PCHMVision: An Open-Source Library of Point Cloud Compression for Human and Machine Vision

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

In today's era, three-dimensional point cloud data is not only voluminous but also widely applicable. Therefore, data compression has become a crucial step prior to processing. Although existing 3D point cloud compression techniques primarily focus on fidelity, in practical applications, the vast majority of compressed data serves machine perception tasks. Therefore, point cloud compression tailored for machine perception becomes particularly significant. To address this problem, we introduce an innovative point cloud compression algorithm library specifically designed for both machine and human perceptual requirements. This library represents the first collection of multi-perception point cloud compression algorithms on the PyTorch platform, integrating eleven advanced, learning-based algorithms. We category and analyze these algorithms in depth, according to different analysis tasks, to facilitate a better understanding and comparison. Moreover, we successfully replicate these algorithms and meticulously organize the preprocessing of point cloud data and the analysis networks for downstream tasks. Ultimately, we conduct experiments on multiple perceptual datasets for compression and analysis tasks, with results comprehensively summarized across various performance metrics. We will continue to update these algorithms to ease their adoption by researchers. The relevant codes and evaluated results are released at https://openi.pcl.ac.cn/OpenPointCloud/PCHMVision.

CCS Concepts

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

Keywords

Point Cloud Compression, Human and Machine Vision.

ACM Reference Format:

Liang Xie and Wei Gao. 2024. PCHMVision: An Open-Source Library of Point Cloud Compression for Human and Machine Vision. 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/10.1145/3664647.3685513

*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).

MM '24, October 28-November 1, 2024, Melbourne, VIC, Australia

ACM ISBN 979-8-4007-0686-8/24/10
https://doi.org/10.1145/3664647.3685513

1 Introduction

With the rapid advancement of 3D technology and deep learning, point clouds become an essential representation of 3D visual data, widely used in augmented reality, autonomous driving, and many other fields. Point cloud data provides rich spatial and structural information crucial for machine vision tasks such as classification, detection, and segmentation. However, the high dimensionality and large volume of point cloud data present challenges in transmission and processing, especially in environments with limited bandwidth and storage space. To effectively manage this data, point cloud compression (PCC) technologies have been developed. Traditional PCC methods, including geometry-based point cloud compression (G-PCC) and video-based point cloud compression (V-PCC), as well as learning-based methods like point-based, octree-based, and voxel-based compression, excel at balancing compression ratios and minimizing distortion, primarily focusing on delivering reconstruction quality at specified bit rates. These technologies primarily cater to human visual quality rather than machine tasks.

With the progress in 5G technologies and the rise of applications, such as smart cities and unmanned aerial vehicle navigation, there is a rapidly growing demand for machine analysis of point cloud data. This demands not only efficient transmission of compressed point clouds but also optimization of compression methods to support machine vision tasks, reducing computational burdens and signal distortion. Although existing image coding methods, such as the feature-based image coding for machines framework, compress image features instead of the image to lower transmission costs and allow downstream tasks to perform intelligent analysis, these approaches have not yet been widely applied to point cloud data and current PCC technologies still focus on human visual quality. Therefore, developing PCC approaches that optimize machine tasks without compromising human quality is crucial.

Although numerous PCC technologies that cater to machine and human perception have emerged, most of these technologies are not open-sourced, such as [3, 4], posing significant challenges in the field who are trying to replicate these methods. Furthermore, the absence of a unified PCC algorithm library means that researchers often have to spend considerable time replicating existing codes, and the variation in environmental setups or developers makes comparing outcomes difficult. Therefore, developing a new PCC algorithm library that can enhance machine vision performance while preserving high-quality human visual effects is crucial for supporting real-time, efficient machine analysis tasks in automated systems and smart applications [19, 24, 25].

Table 1: We summarize the point cloud compression algorithms employed, covering mainstream and representative methods.

Category	Name	Publish	Open	Highlight
Classification	PCGC-AST[21]	ACMMM22	Yes	The paper introduces a machine vision-oriented point cloud lossy compression method featuring a multi-scale channel attention module, optimizing compression performance and integrating semantic information to preserve point cloud classification accuracy.
	LPCC-C[14]	MMSP23	Yes	The method introduces a point cloud codec for classification tasks, achieving high encoding ratios without sacrificing accuracy, and outperforming non-specialized codecs.
	PDNet[20]	DCC24	Yes	PDNet introduces a Transformer-based framework for lossy PCC that balances compact representation with semantic retention, facilitating classification from the bitstream.
	SHM-PCC[15]	arXiv24	No	This paper introduces a scalable codec optimized for classification tasks on edge devices, reducing bitrate without losing inference accuracy and accommodating human viewing.
Detection	R-PCC[18]	ICRA22	Yes	A range image-based method that reduces data while preserving depth information.
	PCHM-Net[8]	ICME23	No	PCHM-Net introduces a novel PCC framework optimized for both human and machine vision, enhancing coding performance while efficiently handling classification tasks.
	JOPCC-OD[7]	ICIP23	No	The approach introduces a learning-based PCC framework optimized for object detection, featuring a gradient bridge for joint optimization and a progressive training strategy.
	SAVD[23]	DCC24	Yes	The method introduces a semantic-aware visual decomposition strategy to optimize PCC by allocating more resources to foreground regions, improving detection performance.
Segmentation	RSPCC-SPR[27]	TCSVT22	No	The proposed system introduces a semantic prior representation for efficient PCC.
	HM-PCGC[10]	ICIP23	No	The HM-PCGC balances human and machine needs by integrating multi-task features.
	SPCGC[22]	ICRA24	Yes	Proposes a scalable framework that compresses point cloud while preserving semantic information, thereby enhancing vision task performance without compromising fidelity.

Therefore, we develop an open-source PCC library, namely PCH-MVision, based on PyTorch. This library is efficient and supports multiple learning-based PCC algorithms. The main contributions include: **Firstly**, to the best of our knowledge, it is the first comprehensive open-source deep learning algorithm library designed specifically for PCC tasks aimed at machine perception, providing researchers with a thorough analysis and compilation of the most advanced technologies in the field. **Secondly**, we integrate and systematically organize 11 representative PCC algorithms and successfully replicated most of the previously unpublished codes. **Lastly**, we conduct a series of evaluation experiments to assess each algorithm's compression performance, classification accuracy, and detection or segmentation precision, facilitating broad comparisons. **Finally**, we establish a complete data pre-processing (format conversion) and post-processing (task evaluation) workflow to ensure uniformity and consistency in evaluations.

2 Supported Algorithm Library

As illustrated in Fig. 1 and Tab. 1, we categorize PCC algorithms into four main paradigms. Typically, the input data includes ModelNet and ShapeNet labeled for classification, SUN RGBD and ScanNet labeled for indoor scene detection and segmentation, and KITTI labeled for outdoor scene segmentation, among others. **(a)** feeds the reconstructed point cloud data into machine analysis tasks, with representative methods including PCGC-AST [21] and SAVD [23]. **(b)** selects features necessary for machine analysis tasks during the compression process, achieving simultaneous reconstruction and analysis, exemplified by LPCC-C [14] and SHM-PCC [15]. **(c)** complements the compression and analysis tasks, allowing downstream tasks to leverage more semantic information, such as PCHM-Net [8]. **(d)** involves jointly optimizing compression and analysis tasks or compressing data through multiple bitstreams based on downstream task requirements, such as SPCGC [22] and JOPCC-OD [7]. We select 11 representative methods, briefly explain the principles, replicate the codes, and evaluate these approaches.

2.1 Classification-oriented PCC Approaches

Several methods exist for integrating point cloud classification tasks with PCC tasks, including: PCGC-AST [21] develops an end-to-end sparse tensor-based PCC network that optimizes feature utilization and classification performance using a novel multiscale SEM block. LPCC-C [14] introduces a specialized point cloud codec, based on PointNet, which offers superior rate-accuracy trade-off and lightweight configurations suitable for low-resource devices, enabling efficient server-side processing. PDNet [20] designs a dual-branch network utilizing a Transformer-based encoder-decoder with local and global attention for efficient, analysis-friendly lossy PCC and simultaneous classification. SHM-PCC [15] introduces a scalable, task-specialized codec for point cloud data based on PointNet++, designed for classification tasks and human vision, which optimizes bitrate usage and maintains accuracy on the ModelNet40 dataset.

2.2 Detection-oriented PCC Approaches

The following approaches simultaneously address point cloud detection and PCC tasks, for example: R-PCC [18] develops a range image-based PCC method that segments large-scale point clouds for efficient compression, achieving a 30 compression ratio while preserving fidelity for 3D object detection and SLAM tasks. PCHM-Net [8] introduces a dual-branch PCC framework with a shared octree-based module, optimizing bitrate efficiency by selecting sparse points for enhanced feature extraction and classification performance on benchmark datasets. JOPCC-OD [7] presents a

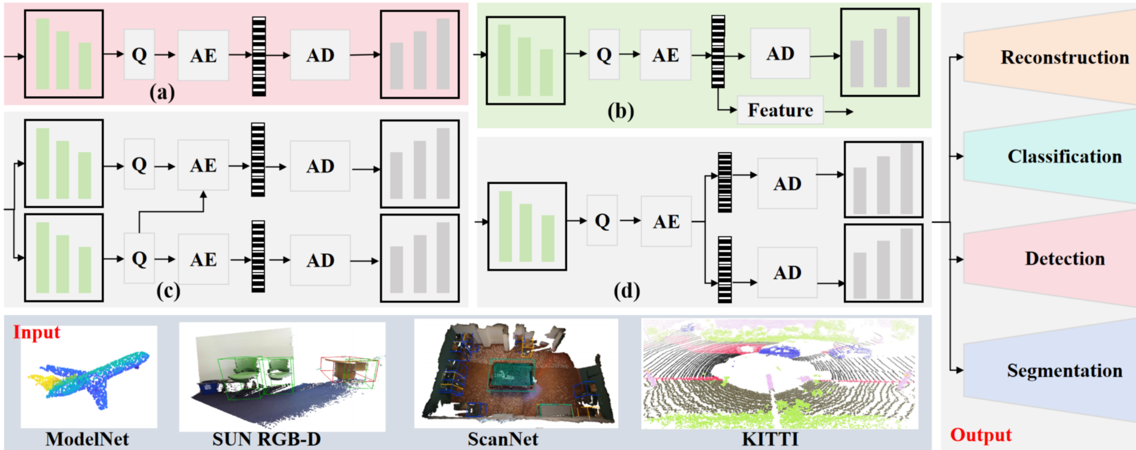

Figure 1: The point cloud compression frameworks for machine and human vision perception.

Table 2: The classification performance of LPCC-C [14] compared with other learning based methods in ModelNet40 dataset. The input point number is 1024 and the classification method is PointNet.

LPCC-C [14]		Draco [5]		OctAttention [1]		G-PCC [11]		IPDAE [26]	
Bpp	OA	Bpp	OA	Bpp	OA	Bpp	OA	Bpp	OA
8.055	45.137	1376	4.052	64	4.052	72	4.052	64	4.052
10.378	59.359	1423.086	43.72	236.762	70.237	82.431	11.831	95.044	41.613
14.498	69.489	1461.099	61.831	395.235	80.91	136.596	42.22	161.511	71.191
16.711	72.487	1516.532	74.554	636.56	84.654	152.055	50.081	515.685	79.026
20.964	76.539	1612.794	80.956	998.616	86.862	165.575	53.768	1017.303	83.144
30.778	81.077	1782.963	83.874	1480.256	87.864	204.963	63.209	1268.956	83.752
46.490	83.954	2036.301	85.778	2032.619	88.252	274.386	73.947	1710.159	85.170
60.244	85.008	2390.648	87.196	5711.102	88.374	359.857	77.715	3651.494	86.993

Table 3: The compression and classification performance of PCGC-AST [21] in ModelNet40 dataset.

PCGC-AST [21]			PointNet [13]		DGCNN [12]		PointMLP [9]	
Bpp	D1	D2	OA	CA	OA	CA	OA	CA
1.699	58.757	63.482	0.778	0.710	0.411	0.331	0.322	0.252
1.650	58.503	63.319	0.777	0.706	0.410	0.333	0.294	0.240
1.547	57.925	62.805	0.761	0.692	0.392	0.316	0.336	0.278
1.462	56.825	61.742	0.713	0.667	0.350	0.290	0.245	0.213
1.385	55.953	60.553	0.669	0.637	0.354	0.299	0.287	0.251
1.360	55.917	60.606	0.671	0.639	0.344	0.279	0.293	0.250

Table 4: The compression and classification performance of PDNet [20] in ModelNet40 and ShapeNet datasets, the classification method is Point Transformer.

	Gao [2]			PDNet [20]		Gao [2]		PDNet [20]	
Input	bpp	D1	D2	D1	D2	OA	CA	OA	CA
ShapeNet	0.007	28.911	31.958	31.009	33.984				
	0.027	32.225	35.629	33.654	37.440				
	0.075	33.759	38.518	33.654	37.433	-		-	
	0.111	33.901	38.414	-	-				
	0.159	33.877	37.969	-	-	0.604	0.505	0.843	0.787
ModelNet	0.020	28.119	30.690	28.757	31.015				
	0.0493	30.283	33.587	31.169	34.142				
	0.122	30.609	34.438	30.874	34.892	-		-	
	0.137	30.502	34.499	31.366	34.730				
	0.204	31.115	34.592	30.895	34.671				

learning-based PCC framework optimized for 3D object detection in autonomous driving, featuring a gradient bridge for continuous learning and a progressive training strategy, enhancing compression without sacrificing accuracy. SAVD [23] introduces a PCC strategy that enhances ROI encoding by distinguishing foreground and background regions and allocating more bitstream to critical areas, thus preserving essential information for machine tasks.

Table 5: The compression and detection performance of SAVD [23] compared with other methods in ScanNet dataset. The larger the value, the better the performance.

Algorithm	Bpp	D1	D2	Bpp	mAP@0.25	mAP@0.5
G-PCC [11]	2.351	63.223	68.341	2.351	0.627	0.416
	0.626	58.495	63.389	0.626	0.535	0.315
	0.178	52.918	57.682	0.178	0.052	0.005
	0.063	47.077	52.122	0.063	0.001	-
AVS PCRM [6]	2.298	51.145	56.232	2.298	0.632	0.421
	0.586	46.320	51.039	0.586	0.532	0.312
	0.155	40.672	45.117	0.155	0.049	0.006
	0.050	34.795	39.381	0.050	0.001	-
PCGCv2 [17]	2.404	52.336	61.392	2.404	0.182	0.062
	2.005	52.479	61.549	2.005	0.177	0.059
	2.005	52.479	61.549	1.715	0.170	0.049
	1.164	52.316	61.427	1.164	0.16	0.054
SparsePCGC [16]	2.946	63.21	68.324	2.946	0.619	0.404
	0.593	57.239	60.083	0.593	0.518	0.273
	0.127	51.498	53.678	0.127	0.039	0.003
	0.032	45.717	48.423	0.032	0.001	-
SAVD [23]	2.179	62.009	68.606	2.256	0.634	0.414
	0.599	58.205	61.368	1.231	0.637	0.398
	0.445	55.53	60.262	0.912	0.600	0.376
	0.096	47.741	52.233	0.391	0.461	0.209

2.3 Segmentation-oriented PCC Approaches

There are several methods for integrating PCC with segmentation tasks, such as: SPCC-SPR [27] presents a novel LiDAR PCC system emphasizing machine perception, featuring a semantic prior representation (SPR) for scene-aware segmentation and lossy compression that optimizes localization accuracy while maintaining

Table 6: The compression and classification performance of SPCGC in ShapeNet dataset, the classification and segmentation method is DGCNN.

Algorithm	Bpp	D1	D2	Bpp	OA	CA	mIoU
G-PCC [11]	0.224	29.215	36.269	-	-	-	-
	0.359	34.963	41.176	0.359	0.717	0.532	0.514
	0.889	40.899	46.419	0.889	0.839	0.670	0.731
	2.866	46.884	51.959	2.866	0.917	0.766	0.820
AVS PCRM [6]	0.107	28.954	34.355	0.107	0.621	0.443	0.386
	0.240	34.932	40.102	0.240	0.731	0.546	0.529
	0.781	40.982	46.142	0.781	0.855	0.680	0.733
	2.842	46.961	52.037	2.842	0.919	0.768	0.821
PCGCv2 [17]	1.475	30.04	42.029	0.286	0.624	0.410	0.442
	2.318	30.219	42.911	2.318	0.769	0.550	0.597
	3.158	30.862	43.777	3.158	0.756	0.537	0.588
	3.957	31.172	44.265	-	-	-	-
SparsePCGC [16]	1.020	35.879	43.102	1.020	0.582	0.497	0.465
	1.735	41.672	48.622	1.735	0.775	0.630	0.694
	3.529	47.376	53.373	3.529	0.878	0.756	0.788
	6.345	52.985	58.184	-	-	-	-
SPCGC [22]	0.541	26.723	28.364	0.541	0.669	0.505	0.462
	1.137	40.987	43.979	1.137	0.772	0.633	0.617
	2.257	45.413	49.684	2.257	0.770	0.644	0.618
	6.442	48.083	52.837	-	-	-	-

Table 7: The compression and classification performance of SPCGC in ModelNet40 dataset, the classification and segmentation method is PointNet2.

Algorithm	Bpp	D1	D2	Bpp	OA	CA
G-PCC [11]	0.237	29.042	34.473	0.406	0.601	0.555
	0.406	34.912	40.025	1.178	0.847	0.789
	1.178	40.874	45.847	4.015	0.911	0.869
	4.015	46.889	51.811	7.172	0.913	0.876
AVS PCRM [6]	0.116	29.015	34.826	0.276	0.607	0.548
	0.276	34.984	40.499	1.066	0.871	0.815
	1.066	40.979	46.244	4.025	0.914	0.880
	4.025	46.998	52.258	7.246	0.914	0.879
PCGCv2 [17]	0.850	30.580	42.060	0.570	0.516	0.572
	1.490	32.310	44.130	1.490	0.661	0.648
	2.220	32.470	44.620	3.550	0.682	0.667
	3.550	33.460	45.000	-	-	-
SparsePCGC [16]	1.163	35.503	42.349	1.163	0.709	0.688
	2.226	41.352	47.673	2.226	0.861	0.828
	4.835	47.223	52.743	4.835	0.904	0.859
	7.777	52.928	57.921	7.777	0.911	0.869
SPCGC [22]	0.535	30.453	31.941	0.535	0.572	0.52
	1.595	41.675	45.382	1.595	0.878	0.829
	4.898	44.956	49.36	4.898	0.893	0.851
	13.584	49.054	53.566	-	-	-

real-time performance. HM-PCGC [10] tailors for both human and machine vision, employing a lightweight backbone with a learned semantic mining module for multi-task feature aggregation, enhancing geometry fidelity and semantic information. SPCGC [22] introduces a scalable point cloud geometry compression framework that enhances machine perception by integrating a base layer for geometry and an enhancement layer for semantic-guided residuals, optimizing rate distortion (RD) for improved task performance.

3 BENCHMARKING RESULTS AND ANALYSIS

Encoding Methods: Based on the analysis and statistics of the compression algorithms above, we conduct experiment analysis across three downstream tasks: **(1)** For the classification task, evaluations will be carried out using three classic algorithms: PCGC-AST [21], LPCC-C [14], SAVD [23]. **(2)** The detection task will be executed using the algorithm: SAVD [23]. **(3)** For the segmentation task, experiments will be conducted using the algorithm SPCGC [22]. We will continuously train and test other learning based algorithms and update the results as they repeat success.

Evaluation Datasets and Metrics: For the tasks described, we utilize three datasets for training and testing: (1) classification datasets: ShapeNet and ModelNet, which are point cloud datasets of objects extracted from CAD meshes with classification labels. (2) detection datasets: SUN RGB-D and ScanNet are large-scale indoor real-scene RGB-D datasets collected using 3D cameras and feature extensive semantic annotations. (3) segmentation datasets: ShapeNetSegment, a smaller, more densely annotated subset of ShapeNet, provides category labels for each point. We also employ the KITTI dataset for segmentation tasks due to its extensive outdoor scenes and higher segmentation challenges. We use bits per point (bpp) to measure bitrate and employ point-to-point distance (D1), point-to-plane distance (D2), and peak signal-to-noise ratio (PSNR) as metrics to evaluate the distortion of PCC. For classification tasks, we measure accuracy using category accuracy (CA) for individual categories and overall accuracy (OA) across all categories. Detection tasks are evaluated using the mean average precision (mAP) at thresholds of 0.25 and 0.5. Segmentation tasks are assessed using the mean intersection over union (mIoU).

Evaluation Results: Due to the architectural differences, we cannot compare all open-source PCC algorithms together. Instead, we assess them by replicating the comparisons made in their respective publications. In classification tasks, we obtain the following results: (1) The LPCC-C algorithm demonstrates superior classification performance at low bitrates compared to OctAttention, Draco, IPDAE, and D-PCC (Tab. 2). (2) For the PCGC-AST algorithm (Tab. 3), using PointNet to process compressed data achieves an accuracy of 0.67%, surpassing traditional methods G-PCC. (3) The PDNet algorithm (Tab. 4), on ShapeNet and ModelNet datasets, achieves a higher accuracy at lower bitrates than Gao's [2] algorithm, with an 20% improvement. In detection tasks, the detection accuracy ranks as SAVD > SparsePCGC > AVS PCRM > G-PCC > PCGCv2, highlighting the importance of bitrate in detection areas (Tab. 5). For segmentation tasks, the SPCGC algorithm outperforms SparsePCGC, AVS PCRM, G-PCC, and PCGCv2, and similar outcomes are observed using either the DGCNN or PointNet segmentation schemes (Tab. 6 and Tab. 7).

4 Conclusion

Existing 3D point cloud compression techniques often focus on human vision fidelity, but the demand for machine perception-oriented compression is growing. We introduce a novel library of point cloud compression algorithms on the PyTorch platform to meet this need, designed for both machine and human perception. This collection, the first of its kind, features eleven advanced, deep learning-based algorithms categorized and analyzed for various perceptual tasks. We conduct detailed experiments on multiple datasets, providing a comprehensive performance evaluation for classification, detection, and segmentation tasks. In the future, we will add more compression algorithms designed for both human and machine vision, and we will continue to maintain the library to provide valuable and up-to-date information.

References

[1] Chunyang Fu, Ge Li, Rui Song, Wei Gao, and Shan Liu. 2022. Octattention: Octree-based Large-scale Contexts Model for Point Cloud Compression. In *AAAI conference on artificial intelligence*, Vol. 36. 625–633.

[2] Linyao Gao, Tingyu Fan, Jianqiang Wan, Yiling Xu, Jun Sun, and Zhan Ma. 2021. Point Cloud Geometry Compression via Neural Graph Sampling. In *IEEE International Conference on Image Processing*. IEEE, 3373–3377.

[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] Wei Gao, Hang Yuan, Ge Li, Zhu Li, and Hui Yuan. 2023. Low Complexity Coding Unit Decision for Video-based Point Cloud Compression. *IEEE Transactions on Image Processing* 33 (2023), 149–162.

[5] Google. 2020. Draco Compression Library. https://github.com/google/draco.

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

[7] Bojun Liu, Shanshan Li, Xihua Sheng, Li Li, and Dong Liu. 2023. Joint Optimized Point Cloud Compression for 3d Object Detection. In *IEEE International Conference on Image Processing*. IEEE, 1185–1189.

[8] Lei Liu, Zhihao Hu, and Jing Zhang. 2023. PCHM-Net: A New Point Cloud Compression Framework for Both Human Vision and Machine Vision. In *IEEE International Conference on Multimedia and Expo*. IEEE, 1997–2002.

[9] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. 2022. Rethinking Network Design and Local Geometry in Point Cloud: A Simple Residual MLP Framework. *arXiv preprint arXiv:2202.07123* (2022).

[10] Xiaoqi Ma, Yingzhan Xu, Xinfeng Zhang, Lv Tang, Kai Zhang, and Li Zhang. 2023. HM-PCGC: A Human-Machine Balanced Point Cloud Geometry Compression Scheme. In *IEEE International Conference on Image Processing*. IEEE, 2265–2269.

[11] 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).

[12] Anh Viet Phan, Minh Le Nguyen, Yen Lam Hoang Nguyen, and Lam Thu Bui. 2018. Dgcnn: A Convolutional Neural Network Over Large-scale Labeled Graphs. *Neural Networks* 108 (2018), 533–543.

[13] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *IEEE conference on computer vision and pattern recognition*. 652–660.

[14] Mateen Ulhaq. 2023. Learned point cloud compression for classification. In *IEEE International Workshop on Multimedia Signal Processing*. IEEE, 1–6.

[15] Mateen Ulhaq and Ivan V Bajić. 2024. Scalable Human-Machine Point Cloud Compression. *arXiv preprint arXiv:2402.12532* (2024).

[16] 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).

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

[18] Sukai Wang, Jianhao Jiao, Peide Cai, and Lujia Wang. 2022. R-pcc: A Baseline for Range Image-based Point Cloud Compression. In *International Conference on Robotics and Automation*. IEEE, 10055–10061.

[19] 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.

[20] 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

[21] 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.

[22] 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.

[23] 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

[24] 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.

[25] 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.

[26] Kang You, Pan Gao, and Qing Li. 2022. IPDAE: Improved Patch-based Deep Autoencoder for Lossy Point Cloud Geometry Compression. In *International Workshop on Advances in Point Cloud Compression, Processing and Analysis*. 1–10.

[27] Lili Zhao, Kai-Kuang Ma, Zhili Liu, and Qian Yin. 2022. Real-time scene-aware LiDAR point cloud compression using semantic prior representation. *IEEE Transactions on Circuits and Systems for Video Technology* 32, 8 (2022), 5623–5637.