International Journal of Electrical and Computer Engineering (IJECE)
Vol. 14, No. 4, August 2024, pp. 4287~4295
ISSN: 2088-8708, DOI: 10.11591/ijece.v14i4.pp4287-4295 ❒ 4287

An efficient object detection by autonomous vehicle using deep learning

Nitalaksheswara Rao Kolukula[1], Rajendra Prasad Kalapala[1], Sundara Siva Rao Ivaturi[1], Ravi Kumar Tammineni[2], Mahalakshmi Annavarapu[3], Uma Pyla[4]

[1]Department of Computer Science and Engineering, School of Technology, Gandhi Institute of Technology and Management (GITAM University), Visakhapatnam, India
[2]Department of Computer Science and Engineering, Aditya Institute of Technology and Management (AITAM), Tekkali, India
[3]Department of Computer Science and Business Systems, RVR and JC College of Engineering, Guntur, India
[4]Department of Computer Science and Engineering, Raghu Engineering College, Visakhapatnam, India

Article Info

Article history:

Received Nov 14, 2023
Revised Mar 19, 2024
Accepted Apr 2, 2024

Keywords:

Convolution neural network
Computer vision
Deep learning
Object detection
YOLO V3
YOLO V4
YOLO V5

ABSTRACT

The automation industries have been developing since the first demonstration in the period 1980 to 2000 it is mainly used on automated driving vehicle. Now a day's automotive companies, technology companies, government bodies, research institutions and academia, investors and venture capitalists are interested in autonomous vehicles. In this work, object detection on road is proposed, which uses deep learning (DL) algorithms. You only look once (YOLO V3, V4, V5). In this system object detection on the road data set is taken as input and the objects are mainly on-road vehicles, traffic signals, cars, trucks and buses. These inputs are given to the models to predict and detect the objects. The Performance of the proposed system is compared with performance of deep learning algorithms convolution neural network (CNN). The proposed system accuracy greater than 76.5% to 93.3%, mean average precision (Map) and frame per second (FPS) are 0.895 and 43.95%.

Corresponding Author:

Nitalaksheswara Rao Kolukula
Department of Computer Science and Engineering, School of Technology, Gandhi Institute of Technology and Management (GITAM University)
Visakhapatnam, Andhra Pradesh 530045, India
Email: kolukulanitla@gmail.com

1. INTRODUCTION

Traditional computer vision techniques have made significant contributions to object detection. However, their limitations in handling complex scenarios with diverse objects and varying environmental conditions have led to the emergence of deep learning as a powerful solution [1]. Deep learning, especially convolutional neural networks (CNNs), has demonstrated remarkable capabilities in learning hierarchical representations, enabling the development of sophisticated object detection models.

As one of the fundamental computer vision problems, object detection provides valuable information of images through videos or pictures. In these fields, the main detection of objects image classification, the autonomous driving meanwhile, integrated with neural networks and learning system. Object detection based on deep learning is divided into one-step detectors and two-step detectors [2]. Their accuracy and detection speed have been greatly improved but still cannot meet the requirements of real-time detection.

Therefore, the problem at hand is to design and evaluate an advanced object detection system that leverages the capabilities of you only look once (YOLO V3, V4, V5) models in comparison to traditional

CNN algorithms for identifying and localizing objects on roads. Addressing this problem is crucial for advancing the field of autonomous vehicles, as the accurate and rapid detection of objects on roads directly impacts the safety and reliability of autonomous systems.

The developed deep learning neural networks mainly impact on object detection techniques which can be considered on this system. Mainly deep learning consists of huge hidden layer data process like recognition, detection of images. It will give fast accurate result [3]. Autonomous vehicles (AVs) development has reached significant progress in the past decades. The main challenge in developing AVs technology is replicating a human driver into an autonomous intelligent system [4].

2. RELATED WORK

Study of deep learning architecture performance tuning is proposed here. Mainly comparison of given evaluation metrics and real-time smart surveillance system is seen by using YOLO architecture. This architecture has been implemented on a smart space dataset [5], [6]. While usually being superior in detection accuracy, the two-step detectors are less efficient than the one-step ones due to the need for object region inference [7]. This research proposed a complete self-delivery of the product vehicle that really can drive on the highway and report the vehicles for the authority's current geographical location in real-time through a Map CNN [8]. While waiting for the navigation value, Raspberry Pi transmits this same camera view finder. The problem has proposed this new method, a computational intelligence approach for automatic localization of road accidents as anomaly detection classification in to one category video [9].

3. METHOD

3.1. YOLO V3

A YOLO V3 is a type of deep convolution neural network (D-CNN) which has been broadly used in the area of object detection or reorganization [10]. The D-CNN for object detection and CNN are classifier-based system. It can process input images as structural arrays of data and identify patterns between them. The advantage of YOLO is that it is much faster than other networks and also maintains accuracy [11]. YOLO V3 using DarkNet-19, DarkNet-53 is Backbone it was extract feature. YOLO was detecting small objects also with accuracy. Figure 1 shows the YOLO V3 operational architecture [12].

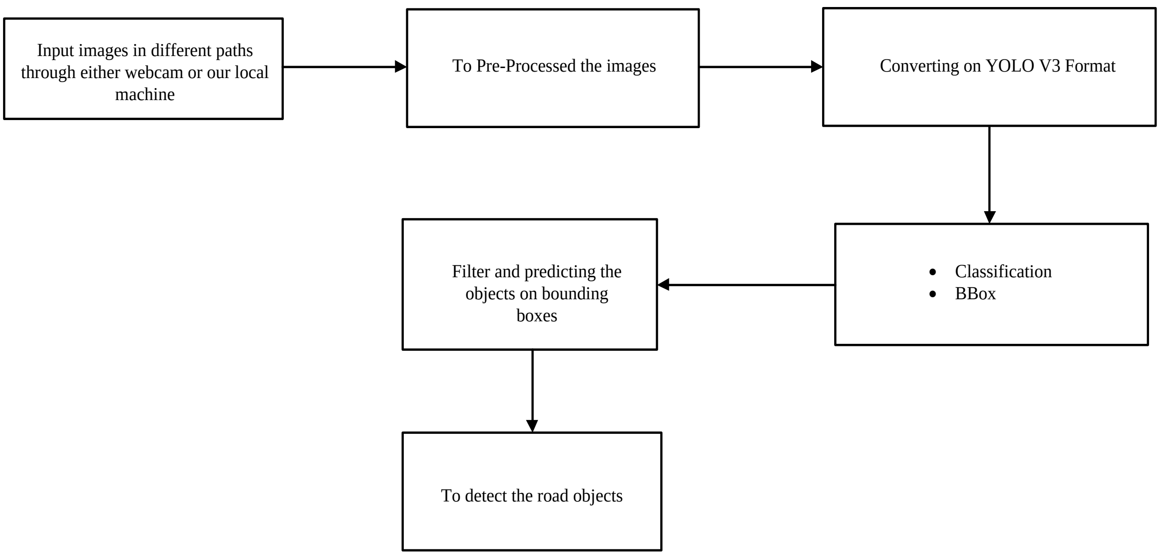

Figure 1. YOLO V3

3.2. YOLO V4

The YOLO V4 model uses neural networks to provide real-time object detection. This algorithm detects and recognizes various objects in a picture YOLO V4 is mainly developed under V3 [13]. YOLO detector can be trained and used on a convolution graphics processing unit (GPU) done as probabilities of the detected images. It has predicted each grid cell in the bounded boxes [14]. Only the bounded boxes for responsible for detecting the objects and this intersection over union is considered a good prediction. Compared to YOLO V3 it was to make a super-fast and one-stage object detector with high quality in terms of class labels, with those probabilities and accuracy [15]. YOLO V4 we are used CSP DarkNet-53 for extracting features, it has neck and head part Neck collects feature maps in different stages, head performs YOLO V3 the hole task performs on over here and it was achieved an accuracy of 5.8% higher than YOLO V3 [16]. YOLO V4 is 43.5% average precision running at 65FPS (frame per second) on it have very good results. Figure 2 shows the YOLO V4 operational architecture.

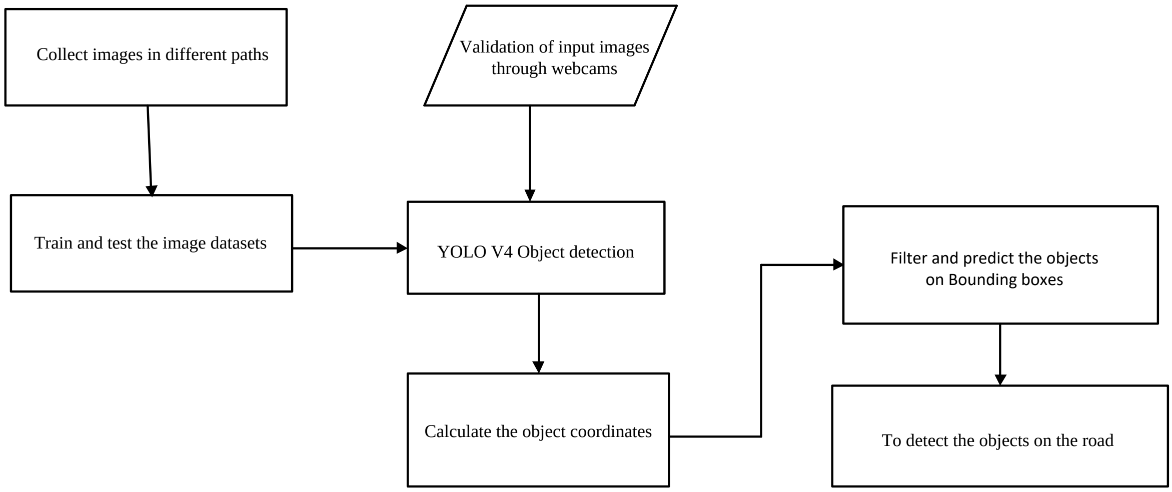

Figure 2. YOLO V4

3.3. YOLO V5

After the release of YOLO V4, within just two months of period, another version of YOLO has been released called YOLO V5. It is different from all other previous YOLO models. It is a family of single-stage deep learning based object detectors [17]. YOLO V4 has CSP Dark net, YOLO V5 has a CSP back bone and PA-NET Neck. The main improvements include Mosaic data augmentation and bounding box anchors and Back bone as the trained COCO model [18]. YOLO V5 is nearly smaller than YOLO V4. YOLO V5 is 27 MB and YOLO V4 has 244 MB; it can be used for real-time object detection based on the data streams. Compared to YOLO V3, V4, the YOLO V5 mean average precision of 92.34%, it claims efficient effectively to compare V3, V4. Figure 3 shows the YOLO V5 operational architecture [19].

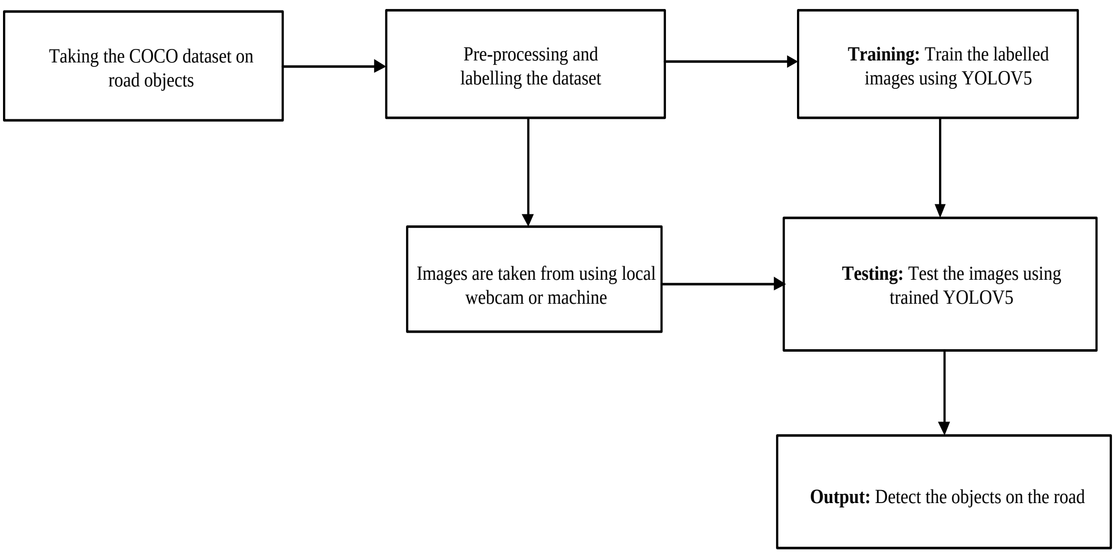

Figure 3. YOLO V5

4. ARCHITECTURE

Architecture describes the dataset used to analyze, classification, predict the performance of the system and the models. After acquiring a unified complete dataset, it is used to apply YOLO V3, YOLO V4 and YOLO V5 algorithms. The object coordinates are evaluated to predict the bounding boxes of the objects. Finally, it will detect the objects on the road and categorize them into objects into labels.

4.1. Dataset

The COCO Dataset visualized dataset mainly it is used in machine learning and deep learning train as well as test sets contain 200,000 new images and 80 moving and static object categories [20]. Size is 27 GB, classification of images, object strongly bounding box localization, semantic pixel-level fragmentation and large-scale object detection common objects in context (COCO). COCO dataset contains various images and its labels. All the labels are categorized into different classes.

4.2. Design overview

Design elaborates how the data is processed and working of YOLO versions V3, V4 and V5. YOLO versions describe deep convolution neural networks to process input images which consist of identification of patterns by considering individual features to have best performance. YOLO V3 has some performance drawbacks overcome by version V4, version V4 performance drawbacks overcome by development of version V5.

4.2.1. Pre-processing

Preprocessing step is necessary for predicting the images [21]. In this preprocessing step different pre-trained that are annotated with three values (x, y, c). The x and y values mark the coordinates, and c indicates the class labels.

4.2.2. Splitting of the data set

Dividing the dataset into training and testing sets is crucial for developing robust deep learning models. In this step, approximately 200,000 images out of the total 330,000 images are labeled across 80 object categories known as the "COCO classes" [22]. This partitioning allows the model to learn from a subset of data while evaluating its performance on unseen data, ensuring generalization. The training process utilizes algorithms to learn patterns and features from the labeled data, while the testing process evaluates the model's performance by assessing its predictions against ground truth labels. This iterative cycle of training and testing is fundamental in refining and improving deep learning models for various tasks.

4.2.3. Models of road object detection

In this step, deep learning algorithms are applied to the data to detect the moving objects and static objects, Real-time object detection. The algorithms used are the convolutional neural network, deep convolutional neural network, neural network, large-scale memory [23]. After applying the algorithms, the performance is evaluated using YOLO models (V3, V4, V5) at different accuracy. The current learning error rate represents 0.001000, V4 Intersection over Union (IoU) loss function greater than 0.5, V5 have 0.1 bounding box loss function [24].

4.2.4. Road object detection

In this step, object detection what the images we are inserted, what we can trained objects are inserted through either local machine or web cam will detecting objects also it detected the objects are real-time static and moving vehicles or objects with these class labels and probability of loss function [25]. In an autonomous vehicle, the best trained model will always help to identify and categorize the real time objects on the road. Misclassification of objects may cause confusion for the movement of autonomous vehicle. To avoid this, all deep learning models need to be trained with a maximum number of images with preprocessed and purified datasets.

5. RESULTS

5.1. Validation loss graph for YOLO V3

The loss values are identified at different iterations to measure the performance of the model. At first iteration, the experimental results show more loss value and by doing further iterations, the loss values are decreased, and the performance value of the model is increased. In the observation of experimental results, it is proven that the best training will result in the best performance of the model. In some cases, the autonomous vehicle is unable to identify the correct object for which it was trained. To overcome this drawback, the only solution is training of the model with best quality and quantity of the dataset. It will automatically resolve the problem and it will perform in a better way for existence of the noise, unnecessary features cause to degrade the performance of the model.

Figures 4 and 5 show high loss value at the first iteration later there were variations of loss value constantly at different at 0.001 learning rates (lrn) 0.01 and 0.001000. In the initial iteration, the model encountered a notably high loss value, indicating potential challenges in convergence. Subsequent iterations at a learning rate of 0.001 demonstrated fluctuations in the loss value, suggesting sensitivity to the chosen learning rate. Further experimentation with learning rates of 0.01 and 0.001000 revealed varying patterns in the loss values, emphasizing the critical role of hyper parameter tuning in optimizing model performance. Adjusting the learning rate appears to be crucial for stabilizing and enhancing the training process.

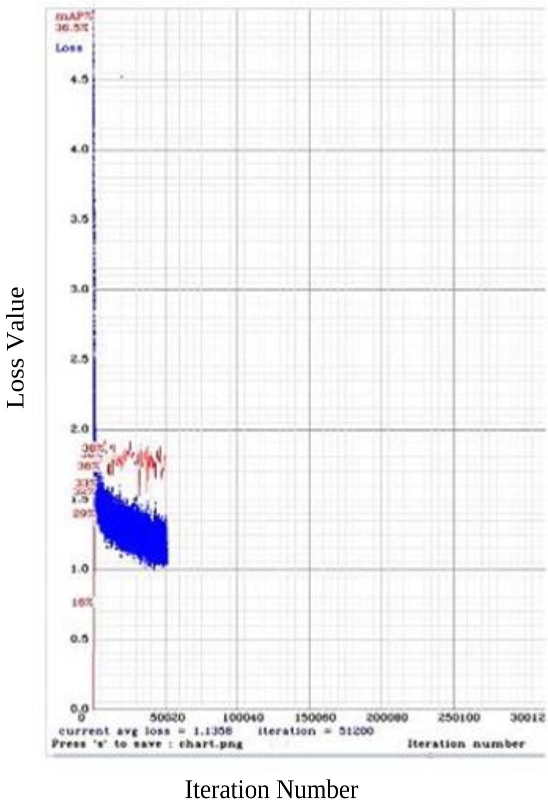

Figure 4. Validation loss graph at 0.001000 lrn

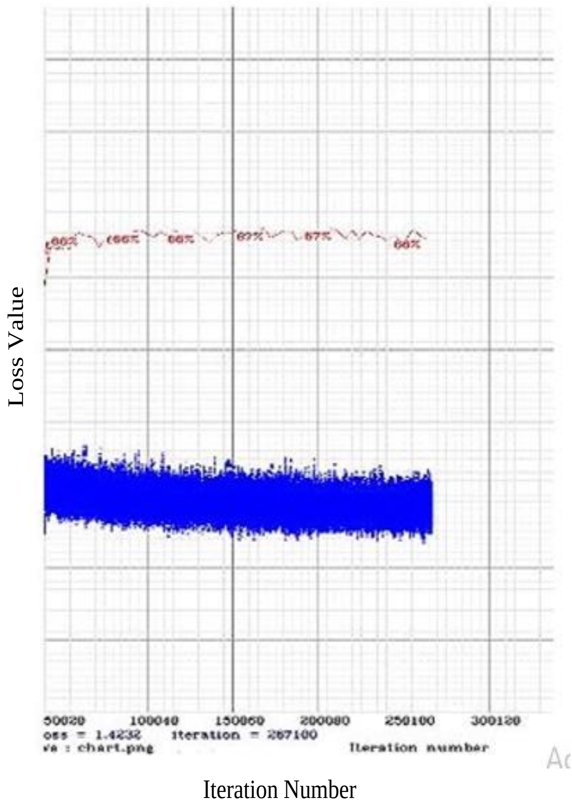

Figure 5. Validation loss graph at 0.00100 lrn

5.2. Validation loss graphs for YOLO V4

YOLO V4 shows the best performance when compared with YOLO V3. In the drawbacks identified in YOLO V3 are overcome by YOLO V4. Generally, the validation loss comes due to the misidentification of objects by the autonomous vehicle and by the non-detection of the objects. By using YOLO V4, the experimental results show the test and train values are equal and it is proven that the model is working with good performance. The learning rate identified will prove that YOLO V4 performs in a better way.

Figure 6 illustrates that when utilizing a learning rate (lrn) of 0.01, both the training and test sets display comparable loss values. This observation suggests that the model's performance remains consistent across different datasets, indicating that it is not overfitting to the training data. The alignment of loss values between the training and test sets at this learning rate signifies that the model generalizes well to unseen data, a crucial aspect in machine learning model evaluation. This balance in loss values across both sets implies that the model is effectively learning meaningful patterns from the training data without memorizing it, thus demonstrating its capability to perform reliably on new, unseen data.

5.3. Loss graphs for YOLO V5

The loss values are compared by using YOLO V5 in both cases of test set and train set. Test set loss value increases due to insufficient training process and the model unable to identify the objects on the road properly. But in the case of test process, loss values are constant due to the adapting of the necessary requirements by the model. Hence, the learning rate is measured, the YOLO V5 performance is less at test set and increases the performance at train set. Figure 7 shows that the loss values of test set were increasing at loss of train set was also constant after a 0.01 learning rate (lrn) for YOLO V3, YOLO V4 and YOLO V5. Table 1 shows comparison between YOLO V3, YOLO V4 and YOLO V5.

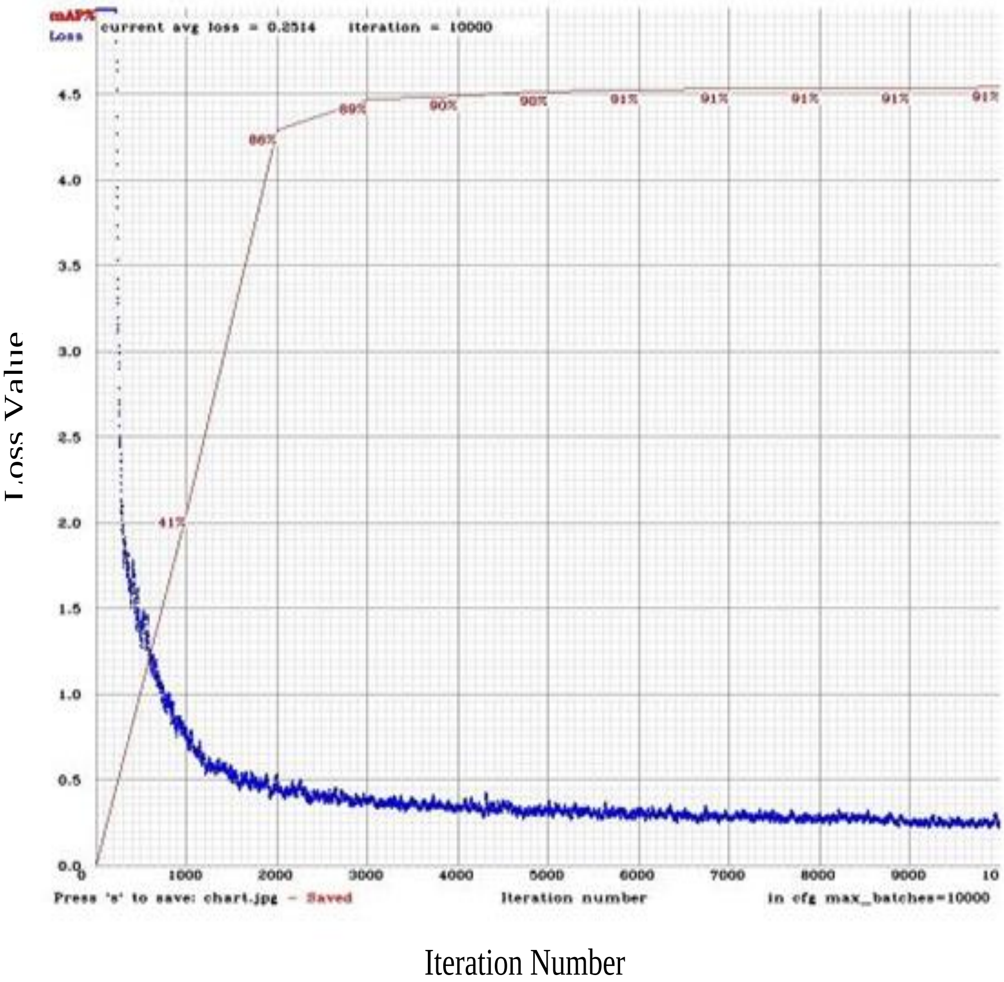

Figure 6. Validation loss graph at 0.05 lrn

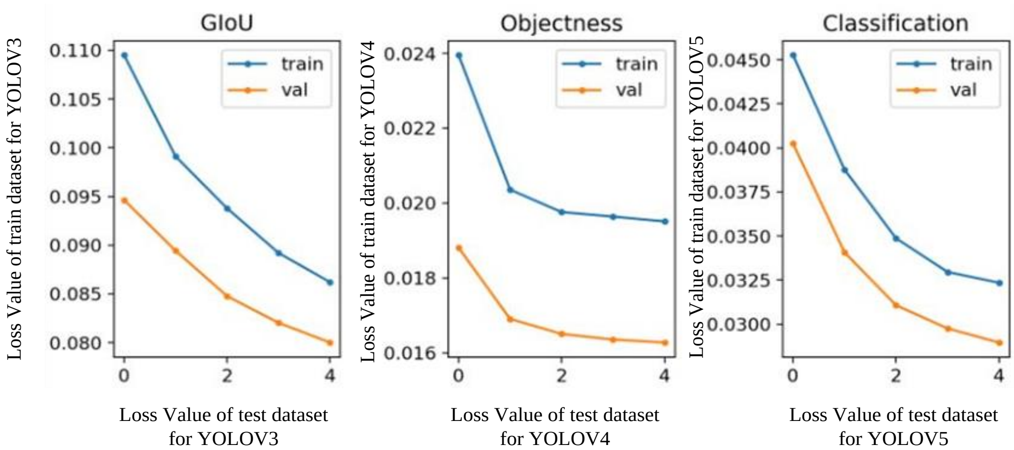

Figure 7. Validation loss graph at 0.01 lrn for YOLO V3, YOLO V4 and YOLO V5

Table 1. Comparison between YOLO V3, V4, V5

Description	YOLO V3	YOLO V4	YOLO V5
Neural network-type	Fully convolution	Fully convolution	Fully convolution
Back bone feature extractor	DarkNet-53	CSP DarkNet-53	CSP DarkNet-53
Loss function	Binary cross entropy	Binary cross entropy	Binary cross entropy and log its loss function
Neck	Feature pyramid network (FPN)	Spatial pyramid pooling (SPP) and Path aggregation network (PANet)	PANet
Head	YOLO layer	YOLO layer	YOLO layer

6. CONCLUSION

In this research, we embarked on a comprehensive exploration of object detection in road scenes using advanced deep learning algorithms, specifically YOLO versions (V3, V4, V5), and compared their performance against the well-established CNN. The objective was to enhance the capabilities of autonomous vehicles by accurate detection. The results obtained from the proposed system demonstrated remarkable accuracy, ranging from 76.5% to 93.3%. The MAP of 0.895 reflects the model's proficiency in precise object localization and classification.

REFERENCES

[1] H. Muslim *et al.*, “Cut-out scenario generation with reasonability foreseeable parameter range from real highway dataset for autonomous vehicle assessment,” *IEEE Access*, vol. 11, pp. 45349–45363, 2023, doi: 10.1109/ACCESS.2023.3268703.

[2] Z. Meng, S. Zhao, H. Chen, M. Hu, Y. Tang, and Y. Song, “The vehicle testing based on digital twins theory for autonomous vehicles,” *IEEE Journal of Radio Frequency Identification*, vol. 6, pp. 710–714, 2022, doi: 10.1109/JRFID.2022.3211565.

[3] P. Scheffe, T. M. Henneken, M. Kloock, and B. Alrifaee, “Sequential convex programming methods for real-time optimal trajectory planning in autonomous vehicle racing,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 661–672, Jan. 2023, doi: 10.1109/TIV.2022.3168130.

[4] Z. Yan, B. Song, Y. Zhang, K. Zhang, Z. Mao, and Y. Hu, “A rotation-free wireless power transfer system with stable output power and efficiency for autonomous underwater vehicles,” *IEEE Transactions on Power Electronics*, vol. 34, no. 5, pp. 4005–4008, May 2019, doi: 10.1109/TPEL.2018.2871316.

[5] J. Betz *et al.*, “Autonomous vehicles on the edge: a survey on autonomous vehicle racing,” *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 458–488, 2022, doi: 10.1109/OJITS.2022.3181510.

[6] W. Hu *et al.*, “Formulating vehicle aggressiveness towards social cognitive autonomous driving,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 3, pp. 2097–2109, Mar. 2023, doi: 10.1109/TIV.2023.3234253.

[7] C. Kim, Y. Yoon, S. Kim, M. J. Yoo, and K. Yi, “Trajectory planning and control of autonomous vehicles for static vehicle avoidance in dynamic traffic environments,” *IEEE Access*, vol. 11, pp. 5772–5788, 2023, doi: 10.1109/ACCESS.2023.3236816.

[8] C. Chatzikomis, A. Sorniotti, P. Gruber, M. Zanchetta, D. Willans, and B. Balcombe, “Comparison of path tracking and torque-vectoring controllers for autonomous electric vehicles,” *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 4, pp. 559–570, Dec. 2018, doi: 10.1109/TIV.2018.2874529.

[9] B. Li *et al.*, “Toward fair and thrilling autonomous racing: governance rules and performance metrics for the autonomous one,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 8, pp. 3974–3982, Aug. 2023, doi: 10.1109/TIV.2023.3298914.

[10] Y. Huang, S. Z. Yong, and Y. Chen, “Stability control of autonomous ground vehicles using control-dependent barrier functions,” *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 4, pp. 699–710, Dec. 2021, doi: 10.1109/TIV.2021.3058064.

[11] Z. Niu, X. S. Shen, Q. Zhang, and Y. Tang, “Space-air-ground integrated vehicular network for connected and automated vehicles: challenges and solutions,” *Intelligent and Converged Networks*, vol. 1, no. 2, pp. 142–169, Sep. 2020, doi: 10.23919/ICN.2020.0009.

[12] Y. Liu *et al.*, “Dynamic lane-changing trajectory planning for autonomous vehicles based on discrete global trajectory,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 8513–8527, Jul. 2022, doi: 10.1109/TITS.2021.3083541.

[13] D. Roper *et al.*, “Autosub long range 6000: a multiple-month endurance AUV for deep-ocean monitoring and survey,” *IEEE Journal of Oceanic Engineering*, vol. 46, no. 4, pp. 1179–1191, Oct. 2021, doi: 10.1109/JOE.2021.3058416.

[14] H. Marzbani, H. Khayyam, C. N. TO, D. V. Quoc, and R. N. Jazar, “Autonomous vehicles: autodriver algorithm and vehicle dynamics,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3201–3211, Apr. 2019, doi: 10.1109/TVT.2019.2895297.

[15] Y. Zhang, T. You, J. Chen, C. Du, Z. Ai, and X. Qu, “Safe and energy-saving vehicle-following driving decision-making framework of autonomous vehicles,” *IEEE Transactions on Industrial Electronics*, vol. 69, no. 12, pp. 13859–13871, Dec. 2022, doi: 10.1109/TIE.2021.3125562.

[16] A. Sharma and R. Gupta, “Efficient detection of small and complex objects for autonomous driving using deep learning,” in *2023 International Conference on Communication System, Computing and IT Applications (CSCITA)*, Mar. 2023, pp. 16–20, doi: 10.1109/CSCITA55725.2023.10104969.

[17] H. Yun and D. Park, “Mitigating overflow of object detection tasks based on masking semantic difference region of vision snapshot for high efficiency,” in *2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Feb. 2022, pp. 138–140, doi: 10.1109/ICAIIC54071.2022.9722651.

[18] M. Zhang, H. Li, G. Xia, W. Zhao, S. Ren, and C. Wang, “Research on the application of deep learning target detection of engineering vehicles in the patrol and inspection for military optical cable lines by UAV,” in *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, Dec. 2018, pp. 97–101, doi: 10.1109/ISCID.2018.00029.

[19] M. Back, J. Lee, K. Bae, S. S. Hwang, and I. Y. Chun, “Improved and efficient inter-vehicle distance estimation using road gradients of both ego and target vehicles,” in *2021 IEEE International Conference on Autonomous Systems (ICAS)*, Aug. 2021, pp. 1–5, doi: 10.1109/ICAS49788.2021.9551167.

[20] T. Swain, M. Rath, J. Mishra, S. Banerjee, and T. Samant, “Deep reinforcement learning based target detection for unmanned aerial vehicle,” in *2022 IEEE India Council International Subsections Conference (INDISCON)*, Jul. 2022, pp. 1–5, doi: 10.1109/INDISCON54605.2022.9862891.

[21] L. A. Silva, V. R. Q. Leithardt, V. F. L. Batista, G. Villarrubia González, and J. F. De Paz Santana, "Automated road damage detection using UAV images and deep learning techniques," *IEEE Access*, vol. 11, pp. 62918–62931, 2023, doi: 10.1109/ACCESS.2023.3287770.
[22] Y. Yang, X. Gao, Y. Wang, and S. Song, "VAMYOLOX: an accurate and efficient object detection algorithm based on visual attention mechanism for UAV optical sensors," *IEEE Sensors Journal*, vol. 23, no. 11, pp. 11139–11155, Jun. 2023, doi: 10.1109/JSEN.2022.3219199.
[23] Meng Joo Er, R. Venkatesan, and Ning Wang, "An online universal classifier for binary, multi-class and multi-label classification," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2016, pp. 003701–003706, doi: 10.1109/SMC.2016.7844809.
[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.
[25] C.-Y. Lee, P. Gallagher, and Z. Tu, "Generalizing pooling functions in CNNs: mixed, gated, and tree," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 863–875, Apr. 2018, doi: 10.1109/TPAMI.2017.2703082.

BIOGRAPHIES OF AUTHORS

Nitalaksheswara Rao Kolukula obtained his Ph.D. in computer science and systems engineering at Andhra University, Visakhapatnam, India. He received his M.Tech in computer science and engineering in2009. Now he is working as senior assistant professor in CSE Department at GITAM University, Visakhapatnam, Andhra Pradesh, India. His current research interest includes artificial intelligence, machine learning software engineering, data engineering and quality assurance. He can be contacted at email: kolukulanitla@gmail.com.

Rajendra Prasad Kalapala received his Ph.D. in computer science and systems engineering from Andhra University, Visakhapatnam in 2009 and received his master's degree M.Tech. He is working as assistant professor in Department of Computer Science and Engineering at GITAM University Visakhapatnam. His current research interests include machine learning and deep learning algorithms and computer vision applications. He can be contacted at email: kalapalarajendrap11@gmail.com.

Sundara Siva Rao Ivaturi obtained his Ph.D. in computer science and engineering from GITAM-A Deemed-to-be University, Visakhapatnam, India, 2017. He received his M. Tech in computer science and engineering from Andhra University in 2009. Currently he is working as associate professor in CSE department at GITAM University Visakhapatnam Andhra Pradesh India. He is a life member of ISTE, IEI, and CSTA. His current research interest includes machine learning, IoT, and artificial intelligence. He can be contacted at email: isro75@gmail.com.

Ravi Kumar Tammineni obtained his Ph.D. in computer science and engineering from GITAM-A Deemed-to-be University, Visakhapatnam, India, 2020. He received his M. Tech in computer science and engineering from JNT University, Hyderabad in 2008. He is currently working as associate professor at AITAM Tekkali Andhra Pradesh India. His current research interest includes machine learning, IoT, software engineering and artificial intelligence. He can be contacted at email: ravi.4u.kumar@gmail.com.

Mahalakshmi Annavarapu completed her M.Tech (computer science and engineering) from Avanthi Institute of Engineering and Technology affiliated to JNTU, Hyderabad. Currently working as an assistant professor at RVR&JC College of Engineering (Autonomous) in Department of Computer Science and Business System, Guntur. She had two patents. She attended and presented papers in different conferences, workshops and symposiums. She published various papers in different international and national journals. She can be contacted at email: mahalakshmi.valluri09@gmail.com.

Uma Pyla completed M.Tech in computer science and Technology at Andhra university Visakhapatnam India and completed B. Tech in computer science and engineering. Now working as assistant professor in Department of computer science and Engineering at Raghu College of Engineering, Vizianagaram, Andhra Pradesh. India. Her current research interests include machine learning, block chain technology and federated learning. She can be contacted at email: pylauma2017@gmail.com.