DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

Dual Architecture Deep Learning Based Object Detection System for Autonomous Driving

Mahmoud M. Mahmoud [1], Ahmed R. Nasser [2]
[12] *Control and Systems Eng. Dept., University of Technology, Baghdad, Iraq*
[1]*mahmud.muthana@gmail.com*, [2]*ahmed.r.nasser@uotechnology.edu.iq*

Abstract— *Object detection of autonomous vehicles presents a big challenge for researchers due to the requirements of accuracy and precision in real-time. This work presents a deep learning approach based on a dual architecture design of the network. A highly accurate multi-class network of convolutional neural networks (CNN) is presented for input data classification. A Region-Based Convolutional Neural Networks (Faster R-CNN) network with a modified Feature Pyramid Networks (FPN) is used for better detection of tiny objects and You Only Look Once (YOLOv3) network is used for general detection. Each network independently detects the existence of an object. The decision maps are then fused and compared to decide whether an object is present or not. Faster R-CNN with FPN model reported a higher intersection over Union (IoU) and mean average precision (mAP) than the YOLOv3. This approach is reliable demonstrating an upgrade on the existing state-of-the-art methods of fully connected networks.*

Index Terms— *autonomous driving, computer vision, deep learning, object detection*

I. INTRODUCTION

Interest in autonomous driving has grown enormously [1] due to the rise of deep learning and the progress of computer software, hardware, and processing power. One of the most essential components in autonomous driving perception systems is object detection. The detection task of occurrences of objects of a specific class (e.g., ‘car’, ‘pedestrian’, etc.) in images, attracted a great deal of attention due to its importance in many applications. Achieving detection with high performance and efficiency is crucial for a safe and functional driving system [2]. Object detection uses deep Convolutional Neural Networks (CNN) to extract features because of the CNN features’ discriminative representations. It contains semantic features that can detect objects better. CNNs are usually incorporated in a backbone network (for image classification) and a detection head [3]. The backbone learns the features of the image based on the Convolutional Neural Network (CNN) architecture whereas the detection head predicts the bounding boxes based on these features. Many spatial considerations are taken into account to improve both the efficiency and performance of the network. The main types of object detectors [4] usually are either two-stage approaches like Region-Based Convolutional Neural Networks (Faster R-CNN) [5] and Region-based Fully Convolutional Networks (R-FCN) [6] or single-shot detectors such as You Only Look Once (YOLO) [7-8] and Single Shot Detector (SSD) [9], the first is more accurate while the latter is generally faster. In two-stage detectors, Regions Of Interest (ROI) are generated and then handled through a deep learning network. In single shots detectors, the search is done over a probable combination of tiles in a single stage.

Received 27/1/2021; Accepted 29/4/2021

ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

The approach described below integrates both of the techniques to get the most accuracy possible while maintaining satisfactory performance. An architecture based on YOLOv3 for general detection and Faster R-CNN with modified Feature Pyramid Network (FPN) [10]-[11] for the detection of tiny objects. Tiny object detection is necessary in real-world applications and differs from general object detection in many aspects, for example, there could be less information from the target object while there're too many distractions in the background. The large Field-Of-View (FOV) features on input images sometimes can mean that tiny objects are captured from a long distance, making tiny object detection very difficult from various poses and viewpoints [11].

The rest of the paper is structured as follows: Section 2 describes the related work, section 3 provides the architecture of the model and the implemented state-of-the-art method, section 4 outlines the experimental results and evaluation of the model, and the final section concludes the paper with discussion and future applications.

II. RELATED WORD

Multi-channel neural networks have been used for a wide variety of applications in the literature. [12] developed a multi-channel two-stream (TM-CNN) model for multiple lanes for the projection of traffic speed with traffic volume effect in consideration by converting the raw traffic and volume data into spatial-temporal matrices so that the CNNs could learn these features and correlates between the lanes. [13] presented a model for pedestrian detection. Several detectors were used to extract proposals from data (RGB, and gradient magnitude), these proposals were converted into input channels for CNN classification. [14] developed a model that generated a three-channel image using spatial, temporal, and thermal information that can be fused as a CNN feature map for enhanced situational awareness detection. Using fast R-CNN the transfer learning techniques were used to generate the multi-channel images. Multi-channel networks are also used in other applications, [15] proposed a stack of YOLOv3 for mask detection in security checkpoints during Coronavirus disease (COVID-19). [16] designed a multi-class CNN to classify input images of liver lesions into sub-groupings of marginal and internal patches. Decisions were fused to classify binary and non-lesion decisions. [17] introduced Dual Denoising Network, a method for denoising images with sparse mechanisms for better generalization. It also fuses the global and local features for more precise denoising tasks. The network is composed of 4 parts: a feature extraction block, a compression block, an enhancement block, and a reconstruction block. [18] proposed a method that utilizes dual CNN architecture for classifying Polarimetric Synthetic Aperture Radar (PolSAR) images. The first part of the network extracts the polarization features whilst the other one extracts spatial features from the RGB image.

III. PROPOSED ARCHITECTURE

One of the most popular methods of performing image classification and object detection is based on the utilization of deep CNNs [19]. Therefore, the dual architecture of CNN networks for detecting objects is proposed as shown in *Fig. 1.* This approach will rely on architectures based on the existing state-of-the-art methods to ensure the best performance. The object detection models: YOLOv3 and Faster R-CNN with FPN. These models are trained from scratch and were chosen based on the promising results obtained from other detection tasks.

Received 27/1/2021; Accepted 29/4/2021

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

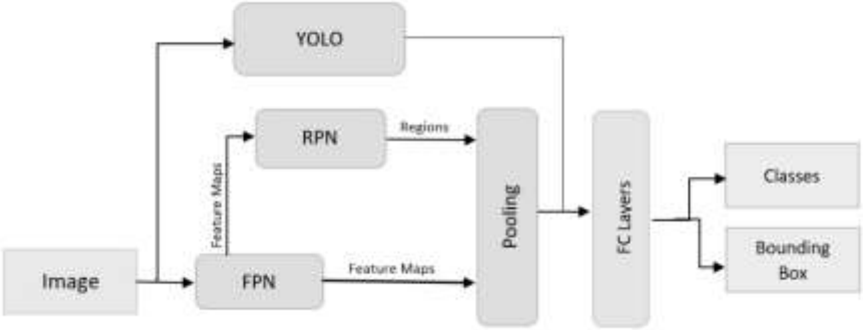

FIG 1: MODEL ARCHITECTURE OVERVIEW.

A. You Only Look Once (YOLO)

YOLO is one of the fastest regression algorithms for object detection. It combines all the components of the detector as it's working on the entire image instead of splitting the image into regions. The input image is divided into a grid, for every grid, a bounding box, and a confidence score is predicted [7]. The YOLOv3 [20] architecture has 20 convolutional and 5 max-pooling layers as shown in *Fig. 2*. The convolutional layers use kernels of size 3x3. Alternating the 1x1 convolution layers will reduce the feature space from previous layers. The convolutional block consists of Leaky Rectified Linear Unit (Leaky RELU), Convolution, Batch Normalization.

This model does not have fully connected layers so it can receive an image of any size as an input. An image size of 448 x 448 is adopted due to the good results it achieved. The number of filters is given by Eq. (1):

$$filters = (C + 4) \cdot A \qquad (1)$$

where A is the number of anchor boxes (A = 6), C represents the number of classes, in our case C = 2, Thus there are 36 filters in the last convolutional layers.

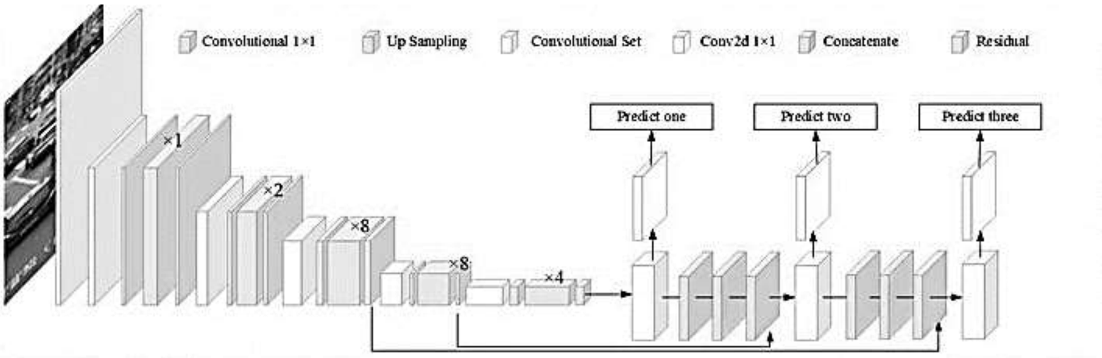

FIG. 2: YOLOV3 ARCHITECTURE [20]

B. Faster R-CNN with Feature Pyramid Network

Faster R-CNN [5] detection occurs in multiple (two) stages; the first stage is the region proposal network (RPN) where images are processed by feature extractors using a loss function.

Feature Pyramid Network (FPN) is used [10] as shown in *Fig. 3* for the task of extracting images. Multiple feature maps are created with better information quality than the regular Faster R-CNN. Instead of the common FPN network, a modified network is used with fusion factor so that deep layers deliver to shallow layers to control information

Received 27/1/2021; Accepted 29/4/2021

for tiny object detection adaptation. The second stage includes box proposals utilized to crop features for the intermediate map to be fed into the feature extractor to predict a class and a bounding box.

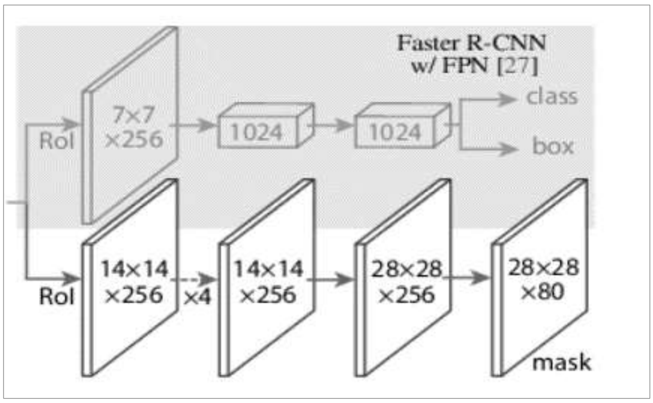

FIG. 3: FASTER R-CNN WITH FPN ARCHITECTURE [21]

C. Feature Maps Fusion

By using the multi-features of an image, feature fusion can complement the advantages of each approach to get more robust and accurate results. Many feature fusion functions can be used such as the summation, product, etc.… In this model, it’s done using the maximum function $y^{max} = f^{max}(x^a, x^b)$ this will use the feature maps of both methods for comparison, and it will take the larger value as the output result as explained in Table 1. and according to Eq. (2)

$$y^{max} = \max\{x^a, x^b\} \qquad (2)$$

TABLE 1: THE ALGORITHM USED IN THE METHOD

```
1. Input Image
2. Initialize weights, parameters, and models for the Faster R-CNN
   network.
3. Initlize weights, parameters and models for the YOLOv3 network.
4. For the Faster R-CNN network
   a. Generate the anchors required for detection.
   b. Generate the proposal layer.
   c. Compute the Feature Pyarmid Network Loss
   d. Use it for the Proposal layer target
   e. Generate ROI pooling
5. For the the YOLOv3 network
   a. Generate the Bounding boxes
   b. Classify the object using the frames and bounding boxes
6. Use feature fusion and compare the results of the two networks.
7. Output the Classification prediciton from the best result
8. Output the bounding box to indicate the location of the object
```

Received 27/1/2021; Accepted 29/4/2021

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

IV. EXPERIMENTAL RESULTS

The experiment was carried on the Karlsruhe Institute of Technology-Toyota Technological Institute (KITTI) dataset described in the next subsection. Experiments were conducted on a personal computer with an i5 processor, 8 GB RAM, and an NVIDIA GTX 1050 Ti GPU with a learning rate of 0.001, 1 batch size and, 3 epochs. F-score, Intersection over Union (IoU) and Mean Average Precision (mAP) are used for evaluating the model.

A. KITTI Dataset

Learning approaches became widely used in recent years and with the emergence of autonomous driving, the need for driving data also increased. In 2012 the KITTI Vision Benchmark [22] provided a large amount of labeled data for the driving scene. It continues to be one of the most widely used datasets in the driving automation context because of the large amount of labeled data available for different classes and the variation of synchronized data available (stereo color images, GPS coordinates, lidar point clouds). 3 main classes: Car, Cyclist, and Pedestrian are used. The model is trained with 2500 labeled images and tested with additional 2500 label images. Images are resized to 448x448. The system is compared to other methods based on the standard evaluation approach.

B. Evaluation and Results

The evaluation of the detection approach is performed by a pixel-to-pixel comparison between the predicted bounding box and the ground truth. The F-score, Intersection over Union (IoU) and Mean Average Precision (mAP) metrics are used. Computing the precision and recall first and then the F-score is needed.

The IoU value is usually a threshold between 1 and 0. If the value of the object is bigger than the threshold, the detection is classified as True Positive (TP). If the value is lower than the threshold then it’s a False Positive (FP). Failure of the ground truth to detect a value would classify the detection as False Negative (FN). Thus, the precision and recall are obtained by Eq. (3) and Eq.(4) subsequently

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \qquad (3)$$

And

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \qquad (4)$$

F-score describes the relationship between precision and recall, where it is shown in Eq. (5)

$$F_{score} = 2 \cdot \frac{precision\ .\ recall}{precision + recall} \qquad (5)$$

The (mAP) values indicate the accuracy of the object detection sets compared to ground truth. (IoU) is used to calculate the (mAP), it specified the amount of intersection between the predicted image and the ground truth, after calculating the mean of the interpolated precision at each recall level for each information, mAP can be calculated using Eq.(6). Table 2 provides the acquired results from the detection model alongside results from the KITTI benchmark suite website that shows the mean average precision only.

Received 27/1/2021; Accepted 29/4/2021

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

$$mAP = \frac{1}{n} \sum_{r \in \{0,0.1\ldots1\}} p_{interp(r)} \qquad (6)$$

where n is the number of interpolations used.

TABLE 2: RESULTS FROM THE MODEL MODIFIED FOR DETECTION OF AUTONOMOUS VEHICLES

Model	F-Score			mAP (%)		
	Car	***Cyclist***	***Pedestrian***	***Car***	***Cyclist***	***Pedestrian***
Proposed Model	0.94	0.91	0.90	91.63	91.12	90.23
YOLOv3	0.91	0.87	0.84	85.32	84.52	86.87
Faster R-CNN	0.93	0.91	0.92	91.21	90.23	92.78
Cascade-RCNN	-	-	-	93.37	-	-
YOLOv4	-	-	-	92.13	-	-
EPENet	-	-	-	91.11	-	-

The predicted bounding boxes and object detection is shown in *Fig. 4*. It should be noted that the proposed model is the most accurate due to the usage of the fusion factor of the FPN network and the fusion of inputs from the detectors. The performance of the model on a GTX 1050 Ti compared to other models is illustrated in Table 3.

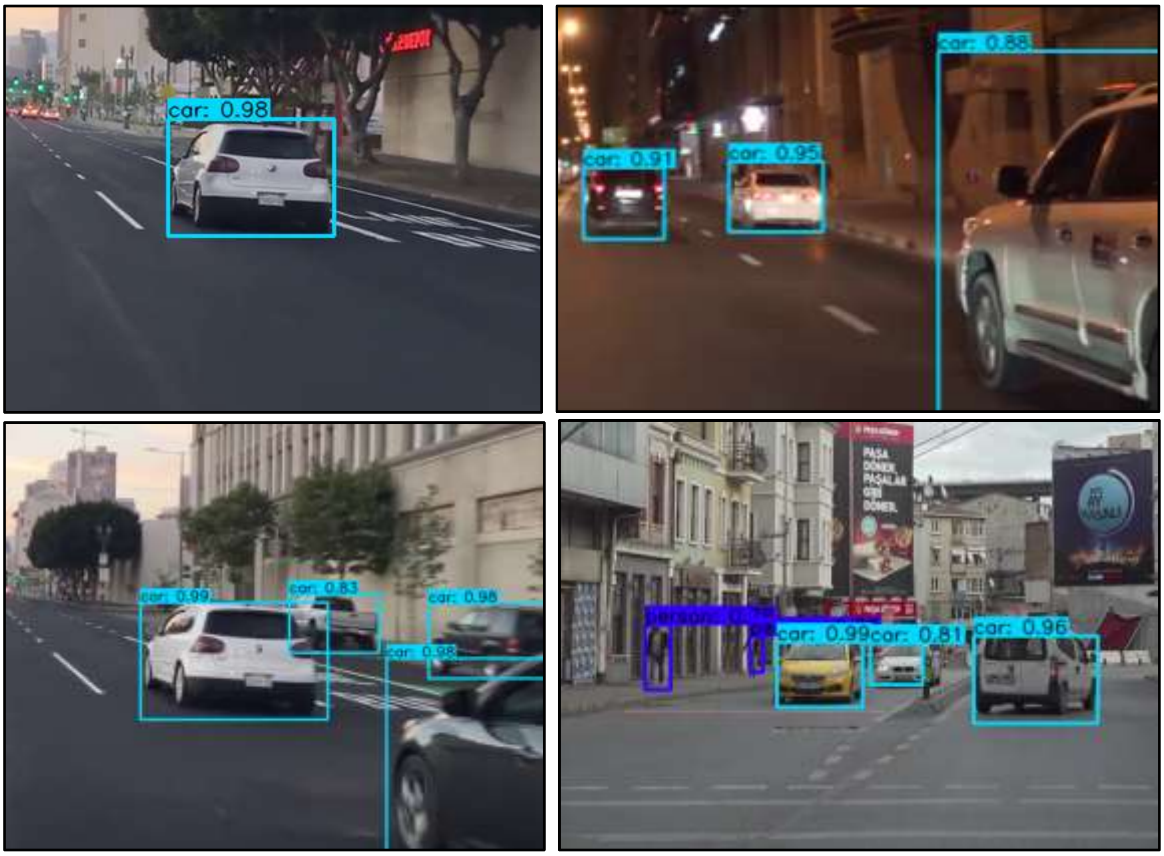

FIG 4: PREDICTED BOUNDING BOXES OF OBJECTS FOR MULTICLASS OBJECT DETECTION.

Received 27/1/2021; Accepted 29/4/2021

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

TABLE 3: MODEL PERFORMANCE COMPARISON

Model	Average Frames per Second (FPS)
Proposed Model	52
YOLOv3	85
Faster R-CNN	42

V. CONCLUSION

A method for using multi-CNN architecture for monocular object detection is presented and implemented for better detection of foreground-background scenes with tiny objects inspired by the fusion factor that affects tiny object detection performance. The approach suggests a bounding box by comparing the fused results of two object detection architectures. The detector is fast, performing better than the method presented in [5] achieving 52 FPS on GPUs. It outperforms other monocular approaches precision-wise and achieves better accuracy on the widely available object detection dataset KITTI. In future work, different networks and algorithms might be experimented with for improved results by configuring different detectors, different backbones, or different datasets. Similarly, more applications beyond object detection of the proposed model with the appropriate modifications might be applied.

REFERENCES

[1] S.Bagloee, Tavana, M., Asadi, M. and Oliver, T. “Autonomous vehicles: challenges, opportunities, and future implications for transportation policies”. *Journal of Modern Transportation*, 24(4), pp.284-303. , 2016.

[2] P., Koopman& M. Wagner “Challenges in Autonomous Vehicle Testing and Validation”. *SAE International Journal of Transportation Safety, 4*(1), 15-24. doi:10.4271/2016-01-0128, 2016.

[3] Z. Zhao, P. Zheng, S. Xu and X. Wu, "Object Detection with Deep Learning: A Review," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212-3232, 2019

[4] P. Soviany, & R. T. Ionescu (2018). “Optimizing the Trade-Off between Single-Stage and Two-Stage Deep Object Detectors using Image Difficulty Prediction”. 2018.

[5] S. Ren, K. He, Girshick, R., & Sun, J. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks” *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(6), 2017.

[6] D., Jifeng, Y. Li, Kaiming He and J. Sun. “R-FCN: Object Detection via Region-based Fully Convolutional Networks.” *ArXiv* abs/1605.06409 (2016): n. pag., 2016.

[7] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] J.Redmon, & A.Farhadi, “YOLOv3: An Incremental Improvement”, 2018.

[9] W. W. et al. “SSD: Single Shot MultiBox Detector”. *In: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision, 2016*.

[10] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature Pyramid Networks for Object Detection," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, 2017.

[11] Y. Gong, X. Yu, Yao Ding, Xiaoke Peng, Jian Zhao, Zhenjun Han; "Effective Fusion Factor in FPN for Tiny Object Detection" ; *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1160-1168, 2021.

[12] K, Ruimin et al. “Two-Stream Multi-Channel Convolutional Neural Network for Multi-Lane Traffic Speed Prediction Considering Traffic Volume Impact.” *Transportation Research Record: Journal of the Transportation Research Board 2674. 459–470. Crossref. Web*, 2020.

[13] D. Ribeiro, G.Carneiro, J. C Nascimento, J. C., & A.Bernardino, “Multi-channel Convolutional Neural Network Ensemble for Pedestrian Detection”. *Pattern Recognition and Image Analysis Lecture Notes in Computer Science*, 2017.

[14] S. Liu, et al. "Multi-Channel CNN-based Object Detection for Enhanced Situation Awareness.", 2017.

[15] S. S. Padmanabula, R. C Puvvada., V. Sistla, , & V. K. Kolli “Object Detection Using Stacked YOLOv3.” *Ingénierie Des Systèmes D Information*, 2020.

Received 27/1/2021; Accepted 29/4/2021

DOI: https://doi.org/10.33103/uot.ijccce.21.2.3

[16] F. Adar, Maayan et al. “Modeling the Intra-Class Variability for Liver Lesion Detection Using a Multi-Class Patch-Based CNN.” *Lecture Notes in Computer* , 2017.

[17] C. Tian et al. “Designing and Training of A Dual CNN for Image Denoising.” ArXiv abs/2007.03951, 2020

[18] F. Gao; T. Huang;, J. Wang; J. Sun; A. Hussain;, E. Yang “Dual-Branch Deep Convolution Neural Network for Polarimetric SAR Image Classification”, 2017.

[19] I. Namatēvs “Deep Convolutional Neural Networks: Structure, Feature Extraction and Training” *Information Technology and Management Science*, 2017.

[20] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[21] L. Tsung-Yi et al. “Feature Pyramid Networks for Object Detection.” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22] A. Geiger, P. Lenz, & R. Urtasun. "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite*" In Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Received 27/1/2021; Accepted 29/4/2021