Journal of Industrial Information Technology and Application *ISSN(Online): 2586-0852*
Vol.9 No. 3

Vehicle Recognition and Speed Monitoring System using YOLOv9

Sang Suh, and Bilal Mushtaq)*

Department of Computer Science East Texas A&M University, U.S.A.

Abstract. Overspeeding is a major cause of fatal accidents and car crashes around the world. Drivers that aren't timely held accountable for their recklessness become a threat for themselves and for those sharing the road with them. Numerous researchers have presented machine learning and deep learning-based object detection and recognition techniques that aim to accurately detect vehicles but have come across hurdles such as varying illumination and weather conditions, overlapping vehicle images in complex traffic situations, the need for powerful GPU based computers at surveillance, and slow detection speeds. The proposed work VRSMY9 (Vehicle Recognition and Speed Monitoring System using YOLOv9) powered by YOLOv9 and EasyOCR focuses on enhancing the overall detection speed in real-time without compromising too much on the detection accuracy whilst being lightweight enough for computing devices that are deprived of powerful GPUs. The system detects vehicles, estimates their travelling speeds, detects the license plates of the overspeeding vehicles resulting in license plate extraction of all the violators, and then keeps a stored record of them. Results show that the system is highly precise and has decent accuracy with a high detection speed but isn't accurate enough when handling challenging images. The scope of this work is extended but not limited to autonomous vehicles, unmanned aerial vehicles, intelligent transportation systems, robotics, intelligent AI agents, domestic security, healthcare and all other areas where high speed accurate detection is needed on an embedded smart device that may not be computationally super powerful.

Keywords; YOLOv9, EasyOCR, Vehicle Detection, License Plate Recognition, Speed Estimation, Real-time Traffic Surveillance

---

*Corresponding author: bmushtaq@leomail.tamuc.edu
Received: May. 7. 2025 Accepted: Jul. 16. 2025 Published: Sep. 31. 2025

Cite this paper as : Sang Suh, and Bilal Mushtaq (2025) “Vehicle Recognition and Speed Monitoring System using YOLOv9”, Journal of Industrial Information Technology and Application, Vol. 9. No. 3, pp. 1130 - 1136

1. Introduction

Computer Vision is the domain of Artificial Intelligence (AI) that focuses on enabling computers to be able to understand images and videos and extract vital information that could be further processed to make decisions and/or predictions. Researchers have worked extensively in this domain, and have used various statistical and probabilistic models, in addition to complex mathematical techniques to address vehicle detection and classification problems. Even though they have seen considerable success, there are still some significant hurdles that need to be overcome to effectively solve this problem.

Traditional methods such as Blob analysis [1][2] and Haar like feature classification [3][4][5][6] techniques aren’t robust enough to perform in low lightning conditions. Support Vector Machines and Principal Component Analysis [7], and Convolutional Neural Networks [8][9][10] fail to perform in congested traffic settings, while ResNet [11] and Fast R-CNN [12] aren’t able provide a fast detection speed due to their deep layered network size. 3D modeling techniques perform best when distinguishing among vehicles of varying shapes and sizes but are not scalable enough to work with large commercial datasets [13] [14] [15].

To address these challenges, we propose a framework based on YOLOv9 [16] detection model and EasyOCR [17] feature extraction model that aims to deliver faster vehicle detection and license plate extraction in real-time traffic scenarios without compromising significantly on the accuracy, along with the ability to estimate the travelling speeds of the respective vehicles so that the violators could be penalized and brought to account. The hallmark of this system is its ability to operate on systems that are deprived of the luxury of having powerful GPUs at their disposal all the time, making it a practical solution for the existing commercial traffic infrastructure.

The model was trained on a dataset containing diverse sets of vehicle images containing varying angles, different weather and lightning conditions, and various license plate designs, for a 1000 epoch. The experimental results show that the framework is highly accurate in avoiding false positives but missed some valid detections, making the overall true positive score decent. Even though the detection accuracy is impressive in general cases, the performance drops a bit when handling difficult cases. Moreover, the framework successfully carries out speed estimation of the vehicles in motion and can differentiate the overspeeding ones from the vehicles that are found to be moving within the designated speed limit.

2. Conclusion

The accuracy of this system can be inferred from its precision, recall, and mean average precision (mAP) metrics, as shown in Figure 1. Precision (0.75–0.95) indicates the model is highly accurate in avoiding false positives, meaning when it predicts a detection, it is correct most of the time. Recall (0.6–0.8) shows the system moderately captures true positives, suggesting it misses some valid detections. mAP@50 (0.7–0.85) represents strong overall detection accuracy at a commonly used IoU threshold (50%). mAP@50-95 (0.25–0.40) highlights stricter evaluation across multiple IoU thresholds, where the system's performance drops, suggesting challenges in precise localization and handling difficult cases.

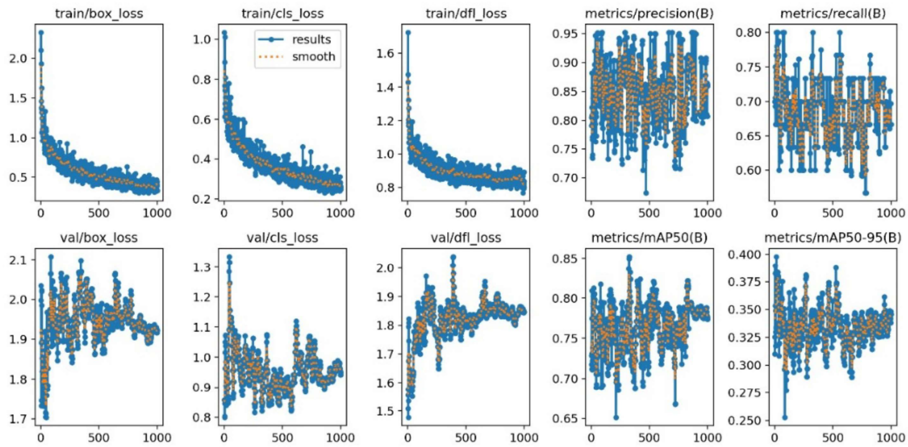

Figure 1. Performance metrics of the VRSMY9 system after 1000 epoch

To test the system in real-time, we set the speed limit at 10 mph and utilized a live traffic surveillance video to see how the system performs in a such a live scenario. Figure 2 shows the results of a vehicle travelling within the speed limit.

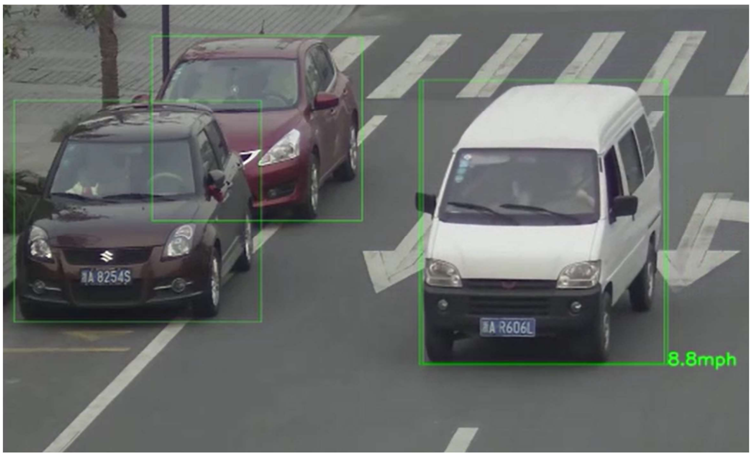

Figure 2. Vehicle travelling within the speed limit using VRSMY9

Figure 3 shows an instance of vehicles travelling over the speed limit. Once the system detects that they are travelling over the speed limit, it detects the license plate, takes a snapshot of the plate and stores it with its contents in the specified folder keeping a record of all the overspeeding vehicles.

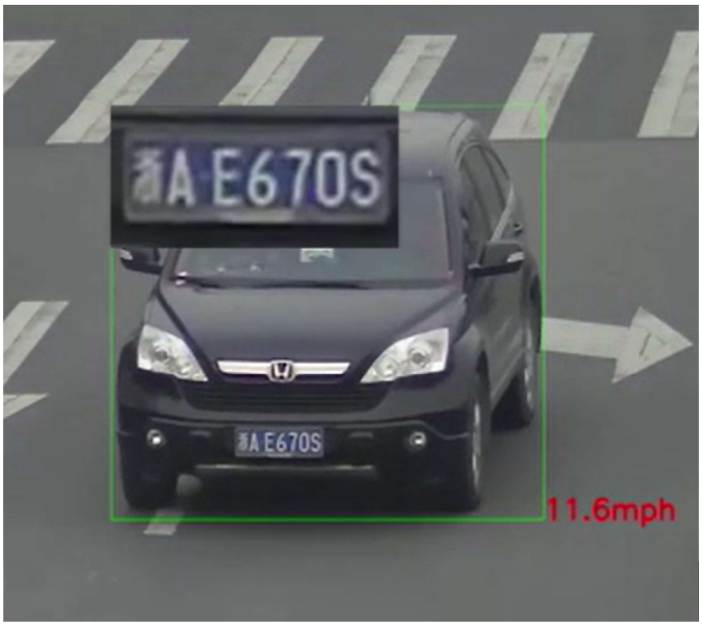

Figure 3. VRSMY9 Vehicle travelling over the speed limit

3. Comparative Analysis and Discussion

Table 1 highlights the comparison of the most renowned techniques with our proposed work.

Table 1. Comparison of related works with the proposed VRSMY9

Technique(s) Used	Contributions	Shortcomings
Blob Analysis [1] [2]	Cost effective compared to traditional methods	Not effective in low lightning, bad weather and high traffic density
Haar like features classification [3] [4] [5] [6]	Better detection results in complex and dense traffic conditions	Lack of robustness and unable to detect accurately in low lightning conditions
SVM and PCA [7]	Improvement in detection time	Not effective whilst detecting real-time congested traffic
Convolutional Neural Network (CNN) [8] [9] [10]	More effective than traditional methods when handling various vehicle shapes and sizes	Relatively lower classification accuracy in adverse traffic conditions
ResNet [11]	Deep layers produce better classification results	Computationally expensive
Fast R-CNN [12]	Faster processing speed compared to its predecessors	Needs large amounts of data and isn't fast enough for real-time applications
3D modeling and 3D Bounding Boxes [13] [14] [15]	Most accurate and effective in distinguishing between closely related vehicle models and variations	Lack of scalability when working with large datasets
Proposed VRSMY9	Lightweight model with an improvement in detection and feature extraction speed	Not accurate enough when handling challenging images

The biggest challenge that we came across was finding the right data with enough images that weren't distorted and had visible license plates. Another lesson learned was that speed estimation powered by computer vision isn't as effective compared to utilizing radar sensors for estimating speeds. Even though our aim to achieve a faster detection speed without compromising too much on accuracy was achieved to some extent, the model didn't showcase a high enough accuracy as we would have liked when handling difficult cases such as overlapping vehicle images and unclear license plates. We hope to take this research as a step forward in the right direction so that one day we can achieve the unmet objectives in their entirety as well.

This framework serves as an important steppingstone for further research in the fields of autonomous vehicles, unmanned aerial vehicles, intelligent transportation systems, robotics, intelligent AI agents, domestic security, healthcare and various other walks of life, where computationally inexpensive systems are to be built without compromising on accuracy and performance.

References

[1] G. Salvi. “An Automated Vehicle Counting System Based on Blob Analysis for Traffic Surveillance”. In Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV), (2012), p.1

[2] S. Ramalingam, and V. Varsani. “Vehicle detection for traffic flow analysis”. In IEEE International Carnahan Conference on Security Technology (ICCST), (2016), p.1-8

[3] Y. Wei, Q. Tian, J. Guo, W. Huang and J. Cao. “Multi-vehicle detection algorithm through combining Harr and HOG features”. In Mathematics and Computers in Simulation Vol: 155, Elsevier, (2019), p.130-145

[4] Y. Hasan, M. U. Arif, A. Asif and R. H. Raza. “Comparative Analysis of Vehicle Detection in Urban Traffic Environment Using Haar Cascaded Classifiers and Blob Statistics”. In 2016 Future Technologies Conference (FTC), (2016), p.547-552

[5] S. Han, Y. Han and H. Hahn. “Vehicle Detection Method using Haar-like Feature on Real Time System”. In International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering Vol: 3, No: 11, (2009).

[6] S. ElKerdawy, R. Sayed, and M. ElHelw. “Real-Time Vehicle Detection and Tracking Using Haar-Like Features and Compressive Tracking”. In First Iberian Robotics Conference. Advances in Intelligent Systems and Computing, vol: 252. Springer, (2013).

[7] X. Wen, L. Shao, W. Fang and Y. Xue. “Efficient Feature Selection and Classification for Vehicle Detection”. In IEEE transactions on circuits and systems for video technology, (2015)-03, Vol.25 (3), p.508-517

[8] D. He, C. Lang, S. Feng, X. Du and C. Zhang, “Vehicle Detection and Classification Based on Convolutional Neural Network”, In Proceedings of the 7th International Conference on Internet Multimedia Computing and Service, (2015), p.1-5.

[9] X. Li, M. Ye, M. Fu, P. Xu, and T. Li. “Domain Adaption of Vehicle Detector based on Convolutional Neural Networks”. In International Journal of Control, Automation, and Systems, (2015), 13(4), pp.1020-1031

[10] L. Chen, F. Ye, Y. Ruan, H. Fan and Q. Chen, “An algorithm for highway vehicle detection based on convolutional neural network”. In EURASIP Journal on Image and Video Processing, (2018), Vol. 1, p.1-7

[11] H. Jung, M.Choi, J. Jung, J. Lee S. Kwon and W. Y. Jung. “ResNet-based Vehicle Classification and Localization in Traffic Surveillance Systems”. In IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), (2017), p.934-940

[12] R. Girshick. “Fast R-CNN”. In IEEE International Conference on Computer Vision (ICCV), 2015, p.1440-1448

[13] Y-L. Lin, V. I. Morariu, W. Hsu, and L. S. Davis. “Jointly Optimizing 3D Model Fitting and Fine Grained Classification”. In ECCV, (2014).

[14]] J. Sochor, A. Herout and J. Havel. “BoxCars: 3D Boxes as CNN Input for Improved Fine-Grained Vehicle Recognition.” In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, p.3006-301

[15] J. Sochor, J. Spanhel and A. Herout. “BoxCars: Improving Fine-Grained Recognition of Vehicles using 3D Bounding Boxes in Traffic Surveillance”. In IEEE Transactions on Intelligent Transportation Systems, 2019, Vol.20 (1), p.97-108

[16] C.Y. Wang, I.H. Yeh, and H.Y.M. Liao, “YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information”. Available at https://arxiv.org/abs/2402.13616, (2024).

[17] JaidedAI, “EasyOCR”. Available at: https://github.com/JaidedAI/EasyOCR, (2020).