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
Manuscript Type:	Original Article
Date Submitted by the Author:	19-Dec-2022
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.

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

THIS 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 ground-breaking 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]. As a multi-layer neural network specially designed to process input data, each layer contains several neurons, each two layers of neurons are linked to each other, and the neural networks between the same layers are connected to each other No link. 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].

A convolutional neural network comprises an input layer, hidden layers, and an output layer. Fig. 1 shows an overview of the architecture. Hidden layers consist of alternating convolutional and pooling layers [13]. Convolutional layers detect local features, while higher levels execute synthesis operations on the locals to gain global information. As for pooling layers, it extracts the primary features and so automatically extracts features. The input layer typically represents the picture's pixel matrix in neural network image processing. Then transition to the processing of the hidden layer. In the first convolution layer, the input image is convolved with n convolution kernels and addable bias vectors to produce n feature maps, and then the local regions of each feature mapping map are summed by weighted averaging, and n new feature mapping maps are obtained by a nonlinear activation function after increasing the bias. In the second convolution layer, these feature mapping maps are convolved with n convolution kernels, and n feature mapping maps are created via the second pooling layer [14]. The second pooling layer's final n outputs are vectorized individually and then fed into a traditional neural network for training.

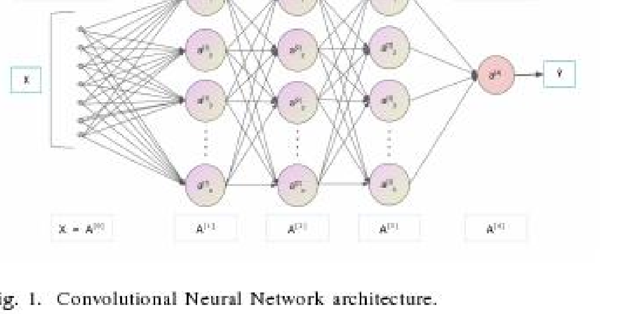

Fig. 1. Convolutional Neural Network architecture.

The application of convolutional neural networks (CNNs) in traffic sign recognition has been emerging in recent years as a promising approach for improving the safety and efficiency of transportation systems. By training a CNN on a large dataset of traffic sign images, it is possible to develop a model that can accurately classify different types of traffic signs based on their visual appearance. This has led to the development of a range of applications in the field of traffic sign recognition, including autonomous vehicles, traffic sign detection systems, and traffic sign tracking systems. Overall, the use of CNNs in traffic sign recognition is expected to continue to grow in the coming years as this technology becomes increasingly prevalent in the transportation industry.

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 [15], [16].

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 [17].

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. 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 [18]. 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

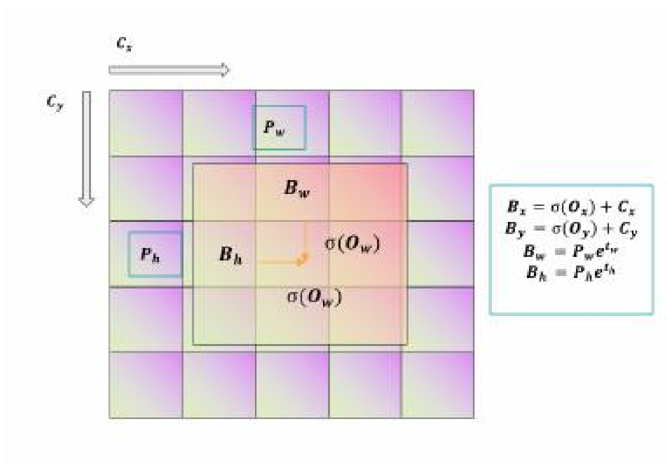

Fig. 2. Bounding boxes graph.

algorithm is good at detecting targets quickly, it is ineffective at detecting tiny targets. The one-stage method does not have a thorough grid division, therefore many targets may appear in the same grid. The YOLOv5 algorithm is able to remedy this deficiency. 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.

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.

This is what our trained model based on YOLOv5 does when predicting some images from GTSRB for image recognition. It can tell us the classes of traffic signs and labels them with bounding boxes.

Fig. 3. Bounding box prediction by YOLO-based model.

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 [19]. 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 [20], [21]. 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 [22]. Since there are some invalid data in the adopted datasets (the pictures do not contain any training objects), we think this affects the YOLO training. Therefore, we apply the method of data cleaning. The data specification is mainly for GTSRB, the former contains invalid attributes, and the latter contains invalid label files in addition to invalid attributes.

First, we carried out data transformation, which is mainly due to the inconsistency between the attributes in GTSRB and the attributes required by YOLO. In GTSRB, the position of the target is represented by pixels, while in YOLO, the position of the target in the image is required to be marked in percentage.

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 [23], [24].

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. The trained image in YOLO set comes in this format: class ID, center of bounding box in x direction, center of bounding box in Y direction, bounding box width, bounding box height. We calculate using the following formula:

$$x_{center-norm} = \frac{x_{max} + x_{min}}{2 * w}, \tag{1}$$

$$y_{center-norm} = \frac{y_{max} + y_{min}}{2 * h}, \tag{2}$$

$$w_{norm} = \frac{x_{max} - x_{min}}{w}, \tag{3}$$

$$h_{norm} = \frac{y_{max} - y_{min}}{h}, \tag{4}$$

Where, w and h represent the width and height of the bounding box in the real RGB image from the GTSRB dataset respectively. $x_{center-norm}$ and $y_{center-norm}$ is the center in normalized horizontal direction and vertical direction of the image. x_{max}, x_{min}, y_{max} and y_{min} represents the max and min object bounding box in x and y direction [25], [26].

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 [27], [28]. 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:

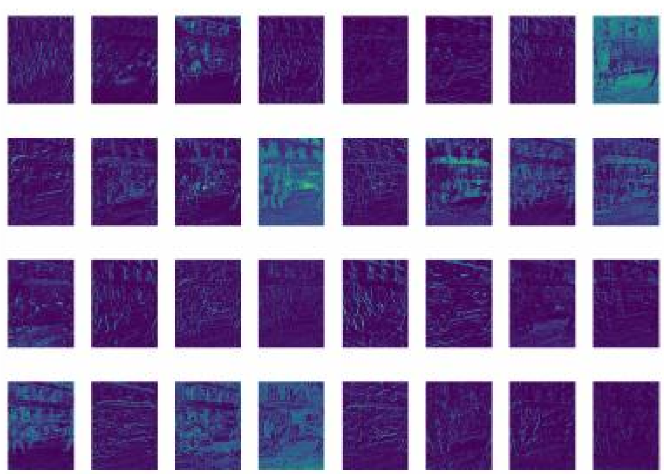

Fig. 4. Feature map 3/24 of the layers

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 [29], [30]. 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 [31]. The CNN architecture of our model is adapted from the YOLOv5 paper. It can be expressed with the following graph.

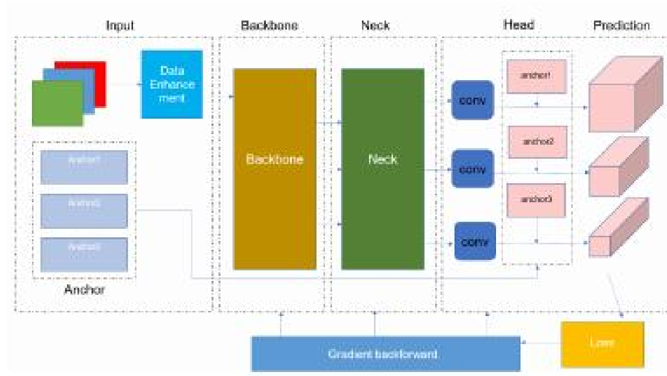

Fig. 5. CNN architecture of proposed model

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.

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. As a well-known open-source dataset, it is undeniable that the labels involved in this dataset are mostly correct [32]–[34]. However, some label bounding boxes didn't quite fit, so we tweaked them a little using LabelMe software. We believe this helps us improve the model in some way. The following label correlation matrix shows the distribution of labels and images for the model. It is generally thought that tags with stronger correlations co-occur more frequently. The label matrix is nearly symmetric, which indicates the good nature of our dataset.

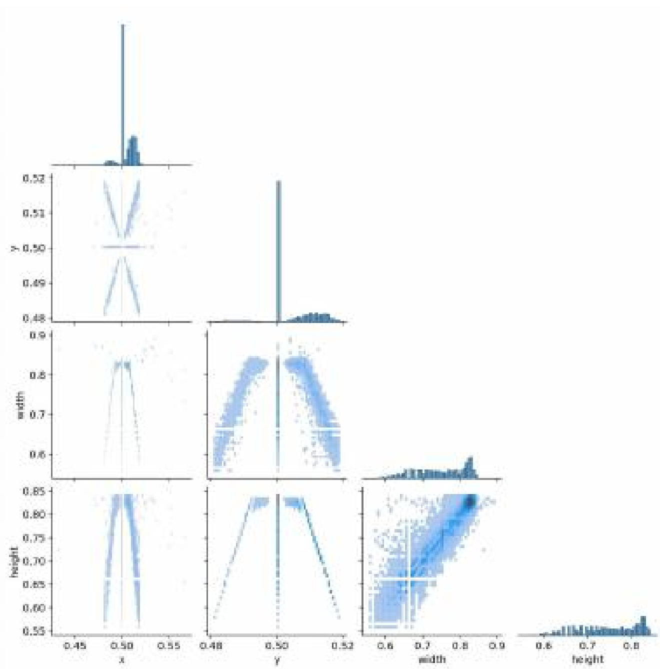

Fig. 6. Label correlation between x, y, width and height, which showing the frequency of label co-occurrence.

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. The precision and recall are defined with the following metrics:

$$Precision = \frac{TP}{TP + FP}, \tag{5}$$

$$Recall = \frac{TP}{TP + FN}, \tag{6}$$

Where TP means that the predicted value is the same as the real value, and both are positive samples. TN represents that the predicted value is the same as the real value, and both are negative samples. FP represents that the predicted value is a positive sample, and the real value is a negative sample. FN represents that the predicted value is a negative sample Negative samples, true values are positive samples [35], [36]. The following is our precision confidence curve and recall confidence curve which shows the change in precision and recall with confidence fluctuation.

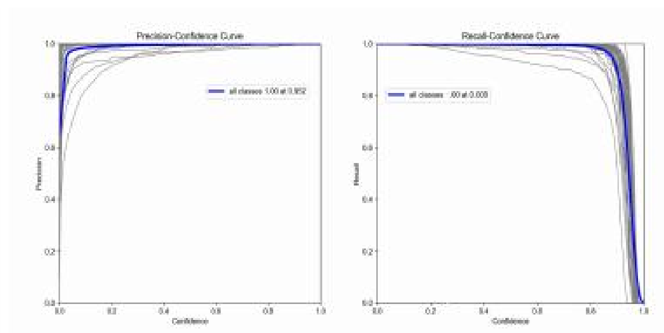

Fig. 7. Precision and recall of the model fluctuate during confidence change (**a**) Precision-Confidence Curve (**b**) Recall-Confidence Curve

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 [37], [38].

$$AP = \int_0^1 P(R)dR, \tag{7}$$

$$mAP = \sum_{i=1}^{C} \frac{AP_i}{C}, \tag{8}$$

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

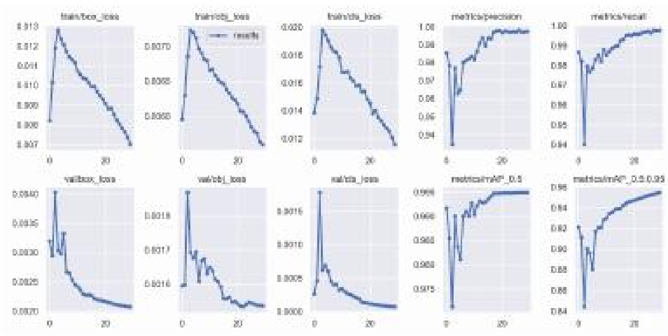

Fig. 8. Evaluation metrics of the model ranging from mAP@0.5 and mAP@0.5:0.95

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.

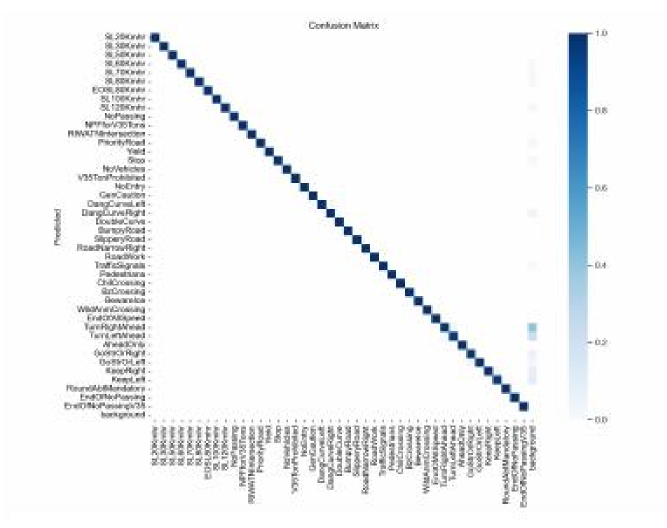

Fig. 9. Confusion matrix of the proposed model.

The proposed method can be compared with other methods working on the same dataset. In [13] 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.

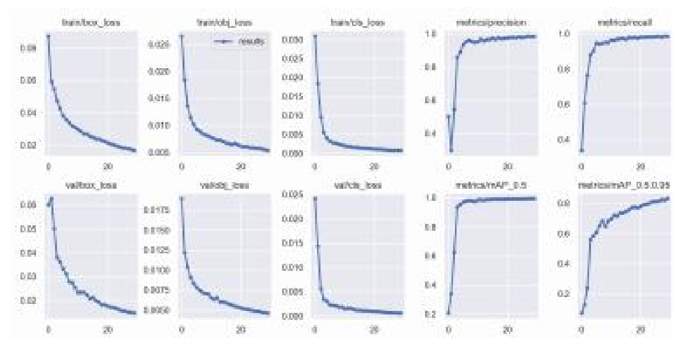

Fig. 10. Evaluation metrics of our model.

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 [39]. 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 [40]. 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] 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.

[14] S. Pei, F. Tang, Y. Ji, J. Fan, and Z. Ning, "Localized traffic sign detection with multi-scale deconvolution networks," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1. IEEE, Conference Proceedings, pp. 355–360.

[15] 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.

[16] 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.

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

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

[19] 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.

[20] 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.

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

[22] 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.

[23] 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.

[24] 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.

[25] Z. Qin and W. Q. Yan, "Traffic-sign recognition using deep learning," in *International Symposium on Geometry and Vision*. Springer, Conference Proceedings, pp. 13–25.

[26] S. P. Rajendran, L. Shine, R. Pradeep, and S. Vijayaraghavan, "Real-time traffic sign recognition using yolov3 based detector," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, Conference Proceedings, pp. 1–7.

[27] 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.

[28] 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.

[29] 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.

[30] 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.

[31] 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.

[32] S. Saleh, S. A. Khwandah, A. Heller, A. Mumtaz, and W. Hardt, "Traffic signs recognition and distance estimation using a monocular camera," in *6th International Conference Actual Problems of System and Software Engineering.[online] Moscow: IEEE*, Conference Proceedings, pp. 407–418.

[33] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *The 2011 international joint conference on neural networks*. IEEE, Conference Proceedings, pp. 2809–2813.

[34] Z. Shufang, W. Qinyu, Z. Tong, and L. Yuhong, "Detection and classification of small traffic signs based on cascade network," *Chinese Journal of Electronics*, vol. 30, no. 4, pp. 719–726, 2021.

[35] S. Singh, B. Kaur, S. Singh, M. Yadav, H. Sharma, and D. K. Ranjan, "Road sign recognition using lenet5 network model," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*. IEEE, Conference Proceedings, pp. 1–5.

[36] Y. Taki and E. Zemmouri, "An overview of real-time traffic sign detection and classification," in *The Proceedings of the Third International Conference on Smart City Applications*. Springer, Conference Proceedings, pp. 344–356.

[37] 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.

[38] H. Wang, "Traffic sign recognition with vision transformers," in *Proceedings of the 6th International Conference on Information System and Data Mining*, Conference Proceedings, pp. 55–61.

[39] M. M. William, P. S. Zaki, B. K. Soliman, K. G. Alexsan, M. Mansour, M. El-Moursy, and K. Khalil, "Traffic signs detection and recognition system using deep learning," in *2019 ninth international conference on intelligent computing and information systems (ICICIS)*. IEEE, Conference Proceedings, pp. 160–166.

[40] 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.