A Customized YOLO NAS Model for Vehicle Detection on Indian Roads

Balvindersingh Bondili[1(✉)] and Ram Prasad Reddy Sadi[2]

[1] AUTDR-Hub, Andhra University, Visakhapatnam, AP, India
balvinder546@gmail.com
[2] Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, AP, India

Abstract. The importance of obtaining details regarding transportation vehicles has been increasing in developing countries such as India. Object detection plays an important role in the automatic identification of the content of an image or video without human intervention. Various deep learning models have been developed for object detection using CNNs (convolution neural networks). This paper proposes a method for vehicle detection from still images using an optimized YOLO-NAS (You Only Look Once-Neural Architecture Search) frame work. This model is verified with earlier YOLO models for improved accuracy and optimization. The experiments were conducted on two publicly available datasets. Indian Driving Dataset (IDD) and DATS_2022 having exclusively images of various traffic scenes on Indian roads. The proposed method out performs the existing object detection models in terms of detection accuracy. Results show that the proposed method is good at detection accuracy measured in Average Precision and Recall.

Keywords: Object detection · Deep learning · YOLO · CNNs

1 Introduction

In the field of "computer vision", object detection is the basic task which localizes and identifies specific parts of an image or video for understanding required information [1]. The remarkable advancement of deep learning methodologies in the past decade has significantly aided in the advancement of object detection [2], resulting in out- standing discoveries and bringing it to the forefront of study with never-before-seen focus. The difficulty of various detection tasks may differ from one another since they have completely distinct goals and limitations. In addition to a few common challenges in other vision tasks like "objects under different viewpoints, illuminations, and intraclass variations, common challenges in object detection include object rotation and scale changes, precise object localization, occluded object detection, and speed up of detection", etc.

Detecting objects is a difficult task since there is minimal inter-class variance and large intra-class variance [3]. Different objects within a single class, such as persons posing differently or dressed differently in an image, might result in high intra-class variance. When objects from different classes have similar looks, there is a low inter-class

P. Pareek et al. (Eds.): IC4S 2024, LNICST 599, pp. 237–249, 2025.
https://doi.org/10.1007/978-3-031-77081-4_19

variation. Concepts like object localization, classification, and recognition in computer vision are examples of processing related to object detection. "Object classification" designates the labels for one or more items in the image and specifies their class. Using a bounding box that directs it, object localization locates one or more objects' locations in an image or video. Object detection is the process of combining the localization and classification of an object [4].

Deep Learning techniques became so popular and widely used for object detection and recognition related tasks. The core of modern object detection techniques is represented by convolution neural networks (CNNs). They are employed in feature extraction. There are numerous CNNs available, including "AlexNet, VGGNet, and ResNet". These networks have been tested on several popular benchmarks and datasets, including ImageNet, and are mostly used for object classification tasks. A single object in an image is classified by the classifier, which also outputs a single category for each image and provides the likelihood of a class match [5]. This process is known as image recognition or classification. In contrast, object detection requires the model to identify several items in a single image and provide the coordinates corresponding to each object's location. Figure 1 shows the general representation of how object detection looks like in an image.

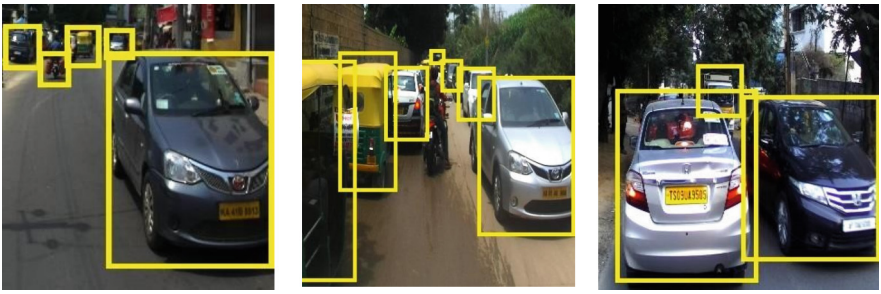


Fig. 1. Examples of vehicle detection for Indian traffic scenario

The object detection challenge of any given category is heavily dependent on the quality of the input image. Some example categories of object detection in real world applications are "face detection", "text detection", "traffic sign detection", fire detection and autonomous driving [6–9]. Creating more expansive datasets with reduced bias is crucial in order to develop effective detection systems. Many popular detection datasets have been made available in the past ten years [10]. Indian Driving Dataset (IDD) and DATS_2022 are the two publicly available datasets for object detection designed for Indian scenario.

The remaining part of the paper is structured as follows. Brief overview on existing deep learning techniques for vehicle detection is presented in "Sect. 2". "Section 3" describes the proposed optimized YOLO NAS framework. "Experimental results and analysis" are presented in "Sect. 4" and the paper is concluded in "Sect. 5".

2 Related Work

Vehicle detection is a vital phase in the design of Intelligent Transport Systems (ITS). Due to camera position, background fluctuations, occlusion, and many foreground objects; vehicle recognition on urban roads is challenging [11]. This section covers the existing methods for vehicle detection using deep learning models.

In [12], the authors compared and trained five separate deep-learning algorithms for detecting road vehicles: R-CNN, SSD, and YOLOv3. The KITTI [13] uses the same training set that was applied to the public database. Three metrics assess the overall algorithm performance: (1) "recall and precision rates" on the KITTI test set; (2) "average precision"; and (3) frame rate (fps). Liu et al. [14] developed faster RCNN-based "two-stage detectors" for detecting generic objects in complicated traffic situations, taking into account large-scale variance. However, it has proven inadequate for detecting small vehicles in real-world applications. Wu et al. [15] proposed a multi-camera vehicle identification system that employs a unique multi-view "region proposal network" to locate potential automobiles on the ground plane. This approach uses multiple cross-camera views to determine the vehicle's position on the ground plane. In addition; it can identify partially and highly obscured vehicles in traffic scenarios.

Yang et al. [16] used a deep CNN for best-performance "object detection". They also devised a data association-based target tracking approach that included appearance features and motion state estimates via the Kalman filter. An updated SSD-based vehicle identification algorithm improves accuracy, particularly for small cars developed [17]. This method improves SSD speed by adding an inception block to the extra layer before prediction. Then a more appropriate "Automatic Vehicle Detection (AVD) approach" is used to set the "scales and aspect ratios" of default bounding boxes, resulting in improved position "regression" and faster performance. In [18], a "vision-based object detection" framework for autonomous driving was developed. The system includes an optimized model based on the YOLOv4 structure to detect various categories of objects and a fine-tuned part. The affinity fields approach was created, and eXplainable Artificial Intelligence (XAI) has been used to improve approximations during risk evaluation.

The "YOLO v2 and YOLO 9000" models were examined in [19]. Their ability to detect and classify items in films in real time makes them helpful for a range of applications. The YOLOv2 is highly effective at identifying and classifying basic things. The GPU features, the Anchor Box technique was employed to achieve the required speed and precision. Moreover, it recognizes object motion in videos with accuracy. A real-time model called "YOLO9000" bridges the gap and maximizes detection and categorization. "YOLOv2 and the YOLO 9000 detection system" are capable of identifying and categorizing a wide range of things, such as multiple examples of the same object or multiple examples of distinct objects. The paper [20], describes the development of YOLOv4, a strong vehicle detection model. This model uses an "attention mechanism to suppress interference in images" based on channel length and spatial dimensions. Furthermore, a modification to the Path Aggregation Network(PAN) has been done which incorporates the Feature Pyramid Network (FPN) to boost its effectiveness. Positioning items steadily in 3D space enhances vehicle detection and classification effectiveness.

The work in [21] demonstrates using the YOLO-v5 architecture to recognize and classify motor vehicles in publicly available datasets. This work uses transfer learning to fine-tune the weights of pre-trained models of YOLO-v5 architecture. The authors collected large amounts of data on congested traffic patterns to apply transfer learning. The datasets were expanded to include features such as traffic density, occlusions, and meteorological conditions. "A type-1 fuzzy attention" (T1FA) [22], in which "fuzzy entropy is used to re-weight the feature map to lower the uncertainty of the feature map" and facilitate the focus on the target center as to efficiently increase "vehicle detection accuracy". Furthermore, to recognize cars of various sizes more efficiently, mixed-depth convolution is used.

3 Proposed Methodology

This section presents the methodology used for vehicle detection in Indian traffic environmental conditions. YOLO models have been used for real-time object detection. The latest model of the YOLO family is NAS (Neural Architecture Search) which outperforms earlier models and pre trained on large data sets for object detection. In this paper, we applied this model as custom vehicle detection for Intelligent Transportation System (ITS).

Deci AI's state-of-the-art object detection model, "YOLO-NAS" optimizes its architecture autonomously through the use of "Neural Architecture Search (NAS) technology". This enhances the model's ability to identify objects in real-time and makes it suitable for production-grade performance. One important feature of "YOLO-NAS" is its quantization-friendly basic block, which addresses earlier constraints in YOLO models and allows deployment on resource-constrained devices like as smart phones and IoT devices [23]. This model in Fig. 2 employs advanced training methods and post-training quantification approaches, including knowledge distillation, mixed precision training, and selective quantization. The following are the key features of YOLO NAS:

1. Quantization-Friendly Basic Block: YOLO-NAS includes a new basic block that is quantizable, solving one of the major short comings of earlier YOLO models.
2. Sophisticated Training and Quantification: YOLO-NAS improves performance by utilizing advanced training schemes and post-training quantization.
3. AutoNAC optimization: pre-trained on popular datasets such as Roboflow 100, Objects365, and COCO. This pre-training makes it ideal for future item detection jobs in production contexts.

The model has 3 primary components: a "backbone, neck, and head". The backbone, a "convolutional neural network", pulls features from incoming images, while the neck analyzes and prepares them for object detection. The head is in charge of detecting actual objects, predicting "bounding boxes", and assigning "class probabilities" to each grid cell. "YOLO-NAS" is equipped with a hierarchical deep CNN structure. The backbone has convolutional and pooling layers, whilst the neck and head use sophisticated techniques like feature fusion and attention processes. Figure 3 shows the work flow of the proposed method.

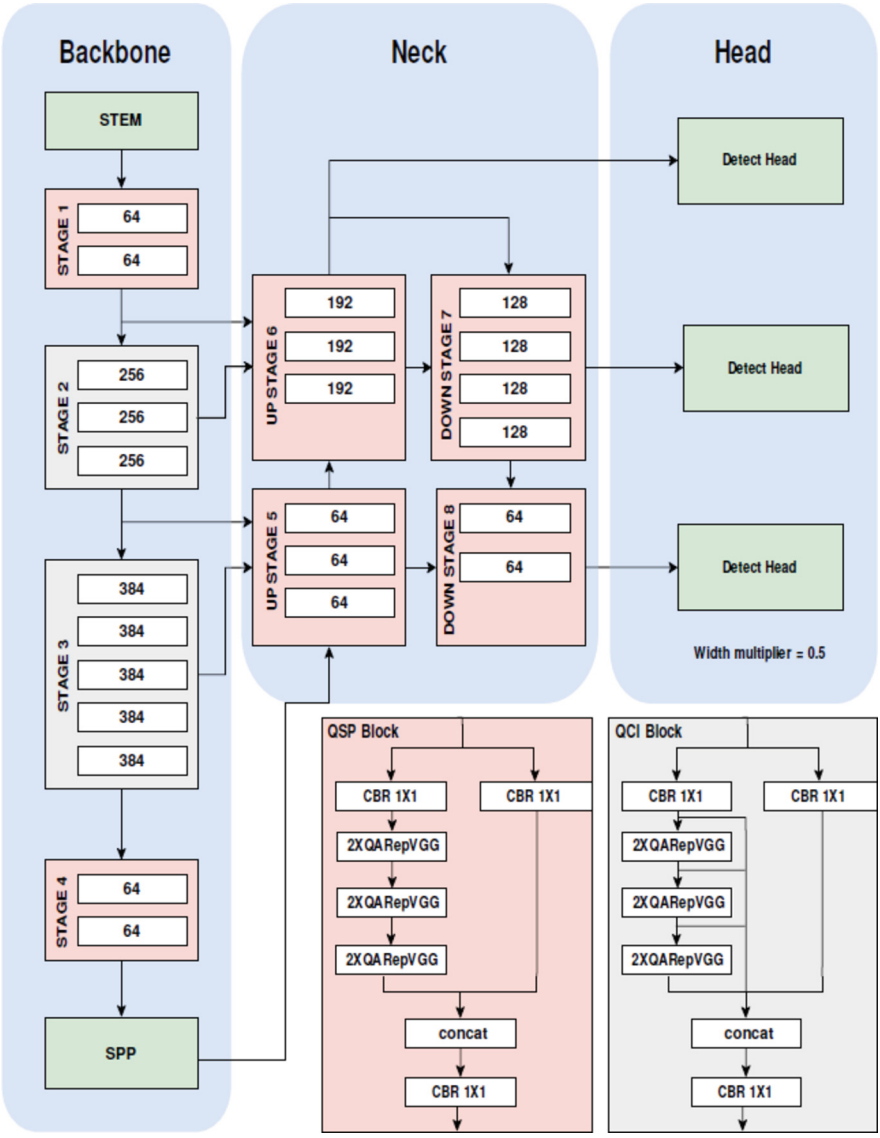


Fig. 2. Architecture of YOLO NAS [23]

A deep learning model is trained using 'N' annotated images {x1, x2,…, xn}. For each image xi, there are Mi objects that fall into C categories:

$$Y_i = \left\{\left(c_1^i, b_1^i\right), \ (c_2^i, b_2^i), \ldots\ldots, (c_M^i, c_M^i)\right\} \tag{1}$$

Where (c^i, b^i) denotes labels of the j object in x respectively.

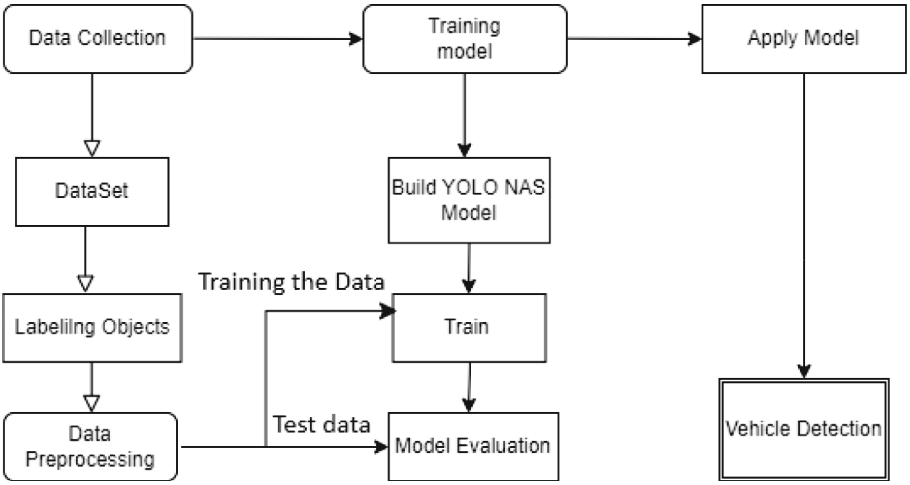


Fig. 3. Workflow of the proposed method

3.1 Data Collection

Using standard datasets such as PASCAL VOC, and MS COCO to train the vehicle detection model is not the best option [24]. Not every kind of vehicle category is present in these datasets. Car and bus are the only two class labels found in PASCAL VOC 2007 and 2012. While the three classes of the MS COCO 2014 are truck, bus, and car.

The objective of this paper is to produce research results of vehicle detection for Indian scenario. Hence we have taken into consideration of Indian datasets namely, IDD and DATS_2022.

3.2 Data Annotation and Augmentation

Categorization and labeling of data is the process of data annotation. Using the labelling tool, all of the photographs in this work are tagged various categories: car, two-wheeler, auto, lorry, bus, and truck. In order to achieve consistent vehicle detection in later phases, it is the correct technique of labeling the classes of the datasets. A critical first step in ensuring that the CNN model is properly trained and produces promising results is *data annotation*. To improve the diversity of data utilized for model training, data augmentation is also employed. Three transformations: horizontal flip, rotation, and shift have been used in this work to balance the dataset.

We obtained the dataset and used empirical analysis to apply the following augmentation strategies:

1. Cropping: At this point, we crop the dataset images to a minimum zoom of 0% and at most of 15%.
2. Saturation: We alter the color ranges of the images in order to get better results. We saturate pictures by about ±20% in this investigation.
3. Brightness changes: Care has been taken to vary the brightness of the image dataset. We made about ±20% of the photos brighter and darker.

An object detection model can utilize the data augmentation technique once it has been applied to the image dataset. We have 3180 test images, 2166 validation images, and 8980 training images overall taken form IDD and DATS_2022 datasets, as can be seen.

3.3 Training the Model

The training and validation datasets were given to the YOLO-NAS model once the data had been collected, processed, and annotated. We choose arrange of parameters for training, such as "batch size, epochs, and image resolution". The method receives validation dataset, a testing path, and a training path. It is observed that when train in our model from the beginning, we must start it with a set of random weights. Because pre-trained COCO weights save a significant amount of time and simplify computations, we decided to employ them for our model training. We obtain the optimal weights following transfer learning by using the pre-trained YOLO-NAS model.

In addition, the batch sizes have been changed to 5, 10, and 20. We've also modified the epochs to 20, 50, and150 to 200. In the present research, the confidence interval is 0.35 to 0.5. After the training phase is over, we utilize the most efficient weights to detect items in the dataset. Finally, we retrieve the projected label values as well as the test pictures' confidence values and bounding boxes.

4 Results and Analysis

The experiments were carried out using the simulation platform *Roboflow*. It is a computer vision development platform that enables better "data gathering, preprocessing, and model training procedures". It has the best computing capabilities for deep learning models for real time applications. Roboflow Train is an AutoML tool that allows developers to train a cutting-edge computer vision models on dataset [25].

The Fig. 4 below shows the plots of "box loss, objectness loss and classification loss" for the proposed object detection model. The x-axis represents the training epochs upto 200 and y-axis shows the loss. After 100 epochs the loss has been almost reduced to improve the model performance.

The box loss shows how successfully the algorithm locates an object's center and how well an object is covered by the predicted bounding box. In essence, objectness is a probability measure for the presence of an object in a proposed area of interest.

An object is probably present in the image window if the objectivity is high. Classification loss provides an estimate of the algorithm's accuracy in predicting an object's right class.

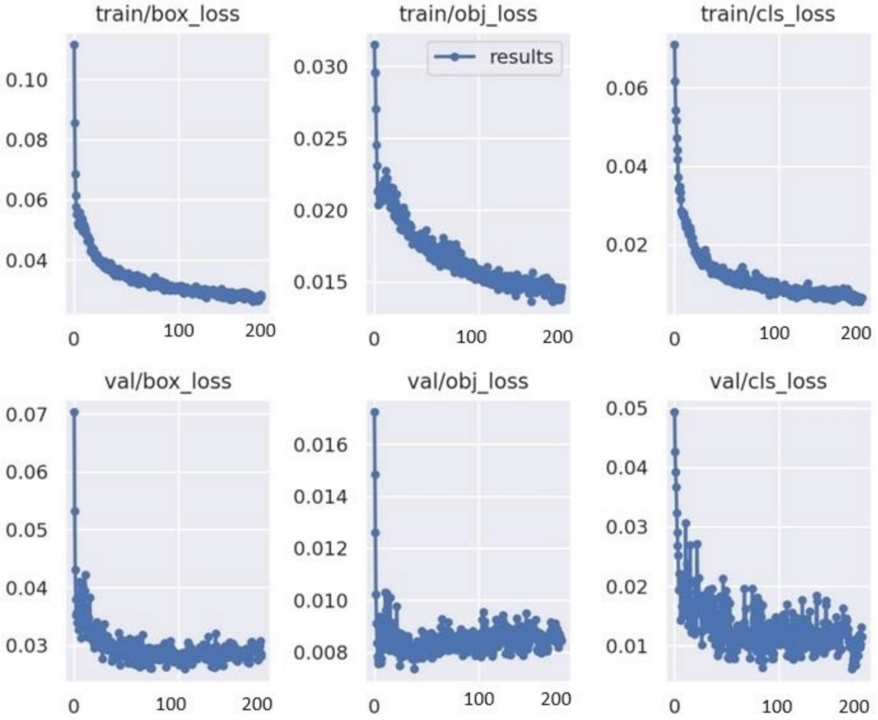


Fig. 4. Training details of the proposed vehicle detection model

"Precision, recall, and mean average precision" all increased quickly in the model before reaching a plateau after roughly 100 epochs. The validation data exhibited a sharp decrease in the "box, objectness, and classification losses" until around epoch 100. Early stopping allowed us to choose the ideal weights. However, another 50 epochs were used for keeping in mind of better training and validation accuracy. But the model seems to be overfit, hence training stopped at 200 epochs. Plots of mean Average Precision (mAP), recall, and precision over the training epochs are displayed in Fig. 5.

In recent years, "Average Precision" (AP), which is the average detection precision under different recall scenarios and is assessed according to a class-specific methodology, has become a widely used metric for object identification evaluation [26]. A final metric for evaluation in the object detection and related domains is mean Average precision (mAP), the average of all object categories. This allows users to compare the effectiveness of all the classes of objects.

The datasets used in this research were having images of Indian road scenes with different classes. The images included in the IDD dataset were taken with an automobile's front-facing camera [27]. The vehicle was driven through the outskirts of Bangalore and Hyderabad. The majority of the photographs have a resolution of 1080p, but there are also those that have 720p and other resolutions. The comprehensive dataset DATS_2022 includes pictures of Indian traffic situations from both urban and rural areas. There are almost 10,000 photos in the dataset, representing 45 different item classifications. The

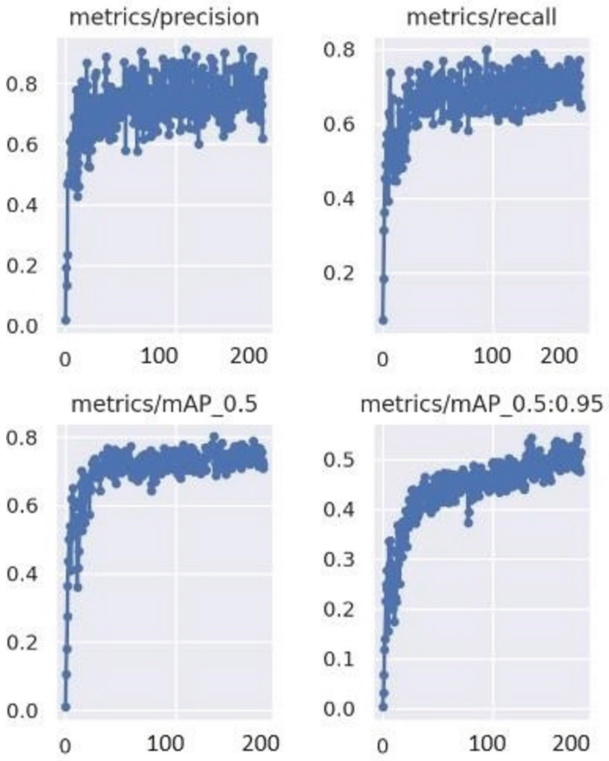
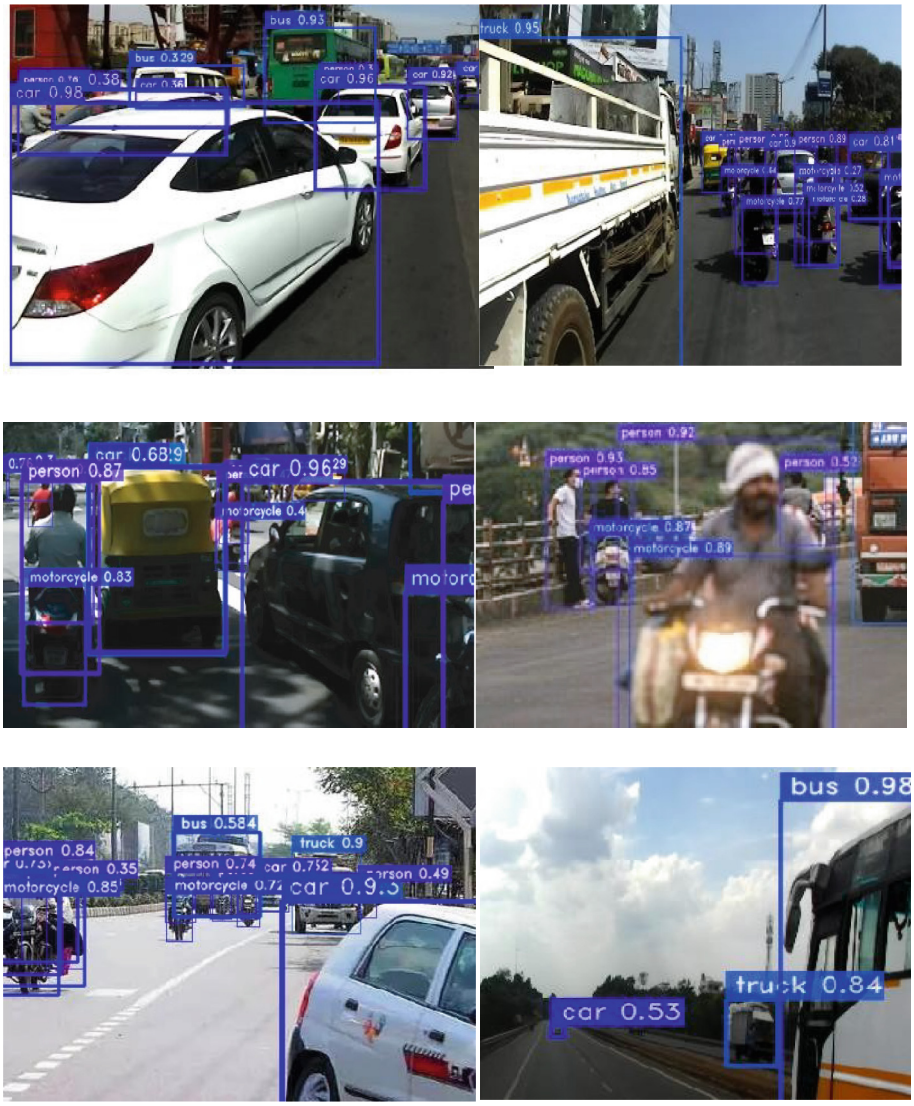


Fig. 5. Plots of model performance by means of Precision, Recall and mAP

dataset is accompanied by over 7000 annotations in various forms. The pictures were taken at various times of the day and in every season of the year. These seasons had distinct weather, and when the place shifted, the local climate likewise did [28]. Figure 6 shows the experimental vehicle detection results on these datasets and Fig. 7 shows detection results of sample images taken by our camera at different locations of Guntur, Vijayawada and Visakhapatnam cities of India.

The performance comparison of the YOLO NAS with earlier models has been done using "mean Average Precision" (mAP) with IoU of 0.5. Various YOLO models have been developed for object detection using deep learning techniques. Table 1 shows the performance of the proposed model on IDD and DATS_2022 Indian datasets. Results show that YOLO NAS gives better values of mAP that is on average 0.86 (86%) on IDD, DATS_2022 and sample images.



(a) (b)

Fig. 6. Detection results on (a) IDD dataset, (b) DATS_2022 under different lighting conditions

Fig. 7. Detection results on sample images

Table 1. Performance comparison of YOLO NAS with existing models

Dataset	Number of images used				mAP_0.5				
	Training	Validation	Testing	Faster-RCNN	YOLO v2	YOLO v3	YOLO v6	YOLO v8	**YOLO NAS**
IDD	4120	1083	1600	0.72	0.68	0.64	0.78	0.77	0.84
DATS_2022	4860	1083	1580	0.75	0.67	0.66	0.78	0.77	0.85
Sample Images	280	120	140	0.79	0.74	0.78	0.80	0.82	0.88

5 Conclusion

In Computer Vision, Object detection has been playing a major role in daily life in almost all applications. Due to the widespread usage of vehicles on roads in developing countries like India, detecting vehicles is an important task for further applications of ITS. This paper proposes YOLO NAS model on two Indian datasets for detection of objects on roads. Experimental results show that this model performs far better than the earlier models that were used for Indian conditions. Thus, this model can be used in ITS for Indian scenario for the detection of vehicles. This model can be further enhanced by considering more training images under different climatic conditions and incorporating self attention mechanism.

References

1. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Patt. Anal. Mach. Intell. **32**(9), 1627–1645 (2009)
2. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning Nature **521**(7553), 436–444 (2015)
3. Zhao, Z.Q., Zheng, P., Xu, S.T., Wu, X.: Object detection with deep learning: a review. IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019)
4. Diwan, T., Anirudh, G., Tembhurne, J.V.: Object detection using YOLO: challenges, architectural successors, datasets and applications. Multimedia Tools Appl. **82**(6), 9243–9275 (2023)
5. Liang, T., et al.: CBNet: a composite backbone network architecture for object detection. IEEE Trans. Image Processing **31**, 6893–6906 (2022)
6. Dhillon, A., Verma, G.K.: Convolutional neural network: a review of models, methodologies and applications to object detection. Prog. Artif. Intell. **9**(2), 85–112 (2020)
7. Jha, S., Seo, C., Yang, E., Joshi, G.P.: Real time object detection and tracking system for video surveillance system. Multimedia Tools Appl. **80**, 3981–3996 (2021)
8. Cao, L., Li, H., Xie, R., Zhu, J.: A text detection algorithm for image of student exercises based on CTPN and enhanced YOLOv3. IEEE Access **8**, 176924–176934 (2020)
9. Singh, B.B., Deepthi, V.H.: Survey on automatic vehicle number plate localization. Int. J. Comput. Appl. **67**(23) (2013)
10. Divvala, S.K., Hoiem, D., Hays, J.H., Efros, A.A., Hebert, M.: An empirical study of context in object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1271–1278. IEEE (2009)

11. Dewangan, D.K., Sahu, S.P.: Towards the design of vision-based intelligent vehicle system: methodologies and challenges. Evol. Intell. **16**(3), pp.759–800 (2023)
12. Wang, H., Yu, Y., Cai, Y., Chen, X., Chen, L., Liu, Q.: A comparative study of state-of- the-art deep learning algorithms for vehicle detection. IEEE Intell. Transp. Syst. Mag.st. Mag. **11**(2), 82–95 (2019)
13. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361. IEEE (2012)
14. Liu, W., Liao, S., Hu, W.: Towards accurate tiny vehicle detection in complex scenes. Neurocomputing **347**, 24–33 (2019)
15. Wu, H., Zhang, X., Story, B., Rajan, D.: Accurate vehicle detection using multi-camera data fusion and machine learning. In: ICASSP, pp. 3767–3771. IEEE (2019)
16. Yang, J., Li, Y., Zhang, Q., Ren, Y.: Surface vehicle detection and tracking with deep learning and appearance feature. In: 5th International Conference on Control, Automation and Robotics (ICCAR), pp. 276–280. IEEE (2019)
17. Chen, W., Qiao, Y., Li, Y.: Inception-SSD: an improved single shot detector for vehicle detection. J. Ambient. Intell. Humaniz. Comput. **13**(11), 5047–5053 (2022)
18. Li, Y., et al.: A deep learning - based hybrid framework for object detection and recognition in autonomous driving. IEEE Access **8**, 194228–194239 (2020)
19. Jana, A.P., Biswas, A.: YOLO based detection and classification of objects in video records. In: 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (RTEICT), pp. 2448–2452. IEEE (2018)
20. Zhao, J., Hao, S., Dai, C., Zhang, H., Zhao, L., Ji, Z.: Improved vision-based vehicle detection and classification by optimized YOLOv4. IEEE Access **10**, 8590–8603 (2022)
21. Farid, A., Hussain, F., Khan, K., Shahzad, M., Khan, U.: A fast and accurate real-time vehicle detection method using deep learning for unconstrained environments. Appl. Sci. **13**(5), 3059 (2023)
22. Kang, L., Lu, Z., Meng, L., Gao, Z.: YOLO-FA: type-1 fuzzy attention based YOLO detector for vehicle detection. Expert Syst. Appl. **237**, 121209 (2024)
23. YOLO-NAS Sets a New Standard for Object Detection. https://analyticsindiamag.com/yolo-nas-sets-a-new-standard-for-object-detection/. Accessed 30 Jan 2024
24. Dvornik, N., Mairal, J., Schmid, C.: Modeling visual context is key to augmenting object detection datasets. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 364–380 (2018)
25. https://roboflow.com/industries/transportation. Accessed 01 Feb 2024
26. Kaur, R., Singh, S.: A comprehensive review of object detection with deep learning. Dig. Sig. Process. **132**, 103812 (2023)
27. Indian Driving Dataset. https://idd.insaan.iiit.ac.in/dataset/details/. Accessed 04 /Feb 2024
28. Paranjape, B.A., Naik, A.A.: DATS_2022: a versatile Indian dataset for object detection in unstructured traffic conditions. Data Brief **43**, 108470 (2022)