Implementation of CNN and YOLO Algorithms to Detect Types of Vehicles on the Highway

Fernandy Jupiter[1], Edi Surya Negara[2], Yesi Novaria Kunang[3], M. Izman Herdiansyah[4]
Department of Computer Engineering, Faculty of Master of Computer Engineering, Bina Darma University, Palembang, Indonesia[1 2 3 4]
Fernandy3jupiter@gmail.com [1], e.s.negara@binadarma.ac.id [2], yesinovariakunang@binadarma.ac.id [3], m.herdiansyah@binadarma.ac.id [4]

***Abstract*—This research integrates the capabilities of two primary algorithms for vehicle type detection on highways, namely Convolutional Neural Network (CNN) and You Only Look Once (YOLO). The objective of this study is to assess the effectiveness of these two algorithms in recognizing various types of vehicles, including motorcycles, cars, trucks, and buses, within a road context. The research methodology involves the collection of datasets containing vehicle images, model training using CNN and YOLO architectures, and performance evaluation based on precision, recall, and F1-score metrics. The results demonstrate that the combined utilization of CNN and YOLO approaches yields a high level of accuracy in identifying vehicle types on highways. These findings hold promising applications in the development of intelligent traffic monitoring systems, traffic measurement, and the enhancement of road safety. This research makes a valuable contribution to the advancement of image processing technology and object detection in the realm of transportation**

***Keywords*— CNN; YOLO; precission; recall; F1-score**

I. INTRODUCTION

In the era of rapid information technology development, traffic monitoring systems have become increasingly crucial in managing vehicle flow on highways and optimizing transportation efficiency. One of the key elements in traffic monitoring systems is surveillance cameras on both highways and toll roads [1]. This monitoring camera is used to detect vehicles passing through a specific highway and recognize the types of vehicles that pass through it, providing crucial data for monitoring, law enforcement, and statistical data collection purposes. In the use of cameras on the highway, object recognition becomes one of the key aspects.

Two of the most popular and successful algorithms in object recognition are Convolutional Neural Network (CNN) and You Only Look Once (YOLO) [2]. CNN algorithms have been proven successful in various object recognition tasks, such as vehicle types [3], building [4] and human face [5]. This algorithm utilizes a neural network consisting of convolutional layers to learn relevant features from images and achieve object detection with a high level of accuracy. However, the weakness of the CNN algorithm lies in its detection speed [6]. On the other hand, the YOLO algorithm offers a different approach to object detection. YOLO treats object detection as a direct regression problem [7] This results in a high detection speed. However, as a consequence of this approach, YOLO may encounter challenges in terms of lower detection accuracy compared to CNN. The combination of these two algorithms can have a significant impact on the performance of the object recognition system on highway cameras. Therefore, it is necessary to implement both of these algorithms to assess their performance when combined.

In the research titled "Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate dan Mixup (Data Augmentation for Vehicle Image Recognition Using Random Crop, Rotate, and Mixup Approaches)," it is explained that using the CNN algorithm to enhance accurate vehicle model detection, techniques such as random crop, rotation, and mixup data augmentation can improve the performance of the ResNet model in terms of accuracy. However, the use of mixup can also increase the loss. Augmentation can assist the model in capturing more features and information for classification. The final results of this study, with a training epoch comparison of 14 for each method, show that the non-mixup method has a training loss of 0.513145, a validation loss of 0.703171, and an accuracy of 0.791234. Meanwhile, the mixup method has a training loss of 0.556042, a validation loss of 0.712334, and an accuracy of 0.824154 [3].

In the research titled "Deep Learning dalam Mengindentifikasi Jenis Bangunan Heritage dengan Algoritma Convolutional Neural Network (Deep Learning in Identifying Types of Heritage Buildings with Convolutional Neural Network Algorithm)," it explains that the study using CNN (Convolutional Neural Network) and KNN (K-Nearest Neighbor) algorithms focuses on problems caused by the lack of public knowledge in recognizing various types of heritage buildings, as well as the insufficient digital documentation available and the challenge of identifying heritage buildings with similarities between them. Therefore, with the deep learning method, this research demonstrates that the combination of CNN and KNN algorithms can be used for heritage building identification. This is shown by the test data resulting in an accuracy of 98%. The excellent performance of this method is mainly attributed to the feature extraction process using CNN, followed by classification with the KNN algorithm [4].

In the study titled "Eksperimen Pengenalan Wajah Dengan Fitur Indoor Positioning System Menggunakan Algoritma CNN (Face Recognition Experiment Using Indoor Positioning System Features with CNN Algorithm)," it is explained that manual attendance recording poses various problems, such as cheating through proxy attendance. The facial recognition process is capable of achieving good results; however, this is

not the case with the position estimation process. This is demonstrated by the research results that display a confusion matrix with a maximum testing accuracy of 92.89%, an accuracy error of 7.11%, and an average accuracy of 91.86% [5].

In a study titled "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Look Only Once) (Detection of Palembang's Signature Dishes Using YOLO (You Look Only Once) Algorithm)," it is explained that due to a lack of understanding of various culinary specialties in Palembang, the accuracy level of the YOLOv3 algorithm-based model can be considered high, as the average accuracy exceeds 80%. This is demonstrated by an average accuracy of 96% and a detection speed of 40.486 milliseconds in identifying 31 traditional dishes from Palembang [8].

In the research titled "Pemanfaatan YOLO Untuk Pengenalan Kesegaran Buah Mangga (Utilization of YOLO for Mango Freshness Recognition)," it is explained that the use of the YOLO algorithm in identifying the freshness and ripeness of mangoes still has limitations in terms of accuracy due to the need for more datasets and variations to improve its performance. In the first scenario, which contains only fresh mangoes, it resulted in an accuracy of 80%, precision of 82%, recall of 87%, and an F1-score of 84%. In the second scenario, which contains only ripe mangoes, an accuracy of 76%, precision of 76%, recall of 87%, and an F1-score of 81% were obtained. In the third scenario, which contains both fresh and ripe mangoes, it yielded an accuracy of 73%, precision of 66%, recall of 81%, and an F1-score of 73% [9].

In the research titled "Pendeteksian Sel Darah Putih Dari Citra Preparat Dengan You Look Only Once (White Blood Cell Detection from Prepared Images with You Look Only Once)," it explains the need to create a system that facilitates the detection of white blood cell prepared image. This research includes two scenarios: stained and unstained cell preparations. In the stained scenario, the results showed accuracy, precision, recall, and F1-score each at 100%. Meanwhile, in the unstained scenario, the results were accuracy at 76.5%, precision at 100%, recall at 55%, and F1-score at 71.5% [7].

II. METHOD

The methodology in this research starts with the problem identification process and is followed by a literature review related to the existing issues. The research begins by exploring topics regarding the development of machine learning algorithms. The focus is specifically on YOLO and CNN because these algorithms are frequently applied to various visual classification problems and are often combined with other algorithms for more optimal model training. Following that, the dataset collection process is carried out, which involves tagging and cropping as part of the training data preparation. Subsequently, the next step is the coding process, where the processed training data is prepared for use with both algorithm combinations that will be employed. The next step involves testing with reference to predefined parameters in order to analyze the performance of the combined algorithms.

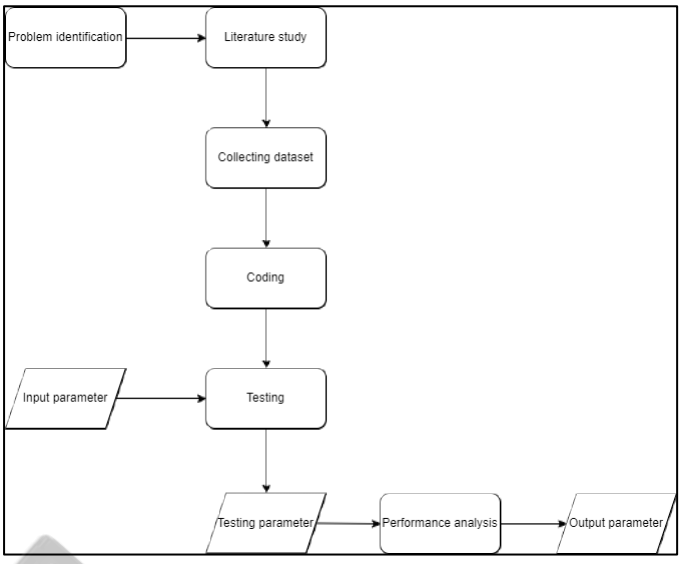

Fig. 1. Research Method Flowchart

A. Data

The data used in this study consists of a dataset of vehicle images categorized into 4 types: motorcycles, cars, buses, and trucks. Data collection is carried out through literature review and manual screen capturing. Data collection is conducted to preprocess the data before the algorithm implementation process. Data collection through literature review is conducted by searching various sources such as journals and online media. In the manual screen capturing process, data collection is performed by capturing screens from the live streaming of road CCTV owned by the Semarang City Transportation Agency on the website http://tiliksemar.semarangkota.go.id/dashboard. This is done to ensure that the data obtained closely resembles the conditions on the roads in Indonesia in general.

B. Pre-processing

Before implementing the CNN and YOLO algorithms, the acquired data will go through a tagging process (YOLO) and cropping process (CNN) so that the collected data can be sorted according to their respective classes and used for the training process in each algorithm. This dataset consists of 484 images and includes 6482 tags for YOLO algorithm training data, where these tags are divided into 4 classes: motorcycles (2050 tags), cars (3650 tags), buses (365 tags), and trucks (417 tags). For the CNN algorithm training data, each class consists of 100 images.

C. Algorithm

A computer working system, involving software, hardware, and humans as its components, is an algorithm. [10]. The absence of any one of these three elements would render a computer useless, with a sole focus on the software used. Software itself consists of a series of programs and writing rules. In the process of designing programs or establishing such writing guidelines, a structured and logical approach is required to solve problems or achieve specific

goals. This is where the importance of algorithms comes into play. An algorithm is a sequence of logical and structured steps used to solve a particular problem or achieve a goal. In the world of computers, algorithms play a central role in software development. In our daily lives, the influence of algorithms has impacted various aspects of our lives [11].

D. CNN (Convolutional Neural Network)

Convolutional Neural Network (CNN) is a type of artificial neural network architecture widely used in the domains of image recognition, computer vision, and related image tasks [3]. CNN is designed to hierarchically recognize patterns and features in image data by leveraging convolution, pooling operations, and non-linear layers. The CNN architecture is divided into two main parts, namely the Feature Extraction Layer and the Fully Connected Layer (MLP) [12], [13]. In the Feature Extraction Layer, there is the "encoding" of images into numerical representations (Feature Extraction), consisting of Convolutional Layers and Pooling Layers. The Convolutional layer, as the first and essential layer in the CNN architecture, performs convolution with filters, typically 3x3 in size, to extract prominent features from the input image. This layer also involves stride parameters to control the filter's movement across the matrix and padding techniques (zero padding) to maintain the original image size. In the pooling layer, the matrix is reduced by selecting portions or groups of features generated from the convolutional layer [14]. These features are then processed to generate values representing the selected group or section. After the pooling layer, there is a fully connected layer, which is the last layer in the CNN architecture. This layer is responsible for classifying the received input. The features generated from the previous layers need to be flattened into a vector to be used as input for the fully connected layer before the classification process is executed. Then, an activation function such as softmax or sigmoid is used to classify based on the highest category value [12].

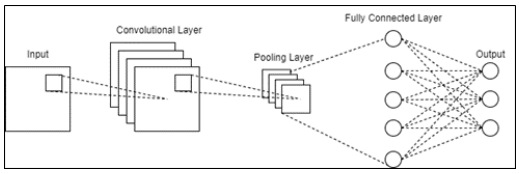

Fig. 2. CNN Architecture

E. YOLO (You Look Only Once)

You Only Look Once (YOLO) is an algorithm and artificial neural network architecture designed for real-time object detection [15]. The object detection process begins with the selection of the object area to be detected in the image (annotation). YOLO utilizes a single neural network approach in the image. This network is responsible for dividing the image into several regions (grids) and making predictions for each region (bounding box) to classify them as objects or not. The YOLO algorithm architecture leverages Convolutional Neural Network (CNN) [16] comprising 24 convolutional layers, followed by 2 fully connected layers. The convolutional layers play a role in extracting features from input images, while the fully connected layers are used to predict output probabilities and object coordinates.

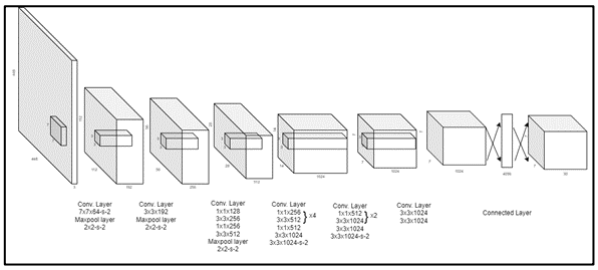

Fig. 3. YOLO Architecture

F. Confusion Matrix

One way to measure the performance of a classification model is by evaluating the values of precision and recall. In the evaluation of the classification model's performance, there are several commonly used terms, namely "positive tuple" and "negative tuple." Positive tuple refers to the tuples that are the focus of discussion, while negative tuple refers to tuples other than the ones being discussed. There are also fundamental terms in the calculation of precision and recall values, such as true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [1]. These terms are often combined in a matrix known as a confusion matrix, as shown in the following figure.

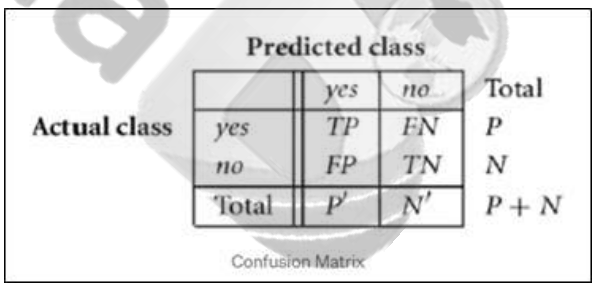

Fig. 4. Confusion Matrix

True Positive (TP) occurs when the model makes accurate predictions for the positive class in test data. True Negative (TN) happens when the model makes accurate predictions for the negative class in test data. False Positive (FP) occurs when the model makes incorrect predictions by incorrectly classifying test data as positive when it should be negative. False Negative (FN) is when the model makes incorrect predictions by incorrectly classifying test data as negative when it should be positive.

G. Final Outcome Parameter

In this research, four final outcome parameters from the testing of the CNN and YOLO algorithms were used, namely precision, recall, and F-1 score. Precision, recall, and F1 score are commonly used evaluation metrics in the field of machine learning and data mining to measure the performance of

Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors*).

classification models or detection systems. Each of these metrics provides different information about the model's ability to make accurate predictions.

- Precission

  The precision measures how well a model correctly identifies positive class instances among all the results predicted as positive class. It focuses on how few false positive results are generated by the model. The precision value can be calculated using the equation:

$$Precission = \frac{TP}{TP - FP} \; x \; 100\%$$

- Recall

  Recall measures how well a model captures all true positive cases from the entire positive dataset. Its focus is on reducing the number of false negatives. Recall can be calculated using the following equation:

$$Recall = \frac{TP}{TP + FN} \; x \; 100\%$$

- F1-score

  The F1 score is the harmonic mean of precision and recall. This metric is useful when we want to find a balance between precision and recall, especially when positive and negative classes are imbalanced. The F1 score reaches its maximum value when precision and recall have the same value. The F1 score can be calculated using the equation:

$$F1 - s = \; 2 \; x \; \frac{Precission \; x \; Recall}{Precission + Recall} \; x \; 100\%$$

III. RESULT AND DISCUSSION

In this research, each algorithm underwent separate training processes, followed by testing with a video that would first be processed by the YOLO algorithm for cropping based on the detected vehicle types. After that, the data processed by the YOLO algorithm would undergo classification by the CNN algorithm. Evaluation was carried out through testing the performance of the training and testing processes that had been arranged. The recorded results will be separated based on the predefined classes, namely motorcycles, cars, buses, and trucks.

A. Training the YOLO Algorithm

The training data for the YOLO algorithm consists of 484 images divided into 4 classes. In the "motor" class, there are 2050 tags, in the "car" class, there are 3650 tags, in the "bus" class, there are 365 tags, and in the "truck" class, there are 417 tags. This testing was conducted until epoch 50 and can be seen in Table 1. The results in terms of precision can be seen in Figure 5, recall results can be seen in Figure 6, and the F1-score can be observed in Figure 7.

TABLE I. YOLO TRAINING DATA

Epoch	Precission	Recall	F1-score
10	0,559	0,632	0,593
20	0,853	0,819	0,835
30	0,901	0,88	0,890
40	0,923	0,915	0,918
50	0,936	0,914	0,924

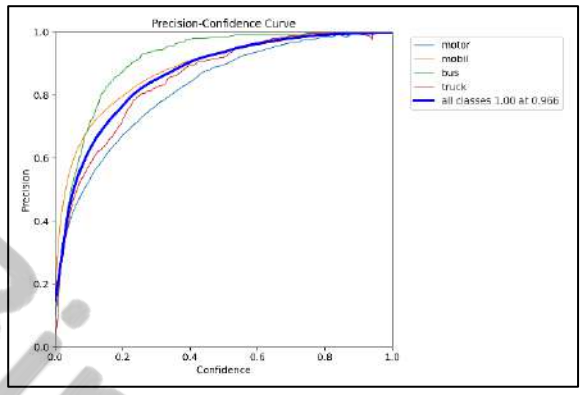

Fig. 5. Precission Graph of YOLO training result

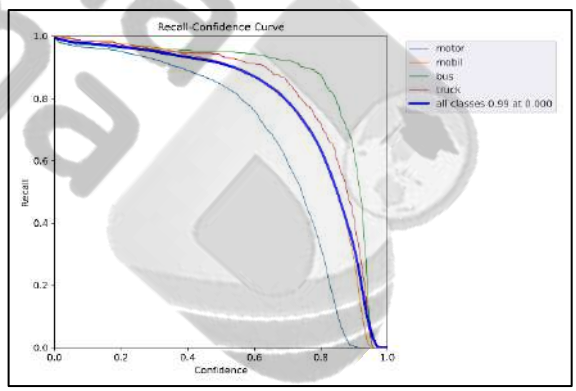

Fig. 6. Recall Graph of YOLO training result

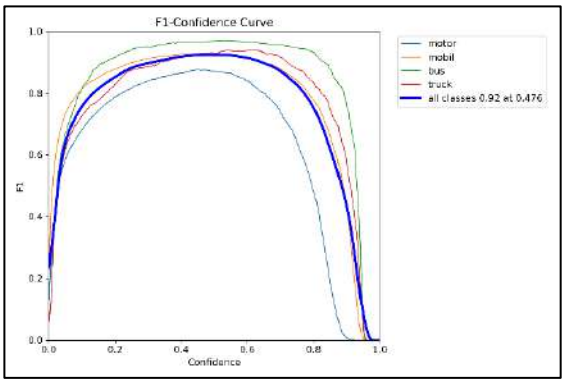

Fig. 7. F1-score Graph of YOLO training result

TABLE II YOLO SUMMARY TRAINING DATA

Class	Precission	Recall	F1-score
Motorcycle	0,897	0,848	0,871
Car	0,94	0,918	0,928
Bus	0,985	0,951	0,967
Truck	0,92	0,94	0,929

From the data generated by the YOLO algorithm, which is divided into 5 epochs starting from epoch 10, 20, 30, 40, and 50, there is a significant improvement in each epoch, as indicated by an increase in each parameter, namely precision, recall, and f1-score.

B. Training the CNN Algorithm

The training on the CNN algorithm consists of 400 images in the form of cropped images that have been divided into 4 classes. The "motor" class contains 100 cropped images, the "car" class contains 100 cropped images, the "bus" class contains 100 cropped images, and the "truck" class also contains 100 cropped images. This testing was conducted up to epoch 50, similar to the YOLO algorithm, the results of which can be seen in Table 3. The results in terms of accuracy can be observed in Figure 8, and the results in terms of loss can be observed in Figure 9.

TABLE III CNN TRAINING DATA

Epoch	Precission	Recall	F1-score
10	0,82	0,82	0,82
20	0,88	0,87	0,87
30	0,90	0,90	0,90
40	0,96	0,95	0,96
50	0,95	0,95	0,95

From the data generated by the CNN algorithm, which is divided into 5 epoch levels, namely epochs 10, 20, 30, 40, and 50, it shows only a slight improvement between levels in all parameters, namely precision, recall, and F1-score.

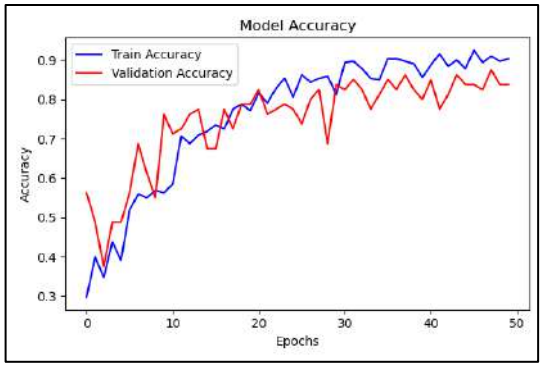

Fig. 8. The accuracy graph of CNN training results

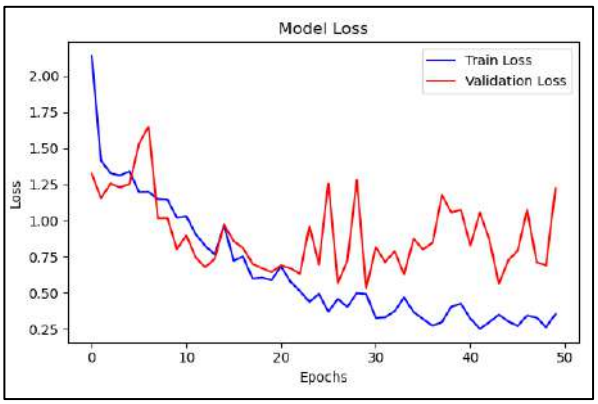

Fig. 9. Loss Graph of CNN Training Results

TABLE IV CNN SUMMARY TRAINING DATA

Class	Precission	Recall	F1-score
Motor	0,897	0,848	0,871
Mobil	0,94	0,918	0,928
Bus	0,985	0,951	0,967
Truck	0,92	0,94	0,929

C. Cropping of Test Data

The data to be used in this research will consist of a 35-second video file displaying a recording of a busy road with multiple vehicles passing through. Initially, the video will undergo a cropping process performed by the YOLO algorithm based on the training data obtained previously. The cropped data will be separated according to predefined classes established during the training process. In this study, two scenarios will be tested. In the first scenario, the cropped data generated by the YOLO algorithm will be processed entirely by the CNN algorithm. In the second scenario, the cropped data from the YOLO algorithm will undergo filtering based on its confidence score with a minimum value of 0.80.

Fig. 10. The video data that will undergo cropping process by the YOLO algorithm.

D. The Process of Classification

The data that has undergone cropping by the YOLO algorithm will be classified by the CNN algorithm using CNN training data (model training epoch 50) that has also been obtained previously. The processing results can be seen in the table below.

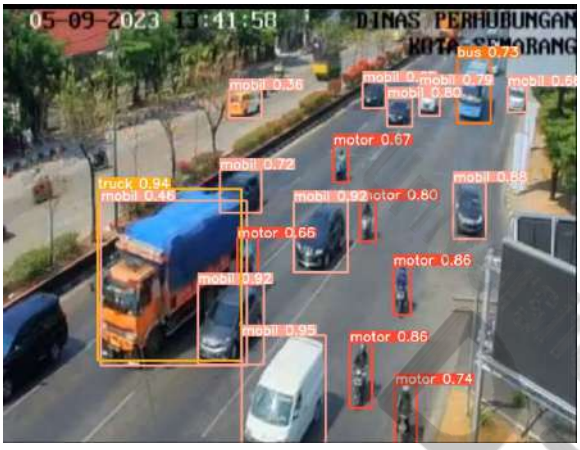

Fig. 11. The result of testing the YOLO and CNN algorithm

TABLE V SUMMARY TESTING DATA YOLO AND CNN (UNFILTERED CROPPED DATA SCENARIO)

Class	Precission	Recall	F1-score
Motor	0,81	0,57	0,67
Mobil	0,82	0,73	0,77
Bus	0,08	0,36	0,13
Truck	0,29	0,28	0,28

Starting from the first scenario (using unfiltered confidence parameter) in the table above, it can be seen from the results obtained after the testing process that the "car" class obtains the highest value among the other classes. It is followed by the "motorcycle" class, then the "truck" class, and finally the "bus" class for each parameter. This is due to several factors, such as the relatively small dataset and the presence of obstacles in the test data, for example, in the test videos, there is a billboard/videotron that obstructs the monitored CCTV's view.

TABLE VI TESTING DATA YOLO AND CNN (UNFILTERED CROPPED DATA SCENARIO)

Epoch	Precission	Recall	F1-score
10	0,33	0,41	0,31
20	0,45	0,43	0,38
30	0,51	0,42	0,36
40	0,50	0,54	0,50
50	0,50	0,48	0,46

From the table above, it can be seen that the results of testing the data using the combination of YOLO and CNN algorithms without data cropping filtering show less favorable results, even though there is an improvement from epoch 10 to epoch 20 in terms of precision, recall, and f1-score parameters. At epoch 30, there is a significant increase in the precision parameter, but there is a slight decrease in the recall parameter, which has an impact on the decrease in the f1-score value. Then, moving on to epoch 40, there is a slight decrease in the precision parameter, but an increase in the recall and f1-score parameters. Finally, at epoch 50, there is no improvement in the precision parameter, but there is a decrease in recall and f1-score values.

In the next scenario (using filtered confidence parameter) the data resulting from cropping by the pre-trained YOLO algorithm will be classified by the pre-trained CNN algorithm. This data will be classified using Google Colab with a training data of 50 epochs. However, in this scenario, the cropped data will first be filtered based on a confidence parameter to determine whether it will affect the classification results by the CNN algorithm. The minimum confidence value that will be considered is 0.80. If the confidence value is below 0.80, cropping will not be performed. This filtering process is carried out in Google Colab in the "detect.py" file by adding a minimum confidence condition of 0.80.

TABLE VI I SUMMARY TESTING DATA YOLO AND CNN (FILTERED CROPPED DATA SCENARIO)

Class	Precission	Recall	F1-score
Motor	0,84	0,62	0,72
Mobil	0,81	0,74	0,77
Bus	0,09	0,39	0,15
Truck	0,35	0,30	0,32

As seen in the table above, from the results obtained after the testing process, it can be observed that the 'car' class received the highest value among the other classes. This is followed by the 'motorcycle' class, then the 'truck' class, and lastly, the 'bus' class in each parameter, and this shows an improvement from the scenario without crop data filtering.

TABLE VIII TESTING DATA YOLO AND CNN (FILTERED CROPPED DATA SCENARIO)

Epoch	Precission	Recall	F1-score
10	0,32	0,42	0,31
20	0,48	0,46	0,42
30	0,54	0,46	0,42
40	0,55	0,58	0,54
50	0,52	0,51	0,49

From the table above, it can be seen that the results of testing the data with the combination of YOLO and CNN algorithms with the data cropping filtering scenario also show less satisfactory results, although there is a slight improvement compared to the scenario without confidence filtering. This can be observed from the comparison of each epoch in both scenarios. This occurs because the test dataset used in this scenario has undergone filtering in the confidence parameter,

so the test results can experience an improvement, albeit not significant.

IV. CONCLUSION AND RECOMMENDATION

From this research, it can be concluded that both the YOLO algorithm and CNN (Convolutional Neural Network) perform well in detecting vehicles based on predefined classes. This is evident from the tables during the training process of each algorithm. However, during the classification process using scenarios without filtering or with filtering based on confidence parameters through a combination of the YOLO and CNN algorithms, the results are not as good as during the training process of each individual algorithm. This could be due to differences in the image recognition characteristics between the YOLO and CNN algorithms. YOLO can detect objects that are partially obscured (partially covered by other objects) based on the tagging results during the training process, while CNN is trained on full-image object cropping (entirely visible). Another factor influencing the results of this research is the amount of dataset used for each algorithm and the type of image cropping used as training data for the CNN algorithm. However, it can be observed from the results of using these two scenarios that the filtering process on the cropping data processed by the CNN algorithm can affect the final parameter results.

The suggestion for further research is to conduct studies using different types of databases, meaning that the training and testing data used can be from different times, such as during the night. Additionally, the dataset and classes used should be more diverse, for example, distinguishing between different types of vehicles like sedans, pickups, and SUVs. Furthermore, within the truck class, distinctions can be made based on their shape and function, such as single-axle trucks, dump trucks, trontons, and trailers. Another recommendation is to increase the minimum confidence parameter value for data generated by the YOLO cropping algorithm.

REFERENCES

[1] K. A. Shianto, K. Gunadi, dan E. Setyati, "Deteksi Jenis Mobil Menggunakan Metode YOLO Dan Faster R-CNN," 2019.

[2] B. P. G. Pamungkas, B. Nugroho, dan F. Anggraeny, "Deteksi Dan Menghitung Manusia Menggunakan YOLO-CNN," vol. 02, no. 1, 2021.

[3] J. Sanjaya dan M. Ayub, "Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup," *JuTISI*, vol. 6, no. 2, Agu 2020, doi: 10.28932/jutisi.v6i2.2688.

[4] S. Winiarti, M. Y. A. Saputro, dan S. Sunardi, "Deep Learning dalam Mengindetifikasi Jenis Bangunan Heritage dengan Algoritma Convolutional Neural Network," *mib*, vol. 5, no. 3, hlm. 831, Jul 2021, doi: 10.30865/mib.v5i3.3058.

[5] Y. Hartiwi, E. Rasywir, Y. Pratama, dan P. A. Jusia, "Eksperimen Pengenalan Wajah dengan fitur Indoor Positioning System menggunakan Algoritma CNN," *Jurnal Sistem Informasi, Teknik Informatika, Software Engineering, dan Multimedia*, vol. 22, no. 2, hlm. 109–116, Sep 2020, doi: 10.31294/p.v22i2.8906.

[6] N. Y. S. Mendrofa, A. Mahfuzie, M. Faisal, A. Haidar, dan P. Rosyani, "Perbandingan Metode YOLO Dan FAST R-CNN Dalam Sistem Deteksi Pengenalan Kendaraan," vol. 1, no. 2, 2023.

[7] F. Andrianson, Lina, dan A. Chris, "Pendeteksian Sel Darah Putih Dari Citra Preparat Dengan You Look Only Once," vol. 9, no. 1, 2021.

[8] L. Rahma, H. Syaputra, A. H. Mirza, dan S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," *Jurnal-NIK*, vol. 2, no. 3, hlm. 213–232, Nov 2021, doi: 10.47747/jurnalnik.v2i3.534.

[9] M. S. Nuha dan R. Alexandro H., "Pemanfaatan YOLO untuk Pengenalan Kesegaran Buah Mangga," *JTI*, vol. 7, no. 1, hlm. 513, Feb 2022, doi: 10.30736/jti.v7i1.747.

[10] F. D. Wahyuningtyas, A. Arafat, A. Stiawan, dan D. Rolliawati, "Komparasi Algoritma Hierarchical, K-Means, dan DBSCAN pada Analisis Data Penjualan Melalui Facebook," *Explore. jurnal. sistem. inf. dan. telematika*, vol. 14, no. 1, hlm. 7, Jun 2023, doi: 10.36448/jsit.v14i1.2931.

[11] A. M. Retta, A. Isroqmi, dan T. D. Nopriyanti, "Pengaruh Penerapan Algoritma Terhadap Pembelajaran Pemrograman Komputer," *Indiktika J. Inov. Pend. Mat.*, vol. 2, no. 2, hlm. 126–135, Mei 2020, doi: 10.31851/indiktika.v2i2.4125.

[12] R. M. Mailoa dan L. W. Santoso, "Deteksi Rompi dan Helm Keselamatan Menggunakan Metode YOLO dan CNN," 2022.

[13] R. A. Hamzah, C. Setianingsih, dan R. A. Nugrahaeni, "Deteksi Pelanggaran Parkir Pada Bahu Jalan Tol Dengan Intelligent Transportation System Menggunakan Algoritma Faster R-Cnn," 2022.

[14] S. Sakib, N. Ahmed, A. J. Kabir, dan H. Ahmed, "An Overview of Convolutional Neural Network: Its Architecture and Applications," MATHEMATICS & COMPUTER SCIENCE, preprint, Feb 2019. doi: 10.20944/preprints201811.0546.v4.

[15] F. Rachmawati dan D. Widhyaestoeti, "Deteksi Jumlah Kendaraan di Jalur SSA Kota Bogor Menggunakan Algoritma Deep Learning YOLO," 2020.

[16] D. S. Aulia, C. Setianingsih, dan M. Kallista, "Deteksi Tanda Kehidupan Pada Korban Bencana Alam Dengan Algoritma YOLO Dan Open Pose," 2021.