LearningPCC: A PyTorch Library for Learning-Based Point Cloud Compression

Liang Xie
liangxie@stu.pku.edu.cn
Peking University Shenzhen Graduate School and Peng Cheng Laboratory, Shenzhen, China

Wei Gao*
gaowei262@pku.edu.cn
Peking University Shenzhen Graduate School and Peng Cheng Laboratory, Shenzhen, China

Abstract

Three-dimensional point cloud data is one of the most extensively used data representation today, favored in various fields for its realistic and lifelike visual effects. However, the substantial volume of data poses significant challenges for storage and transmission. To advance point cloud compression (PCC) technology, we develop a learning-based PCC algorithm library, namely LearningPCC. To our knowledge, this is the first comprehensive set of algorithms that is compatible with all types of point cloud data. This PyTorch library incorporates eleven learning-based algorithms that address both geometry and attribute compression of point cloud data. We categorize the existing methods into six main classes and thoroughly introduce and analyze the principles of these algorithms. Moreover, we conduct performance evaluations using point clouds with various densities, offering detailed test results on several compression metrics, such as RD curves, BD-BR gains, compression ratio improvements, and encoding times. We will provide researchers with convenient access to these methods, replicate codes, and experiment results. Our commitment includes maintaining and updating these algorithms to offer researchers the latest in compression technologies. The related algorithm codes, testing processes, and results will be available at https://openi.pcl.ac.cn/OpenPointCloud/LearningPCC.

CCS Concepts

• **Theory of computation** → **Data compression**.

Keywords

Point Cloud Compression, Open-source Software.

ACM Reference Format:

Liang Xie and Wei Gao. 2024. LearningPCC: A PyTorch Library for Learning-Based Point Cloud Compression. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24), October 28-November 1, 2024, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3664647.3685512

1 Introduction

Point clouds are datasets composed of numerous discrete three-dimensional (3D) points, which can be collected via 3D scanners, LiDAR, and other devices. They are extensively used across various fields, including immersive media, cultural heritage preservation, and 3D medical analysis. With the widespread adoption of 3D visual sensors, the volume of point cloud data has been rapidly increasing, posing significant challenges for the storage and transmission of hardware devices. Hence, it affects the processing speed and efficiency of 3D visual systems. Consequently, developing efficient point cloud compression (PCC) technologies [18–21] have become a critical research priority. With the rise of deep learning [16, 22, 23], an increasing number of neural network-based end-to-end solutions achieve significant success in image and video compression, and are making substantial strides in the area of 3D PCC. Compared to traditional methods, deep learning-based PCC techniques offer end-to-end optimization, adaptive entropy coding, and support for a variety of task-oriented, intelligent coding objectives.

The Moving Picture Experts Group (MPEG) and Audio Video Coding Standard Working Group (AVS) conduct extensive research on PCC technology aimed at applications in immersive visual experiences and autonomous driving. They develop a variety of innovative technologies and coding tools, successively establishing standards such as MPEG video-based point cloud compression (V-PCC) [24], MPEG geometry-based point cloud compression (G-PCC) [7], and AVS PCC [5], as well as reference software like TMC13, TMC2, and point cloud reference model (PCRM). These efforts play a pivotal role in advancing PCC. Inspired by the success of deep learning in image compression, researchers propose a diverse array of learning-based coding techniques, including point-based, octree-based, voxel-based, and sparse convolution-based coding approaches. Point-based compression technologies utilize networks like PointNet and PointNet++. Octree-based compression represents the point cloud position information using an octree, then implements entropy coding through efficient context models. Voxel-based compression technologies start with voxelization and then employ auto-encoders and hyper-prior networks to achieve efficient data representation and compression. Meanwhile, the sparse convolution-based compression, an advanced voxel method, uses sparse tensor representations to learn features and coordinates.

Despite the flourishing development of deep learning-based PCC techniques, most methods are not open-source, presenting significant challenges for newcomers in the PCC field attempting to replicate these new technologies. Furthermore, the absence of a unified PCC library means that researchers often have to develop

*Corresponding author: Wei Gao. This work was supported by The Major Key Project of PCL (PCL2024A02), Natural Science Foundation of China (62271013, 62031013), Guangdong Province Pearl River Talent Program (2021QN020708), Guangdong Basic and Applied Basic Research Foundation (2024A1515010155), Shenzhen Science and Technology Program (JCYJ20230807120808017), and Sponsored by CAAI-MindSpore Open Fund, developed on OpenI Community (CAAIXSJLJJ-2023-MindSpore07).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '24, October 28-November 1, 2024, Melbourne, VIC, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0686-8/24/10
https://doi.org/10.1145/3664647.3685512

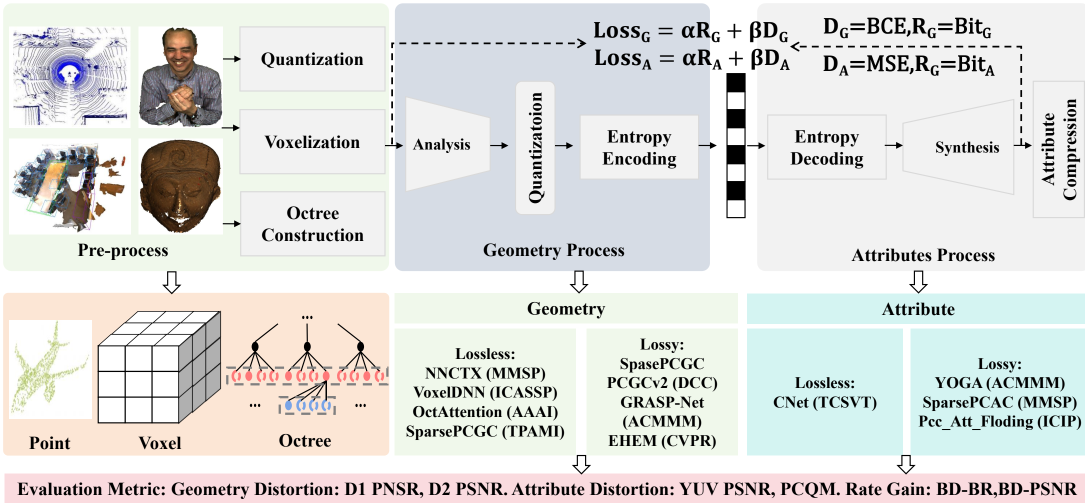


Figure 1: The overview of learning-based PCC frameworks. it include the following steps: pre-processing for geometry compression, encoding transformations, entropy coding, and decoding transformations. Attribute compression follows after geometry compression. The evaluation metrics include PSNR and BD-BR, *etc.* The loss function employs RD optimization loss.

Table 1: Quantitative performance gains in lossless geometry of surfaces PCC over the AVS PCRM codec.

Bpp Gain	NNCTX [6]	G-PCC [7]	V-PCC [24]
MPEG	-6.73%	-15.44%	47.96%
Bpp Gain	**OctAttention [2]**	**SparsePCGC [13]**	**VoxelDNN [9]**
MPEG	-23.49%	-26.58%	-16.54%

Table 2: Quantitative performance gains in lossy geometry surfaces PCC over the AVS PCRM codec.

BD-BR	G-PCC[24]		V-PCC[24]		PCGCv2[14]		SparsePCGC[13]	
	D1	D2	D1	D2	D1	D2	D1	D2
MPEG	-12.3%	-12.9%	-87.4%	-78.0%	-90.4%	-84.6%	-94.5%	-88.8%

underlying codes themselves, consuming valuable time and energy. As each researcher or developer uses different environmental configurations, comparing different outcomes poses excellent challenges, resulting in low re-producibility and diversity of experiment results. Although there exists a PCC algorithm library in [3], its algorithms are relatively outdated, do not cover various data densities, and include only a subset of geometry compression methods. Therefore, developing a unified, open-source library that compasses geometry and attributes and can integrate different data densities is crucial for advancing the PCC. Consequently, we develop a PyTorch-based PCC library, namely LearningPCC. This is an efficient, open-source algorithm library supports various representations for learning-based PCC methods.

The main contributions of LearningPCC are as follows: **(1)** To the best of our knowledge, this is the first open-source deep learning-based algorithm library specifically designed for PCC tasks, providing researchers with a comprehensive insight into the most advanced technologies. **(2)** We integrate 11 representative PCC algorithms, organizing them systematically along various dimensions and replicating most of the code that has not been made public. **(3)** Through multiple sets of evaluation experiments, we calculate the RD performance and gains, runtime, and other metrics for each

Table 3: Quantitative performance gains in lossless attribute surfaces PCC over the AVS PCRM codec.

Bpp Gain	G-PCC [24]	V-PCC [24]	CNeT [8]
MPEG	-18.55%	14.67%	-4.86%

Table 4: Quantitative performance gains in lossless geometry LiDAR PCC over the AVS PCRM codec.

Bpp Gain	G-PCC [24]	OctAttention [2]	SparsePCGC [13]
MPEG	3.84%	-0.82%	-7.41%

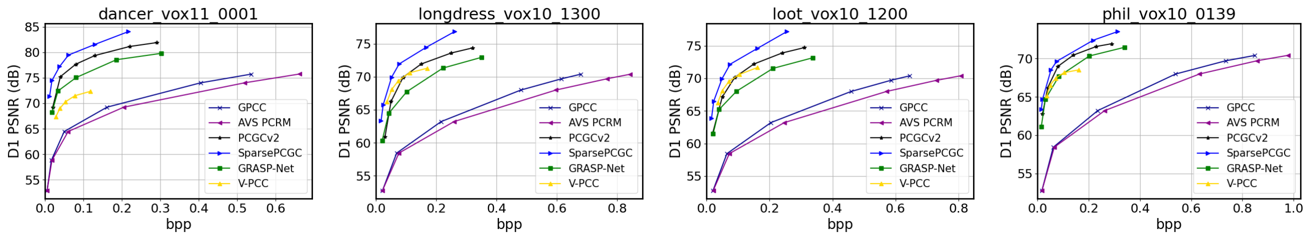
Table 5: Quantitative performance gains in lossy geometry LiDAR PCC over the AVS PCRM codec.

BD-BR	G-PCC [24]		OctAttention [2]		SparsePCGC [13]	
	D1	D2	D1	D2	D1	D2
LiDAR	4.8%	5.1%	-26.0%	-31.3%	-22.0%	-27.6%

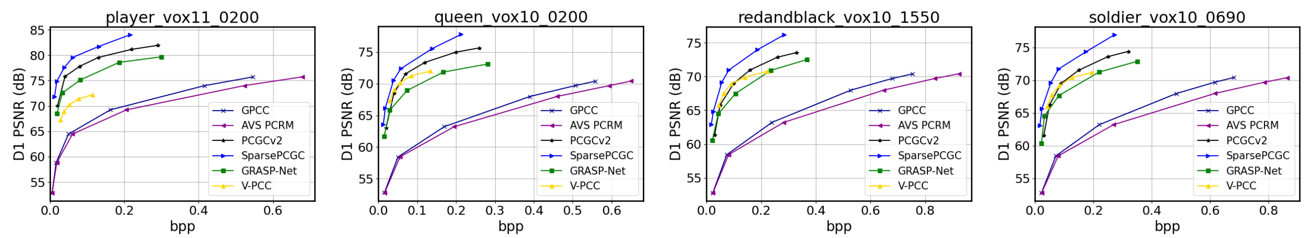
algorithm. We establish processes for data pre-processing (format conversion) and post-processing (various metrics and visualization) to facilitate uniform comparisons.

2 Algorithm Library

Fig.1 shows the workflow of PCC, where the design variations mainly stem from differences in data representation methods. The PCC process includes the following key steps: **Data Pre-processing:** This includes quantization, voxelization, and constructing octrees. **Analysis and Transformation:** Designs for down-sampling processes based on points and voxels are included, excluding those for octrees. **Quantization and Entropy Coding:** Various context-based entropy models are designed for the voxel and octree methods to enhance the accuracy of probability predictions. **Entropy Decoding and Synthesis:** Different up-sampling processes are designed for points and voxels. **Attribute Compression:** This follows a process similar to geometry encoding but uses different coding tools. **Compression Metrics:** This includes various strategies for measuring bitstream fidelity and both subjective and objective evaluation methods. we select 11 representative approaches for classification

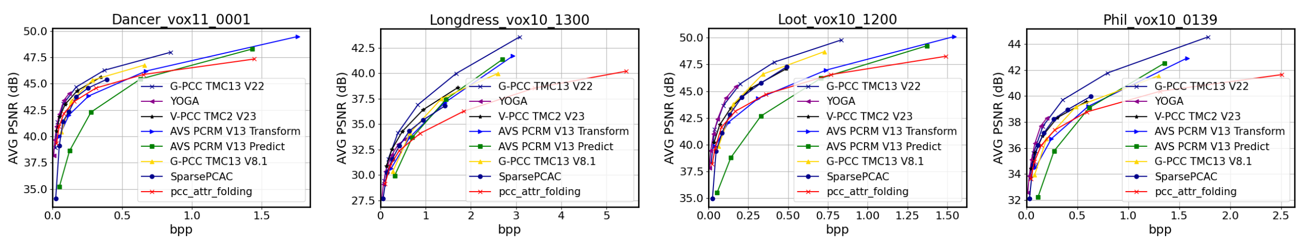


(a) Dancer_vox11_0001_D1 (b) Longdress_vox10_1300_D1 (c) Loot_vox10_1200_D1 (d) Phil_vox10_0139_D1

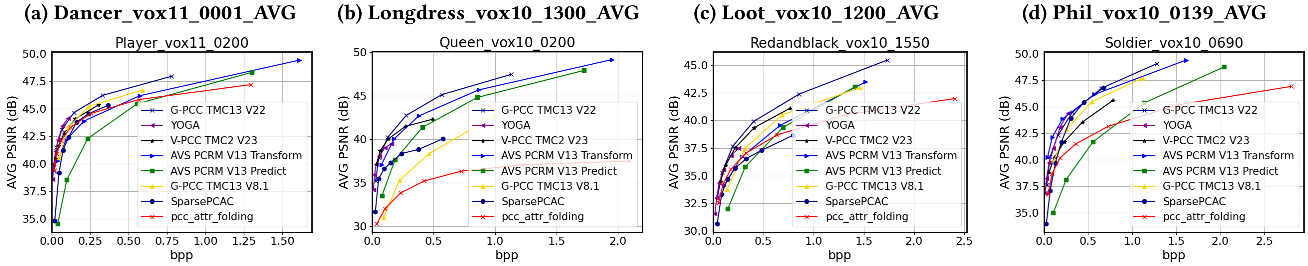


(e) Player_vox11_0200_D2 (f) Queen_vox10_0200_D1 (g) Redandblack_vox10_1550_D1 (h) Soldier_vox10_0690_D1

Figure 2: Comparative analysis of lossy geometry compression for surface point clouds using RD curves of D1 PSNR.



(a) Dancer_vox11_0001_AVG (b) Longdress_vox10_1300_AVG (c) Loot_vox10_1200_AVG (d) Phil_vox10_0139_AVG



(e) Player_vox11_0200_AVG (f) Queen_vox10_0200_AVG (g) Redandblack_vox10_1550_AVG (h) Soldier_vox10_0690_AVG

Figure 3: Comparative analysis of lossy attribute compression for surface point clouds using RD curves of AVG-PSNR.

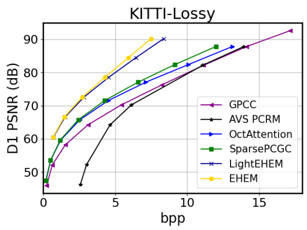
replication and evaluation from the best available methods. Next, we introduce the fundamental principles of these methods.

2.1 Lossless Geometry PCC Approaches

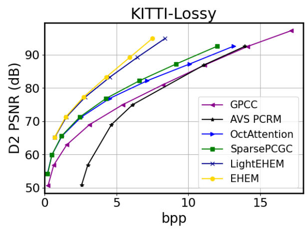
The learning-based lossless point cloud geometry compression methods utilize octree and voxel representations. They encompass four categories: NNCTX [6] presents a lossless PCC algorithm that models voxel occupancy probabilities using a neural network, which encodes 3D contexts from sequential octree layers via arithmetic coding. VoxelDNN [9] uses a hybrid of octree and voxel-based coding, adaptively partitioning point clouds into multi-resolution blocks, leveraging octrees for signaling and neural networks for explicit geometry processing. OctAttention [2] employs an octree structure for efficient point cloud representation, encoding octrees losslessly while leveraging sibling and ancestor contexts with a conditional entropy model and attention mechanism. SparsePCGC [13] compresses point clouds by focusing on most-probable positively-occupied voxels using a sparse convolution-based neural network, exploiting cross-scale and same-scale correlations to enhance compression efficiency. EHEM [12] improves attention model with a hierarchical structure of linear complexity, maintaining global receptive fields and addressing auto-regression in decoding while preserving compression performance.

2.2 Lossy Geometry PCC Approaches

Lossy point cloud geometry compression methods primarily utilize voxelization or sparse tensor representations, compressing point clouds through sparse convolutions or residual encoding. This approach includes three types of algorithms: PCGCv2 [14] introduces a multiscale end-to-end learning framework for point cloud geometry reconstruction, utilizing sparse convolutions and hierarchical resampling, with octree and probabilistic models for compression. SparsePCGC [13] supports both lossless and lossy compression.



(a) KITTI_Lossy_D1



(b) KITTI_Lossy_D2

Figure 4: Comparative analysis of lossy geometry LiDAR PCC using RD curves of D1 PSNR and D2 PSNR.

The lossy compression is achieved through a multi-stage encoding process, allowing flexible exploration of bitstream information of varying sizes based on encoding needs. GRASP-Net [10] employs a deep learning-based heterogeneous approach for lossy PCC, featuring a base and enhancement layer to enhance detail reconstruction, supported by a skip mode for resilience.

2.3 Lossless Attribute PCC Approaches

Due to the significant variability in the distribution of point cloud color attributes, compression presents challenges, and learning-based PCC methods are still in their preliminary stages. The most representative method for lossless attribute compression is CNet [8]. CNet introduces an efficient, learning-based lossless PCC method using sparse tensor-based deep neural networks to model geometry and color probabilities, exploiting feature-wise and point-wise dependencies for precise arithmetic coding.

2.4 Lossy Attribute PCC Approaches

Lossy attribute encoding methods can be categorized into two types: one directly compresses attribute color information, while the other combines geometry and attribute compression. These methods employ auto-encoder structures, and with the increasing use of sparse convolutions, more approaches are adopting sparse representations for compressing both geometry and attribute information. We will introduce three methods: Pcc_Att_Floding [11] treats point clouds as 2D manifolds in 3D space, mapping attributes to a 2D grid via an optimized method, enabling the application of image processing techniques with strategies to minimize mapping distortion. SparsePCAC [15] employs sparse convolutions within a variational auto-encoder framework to compress point cloud color attributes efficiently, using an adaptive entropy model for precise bitrate estimation. YOGA [25] offers flexible, separable lossy compression of geometry and color attributes using a single model. It achieves significant efficiency and quality gains, employs sparse convolutions, and supports variable-rate coding.

2.5 Lossless Geometry LiDAR PCC Approaches

LiDAR point cloud lossless compression primarily utilizes algorithms that employ octree structures, such as SparsePCGC, OctAttention, and EHEM. These methods leverage octrees combined with context encoding to compress the LiDAR point cloud data with sparse distribution efficiently.

2.6 Lossy Geometry LiDAR PCC Approaches

Lossy LiDAR PCC methods are similar to lossless methods, but lossy primarily arises from quantization of the source data to calculate different bitstream distortion values. The primary methods employed include SparsePCGC, OctAttention, and EHEM.

3 Experiment

Encoding Methods: (1) Lossless geometry compression: NNCTX, VoxelDNN, OctAttention, SparsePCGC (lossless part), and EHEM. (2) Lossy geometry compression: SparsePCGC (lossy part), PCGCv2, GRASP-Net [10]. (3) Lossless attribute compression: CNeT. (4) Lossy attribute compression: SparsePCAC, Pcc_Att_Floding, and YOGA. (5) LiDAR PCC: OctAttention, SparsePCGC, and EHEM. (6) Non-Learning compression methods: G-PCC-prelift branch, AVS PRCM-predict branch, and V-PCC. (7) Version Information: G-PCC TMC13-v22, AVS PCRM-v13, V-PCC TMC2-v23.

Evaluation Datasets: The training datasets are sampling from ShapeNet [1], ModelNet [17] and KITTI [4], with testing on the MPEG standards test dataset and KITTI test dataset. We organize these training and testing datasets and make them available at OpenDatasets[1]. The MPEG point cloud dataset features a bit width of 10 bits, with approximately one million points. The LiDAR point cloud test dataset is a bit width of 2cm (14 bits), totaling 110 frames.

Evaluation Criteria: We use bpp (bits per point) to measure bitrate and assess distortion from lossy compression using D1 (point-to-point distance) and D2 (point-to-plane distance) alongside PSNR (Peak Signal-to-Noise Ratio) as metrics. The attribute lossy encoding utilizes YUV PSNR and average YUV component PSNR as metrics. BD-BR (Bjøntegaard Delta Rate) is utilized to quantify performance gains. To evaluate lossless compression performance, we employ the compression ratio (Bpp Gain) as the metric. Additionally, we will assess the encoding complexity based on time, providing insight into the efficiency and effectiveness of various methods.

Geometry and Attributes Evaluation Results: **(1) Lossless geometry compression:** Tab. 1 shows that the performance ranking is SparsePCGC > OctAttention > VoxelDNN > G-PCC > NNCTX > AVS PCRM > V-PCC. **(2) Lossy geometry compression:** The Fig. 2 demonstrates the superior advantage of learning-based methods. Tab. 2 presents the performance ranking for dense point cloud dataset as follows: SparsePCGC > PCGCv2 > V-PCC > GRASP-Net > G-PCC > AVS PCRM. **(3) Lossless attribute compression:** Tab. 3 shows the ranking for dense point clouds: C-Net > G-PCC > AVS PCRM > V-PCC. **(4) Lossy attribute compression:** Fig. 3 presents the ranking for dense point clouds: YOGA > SparsePCAC > G-PCC > AVS > PCC_ATTR_FOLDING. **(5) Lossless LiDAR compression:** Tab. 4 shows the compression performance ranking as follows: SparsePCGC > OctAttention > AVS PCRM > G-PCC. **(6) Lossy LiDAR compression:** Tab. 5 and Fig. 4 present the compression ranking: EHEM > SparsePCGC > OctAttention > AVS PCRM > G-PCC.

4 Conclusion

We present LearningPCC, an innovative open-source library for point cloud compression (PCC) based on deep learning. LearningPCC is the first comprehensive library of its kind, featuring 11 advanced algorithms for geometry and attributes, all systematically organized and including previously unavailable code. We conduct thorough evaluations of rate-distortion performance, runtime, and other key metrics. Additionally, we establish standardized protocols for data pre-processing and post-processing to ensure consistent comparisons. This significantly enhances both the understanding and application of PCC technologies.

[1] https://openi.pcl.ac.cn/OpenDatasets/AVS_Test_Data

References

[1] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An Information-rich 3D Model Repository. *ArXiv Preprint ArXiv:1512.03012* (2015).

[2] Chunyang Fu, Ge Li, Rui Song, Wei Gao, and Shan Liu. 2022. OctAttention: Octree-based Large-scale Contexts Model for Point Cloud Compression. *ArXiv Preprint ArXiv:2202.06028* (2022).

[3] Wei Gao, Hua Ye, Huiming Zheng, Yuyang Wu, and Liang Xie. 2022. OpenPointCloud: An Open-Source Algorithm Library of Deep Learning Based Point Cloud Compression. In *ACM International Conference on Multimedia*. 7347–7350.

[4] Andreas Geiger, Philip Lenz, and Raquel Urtasun. 2012. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3354–3361. https://doi.org/10.1109/CVPR.2012.6248074

[5] AVS Point Cloud Compression Working Group. September, 2022. Reference Software Algorithm Description of AVS Point Cloud Coding. *Audio Video Standard, N3445, Online* (September, 2022).

[6] Emre Can Kaya and Ioan Tabus. 2021. Neural Network Modeling of Probabilities for Coding the Octree Representation of Point Clouds. In *International Workshop on Multimedia Signal Processing*. IEEE, 1–6.

[7] Khaled Mammou, Philip A. Chou, David Flynn, Maja Krivokuća, Ohji Nakagami, and Toshiyasu Sugio. January, 2019. G-PCC Codec Description v2. *ISO/IEC TC1/SC29/WG11 MPEG, N18189, Marrakech* (January, 2019).

[8] Dat Thanh Nguyen and André Kaup. 2023. Lossless Point Cloud Geometry and Attribute Compression Using a Learned Conditional Probability Model. *IEEE Transactions on Circuits and Systems for Video Technology* (2023).

[9] Dat Thanh Nguyen, Maurice Quach, and Giuseppe Valenzise. 2021. Learning-based Lossless Compression of 3D Point Cloud Geometry. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 4220–4224.

[10] Jiahao Pang, Muhammad Asad Lodhi, and Dong Tian. 2022. GRASP-Net: Geometric Residual Analysis and Synthesis for Point Cloud Compression. In *International Workshop on Advances in Point Cloud Compression, Processing and Analysis*. 11–19.

[11] Maurice Quach, Giuseppe Valenzise, and Frederic Dufaux. 2020. Folding-based Compression of Point Cloud Attributes. In *IEEE International Conference on Image Processing*. IEEE, 3309–3313.

[12] Rui Song, Chunyang Fu, Shan Liu, and Ge Li. 2023. Efficient Hierarchical Entropy Model for Learned Point Cloud Compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14368–14377.

[13] Jianqiang Wang, Dandan Ding, Zhu Li, Xiaoxing Feng, Chuntong Cao, and Zhan Ma. 2021. Sparse Tensor-based Multiscale Representation for Point Cloud Geometry Compression. *ArXiv Preprint ArXiv:2111.10633* (2021).

[14] Jianqiang Wang, Dandan Ding, Zhu Li, and Zhan Ma. 2021. Multiscale Point Cloud Geometry Compression. In *Data Compression Conference*. IEEE, 73–82.

[15] Jianqiang Wang and Zhan Ma. 2022. Sparse Tensor-based Point Cloud Attribute Compression. In *International Conference on Multimedia Information Processing and Retrieval*. IEEE, 59–64.

[16] Yuyang Wu, Liang Xie, Shangkun Sun, Wei Gao, and Yiqiang Yan. 2024. Adaptive Intra Period Size for Deep Learning-based Screen Content Video Coding. In *IEEE International Conference on Multimedia and Expo Workshops*. IEEE.

[17] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, and Xiaoou Tang. 2015. 3D ShapeNets: A deep representation for volumetric shapes. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1912–1920.

[18] Liang Xie, Wei Gao, and Songlin Fan. 2024. PDNet: Parallel Dual-branch Network for Point Cloud Geometry Compression and Analysis. In *2024 Data Compression Conference*. 596–596. https://doi.org/10.1109/DCC58796.2024.00113

[19] Liang Xie, Wei Gao, and Huiming Zheng. 2022. End-to-end Point Cloud Geometry Compression and Analysis with Sparse Tensor. In *International Workshop on Advances in Point Cloud Compression, Processing and Analysis*. 27–32.

[20] Liang Xie, Wei Gao, Huiming Zheng, and Ge Li. 2024. SPCGC: Scalable Point Cloud Geometry Compression for Machine Vision. In *IEEE International Conference on Robotics and Automation*. 594–595.

[21] Liang Xie, Wei Gao, Huiming Zheng, and Hua Ye. 2024. Semantic-Aware Visual Decomposition for Point Cloud Geometry Compression. In *2024 Data Compression Conference*. 595–595. https://doi.org/10.1109/DCC58796.2024.00112

[22] Liang Xie, MengHao Hu, and XinBei Bai. 2022. Online Improved Vehicle Tracking Accuracy via Unsupervised Route Generation. In *IEEE 34th International Conference on Tools with Artificial Intelligence*. IEEE, 788–792.

[23] Liang Xie, MengHao Hu, XinBei Bai, and WenKe Huang. 2022. Towards Hardware-Friendly and Robust Facial Landmark Detection Method. In *International Conference on Neural Information Processing*. Springer, 432–444.

[24] Vladyslav Zakharchenko. January, 2019. V-PCC Codec Description. *ISO/IEC TC1/SC29/WG11 MPEG, N18190, Marrakech* (January, 2019).

[25] Junteng Zhang, Tong Chen, Dandan Ding, and Zhan Ma. 2023. YOGA: Yet Another Geometry-based Point Cloud Compressor. In *ACM International Conference on Multimedia*. 9070–9081.