TEM Journal. Volume 13, Issue 1, pages 531-536, ISSN 2217-8309, DOI: 10.18421/TEM131-55, February 2024.

Traffic Violation Detection System on Two-Wheel Vehicles Using Convolutional Neural Network Method

Kusworo Adi [1], Catur Edi Widodo [1], Aris Puji Widodo [2], Fauzan Masykur [3]

[1] *Department of Physics, Faculty of Science and Mathematics, Diponegoro University, Indonesia*
[2] *Department of Informatics, Faculty of Science and Mathematics, Diponegoro University, Indonesia*
[3] *Department of Informatics Enginering, Muhammadiyah University of Ponorogo, Indonesia*

***Abstract* – The number of vehicles is increasing every year, and along with it, the number of traffic violations is also rising.. Traffic violations are one of the causes of traffic accidents. Currently, traffic violation detection still uses conventional methods, involving the police to take action. Preliminary research on traffic violation detection by several researchers mostly uses the Yolo Method. The study aims to design a traffic violation detection system for two-wheeled vehicles using the Convolutional Neural Network (CNN). In this research, the CNN method was used with the Faster RCNN architecture. Faster R-CNN is composed of convolution layers, Relu, and pooling layers which are used to extract features from images. An image in the size of 3264 x 1836 pixels, with the type of marking violation and helmet use was used as a sample. The number of images used was 660 images with 600 images for training and 60 images for testing. The system will detect traffic violations on two-wheeled vehicles, namely helmet use violations and road marking violations. This traffic violation detection system for two-wheeled vehicles produces the highest accuracy, namely 85% with a maxpooling kernel size value of 1x1, stride 1 and a learning rate of 0.003.**

This research has the potential to be applied to areas that are less accessible to the police, because the system will record and analyze violations.

***Keywords* – Traffic violations, convolutional neural network.**

DOI: 10.18421/TEM131-55
https://doi.org/10.18421/TEM131-55

Corresponding author: Kusworo Adi,
Department of Physics, Faculty of Science and Mathematics, Diponegoro University, Indonesia
Email: kusworoadi@lecturer.undip.ac.id

Received: 25 October 2023.
Revised: 07 December 2023.
Accepted: 13 December 2023.
Published: 27 February 2024.

1. Introduction

The learning process in machine learning involves analyzing data by applying specific labels to it. Deep learning applications have been widely applied in various fields that utilize artificial intelligence in computer vision systems [1]. Convolutional Neural Network (CNN) is a type of neural network that is commonly used on image data to recognize objects. CNN consists of neurons that have weight, bias, and activation function. Convolutional layers also consist of neurons arranged in such a way as to form a filter [2].

Rashid *et al.* used the CNN model with VGG-19 and Inception V3. [3], [4], in this study four different types of datasets were used, namely Caltech-101, Bird database, Butterfly database, and CIFAR-100. Each dataset has an accuracy of 95.5%, 100%, 98%, and 68.80%. Then Saini *et al.* [5] classified apple and grape leaves using Convolutional Neural Network (CNN) with the Plant Village dataset which is part of the Plant Village Disease Classification. This research was carried out using CNN with variations parameters in epoch and convolution kernel size, resulting in a fairly high level of accuracy of 99.90%. This research aims to create a system for detecting traffic violations in two-wheeled vehicles, violations of the obligation to use helmets and road markings.

1.1. Deep Learning

Deep learning was first introduced in 2006. However, it did not gain much attention until 2012.

The current development of computing with big data has had an impact on data processing using deep learning. Deep learning methods can perform feature extraction at a high structural level and are very effective. Significant progress has been made in solving various problems in the artificial intelligence community, especially in areas where the data is multidimensional and features difficult to perform traditionally or hand-engineered, including speech or voice recognition, natural language processing, and computer vision. Deep learning is also showing dominance in fields such as business analysis, medical diagnostics, art creation, and image translation. Because it requires little human input, deep learning will continue to have a greater impact in the future with increased computing power and explosive data growth [6].

1.2. Artificial Neural Networks (ANN)

An artificial neural network (ANN) is a system designed to process information by imitating the way the human brain works. Artificial neural networks solve a problem by carrying out a learning process through changing the synapse weights. Examples of cases that can be solved with artificial neural networks are image classification, object detection, image text, example segmentation, voice recognition, and machine translation [7].

ANN is composed by artificial neurons (neurons) that apply simple functions to input data. The output by artificial neurons is called activation. Artificial neurons can continue their output to the next neuron, forming a directed graph structure neuron, mapping input and output [7]. ANN is a computational network (a system of nodes and interconnections between nodes) inspired by the biological properties of neural networks, which are complex networks of neurons in the human brain. The nodes created in the ANN are programmed to behave like real neurons, so they are called artificial neurons [8].

CNN is a deep learning model specifically designed to analyze and understand image data with many rigorously trained layers. CNN has proven to be very effective and is also the most widely used in various computer vision applications. CNN generally consists of three main neural layers, namely convolutional layers, pooling layers, and fully connected layers [1]. There are two stages in conducting network training, namely the forward and the backward stage. First, at this stage the input image is processed with weight and bias parameters at each layer. Then the output predictions are used to calculate the cost of losses with basic truth labels. Second, based on loss costs, the backward step calculates the gradient of each parameter with the chain rule.

All parameters will be updated according to their gradient values which will be used for further computation. After sufficient iteration of the forward and backward stages, network learning can be stopped [1].

1.3. Faster RCNN

Faster RCNN introduces the concept of a region proposal network (RPN). RPN will identify objects in the input image, then the RCNN architecture will perform selective search elimination to search for proposal regions. In Faster RCNN, RPNs share a feature extractor layer with the Fast RCNN architecture, which branches out in aggregated region of interest (ROI) feature maps [7].

This Fast RCNN replaces the slow selective search algorithm used in Fast RCNN. With the RPN, CNN completely predicts regional proposals. First, a set of box anchors, which are rectangles, are generated around the object. In the second step, the loss function is applied to calculate the possible errors. Finally, the backbone network will generate a feature mask and the RPN proposes a series of proposal areas. This set of proposals is sent as input to the next layer, namely ROI in the pooling layer. The pooling layer converts the features obtained from the refinement of the CNN layer into a fixed-sized feature map. Finally, the classification layer predicts the class, while bounding box regression creates a rectangular box surrounded by objects for localization [9], [10].

This research is a development of previous research using machine learning methods. In previous research, machine learning methods have been widely applied to the health and transportation sectors [11], [12], [13], [14], [15]. So that previous research strongly supports this research which is applied to the transportation sector, especially for the detection of traffic violations [16], [17], [18], [19], [20].

2. Material And Method

The tools used in this study were a 13 MP camera and a computer with Intel Core i3-4005U, 1.7GHz, NVIDIA GeForce GT930M 2GB GPU, 250 GB SSD, 500 GB HDD. The data used are 660 traffic images. 50% of the total data are images of violations of the mandatory use of helmets and 50% are images of violations of road markings. The total time needed to capture this traffic image is 6 weeks. Images of violating road markings and violating the mandatory use of helmets were taken in a city in Indonesia. From these data, 600 images were used as training and 60 images as tests.

Software and libraries used include Python 3.6.6, Tensorflow-GPU 1.15.0, Cudnn 7.4.1.5, Cuda 10.0.130, Labelimg 1.8.5, Numpy 1.19.5, Tesseract 4.0.0 and OpenCV 4.5.2.52.

There are two classes in the traffic violation detection system for two-wheeled vehicles, namely helmet violations and marking violations. This study uses the faster RCNN inception V2 model. In general, the research procedure that was carried out was as follows:

1. The dataset in the form of images of helmet violations and marker violations that have been collected will then be resized to 600 x 1024 pixels.
2. In the pre-training stage, there are several processes, namely the image of traffic violations using labels, image labeling is carried out, namely helmet violations and marking violations. The output of this labeling will generate an XML file. Then change the XML format to CSV. Then change the CSV dataset format to TFrecord so that the dataset can be processed in tensorflow. Then create a map label for each class id assignment (helmet violations and marker violations).
3. Before training the dataset, first set the parameters to be varied as shown in Table 1
4. Then the next process is training. After the training is complete, an export graph model is carried out which produces a frozen inference graph.
5. If the accuracy of detecting helmet violations and marking violations is still low, then return to the parameter setting stage. If the accuracy of detecting helmet violations and marking violations is high, then the next process will be continued.
6. Then do the test.
7. Output in the form of traffic violation detection results (helmet violations and marker violations).

Table 1. Scenario

Scenario	***Size of Kernel max polling***	***Stride***	***Learning rate***
Scenario I	2 x 2	2	0.0003
Scenario II	4 x 4	1	0.0002
Scenario III	1 x 1	1	0.003

3. Result and Discussion

In this section, the research results and discussion of the research results will be presented. The CNN method with the Fater RCNN architecture has been implemented for traffic violation detection.

3.1. CNN and Faster RCNN

The initial process of CNN is feature extraction. In this section there are several main layers, namely the convolution layer, the activation function and the pooling layer. The input from the convolution layer is an image. The output of the convolution layer is feature maps which are then performed with mathematical operations using the activation function which aims to add nonlinear properties to the network. The next layer is the pooling layer which serves to reduce the spatial size of the input representation. The benefits of the pooling layer are that it avoids overfitting the model to the input data, reduces computational complexity, and provides a basis for translation invariance to the input representation. The feature extraction process is the repetition of the main layer, starting with the convolution layer then the pooling layer and returning to the convolution layer then the pooling layer. The type of pooling layer used in this study is maxpolling. After going through feature extraction, the next layer is the fully connected layer. The network at the fully connected layer is connected to all the networks in the previous layer. In the fully connected layer, there are possible combinations of features in the previous layer. The last layer is the output layer which produces probabilities for each class.

Faster R-CNN consists of convolution layers, Relu, and pooling layers which are used to extract features from images. Then the extracted eigenvalues are used in the next RPN layer and the fully connected layer. RPN (Region Proposal Networks) is used to generate target area values. The anchor is the main component of the RPN network which will determine the target in the receptive field at each sliding kernel center. Differences in target size and length ratio will cause kernel shifts of various sizes. Therefore, the anchor will provide a kernel size reference by obtaining different kernel sizes based on multiples and the ratio of length and width.

The training process on the RPN network uses ground truth to assess an anchor is a target which is expressed as 0 or 1. If the Intersection-over-Union (IoU) of the anchor in each target area is above 0.7, then the anchor is included in the target area. If the anchor IoU in each target area is below 0.3, then the area is included in the background. IoU is the range of the predicted box and the real box which has a value equal to the intersection of the two boxes divided by the combination of the two boxes. Once the object region is determined, bounding box regression is used to modify the anchors so as to obtain an accurate proposal.

The ROI in the pooling layer determines the feature map and input proposal, then the feature map extraction is carried out and continued at the next layer to determine the target category. The next process is classification, in this process the feature map is used to calculate categories and regression on the bounding box to get the final location of the detection box.

3.2. Training

In this study, three training sessions were carried out with each training number of steps, namely 60,000. List of parameter variations in presented in Table 1. The first training uses scenario I, the second training uses scenario II and the third training uses scenario III. Figures 1, 2, and 3 portray a graphical image of loss in the three trainings:

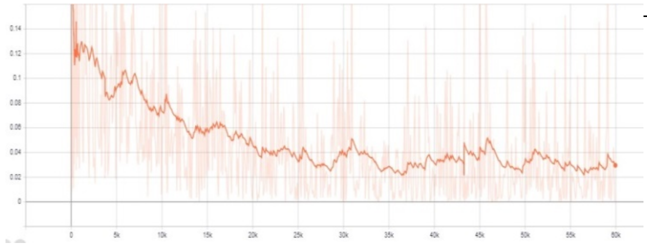

Figure 1. Graph of loss in training to one

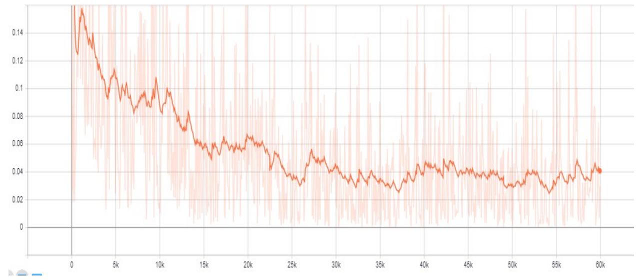

Figure 2. Graph loss in the second training

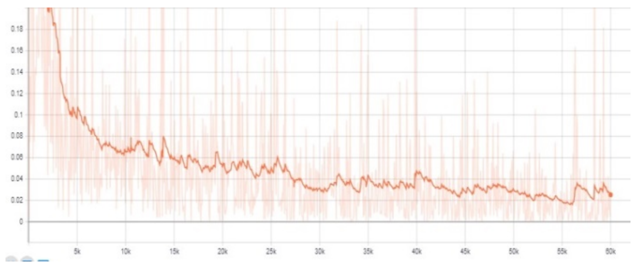

Figure 3. Graph loss in the third training

Based on the three graphs, training has similarities, which shows that there is a decrease in the loss value. More steps will result in a decrease in the loss value. On the first and second training graphs, the loss value decreases gradually. Meanwhile, the third training graph shows a drastic reduction in loss values.

3.3 Testing

Parameters varied in this study were maxpolling kernel size, stride (kernel shift step) and learning rate (learning rate). Results of the classification of each scenario can be seen in Tables 2, 3 and 4, as well as the accuracy can be seen in Table 5.

Table 2. Result of scenario I

No.	*Actual conditions*	*Detection results*
1	Not wearing a helmet = 27	Helmet violation = 21 Not detected = 6
2	Wearing a helmet = 3	Helmet violation = 3 Not detected = 0
3	Being outside the marker = 29	Violation of the marker = 29 Not detected = 0
4	Being in the marker = 1	Violation of the marker = 1 Not detected = 0

Table 3. Result of scenario II

No.	*Actual conditions*	*Detection results*
1	Not wearing a helmet = 27	Helmet violation = 21 Not detected = 6
2	Wearing a helmet = 3	Helmet violation = 3 Not detected = 0
3	Being outside the marker = 29	Violation of the marker = 29 Not detected = 0
4	Being in the marker = 1	Violation of the marker = 1 Not detected = 0

Table 4. Result of scenario III

No.	*Actual conditions*	*Detection results*
1	Not wearing a helmet = 27	Helmet violation = 21 Not detected = 6
2	Wearing a helmet = 3	Helmet violation = 3 Not detected = 0
3	Being outside the marker = 30	Violation of the marker = 30 Not detected = 0
4	Being in the marker = 0	Violation of the marker = 0 Not detected = 0

Table 5. Accuracy of each scenario

Scenario	*Accuracy*
Scenario I	83,33%
Scenario II	83,33%
Scenario III	85%

Scenario I and scenario II have the same accuracy of 83.33%. The learning rate parameter in scenario I is 0.0003 while in scenario II it is 0.0002. The learning rate difference between the two is 0.0001.

This does not affect the level of accuracy because the difference in learning rate between scenario I and scenario II is small. In scenario I and scenario II, the size of the maxpolling kernel is 2 x 2 and 4 x 4. The size of the maxpolling kernel is larger than in scenario III, which is 1 x 1. Meanwhile, the accuracy in scenario III is greater, namely 85% compared to the accuracy of scenario I and scenario II, namely 83.33%. This shows that the smaller the maxpolling kernel size, the resulting accuracy increases. According to Pan *et al*. [10], the accuracy value decreases with increasing maxpolling kernel size. The following is an example of an image resulting from traffic violation detection:

Figure 4. Helmet violation detection with false (left object) and correct (right object) detection results

Figure 5. Helmet violation detection with correct detection results

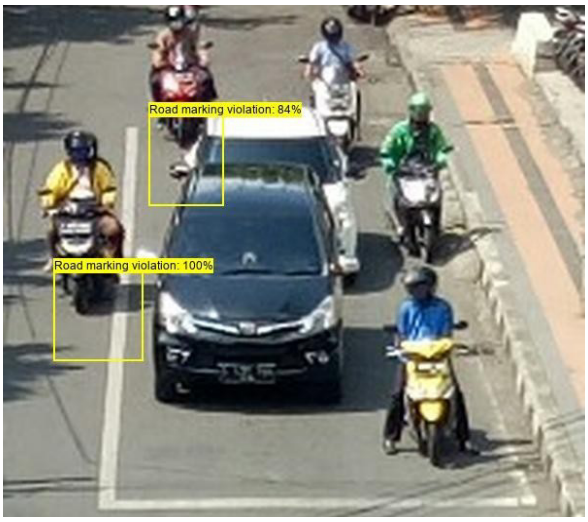

Figure 6. Mark violation detection with correct (left object) and false (right object) detection results

Figure 7. Helmet violation detection with correct and correct detection results

4. Conclusion

The traffic violation detection system for two-wheeled vehicles that has been carried out can produce the highest accuracy of 85% with a maxpooling kernel size value of 1x1, stride 1 and a learning rate of 0.003. Based on the research that has been done, there are suggestions for further research, namely determining the variation in the learning rate value with non-adjacent intervals and setting the appropriate spacing when taking the image of the object to be detected.

The weakness of the results of this research is that it really depends on the camera position, so the next research development should consider providing improvements to the algorithm that is adaptive to several camera positions.

The research possesses great importance in order to enforce vehicle driver discipline. In the future studies, this system can be integrated with the police enforcement system.

Acknowledgment

This research was funded by the Development and Implementation Research Scheme, Diponegoro University in 2021 with Number: 233-110/UN7.6.1/PP/2021

References:

[1]. Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S. and Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187, 27-48.

[2]. Aloysius, N., & Geetha, M. (2017). A review on deep convolutional neural networks. In *2017 international conference on communication and signal processing (ICCSP)*, 0588-0592. IEEE.

[3]. Rashed, H., Yogamani, S., El-Sallab, A., Hassaballah, M. and ElHelw, M., (2020). Deep Semantic Segmentation in Autonomous Driving, In *Deep Learning in Computer Vision: Principles and Applications*, 151.

[4]. Rashid, M., Khan, M. A., Alhaisoni, M., Wang, S. H., Naqvi, S. R., Rehman, A., & Saba, T. (2020). A sustainable deep learning framework for object recognition using multi-layers deep features fusion and selection. *Sustainability, 12*(12), 5037.

[5]. Saini, G., Khamparia, A., & Luhach, A. K. (2020). Classification of plants using convolutional neural network. In *First International Conference on Sustainable Technologies for Computational Intelligence: Proceedings of ICTSCI 2019*, 551-561. Springer Singapore.

[6]. Li, K., Ma, W., Sajid, U., Wu, Y. & Wang, G., (2019). Object detection with convolutional neural networks. *arXiv.org.*

[7]. Pinto, P.C.C. (2019). *Implementation of Faster R-CNN Applied to the Datasets COCO and PASCAL VOC*. Rio de Janerio: COPP UFJR.

[8]. Manaswi, N. K., Manaswi, N. K., & John, S. (2018). *Deep learning with applications using python*, 31-43. Berkeley, CA, USA: Apress.

[9]. Patel, S. & Patel, A. (2021). Object detection with convolutional neural networks. *Machine Learning for Predictive Analysis*, 529-539.

[10]. Pan, S., Guan, H., Chen, Y., Yu, Y., Gonçalves, W. N., Junior, J. M. & Li, J. (2020). Land-cover classification of multispectral LiDAR data using CNN with optimized hyper-parameters. *ISPRS Journal of Photogrammetry and Remote Sensing*, *166,* 241-254.

[11]. Usman, C. D., Widodo, A. P., Adi, K., & Gernowo, R. (2023). Rainfall prediction model in Semarang City using machine learning. *Indonesian Journal of Electrical Engineering and Computer Science*, *30*(2), 1224-1231.

[12]. Setiawan, A., Adi, K., & Widodo, C. E. (2023). Rice Foreign Object Classification Based on Integrated Color and Textural Feature Using Machine Learning. *Mathematical Modelling of Engineering Problems*, *10*(2).

[13]. Adi, K., Widodo, C.E., Widodo, A.P. (2021). Design and implementation of traffic violation detection systems with deep learning to support electronic traffic law enforcement (e-TLE). *ARPN Journal of Engineering and Applied Sciences, 16*(10), 1062–1070.

[14]. Adi, K., Widodo, C. E., Widodo, A. P., & Margiati, U. S. R. I. (2022). Detection of foreign object debris (Fod) using convolutional neural network (CNN). *J. Theor. Appl. Inf. Technol.*, *100*(1), 184-191.

[15]. Kushwaha, S., Bahl, S., Bagha, A. K., Parmar, K. S., Javaid, M., Haleem, A., & Singh, R. P. (2020). Significant applications of machine learning for COVID-19 pandemic. *Journal of Industrial Integration and Management*, *5*(04), 453-479.

[16]. Latif, J., Xiao, C., Imran, A., & Tu, S. (2019, January). Medical imaging using machine learning and deep learning algorithms: a review. In *2019 2nd International conference on computing, mathematics and engineering technologies (iCoMET)*, 1-5. IEEE.

[17]. Xing, L., Krupinski, E. A., & Cai, J. (2018). Artificial intelligence will soon change the landscape of medical physics research and practice. *Medical physics*, *45*(5), 1791-1793.

[18]. Hesamian, M. H., Jia, W., He, X., & Kennedy, P. (2019). Deep learning techniques for medical image segmentation: achievements and challenges. *Journal of digital imaging*, *32*, 582-596.

[19]. Kothai, G., Poovammal, E., Dhiman, G., Ramana, K., Sharma, A., AlZain, M. A., ... & Masud, M. (2021). A new hybrid deep learning algorithm for prediction of wide traffic congestion in smart cities. *Wireless Communications and Mobile Computing*, *2021*, 1-13.

[20]. Fujiyoshi, H., Hirakawa, T., & Yamashita, T. (2019). Deep learning-based image recognition for autonomous driving. *IATSS research*, *43*(4), 244-252.