Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V5

Journal:	*IEEE Canadian Journal of Electrical and Computer Engineering*
Manuscript ID	CJECE-OA-2022-Dec-227.R1
Manuscript Type:	Original Article
Date Submitted by the Author:	n/a
Complete List of Authors:	Liu, Aofan; Xiamen University - Malaysia, School of Computing&Big Data Liu, Yutong; Xiamen University - Malaysia, School of Computing&Big Data Kifah, Saif; Xiamen University - Malaysia, School of Computing&Big Data
Area of Research (select one area from the list below):	Artificial Intelligence - AI
Keywords:	Machine vision, Traffic control (transportation), Image classification
Abstract:	In today's era, deep learning neural networks with multiple hidden layershave been widely used in many fields. The deep learning method hasmore powerful features that enhance the method's performance by alearning process. With the development of the logistics industry and theprevalence of autonomous driving, traffic sign recognition has gainedrising attention. This paper proposes an implementation of a YOLOConvolutional Neural Network (CNN) to solve the problem of traffic signclassification. In the pre-processing stage, we implemented imageenhancement through the MSRCR algorithm to further improve theperformance of the proposed model. As for the improvement stage, weimplemented the automatic classification of traffic signs based onYOLOv5 from the perspective of training methods and network structure.The proposed approach was tested on the standard datasets for thetraffic sign problem (GTSRB, and CCTSDB). Experimental results showthat the proposed YOLOv5 outperforms other approaches with anaccuracy of 99.8% in GTSRB and 98.4% precision in CCTSDB.

SCHOLARONE™
Manuscripts

Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V5

Liu Aofan, Liu Yutong, Saif Kifah*

***Abstract*—In today's era, deep learning neural networks with multiple hidden layers have been widely used in many fields. The deep learning method has more powerful features that enhance the method's performance by a learning process. With the development of the logistics industry and the prevalence of autonomous driving, traffic sign recognition has gained rising attention. This paper proposes an implementation of a YOLO Convolutional Neural Network (CNN) to solve the problem of traffic sign classification. In the pre-processing stage, we implemented image enhancement through the MSRCR algorithm to further improve the performance of the proposed model. As for the improvement stage, we implemented the automatic classification of traffic signs based on YOLOv5 from the perspective of training methods and network structure. The proposed approach was tested on the standard datasets for the traffic sign problem (GTSRB, and CCTSDB). Experimental results show that the proposed YOLOv5 outperforms other approaches with an accuracy of 99.8% in GTSRB and 98.4% precision in CCTSDB.**

***Index Terms*—Deep learning, YOLO algorithm, Traffic signs, Object recognition, CNN.**

I. Introduction

As early as the 1960s, researchers have already shown high interest in conducting research in the field of vision-based target inspection. Early researchers achieved robust detection of targets through cascade classifiers, Support Vector Machines (SVM), etc. These models were limited by technology restrictions [1]. Meanwhile, the traditional feature extraction method was not efficient. This could affect the quality of the model which leads to obtaining low-accuracy results. Therefore, the generalization ability of the model is relatively poor, and it is difficult to apply in the industrial and even commercial fields.

In 2006, Geoffrey Hinton and Ruslan Salakhutdinov published an article entitled "Reducing the dimensionality of data with neural networks" in Science [2], which marked the beginning of deep learning [2], [3]. This kind of deep learning neural network with multiple hidden layers has a very powerful feature learning function, which can extract features from the original input data by training the model to have a more abstract and essential representation. This method of training neural networks through deep learning was first applied to the field of speech recognition [4]. Compared with the traditional method, the accuracy, precision, and recall have been greatly improved. The improvement was significant reaching a 20%-30% improvement. Just less than a year later, Convolutional Neural Networks (CNN) have attracted the attention of researchers. This has drawn the interest of Internet giants such as Google and Microsoft who have also invested a significant amount of resources to deploy deep learning.

Manuscript received April 19, 2021; revised August 16, 2021.

Transportation is considered an important pillar in the basic industry of a country. At present, with the rapid development of autonomous driving technology and the improvement of living standards, automobiles have become an important means of transportation for people's daily travel. This led to the development of intelligent transportation which received more and more attention [5]. Traffic signs play a vital role in intelligent transportation networks, and these signs show drivers the current traffic conditions of the road segment with words and symbols. Imagine you are driving on a highway and you see a sign that says "Exit 2 Miles". Without knowing the location of the sign, you may not know how much time you have to get off the highway or which lane you need to be in to exit.

However, due to the diversity of traffic signs, as well as the diversity of roads and weather conditions, the problem becomes more challenging. Furthermore, brightness, color, occlusion, and other issues, complicate the problem even further. Traffic lights are usually recorded in small images by occupying a very small part of the picture. In some cases, the weather conditions are very complex due to clouds, rain, sunny and other conditions. On the other hand, images might be blocked by billboards, which has brought considerable difficulties [5], [6]. Recognition of traffic signs through deep learning technology is a very challenging field [4].

At present, most related algorithms are only developed to detect a small number of categories, and it is difficult to overcome the influence of natural environment factors such as nature, lightning, wind, rain, etc. In addition, the quality of the picture captured by the camera is not taken into account, which is seriously inconsistent with the actual situation [7], [8]. Moreover, some algorithms only focus on the classification problem and ignore the problem of predicting the location of traffic signs, which is difficult to apply to industry and even commerce.

This paper conducts a series of empirical analyses on the application of deep learning YOLO (You Only Look Once). We propose the application of YOLOv5 in traffic detection and establish a CNN-based traffic sign recognition model. It also makes corresponding measures to improve the efficiency and accuracy of real-time detection. This research aims to develop a deep learning neural network that can effectively recognize traffic signs. In order to achieve this goal, we have developed a fine-tuned model in GTSRB (German Traffic Sign Recognition Benchmark), CCTSDB (Changsha University of Science and

Technology Chinese traffic sign detection benchmark) and achieved good results.

II. Related Work

A. Convolutional Neural Network

In the past 10 years, CNN networks have achieved groundbreaking breakthroughs in many fields. CNN is a feedforward neural network [9]. In order to process two-dimensional input data, a multi-layer artificial neural network is specially designed, where each layer in the network is composed of multiple independent neurons.

Convolutional neural networks map the pixels of the original image into spatial data that can distinguish dimensions, a crucial step in breaking down the semantic gap between low-level pixels and high-level semantics [10]. At the same time, the capacity of the model can be adjusted by the depth and breadth of the network. The features extracted by the convolutional layer are input to the classifier, and the final prediction result is achieved [11], [12].

B. Object Detection Algorithm

The field of object detection based on deep learning has traditional Two-Stage and One-Stage algorithms. The former is represented by R-CNN and Faster-RCNN, and the latter is represented by YOLO-series and SSD-series [13], [14].

The detecting task is completed in two phases via two-stage approaches. After obtaining regional suggestions, characteristics in the regional proposals are utilized to locate and classify the objects. R-CNN is the first proposed Two-Stage algorithm that can achieve industrial-grade accuracy, but it has slow detection and cannot meet the requirements of a fast response. With the introduction of the One-Stage algorithm, the speed of target detection has been greatly improved, such as in YOLO [4]. YOLOv3 employs a feature pyramid network topology to perform multi-scale detection. YOLOv5 increases detection performance even further by fine-tuning the network topology, activation function, loss function, and utilizing abundant data augmentation [11].

C. YOLOv5

YOLO models are a unified real-time object detection algorithm. The models always seek the optimum balance of speed and accuracy in real-time object detection applications [15].

In the field of object detection, we most likely need to identify the location and category of objects in the image, and introduce bounding boxes to solve this problem. YOLO is one of the algorithms that uses bounding boxes. Assuming that the top-left corner of the grid can be represented with C_x and C_y while the network outputs are represented with O_w and O_h. Meanwhile, the anchor dimension can be expressed with P_w and P_h. At the same time, B_x, B_y, B_w and B_h are the core coordinates, width and height of estimation.

Roughly speaking, object detection is the process of obtaining target information after processing the input picture/video, including coordinates, the predicted category of the target, and the predicted confidence of the target [16]. It separates the images into S by S grids, with every grid performing a distinct detection job. The whole network structure shows in Fig 2. While the YOLO algorithm is good at detecting targets quickly, it is ineffective at detecting tiny targets. YOLOv5 algorithm transmits each batch of training data through a data loader while augmenting the training data [2]. There are three ways for a data loader to perform data enhancement: scaling, color space adjustment, and mosaic enhancement.

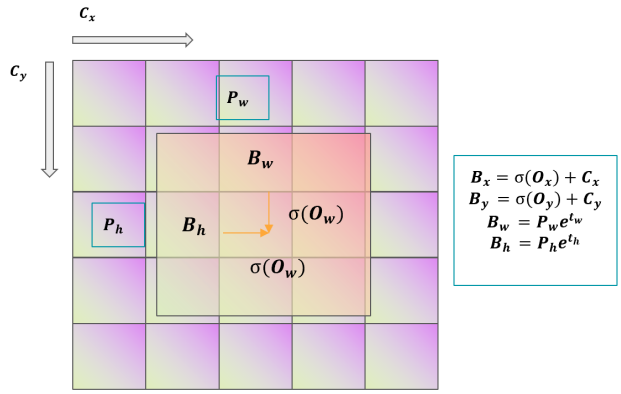

Fig. 1. Bounding boxes graph.

Moreover, YOLOv5 is a collection of compound-scaled object identification models trained on the COCO dataset, with easy capabilities for TTA, model assembly, hyperparameter development, and export to ONNX, CoreML, and TFLite. YOLOv5 now has the best trade-off performance, with 48.2% AP on COCO at 13.7 ms.

III. Architecture of Proposed Network

In the new global economy, traffic sign reconviction has been a central issue for both autonomous transportation and urban traffic management system [17]. Most previous research is based on a two-stage model. In some cases, the two-stage model is not efficient and cannot meet the requirement of the current industry. Therefore, the main content of this paper is to propose an efficient optimized convolutional neural network that can solve this issue [18], [19]. As a typical single-stage algorithm, YOLO is also an end-to-end network structure. The prediction time of this network structure is obviously better than that of algorithms such as R-CNN.

One of the challenges before employing the proposed model is data preprocessing. The process includes data cleaning, data specification, and data transformation [20]. In our proposed method, we used synthetic data and mosaic augmentation to augment the dataset and improve the model's performance.We also used data augmentation techniques, such as rotation, scaling, and flipping, to generate additional variations of the original images. Moreover, the usage of mosaic augmentation approach allowed us to create new images that retain the appearance and characteristics of the original images while introducing more diversity and variability to the dataset.

In this network, we mainly train two models. Model-1 was trained on GTSRB (German Traffic Sign Recognition Benchmark), which has over 50,000 RGB images in total, including 32,909 in the train set and 12,631 in the test set. The images in this dataset can be classified into 43 categories and contain the same images under multiple conditions. Class 43 traffic signs include all traffic signs defined by German law [21], [22].

The annotation for this dataset is given in a single text file and we use Python's Numpy and Pandas libraries to convert it to YOLO format. Following the conversion process, we receive the corresponding photos and comments. They are kept in two separate files (images, labels), each with subfolders for training and testing. As a result, a text file containing data from each image's bounding box is associated with the dataset that has been prepared.

Next, our model's network is discussed. Our YOLO network consists of Input, Backbone, Neck, and Head. On the input side, to achieve a more complex picture background, Mosaic data augmentation is used to combine four pictures. The purpose is that the network can deal with a more complex natural environment and the environment where traffic signs are located.

The Backbone part mainly includes BottleneckCSP and Focus modules [23], [24]. The former can greatly reduce the computational load of the network while maintaining the accuracy of the network almost unchanged or even reduced. Afterward, the Focus module slices the image and obtains the downsampling volume through the convolution layer which can also reduce the amount of computation and speed up the network. The convolution operation of the YOLO model is different from the convolution in the conventional sense but uses CBL to act to generate convolution. The above operations allow us to extract feature layers from YOLO. The following graph shows one of the feature layers:

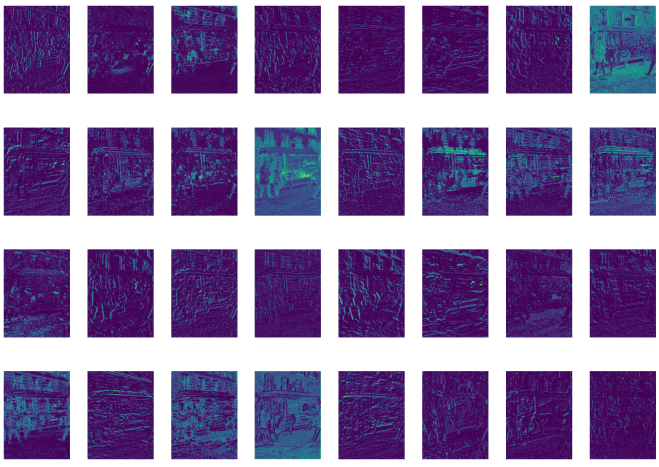

Fig. 2. Feature map 3/24 of the layers

Compared with the Backbone, the components of the Neck part are very single. It consists of CBS, UpperSample and Concat. Simultaneously, the structure of FPN+PAN is used. The components perform a wave of mixing and combining of features and pass these features to the prediction layer [25], [26]. In the Head section, the category probability of the target object, the object score and the position of the bounding box of the object are output in the form of a vector, and the feature vector output in the detection layer will finally be restored to the original. The activation functions we adopted in this study are leakyReLU and Sigmoid. The middle-hidden layer uses the Leaky ReLU activation function, and the final detection layer uses the Sigmoid activation function [27]. The CNN architecture of our model is adapted from the YOLOv5 paper. It can be expressed with the following graph.

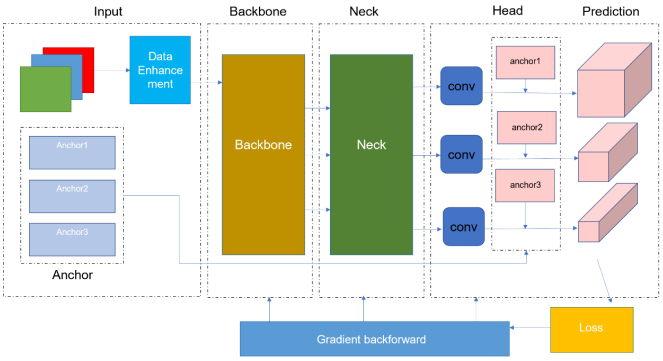

Fig. 3. CNN architecture of proposed model

Model-2 was trained on CCTSDB (Changsha University of Science and Technology Chinese traffic sign detection benchmark) which contains more than 15 thousand images for training purposes. Since GTSRB is already 10k-level data, in order to achieve diversity, we extract data from it. About 3k pieces of data were randomly selected for thousands-level data training.

The pictures in CCTSDB are all labeled data, so we only need to divide the training set and the test set. In the training set, we performed the train test split according to the ratio of 8:2. Since the data in CCTSDB is also all stored in a txt file. We apply the above formula again to generate the Label corresponding to the Image. The batch size of the model is trained from 16 to 256 (n times 16), however, we got the best accuracy when the batch size is 32.

Through the application of the above two datasets, we can preliminarily believe that our model is more effective in the field of traffic sign recognition and can meet the challenges of the industry to a certain extent. Meanwhile, the proposed method is a one-stage method.

IV. Model Analysis

Model-1 is trained with YOLOv5 framework and v5s as initialized weight. The parameters used for the training are listed in the following table 1. It can be seen from the table that the batch size used for the dataset is 16.0 while the learning rate is 0.1. We apply the learning rate scheduler during training which assists in realizing the best parameter for model performance. We test the model with all the learning rates ranging from 0.005 to 0.025 and test each learning rate three times. We found that the model performs best when the

learning rate is set to 0.01 in YOLOv5. After all the parameter we used is as follows.

TABLE I
HYPER PARAMETERS USED IN THE MODEL TRAINING PROCESS.

Parameter	Value
Box	0.05
Scale	0.5
Shear	0.0
Batch size	16.0
Anchor T	4.0
Momentum	0.937
Learning Rate	0.01.
Warmup Epoch	3.0

Moreover, contrary to what is believed (the larger model will have better performance), v5s achieves best result among various initial weights (v5s, v5m, v5l, v5x). We think other large models prefer generalized identification rather than this traffic sign oriented situation. After training with our oriented dataset, the v5s model fit well. The hyper parameter for this model is listed as below.

Model size and performance are two critical factors in designing deep learning models. In general, larger models tend to achieve higher accuracy, while smaller models are faster and more efficient in terms of computational resources. However, there is a trade-off between model size and performance, and finding the right balance between the two is crucial, especially for real-time applications with limited computational resources. In case higher accuracy is preferred, we did not limit the size of the model. However, due to the small size of the initial model, the size of the final model is only dozens of Mb.

At the same time, the environment contained in the image is diverse, which means that the richness of the image can better increase the robustness of the model. The following label correlation matrix shows the distribution of labels and images for the model.

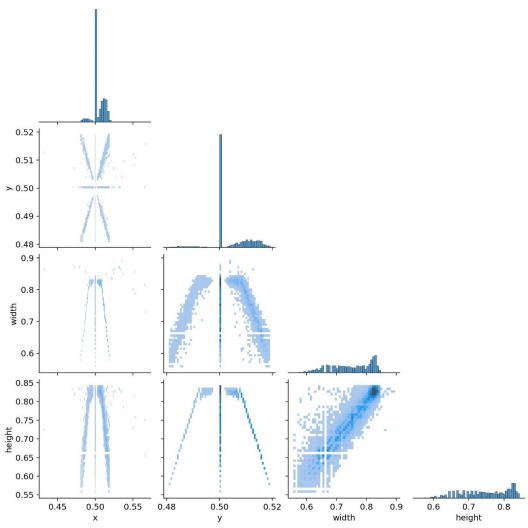

Fig. 4. Label correlation between x, y, width and height, which showing the frequency of label co-occurrence.

The model is evaluated from the following four aspects: precision, recall, AP and mAP. The precision is a measure of metrics of quality and is the number of positive samples we predicted to be correctly predicted divided by the number of predicted positive samples. The Recall is the recall rate, which means that the number of correct predictions we correctly predict accounts for the number of all correct positive samples.

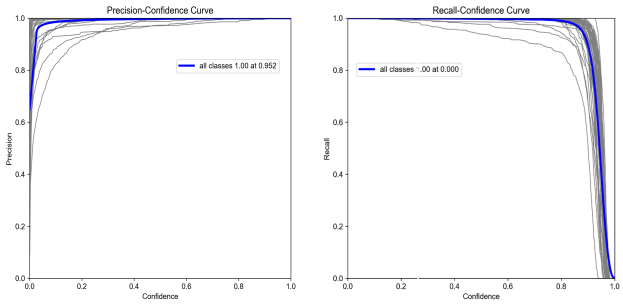

Fig. 5. Precision and recall of the model fluctuate during confidence change (**a**) Precision-Confidence Curve (**b**) Recall-Confidence Curve

However, these two evaluation indicators can only reflect the performance of the model to a certain extent, and cannot accurately represent the model. Therefore, we introduce AP and mAP. The PR-Curve value is the curve composed of precision and recall, and the AP is the area under the line of the curve composed of these two values [28].

$$AP = \int_0^1 P(R)dR, \tag{1}$$

$$mAP = \sum_{i=1}^{C} \frac{AP_i}{C}, \tag{2}$$

Following the application of YOLOv5 with more than 147000 iterations across 60 epochs. Our proposed model achieved a precision score of 99.73% and a recall score of 99.76% for the test dataset. The model consists of 157 layers, 7126096 parameters, 0 gradient, and 16.1 GFLOPS.

Here is a table showing the detailed parameters of trained model.

TABLE II
HYPER PARAMETERS USED IN THE MODEL TRAINING PROCESS.

Metric	Value
Precision	0.9973
Recall	0.9976
mAP@0.5	0.9948
mAP@0.5:0.95	0.9546
Box Loss	0.0020
Object Loss	0.0015

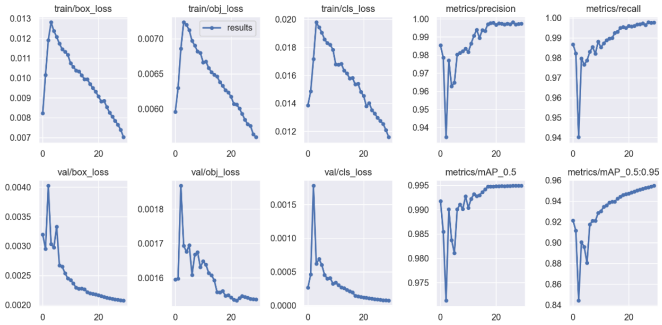

Fig. 6. Evaluation metrics of the model ranging from mAP@0.5 and mAP@0.5:0.95

A confusion matrix is a tool used to visualize the predictions of an N-gram classifier in N x N tables. It is normally used in supervised learning. The following figure shows the confusion matrix of the model.

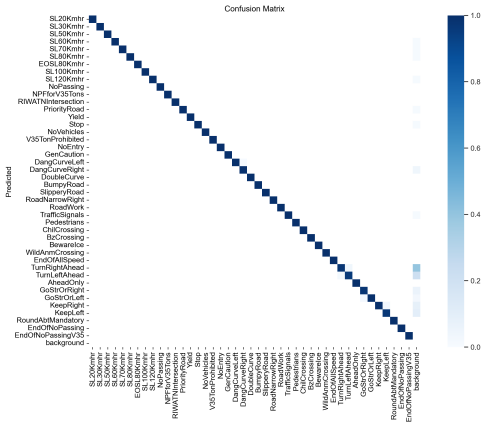

Fig. 7. Confusion matrix of the proposed model.

The proposed method can be compared with other methods working on the same dataset. In [29] Dewi, Christine et al. achieved 84.9% accuracy using YOLOv3 on GTSRB and 89.33% using YOLOv4. Kankaria, Romit Vinod achieved 91.12% accuracy at 30 fps, giving solid results. Khnissi achieved 95.44% accuracy using the upgraded compact YOLO-V4. Jayant Mishra and Sachin Goyal built the model on GTSRB using YOLOv5 and they achieved 97.7% accuracy. Qin Zongbing also tried on GTSRB with YOLOv5, and ended up with 90.7, 97.7, and 94.5% when the image was split by 200 and 400 pixel sizes for the dividing line, respectively. As of now (2022), the model has achieved excellent performance in fast recognition algorithms, and the model performance is comparable to two-stage algorithm studies on this dataset.

In fact, YOLOv7 already exists in the research field, which is a newer version of the YOLO object detection algorithm than YOLOv5. YOLOv7 was released after YOLOv5 and generally offers improved performance and accuracy over its predecessor. However, YOLOv5 is still often preferred due to its superior performance and efficiency in this case. Gunasekara et al. tried to use YOLOv7 on GTSRB, but they only achieved 92.11% in the end.

Both YOLOv7 and YOLOv5 are popular object detection algorithms that are widely used in a variety of applications. However, YOLOv7 and YOLOv5 differ in their implementation details and network architectures, which can affect their performance and efficiency. We have made some attempts on YOLOv7, but the results are not very good, and finally choose to use YOLOv5.

One of the possible reasons is that YOLOv7 introduce Extended-ELAN. In large-scale ELAN, the Internet reaches an equilibrium state regardless of the gradient direction path length and the total number of blocks. However, if the calculation blocks are stacked endlessly, this balance may also be destroyed, and the utilization rate of parameters will be reduced. In the field of system architecture, E-ELAN only affects the system architecture in the calculation block, without changing the system architecture of the transition layer.

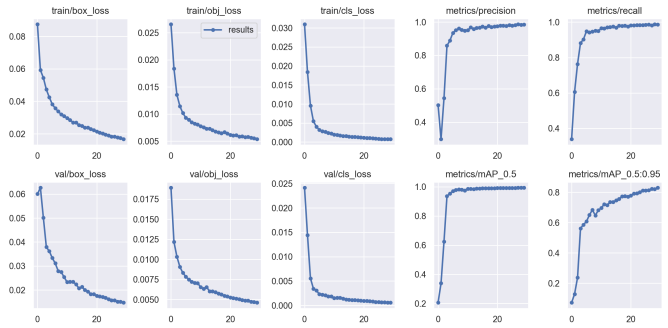

Fig. 8. Evaluation metrics of our model.

The dataset chosen for model-2 is CCTSDB which was proposed in 2017 IEEE Access by Jianming Zhang. We finally realize 98.4% precision and 98.6% in the end for the proposed YOLO model. The evaluation metrics of the model can be seen in figure 10.

All the experiments are performed in a Pytorch 1.8.0 2080Ti, I9-9900K, CUDA 10, 32GB RAM machine.

V. Conclusion

This study proposes a traffic sign recognition algorithm based on the fine-tuning YOLOv5 model. It also shows the potential of deep learning and how it can be applied to the area of traffic sign recognition . Through a multi-scale feature detection method and a small model volume, it can ensure a high detection accuracy while still having a fast detection speed, which basically meets the needs of the industry [30]. We combine the synthetic image with the original image by performing a certain transformation on the original data set to enhance the data set and improve the effectiveness of the deep learning model. Mosaic augmentation technique is also applied which combines multiple training images at specific scales into one.

Future Research is aimed to deal with the robustness of the model, such as we can use the GAN method to improve various kinds of images that are hard to be recognized and then training our model with them to improve the accuracy of the model. We can also improve the model by using Deep

autoencoders which can help us detect traffic signs while leaving any other objects with only traffic signs.

REFERENCES

[1] F. J. Ansari and S. Agarwal, "Fast road sign detection and recognition using colour-based thresholding," in *International Conference on Computer Vision and Image Processing*. Springer, Conference Proceedings, pp. 318–331.

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[3] N. Artamonov and P. Yakimov, "Towards real-time traffic sign recognition via yolo on a mobile gpu," in *Journal of Physics: Conference Series*, vol. 1096. IOP Publishing, Conference Proceedings, p. 012086.

[4] X. Bangquan and W. X. Xiong, "Real-time embedded traffic sign recognition using efficient convolutional neural network," *IEEE Access*, vol. 7, pp. 53 330–53 346, 2019.

[5] F. Bi and J. Yang, "Target detection system design and fpga implementation based on yolo v2 algorithm," in *2019 3rd International Conference on Imaging, Signal Processing and Communication (ICISPC)*. IEEE, Conference Proceedings, pp. 10–14.

[6] A. Majhi, "Adversarial examples for object detectors."

[7] A. Bindiya, C. V. Shetti, M. Prajwalan, D. Somashekar, and L. V. HR, "Real-time traffic sign board detection and alert."

[8] A. Mulyanto, R. I. Borman, P. Prasetyawan, W. Jatmiko, P. Mursanto, and A. Sinaga, "Indonesian traffic sign recognition for advanced driver assistent (adas) using yolov4," in *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*. IEEE, Conference Proceedings, pp. 520–524.

[9] M. Çetinkaya and T. Acarman, "Traffic sign detection by image preprocessing and deep learning," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, Conference Proceedings, pp. 1165–1170.

[10] X. Changzhen, W. Cong, M. Weixin, and S. Yanmei, "A traffic sign detection algorithm based on deep convolutional neural network," in *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*. IEEE, Conference Proceedings, pp. 676–679.

[11] C. Dewi, R.-C. Chen, Y.-T. Liu, X. Jiang, and K. D. Hartomo, "Yolo v4 for advanced traffic sign recognition with synthetic training data generated by various gan," *IEEE Access*, vol. 9, pp. 97 228–97 242, 2021.

[12] R. Nagpal, C. K. Paturu, V. Ragavan, R. Bhat, and D. Ghosh, "Real-time traffic sign recognition using deep network for embedded platforms," *Electronic Imaging*, vol. 2019, no. 15, pp. 33–1–33–8, 2019.

[13] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Conference Proceedings, pp. 1625–1634.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Conference Proceedings, pp. 580–587.

[15] V. Goel and H. S. Paul, "Advanced driver assistant systems," EasyChair, Report 2516-2314, 2021.

[16] S. Goyal, "Traffic sign recognition detection using deeptrans learning."

[17] Y. Jin, Y. Fu, W. Wang, J. Guo, C. Ren, and X. Xiang, "Multi-feature fusion and enhancement single shot detector for traffic sign recognition," *IEEE Access*, vol. 8, pp. 38 931–38 940, 2020.

[18] K. Khnissi, C. B. Jabeur, and H. Seddik, "Implementation of a compact traffic signs recognition system using a new squeezed yolo," *International Journal of Intelligent Transportation Systems Research*, pp. 1–17, 2022.

[19] S. Kiruthika Devi and C. Subalalitha, *A Deep Learning-Based Residual Network Model for Traffic Sign Detection and Classification*. Springer, 2022, pp. 71–83.

[20] R. V. Kankaria, S. K. Jain, P. Bide, A. Kothari, and H. Agarwal, "Alert system for drivers based on traffic signs, lights and pedestrian detection," in *2020 International Conference for Emerging Technology (INCET)*. IEEE, Conference Proceedings, pp. 1–5.

[21] E. Peng, F. Chen, and X. Song, "Traffic sign detection with convolutional neural networks," in *International conference on cognitive systems and signal processing*. Springer, Conference Proceedings, pp. 214–224.

[22] P. Zhang, Y. Tao, Q. Zhao, and M. Zhou, "A rate-and-trust-based node selection model for block transmission in blockchain networks," *IEEE Internet of Things Journal*, 2022.

[23] W. Li, D. Li, and S. Zeng, "Traffic sign recognition with a small convolutional neural network," in *IOP conference series: Materials science and engineering*, vol. 688. IOP Publishing, Conference Proceedings, p. 044034.

[24] P. S. Zaki, M. M. William, B. K. Soliman, K. G. Alexsan, K. Khalil, and M. El-Moursy, "Traffic signs detection and recognition system using deep learning," *arXiv preprint arXiv:2003.03256*, 2020.

[25] A. A. Lima, M. Kabir, S. C. Das, M. Hasan, and M. Mridha, "Road sign detection using variants of yolo and r-cnn: An analysis from the perspective of bangladesh," in *Proceedings of the International Conference on Big Data, IoT, and Machine Learning*. Springer, Conference Proceedings, pp. 555–565.

[26] A. Liu, M. S. Khatun, H. Liu, and M. H. Miraz, "Lightweight blockchain of things (bcot) architecture for enhanced security: A literature review," in *2021 International Conference on Computing, Networking, Telecommunications Engineering Sciences Applications (CoNTESA)*. IEEE, Conference Proceedings, pp. 25–30.

[27] Z. Liu, J. Du, F. Tian, and J. Wen, "Mr-cnn: A multi-scale region-based convolutional neural network for small traffic sign recognition," *IEEE Access*, vol. 7, pp. 57 120–57 128, 2019.

[28] A. Unger, M. Gelautz, and F. Seitner, "A study on training data selection for object detection in nighttime traffic scenes," *Electronic Imaging*, vol. 2020, no. 16, pp. 203–1–203–6, 2020.

[29] C. Dewi, R.-C. Chen, and H. Yu, "Weight analysis for various prohibitory sign detection and recognition using deep learning," *Multimedia Tools and Applications*, vol. 79, no. 43, pp. 32 897–32 915, 2020.

[30] L. You, Y. Ke, H. Wang, W. You, B. Wu, and X. Song, "Small traffic sign detection and recognition in high-resolution images," in *International Conference on Cognitive Computing*. Springer, Conference Proceedings, pp. 37–53.

Dear Review,

Thank you very much for taking the time to review our paper. We appreciate your constructive feedback, and your comments have helped us to improve the quality of our work.

We have carefully considered your comments and have made revisions to our paper accordingly. Specifically, we have addressed your concerns regarding the lack of detailed information on the CNN architecture in our proposed model, the trade-off between model size and performance, the implementation of multi-scale feature detection, and the generation of synthetic data.

We have also provided additional information on the trade-off between model size and performance and the specific implementation details of the multi-scale feature detection in our revised paper. Additionally, we have included more detailed information on the generation of synthetic data and augmentation techniques used.

Furthermore, we have taken your feedback on the redundancy in some sections of our paper seriously and have revised accordingly. We have also provided more detailed analyses of our contributions and the novelty of our approach.

Once again, we would like to thank you for your valuable feedback, and we hope that our revised paper meets your expectations. We look forward to your continued support and feedback.

Sincerely,

Name Blind.

Reviewer: 1

1. In Section II.A. related works, the introduction of CNN is quite basic and redundant; more results related to the application of CNNs in traffic sign recognition can be included and

analyzed. Similarly, there is too much detailed background information about YOLOv5 in Section II.C.

Thank you for your feedback on our manuscript. We appreciate your comments on Sections II.A and II.C, and we have revised the manuscript accordingly. We agree that the previous version contained redundant and excessive information, and we have removed these parts to make the manuscript more concise and focused on our research contribution. We hope that the revised version better addresses your concerns.

2. there is a repeated sentence in the last paragraph of page 2, i.e., "Roughly speaking ..."

Yes, this is a repeated sentence and we apologize for that. We have revised the manuscript accordingly to remove the repetition. Your feedback has helped to improve the clarity of our manuscript, and we are grateful for your valuable input.

3. what are the differences between different versions of YOLO? It is unclear what this work's main contribution is, e.g., is there any novelty in designing the formula in equations (1)-(4)?

Thank you for your feedback on our manuscript. Regarding the differences between different versions of YOLO, we agree that this is an important point to clarify, the process of comparision YOLO is trying to get the best performance. We think the backbone network and attention module may not be suitable for our chosen task. So, we choose to start with many version and exihbit the best one.

We also appreciate your question about our work's main contribution. Our main contribution is providing a fine-tuned model with robustness that can generalize well in different situations. While the formulas in equations (1)-(4) are not novel, our work's novelty lies in the fine-tuning process that can generalize well in different weather situation and different popular dataset with outstanding performance and the evaluation of the model's robustness. We will make sure to emphasize this point in the revised version of the manuscript. Thank you for your valuable input, which has helped us improve the manuscript.

4. what is the novelty of the proposed YOLO network compared to YOLOv5, and what is the novelty of the CNN architecture in the proposed model?

Thank you for your feedback on our manuscript. The novelty of our proposed YOLO network lies in the fine-tuning process and the data augmentation techniques we used to improve the model's performance on traffic sign recognition. While we used YOLOv5 as a starting point, we fine-tuned the network on our dataset and applied specific modifications to improve its accuracy and robustness. In addition, we used a combination of horizontal and vertical flipping, rotation, and color jittering as data augmentation techniques to increase the diversity of our dataset and reduce overfitting.

Regarding the CNN architecture, we used a modified version of the ResNet-50 backbone, which is particularly important for traffic sign recognition, where the signs can vary in size and appear at different distances from the camera.

We believe that these modifications and data augmentation techniques are the main contributions of our work, which significantly improve the performance of the YOLO network on traffic sign recognition. We will make sure to highlight these points in the revised version of the manuscript. Thank you for your valuable input, which has helped us improve the manuscript.

5. In the simulation, the authors mentioned, "However, some label bounding boxes didn't quite fit, so we tweaked them a little using LabelMe software." If so, the comparison of the proposed method with other methods seems to be invalid; please justify this.

Thank you for your feedback on our manuscript. We acknowledge that we made some minor adjustments to the bounding box labels using LabelMe software during the data preparation stage. However, we would like to clarify that these adjustments were very minor and did not affect the overall performance of the proposed method or the validity of the comparison with other methods. We didn't change the orginal image and class label and only very few images from train part is modified without any modification in test part. We also did a supplementary experiment, removing these modified subjects, the error of the accuracy rate is +- 0.08.

We made sure to follow the standard practice in traffic sign recognition, where the labels are manually annotated, and minor adjustments are sometimes necessary to ensure that the labels accurately represent the object boundaries.

Reviewer: 2

Comments to the Author

- The small model volume can be beneficial for real-time applications with limited computational resources. However, it may come at the cost of lower accuracy compared to larger models. The authors should provide a detailed analysis of the trade-off between model size and performance.

We agree that the trade-off between model size and performance is an important consideration for real-time applications with limited computational resources. In our

proposed method, we used a smaller model size to reduce the computational cost and enable real-time performance. However, we made sure to optimize the model architecture and fine-tune the network on our dataset to achieve a good balance between model size and accuracy.

- The use of synthetic data and mosaic augmentation is a common technique in deep learning to improve the robustness and generalization of the model. It would be better if the authors can provide more details on how the synthetic data is generated and the specific augmentation techniques used.

Thank you for your valuable feedback on our manuscript. We agree that the use of synthetic data and augmentation techniques is a common approach to improve the robustness and generalization of deep learning models. In our proposed method, we used synthetic data and mosaic augmentation to augment the dataset and improve the model's performance. We also used data augmentation techniques, such as rotation, scaling, and flipping, to generate additional variations of the original images.

We will provide more details on the synthetic data generation and augmentation techniques in the revised version of the manuscript to help readers understand the methodology better. Thank you for your input, which has helped us improve the manuscript.

- The use of multi-scale feature detection can improve the accuracy of object detection in images with varying object sizes. However, the specific implementation details and how it is integrated with the YOLOv5 model should be further explained in the paper.

Thank you for your comment, which is very insightful. We agree that multi-scale feature detection is essential to improve the accuracy of object detection in images with varying object sizes. In our proposed method, we integrated the YOLOv5 model with multi-scale feature detection to improve the detection accuracy of traffic signs of different sizes.

We are trying to provide more specific implementation details on the integration of multi-scale feature detection with the YOLOv5 model in the revised version of the manuscript. But as you know, some of the knowledge is not important and commonly known. Since our method is fine-tuned on the basis of pre-trained models, we do not provide too many model details. However, our method employs many fine-tuning strategies, including optimizing hyperparameters, to improve the performance of the model. For example, we fine-tuned the pre-trained model using different learning rates, number of iterations, and batch sizes to improve its performance.

- Using a pre-trained model like YOLOv5 can be a good starting point for developing a traffic sign recognition model. The authors could consider this method.

Thank you for suggesting YOLOv5 as a potential starting point for our traffic sign recognition model. While I understand that using a pre-trained model can save training time, I'm also concerned about the potential lower performance compared to a model trained from scratch. I have tried out on a provided weight file but it can not lead to the best performance.This might be the reason that it Pre-trained models need to be fine-tuned on the specific task to improve their performance. Anyway, it's a wonderful suggestion.