

Vehicle Classification Using Digital Image Processing Based on Convolutional Neural Network for The Electronic Ticketing System in Indonesia

Namira Fasya Rahim^{1, a)} Nathaniel Syalomta^{2, b)} Koredianto Usman^{3, c)} Nor Kumalasari Caecar Pratiwi^{4, d)}

Author Affiliations

^{1,2,3,4}Telkom University, Jl. Telekomunikasi No. 1, Bojongsoang, Bandung Regency, West Java, Indonesia 40257

Author Emails

^{a)} Corresponding author: namirafr@student.telkomuniversity.ac.id

^{b)} nathanielgtg@student.telkomuniversity.ac.id

^{c)} korediantousman@telkomuniversity.ac.id

^{d)} nkcp@telkomuniversity.ac.id

Abstract. Traffic violations can lead to traffic congestion or even collisions. However, Indonesia's traffic fining system is still incompatible because officers often take illegal levies. In the development of modern society, a system that can support electronic based to facilitate the community's needs in the field of transportation, especially in the identification and classification system of vehicle types, is needed. Therefore, this study proposed a system that can classify vehicle types into four classes: motorcycles, cars, trucks, and buses using a Convolutional Neural Network (CNN) method for feature extraction and classification with a self-constructed architecture named NN-Net that consist of three hidden layers and AlexNet architecture which has eight hidden layers. The design model uses various input image sizes, optimizers, and learning rates to get the best model. We achieved an accuracy of 99,14% using the AlexNet architecture with an optimum hyperparameter combination model. Based on our results, the proposed system can be helpful to solve Indonesia's ticketing system problem.

INTRODUCTION

The increase in traffic violation cases is still a challenge because it can lead to congestion and traffic accidents. One of the traffic violations that still often occurs in Indonesia is road marking violation. Road marking violation is Indonesia's third most frequent violation after drivers not wearing seat belts and traffic light violations throughout 2019 to 2020 [1]. However, civilian and police officers often twist the ticketing system to compromise and achieve their benefits without following the prevailing procedures. Therefore, an electronic ticketing system should be implemented to support transparent law enforcement to pay ticket fines and be more effective. To support implementing the electronic ticketing system in Indonesia, it is necessary to identify the type of vehicle on the road marking.

Related Works

Previously, many studies identified vehicle types using machine learning and deep learning, one of which was the research that classified vehicle types using the Extreme Learning Machine method. The data are images taken at two meters from the car and one meter from the motorcycle. The number of images used is 80 (50 training data and 30 test data). From 15 images of cars and 15 images of motorcycles at the testing scene, the results achieved an

accuracy of 86.6% [2]. Then, another study recognizes the types of vehicles using the Convolutional Neural Network (CNN) method with the model containing a convolution layer (layer_conv_2d), pooling layer, and dropout flattens and dense layer. The data used is 120 images consisting of three classification classes: cars, motorcycles, and bicycles. The outcomes show an accuracy of 73.3% at the testing part [3]. In the following research, car types were classified based on shape using the CNN method, consisting of three layers: Convolutional layer, Max pooling layer, and Fully connected layer. The data used are 600 images (180 test data images and 420 training data images) classified into three car classes, namely sedans, MPVs, and SUVs. This CNN method can identify the type of car with an accuracy of 94.4% [4].

From the several studies before, the first study only classifies vehicles into cars and motorcycles using the Extreme Learning Machine method with an accuracy rate of 86.6%. The second study identified three types of vehicles using CNN, but the accuracy was only 73.33%. Finally, in the third study, the CNN method was used for classifying digital images and obtained an accuracy rate of 94.4%. However, the classification carried out was only divided into three classes for types of cars. Based on this information, we will propose research that can complement the weaknesses of the previous research. The system created in this study can identify the type of vehicle divided into four classes, namely motorcycles, cars, trucks, and buses, using deep learning with the Convolutional Neural Network method. This study aims to have an accuracy level above 95% to help implement Indonesia's electronic ticketing system.

Convolutional Neural Network

Convolutional Neural Network (CNN) is a deep learning method specializing in processing data images. CNN uses a convolution process by moving a convolution kernel or filter to an image. CNN architecture resembles the pattern of connections of neurons or nerve cells in the human brain.

CNN consists of 2 parts: Feature Learning and Classification [5]. The Feature Extraction Layer section is divided into three layers: Convolutional Layer, Rectified Linear Unit (ReLU) Activation Layer, and Pooling Layer. While in the Classification section, the stages are divided into Fully Connected Layer and Softmax.

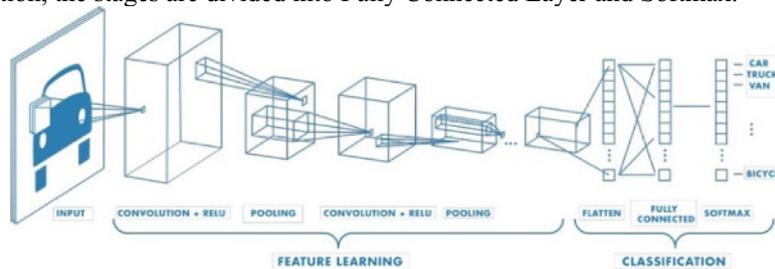


FIGURE 1. Convolutional Neural Network Diagram Block

Convolutional Layer

The convolutional layer is the first layer in the feature extraction layer. In this layer, the input values will be combined and filtered and then produce a feature map and forwarded to the next layer [6]. Parameters that affect the convolutional layer are filter size, stride, and padding.

Pooling Layer

The function of this layer is to progressively reduce the spatial size of the representation, reduce the number of parameters, and control overfitting [6]. Figure 1 illustrates of the Max Pooling Layer and Average Pooling Layer.

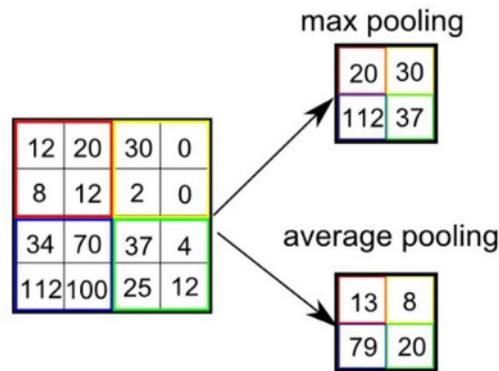


FIGURE 2. Illustrations of Max Pooling Layer and Average Pooling Layer.

ReLU Activation Layer

The ReLU activation function was initially used for the convolutional layer [7]. This layer changes the negative pixel value to 0 on the feature map [8]. The ReLU activation function is given in Equation 2.1 as follows [6].

$$f(x) = \max(0, x). \quad (1)$$

Fully Connected Layer

A Fully Connected Layer carried out flatten operation, which converted a multidimensional array into a one-dimensional array. This layer connects all neurons as in the nervous system from the previous layer to determine which features correlate with a particular class [9].

Softmax

Softmax is another form of logistic function commonly used for classification methods with more than two classes. Softmax is the last activation function to normalize the network output to a likelihood distribution on the predicted output class [10].

SYSTEM OVERVIEW

Proposed System Design

In this study, we used two different architectures. The first architecture is a self-constructed model consisting of an input layer, three hidden layers, and a flattened layer named NN-Net. The other architecture is a well-known AlexNet model.

NN-Net

This self-constructed model consists of 4 layers. The first three layers are hidden layers, then supported by one fully connected layer. The ReLU activation function and a max-pooling layer followed the first three convolutional layers in the hidden layers. The NN-Net architecture is shown in Figure 2.

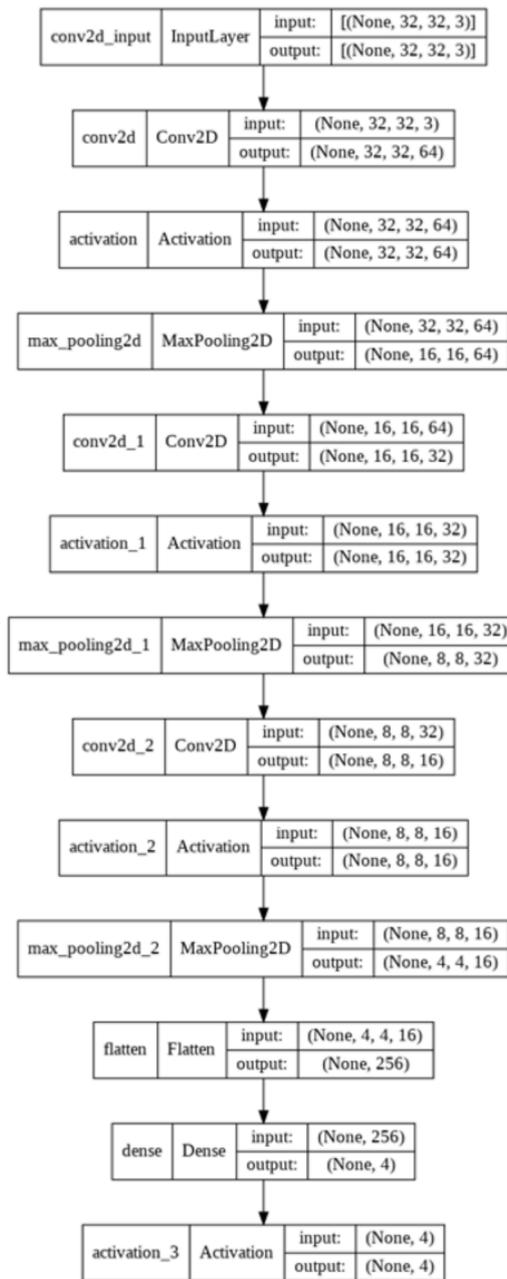


FIGURE 3. Block Diagram of the NN-Net Architecture.

AlexNet

AlexNet consists of 8 layers, where five layers are convolutional, and the following three are fully connected layers [11]. Three convolutional layers are followed by a max-pooling layer and two convolutional layers that do not use a pooling layer in the first five layers. The AlexNet architecture is represented in Figure 3.

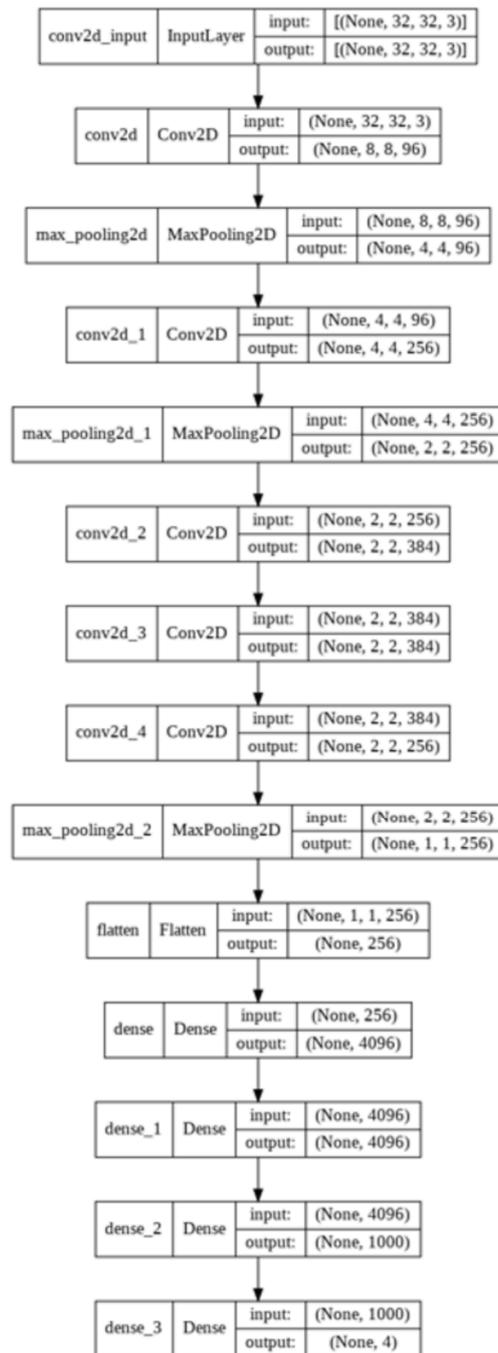


FIGURE 4. Block Diagram of the AlexNet Architecture.

Dataset

The dataset we used in this study is a combination of three sources: Vehicle Image Dataset, BIT Vehicle Dataset, and Indonesia Motorcyclist Dataset. All three sources are RGB labelled images. We use a total of 1400 images with 350 each class (motorcycle, cars, trucks, and buses). The data is split into 75% of training data and 25% for testing. It is fundamental to complete the testing stage in neural networks to ensure consistent results. The following images are an example of vehicle type from the image acquisition, which can be seen in Figure 2.



FIGURE 5. Dataset Example from Each Class (Motorcycle, Cars, Trucks, and Buses)

Evaluation Parameter

Confusion Matrix

A confusion matrix is a method that can be used to measure the performance of a classification system [12]. The confusion matrix contains information about the classification predicted by the system, as shown in Table 1.

TABLE 1. Confusion Matrix

Confusion Matrix	Actual A	Actual B	Actual C	Actual D
Predicted A	x_{AA}	x_{AB}	x_{AC}	x_{AD}
Predicted B	x_{BA}	x_{BB}	x_{BC}	x_{BD}
Predicted C	x_{CA}	x_{CB}	x_{CC}	x_{CD}
Predicted D	x_{DA}	x_{DB}	x_{DC}	x_{DD}

$$TP_{all} = \sum_{j=A}^D x_{jj} \quad (2)$$

$$TN_i = \sum_{\substack{j=A \\ j \neq i}}^D \sum_{\substack{k=A \\ k \neq i}}^D x_{jk} \quad (3)$$

$$FN_i = \sum_{\substack{j=A \\ j \neq i}}^D x_{ij} \quad (4)$$

$$FP_i = \sum_{\substack{j=A \\ j \neq i}}^D x_{ji} \quad (5)$$

The contents of the confusion matrix table are 4, namely [12]:

1. True Positive (TP) is when the prediction results say that the class is true and the actual class is true.
2. True Negative (TN) is when the prediction results say that the class is false and the actual class is false.
3. False Negative (FN) is when the prediction results say that the class is false, but the actual class is true.
4. False Positive (FP) is when the prediction results say that the class is true, but the actual class is false.

Accuracy

Accuracy is a comparison between the data that is predicted correctly with the total predicted data [12]. The accuracy equation can be seen in Equation 6.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

Loss

We use a Categorical Cross-Entropy (CCE) loss function because of our multi-class classification system. Using this loss function, we will train a CNN to have an output of a probability over the classes for each image [13]. The standard categorical cross-entropy loss is given by Equation 7:

$$J_{cce} = -\frac{1}{M} \sum_{k=1}^K \sum_{m=1}^M y_m^k \times \log(h_{\theta}(x_m, k)) \tag{7}$$

Where:

5. M represent the number of training examples
6. K represent the number of classes
7. y_m^k represent the target label for training example m for class k
8. x represent the input for training example m
9. h_{θ} represent the model with neural network weights θ

RESULT AND DISCUSSION

In this study, we state that the batch size is 30, and the Epoch value is 100 in consideration of the achievement from the stabilization model in the previous simulation. We executed the same hyperparameter combination scenario into the two architectures. Using a hierarchical rule, we obtained the results from training the different input sizes followed by different optimizers and then the varied learning rate value. The table below summarizes the result of our accuracy and loss level.

TABLE 2. Input Image Size as Parameter for Result Comparison between NN-Net and AlexNet.

Input Image Size	Architecture	Accuracy	Loss
32 x 32	NN-Net	<u>0.9857</u>	<u>0.0528</u>
	AlexNet	0.9685	0.1451
64 x 64	NN-Net	0.9771	0.1371
	AlexNet	<u>0.9914</u>	<u>0.0181</u>
128 x 128	NN-Net	0.9799	0.1028
	AlexNet	0.9799	0.067

TABLE 3. Optimizer as Parameter for Result Comparison between NN-Net and AlexNet.

Optimizer	Architecture	Accuracy	Loss
Adam	NN-Net	<u>0.9857</u>	<u>0.0528</u>
	AlexNet	0.9685	0.1451
Nadam	NN-Net	0.9799	0.1267
	AlexNet	0.2264	1.3878
RMSprop	NN-Net	0.9857	0.0691
	AlexNet	<u>0.9943</u>	<u>0.065</u>

SGD	NN-Net	0.7822	0.5763
	AlexNet	0.9255	0.2473

TABLE 4. Learning Rate as Parameter for Result Comparison between NN-Net and AlexNet.

Learning Rate	Architecture	Accuracy	Loss
0.1	NN-Net	0.2722	1.3842
	AlexNet	0.2722	1.3842
0.01	NN-Net	0.9799	0.2422
	AlexNet	0.2493	1.3873
0.001	NN-Net	0.9857	0.0528
	AlexNet	0.9685	0.1451
0.0001	NN-Net	0.9857	0.0536
	AlexNet	0.9914	0.0347
0.00001	NN-Net	0.8281	0.5959
	AlexNet	0.9713	0.0817

As the given results above, we can see that a smaller (32x32) Input image size in NN-Net can already give an excellent result with an accuracy of 98,57%. In comparison, AlexNet Architecture needs a bigger input image size (in this case, 64x64) to have a more suitable outcome. We can also state that the Adam optimizer works best for the NN-Net. On the other hand, 99,43% of accuracy was achieved when AlexNet Architecture applied the RMSprop optimizer. The value 0.001 of Learning Rate performs better in the NN-Net model at the last hyperparameter combination scenario. However, with the value 0.0001 of Learning Rate, AlexNet architecture works better than the NN-Net model with an accuracy of 99,14%.

The highest accuracy of 99,43% was achieved using AlexNet architecture with the hyperparameter listed below:

- 10. Input image size: 32x32
- 11. Optimizer: RMSprop
- 12. Learning Rate: 0.001

Nevertheless, the graph of the model shows an overfitting and poor stability pattern.

From the engineer's point of view, the slightest differences matter in terms of significant extra effort. Hence, in this case, the model that we can say the best is the scenario that ran with the hyperparameter listed below:

- 13. Architecture: AlexNet
- 14. Input image size: 32x32
- 15. Optimizer: Adam
- 16. Learning Rate: 0.0001

With an accuracy of 99,14% and only 0,29% differences from the highest accuracy achieved but create a much more stable and fit model.

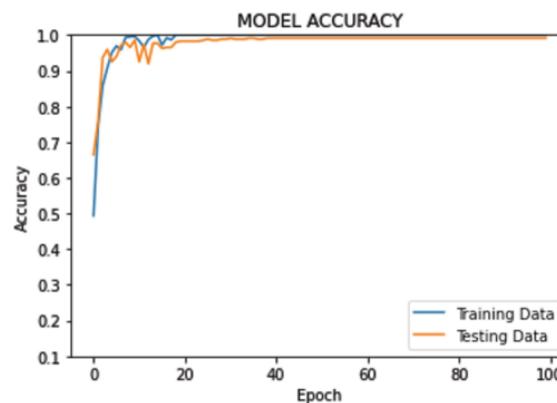


FIGURE 6. Graph of The Model Accuracy from The Best Hyperparameter Combination Scenario.

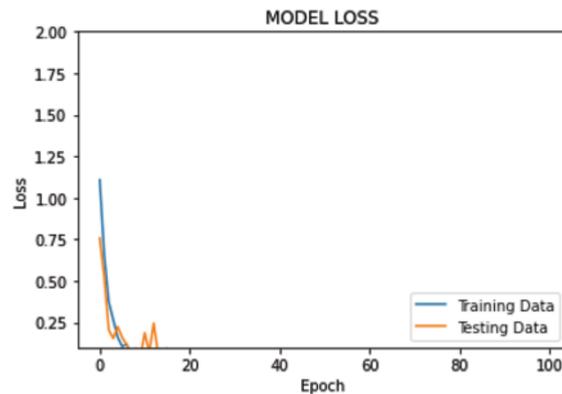


FIGURE 7. Graph of The Model Loss from The Best Hyperparameter Combination Scenario.

CONCLUSION

This study proposed a vehicle types classification system using CNN. Our system was able to recognize four types of vehicle types with a high accuracy outcome. Still, the achievement accuracy varied depending on the hyperparameter combination.

AlexNet performed better in this study with the minimum effort and delivered an optimum outcome. However, the highest accuracy was achieved from a different scenario because of the poor result of the model. Therefore, we can conclude that the best-proposed model hyperparameter combination is 32x32 input image size, adam optimizer, and the value of Learning Rate is 0.0001 for generating a more stable and fit model.

REFERENCES

1. S. P. Yogo, Interviewee, *Kabar Baik! Terapkan Tilang ETLE, Polda Metro Jaya Berhasil Turunkan Angka Pelanggaran Lalu Lintas*. [Interview]. 26 March 2021.
2. H. Rispani and R. A. Johan, "Klasifikasi Jenis Kendaraan Menggunakan Metode Extreme Learning Machine," *JTIM : Jurnal Teknologi Informasi dan Multimedia*, vol. 2, no. 4, pp. 273-243, 2021.
3. N. Fadlia and R. Kosasih, "Klasifikasi Jenis Kendaraan Menggunakan Metode Convolutional Neural Network (CNN)," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 24, no. 3, pp. 207-215, 2019.
4. D. Fiolita, *Identifikasi Jenis Mobil Berdasarkan Bentuk Dengan Menggunakan Metode Convolutional Neural Network (CNN)*, Medan: Universitas Sumatra Utara, 2020.
5. R. M. Awangga and N. A. Batubara, *Tutorial Object Detection Plate Number With Convolutional Neural Network (CNN)*, Bandung: Telkom University, 2020.
6. K. Zakka, "CS231n Convolutional Neural Networks for Visual Recognition," Stanford University, 9 January 2021. [Online]. Available: <https://cs231n.github.io/convolutional-network/#add>. [Accessed 12 November 2021].
7. N. Ageishi, F. Tomohide and A. A. Ben, "Real-time Hand-Gesture Recognition based on Deep Neural Network," in *SHS Web of Conferences*, Aizu-Wakamatsu, 2021.
8. I. W. Suartika, A. Y. Wijaya and R. Soelaiman, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101," *JURNAL TEKNIK ITS*, vol. 5, no. 1, pp. A65-A69, 2016.
9. E. N. Arrofiqoh and Harintaka, "The Implementation of Convolutional Neural Network Method for Agricultural Plant Classification in High Resolution Imagery," *Geomatika*, vol. 24, no. 2, pp. 61-68, 2018.
10. S. Ilayah and A. Nilogiri, "Implementasi Deep Learning Pada Identifikasi Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *JUSTINDO (Jurnal Sistem & Teknologi Informasi Indonesia)*, vol. 3, no. 2, pp. 49-56, 2018.

11. A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti and D. De, Fundamental Concepts of Convolutional Neural Network, India: ResearchGate, 2020.
12. B. Y. Phiadelvira, Klasifikasi Kanker Serviks Berdasarkan Citra Kolposkopi Menggunakan Convolutional Neural Network (CNN) Model AlexNet, Surabaya: Universitas Islam Negeri Sunan Ampel, 2021.
13. Y. Ho and S. Wookey, The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling, California: IEEE Access, 2017.