweizhu@mail.nwpu.edu.cn. This work was supported in part by the TCL science and technology innovation fund.

---

Received 30 August 2024; revised 03 October 2024; accepted 24 October 2024

ISSN 2048-7703; DOI 10.1561/116.00000063

© 2025 L. Wei, Z. Zhu, Z. Wang and S. Wan

13% gain in chroma, and F-QPC achieved an average performance gain of 1.0% in luma and almost 11% in chroma; Under the lossy geometry compression, O-QPC obtained an average of 3.9% gain in luma, and 13% gain in chroma, and F-QPC achieved an average of 3.4% gain in luma and nearly 12% gain in chroma. In particular, F-QPC achieves gains with almost no increase in complexity.

---

*Keywords:* point cloud compression, attribute, RAHT, quantization parameter

## 1 Introduction

A point cloud generally contains millions of points, in which geometry coordinates and a vector of attributes are associated with each point, such as color, reflectance, and normal [20, 25]. Unfortunately, large amounts of data present challenges for corresponding applications as well as research, such as storage, transmission, or processing. Point cloud compression (PCC) is now an acute need for applications for the general market, which has attracted the attention of academia and industry.

Among the standards, geometry-based PCC (G-PCC) is the latest standard developed by the 3D Graphics Coding Group under the Moving Picture Experts Group [5, 34]. In G-PCC, the geometry is coded first, and then the attribute is coded with the help of the reconstructed geometry [12]. This paper focuses on attribute compression in G-PCC, and addresses Region Adaptive Hierarchical Transform (RAHT) [9, 33] which is one of the common tools for attribute compression. Currently, all RAHT residuals share the same quantizer in G-PCC, which does not take advantage of the different contributions of coefficients in different transform depths to the overall rate-distortion (R-D) performance. Intuitively, it makes more sense to employ a smaller quantization parameter (QP) for residuals that considerably influence performance and vice versa [38]. Recent studies have demonstrated the potential of rate-distortion optimized quantization to enhance the efficiency of geometry-based point cloud compression, particularly for the predicting transform method [16]. Additionally, a novel dependence-based coarse-to-fine approach has been proposed to reduce distortion accumulation in attribute compression, further improving the state-of-the-art [15].

However, it is unclear how to configure QP for RAHT. To ascertain the reasonable setting scheme, it is necessary to establish the rate and distortion models first. Most R-D modeling research is focused on video coding instead of point clouds compression. Typically, the QP domain R-D modeling based on QP [4], the  $\lambda$ -domain R-D modeling [22] based on the Lagrangian Multiplier  $\lambda$ , and the  $\rho$ -domain R-D modeling [18] based on the percentage of non-zero

coefficients after quantization  $\rho$  have been explored for video coding. Nevertheless, those models are dedicated for video, which are not directly applicable for the octree-based structure as well as RAHT in G-PCC. As a result, it is necessary to first analyze the distribution of the RAHT coefficients' residuals and then establish the rate and distortion models.

Additionally, since RAHT is a multi-depth transform, there are two factors that affect the bitrate for each depth (the depth is referred to as the layer in this paper). First, it is directly affected by the quantizer for the current layer. Second, it is indirectly affected by the distortion of its reference layer. To the best of the authors' knowledge, this inter-layer interaction of distortion in RAHT has not yet been investigated in previous work.

To address the issues above, we first establish the single-layer rate-quantization (R-Q) and distortion-quantization (D-Q) models by investigating the distribution of residuals. Secondly, the dependency of adjacent layers is studied to understand how the reference layer distortion affects the rate of the coding layer. Then, an R-D model with dependency is established. Finally, we provide a rate-distortion optimization (RDO) guided QP cascading (QPC) as well as a fast implementation as the solutions for improving the R-D performance of G-PCC attribute compression.

The contributions of this paper are summarized as follows.

- By studying the actual distribution of the RAHT residuals, the corresponding R-Q and D-Q models for a single layer are developed to accurately estimate the actual R-D relationships.
- Based on the above models, an inter-layer dependency-based R-D model is proposed, taking into account the inter-layer dependency between a coding layer and its reference layer.
- An RDO guided QPC method and a fast QPC method are designed for the RAHT, respectively. Both of the proposed QPC methods outperform the state-of-the-art and the fast implementation has no increase in encoding/decoding time compared to the original G-PCC.

The rest of this paper is organized as follows. Section 2 presents RAHT related. Section 3 introduces the proposed rate and distortion models. Then, we propose a QPC method and a fast one based on it in Section 4. Experimental results and analysis are given in Section 5. Finally, the conclusion is drawn in Section 6, which is followed by references.

## 2 RAHT related

We start with a brief introduction to RAHT, based on the idea of using the attributes associated with nodes at the higher depths of an octree to predict

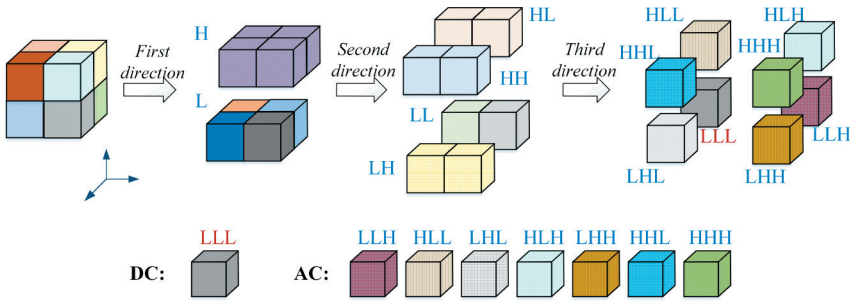


Figure 1: Dyadic RAHT for a transform block.

the attributes of nodes at the next depth. We follow the octree scan backwards, from voxels to the entire space, at each step recombining voxels into larger ones until reaching the root.

A  $2 \times 2 \times 2$  block of nodes serves as the basic unit of RAHT, as shown in Figure 1. First, RAHT is performed along a first direction, where the transformed nodes are split in low (L) and high (H) frequency nodes. Then, the decomposition is applied along a second direction on H and L nodes, respectively, to obtain LL, LH, HL, and HH nodes. Applying the decomposition along the third direction on LL, LH, HL and HH nodes will result in LLL, LLH, LHL, LHH, HLL, HLH, HHL and HHH nodes [27]. A maximum of 7 AC coefficients and just 1 DC coefficient (LLL) are ultimately obtained, with the AC coefficients being quantized and encoded and only the DC coefficient being used to transform in the subsequent layer.

The number of AC coefficients is related to the occupancy information of the block (i.e., density). A full occupied block, for instance, will have 7 AC coefficients, while a block with only one node will have none. The transform depth and point cloud’s density determine the transform block’s occupancy. Therefore, for different types of point clouds, the AC coefficients will be concentrated at different layers, which poses a challenge for optimization.

To further improve the performance, prediction is combined with transform coding for RAHT in G-PCC, i.e., the transform domain up-sampled prediction [26]. It is a multi-resolution prediction, utilizing the 19 neighbors that share a face or an edge with the parent of the sub-node (the central node) to be predicted, as shown in Figure 2. More specifically, for each sub-node of the central node, 7 parent neighbors (i.e., 3 sharing the same face, 3 sharing the same edge, and its own parent) will be used.

The prediction introduced a dependency between adjacent layers, that is, the distortion of the parent nodes, which are used as the reference nodes, directly affects the rate and distortion of the current sub-node. However,

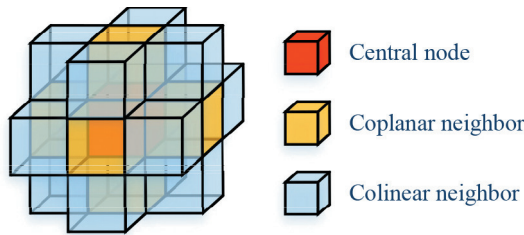


Figure 2: 19 nodes to up-sample the central node.

leveraging the inter-layer dependency to improve R-D performance is still a challenging issue that has not been adequately addressed in the past.

In the latest G-PCC, only a rate-distortion optimization of quantization (RDOQ) method [28] has been proposed to improve the coding efficiency by optimizing and modifying the QPs of coefficients. Precisely, RDOQ is based on a very simple estimation of the number of coded bins and then determines the QP by minimizing the Lagrange cost for each coefficient. However, the estimation is very simple and does not consider the dependency between coefficients at different depths. This will make the performance gain very limited, and the optimal solution cannot actually be obtained. In this paper, we will study the dependency and establish a corresponding model to obtain the optimized solution of QPC.

### 3 RD models with dependencies

This section begins with analyzing the distribution of RAHT residuals, from which the single layer R-Q and D-Q models are established. Then the dependency between adjacent layers is examined and an R-D model with dependency is established. To facilitate the analysis, three typical point clouds are selected as the analysis set, namely *basketball\_player\_vox11*, *longdress\_vox10*, and *facade\_vox14* [6], as shown in Figure 3.

#### 3.1 Distribution of RAHT residuals

Generally speaking, the rate and distortion models are determined by the distribution of the residuals. Knowledge of the residuals' probability distribution is essential in designing and optimizing the quantizer, entropy coder, and related image processing algorithms, especially in RDO. The more accurate the estimate of the distribution of the coefficients is, the more precise the estimation of the rate and distortion will be.



Figure 3: Point clouds in analysis set, *basketball\_player\_vox11*, *longdress\_vox10*, and *facade\_vox14*.

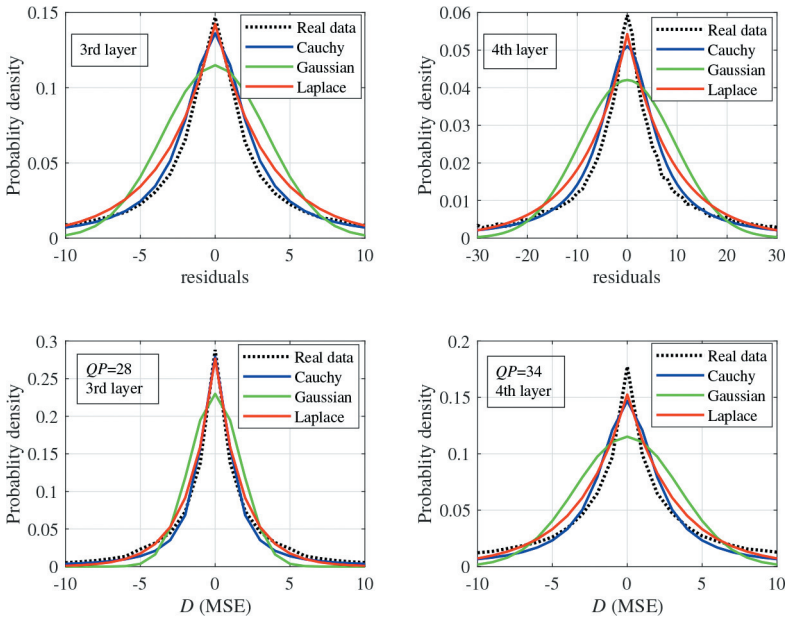


Figure 4: Distribution of the residuals and the distortions in different layers, *longdress\_vox10*.

It is widely acknowledged that residuals generally exhibit a symmetrical distribution with a peak at zero. The distribution can usually be approximated by Gaussian, Laplace, or Cauchy distributions [17, 21]. Since the work in this paper is based on the residuals of each transform layer in RAHT, it is necessary to perform statistics for the residual in different layers and the corresponding distortions, as shown in Figure 4.

Consistent with the literature, we observe a symmetric distribution with a peak located at the zero point, and the distribution decreases rapidly as the residuals deviate from zero. Compared to the Cauchy and Gaussian distributions, the Laplace distribution has a higher fitting accuracy and can fit the peak better because it has the form of an exponential distribution. The probability distribution function of the Laplace distribution is

$$p(x) = \frac{\Lambda}{2} e^{-\Lambda|x|}, \Lambda = \sqrt{2/\sigma^2}, \quad (1)$$

where  $\Lambda$  is the Laplace distribution parameter, and  $\sigma^2$  is the variance of the transform residuals which indicates the property of the input point cloud.

A larger Laplace parameter  $\Lambda$  means smaller but more centralized energy, which means the area is more sensitive to quantization, and a smaller quantization step ( $Q$ ) should be used. On the other hand, a smaller  $\Lambda$  means the energy is decentralized, and a minor change in the quantization parameter has little effect on the overall distortion [36]. Because of the one-to-one mapping between  $\Lambda$  and  $\sigma^2$ , the latter will be used in the following expressions and discussions for simplicity.

### 3.2 R-Q model

Based on the assumption that the residual is Laplace distributed and quantized by a scalar quantizer with  $Q$ , the rate should be  $R(Q) \approx H(Q)$ , where  $H(Q)$  is the empirical entropy [29]. The following is a popular rate estimation formula using entropy [7, 30],

$$H(Q) = \log_2 \frac{\sigma}{Q} + \log_2(\sqrt{2}e), \quad (2)$$

where  $\sigma$  is the standard derivation.

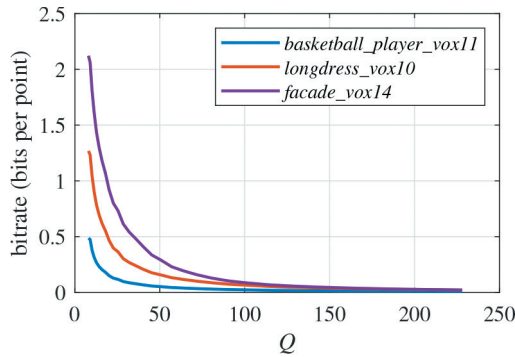
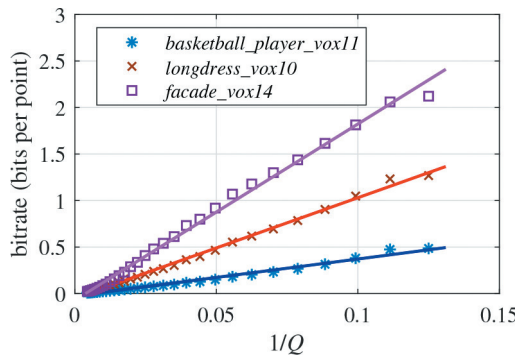
However, we observe that the empirical rate is not exactly the same as that predicted by Eq (2) for G-PCC in practice. This is because the equation above uses assumptions and approximations, while the actual R-Q curves are more heavily damped in the tail, as shown in Figure 5.

One reason is that the quantized residuals are not just entropy-coded individually but run-length-coded. The zero-grouping in the run-length coding reduces the final bit counts. For this, a slightly different model was proposed [10]

$$R = \mu + \frac{\nu}{Q^\gamma}, 0 \leq \gamma \leq 2. \quad (3)$$

To evaluate the model in Eq (3), we plot the  $R-1/Q$  curves in Figure 6.

It can be observed that the model fits the actual data well, while  $\mu$  in the actual data is rather small. We simplify Eq (3) considering that  $\mu \approx 0$ . Then,

Figure 5:  $R$ - $Q$  curves of actual point clouds.Figure 6:  $R$ - $1/Q$  curves.

Eq (3) can be expressed as  $R = \nu/Q$ , where  $\nu$  is related to the distribution of the point cloud, i.e., to the variance of residuals. As a result, the model only has one parameter  $\nu$  and is easily used in practice.

Because the variance shows the dispersion of a random variable concerning its mathematical expectation, the smaller the variance, the more concentrated the random variable's values are [31]. As a result, for entropy coding, the higher the concentration of random variable values, the lower the corresponding bitrate. It can then be concluded that  $\nu$  is almost increasing monotonically with the variance. Figure 7 shows the relationship between  $\nu$  and the corresponding variance  $\sigma^2$ .

The relationship between  $\sigma^2$  and  $\nu$  is fitted as the linear function  $\nu = a\sigma^2 + b$ , where  $a$  and  $b$  are the parameters. Finally, we can get

$$R = \frac{a\sigma^2 + b}{Q} \quad (4)$$

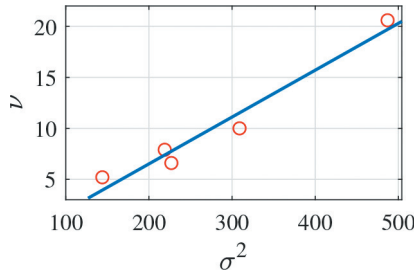


Figure 7: Relationship between  $\nu$  and  $\sigma^2$  using the analysis set with different layers.

### 3.3 D-Q model

Compared with the R-Q model, the distortion model is easier to obtain. With the distribution, distortion in each quantization interval can be calculated. Given the quantization step  $Q$ , the distortion in terms of MSE can be estimated as follows [2],

$$D(Q) = \sum_{i=-\infty}^{\infty} \int_{(i-1/2)Q}^{(i+1/2)Q} |x - iQ|^2 p(x) dx, \quad (5)$$

where  $p(x)$  is the probability density function.

Such a complex issue can be approximated by an empirical power function model [19, 24],

$$D = \xi Q^\gamma, \quad (6)$$

where  $\xi$  and  $\gamma$  are model parameters and can be found using the least-squared-errors solution.

The distortion of the residuals of each layer in *longdress\_vox10* are as shown in Figure 8 in the form of log-log.

It can also be seen that  $D$  is related not only to  $Q$  but also to the distribution of residuals of each layer of the point cloud, so the model parameters  $\xi$  and  $\gamma$  are related to the content of the point cloud, i.e., the variance.

### 3.4 Dependencies between layers

Up-sample prediction introduces dependency between a coding layer and its reference layer, as shown in Figure 9. However, the dependency complicates the RDO problem.

Since the current layer is only directly dependent on the previous layer, any two adjacent layers ( $k+1$ th layer and  $k$ th layer) are used for analysis to evaluate the propagation of distortion.

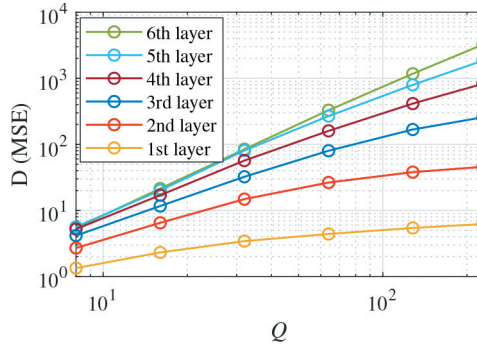


Figure 8:  $D$  versus  $Q$ , *longdress\_vox10*.

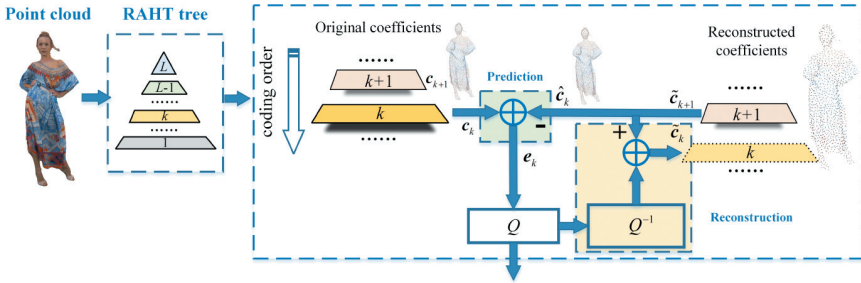


Figure 9: Dependency between adjacent layers.

Let the residuals of the  $k$ th layer be  $e_k$ , we can get

$$e_k = c_k - \hat{c}_k, \tag{7}$$

where  $c_k$  and  $\hat{c}_k$  are the original coefficients and their predicted values of the  $k$ th layer.

The reference layer is the  $k + 1$ th layer, then  $\hat{c}_k = \tilde{c}_{k+1}$ , where  $\tilde{c}_{k+1}$  is the reconstructed coefficients of the  $k + 1$ th layer. Let  $c_{k+1}$  be the original coefficients of the  $k + 1$ th layer, then its distortion is  $d_{k+1} = c_{k+1} - \tilde{c}_{k+1}$ . Substituting into Eq (7), we get  $e_k = c_k - c_{k+1} + d_{k+1}$ .

Let  $r_k = c_k - c_{k+1}$ , which is determined by the content of the point cloud. Finally,  $e_k$  can be expressed as

$$e_k = r_k + d_{k+1}. \tag{8}$$

Eq (8) demonstrates that the residual of the  $k$ th layer is divided into two parts: the first part is related to the coefficients of the current and the previous layer, and the second part is the distortion of the previous layer, which is the dependency between adjacent layers.

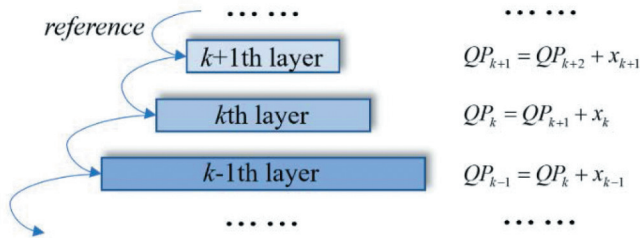


Figure 10: Schematic diagram of QPC.

The variance of  $e_k$  can be found according to Eq (8),

$$\sigma_{e_k}^2 = \sigma_{r_k}^2 + \sigma_{d_{k+1}}^2 - 2Cov(r_k \cdot d_{k+1}), \tag{9}$$

where  $\sigma_{d_{k+1}}^2$  is actually the distortion of the  $k + 1$  layer (denoted as  $D_{k+1}$ ),  $\sigma_{r_k}^2$  is only related to the content of the point cloud, and  $Cov(r_k \cdot d_{k+1})$  denotes the covariance.

Eq (9) is a direct dependency between two adjacent layers. Theoretically, distortion comes from quantization. An approximation that  $r_k$  and  $d_{k+1}$  are uncorrelated in Eq (9) leads to

$$\sigma_{e_k}^2 \approx \sigma_{r_k}^2 + \sigma_{d_{k+1}}^2. \tag{10}$$

Substituting into the Eq (4), we can get the R-D model with dependency, the representation between the rate of  $k$ th layer and the distortion of  $k + 1$ th layer is shown as

$$R_k = \frac{a(\sigma_{r_k}^2 + D_{k+1}) + b}{Q_k}. \tag{11}$$

So far, the dependent R-D model have been achieved.

#### 4 QPC for RAHT

The up-sample prediction of RAHT can be briefly represented as the structure shown in Figure 10.

For any layer other than the layer with the greatest transform depth, set as the  $k$ th layer, it uses the reconstructed values of the previous layer ( $k + 1$ th layer) for up-sampling prediction. Thus, the  $k + 1$ th layer has a more significant impact on R-D performance. It can be concluded that the impact of each layer increases with the depth of the transform.

The layer with a larger influence should be applied with a smaller QP, while the layer with less influence can use a larger QP. This is similar to what video

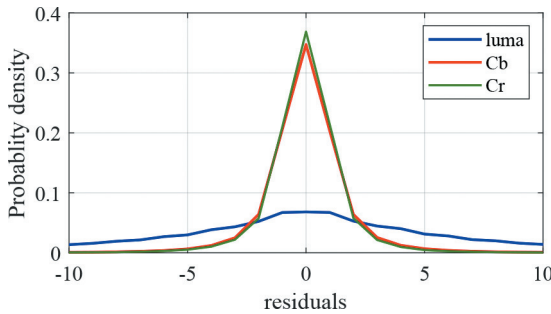


Figure 11: Distribution of residuals of luma, Cb and Cr of the first slice of *facade\_vox14*.

coding does in the time domain, i.e., QPC in hierarchical temporal prediction [14]. Following a similar spirit, let  $QP_k = QP_{k+1} + x_k$ , ( $x_k \geq 0$ ), where  $x_k$  is the  $k$ th layer's QP offset.

Moreover, the distribution of residuals differs for each component of color space in addition to the distribution of different layers. Take the point cloud *facade\_vox14* as an example, and the corresponding residuals of the three components are shown in Figure 11.

The distributions of Cb and Cr are very similar, almost coincide, and have much more zeros than luma. Consequently, the bitrate of luma is much higher than that of chroma. Additionally, luma is the primary source of appropriate information for humans to view since luma is significantly more sensitive to human eyes than chroma. Chroma can therefore withstand more substantial distortion [32].

Accordingly, we set the QP of chroma with a larger value than luma. Let the QP of luma in the  $k$ th layer be  $QP_{k,luma}$ , and then the QP of chroma can be expressed as

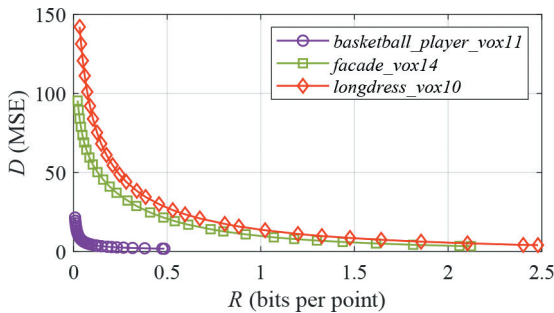
$$QP_{k,chroma} = QP_{k,luma} + \Delta_{chroma}, \quad (12)$$

where  $\Delta_{chroma}$  is the QP offset of chroma, which is set as 1 in this work for simplification.

#### 4.1 Lagrange factor $\lambda$

The fundamental concept of QPC is to provide each layer with a more appropriate quantization step by assigning fewer bits to a layer that can tolerate more distortion. RDO is usually used for this purpose. The RDO-guided QPC can be transformed into a non-constrained optimization problem by introducing the Lagrange factor  $\lambda$ , as,

$$J = \sum_{k=1}^L \omega_k D_k(QP_k, \dots, QP_L) + \lambda \sum_{k=1}^L R_k(QP_k, \dots, QP_L), \quad (13)$$

Figure 12:  $R$ - $D$  curves.

$$\mathbf{QP}^* = \arg \min_{\mathbf{QP}_k \in \mathbf{QP}} J(\mathbf{QP}, \lambda), \quad (14)$$

where  $\mathbf{QP}^* = (QP_1^*, \dots, QP_L^*)$  is the optimal QP vector in the set of all possible and admissible quantization candidates:  $\mathbf{QP}$ ,  $L$  is the total number of transform layers,  $D_k$  and  $R_k$  denote the distortion and the bitrate of the  $k$ th layer respectively, and  $\omega_k$  is the  $k$ th layer's weight, which is obtained by

$$\omega_k = N_k/N, \quad (15)$$

where  $N_k$  is the number of the AC coefficients of the  $k$ th layer and  $N$  is the total number of the coefficients of the point cloud. Apparently,  $\omega_k$  is determined by the percentage of valid coefficients in that layer. The more the AC coefficients, the larger the weight of  $\omega_k$  will be.

The Lagrangian multiplier is commonly regarded as a function of the quantization step in video coding [23, 35]. This paper also adopts the Q-field representation where the  $\lambda$ -Q relationship is offline trained according to  $\lambda = -\partial D/\partial R$  [11, 37]. The R-D curves obtained based on different point clouds are shown in Figure 12. It is observed that the trends of different point clouds are very close to each other.

Approximate  $\lambda = -\partial D/\partial R$  as  $\lambda = -(D_{QP+1} - D_{QP})/(R_{QP+1} - R_{QP})$ , the relationship between  $\lambda$  and  $\mathbf{QP}$  is obtained statistically, as shown in Figure 13.

Estimating  $\lambda$  as an exponential function of QP in the form of

$$\lambda = \alpha e^{QP \cdot \beta}, \quad (16)$$

where  $\alpha=0.04$  and  $\beta=0.25$  (R-squared: 0.9875), as shown by the blue line in Figure 13. The fitting accuracy is evaluated by the square of correlation coefficient (R-square), the closer the value of R-squared is to 1, the better the fit; conversely, the smaller the value of R-squared, the worse the fit.

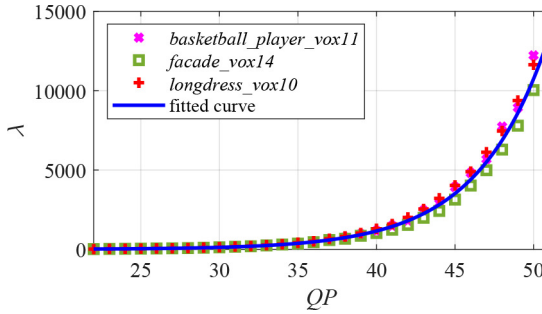


Figure 13: Relationship between  $\lambda$  and  $QP$ .

Substituting Eq (4), (6), and (16) into (13), we can get

$$\begin{aligned}
 J = & \omega_L \xi Q_L^\gamma + \sum_{k=1}^{L-1} \omega_k \xi Q_k^\gamma \\
 & + \left( \alpha e^{QP_L \cdot \beta} \frac{\alpha \sigma_{r_L}^2 + b}{Q_L} + \sum_{k=1}^{L-1} \alpha e^{QP_k \cdot \beta} \frac{a(\sigma_{r_k}^2 + \xi Q_{k+1}^\gamma) + b}{Q_k} \right), \tag{17}
 \end{aligned}$$

where  $Q_k = 2^{(QP_k - 4)/6}$  and  $QP_L = QP_0$ .

### 4.2 Determination of QPC layers

As discussed in Section 2, the weight  $\omega_k$  is related to the point cloud’s density. For the dense point clouds, the number of nodes in each layer will exponentially decline as transform depth increases, and the corresponding AC coefficients will similarly exponentially decrease. However, the number of AC coefficients derived in the earlier several transform depths is greatly limited for the sparse points. Each block will include progressively more nodes as the transform continues. It will resemble a dense point cloud at a certain depth, that is, each layer’s AC coefficients will then dramatically decrease as the transform continues after this specific depth.

As shown in Table 1, the weights  $\omega_k$  derived by Eq (15) of the analysis set are calculated, and the analysis and the results agree in every detail.

Many layers’ weights are extremely low, i.e., almost zero, which can be ignored in Eq (17). Consequently, the optimization can be realized by optimizing for the layers that satisfy  $\omega_k > \theta$  ( $\theta$  is an empirical threshold, 0.04 in this paper). Since the distribution of  $\omega_k$  is monotonic or convex, we mark these layers by finding the starting layer (denoted as the *sth* layer) and the ending layer (denoted as the *eth* layer).

Table 1:  $\omega_k$  of different layers under lossless geometry compression.

Sequences	1st layer	2nd layer	3rd layer	4th layer	5th layer	6th layer	7th layer
<i>basketball_player_vox11</i>	<b>72.60%</b>	<b>20.29%</b>	<b>5.30%</b>	1.36%	0.35%	0.09%	0.02%
<i>longdress_vox10</i>	<b>72.26%</b>	<b>20.49%</b>	<b>5.42%</b>	1.37%	0.34%	0.08%	0.02%
<i>facade_vox14</i>	0.89%	<b>8.73%</b>	<b>48.65%</b>	<b>29.08%</b>	<b>9.27%</b>	2.50%	0.64%

Table 2: values of  $x_s$  selected by O-QPC.

Sequences	$x_s$ ( $QP_0 = 22, 28, 34, 40, 46$ )	
	C1	C2
<i>basketball_player_vox11</i>	(2,1,1,1,3)	(3,1,1,1,2)
<i>longdress_vox10</i>	(1,1,1,0,0)	(3,1,0,1,2)
<i>facade_vox14</i>	(1,0,1,1,3)	(1,1,1,2,2)

Additionally, only the QP offset  $x_s$  of the most influential layer among these layers (i.e., the  $s$ th layer) is calculated to simplify the model, and the QP offsets of the remaining layers are all set to a fixed value of 1. As we propose, the above-described process is the optimized QPC (denoted as O-QPC) method.

Eq (17) can be rewritten as,

$$\begin{aligned}
J \approx & \xi\omega_{s+1}2^{(QP_0-4)\gamma/6} + \sum_{k=e}^s \xi\omega_k 2^{(QP_0+x_s+(s-k)-4)/6} \\
& + \left( \alpha e^{QP_0 \cdot \beta} \frac{\alpha \sigma_{r_{s+1}}^2 + b}{2^{(QP_0-4)/6}} + \sum_{k=e}^s \alpha e^{(QP_0+x_s+(s-k))\beta} \right. \\
& \left. \cdot \frac{a(\sigma_{r_k}^2 + 2^{(QP_0+x_s+(s-k)-4)\gamma/6}\xi) + b}{2^{(QP_0+x_s+(s-k)-4)/6}} \right). \tag{18}
\end{aligned}$$

Set the derivation of  $J$ , i.e.,  $\partial J/\partial x_s$ , to zero. Let  $f(x_s) = \partial J/\partial x_s$ , and the approximate solution of  $\partial J/\partial x_s = 0$  is  $x_s = \Psi - f(\Psi)/f'(\Psi)$  using the Newton-Raphson method [1].

The parameters in Eq (18) are not known before encoding, thus they need to be estimated from the pre-analysis of point clouds. In this work, we pre-code the input point cloud once to obtain the distribution characteristics of the coefficients in each layer, as well as the rate and distortion, which are used to fit the corresponding model parameters. Detailed information of pre-analysis can be found in Section 5.2. Then,  $\Psi$  is taken as the empirical value of 6, and the final approximate optimal value of  $x_s$  is obtained, the corresponding QP is  $QP_s = \lfloor QP_0 + x_s \rfloor$ , where  $\lfloor \cdot \rfloor$  denotes rounding to the nearest possible quantization parameter.

The specific process of the proposed O-QPC is summarized as Algorithm 1.

The results of the analysis set are shown in Table 2. The test conditions for lossy attribute compression in common test conditions (CTC) are applied, i.e., C1 and C2, corresponding to lossless and lossy geometry compression, respectively [6]. Note that for lossless and lossy geometry compression and different QPs, the residuals' distribution of each layer is different. Therefore, separate calculations are required and different results are obtained.

**Algorithm 1** O-QPC

---

**Input:**  $\theta, QP_0$ .  
**Output:**  $QP^* = (QP_1^*, QP_2^*, \dots, QP_L^*)$ .

1. **Initialize:**  $flag = \text{true}, k = 1, s = 0, e = 0, QP^* = (QP_0, \dots, QP_0)$
2. **for**  $k$  in  $[1, L]$  **do** {  
    Calculate  $\omega_k$ ;  
    **if** ( $flag$  and  $\omega_k > \theta$ ) **then** {  $s = k; flag = \text{false};$  }  
    **if** ( $\neg flag$  and  $\omega_k \leq \theta$ ) **then**  $e = k$ ;  
    }
3. **for**  $k$  in  $[L - 1, 1]$  **do** {  
    **if** ( $k == s$ ) **then** Calculate  $x_k$  by Eq (18);  
    **else if** ( $k > s$  and  $k < e$ ) **then**  $x_k = 1$ ;  
    **else**  $x_k = 0$ ;  
     $QP_k^* = QP_{k+1}^* + x_k$   
    }  
    }
4. **return**  $QP^*$

---

Taking *basketball\_player\_vox11* as an example, the O-QPC chooses the values of  $x_s$  as 2, 1, 1, 1, and 3 for  $QP_0=22, 28, 34, 40,$  and  $46,$  respectively. In addition, it can be seen that  $x_s$  is 1 for most  $QP_0$ .

Since the content-related parameters in Eq (18) must only be obtained through pre-coding and fitting, the offline pre-analysis of the optimization process is relatively complex and time-consuming. To avoid the laborious pre-analysis process and to make the optimization more practical while maintaining a significant performance gain, we propose a fast QPC (denoted as F-QPC) method based on the observation of Tables 1 and 2 and the analysis above.

F-QPC specifically means that for any point cloud and any  $QP_0$ , set  $x_s$  to 1 to maintain the performance gain. It completely avoids intricate pre-analysis step and computation of  $x_s$ . Meanwhile, F-QPC requires to identify the starting and ending layers, which is simply a comparison of the associated weight of each depth with a threshold when constructing the RAHT tree. Since the number of nodes in each depth is already available as well as the total number of the points of the point cloud at this time, it will hardly increase the encoding time.

## 5 Experimental results

To evaluate the R-D performance of the proposed methods (O-QPC and F-QPC), we implement both on TMC13 v26.0, the test model for G-PCC [13]. We compare the proposed methods with the G-PCC anchor, as well as with

the RDOQ proposed in [28]. The G-PCC datasets as required by the CTC are tested [6, 8], with the analysis set used for analysis in Section 3 excluded. These datasets are generally classified into the following categories according to the different densities of the point clouds, i.e., solid, dense, sparse, and scant categories. The test conditions for lossy attribute compression in CTC are applied, i.e., C1 and C2, corresponding to lossless and lossy geometry compression, respectively. The objective performance is evaluated using the Bjontegaard-Delta rate (BD-rate) [3]. Since the attribute is encoded separately in PCC, the BD-rate of the attribute (denoted as End-to-End BD-AttrRate) is reported.

### 5.1 Objective Performance

Tables 3 and 4 give the objective performance and time complexity of the O-QPC and F-QPC under C1 and C2, respectively.

Under C1 and C2, O-QPC and F-QPC both improve the R-D performance for all categories of point clouds compared to G-PCC, and significantly outperform RDOQ especially in luma. In overall, the performance gain of Q-QPC is larger than F-QPC in both luma and chroma components, while F-QPC obtains significant performance gain over G-PCC with no increase in complexity in terms of coding time and decoding time. If offline pre-analysis is not practical in applications, F-QPC can serve as a practical strategy for optimization.

The R-D curves of O-QPC, F-QPC, RDOQ, and G-PCC anchor under C1 and C2 are compared in Figure 14, to evaluate the performance differences at different bitrates.

It is observed that the proposed methods consistently outperform G-PCC and RDOQ at different bitrates. A larger performance gain can be observed at higher bitrates. It is because, as the figure illustrates, the attribute bitrate is already quite low when QP is large, even very close to zero. Even though some coefficients' QP has increased, the improvement in performance will be pretty limited. On the other hand, when QP is small, the attribute bitrate is comparatively high. QPC can be used to reduce the bitrate efficiently while maintaining the distortion as much as possible to improve performance. In addition, O-QPC and F-QPC are always very close, indicating that the performance gap between the two methods is marginal.

### 5.2 Complexity analysis

It can be seen from Tables 3 and 4 that although the encoding/decoding time of Q-QPC, FQPC and RDOQ is close to that of the anchor, while O-QPC requires extra complexity of pre-analysis of the input point cloud before encoding.

Table 3: Objective performance and time complexity of RDOQ, O-QPC, and F-QPC under C1.

C1_ai	End-to-End BD-AttrRate [%]											
	RDOQ			O-QPC			F-QPC					
	Luma	Cb	Cr	Luma	Cb	Cr	Luma	Cb	Cr			
Solid average	<b>-0.5%</b>	-10.8%	-10.2%	<b>-2.2%</b>	-10.2%	-7.8%	<b>-1.9%</b>	-8.3%	-6.2%			
Dense average	<b>-0.9%</b>	-11.5%	-11.3%	<b>-1.3%</b>	-11.4%	-11.0%	<b>-0.9%</b>	-9.6%	-9.7%			
Sparse average	<b>-0.1%</b>	-9.5%	-9.8%	<b>-0.6%</b>	-11.1%	-11.2%	<b>-0.2%</b>	-7.9%	-8.1%			
Scant average	<b>-0.6%</b>	-11.3%	-11.7%	<b>-0.7%</b>	-12.8%	-14.3%	<b>-0.2%</b>	-10.1%	-11.2%			
Overall average	<b>-0.6%</b>	-10.9%	-10.9%	<b>-1.2%</b>	-12.5%	-11.4%	<b>-0.7%</b>	-9.2%	-9.3%			
Avg. Enc Time [%]		102%			100% *			100%				
Avg. Dec Time [%]		101%			100% *			100%				

Note: \* denotes that here the time for offline pre-analysis is not included for O-QPC.

Table 4: Objective performance and time complexity of RDOQ, O-QPC, and F-QPC under C2.

C2_ai	End-to-End BD-AttrRate [%]											
	RDOQ			O-QPC			F-QPC					
	Luma	Cb	Cr	Luma	Cb	Cr	Luma	Cb	Cr			
Solid average	-0.0%	-12.6%	-11.4%	-4.1%	-12.3%	-8.9%	-3.5%	-10.2%	-7.0%			
Dense average	-1.4%	-13.0%	-11.0%	-5.2%	-14.5%	-13.4%	-4.8%	-13.3%	-11.9%			
Sparse average	-1.3%	-11.2%	-11.2%	-6.2%	-9.2%	-8.9%	-5.7%	-7.9%	-8.2%			
Scant average	-0.5%	-11.9%	-12.2%	-1.0%	-10.3%	-11.2%	-0.3%	-10.2%	-11.2%			
Overall average	-0.8%	-12.1%	-11.6%	-3.6%	-12.3%	-11.9%	-3.1%	-10.5%	-10.0%			
Avg. Enc Time [%]		98%			100% *			100%				
Avg. Dec Time [%]		97%			100% *			100%				

Note: \* denotes that here the time for offline pre-analysis is not included for O-QPC.

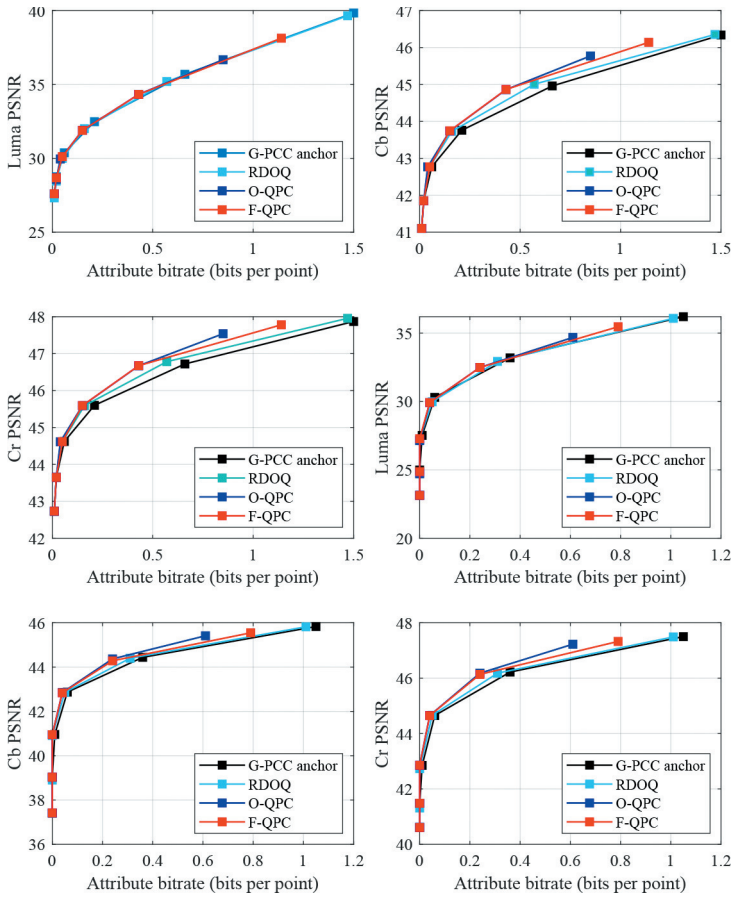


Figure 14: (a)-(c)  $R-D$  curves under C1, (d)-(f)  $R-D$  curves under C2, *facade\_vox11*.

Specifically, the first step of pre-analysis is pre-coding the total point cloud. When building the transform tree, the number of AC coefficients in each depth is available at each transform. In addition, the total number of coefficients (i.e., the total number of points in the point cloud) is known so that the weight of each depth can be calculated by Eq (15), and then the starting and ending layers are determined by comparing the weight with the threshold. Next, the characteristics of each layer’s coefficients distribution, rate, and distortion are collected by pre-coding. With the help of the data above, Eq (4) and (6) are fitted using the least-squared-errors solution to get the parameters used in Eq (18). Finally, the QP offset of the starting layer is calculated by Eq (18).

In contrast, neither RDOQ nor F-QPC requires any pre-analysis. Specifically, RDOQ uses some addition and shift operations to calculate the Lagrange cost for each coefficient, which has negligible impact on the time complexity. F-QPC employs a fixed value for the QP offsets rather than performing complex calculations. In addition, the starting and ending layers are adaptively selected by simply comparing the weight of each layer with the threshold. Since the number of nodes in each depth is already available, as well as the total number of the points of the point cloud at this time, it will hardly increase the encoding/decoding time.

## 6 Conclusions

In this paper, we presented rate and distortion models dedicated to RAHT in G-PCC using Laplace density approximation. Later, we further explored the dependence between the adjacent layers to build the R-D model with dependency. At last, we proposed two QPC methods, i.e., O-QPC and F-QPC for G-PCC attribute compression. The experimental results verify that both of the proposed methods can efficiently improve the R-D performance and outperform the state-of-the-art. In particular, F-QPC has a marginal performance loss when compared to O-QPC and achieves significant performance gain over G-PCC without increasing encoding/decoding time. Future work will involve utilizing the proposed model for G-PCC rate control.

## References

- [1] A. Ben-Israel, “A newton-raphson method for the solution of systems of equations”, *Journal of Mathematical analysis and applications*, 15(2), 1966, 243–52.
- [2] T. Berger, “A mathematical basis for data compression”, *Rate Distortion Theory*, Prentice-Hall, 1971.
- [3] G. Bjontegaard, “Calculation of average PSNR differences between re-curves”, VCEG-M33, document, 2001.
- [4] T. Chiang and Y.-Q. Zhang, “A new rate control scheme using quadratic rate distortion model”, *IEEE Transactions on circuits and systems for video technology*, 7(1), 1997, 246–50.
- [5] P. A. Chou, M. Koroteev, and M. Krivokuća, “A volumetric approach to point cloud compression—part i: Attribute compression”, *IEEE Transactions on Image Processing*, 29, 2019, 2203–16.
- [6] Common Test Conditions for G-PCC, “Standard ISO/IEC JTC1/SC29/WG7 MPEG 3D Graphics Coding, N00867, Rennes”.
- [7] T. M. Cover, *Elements of information theory*, John Wiley & Sons, 1999.

- [8] E. d'Eon, B. Harrison, T. Myers, and P. A. Cho, "8i voxelized full bodies-a voxelized point cloud dataset", *ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document WG11M40059/WG1M74006*, 7(8), 2017, 11.
- [9] R. L. De Queiroz and P. A. Chou, "Compression of 3d point clouds using a region-adaptive hierarchical transform", *IEEE Transactions on Image Processing*, 25(8), 2016, 3947–56.
- [10] W. Ding and B. Liu, "Rate control of mpeg video coding and recording by rate-quantization modeling", *IEEE transactions on circuits and systems for video technology*, 6(1), 1996, 12–20.
- [11] A. Filali, V. Ricordel, N. Normand, and W. Hamidouche, "Rate-distortion optimized tree-structured point-lattice vector quantization for compression of 3d point clouds geometry", in *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2019, 1099–103.
- [12] G-PCC 2nd Edition Codec Description, "Standard ISO/IEC JTC1/SC29/WG7 MPEG 3D Graphics Coding, N00865, Rennes", April 2024.
- [13] G-PCC Test Model v26, "Standard ISO/IEC JTC1/SC29/WG7 MPEG 3D Graphics Coding, N00863, Rennes", April, 2024.
- [14] Y. Gong, S. Wan, K. Yang, Y. Yang, and B. Li, "Rate-distortion-optimization-base d quantization parameter cascading technique for random-access configuration in h. 265/hevc", *IEEE Transactions on Circuits and Systems for Video Technology*, 27(6), 2016, 1304–12.
- [15] T. Guo, H. Yuan, R. Hamzaoui, X. H. Wang, and L. Wang, "Dependence-Based Coarse-to-Fine Approach for Reducing Distortion Accumulation in G-PCC Attribute Compression", *IEEE Transactions on Industrial Informatics*, 2024.
- [16] T. Guo, H. Yuan, L. Wang, and T. T. Wang, "Rate-distortion optimized quantization for geometry-based point cloud compression", *Journal of Electronic Imaging*, 32(1), 2023, 13047–47.
- [17] H.-M. Hang and J.-J. Chen, "Source model for transform video coder and its application. i. fundamental theory", *IEEE Transactions on Circuits and Systems for Video Technology*, 7(2), 1987, 287–98.
- [18] Z. He, Y. K. Kim, and S. K. Mitra, "Low-delay rate control for dct video coding via/spl rho/-domain source modeling", *IEEE transactions on Circuits and Systems for Video Technology*, 11(8), 2001, 928–40.
- [19] N. Kamaci, Y. Altunbasak, and R. M. Mersereau, "Frame bit allocation for the h. 264/avc video coder via cauchy-density-based rate and distortion models", *IEEE Transactions on Circuits and Systems for Video Technology*, 15(8), 2005, 994–1006.
- [20] M. Krivokuća, P. A. Chou, and M. Koroteev, "A volumetric approach to point cloud compression—part ii: Geometry compression", *IEEE Transactions on Image Processing*, 29, 2019, 2217–29.

- [21] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the dct coefficient distributions for images", *IEEE transactions on image processing*, 9(10), 2000, 1661–6.
- [22] B. Li, H. Li, L. Li, and J. Zhang, " $\lambda$  domain rate control algorithm for high efficiency video coding", *IEEE transactions on Image Processing*, 23(9), 2014, 3841–54.
- [23] L. Li, Z. Li, S. Liu, and H. Li, "Rate control for video-based point cloud compression", *IEEE Transactions on Image Processing*, 29, 2020, 6237–50.
- [24] J. Liu, Y. Cho, Z. Guo, and J. Kuo, "Bit allocation for spatial scalability coding of h. 264/svc with dependent rate-distortion analysis", *IEEE Transactions on Circuits and Systems for Video Technology*, 20(7), 2010, 967–81.
- [25] R. Mekuria, K. Blom, and P. Cesar, "Design, implementation, and evaluation of a point cloud codec for tele-immersive video", *IEEE Transactions on Circuits and Systems for Video Technology*, 27(4), 2016, 828–42.
- [26] On an improvement of RAHT to exploit attribute correlation, "ISO/IEC JTC1/SC29/WG11 input document m47378, Geneva, Switzerland".
- [27] On dyadic RAHT, "ISO/IEC JTC1/SC29/WG11 input document m53557, Online", April 2020.
- [28] Optimization of RAHT coefficients by the attribute encoder, "ISO/IEC JTC1/SC29/WG11 input document m52996, Alpbach, AUT".
- [29] J. Ribas-Corbera and S. Lei, "Rate control in dct video coding for low-delay communications", *IEEE Transactions on circuits and systems for video technology*, 9(1), 1999, 172–85.
- [30] J. Ribas-Corbera and D. L. Neuhoff, "Optimizing block size in motion-compensated video coding", *Journal of Electronic Imaging*, 7(1), 1998, 155–65.
- [31] J. A. Rice and J. A. Rice, *Mathematical statistics and data analysis*, Thomson/Brooks/Cole Belmont, CA, 2007, 371.
- [32] Q. Ruan, *Digital Image Processing*, Beijing, China: Publishing House of Electronics Industry, 2013, 60–74.
- [33] G. P. Sandri, P. A. Chou, M. Krivokuća, and R. L. de Queiroz, "Integer alternative for the region-adaptive hierarchical transform", *IEEE Signal Processing Letters*, 26(9), 2019, 1369–72.
- [34] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokuća, S. Lasserre, Z. Li, and others., "Emerging mpeg standards for point cloud compression", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 9(1), 2018, 133–48.
- [35] S. Wan and F. Yang, "New efficient video coding h. 265/hevc: Principle", *Standard and Implementation*, Beijing, China: Publishing House of Electronics Industry, 2014, 313–27.

- [36] T. Wiegand, H. Schwarz, and others., “Source coding: Part i of fundamentals of source and video coding”, *Foundations and Trends® in Signal Processing*, 4(1–2), 2011, 1–222.
- [37] Y. Yu, W. Zhang, F. Yang, and G. Li, “Rate-distortion optimized geometry compression for spinning lidar point cloud”, *IEEE Transactions on Multimedia*, 25, 2022, 2993–3005.
- [38] T. Zhao, Z. Wang, and C. W. Chen, “Adaptive quantization parameter cascading in hevc hierarchical coding”, *IEEE Transactions on Image Processing*, 25(7), 2016, 2997–3009.