3D Point Cloud Attribute Compression Using Diffusion-Based Texture-Aware Intra Prediction

Yiting Shao, Xiaodong Yang, Wei Gao, *Senior Member, IEEE*, Shan Liu, *Fellow, IEEE*, and Ge Li, *Member, IEEE*

***Abstract*— There is an urgent need from various multimedia applications to efficiently compress point clouds. The Moving Picture Experts Group has released a standard platform called geometry-based point cloud compression (G-PCC). However, its k-nearest neighbor (k-NN) based attribute prediction has limited efficiency for point clouds with rich texture and directional information. To overcome this problem, we propose a texture-aware attribute predictive coding framework in a point cloud diffusion model. In our work, attribute intra prediction is solved as a diffusion-based interpolation problem, and a general attribute predictor is developed. It is theoretically proven that G-PCC k-NN based predictor is a degraded case of the proposed diffusion-based solution. First, a point cloud is represented as two levels of details with seeds as the inpainting mask and non-seed points to be predicted. Second, we design point cloud partial difference operators to perform energy-minimizing attribute inpainting from seeds to unknowns. Smooth attribute interpolation can be achieved via an iterative diffusion process, and an adaptive early termination is proposed to reduce complexity. Third, we propose a structure-adaptive attribute predictive coding scheme, where edge-enhancing anisotropic diffusion is employed to perform texture-aware attribute prediction. Finally, attributes of seeds are beforehand encoded and prediction residuals of left points are progressively encoded into bitstream. Experiments show the proposed scheme surpasses the state-of-the-art by an average of 14.14%, 17.52%, and 17.87% BD-BR gains on the coding of Y, U, and V components, respectively. Subjective results on attribute reconstruction quality also verify the advantage of our scheme.**

***Index Terms*— Point cloud attribute compression, progressive intra prediction, diffusion-based interpolation, predictive coding.**

I. Introduction

With the significant progress of three-dimensional (3D) sensing and acquisition technologies, a point cloud has become a popular media format of 3D objects and scenes in a variety of important multimedia applications such as smart city, immersive communication, culture heritage storage, etc [1], [2], [3], [4], and [5]. Along with the rising market requirements of 3D high-resolution representation, the data volume of point clouds is drastically increasing with abundant geometric information and rich attributes. This situation brings heavy pressure on data transmission and storage systems with limited bandwidth and storage capacity. Compressing point clouds while maintaining high fidelity is an urgent demand that has attracted significant attention from both industry and academia. Notably, substantial progress has been achieved in point cloud compression, particularly in the realms of geometry compression [6], [7], [8] and attribute compression [9], [10], [11], [12], [13]. In this paper, we focus on point cloud attribute compression.

The objective of point cloud attribute compression is to utilize attribute similarities to reduce the number of bits while keeping the quality of the decoded point cloud as close to the original one as possible. However, the discrete and irregular geometric distribution of point clouds makes it a challenging task to capture attribute correlation in 3D space. A typical work on point cloud attribute compression is the Geometry-based Point Cloud Compression (G-PCC) [14], a standard platform released by the Moving Picture Experts Group (MPEG). G-PCC is an integrated point cloud coding solution, including a set of geometry codecs (octree-based, predictive tree-based, and trisoup-based compression schemes) and several attribute coding tools (region-adaptive hierarchical transform, attribute prediction and lifting transform).

Specifically, the attribute predictive coding in G-PCC is performed with three steps [14]: point cloud levels of detail (LoD) generation via resampling, attribute intra predictor selection by the k-nearest neighbor (k-NN) search, and attribute prediction with the linear interpolation. However, several issues on the current G-PCC need to be considered. **The first issue** to be addressed is the LoD generation. It only adjusts the sampling distance among points for different LoDs, which fails to ensure the regular geometric distribution of points in each LoD. It is noted that points within lower LoDs are more predominant as they are more frequently exploited for attribute prediction. However, the current G-PCC does not pay

Manuscript received 4 December 2023; revised 30 March 2024; accepted 28 April 2024. Date of publication 3 May 2024; date of current version 30 October 2024. This work was supported in part by the Natural Science Foundation of China under Grant 62172021, Grant 62271013, and Grant 62031013; in part by the Guangdong Province Pearl River Talent Program under Grant 2021QN020708; and in part by the Shenzhen Science and Technology Program under Grant JCYJ20230807120808017. This article was recommended by Associate Editor C. Herglotz. *(Corresponding author: Wei Gao.)*

Yiting Shao and Wei Gao are with the School of Electronic and Computer Engineering (SECE), Shenzhen Graduate School, Peking University, Shenzhen 518055, China, and also with the Peng Cheng Laboratory, Shenzhen 518066, China (e-mail: ytshao@pku.edu.cn; gaowei262@pku.edu.cn).

Xiaodong Yang and Ge Li are with the SECE, Shenzhen Graduate School, Peking University, Shenzhen 518055, China (e-mail: 2101212794@stu.pku.edu.cn; geli@ece.pku.edu.cn).

Shan Liu is with the Media Lab, Tencent, Palo Alto, CA 94306 USA (e-mail: shanl@tencent.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3396694.

Digital Object Identifier 10.1109/TCSVT.2024.3396694

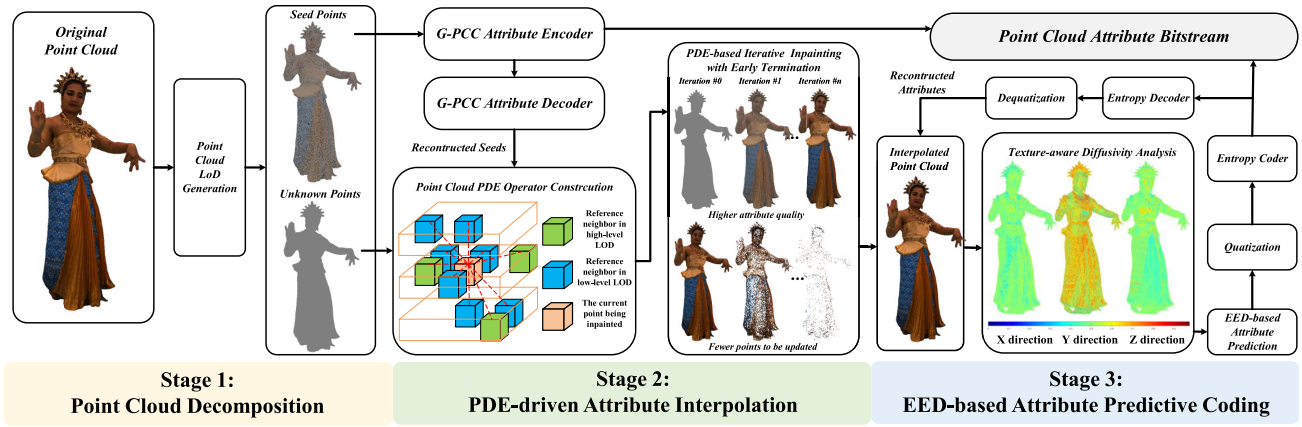


Fig. 1. Pipeline of the proposed PDE-based progressive attribute prediction framework. In the first stage, a point cloud is decomposed into two LoDs with seeds and unknown points. Seed attributes are coded into the bitstream and reconstructed as the attribute inpainting mask for the unknown points. In the second stage, attributes of unknown points are iteratively interpolated by our PDE-driven inpainting module, and an adaptive early termination is introduced to reduce complexity in this iterative process. In the third stage, using the interpolated point cloud as the predictor, texture-aware attribute predictive coding is point-by-point performed after EED-based attribute diffusivity analysis. Finally, attribute predicted residuals of points are coded into the total bitstream.

more attention to the generation of lower LoDs. **The second issue** to be concerned is the predictor selection. The k-NN based selection adopts the Euclidean distance as the only metric, ignoring the geometric distribution of those selected predictors. It may lead to a set of directional-biased predictors, which would impair attribute coding performance. **The third issue** to be studied is the interpolation-based prediction. The inverse distance weighted interpolation is a suboptimal way to generate predicted attributes with selected predictors. A more sophisticated interpolation method needs to be designed to fully utilize the attribute correlation in neighboring regions.

There are some works [15], [16], [17] aimed at handling the aforementioned problems, but they only partially solve those issues and lack the consideration of attribute prediction from the perspective of continuous attribute distribution. Wei et al. [15] provide a content-adaptive LoD generation method to reduce attribute prediction error, but they fail to optimize the attribute predictor selection with the LoD structure. Chen et al. [16] improve the attribute predictor selection with a Hilbert space-filling curve based point order, but they ignore the impact of suboptimal attribute interpolation step on coding performance. Song et al. [17] replace the interpolation-based attribute prediction in G-PCC with a region-aware optimized graph transform, which improves the coding performance but brings high computational complexity. To overcome the aforementioned problems, we reformulate the discrete attribute intra prediction task as a continuous data interpolation problem in a diffusion process. Inspired by diffusion-based image codecs [18], [19], [20], we solve the problem with a partial differential equation (PDE)-driven attribute prediction solution.

As shown in Fig. 1, we propose a diffusion-based intra prediction framework for point cloud attribute compression. Our main contributions are outlined as follows:

- We propose a progressive diffusion-based attribute predictive coding framework, which can perform accurate texture-aware attribute prediction to improve attribute coding performance. The proposed diffusion-based iterative interpolation strategy with adaptive early termination guarantees the effectiveness and efficiency of our scheme.
- We design PDE operators with 3D inpainting masks for point cloud attribute compression. By formulating the discrete attribute prediction as a continuous diffusion-based interpolation task, the proposed PDE-driven attribute predictor provides a general solution to fully exploit attribute spatial correlation. The current k-NN predictor is proved to be a degraded solution of ours.
- We construct a structure-adaptive attribute prediction scheme with edge-enhancing anisotropic diffusion (EED), which can perform smooth prediction in local regions while edge details can be preserved. Texture-aware diffusion tensors can be effectively extracted from the underlying texture structure to boost attribute prediction.

The rest of this paper is organized as follows. In Section II, we briefly review the works on point cloud attribute compression, diffusion-based image and point cloud processing. Section III gives technical background, problem formulation, and an overview of the proposed scheme. Details about the proposed attribute prediction framework are elaborated in Section IV. Section V presents the experimental results and analysis. Finally, the paper is concluded in Section VI.

II. Related Work

A. Point Cloud Attribute Compression

Before conducting attribute compression, discrete and irregular point clouds are usually reorganized into compact structures for better utilizing the local correlation of attributes. Octree is the most popular representation of point clouds. Schnabel and Klein [21] first adopt the octree in a prediction-based point cloud codec. Later, Huang et al. [22] propose an octree-based progressive attribute coding scheme to remove attribute redundancy in octree nodes. Different from octree-based codecs, Zhang et al. [23] first decompose a point cloud into several graphs and adopt graph Fourier transform (GFT) to compress point cloud attributes. Cohen et al. [24] extend the work with an improved graph construction method for more efficient attribute coding.

Then, Shao et al. [13] adopt the k-dimensional tree to conduct block partition before the graph construction. Shao et al. further provide a Lagrangian-optimized GFT in a block-based attribute coding scheme [25]. Xu et al. [26] construct a spatio-temporal GFT for dynamic point cloud attribute compression. Moreover, Song et al. [17] design a block-adaptive graph transform method to improve attribute coding performance. Considering GFT-based codecs are time-consuming, Queiroz and Chou [27] devise a more efficient transform-based attribute coding scheme named RAHT, and this work is adopted as a part of the solution in MPEG G-PCC. Recently, G-PCC [14] has integrated a set of coding tools for efficient attribute compression, while the way to exploit attribute spatial correlation still can be improved in G-PCC for better coding performance.

We highlight a selection of representative studies closely related to our research and elaborate on the distinctive points of our study compared to these works. Song et al. [17] introduce a block-adaptive codec for point cloud attribute compression, incorporating progressive clustering and optimized graph transform. In contrast, our work emphasizes leveraging spatial attribute correlation to enhance intra prediction methods, leading to more compact spatial domain representations. Notably, our novel spatial diffusion model facilitates redundancy removal, making our diffusion-based intra prediction scheme a distinct feature. Zhao et al. [9] propose an image synthesis-based codec, mapping point clouds to 2D space for attribute image generation. In contrast, our approach preserves the 3D structure and exploits attribute correlation in 3D space. We introduce a spatial diffusion model, enabling attribute intra prediction in 3D space, marking a significant distinction from Zhao's projection-based method. Liu et al. [10] present a hybrid codec integrating block-based intra prediction and sparse representation-based transform coding. In contrast, our two-layer point-wise intra prediction scheme fully exploits attribute spatial correlation at the point level, with our texture-aware intra prediction with an edge-enhancing diffusion model setting us apart from Liu's approach. Xu et al. [26] propose a predictive graph transform scheme for point cloud attribute compression, focusing on spatial-temporal attribute correlation in dynamic point clouds. Unlike their emphasis on improved transform coding tools, we concentrate on enhancing intra-prediction methods to leverage spatial attribute correlation for static point cloud compression. Our spatial diffusion model enables spatial redundancy removal, distinguishing our scheme from Xu's work. Shen et al. [28] introduce a block-based attribute codec enhanced with a hierarchical sparse transform coding scheme. While their emphasis is on hierarchical transform coding, we focus on enhancing intra prediction methods to mitigate spatial redundancy. Our texture-aware attribute predictive coding framework, within a point cloud diffusion model, stands out as a distinctive feature compared to Shen's work. Introducing a novel diffusion-based intra prediction scheme for point cloud attribute compression, particularly through the integration of spatial diffusion models and texture-aware predictive coding frameworks, distinguishes our work from existing literature.

B. Diffusion-Based Image Processing

Diffusion models have shown excellent performance on filling in missing data in image generation [29], [30], [31] and inpainting [32], [33], [34]. A pioneer diffusion-based image inpainting work is [34], where the image Laplacian serves as a local smoothness operator and an anisotropic diffusion model is used to simultaneously inpaint missing regions along the direction perpendicular to the image gradient. Extending this to image compression, many diffusion-based image codecs [18], [19], [20] are designed to store a fraction of selected pixels and propagate textures to local regions by inpainting with PDEs. Different PDE operators would lead to different coding performances for different image types. Mainberger et al. [35] propose a PDE-based codec with homogeneous diffusion for cartoon-like image compression, but it fails to handle images with complex textures. To overcome this, Galić et al. [18] propose an image codec with EED-based interpolation, first showing that diffusion-based codecs with suitable PDEs can outperform JPEG at low to medium bit rates. Schmaltz et al. [36] propose a new rectangular subdivision and an enhanced entropy coder to improve the codec [18], achieving better coding performance over JPEG 2000. Based on this, Peter et al. [20] propose a uma-guided anisotropic diffusion-based colorization scheme to achieve more efficient color compression. Those works prove that diffusion-based schemes have much potential in image compression tasks. Drawing on the progress of diffusion-based techniques in tasks such as filling in missing data in image generation and inpainting, we aim to leverage similar principles in our diffusion-based point cloud attribute compression.

C. Diffusion-Based Point Cloud Processing

Similar to diffusion-based image processing, the basic idea of diffusion-based point cloud processing is to propagate from known points to unknown points of local regions with smoothness priors and PDEs. However, extending PDEs from 2D images to 3D point clouds is a difficult problem because of the sparse and discrete geometry structure of point clouds. A 3D cuboidal space subdivision is proposed in a diffusion-based 3D image codec [36], but it is not applicable for point cloud contents. Liang et al. [37] reconstruct the local surface of point clouds to derive differential operators in 3D space. Similarly, Lai et al. [38] propose to construct local meshes and use numerical methods to approximate differential operators for point clouds. Inspired by the nonlocal discrete regularization on weighted graphs for PDE-based image processing [39], Lozes et al. [40] solve PDEs on graphs of point clouds without the extra construction of locally approximated surfaces. However, the calculation of PDEs on point cloud graphs is time-consuming. Hooda and Pan [41] adopt the 3D Sobel operator to obtain the point cloud luma gradient by assuming the current block with the associated 18 neighboring blocks are fully occupied. Nevertheless, this assumption would not hold in point clouds with an irregular geometry distribution. Designing a suitable PDE operator to effectively reconstruct missing data from the provided points presents a significant challenge. Besides, the design of an efficient predictive coding

method for the inpainted points is important for a PDE-based point cloud attribute codec. In this paper, we focus on these key problems.

III. Background, Problem Formulation, and Overview of Our PDE-Based Solution

A. Background

1) G-PCC Interpolation-Based Prediction: In G-PCC, the attribute coding scheme performs an interpolation-based prediction with reconstructed neighboring points [14]. More precisely, when compressing the attribute of point i, the set of decoded k-nearest neighbors N_i with the reconstructed attributes $f_0(j)_{j\in N_i}$ are found via the Euclidean distance metric. Then, the predicted attribute $f(i)$ of point i is calculated by an inverse distance weighted interpolation as follows:

$$f(i) = \text{Round}\left(\frac{1}{k}\cdot\frac{\sum_{j\in N_i}\frac{1}{d_{ij}{}^2}f_0(j)}{\sum_{j\in N_i}\frac{1}{d_{ij}{}^2}}\right), \tag{1}$$

where Round$(\cdot)$ represents a rounding operation on a floating-point number to the nearest integer. d_{ij} denotes the distance between the point i and its neighboring point j. k means the number of neighbors.

2) Continuous Diffusion-Based Inpainting: Consider a continuous 3D data domain $u(x) : \Omega \to \mathbb{R}^3$ maps 3D space to associated values [42]. With the known inpainting mask $J \subset \Omega$, a classic diffusion-based inpainting process usually derives a reconstruction $f(x) : \Omega \to \mathbb{R}^3$ for the missing data $x \in \Omega \setminus J$ by solving the following differential equation on the inpainting mask J:

$$(1 - c(x))Lf(x) - c(x)(f(x) - u(x)) = 0, \tag{2}$$

where L is the difference operator, and $c(x)$ is the characteristic function of x that is 1 when $x \subset J$, and 0 elsewhere. The first term introduces a smoothness constraint on the missing data via the difference operator, while the second term ensures that the values are preserved for the inpainting mask J.

The continuous solution of the diffusion-based inpainting process in Eq. (2) can be obtained by minimizing the diffusion energy defined by the following equation:

$$\min_{f(x)} E(f(x)) = \frac{1}{2}\int_{\Omega}\|Lf(x)\|^2\, dx, \tag{3}$$

where $\|\cdot\|$ means the Euclidean norm.

3) Point Cloud Difference Operator Construction: A solution of the PDE operator in 3D space is provided on point cloud graphs in Lozes's work [40]. To tackle the challenges posed by traditional PDE approaches [37], [38], [39] when handling irregular and sparse geometries within point clouds, a graph-based framework is proposed to leverage the connectivity structure of weighted graphs to define PDE operators. The graph-based PDE operators enable accurate computation of gradients and directional derivatives in point clouds. Designed to capture the intrinsic geometric properties of the data, these PDEs effectively mitigate the effects of irregular and sparse distributions in point clouds. Let $G = (V, E, w)$ be a weighted graph of a point cloud, where $V = \{v_1, \ldots, v_k\}$ is a set of k vertices and E is a set of undirected weighted edges. Let $w(u, v)$ be the weight of the edge (u, v) connecting vertices $u \in V$ and $v \in V$. Given a function $f : V \to \mathbb{R}^3$ mapping each vertex u to its attribute value $f(u)$, the difference operator of f on edge $(u, v) \in E$ is computed by:

$$(df)(u, v) = \sqrt{w(u, v)}(f(v) - f(u)). \tag{4}$$

The directional derivative of f at vertex u along the edge $e = (u, v)$ is defined as:

$$\left.\frac{\partial f}{\partial e}\right|_u = \partial_v f(u) = (df)(u, v). \tag{5}$$

The weighted gradient operator of function f at vertex u is defined as a vector with a set of directional derivatives along the edge (u, v) between vertex v and u as follows:

$$\begin{aligned}\nabla_w f(u) &= (\partial_v f(u) : (u, v))^\top, \\ &= \left(\partial_{v_1} f(u), \ldots, \partial_{v_k} f(u)\right)^\top, \forall (u, v_i) \in E.\end{aligned} \tag{6}$$

By leveraging the connectivity information embedded in weighted graphs, graph-based PDEs demonstrate superior performance compared to traditional PDE-based techniques [37], [38], [39]. However, two main weaknesses in the graph-based PDE method warrant consideration. First, the complexity of graph construction from point clouds poses computational challenges due to intricate edge creation and weight assignment processes. Second, the construction of graph-based PDEs relies on various parameters, such as the p-value in the p-Laplacian operator, weight function, and patch size, across diverse data types, significantly influencing results. Yet, there are no clear guidelines for optimal parameter selection. Hence, there is a need to investigate a more robust and efficient PDE operator for point clouds.

B. Problem Formulation

By formulating the discrete attribute prediction as a 3D data continuous interpolation with a diffusion model, we extend the distance-based attribute interpolation in G-PCC to a general diffusion-based inpainting solution. The principle is that only a subset of specifically chosen points is coded and stored, while attributes of the remaining regions are reconstructed using the PDE-based inpainting operator.

Let us consider a 3D point cloud domain $\Omega \subset \mathbb{R}^3$ with a set of decoded points $J \subset \Omega$ as the inpainting mask. Let $f_0(j)$ describe the known attribute of the decoded point $j \in J$. The attribute function $f : \Omega \to \mathbb{R}^3$ maps 3D positions to attribute values, which is initialized with $f(i) = f_0(i)$ when $i \in J$. Our goal of PDE-based attribute inpainting is to reconstruct the attributes $f(i)$ of the unknown point $i \in \Omega$ with a set of known points J in the inpainting domain $\Omega \setminus J$ based on smoothness assumptions. To this aim, we embed the diffusion-based attribute propagation process in the following PDE:

$$(1 - c(i))Lf(i) - c(i)(f(i) - f_0(i)) = 0, \tag{7}$$

where L denotes the difference operator. Our PDE-based attribute painting is given by the steady state of the evolution

equation:

$$\partial f = Lf \text{ on } \Omega \setminus J, \tag{8}$$

$$f = f_0 \text{ on } J, \tag{9}$$

where the choice of the difference operator L plays a crucial role in the attribute inpainting quality.

To achieve structure-adaptive attribute interpolation, we introduce anisotropic diffusion in the construction of the difference operator with $Lf = \text{div}(D\nabla f)$, and the Eq. (8) is presented as:

$$\partial f = \text{div}(D\nabla f), \tag{10}$$

where D is the diffusion tensor calculated by the attribute information around the point to be inpainted.

As presented in Eq. (3), the inpainting solution on continuous space $x \in \Omega$ can be obtained by minimizing the diffusion energy. Therefore, we can obtain the PDE-based attribute inpainting solution by minimizing the discretization formulation of the attribute diffusion energy defined as follows:

$$\min_{f(i)} E(f(i)) = \frac{1}{2} \sum_{i \in \Omega} \|D\nabla f(i)\|^2, \tag{11}$$

Finally, we formulate the attribute prediction problem in Eq. (1) as a local gradient optimization problem defined as follows:

$$\min_{f(i)} \nabla f(i)^\top D^\top D \nabla f(i), \tag{12}$$

where the design of the PDE operator $\nabla f(i)$ and the diffusion tensor D has a vital effect on the attribute reconstruction quality and the efficiency of the compression. We provide the detailed implementation of the PDE-based iterative optimization process in Section IV.

C. Generalized Diffusion-Based Attribute Inpainting Solution

In Section III-B, we provide the diffusion-based attribute inpainting formulation in Eq. (7) and the PDE-based attribute prediction solution in Eq. (11). In this part, we illustrate that the G-PCC k-NN based attribute predictor in Eq. (1) is a special case of our generalized diffusion-based solution.

The k-NN structure in G-PCC can be regarded as a graph with the vertex i and its neighbors $j \in N_i$ of the point cloud. We define the PDE operator $\nabla f(i)$ as the graph-based PDE operator in Eq. (6) and set the diffusion tensor D as the identity matrix. To get the predicted attribute $f(i)$, we plug in the PDE operator $\nabla_w f(i) = \sqrt{w(i,j)}(f(i) - f(j))$ and the diffusion tensor $D = I$ into the proposed attribute inpainting solution in Eq. (11), and we can get the following formulation:

$$\begin{aligned}\min_{f(i)} E(f(i)) &= \min_{f(i)} \sum_{j \in N_i} \|D\nabla_w f(i)\|^2, \\ &= \min_{f(i)} \sum_{j \in N_i} (\nabla_w f(i)^\top D^\top D \nabla_w f(i)), \\ &= \min_{f(i)} \sum_{j \in N_i} w(i,i)(f(i) - f(j))^2. \end{aligned} \tag{13}$$

To solve the problem in Eq. (13), we convert it to the following formulation and make its derivative equal to zero:

$$\begin{aligned}\frac{\partial E(f(i))}{\partial f(i)} &= 2 \sum_{j \in N_i} w(i,j)(f(i) - f(j)), \\ &= 0. \end{aligned} \tag{14}$$

Then, the predicted attribute $f(i)$ of point i can be obtained by the following formulation:

$$f(i) = \frac{\sum_{j \in N_i} w(i,j) f(j)}{\sum_{j \in N_i} w(i,j)}. \tag{15}$$

It is noted that the derived solution in Eq. (15) is in the same formulation of the G-PCC interpolation-based predictor in Eq. (1), by setting $w(i,j)$ as the reciprocal of the square of the distance between point i and j. We prove that the G-PCC k-NN based attribute predictor is a degraded case of our PDE-based attribute prediction solution.

D. Overview of Our PDE-Based Attribute Prediction Scheme

Based on the proposed diffusion-based attribute inpainting formulation, we propose a texture-aware attribute predictive coding framework in a PDE-based diffusion model. Fig. 1 presents the pipeline of the proposed framework with three processing stages: Point cloud decomposition, PDE-driven attribute interpolation, and EED-based attribute predictive coding. First, a point cloud is represented as two LoDs with seeds and non-seed points via geometry distribution analysis. Seed attributes are coded into the bitstream and then decoded as the attribute inpainting mask for non-seed points. Then, attributes of non-seed points are iteratively generated by our PDE-driven attribute inpainting module. Moreover, we introduce an adaptive early termination to speed up the iterative process. Later, EED-based attribute predictive coding is conducted point by point based on the interpolated point cloud and local diffusion tensors of inpainted points. Finally, attribute predicted residuals of points are coded into the bitstream.

IV. Our Approach

A. Point Cloud Decomposition for LoD Generation

We represent a point cloud as a two-layer LoD with seeds and non-seed points. As point cloud attribute prediction is formulated as a general diffusion-based inpainting problem in Section III-B, the goal of the inpainting process is to exploit a set of seeds as the inpainting mask to reconstruct the attributes of the left points. To ensure effective attribute diffusion with 3D inpainting masks in Fig. 2, a seed generation method is proposed based on geometric distribution analysis. We assume that the point cloud geometry is decoded and voxelized. Details of our LoD generation process are as follows.

First, we arrange all 3D-space points into a 1D point sequence with decoded geometry by the Morton order. Then, the LoD generation is performed iteratively over all reordered points. In the beginning, all points are marked as non-visited. Two LoDs L_0 and L_1 are constructed for gathering seeds and non-seed points, respectively. The process starts from

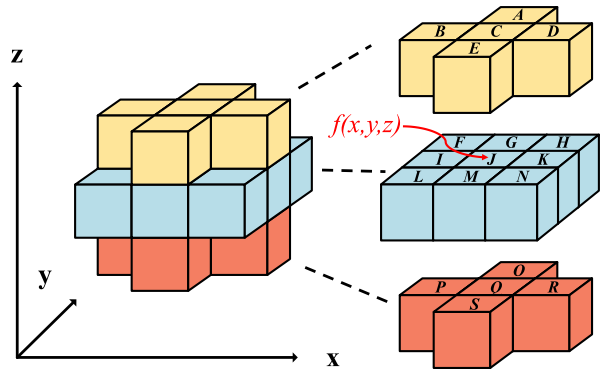


Fig. 2. The proposed 3D PDE-based inpainting mask with the cuboidal subdivision. To get the inpainted attributes $f(x, y, z)$ for point J with the positions (x, y, z), 18 neighboring locations centered around point J are selected for the calculation of partial derivatives $G_x(J)$, $G_y(J)$, and $G_z(J)$.

the first point, and we mark it as visited and assign it to the seed set L_0. Along the process, the current point would be ignored without any actions when it has been visited. Otherwise, we mark it as visited and add it to L_0. Moreover, local geometry analysis is performed for each seed, where all occupied neighbors sharing an edge or a face with the current point would be evaluated: If they are non-visited, we mark them as visited and add them into L_1; Otherwise, no actions would be taken. The proposed geometry distribution analysis ensures that every non-seed point in L_1 would have no less than one corresponding seed in L_0 at the same 3D inpainting mask.

B. Point Cloud PDE Operator Construction

It is difficult to directly extend 2D image PDEs [20] to 3D PDEs by replacing a 2D rectangular subdivision with a 3D cuboidal subdivision due to the sparsity in point cloud geometry. To address the limitations associated with graph-based point cloud PDE operators [40], we develop a robust 3D PDE operator tailored to handle the uncertainty from sparse geometry distributions within point clouds. Our goal is to efficiently calculate gradients and directional derivatives for accurate prediction solutions in point cloud attribute compression. In the proposed method, we introduce a novel 3D PDE-based inpainting mask featuring cuboidal subdivision. This mask is utilized to compute the inpainted attributes for each point within the point cloud.

As shown in Fig. 2, we select 18 neighboring locations to compute the attribute gradient of point J, aiming to ensure both gradient accuracy and computational simplicity. These neighboring locations are specifically selected to share an edge or a face with the central point. This preserves the necessary geometry correlation for accurate diffusion-based attribute interpolation. The utilization of 18 neighbors aligns with the design of our PDE operator, as these neighbors are easily identifiable with the cuboidal subdivision on the point cloud. Moreover, the computational efficiency of our approach benefits from this predetermined number of neighbors. Notably, while the central point is occupied, its neighboring regions may vary in occupancy status. Despite this variability, the proposed method ensures accurate calculation of attribute gradients at the central point by selecting these

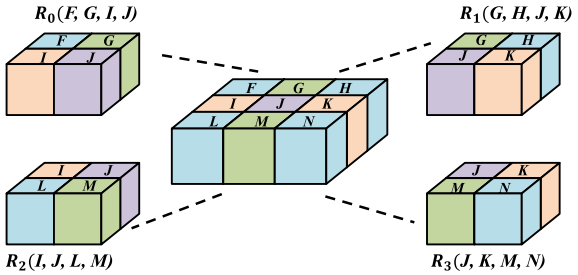


Fig. 3. The proposed local region division template for calculating the inner product of two directional derivatives in the PDE-based inpainting process. We only select 8 neighbors with no displacement along the z-axis of point J. To reduce complexity, nine discrete locations are clustered into four local regions R_0, R_1, R_2, and R_3. Then, partial derivatives $G_x(J)$, $G_y(J)$, and $G_z(J)$ of point J are derived with associated points in the local region.

18 neighboring locations. We define the attribute gradient at point J with the 3D geometric position (x, y, z) as follows:

$$\nabla f(J) = \nabla f(x, y, z) = (G_x(J), G_y(J), G_z(J))^\top, = \left(\frac{\partial f(x, y, z)}{\partial x}, \frac{\partial f(x, y, z)}{\partial y}, \frac{\partial f(x, y, z)}{\partial z}\right)^\top, \quad (16)$$

where $G_x(J)$, $G_y(J)$, and $G_z(J)$ denotes the partial derivatives of the point J's attribute in three dimensions.

With the proposed point cloud cuboidal subdivision presented in Fig. 2, we define G_x, G_y, and G_z as follows:

$$G_x(J) = \frac{1}{10} * [(\frac{1}{2} * \sum_{X=D,H,N,R} o(X)(f(X) - f(J))) - (\frac{1}{2} * \sum_{X=B,F,L,P} o(X)(f(X) - f(J))) + o(K)(f(K) - f(J)) - o(I)(f(I) - f(J))], \quad (17)$$

$$G_y(J) = \frac{1}{10} * [(\frac{1}{2} * \sum_{X=A,F,H,O} o(X)(f(X) - f(J))) - (\frac{1}{2} * \sum_{X=E,L,N,S} o(X)(f(i) - f(J))) + o(G)(f(G) - f(J)) - o(M)(f(M) - f(J))], \quad (18)$$

$$G_z(J) = \frac{1}{10} * [(\frac{1}{2} * \sum_{X=A,B,D,E} o(X)(f(X) - f(J))) - (\frac{1}{2} * \sum_{X=O,P,R,S} o(X)(f(X) - f(J))) + o(C)(f(C) - f(J)) - o(Q)(f(Q) - f(J))], \quad (19)$$

where X is the neighboring location of point J, and it can be 18 neighbors centered around J. $o(X)$ is the occupancy of location X. $o(X)$ is 1 when the location is occupied by points. Otherwise, $o(X)$ is 0. The scalar $1/10$ means the average of ten PDE operators calculated by associated neighbors. The scalar $1/2$ is the distance-based weight of co-planar neighbors, where 2 is the Manhattan distance between the current point and its co-planar neighbors.

As shown in Fig. 3, when calculating the inner product of two directional derivatives, we exploit the point J and

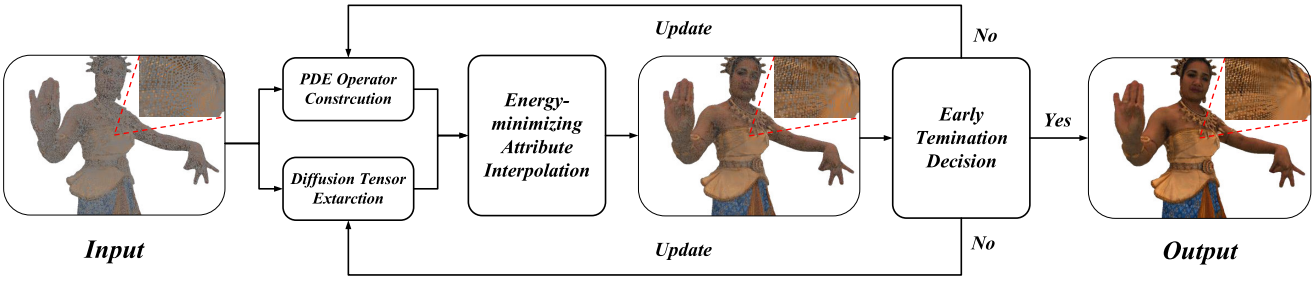


Fig. 4. Pipeline of the proposed PDE-driven attribute iterative interpolation scheme with adaptive early termination. It performs energy-minimizing attribute inpainting from seeds to non-seed points in an iterative update process.

its 8 neighboring points with no movement along the z-axis. There are $2^8 = 512$ possible occupancy situations of these 8 neighbors. To reduce complexity, we divide nine discrete points into four local regions R_0, R_1, R_2, and R_3, where each group R_i contains four locations (P_0, P_1, P_2, P_3). For example in the local region R_0, as presented in Fig. 3, the (P_0, P_1, P_2, P_3) represents the neighbors (F, G, I, J). To illustrate our definition of the inner product of two directional derivatives, we use the inner product derivatives $G_{xy}(R_i)$ of the x and y axes for the example. Therefore, G_{x_i} and G_{y_i} in the region R_i are derived by:

$$\begin{aligned} G_{x_i} = \frac{1}{4} * [&o(P_1) * o(P_0) * (f(P_1) - f(P_0)) \\ &+ o(P_3) * o(P_2) * (f(P_3) - f(P_2)) \\ &+ \frac{1}{2} * o(P_1) * o(P_2) * (f(P_1) - f(P_2)) \\ &+ \frac{1}{2} * o(P_3) * o(P_0) * (f(P_3) - f(P_0))], \end{aligned} \tag{20}$$

$$\begin{aligned} G_{y_i} = \frac{1}{4} * [&o(P_0) * o(P_2) * (f(P_0) - f(P_2)) \\ &+ o(P_1) * o(P_3) * (f(P_1) - f(P_3)) \\ &+ \frac{1}{2} * o(P_1) * o(P_2) * (f(P_1) - f(P_2)) \\ &+ \frac{1}{2} * o(P_0) * o(P_3) * (f(P_0) - f(P_3))], \end{aligned} \tag{21}$$

where the scalar $1/4$ means the average of four PDE operators calculated in a 2×2 local area. The scalar $1/2$ is the distance-based weight of co-planar neighbors, where 2 is the Manhattan distance between the current point and its co-planar neighbors.

Then, the inner product derivatives G_{xy} of x and y axes around the point J is computed via averaging the sum of $G_{x_i} * G_{y_i}$ in local regions:

$$G_{xy} = \frac{1}{4} * \sum_{i=0}^{3} G_{x_i} G_{y_i}. \tag{22}$$

In contrast to graph-based PDEs [40], the proposed PDEs operate directly on point clouds without the need for graph construction or manual parameter tuning. This approach streamlines computation and guarantees robust performance across different types of point clouds. By harnessing intrinsic geometric properties inherent in point clouds, our PDE operator presents an efficient alternative to graph-based PDEs, overcoming their limitations and providing effective solutions for handling complexities in point cloud attribute compression.

C. PDE-Driven Attribute Iterative Interpolation With Adaptive Early Termination

Fig. 4 presents the pipeline of the proposed PDE-driven attribute iterative interpolation scheme to perform energy-minimizing attribute inpainting from seeds to non-seed points.

1) Energy-Minimizing Attribute Interpolation: With our problem formulation in Section III-B, we perform diffusion-based attribute interpolation by minimizing the total attribute energy of all seeds $v \in N_{\text{seed}}$ and non-seed points $u \in N_{\text{nonSeed}}$ in the point cloud as follows:

$$\min \sum_{v \in N_{\text{seed}}} w_{\text{seed}} E(f(v)) + \sum_{u \in N_{\text{nonSeed}}} w_{\text{nonSeed}} E(f(u)), \tag{23}$$

where w_{seed} and w_{nonSeed} are the attribute energy weights of seeds and non-seed points, respectively. As attributes of seeds are more reliable than inpainted attributes of non-seed points, we set w_{seed} to be greater than w_{nonSeed}.

The local attribute energy of a point with the position (x, y, z) is defined as:

$$E(f(x, y, z)) = \nabla f(x, y, z)^{\top} T_{(x,y,z)} \nabla f(x, y, z), \tag{24}$$

where the attribute gradient $\nabla f(x, y, z)$ is a vector with partial derivatives G_x, G_y, and G_z along the x, y, and z axes, representing the rate of attribute changes in 3D space. The diffusion tensor $T_{(x,y,z)}$ is a 3×3 matrix that guides the diffusion process by specifying the amount of diffusivity in the diffusion directions. In our implementation, we use the diffusion tensor T to present $D^{\top} D$ in Eq. (12).

The choice of the diffusion tensor $T_{(x,y,z)}$ is important for point cloud attribute inpainting quality. To enable edge-enhancing anisotropic inpainting in the diffusion-based attribute interpolation, we define T with the Charbonnier diffusivity function g [43] as follows:

$$T = g\left(\nabla f_\sigma \nabla f_\sigma^{\top}\right), \tag{25}$$

$$g\left(\nabla f_\sigma \nabla f_\sigma^{\top}\right) := \frac{1}{\sqrt{1 + \frac{(\nabla f_\sigma \nabla f_\sigma^{\top})}{\lambda^2}}}, \tag{26}$$

$$f_\sigma(i) = \frac{\sum_{j \in N_i} w(i, j) f(j)}{\sum_{j \in N_i} w(i, j)}, \tag{27}$$

$$w(i, j) = e^{-d_{i,j}^2 / 2\sigma^2}, \tag{28}$$

where $f_\sigma(i)$ is the Gaussian smoothing of the attribute $f(i)$ on point i by a Gaussian with standard deviation σ, and $w(i,j)$ is the weighted attribute correlation defined by the $L2$ distance $d_{i,j}$ between point i and j. λ is a contrast parameter. N_i are neighbors being co-linear and co-planar to point i.

To simplify the way to calculate the diffusion tensor T in Eq. (25), we decompose T as a symmetric matrix by:

$$\begin{aligned} T &= (n_\sigma, n_\perp, n_\times) \begin{pmatrix} g(\nabla f_\sigma{}^\top \nabla f_\sigma) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} (n_\sigma, n_\perp, n_\times)^\top \\ &= (n_\sigma, n_\perp, n_\times) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} (n_\sigma, n_\perp, n_\times)^\top \\ &\quad - (n_\sigma, n_\perp, n_\times) \begin{pmatrix} 1 - g(\nabla f_\sigma{}^\top \nabla f_\sigma) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} (n_\sigma, n_\perp, n_\times)^\top \\ &= I + (g(\nabla f_\sigma{}^\top \nabla f_\sigma) - 1) n_\sigma n_\sigma^\top, \end{aligned} \tag{29}$$

where I is a 3×3 identity matrix, and n_σ is the normal vector of the gradient ∇f_σ. $n_\perp$ and $n_\times$ are two orthogonal vectors of n_σ. $(\sigma, n_\perp, n_\times)$ forms an orthogonal matrix.

Plugged our designed PDE operator $\nabla f(x,y,z)$ in Eq. (16) and the diffusion tensor T in Eq. (29) into the energy function in Eq. (24), we obtain the final formulation of the attribute energy function on the point (x,y,z) by:

$$\begin{aligned} E(f(x,y,z)) &= \nabla f(x,y,z)^\top T_{(x,y,z)} \nabla f(x,y,z) \\ &= T_{(x,y,z)}(0,0) G_{x^2} + T_{(x,y,z)}(1,1) G_{y^2} \\ &\quad + T_{(x,y,z)}(2,2) G_{z^2} + 2 * T_{(x,y,z)}(0,1) G_{xy} \\ &\quad + 2 * T_{(x,y,z)}(0,2) G_{xz} \\ &\quad + 2 * T_{(x,y,z)}(1,2) G_{yz}. \end{aligned} \tag{30}$$

By substituting Eq. (24)-(30) into Eq. (23), the total energy minimization problem in Eq. (23) can be reformulated as a quadratic optimization problem. We obtain the optimal inpainted attributes by minimizing the quadratic expression of point cloud attributes through an iterative process with adaptive early termination.

2) Attribute Iterative Inpainting Strategy With Adaptive Early Termination: Instead of separately minimizing the local energy for each point, we perform the energy-minimizing attribute inpainting for all non-seed points simultaneously for the best outcomes. This process ensures the PDE-driven attribute inpainting can fully utilize the attribute spatial correlation of neighbors in a wide-range diffusion process of point clouds.

As presented in Fig. 4, we input the attributes of seed and initial attributes of non-seed points in the pipeline. Attribute PDE operators and diffusion tensors are first calculated for all points. Interpolated attributes are derived by solving the total energy minimization problem with PDE operators and diffusion tensors. After updating non-seed points with interpolated attributes in an iteration, we introduce an early termination decision module to evaluate the reconstruction status of the current attribute interpolation. It is noted that, when the complexity of local textures varies, the required iterations for a better attribute interpolation also vary. Local regions with complex textures usually need more iterations to achieve a good inpainting effect than those with relatively smooth attributes. That is the reason why we introduce an adaptive early termination in the iterative attribute inpainting process. If there are still points to be updated after one round of inpainting, the iteration continues, and attribute PDE operators and diffusion tensors would be recalculated with the latest-updated attributes of points. When the iterative process ends, we finally obtain the inpainted attributes for all non-seed points. Details of the proposed attribute iterative interpolation with an adaptive early termination scheme are given in Algorithm 1.

Algorithm 1 Proposed Attribute Iterative Interpolation With Adaptive Early Termination Scheme

Input: Initial attributes f_0, seed LoD L_0, non-seed LoD L_1.
Output: Interpolated attributes f, diffusion tensor T.

Claculate PDE operator ∇f, initial $T = I$, and compute Gaussian filtered attributes f_σ.
for point u in L_0 **do**
  $\text{flag}_{\text{inpaint}}(u) = \text{False}$, $\text{flag}_{\text{update}}(u) = \text{False}$.
end for
for point u in L_1 **do**
  $\text{flag}_{\text{inpaint}}(u) = \text{True}$, $\text{flag}_{\text{update}}(u) = \text{False}$.
end for
for iteration $i = 1 : N$ **do**
  for point u in $L_0 + L_1$ **do**
    if $\text{flag}_{\text{inpaint}}(u) = \text{False}$ **then**
      continue
    end if
    Compute energy-minimizing inpainted attribute $f(u)$.
    if $f(u) = f_0(u)$ **then**
      $\text{flag}_{\text{inpaint}}(u) = \text{False}$.
    else
      $f0(u) = f(u)$, and $\text{flag}_{\text{update}}(u) = \text{True}$.
      Update $\text{flag}_{\text{update}}(v) = \text{True}$ for all neighbors v in the point u's inpainting mask.
    end if
  end for
  for u in $L_0 + L_1$ **do**
    if $\text{flag}_{\text{update}}(u) = \text{True}$ **then**
      Update $f_\sigma(u)$.
    end if
  end for
  for u in $L_0 + L_1$ **do**
    if $\text{flag}_{\text{update}}(u) = \text{True}$ **then**
      Update $T(u)$.
    end if
  end for
end for
return f and T.

D. EED-Based Attribute Predictive Coding

Because attributes of seeds have already been coded into the bitstream, the attribute predictive coding is performed on

non-seed points in a point-to-point manner. Considering there exists the directional bias in point cloud interpolated attributes, it would be desirable to guide the prediction towards the orientation of underlying attribute distribution. This bias arises from spatial variations or patterns within the distribution, where certain attributes are more pronounced along specific orientations within the point cloud. This phenomenon is particularly notable in regions characterized by intricate textures and abundant edges. Therefore, we introduce the edge-enhancing diffusion (EED) predictor for texture-aware attribute prediction. Details of our EED-based attribute predictive coding scheme are as follows.

We use the interpolated attributes generated from our aforementioned PDE-driven inpainting as attribute predictors of non-seed points to be coded. Then, attribute prediction residuals are quantized and entropy-coded into the bitstream. Meanwhile, those residuals are decoded and reconstructed to update the interpolated attribute of the current non-seed. Later, texture-aware diffusivity analysis is performed to capture the change in the local attribute structure. To avoid the prediction error accumulation and improve the predictive accuracy of left non-seed points, we use the recalculated diffusion tensor in an EED-based attribute inpainting process to generate a new version of the interpolated attributes of left non-seed points. We divide all points into three categories based on their coding status: coded seeds $u \in N_0$, predicted non-seed points $v \in N_1$, and non-seed points to be predicted $i \in N_2$. Then, the updated attributes $f(i)$ of non-seed points to be predicted i are derived by minimizing the total attribute energy function defined as:

$$\min_{f(i)} E(f(i)) = \sum_{u \in N_0} w_0 E(f(u)) + \sum_{v \in N_1} w_1 E(f(v)) + \sum_{i \in N_2} w_2 E(f(i)), \tag{31}$$

where w_0, w_1, and w_2 are attribute energy weights of seeds, predicted non-seed points, and non-seed points to be predicted, respectively. As points in N_0 are deemed more reliable than those in N_1, and similarly, points in N_1 are considered more reliable than those in N_2, we assign weights accordingly. Specifically, w_0 is set to be greater than w_1, and w_1 is set to be greater than w_2.

The solving process of our energy minimization problem is roughly the same as the previous attribute interpolation process in Section IV-C. The difference is that we directly use attributes f to compute the diffusion tensor in Eq. (32) instead of using the Gaussian smoothing f_σ.

$$D = g\left(\nabla f \nabla f^{\top}\right). \tag{32}$$

In this way, we can get the updated attribute predictors for non-seed points to be predicted. Our EED-based attribute prediction scheme can fully utilize the attribute spatial correlation for better texture-aware attribute prediction. Finally, attribute residuals of all non-seed points are encoded into the total bitstream.

Fig. 5. Test point cloud datasets. From left to right and up to down: *Andrew*, *Ricardo*, *David*, *Phil*, *Sarah*, *Facade*, *Queen*, *Thaidancer*, *Soldier*, *Longdress*, *Redandblack*, *Loot*, *Basketball* and *Dancer*.

V. Experimental Results

A. Experimental Setup

We conduct a series of experiments to validate the effectiveness of the proposed point cloud attribute prediction framework. A set of standard point clouds with different attribute characteristics from MPEG [44] and JPEG [45] are tested for point cloud attribute compression. Test datasets are visualized in Fig. 5. We compare the proposed framework with several competitive platforms from industry and academia: (i) **MPEG G-PCC(v23)**, (ii) **BAAC** (Block-Adaptive Attribute Coder [17]), and (iii) **HAC** (Hybrid Attribute Coder [25]). About the evaluation metrics, we follow the MPEG common test conditions [44] and use the bits per point (bpp) to denote the total attribute bitrate. The Bjontegaard Delta Bitrate (BD-BR) value [46] is adopted to measure average bitrate differences between rate-distortion (R-D) curves for the same objective quality. Besides, the Peak Signal-to-Noise Ratio (PSNR) is calculated to measure the reconstructed attribute distortion. Once $\mathrm{PSNR_Y}$, $\mathrm{PSNR_U}$, and $\mathrm{PSNR_V}$ of attribute components Y, U, and V are collected, the combined $\mathrm{PSNR_{YUV}}$ can be computed as $\mathrm{PSNR_{YUV}} = (6\mathrm{PSNR_Y} + \mathrm{PSNR_U} + \mathrm{PSNR_V})/8$.

All experiments are performed on a computer with an Intel i7 8700K CPU (3.7GHz), and 64GB RAM. In the attribute iterative interpolation of Section IV-C, we set w_{seed} as 6 and w_{nonSeed} as 1 in Eq. (23). The maximum of iterations is 30. Regarding the diffusion tensor calculation in Eq. (25),

TABLE I
PERFORMANCE COMPARISONS OF THE PROPOSED FRAMEWORK WITH HAC, BAAC, AND G-PCC ON POINT CLOUD ATTRIBUTE LOSSY COMPRESSION

Test Dataset	HAC					BAAC					G-PCC (v23)					Proposed	
	BD-BR Gain (%)			Time (s)		BD-BR Gain (%)			Time (s)		BD-BR Gain (%)			Time (s)		Time (s)	
	Y	U	V	Enc.	Dec.	Y	U	V	Enc.	Dec.	Y	U	V	Enc.	Dec.	Enc.	Dec.
Andrew	-17.71	-29.29	-26.45	755.26	404.84	-2.33	-11.40	-7.88	470.33	363.19	-13.17	-10.56	-10.79	5.68	5.08	28.92	28.48
Ricardo	-30.57	-43.87	-40.84	561.05	321.08	-7.93	-20.67	-17.59	349.39	290.04	-17.26	-13.60	-14.35%	4.37	3.88	21.17	20.81
David	-27.27	-40.41	-39.13	625.55	271.91	-4.08	-21.13	-21.00	246.77	104.25	-22.01	-21.41	-21.69	7.00	6.17	32.89	32.19
Phil	-32.57	-44.25	-40.62	685.13	295.22	-14.24	-22.75	-18.27	270.27	113.18	-21.46	-16.22	-17.05	7.79	6.75	36.28	35.64
Sarah	-22.26	-35.92	-32.44	506.40	233.07	-3.49	-16.99	-16.59	476.44	377.91	-16.20	-17.52	-11.93	6.58	5.97	27.92	26.67
Facade	-20.81	-25.84	-24.16	1147.35	729.47	-3.50	-8.87	-7.38	782.94	421.80	-2.37	-13.96	-20.21	14.64	12.96	69.30	67.91
Queen	-23.43	-38.52	-33.61	473.10	302.92	-6.41	-13.35	-6.68	150.57	93.99	-10.50	-20.02	-18.55	3.47	3.02	18.77	18.45
Thaidancer	-19.93	-27.07	-25.18	976.80	607.89	-5.31	-12.06	-10.50	666.56	419.82	-11.40	-18.66	-17.24	12.42	10.98	39.27	38.09
Soldier	-18.82	-13.57	-9.67	568.15	318.68	-3.37	-6.28	-3.16	293.83	152.66	-19.78	-31.08	-31.77	4.04	3.63	16.92	16.52
Longdress	-17.77	-35.21	-29.53	626.21	388.42	-0.70	-13.76	-7.95	285.10	152.47	-6.83	-14.92	-12.07	3.11	2.71	14.63	14.18
Readandblack	-15.43	-25.39	-26.97	438.50	255.00	-4.46	-10.58	-13.62	320.45	186.02	-15.29	-15.83	-15.10	2.82	2.40	12.22	11.98
Loot	-19.19	-15.08	-7.85	274.64	210.09	2.13	-5.75	-0.36	284.92	160.25	-15.24	-25.94	-20.35	2.90	2.59	13.05	12.64
Basketball	-19.18	-45.67	-21.21	945.79	571.42	-2.16	-34.84	-13.42	645.40	303.19	-11.72	-11.01	-17.66	11.52	9.92	48.97	46.25
Dancer	-20.04	-35.49	-6.36	871.38	483.20	0.97	-25.30	-2.46	560.74	253.24	-14.70	-14.54	-21.50	9.54	8.34	41.28	40.39
Average	**-21.78**	**-32.54**	**-26.00**	675.38	385.23	**-3.92**	**-15.98**	**-10.49**	414.55	242.29	**-14.14**	**-17.52**	**-17.87**	6.85	6.03	30.11	29.30

the Gaussian standard deviation σ is set as 3, and the λ of diffusivity is set as 0.6. In the attribute prediction of Section IV-D, we set w_0 as 6, w_1 as 4, and w_2 as 1 in Eq. (31). For attribute lossy compression, the quantization step of seeds $\mathrm{qp}_{\mathrm{seed}}$ is set as $\max(1, \mathrm{qp}_{\mathrm{nonSeed}}/2)$, in which $\mathrm{qp}_{\mathrm{nonSeed}}$ is the quantization step of non-seed points.

B. Comparisons With the State-of-the-Art Methods

1) Compression Performance Evaluation: Table I summarizes the performance comparison of our scheme and comparative platforms HAC, BAAC, and G-PCC(v23) on point cloud attribute lossy compression. Compared to HAC, our scheme obtains an average of 21.78%, 32.54%, and 26% BD-BR gains on coding Y, U, and V components. Compared to BAAC, we can averagely get 3.92%, 15.98%, and 10.49% BD-BR gains on Y, U, and V components. HAC and BAAC are two typical graph transform based attribute lossy codecs, while our scheme can beyond the limitations of the graph and achieve better coding performance. Compared to the latest standard platform G-PCC (v23), an average of 14.14%, 17.52%, and 17.87% BD-BR gains can be achieved on coding Y, U, and V components. The coding performance improvement over G-PCC is achieved in the proposed scheme by leveraging all neighbors from different directions in a diffusion-based iterative interpolation process. R-D curves with total attribute bitrates and the combined PSNR values of the proposed scheme and comparative platforms on point cloud attribute lossy compression are presented in Fig. 6. Consistent R-D performance gains can be obtained by the proposed scheme on the vast majority of datasets at different bitrate points. Results demonstrate that the proposed attribute predictive coding scheme can handle different types of point clouds and achieve significant coding performance improvement over the state-of-the-art methods at different bitrate points.

2) Reconstruction Quality Evaluation: To further verify the advancement of the proposed scheme on point cloud attribute reconstruction quality after lossy compression, we conduct subjective and objective quality evaluations on decoded point clouds with distorted attributes generated by our scheme and comparative platforms. Datasets *Soldier* and *Thaidancer* with different attribute characteristics and rich textures are chosen in this quality assessment. Fig. 7 presents the subjective quality comparison of decoded point clouds between our scheme and the state-of-the-art methods at similar attribute bitrates. We zoom in point clouds in the red box for a better representation. From the contents in the yellow circle, we see that the proposed method can better preserve attribute details, especially in areas with complex textures and rich edges.

Moreover, the objective quality comparison between our scheme and the state-of-the-art methods is presented in Fig. 8. Each figure presents the decoded point cloud with the total attribute bitrate and the PSNR for the luma component. The proposed scheme can obtain the highest PSNR values at the lowest bitrate expenses compared to comparative platforms. Besides, we visualize the mean square error (MSE) of the luma component for all points with error color-coding maps. These maps effectively display the attribute distortion between the original and decoded point clouds. Results show that the proposed scheme can achieve better subjective and objective quality compared to other platforms in point cloud attribute lossy compression.

C. Ablation Studies

We perform a set of ablation studies to objectively verify the effectiveness of our key modules in the proposed attribute compression framework. Validation results on point cloud attribute lossy compression are presented in Table II. We independently discuss each module as follows.

1) Validity of PDE-Based Attribute Interpolation: The proposed PDE-based attribute interpolation module illustrated in Section IV-C can generate a reconstructed point cloud with smoothly interpolated attributes, where attributes of non-seed points are inpainted from decoded seeds in a diffusion process. The module can effectively improve the attribute reconstruction quality by fully utilizing the attribute correlation in neighboring regions. Table II shows that our attribute

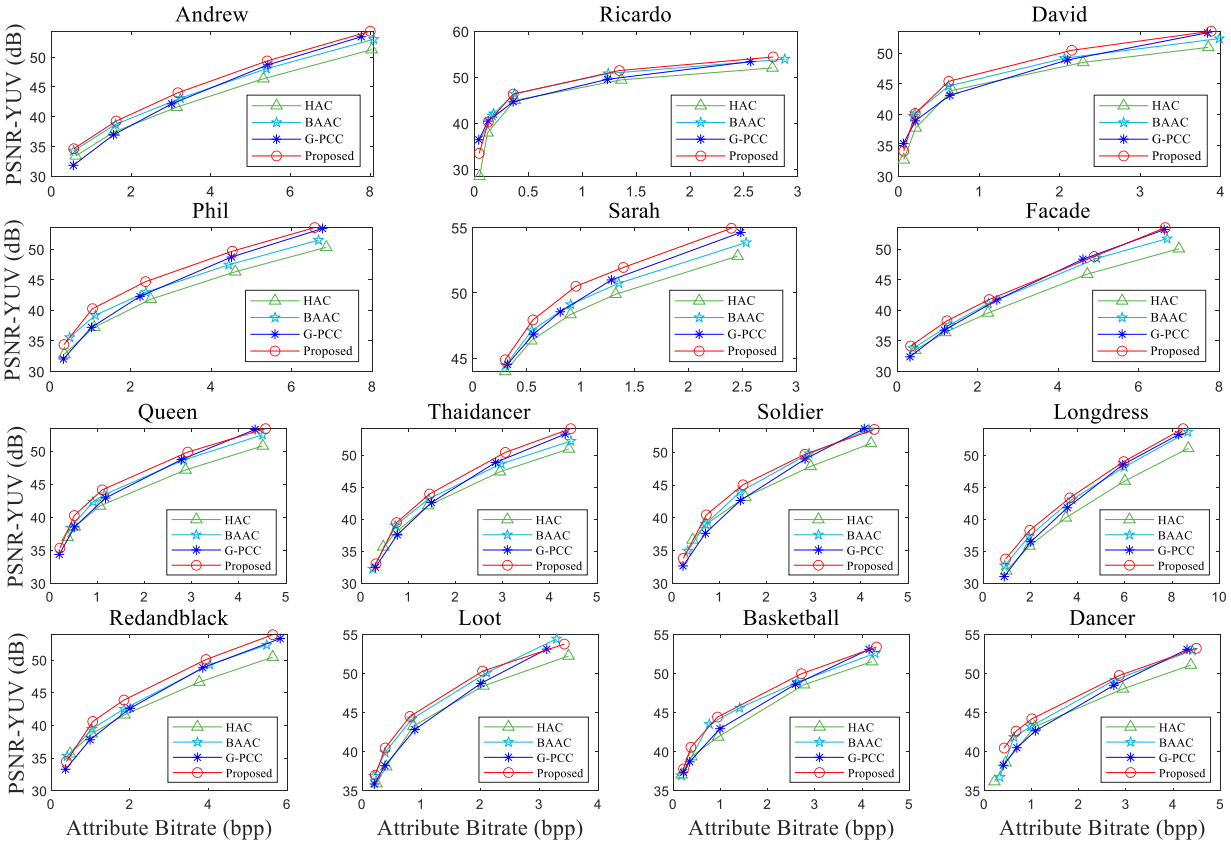
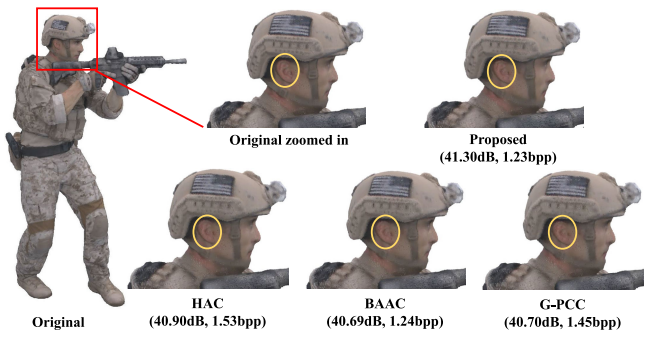
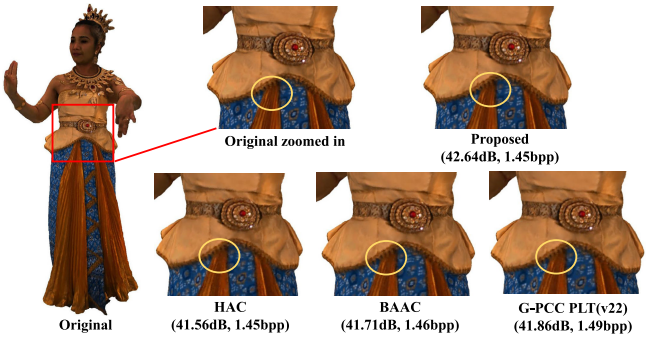


Fig. 6. R-D performance comparison of the proposed method, HAC, BAAC, and G-PCC on point cloud attribute lossy compression.



(a) Subjective quality evaluation on *Soldier*



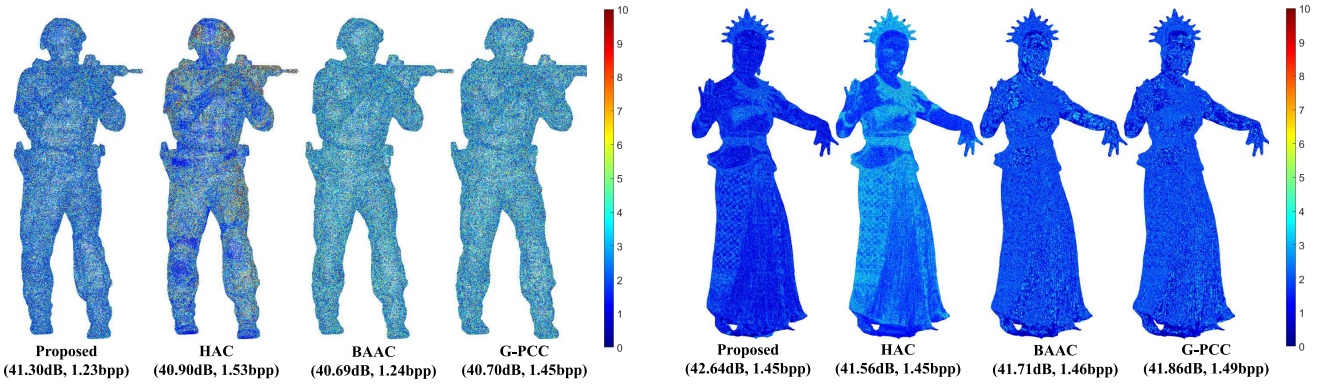
(b) Subjective quality evaluation on *Thaidancer*

Fig. 7. Subjective quality comparison of decoded point clouds between our scheme and the state-of-the-art methods on the *Soldier* and *Thaidancer*. Each sub-figure presents the reconstructed attribute details in the red box, as well as the associated attribute bitrate (bpp) and PSNR (dB) for the luma component. Local details within the yellow circle can display the quality differences of decoded point clouds.

interpolation module can averagely bring 5.77%, 9.55%, and 10.25% BD-BR gains on the coding of Y, U, and V components. Results validate the advantage of the proposed attribute interpolation module in the proposed framework. It outperforms the interpolation-based prediction in G-PCC, which directly uses decoded neighboring attributes as prediction references.

2) Validity of EED-Based Attribute Prediction: The proposed EED-based attribute prediction module, as illustrated in Section IV-D, can generate texture-aware attribute predictors. In this approach, edge-enhancing diffusion tensors are extracted from local regions to effectively capture the underlying attribute structure. To validate the effectuality of EED, we disable the anisotropic diffusion by using an identity array as a diffusion tensor. Table II demonstrates that the EED module yields BD-BR gains of 5.55%, 2.44%, and 2.26% for the attribute lossy coding of Y, U, and V components, respectively. Results demonstrate the benefits of the proposed EED module, which can effectively improve the accuracy of attribute prediction.

3) Validity of Attribute Iterative Inpainting Strategy With Adaptive Early Termination: As shown in Fig. 4, we design an iterative inpainting strategy with adaptive early termination to efficiently perform progressive attribute interpolation. To confirm the effectiveness of this scheme, we present the attribute interpolation quality at different iterations in Fig. 9. As the



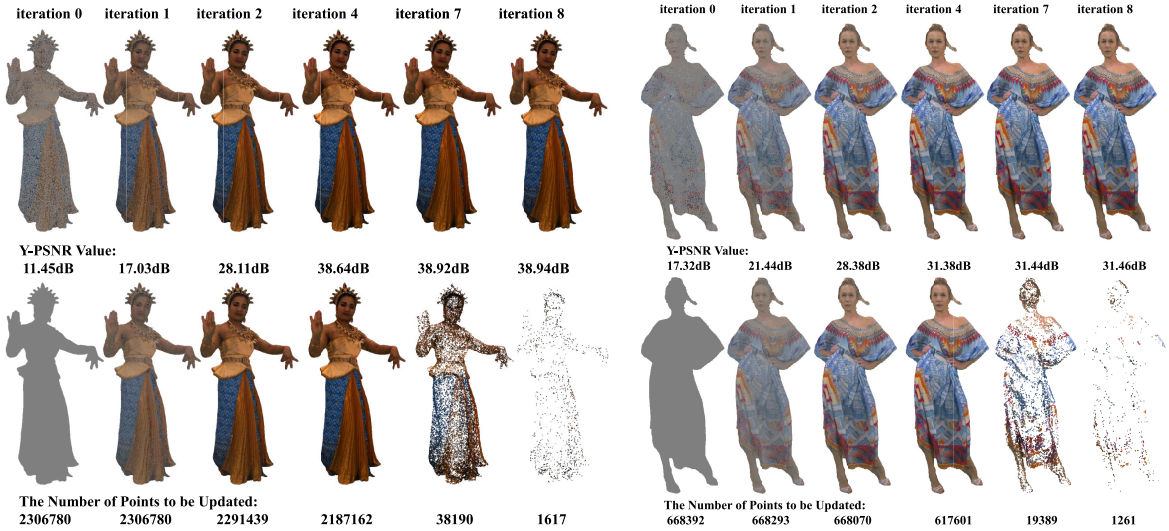
(a) Objective quality evaluation on *Soldier* (b) Objective quality evaluation on *Thaidancer*

Fig. 8. Objective quality comparison of decoded point clouds between our scheme and the state-of-the-art methods on the *Soldier* and *Thaidancer*. The mean square errors of the luma component between the original and decoded point clouds are quantitatively visualized with error color-coding maps.

TABLE II

ABLATION STUDIES ON KEY MODULES OF THE PROPOSED FRAMEWORK ON POINT CLOUD ATTRIBUTE LOSSY COMPRESSION

Test Dataset	w/o Interpolation			w/o EED			w/o Early Termination				
	BD-BR Gain (%)			BD-BR Gain (%)			BD-BR Gain (%)			Time Reduction (%)	
	Y	U	V	Y	U	V	Y	U	V	Enc.	Dec.
Andrew	-2.88	-5.86	-6.46	-6.14	-3.50	-2.95	0.13	0.03	0.00	62.73	63.67
Ricardo	-5.67	-8.05	-5.04	-1.10	-0.48	-0.29	0.34	-0.03	-0.08	63.94	64.43
David	-5.63	-7.61	-8.48	-1.99	-0.87	-0.52	0.30	0.13	-0.05	64.21	64.63
Phil	-5.82	-8.38	-7.54	-5.63	-2.14	-2.62	0.28	0.02	0.10	63.24	63.56
Sarah	-7.68	-10.47	-13.08	-1.03	-0.45	-0.23	0.29	0.12	-0.15	65.78	68.73
Facade	-3.81	-8.13	-11.14	-1.23	2.23	4.80	0.01	0.00	-0.15	65.59	66.40
Queen	-3.81	-8.21	-7.81	-6.94	-4.71	-6.69	0.15	0.02	0.11	65.50	65.64
Thaidancer	-7.80	-11.13	-14.31	-6.96	-2.61	-2.31	0.08	-0.03	-0.06	67.22	67.47
Soldier	-5.97	-14.97	-16.19	-11.46	-2.44	-2.10	0.10	-0.30	-0.22	66.02	66.29
Longdress	-5.31	-9.76	-9.02	-9.92	-10.51	-10.88	0.03	0.00	0.00	62.58	63.17
Readandblack	-6.66	-8.07	-7.38	-8.74	-6.70	-10.32	0.14	0.02	0.14	64.99	65.28
Loot	-5.96	-14.88	-13.87	-9.26	-2.19	-1.45	0.31	-0.07	-0.18	66.10	66.56
Basketball	-6.75	-8.40	-10.35	-3.85	-0.33	1.59	0.11	-0.14	-0.22	65.85	66.82
Dancer	-7.06	-9.78	-12.84	-3.44	0.54	2.39	0.08	-0.21	-0.33	66.27	66.62
Average	**-5.77**	**-9.55**	**-10.25**	**-5.55**	**-2.44**	**-2.26**	**0.17**	**-0.03**	**-0.08**	**65.00**	**65.66**



(a) Attribute progressive inpainting on *Thaidancer* (b) Attribute progressive inpainting on *Longdress*

Fig. 9. Demonstration on the effectiveness of the proposed attribute iterative inpainting scheme on *Thaidancer* and *Longdress*. The first row presents the reconstructed point cloud with inpainted attributes at different iterations, and the second row presents the point cloud composed of points to be updated at different iterations. As the iteration increases, the quality of the reconstructed point cloud increases, while the number of points to be updated decreases.

number of iterations increases, the quality of the reconstructed point cloud gradually improves, while the number of points to be updated gradually decreases. To avoid dispensable iterations at smooth interpolated regions, we use the number of points to

be updated after the iteration as the indicator for the proposed adaptive early termination. Table II shows that the proposed adaptive early termination can obtain 65.00% encoder time and 65.66% decoder time reduction on attribute lossy compression. Meanwhile, 0.17% BD-BR gains are averagely obtained on the lossy coding of the Y component, while a marginal BD-BR loss like 0.03% on the U component and 0.08% on the V component is introduced by the proposed early termination. This is an acceptable loss in the performance tradeoff between the compression ratio and computational complexity. The results manifest the effectiveness of the proposed iterative inpainting strategy with adaptive early termination.

VI. Conclusion

In this paper, we propose a diffusion-based attribute progressive prediction framework for point cloud attribute compression. Instead of the linear interpolation-based prediction paradigm, we solve the discrete attribute prediction as the continuous diffusion-based interpolation, which can achieve accurate intra prediction and improve coding performance. PDE operators are designed to conduct optimized attribute interpolation from seeds to unknowns, fully utilizing attribute spatial correlation in neighboring regions. An iterative update strategy with early termination is proposed to guarantee the effectiveness and efficiency of the diffusion-based attribute interpolation. Moreover, an EED-based attribute prediction is devised to perform texture-aware intra prediction with interpolated attributes, which can obtain smooth prediction in local regions while edge details can be preserved. Experimental results demonstrate that our proposed framework outperforms several competitive platforms in terms of compression performance and attribute reconstruction quality. Besides, ablation studies can prove the effectiveness of key modules in our scheme.

References

[1] Q. Wang and M.-K. Kim, "Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018," *Adv. Eng. Informat.*, vol. 39, pp. 306–319, Jan. 2019.

[2] P. Gao, S. Luo, and M. Paul, "Rate-distortion modeling for bit rate constrained point cloud compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 5, pp. 2424–2438, May 2023.

[3] Z. Wei, B. Niu, H. Xiao, and Y. He, "Isolated points prediction via deep neural network on point cloud lossless geometry compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 407–420, Jan. 2023.

[4] D. T. Nguyen and A. Kaup, "Lossless point cloud geometry and attribute compression using a learned conditional probability model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 4337–4348, Jan. 2023, doi: 10.1109/TCSVT.2023.3239321.

[5] X. Zhang and W. Gao, "Adaptive geometry partition for point cloud compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4561–4574, Dec. 2021.

[6] F. Song, Y. Shao, W. Gao, H. Wang, and T. Li, "Layer-wise geometry aggregation framework for lossless LiDAR point cloud compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4603–4616, Dec. 2021.

[7] W. Zhu, Y. Xu, D. Ding, Z. Ma, and M. Nilsson, "Lossy point cloud geometry compression via region-wise processing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4575–4589, Dec. 2021.

[8] Y. Shao, G. Li, Q. Zhang, W. Gao, and S. Liu, "Nonrigid registration-based progressive motion compensation for point cloud geometry compression," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, doi: 10.1109/TGRS.2023.3321289.

[9] B. Zhao, W. Lin, and C. Lv, "Fine-grained patch segmentation and rasterization for 3-D point cloud attribute compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4590–4602, Dec. 2021.

[10] H. Liu, H. Yuan, Q. Liu, J. Hou, H. Zeng, and S. Kwong, "A hybrid compression framework for color attributes of static 3D point clouds," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1564–1577, Mar. 2022.

[11] S. Gu, J. Hou, H. Zeng, and H. Yuan, "3D point cloud attribute compression via graph prediction," *IEEE Signal Process. Lett.*, vol. 27, pp. 176–180, 2020.

[12] S. Gu, J. Hou, H. Zeng, H. Yuan, and K.-K. Ma, "3D point cloud attribute compression using geometry-guided sparse representation," *IEEE Trans. Image Process.*, vol. 29, pp. 796–808, Aug. 2019, doi: 10.1109/TIP.2019.2936738.

[13] Y. Shao, Z. Zhang, Z. Li, K. Fan, and G. Li, "Attribute compression of 3D point clouds using Laplacian sparsity optimized graph transform," *IEEE Vis. Commun. Image Process. (VCIP)*, pp. 1–4, Dec. 2017.

[14] *G-PCC Test Model V23*, Standard ISO WG7N00645, MPEG 3D Graph. Coding, 2023.

[15] L. Wei, S. Wan, F. Yang, and Z. Wang, "Content-adaptive level of detail for lossless point cloud compression," *APSIPA Trans. Signal Inf. Process.*, vol. 11, no. 1, pp. 1–28, 2022.

[16] J. Chen, L. Yu, and W. Wang, "Hilbert space filling curve based scan-order for point cloud attribute compression," *IEEE Trans. Image Process.*, vol. 31, pp. 4609–4621, 2022.

[17] F. Song, G. Li, X. Yang, W. Gao, and S. Liu, "Block-adaptive point cloud attribute coding with region-aware optimized transform," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 1–12, Jul. 2023.

[18] I. Galić, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, and H.-P. Seidel, "Image compression with anisotropic diffusion," *J. Math. Imag. Vis.*, vol. 31, nos. 2–3, pp. 255–269, Jul. 2008.

[19] P. Peter and J. Weickert, "Colour image compression with anisotropic diffusion," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 4822–4826.

[20] P. Peter, L. Kaufhold, and J. Weickert, "Turning diffusion-based image colorization into efficient color compression," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 860–869, Feb. 2017.

[21] R. Schnabel and R. Klein, "Octree-based point-cloud compression," *PBG SIGGRAPH*, vol. 3, 2006, pp. 111–121.

[22] Y. Huang, J. Peng, C.-C. J. Kuo, and M. Gopi, "A generic scheme for progressive point cloud coding," *IEEE Trans. Vis. Comput. Graphics*, vol. 14, no. 2, pp. 440–453, Mar. 2008.

[23] C. Zhang, D. Florencio, and C. Loop, "Point cloud attribute compression with graph transform," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 2066–2070.

[24] R. A. Cohen, D. Tian, and A. Vetro, "Attribute compression for sparse point clouds using graph transforms," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1374–1378.

[25] Y. Shao, Q. Zhang, G. Li, Z. Li, and L. Li, "Hybrid point cloud attribute compression using slice-based layered structure and block-based intra prediction," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1199–1207.

[26] Y. Xu et al., "Predictive generalized graph Fourier transform for attribute compression of dynamic point clouds," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 5, pp. 1968–1982, May 2021.

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

[28] Y. Shen, W. Dai, C. Li, J. Zou, and H. Xiong, "Multi-scale structured dictionary learning for 3-D point cloud attribute compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 7, pp. 2792–2807, Jul. 2021.

[29] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 10850–10869, Sep. 2023.

[30] J. Ho et al., "Cascaded diffusion models for high fidelity image generation," *J. Mach. Learn. Res.*, vol. 23, pp. 2249–2281, Jan. 2022.

[31] S.-I. Cheng, Y.-J. Chen, W.-C. Chiu, H.-Y. Tseng, and H.-Y. Lee, "Adaptively-realistic image generation from stroke and sketch with diffusion model," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, Jul. 2023, pp. 4054–4062.

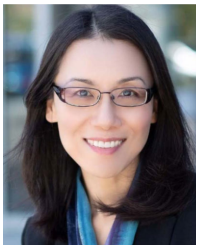
[32] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. van Gool, "RePaint: Inpainting using denoising diffusion probabilistic models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11461–11471.

[33] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, "Image inpainting: A review," *Neural Process. Lett.*, vol. 51, pp. 2007–2028, 2020.
[34] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 417–424.
[35] M. Mainberger, A. Bruhn, J. Weickert, and S. Forchhammer, "Edge-based compression of cartoon-like images with homogeneous diffusion," *Pattern Recognit.*, vol. 44, no. 9, pp. 1859–1873, Sep. 2011.
[36] C. Schmaltz, P. Peter, M. Mainberger, F. Ebel, J. Weickert, and A. Bruhn, "Understanding, optimising, and extending data compression with anisotropic diffusion," *Int. J. Comput. Vis.*, vol. 108, no. 3, pp. 222–240, Jul. 2014.
[37] J. Liang, R. Lai, T. W. Wong, and H. Zhao, "Geometric understanding of point clouds using laplace-beltrami operator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 214–221.
[38] R. Lai, J. Liang, and H.-K. Zhao, "A local MESH method for solving PDEs on point clouds," *Inverse Problems Imag.*, vol. 7, no. 3, pp. 737–755, 2013.
[39] A. Elmoataz, O. Lezoray, and S. Bougleux, "Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1047–1060, Jul. 2008.
[40] F. Lozes, A. Elmoataz, and O. Lézoray, "Partial difference operators on weighted graphs for image processing on surfaces and point clouds," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3896–3909, Sep. 2014.
[41] R. Hooda and W. D. Pan, "Early termination of dyadic region-adaptive hierarchical transform for efficient attribute compression of 3D point clouds," *IEEE Signal Process. Lett.*, vol. 29, pp. 214–218, 2022.
[42] J. Weickert, *Anisotropic Diffusion in Image Processing*, vol. 1. Stuttgart, Germany: B. G. Teubner (Stuttgart), 1998.
[43] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 298–311, Feb. 1997.
[44] *Common Test Conditions for G-PCC*, Standard ISON00650, MPEG 3D Graph. Coding, 2023.
[45] C. Loop, Q. Cai, S. O. Escolano, and P. A. Chou, *8I Voxelized Full Bodies? A Voxelized Point Cloud Dataset*, Standard ISOm38673, 2016.
[46] G. Bjøntegaard, *Calculation of Average PSNR Differences Between RD-curves*, Standard VCEG-M33, Video Coding Experts Group, Austin, Texas, USA, 2001.

Yiting Shao received the B.S. degree from the Minzu University of China in 2016. She is currently pursuing the Ph.D. degree with the School of Electronics Engineering and Computer Science, Peking University. She has been actively contributing to international standards developed by the Moving Picture Experts Group (MPEG) and the Audio Video Coding Standard Working Group of China (AVS) since 2017. She has several technical contributions adopted into various standards, such as MPEG PCC and AVS PCC. Her research interests include image/video/3D point cloud compression.

Xiaodong Yang received the B.S. degree from Dalian University of Technology in 2021. He is currently pursuing the master's degree with Shenzhen Graduate School, Peking University. He was enrolled with the School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University, in 2021. His research interests include image/video/3D point cloud compression.

Wei Gao (Senior Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong in 2017. Since 2019, he has been an Assistant Professor with the School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University, China. His research interests include image and video coding, point cloud compression and processing, visual perception modeling, deep learning, and artificial intelligence.



Shan Liu (Fellow, IEEE) received the B.Eng. degree in electronic engineering from Tsinghua University and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California. She was the Director of the Media Technology Division, MediaTek, USA. She was also with MERL, Sony, and IBM. She is currently a Tencent Distinguished Scientist and the General Manager of the Media Laboratory, Tencent. She has been actively contributing to international standards since the last decade and served as co-editor of HEVC SCC and the emerging VVC. She has numerous technical contributions adopted into various standards, such as HEVC, VVC, OMAF, DASH, and PCC. At the same time, technologies and products developed by her and under her leadership have served several hundred million users. She holds more than 150 granted U.S. and global patents and has published more than 80 journal and conference papers. Her research interests include audio-visual, high-volume, immersive, and emerging media compression, intelligence, transport, and systems. She was appointed as the Vice Chair of the IEEE Data Compression Standards Committee in 2019. She was on the Committee of Industrial Relationship of the IEEE Signal Processing Society from 2014 to 2015 and on the Editorial Board of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2018 to 2021. She was the VP of the Industrial Relations and Development of the Asia-Pacific Signal and Information Processing Association from 2016 to 2017 and was named the APSIPA Industrial Distinguished Leader in 2018.

Ge Li (Member, IEEE) is currently a Professor with the School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University, China. His research interests include image/video process and analysis, machine learning, digital communications, and signal processing.