SEPE-YOLO: A SQUEEZE-AND-EXCITATION AIDED AND PARZEN ESTIMATOR OPTIMIZED YOLO MODEL FOR VEHICLE DETECTION

SOURAJIT MAITY^{1,2}, SAYANI DHAR^{1}, ASFAK ALI^{3}, DMITRII KAPLUN^{4,5}, SERGEI ANTONOV^{6}, RAM SARKAR^{1}

^{1}Department of CSE, Jadavpur University, Kolkata, India
^{2}Department of CSE, Sister Nivedita University, Kolkata, India
^{3}Department of ETCE, Jadavpur University, Kolkata, India
^{4}Artificial Intelligence Research Institute, China University of Mining and Technology, Xuzhou, China
^{5}Intelligent Devices Institute, St. Petersburg Electrotechnical University ”LETI”, St. Petersburg, Russian Federation
^{6}Dept. of Automation and Control Process, St. Petersburg Electrotechnical University ”LETI”, St. Petersburg, Russian Federation
E-MAIL: sourajit.cse.ju@gmail.com, sayani.1910@gmail.com
asfakali.etce@gmail.com, dikaplun@etu.ru, saantonov@etu.ru, ramjucse@gmail.com

Abstract:

Vehicle detection systems (VDS) are growing in popularity as intelligent transportation systems expand rapidly. Autonomous driving, urban planning, and traffic regulation all depend on VDS. This paper develops a dependable VDS model, called SEPE-YOLO (Squeeze-and-Excitation aided and Parzen Estimator optimized YOLO), which can detect various vehicle types from images taken in different lighting and weather conditions, including sunny, rainy, daylight, and midnight. In order to improve the model's performance, Squeeze-and-Excitation (SE) blocks are incorporated into the YOLOv10 architecture. We have used the Tree-structured Parzen Estimator to fine-tune the hyperparameters of the SEPE-YOLO model. The proposed model has been evaluated on two public VDS datasets: JUVDsiv1 and IRUVD. We have obtained mAP scores of 79.9% and 94.8% on JUVDsiv1 and IRUVD dataets, respectively.

Keywords:

Vehicle detection systems, Traffic management, YOLOv10, Squeeze-and-Excitation, JUVDsiv1, IRUVD

1. Introduction

Deploying real-time vehicle monitoring and detection systems in densely populated environments presents significant challenges, particularly in distinguishing vehicles in close proximity. Accurately detecting and tracking vehicles under such conditions requires advanced computational models capable of processing complex visual data. Recent advancements in deep learning, particularly convolutional neural networks (CNNs), have demonstrated strong performance in a variety of computer vision tasks, including object and vehicle detection [1]. Object detection frameworks are typically categorized into two paradigms: two-stage and single-stage detectors. Two-stage methods, such as Region-based Convolutional Neural Networks (R-CNN) [2], and Fast R-CNN [3] begin by generating region proposals, which are subsequently refined and classified. While these models are known for their high accuracy, they often involve greater computational complexity. In contrast, single-stage detectors such as the Single Shot MultiBox Detector (SSD) [4] and various versions of the You Only Look Once (YOLO) family [5], bypass the region proposal step and directly regress bounding boxes and class probabilities, thereby offering faster inference with competitive accuracy. Recent research has increasingly focused on improving single-stage detectors for real-time applications. Among these, the YOLO family has undergone rapid evolution, culminating in the release of YOLOv10 [6], which has demonstrated substantial promise in real-time object detection scenarios. Despite its early stage of adoption, YOLOv10 offers an effective balance between detection speed and accuracy, making it a suitable candidate for deployment in dynamic traffic environments.
Contributions: Motivated by these developments, we propose a new model, called SEPE-YOLO (Squeeze-and-Excitation aided and Parzen Estimator optimized YOLO) for vehicle detection from still images. Here, YOLOv10's feature extraction capability is enhanced by incorporating the Squeeze-and-Excitation (SE) blocks. Furthermore, a Tree-structured Parzen

Estimator (TPE) is employed for hyperparameter optimization, enabling systematic exploration of the parameter space to identify the optimal configuration that maximizes validation accuracy. To evaluate the effectiveness of the proposed model, we conduct experiments on two publicly available datasets: JUVDsiv1 [1], and IRUVD [7]. The results demonstrate the robustness and accuracy of the SEPE-YOLO model in complex, real-world vehicular environments.

2. Literature Survey

In recent years, several advancements have been made in object detection algorithms specifically tailored for vehicle detection under various constraints and environments. In 2021, S and Rani [8] introduced LittleYOLO-SPP, a lightweight vehicle detection model based on YOLOv3-tiny, enhanced with a Spatial Pyramid Pooling (SPP) layer. Their approach also incorporated Mean Squared Error (MSE) and Generalized Intersection over Union (GIoU) loss functions to improve bounding box regression accuracy.

Du et al. [9] proposed a method leveraging YOLOv4 for detecting heavily occluded vehicles in complex infrared aerial imagery. Their contribution includes a secondary transfer learning step to fine-tune the detection model, resulting in improved detection performance in challenging conditions. Shi et al. [10] presented a refined version of YOLOv3 by integrating GIoU and focal loss functions, along with label smoothing and a cosine decay learning rate scheduler. These enhancements collectively contributed to higher average detection accuracy. Tajar et al. [11] focused on resource-constrained environments by employing the Tiny-YOLOv3 architecture for real-time vehicle detection and tracking. They reduced the network complexity by decreasing the number of filters and layers, and omitted batch normalization in scenarios with relatively simple backgrounds to further optimize inference speed. Amrouche et al. [12] introduced a lightweight modification of YOLOv4 based on the YOLO-tiny architecture. Their design utilizes CSPDarknet53-tiny with three convolutional layers and three Cross Stage Partial (CSP) blocks, while replacing the Mish activation function with LeakyReLU to enhance computational efficiency.

More recently, Singh et al. [13] proposed a YOLOv7-based solution for aerial-view vehicle detection and tracking, tailored for automated traffic data collection systems. Their framework integrates vehicle detection, classification, and tracking from UAV imagery to support intelligent transportation monitoring. Yang [14] advanced YOLOv5 for small-object vehicle detection by incorporating a dedicated small-target detection layer. Additionally, their model employs a Bi-directional Feature Pyramid Network (BiFPN) for feature fusion and uses adaptive weighted fusion to effectively integrate multiscale information, reducing both false positives and missed detections.

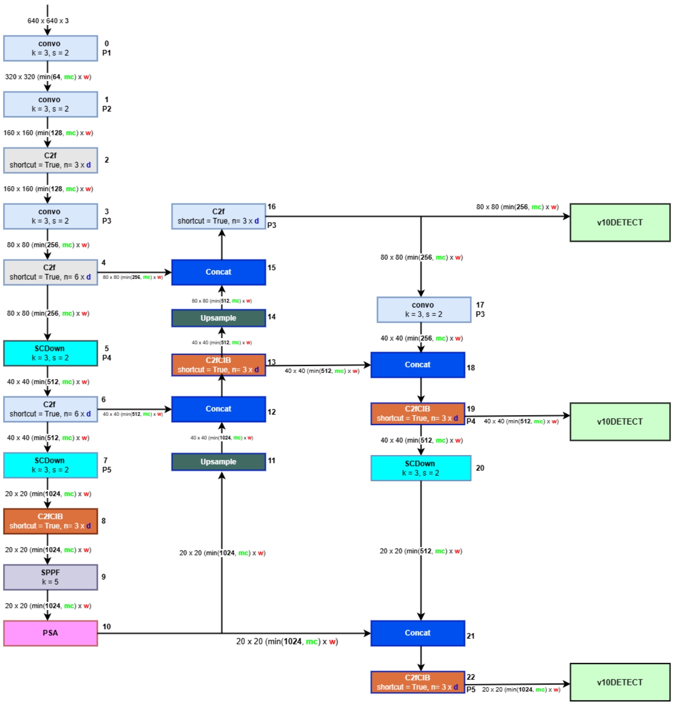
640 x 640 x 3
convo
k = 3, s = 2
0
P1
320 x 320 (min(64, mc) x w)
convo
k = 3, s = 2
1
P2
160 x 160 (min(128, mc) x w)
C2f
shortcut = True, n= 3 x d
2
160 x 160 (min(128, mc) x w)
convo
k = 3, s = 2
3
P3
80 x 80 (min(256, mc) x w)
C2f
shortcut = True, n= 6 x d
4
80 x 80 (min(256, mc) x w)
80 x 80 (min(256, mc) x w)
SCDown
k = 3, s = 2
5
P4
40 x 40 (min(512, mc) x w)
C2f
shortcut = True, n= 6 x d
6
40 x 40 (min(512, mc) x w)
40 x 40 (min(512, mc) x w)
SCDown
k = 3, s = 2
7
P5
20 x 20 (min(1024, mc) x w)
C2fCIB
shortcut = True, n= 3 x d
8
20 x 20 (min(1024, mc) x w)
SPPF
k = 5
9
20 x 20 (min(1024, mc) x w)
PSA
10
20 x 20 (min(1024, mc) x w)
Upsample
11
40 x 40 (min(1024, mc) x w)
Concat
12
C2fCIB
shortcut = True, n= 3 x d
13
40 x 40 (min(512, mc) x w)
Upsample
14
80 x 80 (min(512, mc) x w)
Concat
15
C2f
shortcut = True, n= 3 x d
16
P3
80 x 80 (min(256, mc) x w)
v10DETECT
80 x 80 (min(256, mc) x w)
convo
k = 3, s = 2
17
P3
40 x 40 (min(256, mc) x w)
40 x 40 (min(512, mc) x w)
Concat
18
C2fCIB
shortcut = True, n= 3 x d
19
P4
40 x 40 (min(512, mc) x w)
v10DETECT
40 x 40 (min(512, mc) x w)
SCDown
k = 3, s = 2
20
20 x 20 (min(512, mc) x w)
20 x 20 (min(1024, mc) x w)
Concat
21
C2fCIB
shortcut = True, n= 3 x d
22
P5
20 x 20 (min(1024, mc) x w)
v10DETECT

FIGURE 1. Architectural overview of YOLOv10.

3. METHODS AND MATERIALS

3.1. Basics of YOLOv10

YOLOv10, developed by researchers at Tsinghua University [6], is a real-time object detection framework that addresses limitations in post-processing and model design present in previous YOLO versions. Notably, it eliminates the need for non-maximum suppression (NMS) by integrating a novel end-to-end detection head. YOLOv10 achieves state-of-the-art accuracy-latency trade-offs across multiple model scales while significantly reducing computational overhead. Its architecture retains the strengths of prior YOLO models, incorporating an enhanced Cross Stage Partial Network (CSPNet) in the backbone to improve gradient flow and reduce redundancy. The neck module fuses multiscale features and transmits them to the detection head. An architectural overview is illustrated in Figure 1.

3.2. Squeeze-and-Excitation Blocks

Hu et al. [15] introduced the Squeeze-and-Excitation Network (SENet), a channel attention mechanism designed to model interdependencies between feature channels. SENet enhances feature representations by adaptively recalibrating channel-wise responses, emphasizing informative features while suppressing less relevant ones. The mechanism consists of two key stages: squeeze and excitation. In the squeeze phase, global average pooling (GAP) is applied to aggregate global spatial information into a compact channel descriptor. The excitation phase then uses two fully connected layers to learn non-linear interactions between channels and generate attention weights. These weights are used to rescale the original feature maps, selectively enhancing or diminishing channel responses. An architectural illustration of the SE block is shown in Figure 2.

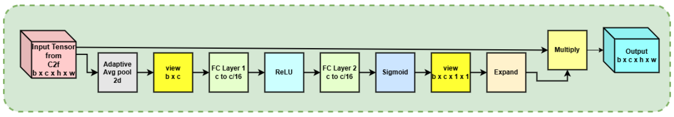

FIGURE 2. Architectural overview of the Squeeze-and-Excitation block

3.3. Hyperparameter Tuning using Parzen Optimizer

The Tree-structured Parzen Estimator (TPE), proposed by Akiba et al. [16], formulates hyperparameter optimization as a sequential model-based optimization problem, where the objective is to minimize or maximize a black-box function that evaluates model performance based on a given set of hyperparameters. TPE distinguishes itself by modeling the objective function using non-parametric density estimators, which separately estimate the distributions of favorable and unfavorable hyperparameter configurations. New samples are drawn by maximizing the expected improvement, effectively biasing the search toward regions of the space that are more likely to yield better performance. A key strength of TPE lies in its define-by-run paradigm, wherein the hyperparameter search space is constructed dynamically during the execution of the objective function. This allows for flexible and conditional configuration of parameters, such as varying the number of network layers or hidden units depending on other hyperparameters. In this formulation, each optimization process is referred to as a study, consisting of multiple trials, with each trial representing a single evaluation of the objective function. The search space is progressively expanded through interaction between the objective function and a trial object, enabling dynamic and expressive hyperparameter definitions beyond the limitations of static search strategies.

3.4. Modified YOLO

To enhance feature refinement and improve the representational capacity of the network, we incorporate a Squeeze-and-Excitation (SE) block into the cf2 feature map. As an intermediate-level representation, cf2 is critical for capturing both semantic and spatial information necessary for robust multi-scale object detection. The SE block performs global average pooling to generate compact channel-wise descriptors, followed by a lightweight gating mechanism that adaptively recalibrates the channel responses. This operation selectively emphasizes informative features while suppressing less relevant ones, thereby strengthening the discriminative power of cf2. In addition, we employ Parzen Estimator-based optimization to automatically tune key training hyperparameters such as learning rate, batch size, and weight decay. This data-driven approach to hyperparameter search not only accelerates convergence but also improves generalization by identifying well-balanced configurations that might be suboptimal under manual tuning. Together, the integration of SE-based refinement and automated hyperparameter optimization contributes to more accurate and stable object detection performance across varied scenarios. Key parameters like learning rate, batch size, optimizer type SGD, weight decay, data augmentation settings (mixup, mosaic), anchor sizes, and intersection-over-union (IoU) thresholds are considered ib the optimization process. The optimizer effectively explores the search space over 50 trials using the TPE sampler, to maximize the mAP@0.5:0.95 on the validation dataset. With the addition of the SE attention module and hyperparameter tuning based on the TPE, the modified YOLOv10 framework outperforms the vanilla architecture in terms of performance. Better feature representation and automated model tuning are made possible by these improvements, which increases the detector's resilience in a variety of difficult image types. The architectural overview of the modified C2f block is shown in Figure 2. To the output of every C2f convolutional block in the YOLO architecture, we have added an SE block, which improves the attention mechanism of the model. The architectural overview of the SEPE-YOLO model is shown in Figure 4.

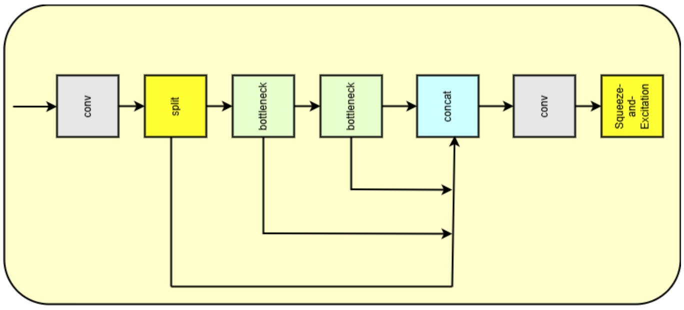

FIGURE 3. Architectural overview of the modified C2f block

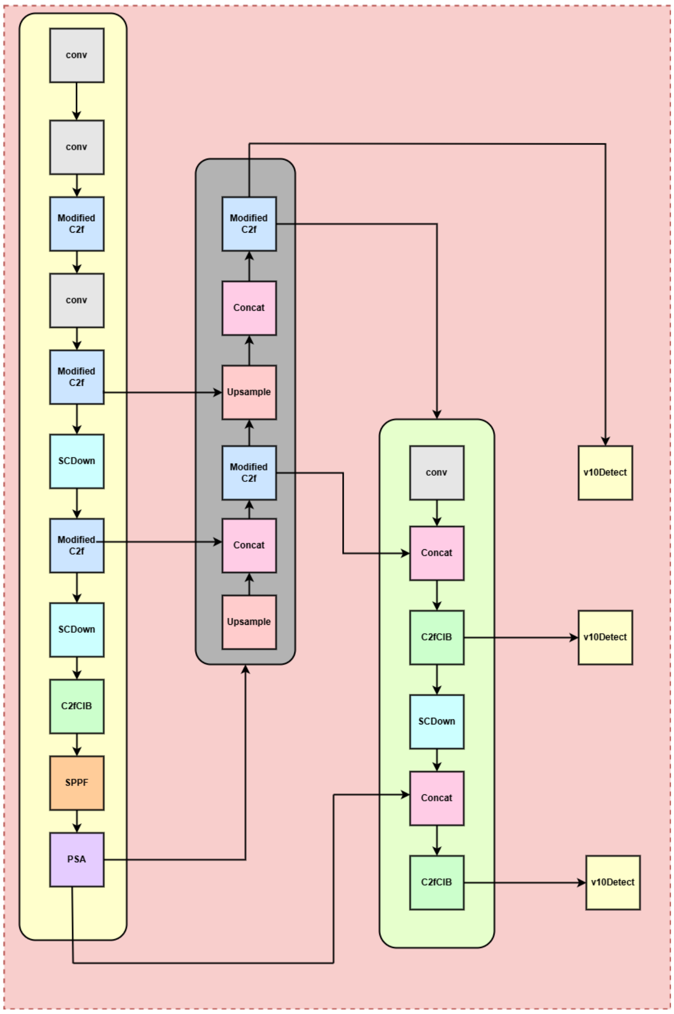

FIGURE 4. Architectural overview of the SEPE-YOLO model proposed here

4. Results and Discussion

4.1. Datasets

For training and testing of the proposed model, we have used two benchmarked datasets, namely JUVDsiv1[1] and IRUVD[2]. The JUVDsiv1 dataset depicts a typical Indian road scenario developed by Bhattacharyya et al. [1]. The images were taken under various circumstances and at various times of the day. The dataset has nine distinct vehicle classes: cars, buses, motorcycles, cycles, trucks, auto-rickshaws, rickshaws, vans, and mini trucks. The training set consists of 651 images in overcast weather and 872 images in sunny weather, and 565 images taken under clear skies at night. The IRUVD dataset, created by Ali et al. [7], contains 13 Indian vehicle classes, including truck, motor-rickshaw, tempo, automobile, taxi, toto, cycle-rickshaw, bus, auto-rickshaw, jeep, and van along with pedestrians. There are 4000 images and 14,343 bounding box annotations in this collection.

4.2. Experimental Setup and Hyperparameters

The proposed model was implemented on an Intel Core i5 machine equipped with an NVIDIA GEFORCE RTX graphics card, 16GB of RAM, and 8GB of GPU. The proposed model was trained with a learning rate of 0.001 for 200 epochs.

4.3 Evaluation Metrics

We have used some standard evaluation metrics: F1 score [17], Precision (Pre) [17], Recall (Rec) [17], and Mean Average Precision (mAP) [18]. By comparing actual and predicted outcomes, these metrics are calculated by analyzing the relationship between the True Positive (instances that are positive by both ground truth and the prediction), True Negative (instances that are negative by both ground truth and the prediction), False Positive (negative instances that are misclassified as positive), and False Negative (positive instances that are misclassified as negative) values.

$$Precision(\%) = \frac{TP}{TP + FP} \times 100 \tag{1}$$

$$Recall(\%) = \frac{TP}{TP + FN} \times 100 \tag{2}$$

[1] https://github.com/JUVDsi/JUVD-Still-Image-database.git

[2] https://github.com/IRUVD/IRUVD.git

$$F1\text{-}score(\%) = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \times 100 \quad (3)$$

Here, *TP*, *TN*, *FP* and *FN* signify the true positives, true negatives, false positives and false negatives, respectively.

4.4. Results

The outcomes of the SEPE-OLO model with SE attention modules and TPE-based hyperparameter optimization on two vehicle detection datasets are shown in Figure 6. We obtained an mAP score of 79.9% at mAP50 and 57.9% at mAP50-95 using the JUVDsiv1 dataset. We obtained a 94.8% mAP score at mAP50 and an 85.2% mAP score at mAP50-95 using the IRUVD dataset. The precision, recall, and F1 confidence curves are shown inFigure 7, Figure 8 after our model has been run on the JUVDsiv1 and IRUVD datasets.

Output images obtained on JUVDsiv1 and IRUVD datasets by applying the SEPE-YOLO model are shown in Figure 5. Table 1 presents comparative results of the base YOLO model with proposed results performed on JUVDsiv1 and IRUVD datasets. From Table 1, we find that our proposed model has achieved better mAP score, precision, and recall scores than base models.

TABLE 1. Performance comparison of our proposed model with base model on two datasets. All scores are in %.

Dataset	Method	mAP50	Pre	Rec	mAP95
JUVDsiv1	YOLOv10	75.3	55.2	79.4	71.3
	Proposed	**79.9**	**76.7**	72.1	57.9
IRUVD	YOLOv10	94.6	85.5	93.4	92.4
	Proposed	**94.8**	**93.5**	88.8	85.2

4.5. Comparative Study

The comparative analysis of the JUVDsiv1 and IRUVD datasets is summarized in Table. 2. On the test set, the proposed model achieves the mAP score of 79.9%, whereas Bhattacharya et al. [1] achieve a 74.5% mAP score using a weighted box fusion approach on the JUVDsiv1 dataset. However, employing a YOLOv8 as the baseline model, [5] achieves a lower mAP score of 94.6% compared to our approach. Our method achieves an impressive test mAP score of 94.8% on the IRUVD dataset, where Ali et al. [7] achieve a lower mAP score than our proposed model.

TABLE 2. Comparative results on both datasets

Work Ref.	**Dataset: JUVDsiv1**	
	Method	***mAP(%)***
Bhattacharya et al. [1], 2021	Weighted Box Fusion	74.5
Jocher et al. [19], 2020	YOLOv5	78.6
Lou et al. [20], 2023	YOLOv8	76.8
Proposed, 2025	SEPE-YOLO	**79.9**
	Dataset: IRUVD	
	Method	***mAP(%)***
Ali et al.[7], 2023	YOLOv5+scaled YOLOv4	94.7
Maity et al.[5], 2023	YOLOv8	94.6
Jocher et al.[19], 2020	YOLOv5	92.2
Ali et al.[7], 2023	YOLOv4+scaled YOLOv4	94.3
Proposed, 2025	SEPE-YOLO	**94.8**

4.6. Conclusion

Vehicle detection systems are essential for enhancing transportation safety and efficiency by enabling real-time vehicle detection, reducing traffic congestion, and managing autonomous vehicles. In this study, we proposed SEPE-YOLO, a vehicle detection model that integrates SE blocks in the YOLOv10 model, aided with hyperparameter tuning by the TPE optimizer. By incorporating SE blocks into the YOLOv10 architecture, we enhanced the model's attention mechanism. We evaluated the SEPE-YOLO on two vehicle detection datasets, namely JUVDsiv1 and IRUVD, demonstrating its effectiveness across diverse weather conditions. While the performance is promising, further exploration of advanced attention modules could lead to even better results. To further validate the robustness and generalizability of our approach, future work will involve testing SEPE-YOLO on additional datasets, such as surveillance and other object detection benchmarks. These evaluations will help refine the model and ensure its applicability in a broad range of real-world scenarios.

Acknowledgements

We are grateful to the CMATER Research Lab, Department of Computer Science and Engineering, Jadavpur University, Kolkata, India for providing the infrastructural assistance.

References

[1] Avirup Bhattacharyya, Avigyan Bhattacharya, Sourajit Maity, Pawan Kumar Singh, and Ram Sarkar. Juvdsi v1:

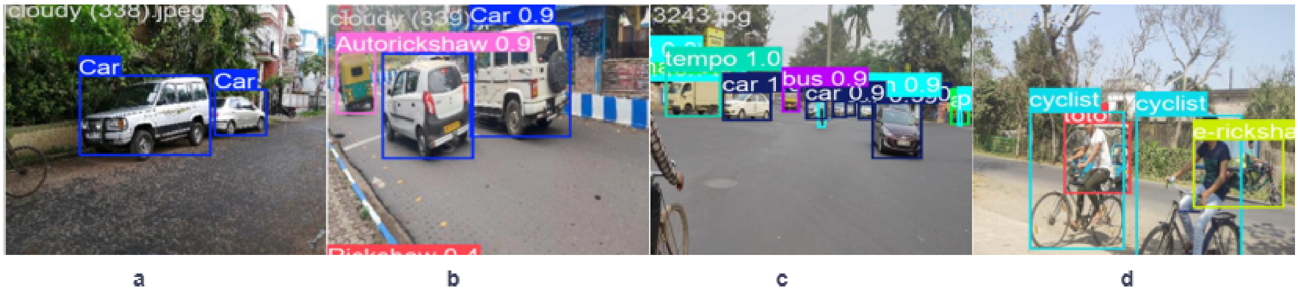

FIGURE 5. Some outputs of the proposed model on (a)-(b) IRUVD dataset, and (c)-(d) JUVDsiv1 dataset.

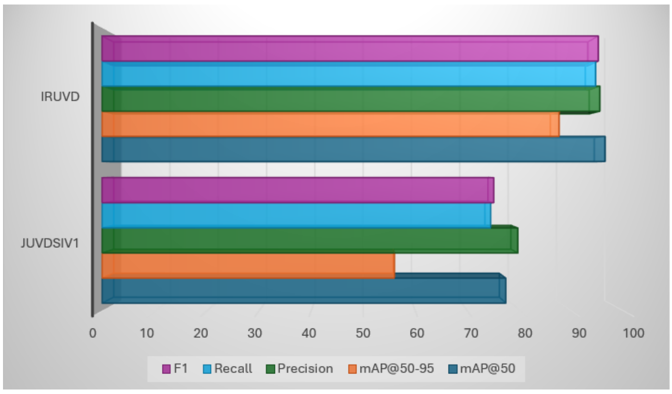

FIGURE 6. Results of the proposed model on JUVDsiv1 and IRUVD datasets.

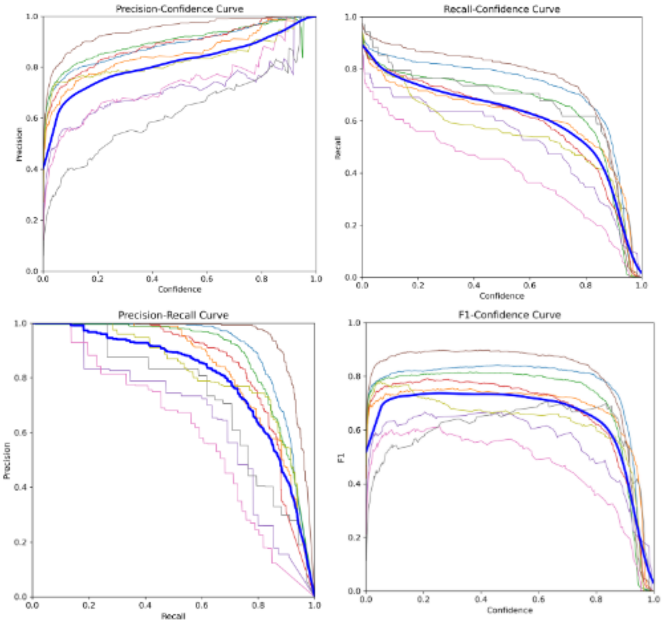

FIGURE 7. Precision, Recall, Precision-Recall and F1 score confidence curves on the JUVDsiv1 dataset.

developing and benchmarking a new still image database in indian scenario for automatic vehicle detection. *Multimedia Tools and Applications*, pages 1–33, 2023.

[2] Puja Bharati and Ankita Pramanik. Deep learning techniques—r-cnn to mask r-cnn: a survey. *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*, pages 657–668, 2020.

[3] Ross Girshick. Fast r-cnn. *arXiv preprint arXiv:1504.08083*, 2015.

[4] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.

[5] Sourajit Maity, Arpan Chakraborty, Pawan Kumar Singh, and Ram Sarkar. Performance comparison of various yolo models for vehicle detection: An experimental study. In *International Conference on Data Analytics & Management*, pages 677–684. Springer, 2023.

[6] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yolov10: Real-time end-to-end object detection. *arXiv preprint arXiv:2405.14458*, 2024.

[7] Asfak Ali, Ram Sarkar, and Debesh Kumar Das. Iruvd: a new still-image based dataset for automatic vehicle detection. *Multimedia Tools and Applications*, 83(3):6755–6781, 2024.

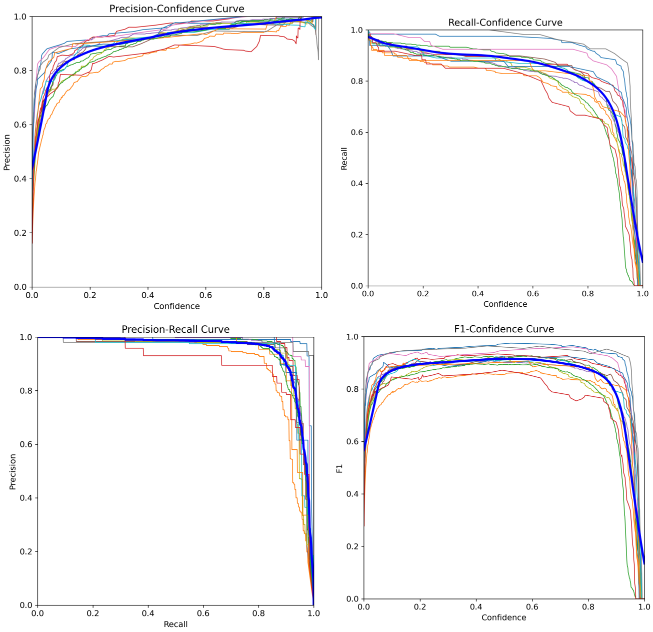

FIGURE 8. Precision, Recall, Precision-Recall, and F1 score confidence curves on the IRUVD dataset.

[8] Esther Rani et al. Littleyolo-spp: A delicate real-time vehicle detection algorithm. *Optik*, 225:165818, 2021.

[9] Shuangjiang Du, Pin Zhang, Baofu Zhang, and Honghui Xu. Weak and occluded vehicle detection in complex infrared environment based on improved yolov4. *IEEE Access*, 9:25671–25680, 2021.

[10] Junjie Shi, Xiujie Qu, Yukun Feng, and Chuang Wang. A vehicle detection method based on improved yolov3. In *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, volume 5, pages 2201–2207. IEEE, 2021.

[11] Alireza Taheri Tajar, Abbas Ramazani, and Muharram Mansoorizadeh. A lightweight tiny-yolov3 vehicle detection approach. *Journal of Real-Time Image Processing*, 18(6):2389–2401, 2021.

[12] Aissa Amrouche, Youssouf Bentrcia, Ahcène Abed, and Nabil Hezil. Vehicle detection and tracking in real-time using yolov4-tiny. In *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*, pages 1–5. IEEE, 2022.

[13] Vishakha Singh, Phisan Kaewprapha, and Choompol Boonmee. Ad-hoc aerial-view vehicle detection and tracking for real-time traffic monitoring using yolov7. In *2023 International Electrical Engineering Congress (iEECON)*, pages 327–331. IEEE, 2023.

[14] Wuping Yang. Multi-type vehicle detection algorithm based on improved yolov5. In *2023 3rd International Conference on Frontiers of Electronics, Information and Computation Technologies (ICFEICT)*, pages 599–603. IEEE, 2023.

[15] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[16] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.

[17] Luis Torgo and Rita Ribeiro. Precision and recall for regression. In *Discovery Science: 12th International Conference, DS 2009, Porto, Portugal, October 3-5, 2009 12*, pages 332–346. Springer, 2009.

[18] Kehuang Li, Zhen Huang, You-Chi Cheng, and Chin-Hui Lee. A maximal figure-of-merit learning approach to maximizing mean average precision with deep neural network based classifiers. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4503–4507. IEEE, 2014.

[19] Glenn Jocher, Alex Stoken, Jirka Borovec, Liu Changyu, Adam Hogan, Laurentiu Diaconu, Jake Poznanski, Lijun Yu, Prashant Rai, Russ Ferriday, et al. ultralytics/yolov5: v3. 0. *Zenodo*, 2020.

[20] Haitong Lou, Xuehu Duan, Junmei Guo, Haiying Liu, Jason Gu, Lingyun Bi, and Haonan Chen. Dc-yolov8: Small-size object detection algorithm based on camera sensor. *Electronics*, 12(10):2323, 2023.