International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

Utilization of Deep Learning for LiDAR Point Cloud Classification

[1]Shahlaa Falah Hasan Al-Tameemi, [2]Alaa Ali Ghaith, [3]Ahmad Ghandour

[1,3]Department of Computer and Communications, Faculty of Engineering, Islamic University of Lebanon, Wardanieh, Lebanon
[2]Faculty of Sciences 1, Lebanese University, Beirut, Lebanon

***Abstract* - With the increasing availability of LiDAR and comparable sensor technologies, the accessibility of point cloud data has significantly improved. Nevertheless, the magnitude and intricacy of point cloud data pose challenges for processing and interpretation. A point cloud classification model utilizing deep learning is proposed as a solution to the difficulties encountered in evaluating point cloud data, primarily acquired via LiDAR and comparable sensor technologies. The utilization of data augmentation technologies and meticulous preprocessing enhances the efficiency of the Point-Net model in processing and categorizing point cloud data. Empirical findings validate the significance of optimizing hyperparameters, such as number of epochs, batch size, and learning rate, in order to enhance the performance of the model. The enhanced Point-Net model demonstrated a notable enhancement in classification accuracy, reaching a maximum accuracy of 0.7097, which is a substantial improvement compared to the initial performance.**

Keywords: LiDAR, Deep Learning, Point-Net Network, Fine-Tuning Hyperparameter, Point Cloud Classification.

I. INTRODUCTION

The widespread adoption of LiDAR and comparable sensor technologies has resulted in a significant rise in the accessibility of point cloud data in recent years [1]. The potential uses of these data sets are vast, encompassing fields such as urban planning, infrastructure management, autonomous driving, and environmental monitoring. Nevertheless, the large quantity and intricate nature of point cloud data pose considerable obstacles for effective analysis and interpretation.

Conventional approaches to handling point cloud data typically depend on manually extracting features and classifying them, which is demanding in terms of labor, time, and susceptible to mistakes. Deep learning approaches, such as convolutional neural networks (CNNs) and point-based models like Point-Net, are highly effective tools for automated feature learning and classification tasks in the processing of point cloud data [2]. The Point-Net model, proposed by Qi et al. in 2017, brought about a significant change in point cloud processing [3]. It achieved this by simply working with unordered point sets, hence avoiding the requirement for computationally intensive data alignment. Nevertheless, the original Point-Net model, while useful, has constraints when it comes to dealing with fluctuations in point cloud data, including noise, occlusions, and changes in viewpoint. Researchers have suggested adding attention mechanisms, graph-based convolutions, and hierarchical feature learning modules to Point-Net to address these issues [4]. Data augmentation improves the robustness and generalization of deep learning models trained on point cloud data.

The aim of this study is to directly address these difficulties and improve the accuracy and reliability of point cloud classification using deep learning. Point-Net does not apply convolutions directly to the data. Instead, it focuses on individual points within the point cloud and treats each point as an independent entity thus being able to effectively capture spatial relationships and patterns without relying on traditional CNN procedures. This research is a major advance in the development of automated point cloud classification and highlights the importance of interdisciplinary collaboration across computer vision, robotics, and geospatial science in tackling challenging real-world problems.

II. RELATED WORK

The development of deep learning techniques for analyzing point clouds is constantly progressing, fueled by the growing abundance of data, advancements in computer power, and the need for more precise and efficient solutions in several fields. Projection-based methods necessarily face degradation of spatial information. In addition to using point-based grids. In [5] a structure-aware approach to address this problem is presented. An auxiliary network is used to actively supervise the convolutional layers in order to preserve structural information. The auxiliary network converts the convolutional features obtained from the basic network into point-level representations. This process is improved in a collaborative manner. Once the training phase is complete, the auxiliary network can be removed to speed up inference. A method for automatic point cloud classification was developed in [6] that is based on advanced deep learning techniques and includes a

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

special algorithm for transforming point cloud data into a structured matrix, combined with a specially designed convolutional neural network architecture.

In [7], a four-layer MLP architecture, is presented for learning embedded point features. However, when it comes to elements that lack distinct characteristics, such as shrubs or trees, their specific spatial arrangement is not fully exploited. In order to make greater use of the abundant spatial information of objects, spatial pooling as a method of acquiring point properties was introduced in [8]. The input data was divided into groups, and then a spanning tree-based clustering technique was used to extract spatial information between points in the clustered point groups. Finally, MLP was used to perform classification using features. In [9] the similarity group-SGPN was proposed to accomplish several tasks, including instance segmentation and object detection, using an explicit architecture. The Point-Net model extracted local and global point features, resulting in a matrix. This matrix was then divided into three subsets, each of which was processed by a separate Point-Net layer to provide three similarity matrices. The similarity matrix, confidence map, and semantic segmentation map were generated using these three matrices. Point-Net lacks the ability to generate local features for adjacent points, but Point-Net++ [10] incorporates a class pyramid feature aggregation method. The system consists of three vertically arranged layers: the sample layer, the grouping layer, and the Point-Net layer. Point-Net++ utilizes a hierarchical approach, akin to conventional picture learning, to extract features, hence minimizing the loss of local information.

The performance of a CNN based image classification strategy was evaluated in [11]. The objective was to classify 3D point clouds of seven tree species using a 2D representation in a computationally efficient manner.In [12], airborne LiDAR data to segment individual trees in a thick natural forest was utilized. The segmented crowns were then used as input for a CNN to perform tree species categorization. Various shallow and deep learning experiments were conducted as part of this analysis. A multitude of researchers have directed their attention towards the distinctive geometric properties of point clouds. By harnessing the power to comprehend intricate hierarchical arrangements, deep learning has attained remarkable triumph in utilizing images obtained from cameras. The categorization of existing point cloud feature learning methods can be divided into two types: point-based and tree-based [13]. The initial point cloud is immediately used as input for deep learning. The second employs a k-dimensional tree structure to depict the point cloud using a standardized representation. Subsequently, these representations are inputted into deep learning models.

III. METHODOLOGY

Deep learning methods will be used to increase the accuracy of object detection within spatial datasets and reveal the complexities of point cloud classification. Point-Nets will be used to process point cloud data obtained through LiDAR analysis. The methodology shown in the Figure 1 starts by loading a set of diverse point cloud data and to make the data set more robust and diverse, data augmentation techniques will be used. Then comes the pre-processing stage of the data to ensure that it is consistent with the structure of the model. Next, the structure of the network point model is determined which includes setting up the network layers and activation functions, then setting the training parameters, feeding the training data, and adjusting the parameters.

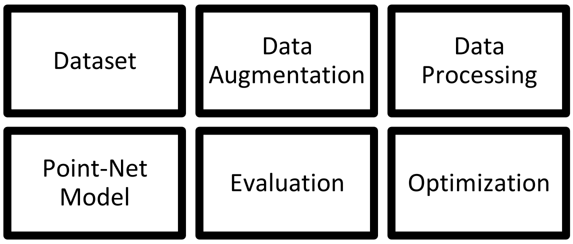

Figure 1: Proposed methodology for point cloud classification

The model is evaluated using quantitative and empirical methodologies. In the quantitative section, accuracy, precision, recall, and mean Average Precision (mAP) evaluate the performance of the model on the dataset. Experimental procedures are applied that modify the Point-Net architecture settings such as era, batch size, and learning rate to enhance the classification results.

3.1 Dataset

The Sydney Urban Objects dataset is a rigorously prepared collection of point cloud data captured using lidar sensors. It has been curated by the University of Sydney and consists of 100 identified objects over 14 various categories, such as vehicles, pedestrians, and buses. Figure 2 shows some samples from the dataset.

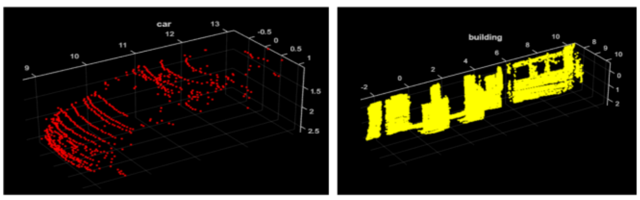

Figure 2: Samples from dataset (Left is a car, Right is a building)

By counting the number of points assigned to each label, a deeper understanding of the label distribution within the dataset is gained as shown in Figure 3. A preliminary analysis

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

reveals an imbalance that is biased towards categories such as cars and pedestrians.

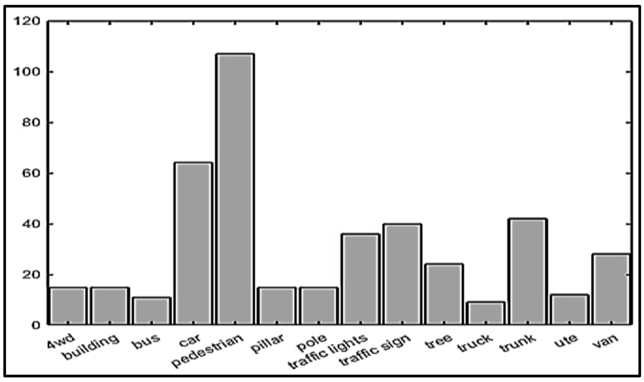

Figure 3: Data distribution in the dataset

3.2 Data Augmentation

In order to address the issue of data imbalance, data augmentation techniques will be employed. The point cloud transformation and augmentation functions will be utilized to apply a range of modifications to enhance the point cloud data. These transformations include the following changes:

- The point cloud undergoes random rotation, which creates variations in the orientation of objects.
- Removing the random point to create missing data, forcing the model to learn from partial information.
- Random perturbation of points using Gaussian noise to simulate the presence of noise in the process of collecting data from the real world.

By randomly applying these changes to the training data, the dataset's variety is effectively enhanced without introducing any duplicate samples. This aids in mitigating overfitting and enhances the model's capacity to generalize to new data.

3.3 Data Processing

A certain number of points were chosen from each point cloud in order to make the process of batch processing during training more manageable. In order to ascertain this, the maximum and average number of points for each category in the data set was computed, and the results are presented in Table 1.

An experimental investigation revealed that selecting 1024 points strikes a compromise between properly capturing object shapes and economically managing computer resources. This was discovered through the use of experimental analysis. The selection of 1024 points from each point cloud was accomplished with the help of a transform function. Point cloud data values show significant variation due to object distances from lidar sensors. Objects near the sensor may have different values than those farther away, leading to disparities that can impair network convergence. Normalization (converting point cloud data to a range from 0 to 1 and formatting it for the point model) solved the problem.

Table 1: Statistics of points of the dataset classes

	classes	numObservations	minPointCount	maxPointCount	meanPointCount
1	4wd	15	140	1955	751
2	building	15	193	8455	2708
3	bus	11	126	11767	2190
4	car	64	52	2377	528
5	pedestrian	107	20	297	110
6	pillar	15	80	751	357
7	pole	15	13	253	90
8	traffic lights	36	38	352	161
9	traffic sign	40	18	736	126
10	tree	24	53	2953	470
11	truck	9	445	3013	1376
12	trunk	42	32	766	241
13	ute	12	90	1380	580
14	van	28	91	5809	1125

3.4 Point-Net Model

Point-networks were chosen because they can handle point cloud data and record spatial relationships efficiently. Point-Net networks employ a set of multi-layer perceptron (MLPs) to individually evaluate each point, extracting distinctive characteristics. These features are subsequently aggregated across all points to obtain a comprehensive representation of the entire point cloud. This comprehensive depiction is subsequently inputted into additional MLP layers for the purpose of classification or segmentation tasks. The Point-Net classification approach has two essential components for effectively identifying point cloud data. The point cloud encoder converts sparse point cloud data into a dense feature vector. Second, the classifier predicts each encoded point cloud's categorization class.

Point-Net Encoder: The central component of the model is the point cloud encoder, which is responsible for converting the sparse input data (a collection of 3D points) into a compact feature vector that is suited for classification tasks. The encoder consists of four sub-models:

- Input Transform Model: It makes the point cloud more orientation-resistant by applying a transformation to each point.
- Shared MLP-1: Processes input data using convolution, batch normalization, and ReLU.
- Shared MLP-2: Convolutions, batch normalization, and ReLU activations extract shared features from each point. This layer's weights are shared across all points.
- Feature Transform Model: Like the input transform, this model learns another affine transformation for each point to align them in feature space for better feature extraction.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

Ultimately, a max pooling operation consolidates the encoded features from all points, yielding a solitary dense feature vector that represents the entirety of the point cloud. The table-- includes the number of input channels and hidden channel sizes.

Table 2: Encoder Model Parameters

	Input channel size	Hidden channel size
Input Transform Model	3	64, 128, 256
Shared MLP-1	3	64
Shared MLP-2	64	64, 128, 256
Feature Transform Model	64	64

Point-Net Classifier: Classifier maps encoded feature vectors from encoders to class labels. The input size is set to 64 and the hidden channel sizes are set to 512 and 256. It's essential elements:

- Final shared MLP layer processes encoded features to retrieve classification-relevant high-level information.
- Fully Connected Layer: Flattens the preceding layer's output into a vector for final categorization.
- SoftMax Activation: This function probabilism each class to forecast the input point cloud's most likely category.

A specialized training loop was established to regulate the training process, and prior to each epoch, the complete dataset was rearranged to guarantee random sampling throughout training. The loss gradients were computed and L2 normalization was employed to discourage the presence of excessively large weights. The Adam optimizer was employed to modify the model parameters in accordance with the computed gradients.

3.5 Evaluation

This section shows the evaluation results of the pre-trained Point-Net model. Figure 4, shows the validation confusion matrix.

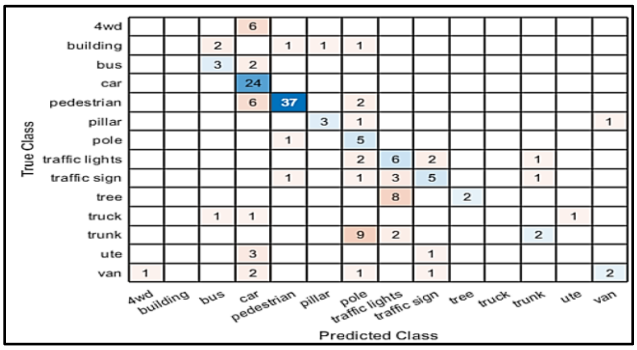

Figure 4: Validation confusion matrix

The accuracy, precision, and recall metrics were calculated based on the confusion matrix. According to the evaluation results, the pre-trained model's accuracy was unsatisfactory, with a recorded value of 0.5742, suggesting a significant inadequacy. This limitation becomes most apparent when assessing the effectiveness of the model on new, unknown input data. Figure 5 presents a bar graph illustrating the precision and recall values for each class.

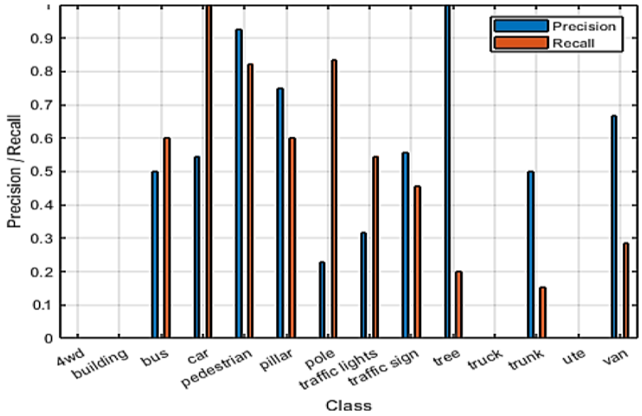

Figure 5: Precision and recall of each class

3.6 Optimization

Tuning critical hyperparameters, such as the number of epochs, batch size, and learning rate, enhances the performance of the model. The specifics of this optimization procedure are displayed in Table 3.

Table 3: Accuracy of the tested hyperparameter

num of epoch	Batch Size	learning rate	accuracy
50	32	0.001	0.6323
		0.002	0.6516
		0.003	0.6452
	128	0.001	0.6452
		0.002	0.6452
		0.003	0.6774
	512	0.001	0.671
		0.002	0.6
		0.003	0.6387
100	32	0.001	0.6452
		0.002	0.6774
		0.003	0.6774
	128	0.001	0.6645
		0.002	0.671
		0.003	0.6645
	512	0.001	0.6839
		0.002	0.671
		0.003	0.7032
500	32	0.001	0.7097
		0.002	0.6903
		0.003	0.6452
	128	0.001	0.6
		0.002	0.6581
		0.003	0.7032
	512	0.001	0.6581
		0.002	0.6516
		0.003	0.6387

As shown in the results, after completing the custom training phase and tuning the hyperparameters across different

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

values, optimal accuracy of the Point-Net model was achieved the highest accuracy of 0.7097 was achieved when using 500 epochs, a batch size of 32, and a learning rate of 0.001. Figure 6 represents the validation confusion matrix and Figure 7 represents precision and recall values for each class after fine-tuning hyperparameter.

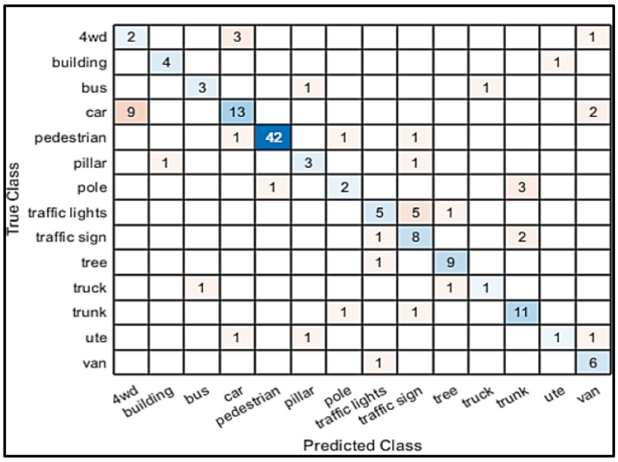

Figure 6: Validation confusion matrix after fine-tuning hyperparameter

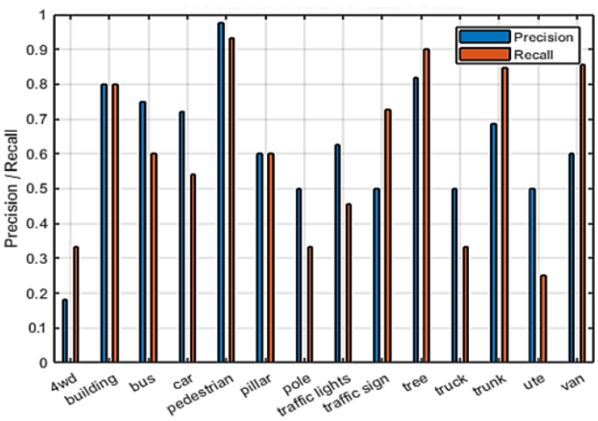

Figure 7: Precision and recall of each class after fine-tuning hyperparameter

In overall, the findings of this study emphasize the significance of customized training and fine-tuning of hyperparameters to optimize the performance of deep learning models for point cloud categorization.

IV. CONCLUSION

This study provides significant progress in the field of point cloud classification by utilizing the implementation of Point-Net model. The results confirm the effectiveness of this method in addressing inherent difficulties associated with LiDAR-generated point cloud data, such as sparse distribution, interference susceptibility, and non-uniform arrangements. The effectiveness of the Point-Net model in processing and classifying point cloud data was enhanced by the proposed methodology, which involved robust data augmentation and comprehensive preparation. The experimental findings emphasize the significance of optimizing model performance by adjusting hyperparameters, such as the number of epochs, batch size, and learning rate. The optimized Point-Net model exhibited a significant increase in classification accuracy, reaching a maximum accuracy of 0.7097, which is a noteworthy improvement compared to its initial performance. This enhancement was crucial in rectifying the early deficiencies of the model, specifically in categorizing novel and unfamiliar input data. The Point-Net architecture, with its distinctive treatment of individual points and capacity to capture spatial correlations, shown its resilience as a framework for point cloud classification tasks. This research not only enhances the technical progress in the domain of point cloud data processing but also paves the way for future investigations into more intricate and varied applications, such as autonomous navigation, 3D mapping, and urban planning.

REFERENCES

[1] Ding. Z, Sun. Y, Xu. S, Pan. Y, Peng. Y, and Mao. Z, “Recent Advances and Perspectives in Deep Learning Techniques for 3D Point Cloud Data Processing.” *Robotics*, vol. 12, no. 4, 2023.

[2] Yao. X, Guo. J, Hu. J, and Cao. Q “Using deep learning in semantic classification for point cloud data.”*IEEE Access*, vol. 7, pp. 37121-37130, 2019.

[3] Qi. C. R, Su. H, Mo. K, and Guibas. L. J, “Pointnet: Deep learning on point sets for 3d classification and segmentation.”in Proc. *IEEE conference on computer vision and pattern recognition*, pp. 652-660, 2017.

[4] Khan. S. A, Shi. Y, Shahzad. M, and Zhu. X. X, “FGCN: Deep feature-based graph convolutional network for semantic segmentation of urban 3D point clouds.”in Proc. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*”, pp. 198-199, 2020.

[5] He. C, Zeng. H, Huang. J, Hua. X. S, and Zhang. L, “Structure aware single-stage 3d object detection from point cloud.” in Proc.*IEEE/CVF conference on computer vision and pattern recognition*, pp. 11873-11882, 2020.

[6] Dominik. W, Bożyczko. M, and Tułacz-Maziarz. K, “Deep learning for automatic LiDAR point cloud processing, 2021.

[7] Zhang. L., Li. Z, Li. A, and Liu. F,“Large-scale urban point cloud labeling and reconstruction,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 138, pp. 86-100, 2018.

[8] Z. Wang, L. Zhang, L. Zhang, R. Li, Y. Zheng and Z. Zhu, "A Deep Neural Network With Spatial Pooling (DNNSP) for 3-D Point Cloud Classification," in *IEEE*

ISSN (online): 2581-3048

Volume 8, Issue 3, pp 167-172, March-2024

https://doi.org/10.47001/IRJIET/2024.803022

Transactions on Geoscience and Remote Sensing, vol. 56, no. 8, pp. 4594-4604, Aug. 2018.

[9] W. Wang, R. Yu, Q. Huang and U. Neumann, "SGPN: Similarity Group Proposal Network for 3D Point Cloud Instance Segmentation," *IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA*, pp. 2569-2578, 2018.

[10] Qi, C. R., Yi, L., Su, H., and Guibas, L. J, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space."*Advances in neural information processing systems*, 2017.

[11] Seidel, D., Annighöfer. P., Seifert. Q. E., and Ammer. C. "Predicting tree species from 3D laser scanning point clouds using deep learning."*Frontiers in Plant Science*, 2021.

[12] Hamraz. H, Jacobs. N.B, Contreras, Marco. A, Clark and Chase. H, "Deep learning for conifer/deciduous classification of airborne LiDAR 3D point clouds representing individual trees." ISPRS Journal of *Photogrammetry and Remote Sensing*, vol. 158, pp. 219-230, 2019.

[13] Liu. W, Sun. J, Li. W, Hu. T and Wang. P. "Deep Learning on Point Clouds and Its Application: A Survey."*Sensors (Basel)*, vol. 19, no. 19, 2019.

Citation of this Article:

Shahlaa Falah Hasan Al-Tameemi, Alaa Ali Ghaith, Ahmad Ghandour, "Utilization of Deep Learning for LiDAR Point Cloud Classification" Published in *International Research Journal of Innovations in Engineering and Technology* - *IRJIET*, Volume 8, Issue 3, pp 167-172, March 2024. Article DOI https://doi.org/10.47001/IRJIET/2024.803022

****