MM-023671

Scalable Graph-Guided Transformer for Point Cloud Geometry Coding

Mohammadreza Ghafari, *Graduate Student Member*, *IEEE*, André F. R. Guarda, *Member*, *IEEE*, Nuno M. M. Rodrigues, *Senior Member*, *IEEE*, and Fernando Pereira, *Fellow*, *IEEE*

***Abstract*—Attention models, particularly Transformers, have significantly advanced deep learning in fields like natural language processing and computer vision by capturing contextual relationships in both sequential and spatial data. This ability is valuable for Point Clouds (PC), which are unstructured sets of points in 3D space. Transformers can effectively identify correlations between distant points, allowing them to focus on the most critical regions of the data. To demonstrate this capability, this paper proposes a novel, scalable Graph-Guided Transformer model, labeled 2GFormer, for static PC geometry. This model is built using a scalable architecture that leverages Graph Convolutions to enhance a Relational Neighborhood Self-Attention (RNSA) base layer model. Both models are integrated into the JPEG Pleno Learning-based Point Cloud Coding (JPEG PCC) standard, resulting in the creation of two attention-enabled codecs for static PC coding: JPEG RNSA and JPEG 2GFormer. While JPEG RNSA codec delivers significant compression improvements for solid and dense PCs compared to the baseline JPEG PCC standard, JPEG 2GFormer extends these gains to solid, dense, and sparse PCs with only a marginal increase in model parameters. Additionally, JPEG 2GFormer outperforms both conventional and learning-based state-of-the-art PC codecs. These results position JPEG 2GFormer as a highly efficient solution for versatile PC coding.**

***Index Terms*— Scalable Transformer, Graph Convolutions, Self-Attention, JPEG Pleno, Point Cloud Coding**

I. INTRODUCTION

THE rapid proliferation of immersive technologies has spurred an unprecedented demand for high-quality 3D content. Among the various available 3D data representations, Point Clouds (PC) have emerged as a key approach for high-quality, immersive object and scene user experiences. A PC consists of an unordered set of points in 3D space, where each point carries geometric information and may also include additional attributes such as color. Due to their capacity to capture intricate spatial details, PCs play a vital role in emerging applications that are shaping modern multimedia experiences such as Augmented (AR) and Virtual Reality (VR).

Despite these advanced capabilities, PCs pose significant challenges, particularly in storage and transmission since high-quality representations often require millions of points and substantial data volumes, making the development of efficient Point Cloud Coding (PCC) solutions critical for practical applications. While conventional, handcrafted PCC methods, such as Geometry-based PC Compression (G-PCC) and Video-based PC Compression (V-PCC) [1] MPEG standards, have made notable progress, they often struggle to achieve high compression efficiency, especially when handling complex PC geometry.

Deep Learning (DL) has revolutionized multimedia signal processing, and more specifically coding, by enabling end-to-end coding models that surpass conventional coding solutions in compression performance. Building on this success, the JPEG Pleno Learning-based Point Cloud Coding (JPEG PCC) standard [2][3] has specified the first DL-based PCC models which are competitive regarding conventional PCC solutions, notably the MPEG standards. However, despite leveraging DL-based models, the JPEG PCC standard does not incorporate attention-based models, which may allow improving the compression performance through their superior capabilities in capturing long-range dependencies and complex spatial correlations [4]. While the JPEG PCC standard is a competitive PCC solution, its ability to effectively handle diverse PC types, particularly sparse PC, remains limited, highlighting the need for more efficient PCC solutions.

To address these limitations and advance the PCC performance, attention models, notably Transformers, seem to be very promising tools which have already demonstrated remarkable success in Natural Language Processing (NLP) [5] and Computer Vision (CV) [6] tasks. The Transformer's outstanding performance is predominantly driven by its Self-Attention (SA) mechanism, which allows for capturing long-range dependencies and contextual relationships. In the context of PC processing, the SA mechanism is particularly valuable as it enables the processing model to capture correlations between distant points, unlike 3D convolutional kernels, which are limited to local receptive fields. Furthermore, Transformers also incorporate a Feed-Forward Network (FFN) component [5], which is often overshadowed by the SA mechanism. The FFN plays a critical, yet underexplored role in the model's efficiency. A poorly designed FFN not only fails to make positive contributions but can also introduce redundant parameters, which increases the risk of overfitting and degrading the overall performance.

This work was funded by the Fundação para a Ciência e a Tecnologia (FCT, Portugal) through the research project PTDC/EEI-COM/1125/2021, entitled "Deep Learning-based Point Cloud Representation."

Mohammadreza Ghafari is with Instituto Superior Técnico – Universidade de Lisboa, Instituto de Telecomunicações, Lisbon, Portugal (e-mail: mreza.ghafari@lx.it.pt).

André F. R. Guarda is with Instituto de Telecomunicações, Lisbon, Portugal (e-mail: andre.guarda@lx.it.pt).

Nuno M. M. Rodrigues is with ESTG – Politécnico de Leiria, Instituto de Telecomunicações, Lisbon, Portugal (e-mail: nuno.rodrigues@co.it.pt).

Fernando Pereira is with Instituto Superior Técnico – Universidade de Lisboa, Instituto de Telecomunicações (e-mail: fp@lx.it.pt).

To bridge this gap, this paper proposes a novel, scalable Graph-Guided Transformer, labeled 2GFormer, designed to enhance relational point feature extraction in static PC geometry. This scalable model builds upon a foundational Relational Neighborhood Self-Attention (RNSA) model [4], used as base layer model. By leveraging SA mechanisms that prioritize the neighboring points of each query point, the proposed attention models improve feature extraction from intricate PC patterns through an attention map. Furthermore, to improve its efficiency, the attention weights are refined through Graph Convolutions during the Feed Forward process in the Transformer model. Several novel modules have been proposed such as Differential Positional Embedding (DPE), Relational Scoring, Sparsemax, and the Graph-Guided Feed-Forward (GGFF). These modules are not generic adaptations but rather specifically designed for the structural characteristics of PC data, distinguishing them from the existing counterparts, developed for image and language processing. The effectiveness of the novel modules has been rigorously validated through comprehensive ablation studies, demonstrating a clear advantage in terms of compression performance.

The proposed RNSA and 2GFormer attention models are integrated into the JPEG PCC standard to further enhance PC compression performance, leading to the so-called JPEG RNSA and JPEG 2GFormer PCC solutions, respectively. These codecs show very competitive compression performance regarding conventional and learning-based state-of-the-art (SOTA) codecs for different types of PCs, showing the benefit of their scalable design. This scalability only regards their design since there is no coding stream scalability. In fact, this paper extends a paper presented at the IEEE 26th International Workshop on Multimedia Signal Processing (MMSP 2024) [4] which proposed the base layer, RNSA model; the MMSP paper was recognized among the top accepted papers and the authors were invited to submit this extension paper.

The key novelties of the proposed 2GFormer model and associated JPEG 2GFormer codec compared to the previously proposed RNSA model and associated JPEG RNSA codec [4] are:

- 2GFormer model extends the RNSA model by offering a scalable Transformer design that builds upon RNSA as its base layer.
- 2GFormer model includes a novel Graph-Guided Feed Forward within the Transformer model, which incorporates a graph structure to refine the attention weights computed from the neighbors' connectivity.
- 2GFormer model is integrated into the JPEG PCC codec, which utilizes sparse convolutions, enhancing the compression efficiency by constructing the graph structure only for occupied voxels, limiting computational complexity.
- JPEG 2GFormer codec offers a trade-off between compression performance and complexity by using a base layer and an extended model, each offering distinct advantages for different types of PC data.
- JPEG 2GFormer codec enhances compression efficiency regarding JPEG RNSA [4] for different types of PCs with only marginal increase in the number of model parameters.

Compared with the SOTA, the proposed JPEG 2GFormer codec offers better compression performance than conventional and learning-based PC geometry coding solutions for different types of PCs, as detailed in Section VI. This is demonstrated using the JPEG PCC test dataset [7], which includes solid, dense, and sparse PCs, covering a wide range of PC categories. Moreover, as a learning-based codec, JPEG 2GFormer creates latents' streams which can be efficiently used for compressed domain computer vision tasks, e.g., classification [8]. This fact allows offering for the first time a compressed language which may be effective for both man and machine consumption.

The proposed solutions demonstrate the value of integrating appropriate attention models, specifically into the JPEG PCC standard, potentially informing the JPEG Committee on how to proceed with the standardization of a new version of the JPEG PCC standard, notably by incorporating attention models and, more specifically, Transformers. In addition to its practical added value, the proposed coding approach opens new research directions by demonstrating how graph and sparse convolutions can be successfully integrated together, paving the way for further advancements in PC geometry coding.

To achieve its objectives, this paper is structured as follows: Section II provides a brief review of the SOTA in PC coding; moreover, the latest attention models, notably Transformers, developed for PC processing, are also reviewed. Section III provides a brief overview of the JPEG PCC standard used as baseline in this paper. Section IV introduces the novel scalable Graph-Guided Transformer, including both the base layer and full attention models. Section V describes the integration of the novel attention models into the JPEG PCC standard. Section VI presents a detailed analysis of the compression performance and complexity assessment for both models, supported by ablation studies to validate the role of the novel components in each model. Finally, Section VII concludes the paper, highlighting the key findings and potential for future work.

A repository with the relevant decoded PCs from the test dataset and the full experimental results will be made publicly available to enable the research community to use JPEG PCC attention-enabled codecs as benchmarks for new PC geometry codecs.

II. Literature Review

PCC solutions may be categorized into conventional and learning-based methods. The most relevant conventional PCC solutions are the MPEG standards for PC coding, notably the Geometry-based PC Compression (G-PCC) standard for static PCs, which uses an octree to code the 3D PC data, and the Video-based PC Compression (V-PCC) standard for static and dynamic PCs, which projects the 3D data onto 2D images, coded with conventional 2D image/video codecs [1].

Conversely, learning-based coding solutions, particularly voxel-based approaches, utilize Convolutional Neural Networks (CNNs), usually with an autoencoder-based architecture, to successively extract PC features, commonly referred as latents, into a compressed learned-based representation. In this context,

Quach *et al.* [9] improved the PC geometry compression performance using an autoencoder enhanced by a scale hyperprior model to obtain more efficient entropy coding. ADL-PCC [10] uses an adaptive approach to encode the PC geometry, encoding each PC block separately with a different model, adapted to the PC density. IT-DL-PCC [11] adopts a joint geometry and color PCC strategy, integrating down/up-sampling and super-resolution methods to enhance compression performance. PCGCv2 [12] uses sparse convolutions in an autoencoder architecture with an inception residual module, enhanced by a scale hyperprior, while reducing the computational complexity of dense CNN-based solutions, as sparse convolutions operate only on occupied voxels. This was further improved by SparsePCGC [13], which supports both lossy and lossless coding. At the decoder side, during the upscaling process, the decoder considers only the most probable occupied voxels, estimated through cross-scale and same-scale context modeling. GRASP-Net [14] adds an enhancement layer on top of a G-PCC base layer, coding the resulting residue between the original PC and the base layer decoded PC. The enhancement layer consists on a point-based network followed by a sparse convolution network, improving PCC for complex patterns by transforming the disordered and noisy local details of the residue into refined latent features.

Attention models have been widely used in various PC processing tasks and, recently, also for PCC. The SA model computes the correlation between each input element (*query*) and its surrounding elements/neighbors (*key*) to generate attention scores. These scores are then applied to the input features (*value*) to generate the attended output. Point Transformer v1/2 [15][16] leverages SA for PC processing and proves its efficiency for various tasks such as PC segmentation and classification. It captures the relational correlations between the points and enhances the attention process with positional embedding. Point Transformer v3 [17] addresses the complexity of finding k nearest neighbor (kNN) in Point Transformer v2 by introducing space-filling encoding methods, specifically Z-order and Hilbert scanning, along with patching methods to capture the locality of PC data. While this approach can offer a faster method compared to traditional kNN, it cannot achieve the same level of accuracy as the kNN approach in identifying the neighbors. PCGFormer [18] improves PCGCv2 compression performance by adding a local SA to capture correlations among neighboring points, integrating spatial neighbors' features and positional information into the attention process to more effectively learn the spatial relationships between the points. NPAFormer [19], which is similar to PCGFormer while incorporating multi-head attention, demonstrates improved PC compression performance for LiDAR PCs. Later, PCGFormer was employed to enhance the compression efficiency of another similar PCC model [20]. Again, a similar attention model as PCGFormer was used in the Unicorn codec [21], notably stacking multiple attention layers to enhance the overall PC compression performance. This solution was also proposed as a replacement for the Inception ResNet (IRN) used in [13], offering a trade-off between computational complexity and compression efficiency compared to IRN.

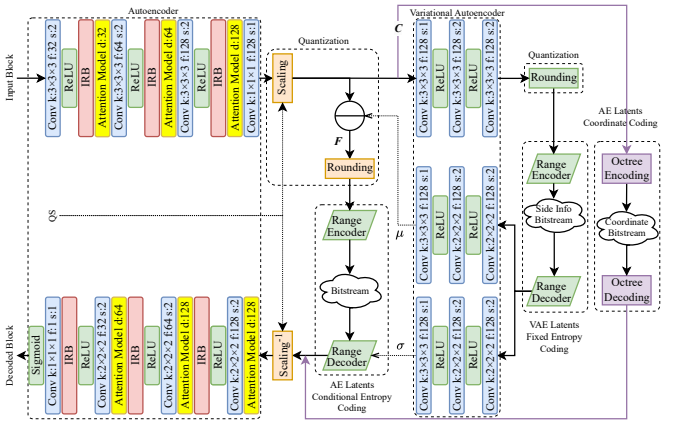


Fig. 1. Attention-enabled JPEG PCC DL-based coding model for PC geometry coding. The yellow layers correspond to the novel attention model layers, each with a feature dimension of d, integrated as described in Section V.

TopNet [22] targets octree-based geometry coding with a Transformer-efficient architecture, introducing four key modules: a locally enhanced context encoder, an adaptive sliding window attention, a spatial-gated channel mixer, and a latent-guided occupancy predictor. These modules collectively enhance feature modeling and occupancy prediction, achieving SOTA PC compression performance. To address the trade-off between perceptual quality and the associated rate, RO-PCAC [23] proposes a rendering-oriented PC attribute coding framework that integrates lossy compression with differentiable rendering to minimize the rate while optimizing the quality. Its core component, SP-Trans, is a sparse tensor-based Transformer that leverages voxel hashing and adaptive local Self-Attention to efficiently model PC density variations by extracting neighbors.

In terms of standardization, JPEG *shook the waters* by developing the first learning-based PCC standard, offering competitive compression performance compared to the previously available coding solutions. The final version of the JPEG PCC serves in this paper as the baseline coding model to integrate the proposed RNSA and 2GFormer attention models. A brief review of JPEG PCC is offered in the next section.

In terms of attention mechanisms, none of the previously designed models offers a scalable design which is able to balance compression performance and computational complexity, what is critical for practical deployment in diverse real-world environments where resource constraints vary and makes the proposed attention models and associated codecs distinctive from the existing ones.

III. JPEG PCC Standard Framework

This section provides a brief overview of the geometry coding model of the DL-based JPEG PCC standard [2][3], illustrated in Fig. 1 (without considering the yellow layers). JPEG PCC adopts a sparse tensor representation, where only occupied voxels are explicitly represented by their coordinates (C) and associated features (F). The reconstructed PC geometry is optionally post-processed using a DL-based super-resolution model. Overall, the process applied to 3D (cubic) blocks may be summarized as follows:

Autoencoder (AE): Transforms the input PC into a latent

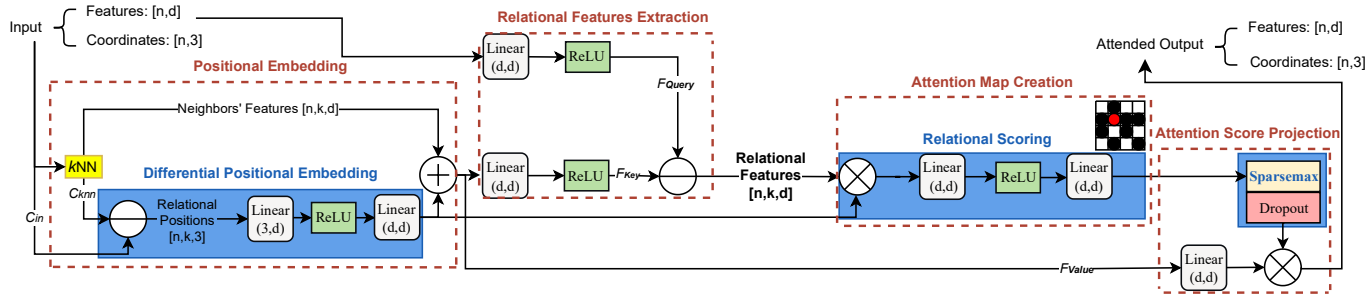


Fig. 2. Relational Neighborhood Self-Attention (RNSA) architecture, highlighting in blue the novel components, notably Differential Positional Embedding (DPE), Relational Scoring and Sparsemax.

representation using a non-linear transform, at the encoder; it comprises four sparse convolution layers intercalated with three Inception Residual Blocks (IRB) [2]. At the decoder, the inverse process is performed, using a symmetric architecture.

Coordinates Coding: The coordinates of the sparse latent representation produced by the AE are losslessly encoded using G-PCC Octree coding mode [1].

Variational Autoencoder (VAE): Captures the structural information in the latent representation, which is used as hyperprior for the latents' entropy coding; it consists of an encoder part with three sparse convolution layers, and two symmetric decoder parts, the first generating a prediction for the latent representation – the mean (μ) – and the second one estimating the standard deviation (σ). The VAE generates its own latent representation, which must also be encoded.

Residue Computation: The prediction μ is subtracted from the AE latent representation, generating a residue which is quantized before conditional entropy coding.

Conditional Entropy Coding: The quantized residual latents are entropy coded with a conditional entropy coding model using zero-mean Gaussian distributions with the standard deviation (σ) estimated by the VAE.

Fixed Entropy Coding: The latent representation generated by the VAE is quantized and entropy coded, using a fixed entropy coding model learned during the training process.

DL-based Super-Resolution Model (optional): Depending on the PC characteristics, down/up-sampling may be applied before/after the coding process. To maximize compression efficiency, a DL-based super-resolution model may be employed following the up-sampling process to densify the PC, effectively enhances its reconstruction quality.

IV. Proposed Novel Scalable Graph-Guided Transformer

This section proposes a novel, scalable Graph-Guided Transformer for PC geometry coding including two attention models; it uses a scalable design, with a base layer consisting on a Self-Attention (SA) and a second layer based on a Transformer architecture built on the base layer. These models can effectively extract local PC geometric relationships, particularly through the neighboring points, despite having a low number of parameters. The first model, called Relational Neighborhood Self-Attention (RNSA), employs an SA-based architecture, while the second model, called Graph-Guided TransFormer (2GFormer), employs a Transformer architecture.

By adopting a scalable design, 2GFormer extends the RNSA model by adding a novel Graph-Guided Feed Forward network, resulting in a powerful and innovative PC attention framework. In fact, the second model is a hierarchical evolution of the first model within a Transformer-based scalable design which incorporates RNSA as its SA module. Each model introduces several novel modules, which are not merely generic adaptations of existing solutions (e.g., from [5] or [6]), but rather carefully designed solutions able to capture the unique structural properties of PC geometry data, naturally leading to better PC feature extraction, effectively addressing its inherent challenges and limitations. Due to this scalable relationship between the base and the full complexity layer, both models (RNSA and 2GFormer) can be used as alternative solutions with conceptually different approaches, notably offering distinct advantages and complexity trade-offs. This scalable model enhances relational point feature extraction and will be integrated into JPEG PCC, as detailed in Section V.

A. Relational Neighborhood Self-Attention Model

This section describes the proposed RNSA model (Fig. 2), which forms the base layer of the scalable 2GFormer model. More information regarding RNSA can be found in [4].

1) Architecture and Walkthrough

The high-level RNSA model adopts an SA architecture and is composed by four main modules, represented in Fig. 2 by the red dashed boxes. The RNSA model input is a sparse tensor representing n points, including their coordinates (x, y, z), and features, with dimension d. The main RNSA modules are:

Positional Embedding: The objective of this module, for each query point, is to embed the information of the features and the corresponding relational positions from the *k*NN regarding the query point. This positional embedding allows the RNSA model to learn complex features and better identify correlations between points.

Relational Features Extraction: The objective of this module is to generate a unique relational feature from points and their neighbors, by measuring the difference between their features. This difference yields more robust features, since each relational feature incorporates the relational information from each query point's features and its neighbors' features.

Attention Map Creation: The objective of this module is to create an attention map based on the relation between each query point and its neighbors, allocating higher attention scores

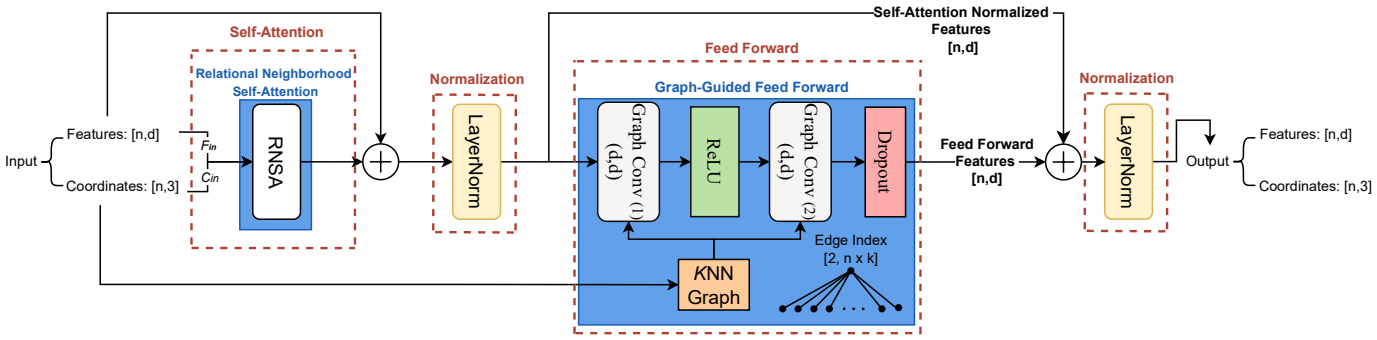


Fig. 3. Graph-Guided TransFormer (2GFormer) architecture, highlighting in blue the novel components, notably RNSA and Graph-Guided Feed Forward.

to the neighboring points with stronger relational features relative to the query points.

Attention Score Projection: The objective of this module is to project the normalized attention scores onto the input point features, generating a new and enhanced representation of the input data which emphasizes the most relevant parts of the input, finally generating the attended PC geometry output, including points with corresponding enhanced features [n, d].

2) Novel RNSA Model Components

Besides the high-level modules, Fig. 2 includes further details for the lower-level design of each RNSA module, which include learnable blocks, represented by the rounded edge squares, as well as other components. The novel components in the proposed RNSA model are represented in Fig. 2 using blue boxes. The technical novelties regard:

Differential Positional Embedding (DPE): The goal of DPE is to incorporate the neighbors' coordinates, (x, y, z) into the RNSA model. Rather than using only the neighbors' coordinates (as in [18]), DPE is adopted [19] to effectively capture the relational positions between each query point and its k_{RNSA} neighboring elements. However, differently from previous related solutions [16][19], DPE is used both in the Positional Embedding and Attention Map Creation modules, with a novel internal DPE architecture. DPE applies two linear transformations: the first maps the relational positions into higher-dimensional space (from 3 to d), while the second refines them, enhancing feature robustness.

Relational Scoring: Inspired by [16], the RNSA model uses a novel Relational Scoring method in the Attention Map Creation module. This method differs from the simple dot product and element-wise product method used in previous works [5][18][19][20][21], generating the attention map using a linear function followed by a ReLU to capture both linear and non-linear relationships between the points in the PC geometry. Introducing non-linearities in the attention process is important for efficient PC processing, since, unlike other domains (e.g., [5]), the relationships between the points cannot be efficiently captured using only linear operations. The novel Relational Scoring method improves attention control over neighboring points by capturing their linear and non-linear relationships in a fixed input PC data.

Sparsemax: The attention scores require normalization to yield probabilities representing the relative importance of neighboring points. Typically, this involves normalizing the attention scores across neighbors through a Softmax [5][16][18], followed by dropout [16] to improve the model generalization capabilities. However, Softmax has an inherent limitation as it produces a probability distribution with full support, meaning that no output probability is exactly zero [24]. To address this limitation, Sparsemax [24] is used, thus zeroing the probabilities of the less influential neighbors in favor of the most important neighbors. To mitigate excessive reliance on attention weights and address model overfitting, a common challenge in SA models, a dropout layer (with a dropout probability, DP_{RNSA}) is used in the Projection component of RNSA. Dropout is applied in training and disabled in inference.

A key advantage of the RNSA model lies in its lower complexity in terms of both number of parameters and computational complexity compared to stacking SA [18][21] and Multi-Head attention mechanisms [19], respectively.

B. *Graph-Guided Transformer*

This section describes the high-level architecture and detailed components of the proposed Graph-Guided TransFormer (2GFormer) model, which builds upon the RNSA base layer model, as represented in Fig. 3.

1) Architecture and Walkthrough

The 2GFormer model architecture consists of three fundamental high-level modules: Self-Attention (SA), Feed Forward (FF), and Normalization, identified in Fig. 3 by red dashed boxes. These modules interact to effectively process and transform the input data, generating enhanced features through the attention mechanism, helping to learn and exploit the relationships between the points in the PC. The 2GFormer architecture features a scalable design regarding RNSA, further enhancing its performance, as detailed in Section VI.

Self-Attention (SA): This module is essential for capturing the relationships between query points and their neighboring elements and assigning attention weights to each neighbor. These attention weights enhance the input PC feature, represented as a sparse tensor including n points with their corresponding coordinates and features. This process transforms the features with dimension d into contextually meaningful representations, allowing the model to identify and focus on the most significant patterns within PC.

Feed Forward (FF): This module refines the SA weights, typically through a Multi-Layer Perceptron (MLP), before passing them to the next layer. This enhances feature flow, preserves spatial-contextual relationships in PC data. The

integration of non-linearities further strengthens the model's ability to capture intricate, non-linear patterns inherent in PC.

Normalization: Normalizing attention outputs is essential for stabilizing the training process in Transformer models. The SA mechanism often produces outputs with varying scales, which can lead to unstable training behaviors, such as exploding or vanishing gradients, limiting proper convergence in SA models. The SA and FF output features consist of n points, each with d features ($[n, \mathrm{d}]$), which are normalized along the feature dimension using LayerNorm [25]. LayerNorm normalizes the input features by computing their mean and standard deviation across each data instance while incorporating scaling and bias as learnable parameters.

2) Novel 2GFormer Model Components

In addition to the high-level modules, Fig. 3 also presents the detailed design of the novel components in the proposed Transformer model. The learnable blocks are depicted as round-edge rectangles, complemented by other operators. The novel 2GFormer model components, including the RNSA module (Fig. 2) and the components of the Graph-Guided Feed Forward module (Fig. 3), are highlighted with blue boxes. Each novel component introduces distinct technical innovations as follows:

RNSA: The previously described RNSA model is integrated into the proposed 2GFormer architecture as a base layer. RNSA captures the relational patterns between query points and their neighboring elements, ensuring that the attention weights accurately express their relational proximity.

Graph-Guided Feed Forward (GGFF): The 2GFormer model proposed in this paper incorporates a novel light-weight but powerful graph-guided approach, notably in terms of memory efficiency, considering the small added number of parameters compared with the noticeable compression performance gains when compared to the base layer model, i.e., RNSA; this model replaces the traditional MLP [5][15][16][17] or the linear transformation in Feed Forward [18][21]. This pioneering approach (to the best of authors' knowledge) integrates graph concepts with sparse convolutions. This not only enhances the model's ability to capture complex neighborhood relationships between PC points, but also reduces redundancies in linear layers, thereby mitigating the risk of overfitting. The novel GGFF component consists of five main steps, as illustrated in Fig. 3, notably:

1. ***k*NN Graph:** Prior to applying a Graph Convolution (Graph Conv), it is necessary to first build a graph. In this graph, the input points act as nodes (n), and the directed edges represent the connections between each point and its nearest neighbors (k_{FF}). To facilitate graph processing, an edge index tensor of shape $[2, n \times k_{FF}]$ is defined to represent the graph structure, notably through relationships between the points. The first dimension (size 2) represents the source to destination path, while the second dimension (size $n \times k_{FF}$) represents the total number of edges in the graph obtained from the points' neighborhoods. The structured graph captures the points' connectivity, enabling efficient propagation of attention weights in the GGFF module. This process enhances point features by refining contextual relationships in the PC.
2. **Graph Conv (1):** Graph Conv [26] offers a more effective solution for point processing compared to 3D convolution, which relies on fixed regular receptive fields defined by kernel size. The first Graph Conv layer propagates attention weights into an intermediate feature space, similar to the hidden dimension in a typical MLP approach but considering the graph structure. By considering the neighbor's connectivity (corresponding to the graph structure), the Graph Conv layer can adapt the receptive field to the local structure of the PC geometry. This feature is especially relevant for complex PCs since Graph Conv can retain important contextual information for relevant far-off neighbors even in irregular PC structure. During Graph Conv, each node forwards attention weights and aggregates feature information from its neighboring nodes, weighted by the edge connections, which capture the importance or strength of each link in the graph.
3. **ReLU:** In the GGFF network, the ReLU activation function, placed between two Graph Conv layers to introduce a non-linearity, enables the network to learn complex and non-linear relationships between the points. The ReLU preserves positive features of the points, highlighting important information while filtering out irrelevant information.
4. **Graph Conv (2):** The second Graph Conv layer further refines the features generated by the first Graph Conv layer and filtered by the ReLU, mapping them to a new representation that better captures the underlying relationships between the points. This layer effectively aggregates additional contextual information, allowing the network to produce more accurate and refined feature representations.
5. **Dropout:** A dropout in FF component of GGFF (DP_{FF}) is applied (only during training) after graph processing, to balance the graph-based forwarded features with the RNSA normalized features. This helps to prevent overfitting and reduces reliance on graph-forwarded features, thus providing a more robust learning process.

The novel 2GFormer model offers a more powerful solution due to its scalable design, notably for forwarding and refining the attention weights, surpassing the typical MLP approach used in [5][15][16][17] for PC coding as shown in Section VI. In fact, the proposed solution is a pioneering attention model which effectively addresses the complexity issues typically associated with such models. Its scalable design enables a choice between a base and a full complexity model, making it suitable when resources vary or are constrained. Beyond scalability, the attention model uniquely integrates graph convolution for capturing local geometric structure and sparse convolution for operating only on occupied voxels, an effective combination that, to the best of authors' knowledge, has not been explored in prior works.

V. Proposed Attention-Enabled JPEG PCC Framework

The novel attention models described in Section IV are designed to enhance the relational feature extraction capabilities for PC geometry and must be integrated into the standard JPEG PCC DL-based coding model (taken as baseline) targeting to improve the PC compression performance. The

integration process for both attention models, RNSA and 2GFormer follows a similar approach, as outlined below.

A. Integrated Architecture

The integration of one or more RNSA or 2GFormer models in the baseline JPEG PCC DL-based coding model (presented in Section III) requires careful consideration of various integration dimensions. These include the number of integrated attention models, named as attention layers, and their placement inside the baseline JPEG PCC coding model, as comprehensively studied in [27]. Fig. 1 illustrates the proposed attention-enabled JPEG PCC coding model. The integration is performed by placing RNSA or 2GFormer models as attention layers in the positions highlighted in yellow in Fig. 1. The new attention-enable JPEG PCC models will hereafter be referred to as JPEG RNSA and JPEG 2GFormer, respectively, naturally depending on the model used for the attention layers.

The attention layers are strategically placed at key stages of the baseline model, to ensure a coherent feature extraction improvement for the final PCC models. Consequently, both the JPEG RNSA and JPEG 2GFormer architectures are designed to maintain symmetry, notably ensuring consistency in improving feature extraction within the JPEG PCC encoder and decoder. The proposed integrated PCC solution is designed with the following key characteristics:

- The RNSA and 2GFormer layers are uniformly distributed across the feature extraction layers of the AE encoder (3 layers) and AE decoder (3 layers).
- The RNSA and 2GFormer layers are placed after (before) each Convolution + ReLU + IRB set in the baseline JPEG PCC encoder (decoder).
- These layers process features with dimensions d, ranging from 32 to 128 (as shown in Fig. 1), corresponding to the JEG PCC number of channels in sparse convolutions.

Following the proposed attention layers integration, the next step is to train the JPEG RNSA and JPEG 2GFormer coding models to achieve optimal PC compression performance.

B. Training

The JPEG RNSA and JPEG 2GFormer coding models are trained using an end-to-end approach, under the same conditions used for the baseline JPEG PCC model. Since these are critical for the final performance assessment, the Pleno PCC Common Training and Test Conditions (CTTC) [7] have been adopted for all coding models.

Different coding models are trained to optimize the PC compression performance at different target rates by minimizing a Rate-Distortion (RD)-driven loss function with a Lagrangian multiplier: $Loss = Distortion + \lambda * Coding\ Rate$ where λ allows to obtain different rate versus distortion trade-offs. The distortion term is computed as the average voxel classification error (empty/occupied), measured by the $Focal\ Loss\ (FL)$ [2][3]. The coding rate during training is estimated as the entropy of the latent representations produced by the AE and VAE, using the computed conditional entropy coding model and the fixed entropy coding model, respectively.

Early stopping is employed with a patience of 25 epochs and a tolerance of 0.1%, meaning that training is halted when the validation loss failed to improve by at least 0.1% of the previous minimum for 25 consecutive epochs. Adam optimizer [28] is used with a learning rate initialized to 10^{-4} and reduced to 10^{-5} using scheduler, if the validation loss does not improve by at least 0.1% for 10 consecutive epochs.

VI. Compression Performance And Complexity Assessment

This section reports the compression performance and complexity assessment of the proposed JPEG RNSA and JPEG 2GFormer codecs, in comparison with relevant benchmarks for geometry-only static PC coding. Due to differences in the dataset, and the use of an updated JPEG PCC version and coding configurations, the JPEG RNSA results may slightly differ from those previously presented in [4].

A. Training Conditions

Training Dataset: The training dataset defined in the JPEG Pleno CTTC [7] has been used. Five DL-based geometry coding models were trained by using five different values of λ, notably 0.0025, 0.005, 0.01, 0.025, and 0.05, in the RD-driven loss function. A sequential training strategy was used: the model for the smallest λ (corresponding to the highest rate) was trained first using random initialization; for subsequent λ values, each model was initialized with the weights corresponding to the previously trained model.

Training conditions: The JPEG RNSA coding model was trained with $k_{RNSA} = 16$ nearest neighbors and a dropout probability of 0.5 for the RNSA Projection component to prevent overfitting and improve generalization. This configuration was used to replicate the previously presented JPEG RNSA model [4]. The JPEG 2GFormer coding model was trained with $k_{RNSA} = 32$ and a dropout probability of 0.5 for the RNSA Projection component. Its graph was constructed using $k_{FF} = 32$ to identify nearest neighbors and apply the Graph Conv operation. A dropout probability of 0.5 was applied to the GGFF component to maintain a good balance between the normalized attention weights and GGFF features. All dropout layers were applied to the model only during training.

B. Test/Inference Conditions

Test dataset: The evaluation was conducted using the CTTC recommended JPEG PCC test dataset (samples shown in Fig. 4), which consists of 12 static PCs categorized into three distinct types depending on their point density: solid, dense, and sparse PCs [7], ensuring a diverse representation of scenes and objects with varying characteristics, including density, precision, and homogeneity, as required by the JPEG Pleno CTTC [7]. For inference, the optimal coding configurations for the JPEG PCC family codecs were selected, notably incorporating super-resolution as defined in [2], for four target rates as defined in [7].

The JPEG PCC test dataset was selected since it includes a wide range of PCs with diverse and complex geometries, thus offering a more comprehensive, meaningful and representative performance assessment than would be possible with datasets

Fig. 4. JPEG PCC test dataset with a diverse set of PCs such as solid (first row), dense (second row) and sparse (third row) PCs with different characteristics.

which focus on a particular type of PC, such as ShapeNet (3D CAD models) [29] or 8iVFB [30] (primarily human body contents).

Benchmarks: To ensure a fair comparison, the optimal coding configurations and recommended settings were used also for the benchmarks, when available. Since the proposed codecs are the same for solid, dense, and sparse PCs, benchmarks which use different models for specific types of PCs were not considered. The selected PC geometry-only coding benchmarks and their respective configurations are:

Conventional Benchmarks

- **G-PCC Octree:** Corresponds to the MPEG G-PCC Octree standard, using the reference software version 24 under the coding configurations provided in [7].
- **V-PCC Intra:** Corresponds to the MPEG V-PCC Intra standard, using the reference software version 23 under the configurations provided in [7].

Learning-based Benchmarks

- **(Baseline) JPEG PCC:** Corresponds to the baseline JPEG PCC standard, using the Verification Model version 4.1 under the configurations described in [2][3].
- **PCGCv2:** Corresponds to the PCGCv2 codec, configured with its default settings as defined in [31] and parameter adjustments based on PC characteristics.
- **GRASP-Net:** Corresponds to the GRASP-Net codec, configured with its default settings and the pretrained models provided in[32]; it should be noted that GRASP-Net uses different models trained separately for each PC category.
- **PCGFormer:** Corresponds to the PCGFormer codec available in [33], which incorporates local SA to enhance PC compression, using the same configurations as for PCGCv2.

Performance Metrics: To evaluate the quality of reconstructed PCs, two popular geometry quality metrics were used: (point-to-point) PSNR D1 and (point-to-plane) PSNR D2. Additionally, Bjontegaard Delta (BD)-Rate and BD-PSNR were employed to compare the RD performances for different codecs against a reference. A negative BD-Rate indicates that rate savings are achieved for a constant quality level, while a positive BD-PSNR expresses the quality improvement (in dB) attainable for a constant rate.

Running Conditions: All experiments were conducted on an Intel Core i9-13900k CPU @ 5.80 GHz, equipped with NVidia GeForce RTX 4090 24GB GPU and 64GB of RAM, running on Debian 12.

C. RD Performance Assessment: JPEG PCC Family

This subsection presents the RD performance for the JPEG

PCC family codecs, notably the attention-enabled JPEG PCC codecs, i.e., JPEG RNSA and JPEG 2GFormer, versus the baseline JPEG PCC codec. Table I presents the BD-Rate and BD-PSNR results for the JPEG PCC family codecs using as reference the baseline JPEG PCC codec. The compression gains achieved by the attention-enabled coding models, i.e., negative BD-Rate and positive BD-PSNR, are highlighted in bold. For clarity, this highlighting is applied only for the three bottom rows in the table, corresponding to the average results for each PC category). From Table I, the following conclusions may be drawn:

- JPEG RNSA provides improvements over the baseline JPEG PCC standard for both solid and dense PCs, but struggles with sparse PCs.
- JPEG 2GFormer further provides significant compression improvements over JPEG RNSA for all categories, notably managing to outperform JPEG PCC for sparse PCs as well.
- The only exception where JPEG 2GFormer fails to outperform JPEG RNSA is for the *HouseWoRoof* PC; this may be attributed to its highly heterogeneous density nature (as detailed in CTTC [7]), thus GGFF of JPEG 2GFormer needs specific optimization.

In summary, JPEG 2GFormer consistently outperforms JPEG RNSA and JPEG PCC standard across all PC categories. These results clearly show the value of the adopted scalable design for the RNSA and 2GFormer attention models. The full model, i.e., JPEG 2GFormer, significantly enhances PC coding capabilities, notably addressing the inefficiency of the base model, i.e., JPEG RNSA, when dealing with sparse PCs.

TABLE I. BD-RATE AND BD-PSNR FOR JPEG PCC FAMILY: ATTENTION-ENABLED JPEG PCC VERSUS JPEG PCC.

Point Cloud		JPEG RNSA				JPEG 2GFormer			
		PSNR D1		*PSNR D2*		*PSNR D1*		*PSNR D2*	
		BD-Rate	***BD-PSNR***	***BD-Rate***	***BD-PSNR***	***BD-Rate***	***BD-PSNR***	***BD-Rate***	***BD-PSNR***
Solid	*StMichael*	-8%	0.35	-6%	0.33	-11%	0.53	-11%	0.58
	Bouquet	-15%	0.46	-11%	0.44	-25%	1.00	-27%	1.20
	Soldier	-6%	0.27	-5%	0.29	-11%	0.47	-11%	0.56
	Thaidancer	-9%	0.38	-10%	0.50	-14%	0.49	-13%	0.61
Dense	*Boxer*	-6%	0.07	-11%	0.36	-17%	0.24	-13%	0.46
	HouseWoRoof	-16%	0.34	-2%	0.04	-6%	0.09	9%	-0.20
	CITIUSP	-0.05%	-0.01	-1%	0.03	-15%	0.46	-31%	1.75
	Facade9	-16%	0.47	-0.4%	0.01	-22%	0.61	-5%	0.15
Sparse	*EPFL*	8%	-0.20	13%	-0.45	-10%	0.30	-10%	0.56
	ArcoValentino	-3%	0.08	-4%	0.11	-7%	0.30	-12%	0.48
	Shiva	-1%	0.03	-3%	0.11	-1%	0.03	-2%	0.01
	Unicorn	18%	-1.15	-2%	0.11	-2%	0.01	-24%	1.05
Average Solid		**-9%**	**0.36**	**-8%**	**0.39**	**-15%**	**0.62**	**-15%**	**0.74**
Average Dense		**-10%**	**0.22**	**-3%**	**0.11**	**-15%**	**0.35**	**-10%**	**0.54**
Average Sparse		6%	-0.31	1%	-0.03	**-5%**	**0.16**	**-12%**	**0.52**

D. RD Performance Assessment: Other Benchmarks

Since JPEG 2GFormer is the best performing codec in the JPEG PCC family, it is used for comparison with the selected benchmarks. From Fig. 5 and Table II, it is possible to observe that the JPEG family of PCC solutions has a clear overall advantage over the conventional codecs (G-PCC and V-PCC) but also over the DL-based codecs. The BD-Rate values of "-100%" correspond to cases where there is no intersection in quality between the RD curves under comparison what prevents the reliable computation of BD-Rate values, as JPEG 2GFormer curve lies entirely above than the others; for this reason, these values were excluded from the average rows per PC category (what penalizes the proposed solutions). Table II provides a finer analysis of the RD performance, allowing to draw the following main conclusions:

- **Solid PCs:** JPEG 2GFormer outperforms all the conventional and DL-based benchmarks with rate savings ranging from 21% (8%) compared to PCGFormer, up to 89% (84%) compared to G-PCC, for PSNR D1 (PSNR D2). These gains are more substantial over conventional codecs, though comparatively smaller against learning-based codecs, particularly PCGFormer, which also employs attention models.
- **Dense PCs:** JPEG 2GFormer outperforms all the conventional and DL-based benchmarks, with only one exception related to V-PCC, with rate savings ranging from 11% (23%) compared to GRASP-Net, up to 75% (48%) compared to PCGFormer, for PSNR D1 (PSNR D2). The only exception is *Facade9* PC (as shown in Fig. 4) with V-PCC, where JPEG 2GFormer loses significantly. This behavior is attributed to *Facade9*'s better alignment with V-PCC Intra's projection-based approach. However, despite the large BD-Rate difference, the perceptual assessment for the two decoded PCs is not significantly different.
- **Sparse PCs:** JPEG 2GFormer outperforms once more all the conventional and DL-based benchmarks, with one single (but different) exception. On average, the RD performance gains range from 12% (33%) compared to G-PCC, up to 60% (51%) compared to PCGCv2, for PSNR D1 (PSRN D2). The only exception to this trend comes from GRAPS-Net, where JPEG 2GFormer loses 20% in terms of BD-Rate PSNR D1. In this comparison, JPEG 2GFormer marginally outperforms GRASP-Net for *ArcoValentino* and *Shiva*, but underperforms on *EPFL* and *Unicorn*, resulting in average rate increases of 20% for PSNR D1; however, it achieves 16% average rate savings for PSNR D2. It is important to notice that GRAP-Net uses three different models, optimized independently for solid, dense and sparse PCs; however, JPEG 2GFormer utilizes a single unified model across all PC categories. As a result, GRASP-Net is naturally better suited for handling sparse point clouds due to its specifically trained sparse models.

In summary, JPEG 2GFormer achieves superior average RD performance for solid, dense and sparse PCs, significantly outperforming all conventional and learning-based benchmarks using a single unified model. Moreover, JPEG 2GFormer achieves larger compression gains (for solid PCs) over PCGCv2 (without attention) than PCGFormer (with attention), reinforcing the effectiveness of attention mechanisms in learning-based codecs.

E. Complexity Assessment

Table III compares JPEG 2GFormer and JPEG RNSA with JPEG PCC standard in terms of the increase in the model's number of parameters and the encoding and decoding time variations (in percentage), averaged across the four target rates for each PC. It is worth noting that while floating-point operations per second (FLOPs) is a common metric for measuring model complexity, it is not possible to use for the

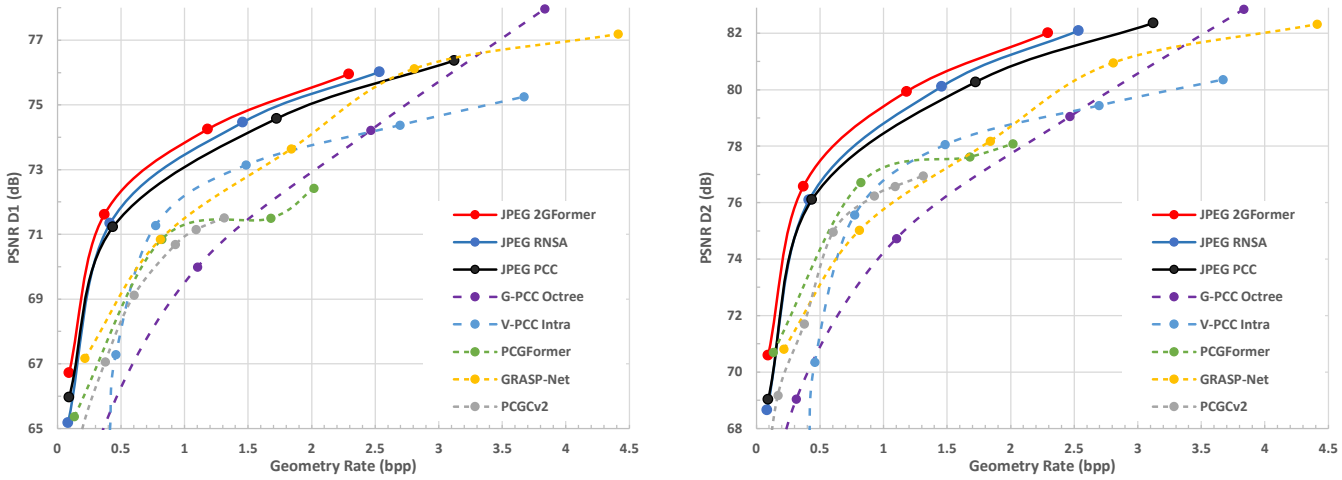


Fig. 5. Average RD performance across the full JPEG PCC test dataset for SOTA PC geometry codecs. JPEG 2GFormer (in red) demonstrates superior compression performance compared to existing SOTA codecs.

TABLE II. BD-RATE AND BD-PSNR FOR JPEG 2GFORMER AGAINST BENCMARK CODECS. LEFT SIDE REGARDS CONVENTIONAL PC CODECS AND RIGHT SIDE REGARDS LEARNING-BASED PC CODECS.

	PC Name	Ref: G-PCC Octree [1]				Ref: V-PCC Intra [1]				Ref: PCGCv2 [12]				Ref: GRASP-Net [14]				Ref: PCGFormer [18]			
		PSNR D1		*PSNR D2*		*PSNR D1*		*PSNR D2*		*PSNR D1*		*PSNR D2*		*PSNR D1*		*PSNR D2*		*PSNR D1*		*PSNR D2*	
		BD-Rate	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*	*BD-Rate*	*BD-PSNR*
Solid	*StMichael*	-87%	8.97	-79%	7.49	-64%	4.33	-62%	4.92	-25%	1.16	-23%	1.23	-32%	1.70	-36%	2.26	-3%	0.20	-8%	0.50
	Bouquet	-88%	8.24	-85%	7.29	-69%	4.63	-71%	5.29	-8%	0.12	-20%	0.49	-34%	1.30	-45%	1.96	5%	-0.19	-6%	0.06
	Soldier	-89%	10.78	-84%	9.73	-49%	3.53	-49%	4.19	-24%	1.28	-24%	1.46	-52%	3.00	-53%	3.72	-10%	0.48	-11%	0.65
	Thaidancer	-92%	10.39	-86%	9.62	-39%	2.17	-34%	2.43	-66%	4.03	-30%	1.62	-66%	3.50	-64%	4.44	-75%	3.99	-8%	0.51
Dense	*Boxer*	-73%	5.48	-89%	8.75	-41%	0.95	-39%	2.01	-100%	3.43	-100%	5.00	-48%	0.73	-51%	1.98	-100%	3.40	-65%	3.88
	HouseWoRoof	-66%	3.68	-62%	3.87	-29%	0.61	-56%	2.34	-58%	1.73	-63%	1.86	-14%	0.20	-29%	0.83	-77%	2.43	-57%	1.63
	CITIUSP	-44%	2.18	-45%	2.71	-4%	0.004	-20%	0.32	-72%	3.78	-44%	1.90	27%	-0.80	24%	-0.82	-70%	4.22	-27%	1.35
	Facade9	-49%	2.62	-61%	3.92	428%	-5.19	91%	-3.22	-62%	2.35	-49%	2.03	-8%	0.21	-37%	1.31	-77%	2.85	-42%	1.86
Sparse	*EPFL*	-19%	0.72	-24%	1.03	-30%	0.89	-25%	0.92	-58%	1.34	-39%	1.79	1%	-0.12	-25%	0.97	-40%	1.57	-60%	2.49
	ArcoValentino	-22%	1.37	-19%	0.94	-68%	3.42	-77%	4.69	-65%	2.85	-58%	2.81	-4%	-0.05	-9%	0.18	-63%	3.05	-57%	2.97
	Shiva	-48%	2.38	-36%	1.74	-72%	2.75	-87%	4.92	-22%	0.64	-15%	0.72	-13%	0.33	-17%	0.56	-20%	0.60	27%	-0.41
	Unicorn	39%	-1.25	-53%	3.01	19%	-0.55	-53%	-0.49	-94%	18.95	-92%	17.87	97%	-1.32	-15%	0.98	-100%	6.13	-88%	5.79
	Average Solid	**-89%**	**9.60**	**-84%**	**8.53**	**-55%**	**3.67**	**-54%**	**4.21**	**-31%**	**1.65**	**-24%**	**1.20**	**-46%**	**2.38**	**-49%**	**3.10**	**-21%**	**1.12**	**-8%**	**0.43**
	Average Dense	**-58%**	**3.49**	**-64%**	**4.81**	88%	-0.91	6%	**0.36**	**-64%**	**2.82**	**-52%**	**2.70**	**-11%**	**0.08**	**-23%**	**0.83**	**-75%**	**3.23**	**-48%**	**2.18**
	Average Sparse	**-12%**	**0.80**	**-33%**	**1.68**	**-38%**	**1.63**	**-61%**	**2.51**	**-60%**	**5.95**	**-51%**	**5.80**	20%	-0.29	**-16%**	**0.67**	**-41%**	**2.84**	**-45%**	**2.71**

JPEG family codecs, as they utilize sparse convolutions implemented with Minkowski Engine v0.5.4 [34][35] which does not support FLOPs computation. The results allow deriving the following conclusions:

Number of Parameters

- Both JPEG RNSA and JPEG 2GFormer are highly parameter-efficient compared to JPEG PCC, requiring only 6.99% and 10.56% additional parameters, respectively, to provide the presented compression gains.
- While JPEG 2GFormer outperforms JPEG RNSA in compression performance, it introduces only a 3% (approximately) increase in parameter overhead, demonstrating the efficiency of its Graph Conv operations.

Coding Times

- Due to the added attention layers in the baseline JPEG PCC, JPEG RNSA and JPEG 2GFormer increase the coding times for all types of PCs.
- This increase is more noticeable for JPEG 2GFormer, as it builds upon RNSA as its base layer. While JPEG RNSA itself adds computational overhead, JPEG 2GFormer introduces additional complexity, thus leading to longer processing times. However, this added complexity can be mitigated through hyperparameter tuning, particularly with graph construction tailored to each PC category.
- This added complexity mostly comes from graph processing, which is inherently time-consuming. Unlike JPEG RNSA, which primarily relies on *k*NN, JPEG 2GFormer requires both finding the neighbors and graph construction, a process that is more computationally intensive than using *k*NN alone.
- The implemented Graph Conv [26] uses an optimized implementation, offering significantly lower complexity than generic graph convolutions [36] while maintaining an efficient graph processing. Further reductions in computational complexity can be achieved by tuning the graph structure, such as adjusting the number of nodes based

TABLE III. MODEL PARAMETERS, ENCODING AND DECODING TIMES FOR JPEG PCC FAMILY: ATTENTION-ENABLED JPEG PCC VERSUS JPEG PCC.

Point Cloud	JPEG PCC		JPEG RNSA		JPEG 2GFormer	
	EncT	*DecT*	*EncT*	*DecT*	*EncT*	*DecT*
Average Solid	80.4s	5.9s	+7.7%	+42.3%	+27.7%	+206.5%
Average Dense	220s	52.0s	+14.0%	+38.1%	+95.5%	+284.2%
Average Sparse	159s	63.8s	+23.0%	+54.6%	+199.2%	+469.7%
# Model Parameters	5081121		+6.99%		+10.56%	

on the PC category.

- A key benefit of the proposed scalable design is the option to use the base layer model when computational resources are limited, or faster encoding and decoding are required.

F. Error Map Visualization

In order to evaluate the visual impact of the enhanced JPEG PCC family of codecs, particularly those incorporating attention models, Fig. 6 presents point-to-point distance errors between the compressed and reference PC, visualized using a colormap approach similar to that in [21]. This error has been computed for JPEG PCC, JPEG RNSA, and JPEG 2GFormer for solid, dense, and sparse PCs, illustrated with one example for each category. It can be observed that JPEG RNSA struggles to achieve better visual quality compared to JPEG PCC, especially for solid and dense PCs, at low rates, as shown by the black boxes. However, JPEG 2GFormer substantially reduces these errors, indicated by fewer red regions and more blue areas, while maintaining a comparable bitrate to JPEG RNSA. These improvements are especially pronounced in geometrically complex areas, such as edges, highlighted in the black boxes. In these challenging regions, the GGFF module in JPEG 2GFormer effectively refines the attention weights initially provided by the RNSA model, leading to improved reconstructions.

G. Ablation Studies

The ablation studies assess the novel components of both JPEG RNSA and JPEG 2GFormer, analyzing their impact on the overall RD performance. Since JPEG RNSA was previously proposed for solid PCs [4], its ablation results are specifically presented for solid PCs. However, since JPEG 2GFormer is this paper's proposed solution for solid, dense and sparse PCs, its ablation results will consider all types of PCs.

Table IV shows the performance impact of removing each novel JPEG RNSA attention model component. This evaluation considers RD performance (measured using BD values with the full JPEG RNSA as reference) and complexity (measured by the variation in percentage of the total number of model parameters, encoding and decoding times). From Table IV, the following observations may be derived:

- **RNSA without (w/o) DPE:** Removing the novel DPE component from RNSA leads to an RD performance reduction, notably BD-Rate increases by 1.4% (1.6%) for PSNR D1 (PSNR D2). This highlights the role of relational positional encoding in the attention model, emphasizing the importance of generating attention maps with an awareness of neighbors' spatial relationships.
- **RNSA w/o Relational Scoring:** Replacing the novel Relational Scoring component with a dot product operation results in a significant reduction in RD performance, notably BD-Rate increases by 5.9% (3.7%) for PSNR D1 (PSNR D2). This underscores the effectiveness of Relational Scoring in PCC to capture meaningful relationships through learned linear and non-linear transformations.
- **RNSA w/o Sparsemax:** Replacing Sparsemax with Softmax results in an RD performance drop, notably BD-Rate increases by 1.4% (1.1%) for PSNR D1 (PSNR D2). Sparsemax's gain is limited by the normalization across 16 neighbors, with larger neighbor sets or dimensions providing more noticeable gains [24].

Naturally, removing the novel components reduces both model parameters and coding time. While the number of parameters and encoding time remain relatively low, the impact is more pronounced on decoding time, resulting in a decoding time reduction from 5% to 24.3%.

Table V shows the impact of removing and replacing the novel GGFF component of JPEG 2GFormer compared with JPEG 2GFormer as reference. From Table V, the following observations may be derived:

- **2GFormer w/o GGFF:** Removing the novel GGFF component from JPEG 2GFormer downgrades it to its base layer, notably RNSA, with $k_{RNSA} = 32$. This leads to an RD performance reduction, notably BD-Rate increases of 1.2% (4.2%) for PSNR D1 (PSNR D2). These results highlight the beneficial impact of graph processing in enhancing the RNSA base layer.
- **2GFormer with PostLN-MLP**: Replacing GGFF with the MLP-based solution from vanilla Transformer [5] and applying LayerNorm after [5], results in BD-Rate increases of 4.7% (6.3%) for PSNR D1 (PSNR D2). This

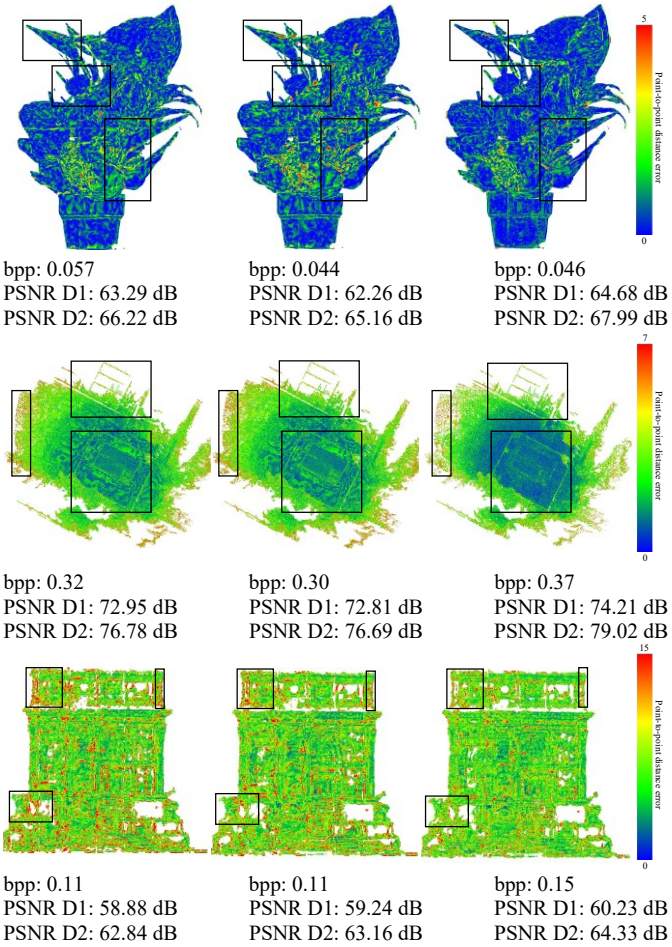


Fig. 6. Point-to-point distance errors for JPEG PCC, JPEG RNSA, and JPEG 2GFormer reconstructed PCs (left to right) using the color code on the right for solid, dense, and sparse PCs (*Bouquet*, *CITIUSP* and *ArcoValentino* from top to bottom).

TABLE IV. JPEG RNSA ABLATION STUDIES.

Solution: JPEG RNSA w/o	Reference: JPEG RNSA (#Parameters: 5436769)						
	PSNR D1		PSNR D2		# Params.	Inference Time (s)	
	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR		EncT	DecT
w/o DPE	1.4%	-0.03	1.6%	-0.06	-1.1%	-2.3%	-5.0%
w/o Rel. Scoring	5.9%	-0.19	3.7%	-0.17	-2.2%	-5.5%	-24.3%
w/o Sparsemax	1.4%	-0.04	1.1%	-0.05	0	-7.6%	-19.5%

TABLE V. JPEG 2GFORMER ABLATION STUDIES.

Solution: JPEG 2GFormer w/o or with	Reference: JPEG 2GFormer (#Parameters: 5617857)						
	PSNR D1		PSNR D2		# Params.	Inference Time	
	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR		EncT	DecT
w/o GGFF	1.2%	-0.03	4.2%	-0.17	-3.2%	-26.1%	-42.6%
with PostLN-MLP	4.7%	-0.13	6.3%	-0.22	-1.0%	-24.1%	-38.4%
with PreLN-MLP	5.3%	-0.16	8.7%	-0.35	-1.0%	-24.6%	-37.7%

demonstrates the effectiveness of the RNSA base layer in leveraging linear and non-linear transformations; introducing additional MLP layers with similar functionality leads to redundancy, excessive parameterization, and overall, performance degradation.

- **2GFormer with PreLN-MLP:** Replacing GGFF with the MLP-based solution similar to Point Transformer v2 [16] and applying LayerNorm before, similarly to Point Transformer v3 [17], results in BD-Rate increases of 5.3% (8.7%) for PSNR D1 (PSNR D2). While this approach has been used with benefit for PC segmentation, it was not beneficial for PC coding. The comparison between PostLN and PreLN suggests that applying LayerNorm after MLP is more advantageous, as it effectively normalizes the Transformer's outputs, leading to more stable and efficient representation learning.

Removing and replacing the novel GGFF component reduces both the model parameters as well as encoding and decoding times. However, the parameter reduction is relatively minor compared to the RNSA base layer (-3.2%) and MLP-based solutions (-1.0%), demonstrating that GGFF is highly memory-efficient. This complexity reduction becomes more pronounced when encoding and decoding times are considered; however, the added complexity is less noticeable during encoding than decoding.

VII. CONCLUSION AND FUTURE WORKS

This paper proposes a novel, scalable Graph-Guided Transformer (2GFormer) model, built on top of a Self-Attention model, used as base layer. Both models were integrated into the JPEG PCC standard coding model to enhance compression performance. The 2GFormer model incorporates several innovative components, each contributing to improved PC geometry compression, as validated by ablation studies. Experimental results demonstrate that JPEG RNSA achieves significant compression gains over JPEG PCC for solid and dense PCs, while JPEG 2GFormer offers more compression gains for all types of PCs, surpassing both conventional and learning-based SOTA codecs. These findings highlight the effectiveness of graph-based architecture coupled with sparse convolutions, offering a promising direction for future research in efficient PC coding. Future work includes optimizing graph construction using an adaptive nearest neighbors based on PC category.

REFERENCES

[1] D. Graziosi et al., "An Overview of Ongoing Point Cloud Compression Standardization Activities: Video-Based (V-PCC) and Geometry-Based (G-PCC)," APSIPA Trans. Signal Inf. Process., vol. 9, Mar. 2020.

[2] ISO/IEC FDIS 21794-6:2025, "Information technology—Plenoptic image coding system (JPEG Pleno)—Part 6: Learning-based point cloud coding," Jan. 2025.

[3] A. F. R. Guarda, N. M. M. Rodrigues and F. Pereira, "The JPEG Pleno Learning-based Point Cloud Coding Standard: Serving Man and Machine," IEEE Access, Mar. 2025.

[4] M. Ghafari, A. F. R. Guarda, N. M. M. Rodrigues and F. Pereira, "Point Cloud Geometry Coding with Relational Neighborhood Self-Attention," IEEE Int. Workshop on Multimedia Signal Process., West Lafayette, IN, USA, Oct. 2024.

[5] A. Vaswani et al., "Attention is All you Need," Advances Neural Inf. Process. Sys., Long Beach, CA, USA, Dec. 2017.

[6] Z. Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," Proceedings IEEE/CVF Int. Conf. Comput. Vis., Montreal, QC, Canada, Oct. 2021.

[7] ISO/IEC JTC 1/SC 29/WG1 N100909, "JPEG Pleno Point Cloud Coding Common Training and Test Conditions v2.2," 104th Meeting, Japan, July 2024.

[8] A. Seleem, A. F. R. Guarda, N. M. M. Rodrigues and F. Pereira, "Deep Learning-Based Compressed Domain Multimedia for Man and Machine: A Taxonomy and Application to Point Cloud Classification," IEEE Access, vol. 11, pp. 128979-128997, Nov. 2023.

[9] M. Quach, G. Valenzise and F. Dufaux, "Improved Deep Point Cloud Geometry Compression," IEEE Int. Workshop Multimedia Signal Process., Tampere, Finland, Sep. 2020.

[10] A. F. R. Guarda, N. M. M. Rodrigues and F. Pereira, "Adaptive Deep Learning-Based Point Cloud Geometry Coding," IEEE J. Sel. Topics Signal Process., vol. 15, no. 2, pp. 415-430, Feb. 2021.

[11] A. F. R. Guarda et al., "Deep Learning-Based Point Cloud Coding and Super-Resolution: a Joint Geometry and Color Approach," IEEE Trans. Multimedia., Nov. 2023.

[12] J. Wang, D. Ding, Z. Li and Z. Ma, "Multiscale Point Cloud Geometry Compression," Data Compression Conf., Snowbird, UT, USA, Mar. 2021.

[13] J. Wang et al., "Sparse Tensor-Based Multiscale Representation for Point Cloud Geometry Compression," IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 7, July 2023.

[14] J. Pang, M. Lodhi, D. Tian, "GRASP-Net: Geometric Residual Analysis and Synthesis for Point Cloud Compression," Int. Workshop Adv. Point Cloud Comp., Lisboa, Portugal, Oct. 2022.

[15] H. Zhao, et al., "Point Transformer," Proceedings. IEEE/CVF Int. Conf. Comput. Vis., BC, Canada, Oct. 2021.

[16] X. Wu et al., "Point Transformer V2: Grouped Vector Attention and Partition-based Pooling," Advances Neural Inf. Process. Sys., New Orleans, MS, USA, Nov. 2022.

[17] X. Wu et al., "Point Transformer V3: Simpler, Faster, Stronger," Conf. on Compt. Vision and Pattern Recognit., Seattle, Washington, USA, June 2024.

[18] G. Liu, J. Wang, D. Ding and Z. Ma, "PCGFormer: Lossy Point Cloud Geometry Compression via Local Self-Attention," IEEE Int. Conf. Vis. Com. and Image Process., Suzhou, China, Dec. 2022.

[19] R. Xue, J. Wang, Z. Ma, "Efficient LiDAR Point Cloud Geometry Compression Through Neighborhood Point Attention," arXiv:2208.12573 [cs.CV], Aug. 2022.

[20] J. Zhang et al., "DeepPCC: Learned Lossy Point Cloud Compression," IEEE Trans. Emerging Topics Comput. Intell., Oct. 2024.

[21] J. Wang et al. "A Versatile Point Cloud Compressor Using Universal Multiscale Conditional Coding – Part I: Geometry," IEEE Trans. Pattern Anal. Mach. Intell., vol. 47, no. 1, pp. 269-287, Jan. 2025.

[22] X. Wang, et al., "TopNet: Transformer-Efficient Occupancy Prediction Network for Octree-Structured Point Cloud Geometry Compression," IEEE/CVF Comput. Vis. and Pattern Recognit. Conf., Nashville TN, USA, June 2025.

[23] X. Huo, J. Hou, S. Wan and F. Yang, "Rendering-Oriented 3D Point Cloud Attribute Compression using Sparse Tensor-based Transformer,"

IEEE Trans. Circuits Syst. Video Technol., Feb. 2025, Early Access, doi: 10.1109/TCSVT.2025.3546626.

[24] A. F. T. Martins, R. F. Astudillo, "From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification," Int. Conf. Mach. Learn., New York, NY, USA, June 2016.

[25] J. L. Ba, J. R. Kiros, G. E. Hinton, "Layer Normalization," arXiv:1607.06450v1 [stat.ML], July 2016.

[26] F. Wu et al., "Simplifying Graph Convolutional Networks," Int. Conf. Machine Learning, California, USA, June 2019.

[27] M. Ghafari, A. F. R. Guarda, N. M. M. Rodrigues and F. Pereira, "Deep Learning-based Point Cloud Geometry Coding with Attention Models," IEEE Int. Symp. Multimedia, Laguna Hills, CA, USA, Dec. 2023.

[28] D. P. Kingma and J. Ba, "Adam: a Method for Stochastic Optimization," Int. Conf. Learn. Representations, San Diego, CA, USA, May 2015.

[29] https://shapenet.org/

[30] https://plenodb.jpeg.org/pc/8ilabs

[31] https://github.com/NJUVISION/PCGCv2

[32] https://github.com/InterDigitalInc/GRASP-Net

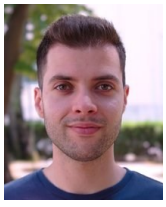
[33] https://github.com/3dpcc/PCGFormer

[34] M. Ghafari, A. F. R. Guarda, N. M. M. Rodrigues, F. Pereira, "Learning-Based Point Cloud Decoding with Independent and Scalable Reduced Complexity," IEEE Int. Conf. Image Process., Abu Dhabi, UAE, Oct. 2024.

[35] C. Choy, J. Gwak and S. Savarese, "4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks," IEEE Conf. Comput. Vision and Pattern Recognit., Long Beach, CA, USA, June 2019.

[36] T. N. Kipf, M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," Int. Conf. Learn. Represent., Toulon, France, Apr. 2017.

MOHAMMADREZA GHAFARI (Graduate Student Member, IEEE) received the B.Sc. degree in Electrical Engineering (Telecommunications) from IRIB University, Iran, in 2018, graduating as the top student of the year. He obtained the M.Sc. degree in Electrical Engineering – Telecommunication Systems from Amirkabir University of Technology, Iran, in 2021. He has been a researcher at Instituto de Telecomunicações, Portugal, contributing to the learning-based JPEG Pleno Point Cloud Coding standardization process. He is also actively involved in enhancing the quality of service (QoS) and quality of experience (QoE) for low-latency, high-bitrate applications such as cloud gaming. His research interests include multimedia signal processing and computer networking, with a focus on networked gaming applications.



ANDRÉ F. R. GUARDA (Member, IEEE) received his B.Sc. and M.Sc. degrees in electrotechnical engineering from Instituto Politécnico de Leiria, Portugal, in 2013 and 2016, respectively, and the Ph.D. degree in electrical and computer engineering from Instituto Superior Técnico, Universidade de Lisboa, Portugal, in 2021. He has been a researcher at Instituto de Telecomunicações since 2011, where he currently holds a Post-Doctoral position. His main research interests include multimedia signal processing and coding, with particular focus on point cloud coding with deep learning. He has authored several publications in top conferences and journals in this field, and is actively contributing to the standardization efforts of JPEG and MPEG on learning-based point cloud coding.

NUNO M. M. RODRIGUES (Senior Member, IEEE) graduated in electrical engineering in 1997, received the M.Sc. degree from the Universidade de Coimbra, Portugal, in 2000, and the Ph.D. degree from the Universidade de Coimbra, Portugal, in 2009, in collaboration with the Universidade Federal do Rio de Janeiro, Brazil. He is a Professor in the Department of Electrical Engineering, in the School of Technology and Management of the Polytechnic University of Leiria, Portugal and a Senior Researcher in Instituto de Telecomunicações, Portugal. He has coordinated and participated as a researcher in various national and international funded projects. He has supervised three concluded PhD theses and several MSc theses. He is co-author of a book and more than 100 publications, including book chapters and papers in national and international journals and conferences. His research interests include several topics related with image and video coding and processing, for different signal modalities and applications. His current research is focused on deep learning-based techniques for point cloud coding and processing.

FERNANDO PEREIRA (Fellow, IEEE) graduated in electrical and computer engineering in 1985 and received the M.Sc. and Ph.D. degrees in 1988 and 1991, respectively, from Instituto Superior Técnico, Technical University of Lisbon. He is with the Department of Electrical and Computers Engineering of Instituto Superior Técnico, University of Lisbon, and Instituto de Telecomunicações, Lisbon, Portugal. He is or has been Associate Editor of IEEE Transactions of Circuits and Systems for Video Technology, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Signal Processing Magazine and EURASIP Journal on Image and Video Processing, and Area Editor of the Signal Processing: Image Communication Journal. In 2013-2015, he was the Editor-in-Chief of the IEEE Journal of Selected Topics in Signal Processing.

He was an IEEE Distinguished Lecturer in 2005 and elected as an IEEE Fellow in 2008 for "contributions to object-based digital video representation technologies and standards". He has been elected to serve on the IEEE Signal Processing Society Board of Governors in the capacity of Member-at-Large for 2012 and 2014-2016 terms. He has been IEEE Signal Processing Society Vice-President for Conferences in 2018-2020 and IEEE Signal Processing Society Awards Board Member in 2017. He was the recipient of the 2023 Leo L. Beranek Meritorious Service Award.

Since 2013, he is also a EURASIP Fellow for "contributions to digital video representation technologies and standards". He has been elected to serve on the European Signal Processing Society Board of Directors for a 2015-2018 term. He was the recipient of the 2023 EURASIP Meritorious Service Award. Since 2015, he is also an IET Fellow.

He has also held key leadership roles in numerous IEEE Signal Processing Society conferences and workshops, mostly notably serving twice as ICIP Technical Chair in two continents, Hong Kong (2010) and Phoenix (2016).

He has been MPEG Requirements Subgroup Chair and is currently JPEG Requirements Subgroup Chair. Recently, he has been one of the key designers of the JPEG Pleno and JPEG AI standardization projects. He has contributed more than 300 papers in international journals, conferences and workshops, and made several tens of invited talks and tutorials at conferences and workshops. His areas of interest are video analysis, representation, coding, description and adaptation, and advanced multimedia services.