Received: March 25, 2025. Revised: May 11, 2025.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Human Pose Prediction Using Convolution Neural Network 3D based on Binary Voxel Feature Extraction (BIVFE) of Light Detection and Ranging (LiDAR) Point Cloud Data

Farah Zakiyah Rahmanti[1,2] Moch. Iskandar Riansyah[1,3] Oddy Virgantara Putra[1,4]
Eko Mulyanto Yuniarno[1,5] Mauridhi Hery Purnomo[1,5]*

[1]*Department of Electrical Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia*
[2]*Department of Information Technology, Telkom University, Surabaya Campus, Surabaya, 60231, Indonesia*
[3]*Department of Electrical Engineering, Telkom University, Surabaya Campus, Surabaya, 60231, Indonesia*
[4]*Department of Informatics, Universitas Darussalam Gontor, Ponorogo, 63472, Indonesia*
[5]*Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, 60111, Indonesia*
* Email: hery@ee.its.ac.id

Abstract: Classifying 3D human poses with 3D point cloud data is an important task in predicting human activities. But spatio-temporal 3D point cloud data processing is complicated. Therefore, a solution is needed to handle it properly. A lightweight model without sacrificing its accuracy value is critical. This research presents Convolutional Neural Network (CNN) 3D based on Binary Voxel Feature Extraction (BIVFE) from Light Detection and Ranging (LiDAR) 3D point cloud data to enhance human poses prediction. This method is effective for analyzing temporal data from LiDAR systems, achieving excellent accuracy so that it can recognize several human poses very well. This method influences the discovery of a fit model. This research uses a novel dataset consisting of four classes: the stand-up pose, the sit-down pose, the squat-down pose, and the hands-to-the-side pose. This research also investigates the convolution of structure and the hyperparameter tuning. The average accuracy is 99.25% with inference time 1.7 seconds, and the best conditions when the hyperparameter settings are Adam optimizer, learning rate 0.0001, batch size 1, and epoch 50. These findings suggest that our proposed method presents promising opportunities for enhanced learning outcomes. These findings validate the CNN 3D based on BIVFE from LiDAR 3D point cloud data is accurately predicting human pose while handling the challenging of spatio-temporal 3D point cloud data. This research was also conducted on the benchmark dataset ModelNet10, ModelNet40, and ModelNet40-C. The results have proven our findings to be suitable and reliable for multi-class prediction case and quite reliable on data conditions containing noise as in ModelNet40-C. Our novel dataset can be visited at https://github.com/fzrahmanti/3Dhumanpose.

Keywords: LiDAR 3D, Human pose prediction, Binary voxel feature extraction (BIVFE), Convolutional neural network (CNN) 3D, Spatio-temporal 3D point cloud.

1. Introduction

Robustness in 3D point cloud human pose is an indispensable part of various apps, such as pedestrian detection focuses on identifying pedestrian or human locations [1 - 9], human pose estimation focuses on detecting joint points in the human body with the aim of understanding human movement [10 - 15], human action recognition focuses on the identification and classification of human actions derived from visual data [16 - 18], and human activity recognition focuses on recognizing the human long-term sequence of actions [19]. They are mainly focused on computer vision applications that use images or videos. However, advanced technologies that use sensors other than cameras, such as Light Detection and Ranging (LiDAR) 3D, led to additional difficulties

in object detection, pose prediction, activity recognition, etc.

Human pose prediction is a basic task in recognizing human activities. Pose prediction falls under the field of computer vision and artificial intelligence. Human pose prediction is the first step to make a machine or robot able to recognize everyday human activities.

Efficiency and accuracy in recognizing human poses are critical, so computer vision technology is expected to be applied in the real world. A lightweight model without sacrificing its accuracy value is critical.

Human pose prediction is possible to use several devices, such as cameras, radar, and LiDAR. This research focuses on the use of LiDAR for human pose prediction. LiDAR emits light on surrounding objects when scanning. The closer the distance between the object and the LiDAR, the higher the point density. The further away the object is, the lower the point density.

LiDAR produces data with spatial coordinates, namely point cloud data. This data has specific geographic information within a certain time span. It refers to a number of vectors with geometric positions in a coordinate system. Our research used LiDAR 3D. Each point consists of 3D coordinates, so it has x, y, and z values.

Some approaches that are often used in handling 3D points include point-based approaches by utilizing 3D points [3, 16, 20-23] but difficulty in handling non-uniform data density, projection-based approaches by converting into a 2D representation [6, 23] but loss of 3D information, and graph-based approaches by representing 3D points in a graph structure [16, 24] but difficulty handling temporal dynamics.

Voxel-based approach is the best solution for human pose prediction case. Voxel-based approaches by utilizing 3D points that are transformed into 3D voxel [16, 23, 25, 26]. Our strong approach is to use Binary Voxel Feature Extraction (BIVFE) to handle points in 3D space. This approach uses grids of a certain size. Each 3D point is inserted into the grids. The voxel approach ignores the density of points because they are already represented in the voxel grid so that it can speed up the feature retrieval process. Then, the voxel transformed into binary form. This feature approach is powerful and effective in recognizing human poses. The features that have been obtained as input values in the deep learning stage. This stage aims to train the spatio-temporal 3D point cloud data using CNN 3D based on Binary Voxel Feature Extraction (BIVFE). The CNN 3D algorithm is very effective in handling volumetric data or data with three dimensions: video (spatial and temporal dimensions) and point cloud data. The CNN 3D captures spatial and temporal features simultaneously. Capable of extracting more challenging and detailed features than CNNs 2D due to the 3D convolution kernels that operate on data in three dimensions. The fit model is used for testing on test data.

The main contributions of this paper are:

- This research proposes a CNN 3D based on BIVFE to enhance human pose prediction in spatio-temporal 3D point cloud data. This approach is effectively reliable for predicting human pose in multiple classes.
- The BIVFE approach is an efficient approach to obtain features in 3D point cloud data. Without reducing the information of the data and not considering the density of points to be efficient in the process of obtaining features. The illustration of points in a 3D point cloud is represented in a voxel grid as in Fig. 1. An example of the basic human pose is a standing pose, such as in the figure. Several points are in 3D space. Each point is represented in a voxel grid that has a specific size. The dimensions of the grid size can be determined in advance. One voxel can have more than one 3D point. After voxelization, it transformed into binary form. This approach called Binary Voxel Feature Extraction (BIVFE). The BIVFE output is an input into the learning process.
- This research investigates the convolution structure, CNN 3D, and CNN 2D with several models. It also investigates the use of an optimizer by using evaluation metrics that consider simplicity, accessibility, convergence stability, and computational efficiency.
- This research represents a novel human pose 3D point cloud dataset that consists of stand-up, sit-down, squat-down, and hands to the side poses. The human poses chosen are basic poses for carrying out daily activities. If the developed system can recognize basic poses well, then future developments can recognize other more challenging poses and even recognize human activities. Our novel dataset can be accessed on this link https://github.com/fzrahmanti/3Dhumanpose.git .
- This research also conducted training and testing on the benchmark ModelNet10, ModelNet40, and ModelNet40-C datasets. The results of this benchmark dataset have proven our findings to be suitable and reliable for multi-class prediction case and quite reliable on data conditions containing noise, as in ModelNet40-C.

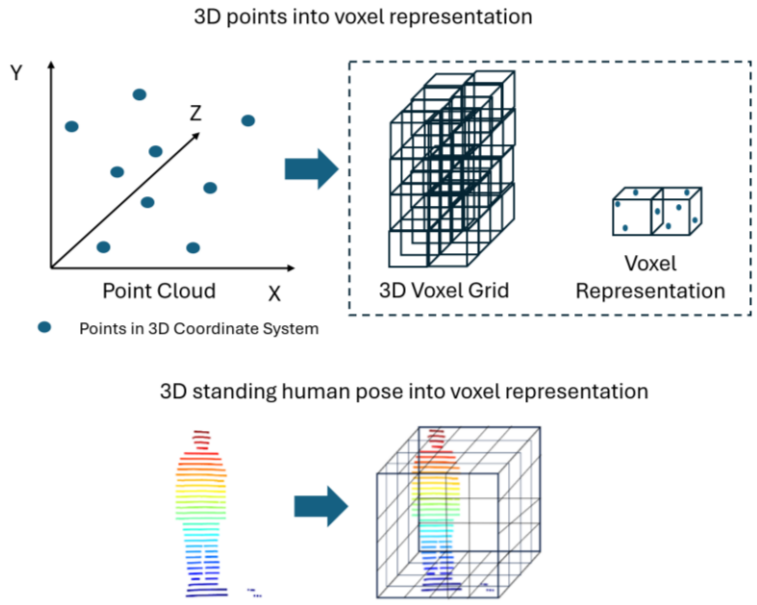

Figure. 1 Illustration of 3D points into voxel representation and 3D standing human pose into voxel representation

The rest of the paper is organized as follows, Section 1 presents the background problem raised, the proposed solution, a brief explanation of the research carried out, and the main contribution of this research. Section 2 discusses the related works and their gaps. Section 3 explains the dataset used and the proposed method. Section 4 shows the experiment results and discusses them. Section 5 discusses the conclusion of this research.

2. Related works

In this section, we represent the related works and their gaps. Several main research have been used as references in our research.

The research is conducted by [16] using NodeNs sensors to recognize human activity. The output data from this device is point cloud data. This research uses a single object and multiple objects, combining several human activities. It also compares with previous research that used different benchmark datasets. The first step is segmentation using DBSCAN to determine which areas have high and low point densities. Then, this research used an LSTM approach to recognize human activity. The accuracy result for a single subject reaches 95.75%. Therefore, there is room to improve accuracy with several activities and different devices.

In the other related work by [25], the research uses two approaches to processing point cloud data: a point-based and a voxel-based approach. The purpose of the research is to detect 3D objects with the Waymo dataset and KITTI benchmark dataset. The objects used are car, pedestrian, and cyclist. It outperformed other methods in the cyclist class alone, with the best score of 83.04%. The research reveals a gap in the method approach's application to human pose prediction and accuracy can be improved.

Several research by [11] and [27] focused on LiDAR-based human pose. The research by [11] combined LiDAR and camera data to estimate human pose. Therefore, the present research maximized and used data derived from LiDAR to predict human poses. [27], applied 3D convolutional methods in temporal learning tasks, specifically in estimating the 3D point cloud human pose. The present one implemented CNN 3D based on BIVFE of LiDAR 3D point cloud data for human pose prediction in spatio-temporal. Our approach has proven effective in predicting human pose because it reaches 99.25% the accuracy value better than previous research.

The research is conducted by [28], the stochastic gradient descent optimizer has the best results in detecting breast cancer in medical images. The stochastic gradient descent optimizer also has the shortest time during the training process compared to Adam and RMSprop. This inspired us to try using the stochastic gradient descent optimizer in recognizing human poses considering the accuracy and speed factors needed in the real world. Fast training time is needed to process challenging 3D point cloud data, which is a challenge for this research. But we also investigate Adam optimizer for our case.

Several strategies for assessing human pose classification, including dataset handling, preprocessing, convolutional layers, deep learning, and validation methods were also explored. A literature overview of several previous

Table 1. A literature overview of several previous methodologies

Research	Methodology	Limitations	Device Used	Dataset	Research Gap
[1], 2020	Pedestrian Planar LiDAR Pose (PPLP) Network: OrientNet, Region Proposal Network (RPN), PredictorNet	Handling 2D point cloud data and RGB images	LiDAR 2D, monocular camera imagery	CMU Panoptic Dataset and a newly collected FCAV M-Air Pedestrian (FMP) Dataset	Data processing is not much challenging 3D point cloud data, but limited information on data
[6], 2022	Bird Eye View (BEV) feature extraction network	Object detection	LiDAR and camera	KITTI dataset and mobile robot	Does not focused on human poses derived from 3D point cloud
[11], 2020	RGB and RGB-D approach	Human pose estimation	LiDAR	Raw dataset	Does not focused on human poses prediction derived from 3D point cloud
[12], 2019	Support Vector Machine (SVM)	3D dense skeleton and corresponding joint locations	LiDAR Full Motion Video (FMV)	Raw dataset	Differences in the used method and the comparison method
[16], 2017	DBSCAN + LSTM	Human activity	NodeNS	Raw dataset	Differences device used
[24], 2023	Modified GDANet	Noise and clutter point cloud data	LiDAR 3D	Raw dataset, ModelNet40-C	Multi-class object detection and static data
[26], 2023	Modified VGG-16	Human classification	LiDAR 3D	Raw dataset	Does not handling on temporal data
[27], 2019	3D convolutional neural network	3D human estimation	Two depth cameras	Raw dataset and public dataset (ITOP, EVAL, PDT)	Differences in the form of data processing because it comes from a different device
[29], 2020	FatNet	Focused on data processing then will be classified	(-) no device required	ModelNet40 dataset	Point cloud classification
[30], 2019	Dynamic Graph CNN (DGCNN)	Object detection	(-) no device required	ModelNet40, ShapeNetPart, and S3DIS dataset	Point cloud segmentation on static data
[31], 2018	PointCNN	Focused on point cloud handling problem	(-) no device required	Net40, ScanNet, TU-Berlin, Quick Draw, MNIST, CIFAR10, ShapeNet Parts, S3DIS, ScanNet	Does not work with 3D point cloud data
[32], 2024	DRF-SSD	Object detection	(-) no device required	KITTI dataset	Point cloud segmentation by reducing information loss
[33], 2025	NCFDet	Object detection from multi modals	6D rotating platform	Raw dataset, KITTI dataset	Robustness of image and point cloud features
Current	Binary Voxel Feature Extraction (BIVFE) + CNN 3D (Proposed Method)	Focused on spatio-temporal point cloud data	LiDAR 3D	Novel human pose dataset, ModelNet10, ModelNet40, ModelNet40-C	Binary voxel feature extraction and addresses spatio-temporal data of human pose prediction.

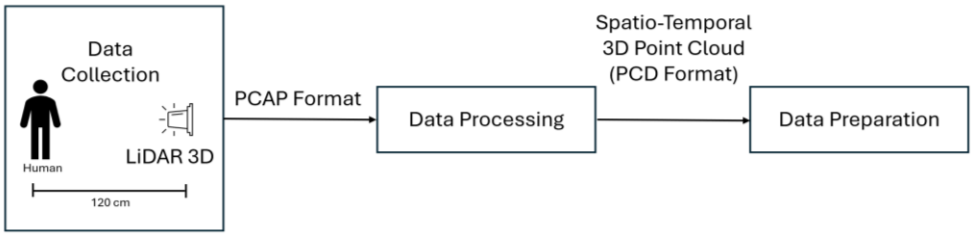

Figure. 2 The block diagram of getting materials

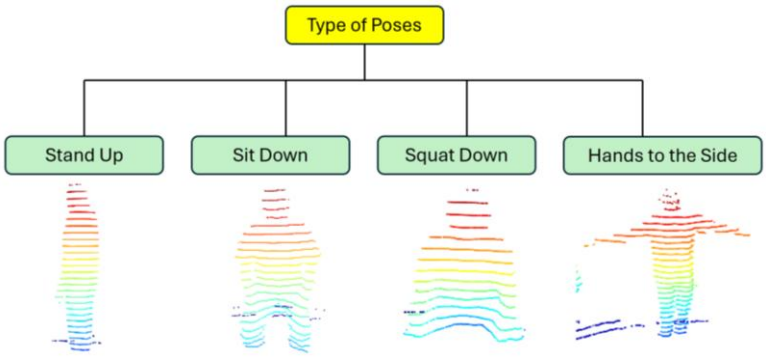

Figure. 3 Type of human poses

methodologies is shown in Table 1. The differences are visible based on methodologies, limitations, device used, dataset, and research gap. Our proposed method, CNN 3D based on BIVFE of LiDAR 3D point cloud data. The BIVFE approach is an efficient approach to obtain features in 3D point cloud data. Our proposed method is effective in enhancing human pose prediction. It uses a 3D convolution structure, which proves that it is excellent result at recognizing human poses in spatio-temporal 3D point cloud data.

3. Materials and method

In this section, we explain the materials and the proposed methods of this work. The materials consist of the data collection, data processing, and data preparation. The block diagram of getting materials is shown in Fig. 2.

Fig. 3 shows the types of human poses. Several types of poses were chosen, including sitting, standing, squatting, and sideways hand poses. Sitting and standing poses were chosen because they are basic human poses for various activities. The sideways hand pose is one of the human poses when exercising, and the squatting pose is one of the human poses when in the toilet.

3.1 Materials

The primary dataset used during this research was collected by itself using 3D Light Detection and Ranging (LiDAR). The distance between the blade and the LiDAR sensor is about 120 cm. The output from the LiDAR scanning process is in PCAP format. Data processing is needed, this step is used to process PCAP data into PCD format and extract each frame from the temporal 3D point cloud data. The sequence of steps is shown in Fig. 3, a block diagram of getting materials. Our dataset can be accessed on this link https://github.com/fzrahmanti/3Dhumanpose.git .

3.2 Proposed method

The proposed method of this research has a block diagram as in Fig. 4. Several steps taken include taking human pose data using LiDAR 3D, preprocessing data with BIVFE, and training using CNN 3D-based models so that it produces human pose predictions. This research also investigates the hyperparameter changes. The challenge of the proposed method is whether it also excels in the multiclass case when using the benchmark dataset, so we also added testing against ModelNet10, ModelNet40, and ModelNet40-C.

The data used in this research are discussed in the materials section. We will now discuss how to obtain features from 3D point cloud data using BIVFE approach and train using CNN 3D-based models. Each step from Fig. 4 will be explained in more detail in the following review.

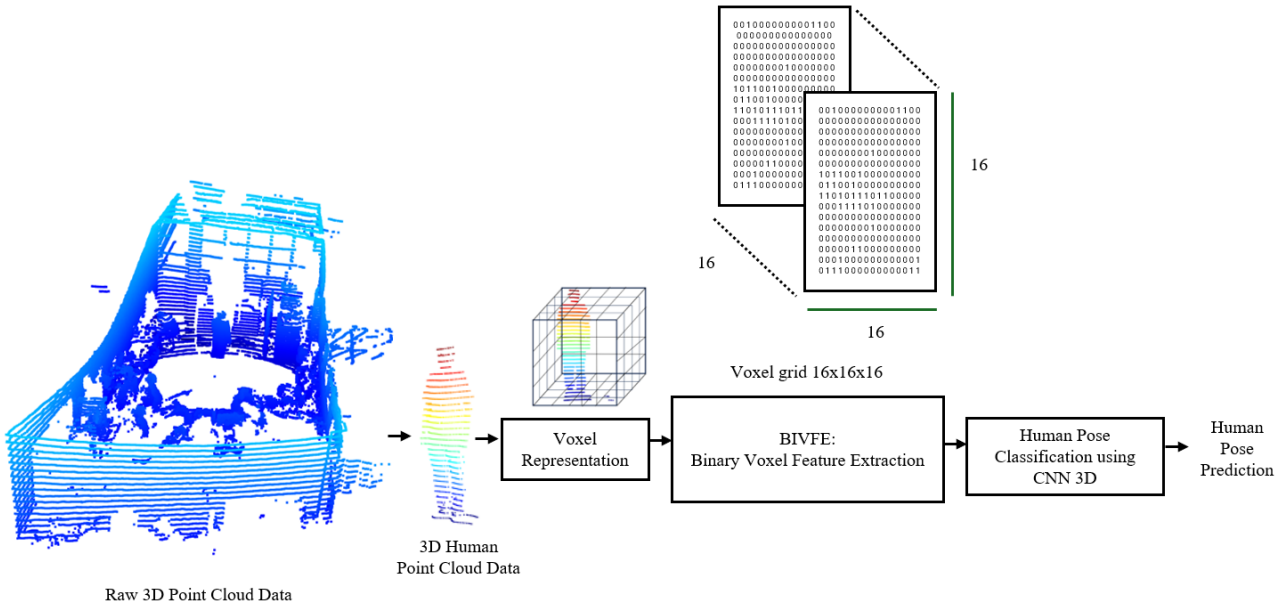

Figure. 4 The Block diagram of human pose prediction using CNN 3D based on BIVFE from LiDAR 3D point cloud data

Input_1	Input:	[(None,16,16,16,1)]
InputLayer	Output:	[(None,16,16,16,1)]
conv3d	Input:	(None,16,16,16,1)
Conv3D	Output:	(None,16,16,16,16)
max_pooling3d	Input:	(None,16,16,16,16)
MaxPooling3D	Output:	(None,8,8,8,16)
conv3d_1	Input:	(None,8,8,8,16)
Conv3D	Output:	(None,8,8,8,32)
max_pooling3d_1	Input:	(None,8,8,8,32)
MaxPooling3D	Output:	(None,4,4,4,32)
conv3d_2	Input:	(None,4,4,4,32)
Conv3D	Output:	(None,4,4,4,64)
conv3d_3	Input:	(None,4,4,4,64)
Conv3D	Output:	(None,4,4,4,64)
conv3d_4	Input:	(None,4,4,4,64)
Conv3D	Output:	(None,4,4,4,64)
max_pooling3d_2	Input:	(None,4,4,4,64)
MaxPooling3D	Output:	(None,2,2,2,64)
Flatten	Input:	(None,2,2,2,64)
Flatten	Output:	(None,512)
Dense	Input:	(None,512)
Dense	Output:	(None,4096)
Dense_1	Input:	(None,4096)
Dense	Output:	(None,4096)
Dense_2	Input:	(None,4096)
Dense	Output:	(None,4)

Figure. 5 CNN 3D-BIVFE Model

3.3 Binary voxel feature extraction (BIVFE)

The Binary Voxel Feature Extraction (BIVFE) consists of three parts, they are feature scaling, transformation, and voxel representation.

Feature scaling aims to handle numeric data with a different range of values, so it is necessary to present the numeric data on the same scale. At this part, this research uses a min-max scaler. The formula can be seen in Eq. (1). All features are scaled between a specified minimum and maximum value. At this stage, we have *x'*, *y'*, and *z'* values.

These results will later become input values at the transformation perform. The transformation performs used scaler transform, the output is *x''*, *y''*, and *z''*. Those 3D points will be represented into voxel grid.

The voxel approach starts from 3D points of point cloud data represented by a voxel grid of a certain size. The dimension of the grid is 16 x 16 x 1. The voxel representation of 3D points that have transformation and feature scaling produces binary voxels, where these results are used as feature extraction that enters the learning process. There are only values 1 and 0, these values mean the existence of a point in a voxel grid, which are the results of feature extraction and namely Binary Voxel Feature Extraction (BIVFE). It is easy and fast enough that the obtained features are immediately fed into the CNN 3D. The BIVFE described in Fig. 4 is an input learning.

$$X\,new = \frac{X - Xmin}{Xmax - Xmin}.(max - min) + min \quad (1)$$

3.4 Adaptive moment estimation (Adam) optimizer

Adaptive Moment Estimation or Adam Optimizer combines AdaGrad and RMSprop Optimizers. Adam optimizer uses the first moment (mean) and second moment (variance) estimates of the gradient to update the parameters.

The Adam algorithm calculates the gradients g of the loss function $\mathcal{L}$ on Eq. (2). Adam is refreshing the first-moment estimations m and the second-moment estimations v, which are in Eq. (3) and Eq. (4), respectively. Then, the bias in the first and second moment estimates is corrected as in Eq. (5). After that, calculate the adaptive learning rate α as in Eq. (6). Adam is refreshing the parameter model using Eq. (7).

$$g^{(t)} = \nabla\mathcal{L}\,(\theta^{(t-1)}) \quad (2)$$

$$m^{(t)} = \beta 1\, m^{(t-1)} + (1 - \beta 1)\, g^{(t)} \quad (3)$$

$$v^{(t)} = \beta 2\, v^{(t-1)} + (1 - \beta 2)\,(g^{(t)} \odot g^{(t)}) \quad (4)$$

$$\hat{m}^{(t)} = \frac{m^{(t)}}{1-\beta_1^t} \quad , \hat{v}^{(t)} = \frac{v^{(t)}}{1-\beta_2^t} \quad (5)$$

$$\alpha^{(t)} = \frac{\alpha^{(t-1)}\sqrt{1-\beta_2^t}}{1-\beta_1^t} \quad (6)$$

$$\theta^{(t)} = \theta^{(t-1)} - \frac{\alpha^{(t)}\,\hat{m}^t}{\sqrt{\hat{v}^{(t)}}+\epsilon} \quad (7)$$

3.5 Stochastic gradient descent (SGD) optimizer

SGD uses a subset of the data to update parameters, thereby reducing the computational burden compared to the entire dataset as in the Gradient Descent (GD) method. SGD has better generalization ability with new data. More frequent parameter updates (every mini-batch) allow SGD to find local minima faster than batch gradient descent methods that update parameters only once per epoch [11].

The SGD optimizer requires less memory because it only processes one mini-batch of data at a time, while other methods may require storing gradients for the entire dataset. The SGD algorithm is relatively simple and more accessible than other complex optimizers.

$$gt = \nabla\theta t J(\theta t) \quad (8)$$

$$\Delta\theta t = -\eta.\, gt \quad (9)$$

$$\theta t' = \theta t + \Delta\theta t \quad (10)$$

Where the iteration t, η learning rate on Eq. (8), Eq. (9), and Eq. (10). Some things to consider when choosing a learning rate when using SGD are that the inaccuracy of the learning rate can cause slow convergence or the model not to converge at all, and

the loss value can fluctuate and be unstable from iteration to iteration.

3.6 Convolutional neural network (CNN) 3D

Much research has used artificial intelligence, machine learning, or deep learning to solve problems such as optimization case [34], improving network security [35], detection case [36]-[37], decision support [38], and expert system [39].

3D convolution is usually used for volumetric and sequential data, the structure of 3D convolution. It is suitable for detecting 3D objects in 3D point cloud data. 3D convolution can capture spatial-temporal information that cannot be done with 2D convolution. 3D convolution works by using a 3D filter that covers volumetric input data such as 3D point cloud data. This filter moves along 3D to capture spatial and temporal features simultaneously, so this 3D convolution requires more computing resources. The 3D input size is d_{in} *x* h_{in} *x* w_{in}, the filter 3D convolution size is d_k *x* h_k *x* w_k and the feature map size is d_{out} *x* h_{out} *x* w_{out}.

The convolution architecture of the proposed CNN 3D based on BIVFE of LiDAR 3D point cloud in this research is shown in Fig. 5. The 3D point cloud data is represented into a voxel. This research uses Binary Voxel Feature Extraction (BIVFE), then results as an input layer in the learning process.

The convolution layers consist of three blocks. Block one consists of 3D convolution and max pooling. This 3D convolution used filter 16 and kernel dimensions *3 x 3 x 3*. The dimension of max pooling used in Block one is *2 x 2 x 2*.

Block two consists of two 3D convolution and max pooling. The 3D convolution used filter 32 and kernel dimensions *3 x 3 x 3*. The dimension of max pooling used in Block two is *2 x 2 x 2*.

Block three consists of three 3D convolutions and max pooling. The 3D convolutions used filter 64 and kernel dimensions *3 x 3 x 3*. The dimension of max pooling used in Block three is *2 x 2 x 2*. Each convolution in each block used ReLU activation. After a fully connected layer follows the convolution layer and produces a human pose prediction class. A list of abbreviations of this research can be seen in Table 2.

4. Result and discussion

The total data used in this research is 1000 frames of 3D point cloud. We divide the data preparation into two: using cross-validation and without cross-validation. Data preparation with cross-validation amounts to 250 frames for each human pose class, while data preparation without cross-validation amounts to 100 frames. Table 3 shows the scenario experiment and division of training and test data. The k-fold cross-validation was used, and this research used 5-fold during the experiment. This research also tests the proposed method against benchmark datasets such as ModelNet10, ModelNet40, and ModelNet40-C.

In addition to the dataset, another essential thing is the computer specifications used during the training and testing process. This research uses computer specifications CPU Intel(R) Core(TM) i5-8350U CPU @1.70GHz and memory 16GB speed 2400 MHz.

First of all, we did cross-validation using 5-fold during the training and testing process, and the results we have observed are the best results according to Table 4. The AlexNET model with Adam optimizer has the best results when fold 2 is used. The human pose class that was successfully predicted perfectly is the hands to the sides (arm stretching pose). At the same time, the sit-down and squat-down poses have the same f-score value and are very good. So, it can be concluded that this model is very accurate and sensitive to data.

The LeNET model with Adam optimizer has the best results when fold 3. Almost the same as the results of the AlexNet model, the human poses that were successfully predicted perfectly were the stretching arms pose and the squat down pose. The human pose with quite good results, with an f-score of 0.91, was the sit-down pose compared to the stand-up pose, with an f-score of 0.89. So, it can be concluded that the prediction results for the sit-down human pose are more accurate and sensitive than the stand-up pose prediction.

CNN 3D with Adam optimizer was also implemented on the VGG16 model. In this experiment, it had the best results on fold 3. This model recognizes human poses very well and is sensitive to data, as evidenced by the f-score value for each human pose class, which is 0.95.

Table 4 shows the evaluation results with CNN 3D with Adam optimizer. It has the best results on the AlexNet CNN model with fold 2, which is able to predict poses very well in all human poses. This is evidenced by the perfect f-score value of 1 in the stand-up and hands-to-the-side pose classes and a very good f-score value in the sit-down and squat-down poses.

Experiments were also conducted with a 2D convolution structure that aims to provide evidence and more deeply analyze the 3D convolution structure, which is suitable for handling spatio-temporal data such as spatio-temporal 3D point

Table 2. Abbreviation during experiments

Abbreviation	Details
N	The number of frames
$d_{in,}$ $h_{in,}$ $w_{in,}$	Input size
$d_{k,}$ $h_{k,}$ w_{k}	Kernel size
$d_{out,}$ $h_{out,}$ w_{out}	Output size
x, y, z	3D points
K	K-Fold, Division of a dataset into K subsets of equal size
E	Evaluation
Θ	Model parameters
A	Learning rate
$\beta1, \beta2, \varepsilon$	Hyperparameters

Table 3. Scenario experiment of this research

Scenario	Dataset	With Cross-Validation	Total	Total Data for Each Class	Convolution Dimension	Optimizer
1	Our novel dataset	Yes	1000	250 stand-up (200 training, 50 testing) 250 sit-down (200 training, 50 testing) 250 squat-down (200 training, 50 testing) 250 hands-to-the-side (200 training, 50 testing)	CNN 3D	Adam
2	Our novel dataset				CNN 2D	Adam
3, 4	Our novel dataset	No	400 and 100	100 stand-up (80 training, 20 testing) 100 sit-down (80 training, 20 testing) 100 squat-down (80 training, 20 testing) 100 hands-to-the-side (80 training, 20 testing)	CNN 3D	Adam
5, 6			400 and 100		CNN 3D	SGD
7	ModelNet10	No	4899	The amount of data varies for each class	CNN 3D	Adam
8	ModelNet40	No	12311	The amount of data varies for each class. (2468 testing)	CNN 3D	Adam
9	ModelNet40-C	No	12311	The amount of data varies for each class. (2468 testing)	CNN 3D	Adam

Table 4. Best evaluation result on CNN 3D based on BIVFE with Adam optimizer (scenario 1)

CNN Model	K-Fold	Evaluation	Sit Down	Squat Down	Stand Up	Hands to the Sides
AlexNet	2	Recall	0.91	1.00	1.00	1.00
		Precision	1.00	0.90	1.00	1.00
		F-Score	**0.95**	0.95	1.00	1.00
LeNet	3	Recall	0.83	1.00	1.00	1.00
		Precision	1.00	1.00	0.80	1.00
		F-Score	0.91	1.00	0.89	1.00
VGG16	3	Recall	0.91	1.00	0.91	1.00
		Precision	1.00	0.90	1.00	0.90
		F-Score	**0.95**	0.95	0.95	0.95

Table 5. Evaluation result on CNN 2D based on BIVFE with Adam optimizer (scenario 2)

Model	K-Fold	Evaluation	Sit Down	Squat Down	Stand Up	Hands to the Sides
AlexNet	2	Recall	0.83	1.00	1.00	1.00
		Precision	1.00	0.90	0.90	1.00
		F-Score	0.91	0.95	0.95	1.00
LeNet	2	Recall	0.83	1.00	1.00	1.00
		Precision	1.00	0.90	0.90	1.00
		F-Score	0.91	0.95	0.95	1.00
VGG16	2	Recall	0.77	0.78	1.00	1.00
		Precision	1.00	0.70	0.80	1.00
		F-Score	0.87	0.74	0.89	1.00

cloud data from LiDAR. Therefore, this research also implements a CNN 2D with the Adam optimizer, which has evaluation results according to Table 5. The AlexNet, LeNet, and VGG16 models with CNN 2D have the best results on fold 2. However, each has different precision, recall, and f-score values results. The AlexNet model can recognize human hand poses to the side ideally but can still recognize other human poses, such as sit-down, squat-down, and stand-up poses, very well with f-score values of 0.91, 0.95 and 0.95, respectively. The results of this evaluation on the AlexNet model with a CNN 3D are better than those of the AlexNet model with a CNN 2D.

Meanwhile, the LeNet model with CNN 2D and Adam optimizer has very good evaluation results, so it is able to recognize human poses with f-score values in each class of human poses: sit-down 0.91, squat-down 0.95, stand-up 0.95, and hands to the sides 1. The results of this evaluation on the LeNet model with CNN 3D have perfect results on the squat-down, and hands-to-the-sides poses compared to the LeNet model with CNN 2D.

The VGG16 model with a CNN 2D has quite good results in recognizing human poses but still has lower results than the VGG16 model with a CNN 3D. Table 5 shows that the f-score evaluation results for each pose of sit-down, squat-down, stand-up, and hands to the sides are 0.87, 0.74, 0.89, and 1.00, respectively. These results indicate that the VGG16 model with CNN 3D and Adam optimizer can recognize human poses and is very sensitive to new data compared to the VGG16 model with CNN 2D and Adam optimizer.

Comparison of the accuracy of CNN 3D and CNN 2D according to Table 6. The accuracy results presented in the table show that the CNN 3D with the best model is the AlexNet model with Adam optimizer on fold 2 with an accuracy value of 97%. At the same time, the CNN 2D with the best model is the AlexNet model with Adam optimizer on fold 2 with an accuracy value of 95%. A less favorable scenario occurred with 5-fold cross-validation, resulting in a significantly lower accuracy of 55% and 53%. These results show that the CNN 3D, accurately predicted human pose. The chart's visualization and error bar can be shown in Fig. 6.

In this scenario of current experiment, it can be highlighted that the best performance was observed when CNN 3D with Adam optimizer was used in the AlexNet CNN model, which achieved an impressive accuracy of 97. This best model of CNN 3D is very accurate and sensitive to data.

The comprehensive accuracy comparison between CNN 3D and CNN 2D methods. Meanwhile Fig. 7 depicts the confusion matrix for the top-performing model, with instances of incorrectly and correctly predicted 3D point cloud human pose in this setting. This is due to the ambiguity of human poses, which have similarities between squatting and sitting down poses. In addition, the proposed model has difficulty recognizing human poses in detail due to the lack of local information caused by the point density variance. Meanwhile, the results of the evaluation metrics are obtained by getting FP, FN, TP, and TN. This research is categorical cases based on its data, so the test evaluation is done using the metrics of accuracy, precision, recall, and f-score. Accuracy can be calculated from the TP, TN, FP, and FN values.

Fig. 8 shows the accuracy and loss model graph, the accuracy graph for training and validation, and the loss graph for training and validation for the best CNN 3D with Adam optimizer in this experiment.

Moving from Fig. 8 to Table 7, another experiment, this time using changes in data, the number of epochs, and the optimization method used. It can be concluded from this table that our proposed method, namely CNN 3D based on BIVFE,

is very effective and efficient in recognizing human poses with the best results using Adam optimizer and 50 epochs. The time required to perform training is quite short, 50 epochs, compared to using other optimizations, namely SGD. So, the use of Adam optimizer is quite fast in recognizing human poses.

Table 6. Comparison the accuracy results between CNN 3D and CNN 2D with Adam optimizer

Convolution Structure	Model	K-Fold	Accuracy (%)	Convolution Structure	Model	K-Fold	Accuracy (%)
3D	AlexNet	1	88	2D	AlexNet	1	85
		2	**97**			2	95
		3	95			3	88
		4	90			4	95
		5	60			5	75
	LeNet	1	88		LeNet	1	95
		2	93			2	95
		3	**95**			3	90
		4	90			4	85
		5	60			5	65
	VGG16	1	88		VGG16	1	85
		2	88			2	88
		3	**95**			3	82
		4	80			4	75
		5	55			5	53

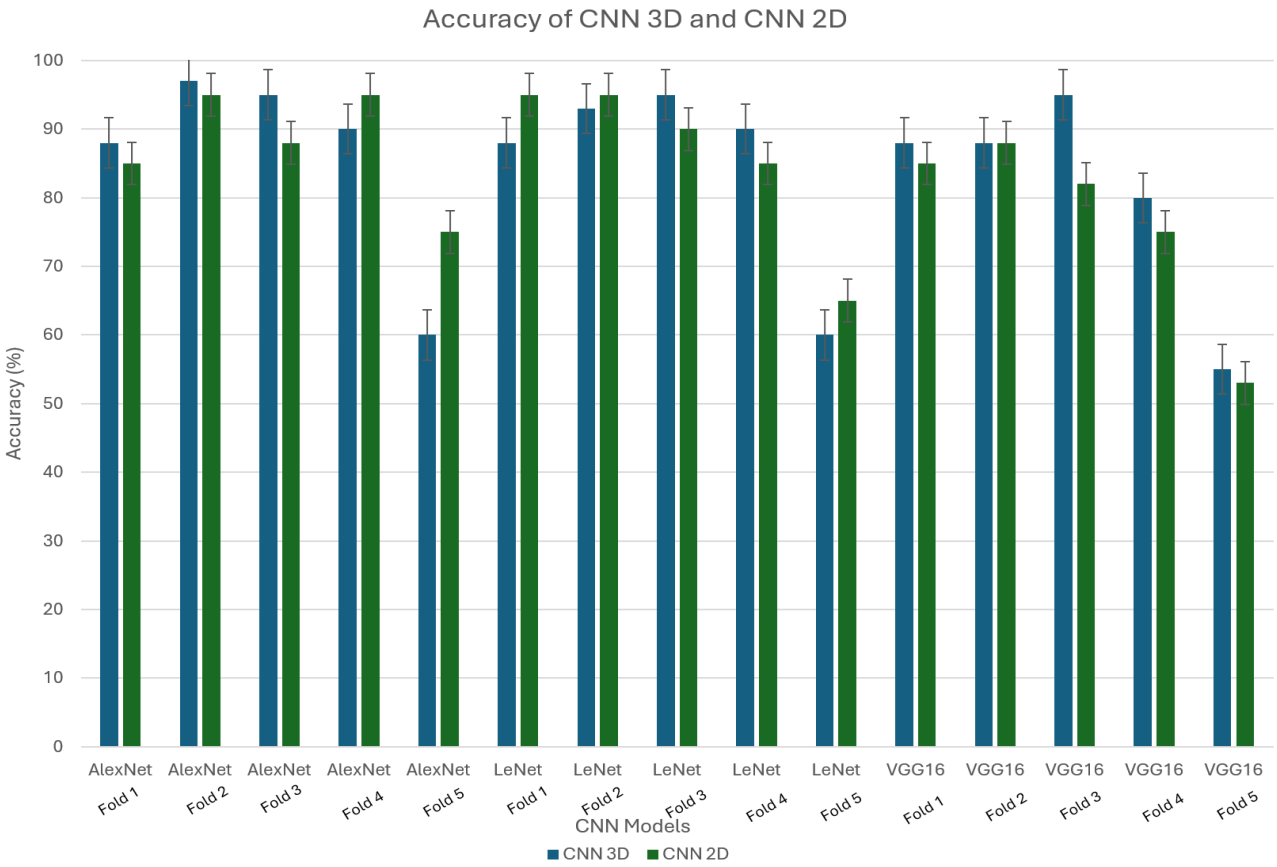

Figure. 6 Comparison the accuracy results between CNN 3D and CNN 2D with Adam optimizer and their error bar

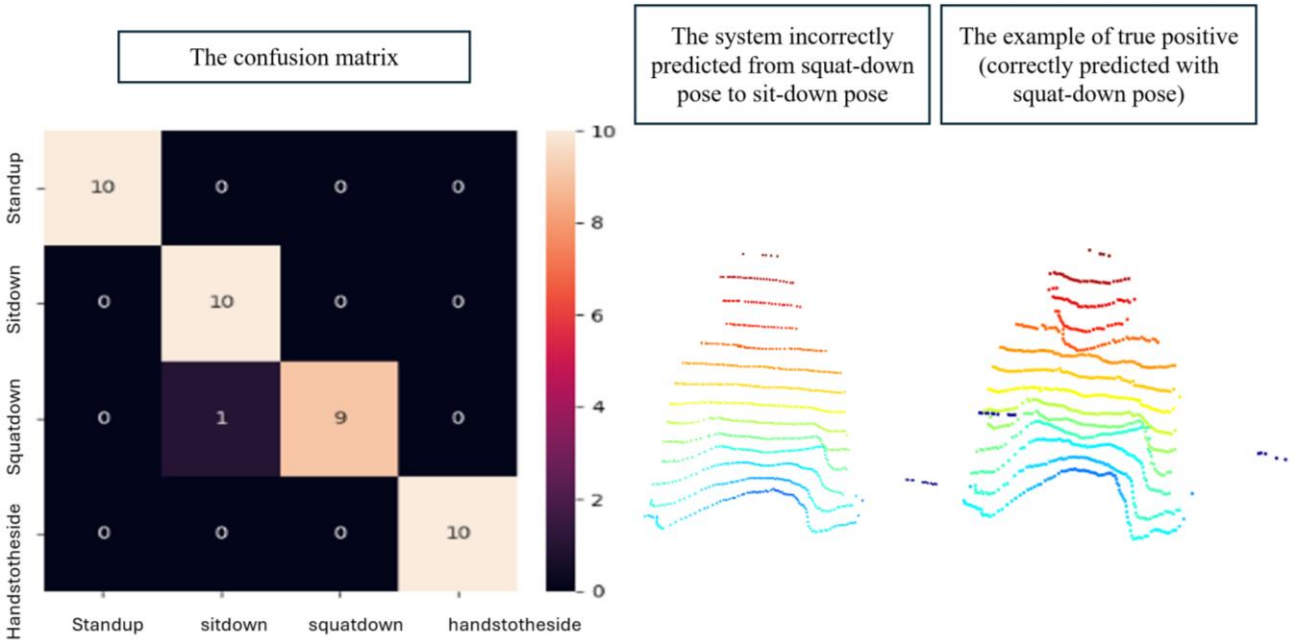

Figure. 7 CNN 3D with Adam optimizer and epoch 50

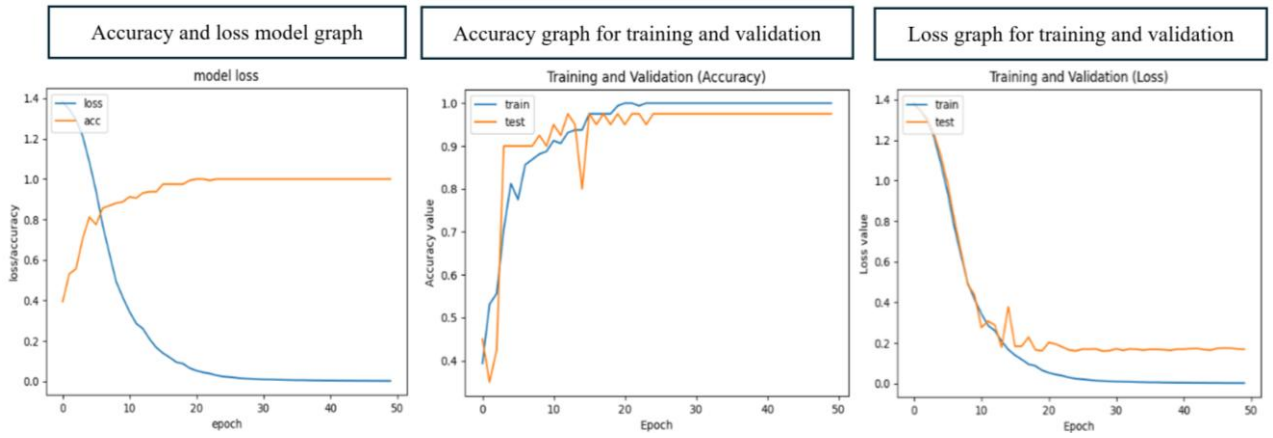

Figure. 8 CNN 3D based on BIVFE with Adam optimizer and epoch 50

Table 7. Experiment results of CNN 3D based on BIVFE with hyperparameter changes (scenario 3 until 6)

Scenario	Total Data	Epoch	Learning Rate	Optimizer	Testing Accuracy (%)	Inference Time (seconds)
3	100	850	0.01	SGD	100	1.2
4	400	850	0.01	SGD	100	1.2
5	100	50	0.0001	Adam	97	1.7
6	400	50	0.0001	Adam	100	1.7

Table 8. Comparison of average accuracy using different datasets based on total data and average accuracy

Dataset	Total Data	Average Accuracy (%)	Average Inference Time (seconds)
ModelNet10	4899	89.43	13.86
ModelNet40	12311	91.90	30.39
ModelNet40-C	12311	93.11	28.87
Our novel dataset	1000	**99.25**	1.45

Table 9. Comparative research of different methodology

Research	Methodology	Focus	Dataset	Accuracy (%)
[6], 2022	Bird Eye View (BEV) feature extraction network	Object detection	KITTI dataset and mobile robot	87.43
[26], 2023	Modified VGG-16	Human classification	Raw dataset point cloud	90.00
[24], 2023	Modified GDANet	Noise and clutter point cloud data	Raw dataset, ModelNet40-C	96.70
[27], 2019	3D convolutional neural network	3D human estimation	Raw dataset and public dataset (ITOP, EVAL, PDT)	98.00
[32], 2024	DRF-SSD	Object detection	KITTI dataset	88.74
[33], 2025	NCFDet	Object detection from multi modals	Raw dataset, KITTI dataset	91.77
Current	Proposed Method: CNN 3D based on BIVFE	Human pose prediction that focused on spatio-temporal 3D point cloud data	Novel human pose dataset, benchmark dataset: ModelNet10, ModelNet40, ModelNet40-C	**99.25**

Another experimental scenario is an experiment with CNN 3D based on BIVFE with hyperparameter changes. Those experiments do not use cross-validation, so the data distribution between training and testing data is also different. Those experiments use 400 data, with 320 training data and 80 testing data. Table 7 shows the results of the experiments conducted in this research with epoch 50 for Adam optimizer and 850 for SGD optimizer. The best experiment is when this research uses Adam optimizer with epoch 50, batch size 1, learning rate 0.0001, and it has an inference time of 1.7 seconds.

This research proposes CNN 3D based on BIVFE with Adam optimizer, which has been proven to recognize human poses better than other comparative methods. The number of datasets has been proven to affect the accuracy value and computation time. The more data is trained, the better the machine recognizes human poses. The experiment with Adam produced excellent accurate results. It was efficient in recognizing human poses by remembering the epoch value.

Testing the reliability of the proposed method requires using not only our dataset but also other public datasets such as ModelNet10, ModelNet40, and ModelNet40-C. In a comparison of experiments using our proposed method, the ModelNet10 dataset achieve an accuracy value of 89.43%, the ModelNet40 dataset achieve an accuracy value of 91.90%, and the ModelNet40-C dataset achieve an accuracy value of 93.11%. In comparison, our dataset has an average accuracy of 99.25%. Table 8 shows that the proposed method is very effective for multi-class cases, quite reliable on data conditions containing noise as in ModelNet40-C, and efficient because it only consumes 50 epochs.

Table 9 is comparative research of different methodology, focus, and accuracy results. The results analyzed in this current scenario show that the best models are an architecture with a CNN 3D based on BIVFE with Adam optimizer. The proposed method successfully outperforms other methods.

Fig. 9 shows the evaluation metrics such as accuracy, precision, recall, and f-score. The f-score from the best performance has an excellent result; there is a balance between precision and recall results, which means the model accurately predicts and captures the positive class. Our proposed method is very good at predicting human poses of LiDAR 3D point clouds.

5. Conclusion

This research proposes a CNN 3D based on BIVFE from LiDAR 3D to enhance human pose prediction in spatio-temporal 3D point cloud data. This approach is effectively reliable for predicting human pose in multiple classes.

The experiment in this research used cross-validation and the one without cross-validation to transmit model performance. The existence of this scenario aims to determine the effect of cross-validation. Analysis of the experimental results shows that the computing time is longer if cross-validation is used because the model is trained and

Our Novel Dataset

Test accuracy: 100.00% Test loss: 0.007

	precision	recall	f1-score	support
standup	1.00	1.00	1.00	10
sitdown	1.00	1.00	1.00	10
squatdown	1.00	1.00	1.00	10
handstotheside	1.00	1.00	1.00	10
accuracy			1.00	40
macro avg	1.00	1.00	1.00	40
weighted avg	1.00	1.00	1.00	40

ModelNet10

Test accuracy: 89.43% Test loss: 0.317

	precision	recall	f1-score	support
bathtub	0.90	0.90	0.90	50
bed	0.96	0.91	0.93	106
chair	0.98	0.98	0.98	100
desk	0.83	0.79	0.81	90
dresser	0.85	0.74	0.79	98
monitor	0.97	0.98	0.97	99
night_stand	0.55	0.90	0.68	52
sofa	1.00	0.95	0.98	105
table	0.82	0.85	0.83	97
toilet	0.97	0.99	0.98	98
micro avg	0.89	0.90	0.89	895
macro avg	0.88	0.90	0.89	895
weighted avg	0.90	0.90	0.90	895

ModelNet40

Test accuracy: 91.90% Test loss: 0.239

	precision	recall	f1-score	support
0	1.00	1.00	1.00	100
1	0.94	0.89	0.91	53
2	0.99	0.95	0.97	104
3	0.65	0.46	0.54	28
4	1.00	0.91	0.95	110
5	0.99	0.97	0.98	102
6	0.90	0.78	0.84	23
7	1.00	1.00	1.00	100
8	0.87	1.00	0.93	87
9	0.95	0.90	0.93	21
10	0.60	1.00	0.75	12
11	0.85	0.74	0.79	23
12	0.83	0.95	0.88	75
13	0.95	0.86	0.90	22
14	0.94	0.75	0.84	108
15	0.20	0.50	0.29	8
16	0.83	1.00	0.91	83
17	0.99	0.97	0.98	102
18	0.80	0.94	0.86	17
19	0.85	0.81	0.83	21
20	1.00	1.00	1.00	20
21	1.00	0.93	0.96	108
22	0.96	0.98	0.97	98
23	0.79	0.87	0.83	78
24	0.80	0.80	0.80	20
25	1.00	0.98	0.99	102
26	0.98	0.88	0.92	112
27	0.60	0.71	0.65	17
28	0.98	1.00	0.99	98
29	0.80	1.00	0.89	16
30	1.00	0.98	0.99	102
31	0.90	0.95	0.92	19
32	0.50	0.43	0.47	23
33	0.91	0.93	0.92	98
34	0.90	0.95	0.92	19
35	0.98	0.99	0.98	99
36	0.92	0.96	0.94	96
37	0.95	0.90	0.93	105
38	0.35	0.54	0.42	13
39	0.70	0.54	0.61	26
accuracy			0.92	2468
macro avg	0.85	0.87	0.85	2468
weighted avg	0.93	0.92	0.92	2468

ModelNet40-C

Test accuracy: 93.11% Test loss: 0.209

	precision	recall	f1-score	support
0	1.00	1.00	1.00	100
1	0.98	0.89	0.93	55
2	0.99	0.93	0.96	106
3	0.45	1.00	0.62	9
4	1.00	1.00	1.00	100
5	1.00	1.00	1.00	100
6	0.85	1.00	0.92	17
7	1.00	1.00	1.00	100
8	1.00	0.94	0.97	106
9	0.95	0.95	0.95	20
10	0.90	0.82	0.86	22
11	0.80	0.89	0.84	18
12	0.93	0.84	0.88	95
13	0.95	0.79	0.86	24
14	0.98	0.64	0.77	131
15	0.50	0.91	0.65	11
16	0.94	0.96	0.95	98
17	1.00	0.98	0.99	102
18	0.95	0.90	0.93	21
19	0.75	1.00	0.86	15
20	1.00	1.00	1.00	20
21	0.94	0.99	0.96	95
22	1.00	0.99	1.00	101
23	0.41	1.00	0.58	35
24	1.00	0.95	0.98	21
25	0.99	0.96	0.98	103
26	0.95	0.90	0.93	105
27	0.65	0.68	0.67	19
28	1.00	0.99	1.00	101
29	0.65	0.87	0.74	15
30	0.99	1.00	0.99	99
31	0.85	0.85	0.85	20
32	0.65	0.81	0.72	16
33	0.94	0.92	0.93	102
34	0.95	0.90	0.93	21
35	0.97	0.99	0.98	98
36	0.97	0.92	0.95	105
37	0.96	0.93	0.95	103
38	0.65	0.81	0.72	16
39	0.80	0.70	0.74	23
accuracy			0.93	2468
macro avg	0.88	0.92	0.89	2468
weighted avg	0.95	0.93	0.94	2468

Figure. 9 Precision, recall, f-score, support from each class from best performance

tested several times according to the fold that was determined at the beginning of the experiment. If cross-validation is not done, it will be faster because it only involves one training and testing process.

However, if you do not use cross-validation, it will be susceptible to bias if the train-test split data is not done representatively. However, this did not happen in the experiments that have been carried out. The cross-validation scenario can provide excellent results for the dataset used.

This research also investigates the convolution structure, CNN 3D and CNN 2D with several models. It also investigates the hyperparameter changes by using evaluation metrics that consider simplicity, accessibility, convergence stability, and computational efficiency. The experimental results showed that integrating 3D convolution in convolutional layer of a deep learning model with Adam optimizer increased accuracy in recognizing human pose until it reached 99.25% with inference time 1.7 seconds.

This research represents a novel human pose 3D point cloud dataset that consists of stand-up, sit-down, squat-down, and hands to the side poses. But this research was also conducted training and testing on the benchmark ModelNet10 dataset and it reached 89.43% for accuracy value. In addition, this research also conducted experiments on the benchmark dataset ModelNet40 and ModelNet40-C, the accuracy results of which reached 91.90% and 93.11%. The results of this benchmark dataset have proven our findings to be suitable and reliable for multi-class prediction case and quite reliable on data conditions containing noise.

There is a case where the proposed model fails to predict the 3D point cloud human pose due to the ambiguity of human poses, which have similarities between squatting and sitting down poses. Therefore, the proposed method has difficulty recognizing human poses in detail due to the lack of local information caused by the point density variance.

Research suggestion to the future research is integration of other feature extraction approach and other learning algorithm to improved prediction outcomes. In addition, hardware specifications need to be improved to support better computing speed and computing costs.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, Farah Zakiyah Rahmanti, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomo; methodology, Farah Zakiyah Rahmanti, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomoand; software, Farah Zakiyah Rahmanti; validation, Eko Mulyanto Yuniarno; formal analysis, Farah Zakiyah Rahmanti, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomo; investigation, Farah Zakiyah Rahmanti, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomo; resources, Farah Zakiyah Rahmanti; data curation, Farah Zakiyah Rahmanti, Moch. Iskandar Riansyah, Oddy Virgantara Putra; writing—original draft preparation, Farah Zakiyah Rahmanti; writing—review and editing, Farah Zakiyah Rahmanti, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomo; visualization, Farah Zakiyah Rahmanti; supervision, Eko Mulyanto Yuniarno, Mauridhi Hery Purnomo; project administration, Farah Zakiyah Rahmanti, Moch. Iskandar Riansyah, Oddy Virgantara Putra.

Acknowledgments

This work is supported by Beasiswa Pendidikan Indonesia (BPI) through the PhD completion scholarship program in 2023.

References

[1] F. Bu, T. Le, X. Du, R. Vasudevan, and M. Johnson-Roberson, "Pedestrian planar LiDAR pose (PPLP) network for oriented pedestrian detection based on planar LiDAR and monocular images", *IEEE Robotics and Automation Letters*, Vol. 5, No. 2, pp. 1626–1633, 2020, doi: 10.1109/LRA.2019.2962358.

[2] H. Wang, B. Wang, B. Liu, X. Meng, and G. Yang, "Pedestrian recognition and tracking using 3D LiDAR for autonomous vehicle", *Robotics and Autonomous Systems*, Vol. 88, pp. 71–78, 2017, doi: 10.1016/j.robot.2016.11.014.

[3] Z. Li, Y. Yao, Z. Quan, J. Xie, and W. Yang, "Spatial information enhancement network for 3D object detection from point cloud", *Pattern Recognition*, Vol. 128, p. 108684, 2022, doi: 10.1016/j.patcog.2022.108684.

[4] Y. Ye, H. Chen, C. Zhang, X. Hao, and Z. Zhang, "SARPNET: Shape attention regional proposal network for LiDAR-based 3D object detection", *Neurocomputing*, Vol. 379, pp. 53–63, 2020, doi: 10.1016/j.neucom.2019.09.086.

[5] J. Zhao, H. Xu, H. Liu, J. Wu, Y. Zheng, and D. Wu, "Detection and tracking of pedestrians and vehicles using roadside LiDAR sensors", *Transportation Research Part C: Emerging Technologies*, Vol. 100, pp. 68–87, 2019, doi: 10.1016/j.trc.2019.01.007.

[6] J. Li, R. Li, J. Li, J. Wang, Q. Wu, and X. Liu, "Dual-view 3D object recognition and detection via LiDAR point cloud and camera image", *Robotics and Autonomous Systems*, Vol. 150, p. 103999, 2022, doi: 10.1016/j.robot.2021.103999.

[7] L. H. Wen and K. H. Jo, "Fast and accurate 3D object detection for LiDAR-camera-based autonomous vehicles using one shared voxel-based backbone", *IEEE Access*, Vol. 9, pp. 22080–22089, 2021, doi: 10.1109/ACCESS.2021.3055491.

[8] D. Krawczyk and R. Sitnik, "Segmentation of 3D point cloud data representing full human body geometry: a review", *Pattern Recognition*, Vol. 139, p. 109444, 2023, doi: 10.1016/j.patcog.2023.

[9] S. L. Prathapareddy et al., "Implementation of Video-Based Human Anomalous Activity Detection Using LSTM-RNN Network", In: *Proc. of International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)*, pp. 853–858, 2023, doi: 10.1109/ICSSAS57918.2023.10331856.

[10] JJ. Kaltenthaler, H. A. Lauterbach, D. Borrmann, and A. Nüchter, "Pose estimation and mapping based on IMU and LiDAR", *IFAC-PapersOnLine*, Vol. 55, No. 8, pp. 71–76, 2022, doi: 10.1016/j.ifacol.2022.08.012.

[11] M. Fürst, S. T. P. Gupta, R. Schuster, O. Wasenmüller, and D. Stricker, "HPERL: 3D human pose estimation from RGB and LiDAR", In: *Proc. of 2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 7321, 2021, doi: 10.1109/ICPR48806.2021.9412785.

[12] A. Glandon et al., "3D Skeleton Estimation and Human Identity Recognition Using LiDAR Full Motion Video", In: *Proc. of International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2019, doi: 10.1109/IJCNN.2019.8852370.

[13] D. Skorvankova and M. Madaras, "Human pose estimation using per-point body region assignment", *Computing and Informatics*, Vol. 40, pp. 387–407, 2021.

[14] K.-C. Chan, "On the 3D Point Cloud for Human-Pose Estimation", *Purdue University*, 2016.

[15] W. Wang et al., "A Novel Identification Method of Helmet Wearing Based on Human Pose Estimation", In: *Proc. of International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, pp. 180–185, 2022, doi: 10.1109/PRAI55851.2022.9904284.

[16] Z. Yu et al., "A radar-based human activity recognition using a novel 3D point cloud classifier", *IEEE Sensors Journal*, Vol. 22, No. 19, pp. 18218–18227, 2022, doi: 10.1109/JSEN.2022.3198395.

[17] H. Wang et al., "Implementation of video-based human and key frame selection for human activity prediction", *Neurocomputing*, Vol. 318, pp. 109–119, 2018, doi: 10.1016/j.neucom.2018.08.037.

[18] M. Liandana, D. P. Hostiadi, N. R. Hendrawan, G. A. Pradipta, P. D. W. Ayu, "Enhanced Human Activity Recognition (HAR): Leveraging Sub-Window Techniques and Feature Ratios from Triaxial Accelerometer Data", *International Journal of Intelligent Engineering and Systems*, Vol.18, No. 1, 2025, doi: 10.22266/ijies2025.0229.34.

[19] F. Goldau et al., "DORMADL - Dataset of Human-Operated Robot Arm Motion in Activities of Daily Living", In: *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11396–11403, 2023, doi: 10.1109/IROS55552.2023.10341459.

[20] Y. Zhou and O. Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection", In: *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[21] Z. Wang et al., "3D MSSD: A multilayer spatial structure 3D object detection network for igmobile LiDAR point clouds", *International Journal of Applied Earth Observation and Geoinformation*, vol. 102, p. 102406, 2023, doi: 10.1016/j.jag.2021.102406.

[22] A. H. Lang et al., "PointPillars: Fast Encoders for Object Detection from Point Clouds", In: *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12689–12697, 2019, doi: 10.1109/CVPR.2019.01298.

[23] Y. Wu et al., "Deep 3D Object Detection Networks Using LiDAR Data: A Review", *IEEE Sensors Journal*, Vol. 21, No. 2, 2021.

[24] O. V. Putra et al., "Enhancing LiDAR-Based Object Recognition Through a Novel Denoising and Modified GDANet Framework", *IEEE Access*, 2023.

[25] J. S. K. Hu, T. Kuai, and S. L. Waslander, "Point Density-Aware Voxels for LiDAR 3D Object Detection", In: *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, doi: 10.1109/CVPR52688.2022.00828.

[26] F. Z. Rahmanti et al., "Enhancing Voxel-Based Human Pose Classification Using CNN with Modified VGG16 Method", In: *Proc. of IEEE International Conference on Information Technology and Digital Applications (ICITDA)*, 2023.

[27] M. Vasileiadis, C. S. Bouganis, and D. Tzovaras, "Multi-person 3D pose estimation from 3D cloud data using 3D convolutional neural networks", *Computer Vision and Image Understanding*, Vol. 185, pp. 12–23, 2019, doi: 10.1016/j.cviu.2019.04.011.

[28] D. Titisari et al., "Enhancing Breast Cancer Detection: Optimizing YOLOv8's Performance Through Hyperparameter Tuning", In: *Proc. of International Conference on Information Technology and Digital Applications (ICITDA)*, 2023, doi: 10.1109/ICITDA60835.2023.10427255.

[29] C. Kaul, N. Pears, and S. Manandhar, "FatNet: A Feature-attentive Network for 3D Point Cloud Processing", In: *Proc. of 25th International Conference on Pattern Recognition (ICPR)*, pp. 7211–7218, 2021, doi: 10.1109/ICPR48806.2021.9412731.

[30] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds", *ACM Trans. Graph*, Vol. 1, No. 1, 2019, doi: 10.1145/3326362.

[31] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points", In: *Proc. of 32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[32] A. Liang, H. Hua, J. Fang, H. Zhao, T. Liu, "Boosting 3D point-based object detection by reducing information loss caused by discontinuous receptive fields, " *International Journal of Applied Earth Observation and Geoinformation*, Vol. 132, pp. 104049, 2024, doi: 10.1016/j.jag.2024.104049.

[33] Y. Xu, M. Xu, Y. Wang, B. Li, "NCFDet: Enhanced point cloud features using the neural collapse phenomenon in multimodal fusion for 3D object detection", *Journal of Computational Design and Engineering*, Vol. 12, No. 1, 2025, pp. 300–311, doi: 10.1093/jcde/qwae115.

[34] V. Singh, V. S. Yadav, M. Kumar, and N. Kumar, "Optimization and validation of solar pump performance by MATLAB Simulink and RSM", *Evergreen*, Vol. 9, No. 4, pp. 1110–1125, 2022, doi: 10.5109/6625723.

[35] S. M. S. Shruthi, V. Jain, G. K. Kumar, and Z. Z. Kha, "Using artificial intelligence (AI) and internet of things (IoT) for improving network security by hybrid cryptography approach", *Evergreen Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, Vol. 10, No. 2, pp. 1133–1139, 2023.

[36] A. Chaudhary and P. Verma, "Road surface quality detection using lightweight neural network for visually impaired pedestrian", *Evergreen Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, Vol. 10, No. 2, pp. 706–714, 2023.

[37] P. Panwar, P. Roshan, R. Singh, M. Rai, and A. R. Mishra, "DDNet: A deep learning approach to detect driver distraction and drowsiness", *Evergreen Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, Vol. 9, No. 3, pp. 881–892, 2022.

[38] M. Ayundyahrini, D. A. Susanto, H. Febriansyah, F. M. Rizanulhaq, and G. H. Aditya, "Smart farming: Integrated solar water pumping irrigation system in Thailand", *Evergreen Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, Vol. 10, No. 1, pp. 553–563, 2023.

[39] L. K. Sagar and D. B. Das, "Fuzzy expert system for determining state of solar photovoltaic power plant based on real-time data", *Evergreen Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, Vol. 9, No. 3, pp. 870–880, 2022.