International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 12, No.1 (Aug-2022)

https://dx.doi.org/10.12785/ijcds/120139

Improved YOLOv4 Approach:
A Real Time Occluded Vehicle Detection

Sunil Kumar[1], Manisha Jailia[2] and Sudeep Varshney[3]

[1]*Research Scholar, Department of Computer Science, Banasthali Vidyapith, Banasthali, Rajasthan, India*
[2]*Department of Computer Science, Banasthali Vidyapith, Banasthali, Rajasthan, India*
[3]*Department of Computer Science and Engineering, School of Engineering and Technology, Sharda University*

Received 6 Oct. 2021, Revised 10 Mar. 2022, Accepted 25 May. 2022, Published 6 Aug. 2022

Abstract: A major challenge in computer vision is detecting and tracking vehicles in real-time. However, existing algorithms fail to detect vehicles at high speed and accuracy. Therefore, an algorithm that detects vehicles with higher accuracy is required for surveillance in traffic scenarios. This paper proposed an improved algorithm for vehicle detection based on YOLO (You Only Look Once) Version 4 through Convolution Neural Network (CNN) and Hard Negative Example Mining (HNEM) dataset in the training process to improve the accuracy of the vehicle detection. In the end, videos are used to detect vehicles using a deep learning technique called You Only Look Once (YOLO). The test results indicate good real-time performance and high detection accuracy of the proposed algorithm. Several parameters such as accuracy, precision, recognition recall, F1, and mAP have been used to measure the proposed algorithm's performance. The experiments have proved that the proposed algorithm achieved satisfactory performance in real-time due to occlusion and change in viewpoint. Finally, our proposed algorithm achieves improved precision, recall and mAP compared to the existing algorithms for occluded vehicle detection.

Keywords: CNN, HNEM, YOLO, Real-Time Vehicle Detection , Occluded Vehicle

1. Introduction

As object detection technology matures, it is widely used across a variety of fields, such as image recognition, and has attained its top in the detection of traditional images [1]. Despite this, object detection technologies still need improvements when dealing with some scenarios like the weak, dim, and severely occluded detection of vehicles with complicated backgrounds in the infrared images [2]. Current research focuses on small object detection in the infrared spectrum without considering the impacts of complex backgrounds, which are additionally difficult to identify. Innovative research and improvements by the advancement of deep learning techniques have essentially advanced the progress in object detection through Convolutional Neural Networks (CNN). Object detection based on region-based CNN accomplished a great capability [3]. Therefore, this paper presents the methods of detecting the infrared objects affected by occlusion and change in viewpoint.

Images captured by infrared cameras are susceptible to a number of inherent defects, including a distance of image, response to changes in light and point of view. Also, the images might be effortlessly influenced by the radiation of the atmosphere and objects occluding the light flow in an insufficient imaging effect [4]–[6]. Furthermore, a significant amount of noise, the smaller contrast and the fuzzy boundary among the target and the background causes difficulties to detect infrared images compared to the normal datasets like MS COCO and ImageNet.

In the case of long-range aerial images in infrared, the target size of the image is much smaller as well as the resolution is too lower than the actual image, which typically has a mean pixel size of 30x30 [7]. There has been a growing interest in small-scale object detection in the object detection field. Multi-scale fusing [8] and feature fusion [9] are the two most common approaches to small-scale objects. Specifically, the difficulties of detecting objects in difficult backgrounds deliberated in this research are compounded by noise and small scales. Together with the obstructions of surrounding trees as well as other issues, this creates a huge disturbance to detecting model features. It is possible, in some cases, that in the background, non-object features can baffle the detector, leading to having false decisions, which would result in a maximum rate of false detection (i.e. low precision). On the other hand, there is a great practical consequence in studying how to increase the rate of accuracy in detection of the detector to allow it to have still high performance in complex environments with severe occlusive targets. This can liberate the human work-

E-mail address: sunildangikc107@gmail.com, manishajailia@yahoo.co.in, sudeep149@gmail.com

force from having to recognize large volumes of images, especially in the detection of military targets [10]–[12].

Serial methods like YOLO and their improved variants are complex in terms of their network structure and include a greater number of parameters. They need high-quality GPUs (Graphic Processing Units) which have the computational power to detect real-time objects [13]. These devices have limited computing power and memory, and they may need to detect objects in real-time in real-world applications for some mobile and embedded devices (for example, autonomous driving, augmented reality, and other smart devices) [14]. When a real-time inference is needed, including on smartphones and embedded video surveillance, the available computing resources are limited, such as low-powered embedded GPUs or even just CPUs with a limited amount of memory [15]. Because of this, real-time object detection on mobile devices and embedded devices remains a big challenge. Many researchers have promoted lightweight object detection techniques to solve the problem. Compared to conventional methods, lightweight methods employ fewer parameters and simpler network structures [16]. In turn, that means they do not require as much computing power and memory and can detect things faster. In general, they are more suited to being installed on mobile and embedded devices. Even though the detection accuracy of these sensors is low, it is sufficient for the actual requirements. Deep learning-based lightweight object detection methods can be applied in various applications, such as detecting vehicles, pedestrians, and passengers on buses, in agricultural applications, and in the detection of abnormal human behaviour.

A common problem is the shortage of infrared data with labels in remote sensing. The visible image dataset is huge, whereas the infrared image dataset is relatively small, which complicates the training procedure for detecting infrared objects. The idea is to introduce a mining block of hard-negative examples with the YOLOv4 model to solve the high false-positive rate caused by the complex background. A ratio of approximately 1:3 was used for the addition of negative and positive samples to ensure the balance of positive and negative samples during secondary training. As a final point, the accuracy rate in the detection of the modified YOLOv4 network model increased from 89.45 percent to 92.27 percent, proving that the improved model may satisfy the demand. In general, the proposed work makes these major contributions: A technique of secondary transfer learning is recommended to address the problem of limited datasets. It also works on challenging scenes involving highly occluded objects with complex backgrounds. The YOLOv4 model is enhanced by including Hard Negative Example Mining blocks to improve the accuracy detection.

The main contributions of our proposed work are mentioned as follows:

1) We propose an improved algorithm for occluded vehicle detection by applying augmentation policy on the YOLO model and computing the class confidence value of each bounding box and intersection over the union.
2) The CSPBlock module uses CSPDarknet53-tiny as the backbone of YOLOv4-tiny, which detects 371 frames to improve the accuracy rate in vehicle detection.
3) We obtain the state-of-the-art (SOTA) performance on both the MS COCO dataset and PASCAL dataset using YOLOv4 and Hard Negative Example Mining (HNEM).
4) The experimental results show that this proposed model achieved a better performance in terms of accuracy, precision, recall and F1-score for occluded vehicle detection.

The novelty of this study is that it proposes a YOLO-based detection model for occluded objects with HNEM and augmentation policy optimization. The hard-positive and hard negative detection of objects are to be extracted by a feature vector of the boundary with confidence score thresholding for this detection model.

The rest of the paper is organized as follows: In section 2, we briefly review the vehicle detection research and object detection due to occlusions; In section 3, the proposed work is elaborated; In section 4 some experiments are conducted to verify the performance of our proposed method by comparing it with other existing methods; and Finally, Section 5 concludes this research paper.

2. Related Work

Object control is a fundamental computer vision technology used in various industries, including healthcare, transportation, and medical services. The object detection method is indeed a computerized method of separating and identifying the objects of interest from its image's background [17]. However, unlike the object identification system, which merely recognizes specific objects, this skill acquired the location information of the concerned objects and perhaps even the location data of many types of objects in a single image. Deep learning algorithms are now being studied in relation to object detecting technologies [18], [19]. One-stage and two-stage detectors are two forms of deep learning computer vision algorithms. One-stage detectors are detection frameworks that recognize objects rapidly using the bounding box's size, with a dimension of the structuring element purposely designed to be helpful in a variety of contexts [20]. One-stage sensors like the YOLO, SSD, and RetinaNet allow fast identification by simultaneously running the region proposal and object separation even though such systems are based on real-time detection but less accurate and have low efficiency in detection [20]. Two-stage detectors models are sequentially operating the region proposal and object separation by involving the objects of interest (Region Proposal Network). Faster-RCNN and R-FCN systems are the two-stage detectors that

provide higher accuracy than one stage detectors, but the detection speed in real-time is low [21]. The YOLO (You Only Look Once) model was implemented with the single neural network by dividing a single image into grid cells, with each cell utilizing the attributes of the overall image. Figure 1 shows the YOLO model's approach for finding the objects of interest from the target image. The learnt channel is partitioned into M*N grid cells inside the model learning process, as well as the confidence score of every cell are analyzed simultaneously by calculating the confidence score of individual objects [22]. Finally, the models detect the objects of interest-based on the class's resulting image for predictions. In research on image classification, S. Jeon et al. developed a real-time roadway-driven lane recognition system based mainly on the YOLO paradigm for highway image classification [23]. The Jetson Nano Developer Kit and CSI camera are used in this system to gather and analyze driving data in various conditions. Faster R-CNN is the improved version of Fast R-CNN, which generates candidate regions by using RPN (Region Proposal Network) [24]–[28]. While extracting the candidate region, each pixel generates the multiple bounding boxes in the feature map in different scales, based on the offset of the corresponding anchors. The feature maps for classification and regression are routed to two networks when selected candidate regions [29]–[31]. The feature maps are sent into two networks for classification and regression when selected candidate regions. RPN classification is used to distinguish the regression in RPN predicts the offset coordinates between the anchor box and the ground truth, regardless of whether the corresponding anchor belongs to the foreground or the background.

3. Proposed Work

In this section, the YOLOv4-tiny model combined with Hard Negative Example Mining (HNEM) uses the CSP-Block module by using the CSPDarknet-53 as the backbone to improve the detection rate of occluded vehicles.

A. Analysis of Network Structure

In order to make Yolov4-tiny have a faster rate of object detection, it is designed from the Yolov4 method. Yolov4-tiny can detect objects at a rate of 371 frames per second using the 1080 Ti GPU with an accuracy that meets the demands of real-world applications. Therefore, mobile devices or embedded systems can use this object detection method with considerable ease. Instead of using the CSPDarknet53 network used in the Yolov4 method, this Yolov4-tiny method uses CSPDarknet53-tiny as the backbone network. Instead of using the ResBlock module in the residual network, the CSPBlock module is used by the CSPDarknet53-tiny network in the cross-stage partial network module. All the feature maps are divided by the CSPBlock module into two parts. It merges the two parts by cross-stage residual edges [32]. In this way, the gradient information propagates along with two different network pathways, resulting in an increased correlation difference. The convolutional networks can be more easily learned by using the CSPBlock module instead of the ResBlock module. With the Yolov4-tiny method for feature fusion, the feature pyramid network is used to extract feature maps with different scales to enhance object detection speed. This does not require the use of the path aggregation networks and spatial pyramid pooling available in the Yolov4 method. To predict the detection results, the Yolov4-tiny simultaneously uses two different scale feature maps: 13x13 and 26x26. Based on an input figure that has a size of 416x416 and a feature classification of 80, Figure 1 shows the structure of the Yolov4-tiny network.

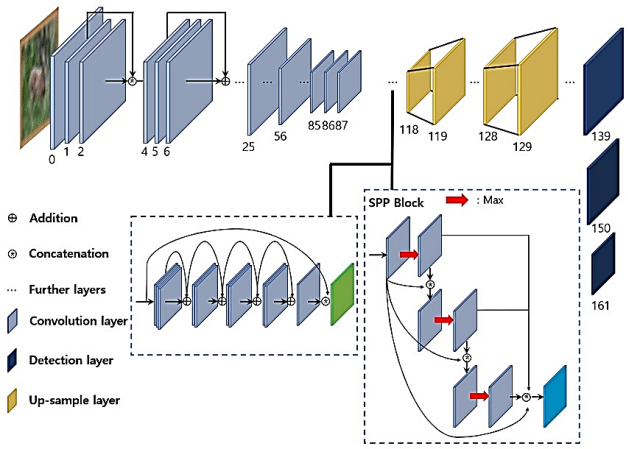


Figure 1. Structure of YOLOv4 Network [10]

B. Exploration of Hard Negative Example Mining

Despite the heavy occultation by trees in the infrared image, there are many differences between the background and target. That results in many confusing images. Because complex backgrounds will affect detection performance and increase false detections, an example mining module for hard negatives is proposed with the YOLOv4 model at the end [33]. As a starting point, it is necessary to calculate each bonding box's class confidence of C1 predicted using the classifier and the IOU values among the ground truth labels and the bounding boxes. As a general rule, a bounding box for which $C2 < 0 : 4$ is an IOU value indicates an FP (False Positive) sample, meaning it is expected to be an object but is, in fact, background information. The higher the class confidence C1 is for these bounding boxes, the harder it would be for the classifier to identify them correctly. Those are the kind of hard-negative examples used in this experiment. In this case, the samples are sorted according to the confidence value C1, thus forming the sample dataset DV as illustrated in figure 2. Three different YOLO Heads are included with the YOLOv4 model. Every head is fed with a different layer of scales. Consider that a scale of 608x608x3 is input. The three layers prior to the heads are 19x19 x1024, 38x38 x512 and 76x76 x256, respectively.

C. Occluded Object Framework using Hard Negative Example Mining

This paper proposes an algorithm to detect the occluded vehicle based on YOLOv4 and HNEM in real-time. This

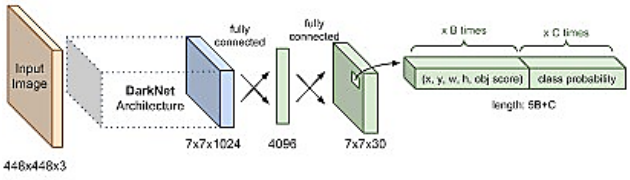


Figure 2. Hard Negative Mining Module based on YOLOv4

proposed algorithm is executed in three phases: Augmentation Policy on Model, second Hard Example Mining, and finally, YOLOv4 modeling. The framework based on YOLOv4 occluded vehicle detection model is shown in Figure 3. The data was collected for occlusion vehicle detection obtained from the IARA (Intelligent Robotic Autonomous Automobile) and GOPRO.

Above mentioned dataset contains a wide range of objects that may be found in highway vehicle dynamics. It also includes high-resolution picture datasets with respect to time, space and climates. The data was utilized to develop a model that detects objects that due to object occlusion in this research. The obtained data were subjected to nine enhancement policies to develop new training examples. Mix Up, Cut Out, Cut Mix, Mosaic, Blurring, Brightness, Contrast, Hue, and Gray Scale were some of the enhancement policies available [34]. These strategies are then proposed towards the main YOLO learning process and then obtaining the object bounding box regression value of performing the IOU (Intersection over Union) index. The comparisons were made with truth values, and favorable impact by predicting regression value on occluded crosswalk objects detection were permitted through all the gradient-based residual.

In the next stage, the hard example mining is done to obtain the hard-positive and hard negative examples after applying the augmentation policy on training data. The rate for false-negative detection and false-positive detection has

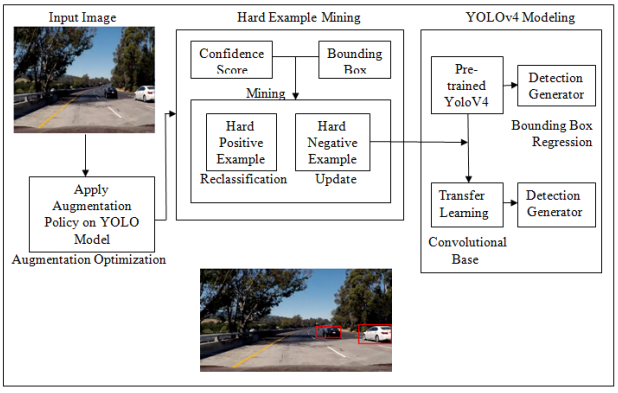


Figure 3. Working Model for Vehicle Detection due to Occlusion using HNEM

increased after calculating the confidence score and bounding box parameters. In order to overcome the problem of increment rate in false-negative detection and false-positive detection, reclassification and bounding box update were performed in the hard mining stage. YOLOv4 modeling is done based on a convolutional base and bounding box regression at the final stage. Therefore, the proposed method achieved a better result compared to the results of existing method [10].

D. Evaluation of Model Structure

Figure 4 shows the entire optimization process for the improved model using the HNEM optimizer. Since YOLOv4 has many more desirable properties than previous versions of YOLO [35]–[37], it has been adopted as the basis for the research. For example, the backbone has a much deeper structure than the previous Darknet-53 that is far more powerful than YOLOv2's Darknet-19, but ResNet-152 and ResNet-101 are significantly more efficient [21]. In addition, YOLOv4 provides the Cross-Stage-Partial, which can be used to improve both the accuracy and the speed of Darknet53 compared to YOLOv3.

Keeping the small-scale features in the cross-stage network and the residual part is possible. Thus, the original YOLOv4 will not require a change to the connection relationship. TABLE I and TABLE II show the comparison of the versions from the proposed work on the set of images and existing algorithms with the proposed work. The backbone structure, function loss, FPS, and mAP are compared on the Pascal VOC 2007 and the MS COCO. The YOLOv4 is better at detecting objects with higher accuracy and speed. As evidenced in the published literature, YOLOv4 models always perform better than YOLO and YOLOv2, the YOLOv4 model is selected for testing, and no comparison experiments are conducted on YOLO or YOLOv2 models. Before the training, the anchor box sizes are calculated based on the K-means cluster method. After the experiments, k D 9 is set and in the result nine different anchor box sizes are observed: (10, 25), (12, 44), (12, 38), (14, 23), (16, 32), (18, 55), (19, 22), (24, 26), (44, 35), and the image pixels size was set to 416 x416.

4. Results and Analysis

Table II shows the mAP and average processing time per frame along with AP (Average Precision), FPS and F1 Score for various models and Table III.Generally, Average Precision is used as an evaluation metric to detect the

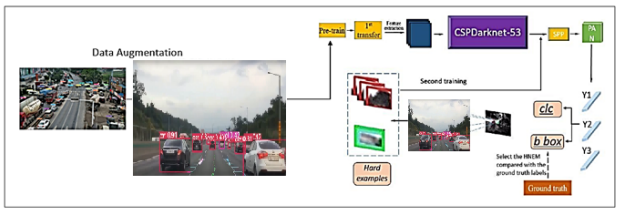


Figure 4. YOLOv4 Model Process with HNEM Optimizer [10]

Algorithm 1: Vehicle Detection Algorithm Using Hard Negative Example Mining for Occluded Vehicle

Input: n (set of trained input images)
Output: Groups (Set of *Hard_Negative*, *Hard_Positive*, *Positive_Example*, *Negative_Example*)
Method:
Initialisation:;
$C_x \longrightarrow Confidence\_Score$;
$PE \longrightarrow Positive\_Example$;
$NE \longrightarrow Negative\_Example$;
$HNE \longrightarrow Hard\_Negative\_Example$;
$HPE \longrightarrow Hard\_Positive\_Example$;
$IoU \longrightarrow$ Intersection over Union ;
$IoU_{pred\_bbox} \longrightarrow$ Intersection over Union of Predicted bounding box ;
$GT_{object} \longrightarrow$ Ground Truth Object ;
$GT_{bbox\_reg(n)} \longrightarrow$ bounding box region of ground truth object ;
$x \longrightarrow Position\_feature - vector$;
$y \longrightarrow Position\_example$;
$p \longrightarrow Hard\_Negative\_feature - vector$;
$q \longrightarrow Hard\_example$;
$C_x[n] \longrightarrow$ NULL;
$PE[n] \longrightarrow$ NULL;
$NE[n] \longrightarrow$ NULL;
$HNE[n] \longrightarrow$ NULL;
$HPE[n] \longrightarrow$ NULL;
while $i \neq n$) **do**
  if GT_{object} == *input images [n]* **then**
    IoU = *overlapped_region*;
  else if $IoU_{pred\_bbox} \&\& GT_{bbox\_reg(n)} \geq 0.5$ **then**
    $C_x[n]$ = Predicted Object(n) * IoU;
  else if $C_x[n] \geq 0.5$ **then**
    x = predicted bounding box region (n);
    y = ground truth class labelling (x[n]);
  Compute hard negative mining by comparing bounding box and reclassification;
  else if $IoU_{pred\_bbox} \&\& GT_{bbox\_reg(n)} < 0.5$ **then**
    $C_x[n]$ = Predicted Object(n) * IoU;
  else if $C_x[n] < 0.5 \| C_x[n] \geq 0.25$ **then**
    Compute hard positive mining with class reclassification;
  if $C_x[n] < 0.25$ **then**
    Compute negative example mining by setting threshold value of confidence score;
  else
    Ground truth objects are not present in the trained dataset
  ++i;
end

objects, but when binary coding is applied to alleviate the effects of imbalanced negative and positive examples, the score of F1 is also combined as a comprehensive rating scale. As compared to the other models, the original model of YOLOv4 has a greater average mean average precision and average processing time for the testing set. This may be because of the superior structure of the YOLOv4 model and the algorithm of cosine annealing used during the training process to enable a more optimal solution for the parameters. As a result of adding the hard-negative example mining module, there is an improvement in the model's average precision of about 1.58 per cent, and the score of F1 increased by 0.48 per cent, indicating that the HNEM module can affect the model's improvement. But with IOU D 0.5, there is an increment in the precision rate, whereas there is a slight decrement in the recall rate, indicating that the model is more able to distinguish hard-to-detect negative examples but not to detect hard-to-detect positive examples. The reason for this phenomenon may be attributed to a combination of factors and may be due to an imbalance of negative and positive samples within the datasets. The next step is to modify the loss function of the YOLOv4 modeling order to achieve the HNEM effect by balancing the positive and negative samples. Each model has its precise-recall curve, which is an intuitive way of identifying the differences in detection accuracy among the models.

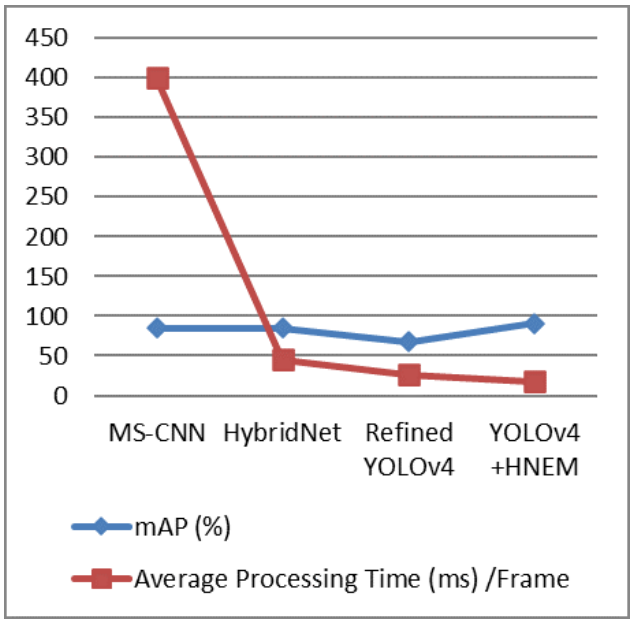


Figure 5. Comparison YOLOv4 with Existing Methods

Maps of validation losses and training for the YOLOv4 model are shown in figure 5, together with secondary transition, one-transfer, and without transfer learning. The objective of this study is to test the promoting performance of transfer learning in the proposed model. In accordance with figure 3, it can be seen that without transfer learning, the model could not converge to a very small value for the parameters, and their convergence speeds are much slower because of the inadequate datasets and the end test

TABLE I. Performance of the YOLOv4 + HNEM on Datasets

Images	Targets	Precision	Recall	mAP 0.5
130	530	0.94	0.947	0.953
130	91	0.946	0.935	0.954
130	78	0.934	0.953	0.946
130	43	1	1	0.957
130	65	0.932	1	0.958

TABLE II. Performance Comparison of Proposed Algorithm with Existing Methods

SNo	Techniques	Easy	Moderate	Hard	mAP(%)	Average Processing Time (ms) / Frame
1	MS-CNN [38]	90.03	89.02	76.11	85.05	400
2	HybridNet [39]	88.68	87.91	79.07	85.22	45
3	Refined YOLOv4 [8]	90.67	89.56	82.39	67.70	26
4	YOLOv4 + HNEM	93.45	92.34	84.25	91.05	18

TABLE III. The Comparison table of Backbone, GPU, Residual Network for proposed work

Method	Size	Backbone	GPU	FPS
YOLOv2 [28]	544 x 544	Darknet-19	(M)Nvidia Titan X	40
YOLOv3 [29]	416 x 416	Darknet-53	(M)Nvidia M40	35
YOLOv4 [1]	416 x 416	CSPDarknet-53	(M)Nvidia M40	38
FSAF [40]	800 x 800	ResNet-101	(P)Nvidia Titan X	276
Proposed	416 x 416	CSP Block-tiny	NVIDIA V100	317

results also confirm this. Training losses do not differ much between one transfer and secondary transfer learning. However, secondary transfer learning has a faster convergence speed, and also the validation loss is better matched to the training loss than one transfer learning.

An example of detection results is shown in figure 6, which pertains to a test set on YOLOv4 C HNEM. There are three boxes on the plot: in the blue box, GT (i.e. ground truth label) is displayed, whereas, in the green box, TP (i.e. true positive sample) was identified by the model, while in the red box, FP (i.e. False Positive) sample was identified by the model. Listed below are the connections between them. Essentially, class confidence is calculated for each bounding box C1 projected by the classifier and the IOU values C2 among each bounding box and the labelled ground truth labels. The bounding box whose C_2 < 0:4 IOU value indicates an FP (False Positive) sample, meaning it is projected as an object when it is the background information. This implies that the higher the C1 confidence level is, the harder it will be for the classifier to identify these bounding boxes correctly. These were the examples considered in this experiment to be hard negatives. The data were sorted in descending order for the sample dataset DV using the confidence value C1.

$$D = \{C_1^i | C_2^i < 0.4, C_1^i > C_1^j, 1 \leq i, j \leq N\} \quad (1)$$

$$D' = \{C_1^i | C_2^i < 0.4, C_1^i > C_1^j, 1 \leq i, j \leq n\} \quad (2)$$

With the above equations 1 and 2, it is evident that the more red boxes there are, the lesser the precision of the model is, while the fewer green boxes there are, and the lesser the recall rate of the model is. Thus, if the target is deeply occluded, the model is less likely to detect it, or some of the deceptive features may identify it in the background with a high degree of confidence, illustrated in the 3rd column of figure 7.The following equations 3, 4 and 5 show the precision, recall and F1 Score respectively.

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{GT} \quad (4)$$

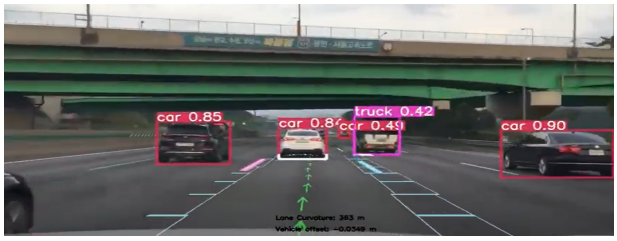
$$F1Score = \frac{Precision * Recall}{Precision + Recall} \quad (5)$$

In figure 7, a sample of the detection results of the 3 models is illustrated by occluded images of vehicles. It is possible to identify the misclassified objects from the non-improved models with the improved model. Therefore, incorporating the HNEM module in the real complex

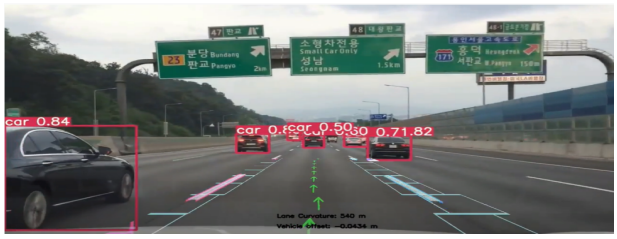
Figure 6. Detection Result based on YOLOv4 C using HNEM Testing Set

environment makes it possible to reduce the impact of misleading background information like trees and changes in light.

(a) Vehicle Detection in Normal Condition using YOLOv4 and HNEM



(b) Vehicle Detection on Highways with YOLOv4 due to Occlusion



(c) Vehicle Detection on Highways with Improved YOLOv4+HNEM

Figure 7. (a), (b), (c) Shows the Results of Vehicle Detection in Different Conditions

5. Conclusion

An infrared aerial image with a highly weak occluded vehicle detection using the YOLOv4 model was analyzed under complicated background conditions. The YOLOv4 model exploits the extra second-order transfer learning method to deal with the insufficient datasets problem. The hard-negative example mining technique simultaneously reduces the large rate of false detection due to the original model's complex background and occlusion influences. The detection and tracking of moving vehicles in real-time are one of the most challenging aspects of computer vision. Currently, available algorithms can detect vehicles, but they're slow and inaccurate. For surveillance in traffic scenarios, high-precision vehicle detection algorithms are necessary. This improves the surveillance system's ability to manage traffic since it enables the tracking of every vehicle with more accuracy. Using YOLO (You Only Look Once) V4, an improved algorithm for vehicle detection is proposed that increases vehicle detection speed and accuracy. In the end, videos are used to detect vehicles using a deep learning technique called You Only Look Once (YOLO). The proposed work uses the YOLOv4 model and HNEM by applying the data augmentation technique in real-time, which improves the accuracy rate of occluded vehicles. The advantage of this proposed work is to reduce the vehicle detection time compared with the existing methods and the limitation of this work is to achieve the better detection rate for heavy vehicle in changed climate conditions. Finally, our proposed algorithm shows that the precision, recall, and mAP are 94.6%, 95.3% and 91.05%, respectively. Consequently, the proposed method achieves better results than the other existing methods discussed in the manuscript.

References

[1] J. Sang, Z. Wu, P. Guo, H. Hu, H. Xiang, Q. Zhang, and B. Cai, "An improved YOLOv2 for vehicle detection," *Sensors (Basel)*, vol. 18, no. 12, p. 4272, 2018.

[2] H. Nguyen, "Improving faster R-CNN framework for fast vehicle detection," *Math. Probl. Eng.*, vol. 2019, pp. 1–11, 2019.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014.

[4] J. Cao, C. Song, S. Song, S. Peng, D. Wang, Y. Shao, and F. Xiao, "Front vehicle detection algorithm for smart car based on improved SSD model," *Sensors (Basel)*, vol. 20, no. 16, p. 4646, 2020.

[5] M. J. Shafiee, B. Chywl, F. Li, and A. Wong, "Fast YOLO: A fast you only look once system for real-time embedded object detection in video," 2017.

[6] H. Mao, S. Yao, T. Tang, B. Li, J. Yao, and Y. Wang, "Towards real-time object detection on embedded systems," *IEEE Trans. Emerg. Top. Comput.*, vol. 6, no. 3, pp. 417–431, 2018.

[7] F. Lu, F. Xie, S. Shen, J. Yang, J. Zhao, R. Sun, and L. Huang, "The one-stage detector algorithm based on background prediction and group normalization for vehicle detection," *Appl. Sci. (Basel)*, vol. 10, no. 17, p. 5883, 2020.

[8] V. Sowmya and R. Radha, "Heavy-vehicle detection based on YOLOv4 featuring data augmentation and transfer-learning techniques," *J. Phys. Conf. Ser.*, vol. 1911, no. 1, p. 012029, 2021.

[9] P. Mahto, P. Garg, P. Seth, and J. Panda, "Refining yolov4 for vehicle detection," *International Journal of Advanced Research in Engineering and Technology (IJARET)*, no. 5, 2020.

[10] S. Du, P. Zhang, B. Zhang, and H. Xu, "Weak and occluded vehicle

detection in complex infrared environment based on improved YOLOv4," *IEEE Access*, vol. 9, pp. 25 671–25 680, 2021.

[11] L. Cheng, J. Li, P. Duan, and M. Wang, "A small attentional YOLO model for landslide detection from satellite remote sensing images," *Landslides*, vol. 18, no. 8, pp. 2751–2765, 2021.

[12] Z. Wang, J. Zhan, C. Duan, X. Guan, and K. Yang, "Vehicle detection in severe weather based on pseudo-visual search and HOG–LBP feature fusion," *Proc. Inst. Mech. Eng. Pt. D: J. Automobile Eng.*, p. 095440702110363, 2021.

[13] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, 2019.

[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," 2015.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[16] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[17] D.-H. Shin, K. Chung, and R. C. Park, "Prediction of traffic congestion based on LSTM through correction of missing temporal and spatial data," *IEEE Access*, vol. 8, pp. 150 784–150 796, 2020.

[18] T. Tang, S. Zhou, Z. Deng, H. Zou, and L. Lei, "Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining," *Sensors (Basel)*, vol. 17, no. 2, p. 336, 2017.

[19] Y. Koga, H. Miyazaki, and R. Shibasaki, "A CNN-based method of vehicle detection from aerial images using hard example mining," *Remote Sens. (Basel)*, vol. 10, no. 1, p. 124, 2018.

[20] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sens. (Basel)*, vol. 13, no. 1, p. 89, 2020.

[21] X. Zhou, V. Koltun, and P. Krähenbühl, "Probabilistic two-stage detection," 2021.

[22] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020.

[23] S. Jeon, D. Kim, and H. Jung, "YOLO-based lane detection system," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 25, no. 3, pp. 464–470, 2021.

[24] S. Kumar and M. Jailia, "Smart cities with spatial data infrastructure and big data-a critical review," *International Journal of Advanced Studies of Scientific Research*, vol. 3, no. 11, 2018.

[25] S. Jamiya and S. Rani, *LittleYOLO-SPP: A delicate real-time vehicle detection algorithm. arXiv e-prints*, 2020.

[26] X. Feng, Y. Piao, and S. Sun, "Vehicle tracking algorithm based on deep learning," *J. Phys. Conf. Ser.*, vol. 1920, no. 1, p. 012065, 2021.

[27] S. Song, Y. Li, Q. Huang, and G. Li, "A new real-time detection and tracking method in videos for small target traffic signs," *Appl. Sci. (Basel)*, vol. 11, no. 7, p. 3061, 2021.

[28] A. Malta, M. Mendes, and T. Farinha, "Augmented reality maintenance assistant using YOLOv5," *Appl. Sci. (Basel)*, vol. 11, no. 11, p. 4758, 2021.

[29] A. I. B. Parico and T. Ahamed, "Real time pear fruit detection and counting using YOLOv4 models and deep SORT," *Sensors (Basel)*, vol. 21, no. 14, p. 4803, 2021.

[30] K. Itakura, Y. Narita, S. Noaki, and F. Hosoi, "Automatic pear and apple detection by videos using deep learning and a kalman filter," *OSA Continuum*, vol. 4, no. 5, p. 1688, 2021.

[31] Q. Wen, Z. Luo, R. Chen, Y. Yang, and G. Li, "Deep learning approaches on defect detection in high resolution aerial images of insulators," *Sensors (Basel)*, vol. 21, no. 4, p. 1033, 2021.

[32] S. Wu, G. Li, L. Deng, L. Liu, D. Wu, Y. Xie, and L. Shi, "$L1$-norm batch normalization for efficient training of deep neural networks," *IEEE transactions on neural networks and learning systems*, vol. 30, pp. 2043–2051, 2018.

[33] J. Hu, Y. Zhao, and X. Zhang, "Application of transfer learning in infrared pedestrian detection," in *2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC)*. IEEE, 2020.

[34] H. Xie and Z. Wu, "A robust fabric defect detection method based on improved RefineDet," *Sensors (Basel)*, vol. 20, no. 15, p. 4260, 2020.

[35] X. Zhang and X. Zhu, "Vehicle detection in the aerial infrared images via an improved yolov3 network," in *2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP)*. IEEE, 2019.

[36] A. Fadhil, "Optimization of COCOMO II model to estimate software cost using squirrel algorithm," *International Journal Of Computing and Digital System*, 2021.

[37] P. Kakade, A. Kale, I. Jawade, R. Jadhav, and N. Kulkarni, "Optic disc detection using image processing and deep learning," *International Journal Of Computing and Digital System*, 2021.

[38] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Computer Vision – ECCV 2016*. Cham: Springer International Publishing, 2016, pp. 354–370.

[39] X. Dai, "HybridNet: A fast vehicle detection system for autonomous driving," *Signal Process. Image Commun.*, vol. 70, pp. 79–88, 2019.

[40] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

Sunil Kumar Sunil Kumar received the B.E. degree in Computer Science Engineering from R.G.P.V Bhopal, MP, India and M Tech degree from NIT Bhopal, M. P, India. He is currently pursuing the Ph.D. degree in Computer Science Engineering, from Banasthali Vidyapith, Rajasthan, India. His research interests include Computer Vision, Deep Learning.

Manisha Jailia Manisha Jailia, currently working as Associate Professor in department of computer science, having 15 years of teaching experience. She published more than 30 papers in journals/international and national conferences. Her research interests include data analytics, data mining, data bases and software engineering.

Sudeep Varshney Sudeep Varshney received the B.E. degree in Computer Science Engineering from Dr. B.R.A. University Agra, UP, India and MS degree from BITS Pilani, Rajasthan India. He is currently pursuing the Ph.D. degree in Computer Science Engineering, from Indian Institute of Technology Dhanbad, India. Currently he is working as Assistant professor in Sharda University, Greater Noida, UP, India.