# CHAPTER 6
# A NEW PROPOSED CASCADE DEEP LEARNING-BASED QUANTIZATION METHOD

**Mrs. A. CHRISTOPER TAMILMATHI**

*Hindustan College of Arts and Science, Chennai, India*

**Dr. PL. CHITHRA**

*University of Madras, Chennai, India*

**Mr. M. MADAN MOHAN**

*Nehru Institute of Engineering and Technology, India*

**Mr. S. VENKATESH**

*Nehru Institute of Engineering and Technology, India*

**Dr. P. K. MANOJ KUMAR**

*Nehru Arts and Science College, India*

## ABSTRACT

*This work presents a novel cascade deep learning-based spatial and color attributes codec for 3D LiDAR (Light detection and ranging) point clouds. This cascade deep learning model performs the codec operation on more than one attribute of 3D LiDAR point clouds. This proposed algorithm consists of three steps. The first step is to split the spatial and color attribute data from the original raw point cloud data. The second step is to enhance the raw 3D LiDAR point cloud (PCD) data by using the combination of Nyquist sub sampling and bi-normalization methods. Finally, apply the cascade deep learning-based quantization method to reduce the number of bits for transmission and storage purposes. The experimental results show that the proposed algorithm outperforms the existing PCD compression algorithm by compressing the PCD into constant 128 bytes without considering the original size of a PCD and increases the quality of the decompressed PCD approximately by 25% than the existing discrete wavelet transformation based PCD compression algorithm.*

*Keywords: — 3D LiDAR point cloud, bi-normalization methods, Nyquist subsampling, cascade deep learning etc*

## INTRODUCTION

The point cloud is a massive collection of 3D points with attributes generated from the 3D scanners and LiDAR. It is a base component of the 3 D object model. Each point in a PCD carries multiple attributes,


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

namely 3D coordinates (X, Y, Z), color attributes (R, G, B), counter, flight angle, time, direction, etc. Nowadays, all the real-time applications and industries are moving into 3D technology which is based on PCD [1] [2]. Single PCD can be formed by the millions or billions of 3D points with attributes. It is a very tedious process to manipulate, store the data, and transmit the data. This situation is insisted to develop the efficient PCD compression model with high execution speed and increase the utilization of the storage. In earlier, many of the traditional-based PCD compression algorithms have been developed and tested on the dataset. The 3D point cloud data has been converted into multiple 2D frames then a video compression algorithm is applied on the array of 2D data [2]. Many PCD compression algorithms are designed based on mathematical and signal-based computations by using tensor structure [3] [4]. The tensor block-wise singular value decomposition process helps to compress the geometry attribute in PCD [5]. In the 2D array of 3D point cloud data, considered as inter-frame and intra-frame structures to minimize the storage of the frame-based compression algorithm has been developed to improve the execution speed and avoid the redundant data [6]. The traditional-based PCD compression algorithm stated for compresses the geometry and color attributes by using the combination of plan fitting and the discrete wavelet transform method [7]. The loss conventional algorithm based on 2D manifolds on 3D point cloud space has been developed to compress the PCD data by comparing 2D patches in grid form [8].

In another way, a neural network-based point cloud compression model has been generated by using two different types of architecture for compressing the two different attributes namely, the geometry and texture of a point cloud [9]. An octree-based framework has been designed for static and dynamic point cloud compression processes. It compresses the point cloud data by using the advantages of the octree method and the voxel-based context method [10]. Learning–based compression PCD model has been developed based on the voxel convolution neural networks for the portioning process and the octree structure for eliminating the sparse values in the voxel blocks [11]. In recent days, the PCD compression algorithm has been designed to compress the augmented and virtual reality data in which the curvature-based hierarchical refining process algorithm segments the voxel PCD into high and low curvature points. Then, the video compression algorithm is applied to reduce the redundant data [12]. The performance of the 2D compression algorithm works better than the 3D PCD compression algorithm on some of the datasets [13].

A deep learning-based compression network has been designed to compress the color and reflectance attributes of a PCD. This learning-based algorithm reduces storage usage [14]. The deep learning quantization–based algorithm designed for airborne LiDAR PCD, was implemented and tested on different unbalanced datasets [15]. The compression algorithm is based on packing with path correlation improvement algorithm to eliminate the uncorrelated patch between correlated portions during the inter-prediction method [16]. One of the recent surveys on PCD cloud compression stated that the learning-based attributes compression approaches are considered for less research compared to geometry compression [17].

## CONTRIBUTIONS

The main contributions of this research work are as follows

- Bi-normalization Method: This proposed method reduces the computational complexity in time and storage. This method squeezes the 3D world coordinates values into the minimum range between [0.00, 0.01]. Hence it increases the speed of execution and utilization of memory storage than the well-known existing normalization methods produce the range between [0, 1].

- Cascade deep learning (CDL) method: This method compresses the multi-attributes of the PCD in a parallel manner, not like an existing learning-based and conventional-based PCD codec. The proposed parallel and learning-based model increases the speed of execution and utilization of memory by reducing the dimensionality of the multi-attributes of a PCD. It is a size-independent compression process.

The rest of the article has been organized as follows. The proposed CDL-based multi attribute codec has been explained in Section 2. The experimental results and analysis have been given in Section 3. Finally, the conclusion and the future work are mentioned in Section 4.

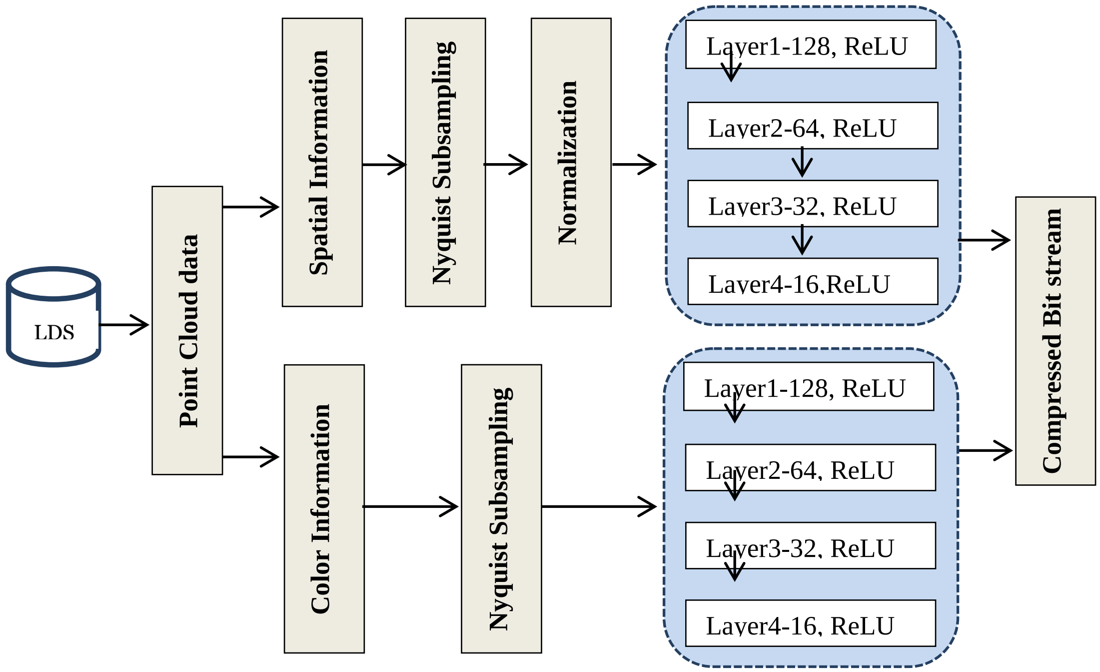



**Figure 1. The architecture of the proposed CDL based multi-attribute compression model.**

## PROPOSED METHODOLOGY

The overall Proposed point cloud compression architecture is shown in Fig.1. In this work, N numbers of unbalanced color point clouds are stored in the LiDAR dataset (LDS). The proposed CDL work consists of three steps to compress the point cloud data. First, spatial and color attributes are extracted from the raw point cloud then, the preprocessing methods such as Nyquist subsampling and bi-normalization methods are applied to enhance the raw spatial and color attributes for deep learning process. Finally, the normalized attributes are fed into the deep learning architectures to learn the dimensionality reduction process. The outcome of both deep learning architectures is cascaded together to form a compressed bitstream of the proposed CDL compression process. A detailed explanation of the proposed multi-attribute compression process has been given below.

Each PCD $P_i$ in the LDS has been denoted as $\{\boldsymbol{P_i}\}_{i=1}^{N}$ where N is a number of PCD in the dataset. Generally, PCD can be expressed by the following eqn. (1) [4].

$$P = \{P_i\}_{i=1}^{N} = \{(p_j): j = 1, 2, 3, \ldots, M\} \quad (1)$$

Where $P_i$, $p_j$ are i$^{th}$ point cloud and j$^{th}$ point in the point cloud respectively. Constant N, M is the number of point clouds in the dataset and the number of 3D points in a point cloud respectively. Each point $p_j$ can be as [4]

$$p = \Big(\big((X_M, Y_M, Z_M), [c], [a_1, a_2, \cdots, a_M]\big): X, Y, Z \in R, [c \in (R, G, B)], \big[a_i \in [0,1]\big]\Big) (2)$$

Where each point $p$ carries the attributes of 3D world coordinates$(X_M, Y_M, Z_M)$, color attributes (c), intensities, angle, count, etc. This research only concentrates on spatial and color attributes for


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

compression process in a parallel manner. First, the spatial and color attributes are extracted from each point ($p_j$) in a PCD ($P_i$).

Attribute1: Spatial: $p_S = \{(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), \cdots, (X_M, Y_M, Z_M)\}$ (3)

Attribute 2: Color: $p_C = \{(R_1, G_1, B_1), (R_2, G_2, B_2), (R_3, G_3, B_3), \cdots, (R_M, G_M, B_M)\}$ (4)

Where$p_S$,$p_C$, $M$ is the 3D spatial points, color values and number of 3D points in a point cloud respectively.

## NYQUIST SUBSAMPLING METHOD

The subsampling method plays a vital role in the learning algorithm to create a generalized model suitable for all data in the dataset. The LDS has the N number of massive unbalanced PCD data since the size of a PCD data always depends on the reflected pulses from the earth's ground object. These unbalanced PCD pulses are balanced by the Nyquist subsampling method without consideration of the size of a PCD.

The Nyquist subsampling method selects the point cloud signal based on the rate of signal interval (T/λ) where λ is a constant value 2[15]. The subsampled point cloud *P*in the dataset is denoted by the following Eqn. (5).

$$P = \{p_j, p_{j+2}, p_{j+4}, \cdots, p_K\} \quad (5)$$

Where $K$ is a number of subsampled 3D points which is less than $M$.

## BI-NORMALIZATION METHOD

The raw point cloud data consists of a large value of 3D world coordinates since it calculates from the distance between the sensor and the earth's surface object which is difficult to manage and do the computation. The proposed bi-normalization method scale down the word coordinates into the window coordinates, which further helps to reduce the complexity and time of the computational task than the existing normalization techniques. This process is squeezing the original data values into a particular specified range to involve all the data in the learning process. Many of the normalization processes are squeezing the data values into the range in-between [0, 1]. However, the proposed bi-normalization method scales the normalized value into half of the range [0, 1] for minimizes the computational complexity. This method not need the color attributes since it is always in the range of [0, 1]. The proposed bi-normalization method has been expressed by the following equations (6) and (9).

$$s_M = \max \ (p_S) \quad (6)$$

$$p_S' = \{(x_j, y_j, z_j)\}_{j=1}^{K} = \{((X_j, Y_j, Z_j)/s_M)/2\}_{j=1}^{K} (7)$$

Where$s_M$,$p_S'$is the maximum value of spatial coordinates and normalized coordinates respectively?

## CASCADE DEEP LEARINING METHOD

The proposed CDL method replaces the quantization process in the proposed compression model to reduce the storage size of PCD data. The normalized spatial coordinates ($p_S'$) and the subsampled color attributes ($p_C$) are fed into the two deep learning architectures for each. Both the deep learning (DL) processes have the same architecture. It consists of four dense layers with 128,64,32,16 neurons respectively;ReLU is an activation function for all the dense layers. The number of coordinates has been reduced based on the number of neurons in each layer. The final 16 neuron values from each two DL layers are cascaded (32 neurons) to form a compressed bit stream for the storage and transmission process. The quantization process using CDL in the compression method has been exhibited by the following equations.


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

$$\text{DL1: } Q_S = L_4\left(L_3\left(L_2\left(L_1(p_S^{'})\right)\right)\right) \quad (8)$$

$$\text{DL2: } Q_C = l_4\left(l_3\left(l_2\left(l_1(p_C)\right)\right)\right) \quad (9)$$

Where $Q_S$, $Q_C$,$L_i$,$l_i$is the quantized bitstream in spatial attributes, quantized bitstream in color attributes, $i^{th}$ layer in first deep learning (DL1), and $i^{th}$ layer in second deep learning (DL2) respectively. Both the DL models are working in a parallel manner. The optimized hyper parameters such as, epochs (1500), momentum (β=0.9), learning rate (α=0.3), loss function (mean squared error), and optimizer (SGD) for the DL are taken from [15]. The outputs of both DL models are cascaded to produce the compressed PCD bit stream ($Q$) for further storage and transmission process.

$$Q = cascade\{Q_S, Q_C\} \quad (10)$$

Figure 1 shows the compression process of the proposed algorithm. The decompression process is carried out by the inverse process of the compression method in which the four layers contain the 16, 32, 64, 128 neurons respectively.

## EXPERIMENTAL RESULTS

## SYSTEM CONFIGURATION

The proposed multi-attribute codec model has been implemented and tested on two inconsistent 3D Airborne LiDAR point cloud datasets by using the Python 3.7.3 in Jupiter environment on Windows 10 with an x64-based processor, 64-bit operating system, and 12 GB RAM. The performance of the proposed work compared with the existing DWT-based compression model[7].

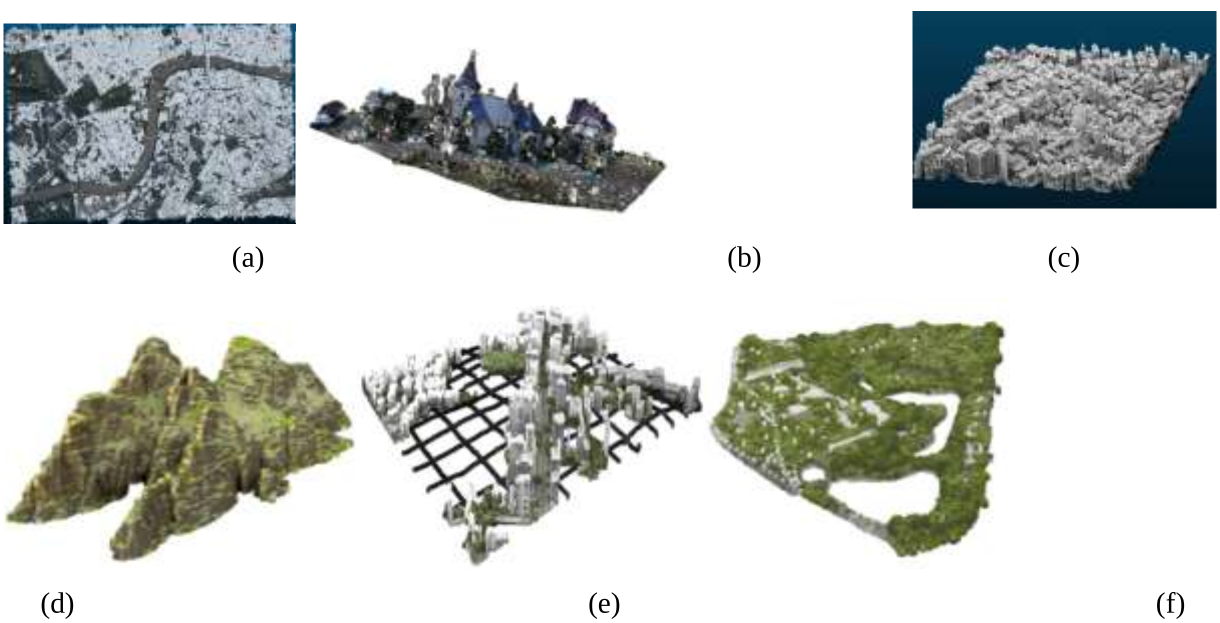



**Figure 2. Sample Point clouds from two different datasets. a) London test (B), b)Washingtonsq (B), c) Weglowka_v1, d) Skelling (L), e) Linabobardi (L), f) Ibirapuera (L).**

## DATASET


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

The suggested approach was tested on two 3D LiDARPCD datasets that were inconsistent, distinct, dense, and unlabeled, such as the Buildings dataset (B) and the Landscapes dataset (L). The Landscape dataset contains seven large landscapes PCD. The Building dataset contains eight massive buildings PCD. These two datasets were transformed into PCD format by utilizing the CloudCompare tool to work on 3D points and color properties. The PCD images in the datasets were separated into two processes: training (80%) and testing (20%). Each PCD in the datasets has a large number of unbalanced 3D points. Figure 2 depicts some of the sample point clouds in the datasets.

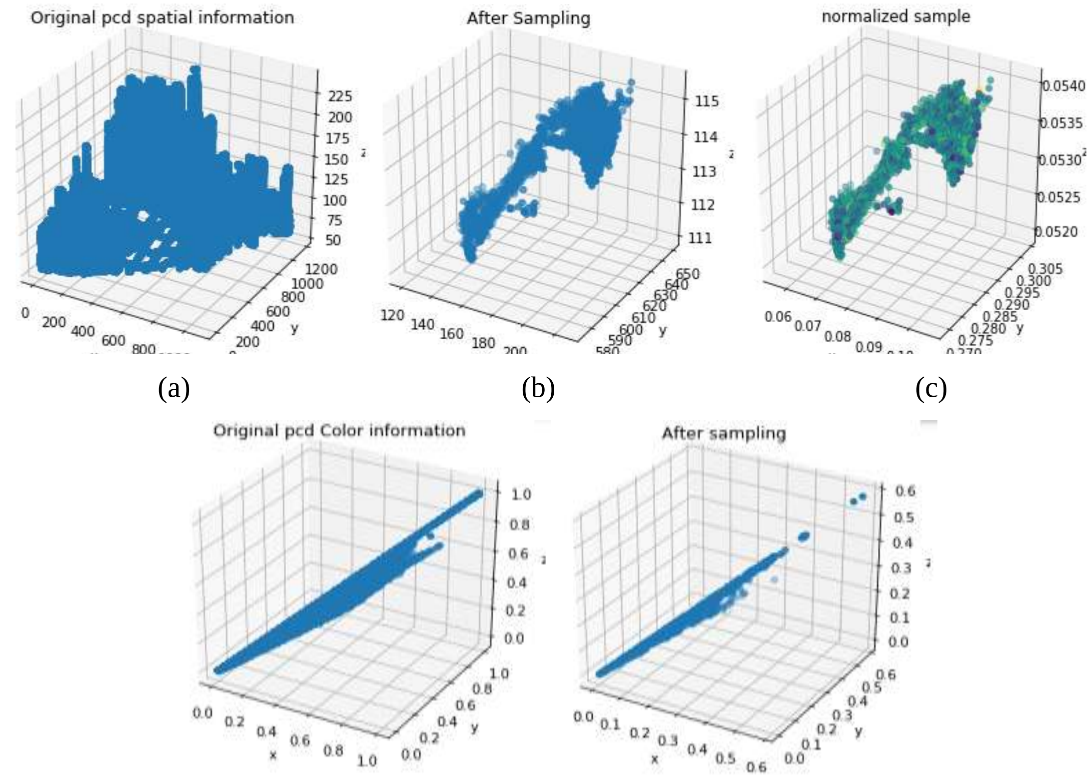
The description of the point clouds in the LDS has been shown in Table 1. The proposed PCD compression has been carried out in three steps. First, the 3D coordinates and color attributes are extracted from the PCD data then, the preprocessing methods such as Nyquist sub sampling and bi-normalization are applied to both attributes for cleansing the data for further processing. Finally, the normalized spatial and color information are fed into the two DL architectures for the quantization process. The output of the two DL models is cascaded to produce the compressed bit stream.

**Table 1. Description of the original PCD in Buildings and Landscapes LiDARdataset.**

| **Dataset Name** | **PCD Name** | **Size in KB** | **Number of Points** | **After Sub sampling** |
|---|---|---|---|---|
| Buildings(B) | Autzen - Cloud | 104,037 | (5326668,3) | (2048,3) |
| | Building | 76,156 | (3899161,3) | (2048,3) |
| | ChristChurch | 39,328 | (1677960,3) | (2048,3) |
| | London Test | 83,562 | (4278353,3) | (2048,3) |
| | Szymbarkmodel | 75,157 | (2404995,3) | (2048,3) |
| | UFO_2735 | 102,383 | (5241988,3) | (2048,3) |
| | Washingtonsq | 47,528 | (2433393,3) | (2048,3) |
| | Weglowka_v1 | 135,619 | (4339794,3) | (2048,3) |
| Landscapes (L) | Arizona sunset crater | 21,106 | (1080597, 3) | (2048,3) |
| | Fortvechten | 43,249 | (1383949, 3) | (2048,3) |
| | Ibirapuera | 92,674 | (4744881, 3) | (2048,3) |
| | Kinderdijk | 68,516 | (2923312, 3) | (2048,3) |
| | Linabobardi | 102,466 | (3278903, 3) | (2048,3) |
| | Mine near rigalavatia | 56,853 | (2910835, 3) | (2048,3) |
| | Skelling | 71,706 | (3671337, 3) | (2048,3) |

Figure 2 shows the scatter plot representation of output from the proposed preprocessing methods on one of the input sample PCD “Kinderdijk”. The original, subsampled, normalized spatial information of “Kinderdijk” PCDis shown in Figures 3a, 3b, and 3c respectively.

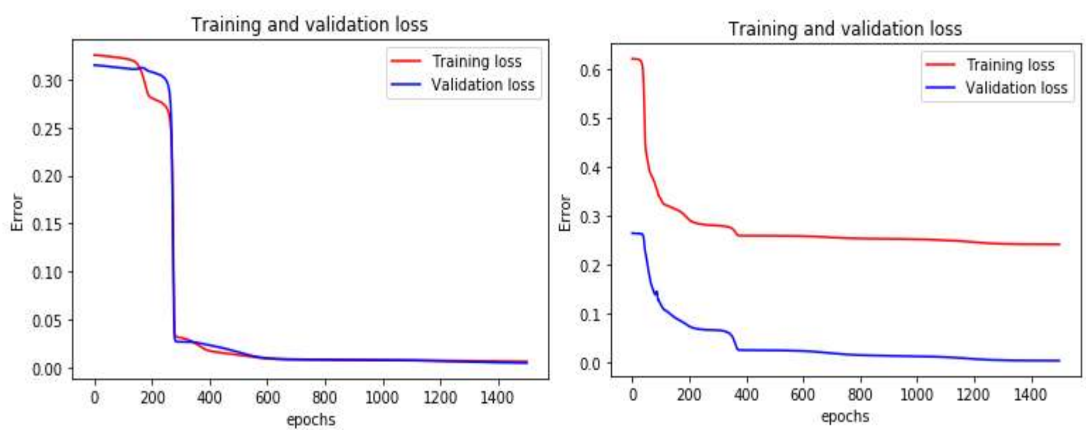

ISBN : 978-93-5782-920-5 , www.aetsjournal.com “ Handbook of research in Big data Analytics, Artificial intelligence and Machine learning”




(a) (b) (c)

(d) (e)

**Figure 3. Scatter plot representation of the spatial and color information of sample "Kinderdijk" PCD from the proposed preprocessing methods. a) Original PCD spatial information, b) Subsample spatial data, c) Normalized spatial data, d) original PCD color information, e) Subsample color data.**




(a) (b)

**Figure 4. Graphical representation of training and validation process. a) Buildings dataset, b) Landscapes dataset.**


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

The original, subsample color information of “Kinderdijk” PCD is shown in Figures 3d, and 3e respectively. After the Nyquist subsampling method, the number of 3D points in each PCD is sampled into (2048×3) size without considering the original size of a PCD shown in Table 1. Then, normalized PCD data are fed into the corresponding DL model for the learning process of quantization. The optimized hyper parameters such as, epochs (1500), momentum ($\beta$=0.9), learning rate ($\alpha$=0.3), loss function (mean squared error), and optimizer (SGD) for the DL are taken from [15]. The graph of training and the validation process of both datasets are shown in Fig. 4.

From Fig.4, it was observed that both validation processes, the error value reaches zero before reaching the maximum epoch value due to the optimized hyper parameters. The output of the cascaded DL models is treated as a compressed bit stream that is 32 neurons (32×4 bytes=128 bytes). During the decompression process, the compressed bit stream is included for the inverse process of compression to produce the decompressed PCD same as in the original PCD.


**Figure 5. Sample targeted and actual spatial and color attribute data of sample “Skelling” PCD from the proposed CDL model. a)Targeted spatial data, b) Targeted color attribute data, c) Actual output spatial data, d) Actual output color attribute data.**

One of the samples targeted and actual data of tested “Skelling” PCD from the proposed PCD compression model is shown in Fig. 5. The expected output of spatial and color attributes is displayed in Figures 5a and 5b, the actual output from the proposed model is depicted in Figures 5c and 5d. From this Fig 5, is observed that there is no noticeable difference in between expected and actual output of both the attributes. Hence, this proposed CDL-based multi-attribute model is suitable for the 3D LiDAR PCD codec process.

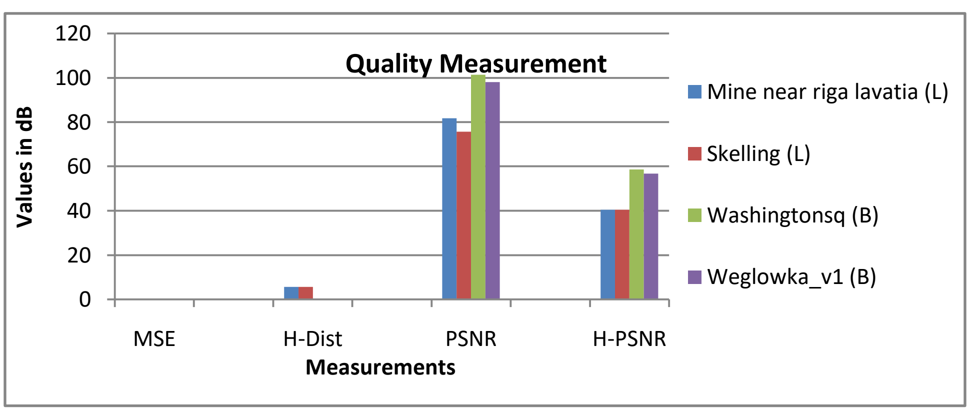



**Figure 6. Graphical representation of the quality metrics between the original and decompressed PCD from the proposed compression model.**


ISBN : 978-93-5782-920-5 , www.aetsjournal.com “ Handbook of research in Big data Analytics, Artificial intelligence and Machine learning”

The quality metrics of point cloud compression namely mean squared error (MSE), peak signal-to-noise ratio (PSNR), Hausdorff distance (H-Dist.), Hausdorff PSNR(H-PSNR) methods are applied to original and decompressed PCD data to measure the performance of the proposed CDL-based multi-attribute codec[18]. The observed measurement values are depicted in Fig. 6. From this graphical illustration, it was observed that the unnoticeable error in-between the original and the decompressed PCD and increases the quality (PSNR) of the decompressed PCD as in the original one.

The performance of the proposed PCD compression model compared with the conventional PCD compression model based on the discrete wavelet transform (DWT) method with their input photogrammetric point clouds[7] and tabulated in Table 2.

**Table 2. The performance comparison of proposed PCD compression model with existing DWT based PCD compression** [7]**.**

| PCD Name | Compression Algorithm | MSE | H-Dist. | PSNR | H-PSNR |
|---|---|---|---|---|---|
| Banana | Existing DWT | 0.02 | 0.1 | 48.36 | 52.8 |
| | Proposed CDL based Codec | **3.82E-08** | 0.032 | **122.3** | 63.02 |
| Milk | Existing DWT | 0.01 | 0.2 | 39.5 | 48.1 |
| | Proposed CDL based Codec | 1.1 e-05 | **0.01** | 97.5 | **65.3** |
| Plant | Existing DWT | 0.018 | 0.2 | 41.14 | 48.1 |
| | Proposed CDL based Codec | 0.05 | 1.1 | 60.8 | 47.4 |

From the above table, it was observed that the proposed work produces a minimum error and maximum quality of the decompressed PCD than the existing traditional algorithm [7].

From Figures 3, 4, 5, 6, and Tables 1, 2 we concluded that the proposed CDL-based multi-attribute compression method outperforms well with the existing DWT-based conventional algorithm.

## CONCLUSION

This research work presents a novel cascade deep learning-based multi-attribute codec for 3D LiDAR point clouds. This work concentrates to compress the important attributes, namely spatial 3D coordinates and color attributes (R, G, B) in PCD data. It consists of three processes to accomplish the compression task. Those are extraction processes to split the coordinate values and color values from each point in a PCD. Then, the preprocessing processes, namely Nyquist sub sampling and bi-normalization methods enhance the PCD data for the cascade deep learning process. Finally, the storage space of the spatial and color attribute values is reduced by the CDL model. The output of the CDL model is considered a compressed bit stream. This proposed work has been implemented and tested by two dense PCD datasets namely, Buildings (B) and Landscapes (L). The experimental results show that the proposed algorithm outperforms the existing PCD compression algorithm by compressing the PCD into constant 128 bytes without considering the original size of a PCD and increasing the quality of the decompressed PCD approximately by 25% than the existing discrete wavelet transformation based PCD compression algorithm. In the future, the learning-based model will be developed to compress the remaining attributes of a point cloud.


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

## REFERENCE

[1] "What is lidar data?—Help | ArcGIS Desktop." https://desktop.arcgis.com/en/arcmap/10.3/manage-data/las-dataset/what-is-lidar-data-.htm (accessed Jan. 10, 2021).

[2] X. Sun, S. Wang, M. Wang, Z. Wang, and M. Liu, "A Novel Coding Architecture for LiDAR Point Cloud Sequence," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5637–5644, 2020, doi: 10.1109/LRA.2020.3010207.

[3] P. L. Chithra and A. C. Tamilmathi, "Tensor Tucker Decomposition based Geometry Compression on Three Dimensional LiDAR Point Cloud Image," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 3, pp. 1897–1903, 2020, doi: 10.35940/ijitee.c8551.019320.

[4] P. L. Chithra and A. C. Tamilmathi, "3D LiDAR point cloud image codec based on Tensor," *Imaging Sci. J.*, vol. 68, no. 1, pp. 1–10, 2020, doi: 10.1080/13682199.2020.1719747.

[5] A. C. Tamilmathi and P.L . Chithra, "Tensor block - wise singular value decomposition for 3D point cloud compression," *Multimed. Tools Appl.*, no. 0123456789, 2022, doi: 10.1007/s11042-021-11738-7.

[6] Y. Feng, S. Liu, and Y. Zhu, "Real-time spatio-temporal LiDAR point cloud compression," *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 10766–10773, 2020, doi: 10.1109/IROS45743.2020.9341071.

[7] P. L. Chithra and T. A. Christoper, "3D Color Point Cloud Compression with Plane fitting and Discrete Wavelet Transform," *2018 10th Int. Conf. Adv. Comput. ICoAC 2018*, pp. 20–26, 2018, doi: 10.1109/ICoAC44903.2018.8939106.

[8] M. Quach, G. Valenzise, and F. Dufaux, "Folding-Based Compression of Point Cloud Attributes," *Proc. - Int. Conf. Image Process. ICIP*, vol. 2020-October, pp. 3309–3313, 2020, doi: 10.1109/ICIP40778.2020.9191180.

[9] E. Alexiou, K. Tung, and T. Ebrahimi, "Towards neural network approaches for point cloud compression," p. 4, 2020, doi: 10.1117/12.2569115.

[10] Z. Que, G. Lu, and D. Xu, "VoxelContext-Net: An Octree based Framework for Point Cloud Compression," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 6038–6047, 2021, doi: 10.1109/CVPR46437.2021.00598.

[11] D. T. Nguyen, M. Quach, G. Valenzise, and P. Duhamel, "Learning-based lossless compression of 3D point cloud geometry," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 2021-June, pp. 4220–4224, 2021, doi: 10.1109/ICASSP39728.2021.9414763.

[12] S. Yu, S. Sun, W. Yan, G. Liu, and X. Li, "art22.pdf." Sensors,MDPI, pp. 1–18, 2022, [Online]. Available: https://doi.org/10.3390/s22031262.

[13] F. Nardo, D. Peressoni, P. Testolina, M. Giordani, and A. Zanella, "Point Cloud Compression for Efficient Data Broadcasting: A Performance Comparison," 2022, [Online]. Available: http://arxiv.org/abs/2202.00719.

[14] G. Fang, Q. Hu, H. Wang, Y. Xu, and Y. Guo, "3DAC: Learning Attribute Compression for Point Clouds," 2022, [Online]. Available: http://arxiv.org/abs/2203.09931.

[15] A. C. Tamilmathi and P. L. Chithra, "Deep Learned Quantization-Based Codec for 3D Airborne LiDAR Point Cloud Images," *Front. Robot. AI*, vol. 8, May, pp. 1–10, 2021, doi: 10.3389/frobt.2021.606770.


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"

[16] H. Shi and F. Li, "Patch Re-segmentation and Packing for Dynamic Point Cloud Compression via Back-and-Forth Structure," *IEEE Open J. Signal Process.*, vol. 3, November 2021, pp. 155–168, 2022, doi: 10.1109/OJSP.2022.3160392.

[17] M. Quach *et al.*, "Survey on Deep Learning-based Point Cloud Compression To cite this version : HAL Id : hal-03579360 Survey on Deep Learning-based Point Cloud Compression," 2022.

[18] G. Meynet *et al.*, "PC-MSDM : A quality metric for 3D point clouds To cite this version : HAL Id : hal-02162104 PC-MSDM : A quality metric for 3D point clouds," no. QoMEX, pp. 3–6, 2019.

Dr.PL.Chithra is working as a professor in the Department of Computer Science, University of Madras. Main areas of research include 3D Digital Image Processing, Pattern Recognition, Artificial Intelligence, Signal Processing, Network Security, and Video Processing. She received Ph.D. in titled Multiresolution based image codecs for Lossy/Lossless compression, from the University of Madras and M.C.A. degree from the Alagappa University, Tamil Nadu, India. She is passionate on teaching, imparting, training, motivating and helping of young minds with knowledge and good discipline. She has 30 years of teaching and 20 years of research experience in Computer Science. Serving an Organizing Chair, Program Chair and Program Committees of several national and International conferences. She has awarded UGC FIP program for two years, conducted and act as a resource person for several refresher courses. She published more than 96 papers in National and International reputed journals with 70 citations and 5 hindex. Serving as one of the Computer Science staff selection committee members, BOS member and inspection committee member for various affiliated colleges of University of Madras since 2000.


ISBN : 978-93-5782-920-5 , www.aetsjournal.com " Handbook of research in Big data Analytics, Artificial intelligence and Machine learning"