Vehicle-Specific Object Detection and Temporal Tracking in Video Stream

Aryan Parekh
Department of Computer Science
University of Western Ontario
London, Canada
aparek8@uwo.ca

Ravindranath Sawane
Department of Computer Science
University of Western Ontario
London, Canada
rsawane@uwo.ca

***Abstract*—In the field of vehicle systems one of the aspects is ensuring accurate and immediate detection and tracking of objects. This research delves into how the YOLOv8 object detection algorithm can be applied to a dataset focused on vehicles, which was collected from BDD100K. The findings demonstrate performance, in detecting objects with a precision rate of 56.2% a recall rate of 34.7% and a mean Average Precision (mAP) of 40.4% at an Intersection Over Union (IoU) threshold of 0.5. Additionally this study includes an analysis, between the ByteTrack and BoT SORT tracking algorithms revealing that ByteTrack surpasses BoT SORT in effectively maintaining object identities under challenging conditions frequently encountered in vehicular environments. The results of this study not only enhance the accuracy of detection algorithms for transport systems but also endorse the use of ByteTrack for scenarios requiring reliable tracking. The findings of this study have substantial ramifications for the advancement of dependable and effective navigation systems.**

***Index Terms*—Object Detection; Object Tracking; YOLO; Multiple Object Tracking; ByteTrack; BoT-SORT**

I. Introduction

In the dynamic realm of vehicles and intelligent transportation systems, the precise identification and continuous monitoring of objects in real-time play a pivotal role. The present state of vehicular technologies highlights the necessity for solid object detection and tracking capabilities to improve safety, navigation, and traffic management. Nevertheless, there are ongoing difficulties in guaranteeing consistent and dependable functionality in a wide range of constantly changing circumstances. As a result, more research is needed to develop efficient real-time applications tailored explicitly for intricate environments.

The primary objective of this study is to address the existing disparity by incorporating and evaluating the effectiveness of YOLOv8 for object detection and ByteTrack and BoT-SORT in the context of object tracking, particularly from the standpoint of a vehicle. The aim is to create a system that can accurately and rapidly identify and track a specific group of essential objects while considering the distinct difficulties posed by environments involving vehicles.

The study encompasses a comprehensive implementation of the detection and tracking algorithms, followed by a meticulous comparative analysis of the tracking algorithms. The results obtained in this study provide evidence supporting the feasibility and efficiency of the proposed integration. The findings also highlight each tracking method's unique performance characteristics and suitability.

The novelty of this research resides in the customized implementation and comparative analysis of these sophisticated algorithms within the domain of vehicular systems, thereby making a valuable contribution to the field by addressing the identified research deficiency. The advancements contribute to the theoretical comprehension of algorithmic performance in object tracking and detection while offering practical implications for real-time implementations in vehicles and traffic systems.

The findings have broad implications for both theoretical understanding and practical applications, providing a basis for further investigation and advancement in vehicle technologies and intelligent transportation systems. It establishes the foundation for enhanced safety protocols, optimized traffic dynamics, and cutting-edge navigation technologies.

In this study, the research paper outlines the necessity for advanced object detection and tracking in vehicles, setting this as its backdrop in the "Background and Related Work" section. It then states the objective to evaluate YOLOv8, ByteTrack, and BoT-SORT for vehicular environments in the "Research Objectives" section. The "Research Methodology" details these algorithms' implementation and comparative analysis. "Results" present the findings, demonstrating the effectiveness of the integrated system. Finally, "Conclusions and Future Work" discusses the implications for vehicular technology and proposes directions for further research, emphasizing the study's contribution to the field's theoretical and practical advancements.

II. Background and Related Work

The pursuit of creating more reliable and efficient vehicles and intelligent transportation systems has led to significant advancements in object detection and tracking technologies. These technologies play a role, in improving vehicle safety, navigation and traffic management. However the real challenge lies in ensuring dependable performance in the changing and varied environments where vehicles operate. In this section we will delve into the background and existing research with a focus, on the methods and technologies used to integrate YOLOv8 for object detection and enhance object tracking using ByteTrack and BoT-SORT.

A. Object Detection and Tracking in Vehicles

Detecting and tracking objects are aspects of vehicle systems as they provide information, for decision making and vehicle control. To ensure the technologys reliability it needs to overcome challenges such as changes in lighting conditions, obstructions and the presence of moving objects. One possible solution to address these challenges is to incorporate algorithms like YOLO (You Look Once) for object detection along with ByteTrack and BoT SORT, for object tracking.

B. YOLO (You Only Look Once) Series

The YOLO algorithms, YOLOv8 have had an impact, on advancing real time object detection. J. Redmon et al. [1] introduced YOLO as a system for detecting objects in time which has undergone multiple enhancements to improve both speed and accuracy. The ability of YOLO to perform object detection in time makes it an excellent option, for integrating into vehicle systems. The latest versions, including YOLOv8 are designed to offer detections while utilizing fewer computational resources making them well suited for onboard vehicle systems.

C. ByteTrack

ByteTrack, a object tracking algorithm developed by Y. Zhang et al. [2] is widely recognized for its simplicity and efficiency. It excels in tracking objects in busy scenes making it an appealing choice, for enhancing vehicular object tracking systems especially when dealing with complex environments.

D. BoT-SORT

The BoT-SORT algorithm, created by Aharon, N., Orfaig, R. And Bobrovsky, B. [3] is a method specifically developed to track pedestrians. Its main focus is, on creating associations between individuals making it highly efficient in constantly changing environments. By incorporating BoT SORT into the system the tracking of pedestrians and other important objects, around vehicles can be significantly improved. This enhancement contributes to navigation. Helps prevent collisions.

E. Comparative Analysis and Implementation

The efficiency of these algorithms is frequently assessed by conducting analyses in operational settings. To evaluate the performance and applicability of YOLOv8, for object detection as to compare ByteTrack and BoT SORT for object tracking extensive experimentation is necessary. This involves examining their aspects implementing them practically and testing their efficacy in real world scenarios using video streams.

F. Datasets

The BDD100K dataset, introduced by F. Yu et al. [4], provides a diverse driving dataset for heterogeneous multitask learning. Such comprehensive datasets are critical for training, testing, and validating the object detection and tracking algorithms, ensuring the systems are robust and reliable under different conditions. This study employed a systematic methodology by selecting a reduced and controllable subset of 10,000 images from the comprehensive BDD100K dataset for training and refining the YOLOv8 model. The BDD100K dataset, renowned for its extensive and varied assortment of driving-centric videos and images, poses notable computational resource and training time constraints. By systematically sampling a subset of the dataset, the study guarantees that the YOLOv8 model encounters a diverse range of driving conditions and objects, striking a harmonious equilibrium between the comprehensiveness of the data and the limitations imposed by practical training considerations. This subset comprises images that have been appropriately labelled, which are essential for tasks related to object detection. This subset facilitates the training of models more efficiently without compromising the diversity and quality of the training data. The decision to utilize a reduced subset effectively tackles constraints related to resources while still harnessing the valuable real-world scenarios contained in the BDD100K dataset.

III. Methods

Objective [O1]: To deploy and incorporate YOLOv8, a state-of-the-art deep learning model, for real-time identification and localization of objects within a vehicle-centric setting.

Significance [O1]: This objective pertains to the fundamental requirement of achieving precise and rapid object detection as a preliminary step towards achieving efficient tracking. The deployment of YOLOv8, renowned for its high efficiency and accuracy, is essential in augmenting self-driving vehicles' safety and navigational functionalities. The above process establishes the foundation for subsequent monitoring and examination, enhancing the theoretical comprehension and practical implementations of object detection in dynamic environments.

Objective [O2]: To evaluate and contrast the efficacy of the ByteTrack and BoT-SORT algorithms in object tracking after detection.

Significance [O2]: Understanding the performance and appropriateness of diverse tracking algorithms under different conditions is imperative for optimizing real-time vehicle systems. This objective aims to identify the strengths and limitations of each method, providing valuable insights for researchers and practitioners in selecting or designing the appropriate tracking solutions for specific vehicular applications. Using comparative analysis enhances the optimization of tracking systems, thereby directly influencing the efficiency and reliability of vehicles and intelligent transportation systems.

Objective [O3]: To evaluate the real-time suitability and precision of the integrated system under various environmental conditions.

Significance [O3]: Assessing the system within real-world contexts ensures that the scientific investigation is effectively translated into feasible solutions for practical implementations. This objective is vital in conducting stress tests to evaluate the system's adaptability and robustness, resulting in real-time performance and reliability enhancements. The results will provide valuable insights for enhancing and advancing vehicular technology, impacting systems' theoretical foundations and practical applications.

This study involves the incorporation and comparative examination of YOLOv8 for object detection, ByteTrack, and BoT-SORT for object tracking within a system for detecting and tracking objects centred around vehicles. The methodology has been devised to evaluate these algorithms' suitability, precision, and effectiveness in real-time settings. The process encompasses a sequence of systematic procedures, encompassing the execution, experimentation, comparison, and verification of the system across diverse circumstances.

A. YOLOv8 for Object Detection

YOLOv8 is employed as the primary module for object detection. The algorithm is selected based on its high computational efficiency and precise object detection capabilities across various environmental conditions.

1) ***Architecture****:* Fig. 1 depicts the intricate structure of YOLOv8. YOLOv8 employs a comparable underlying structure to YOLOv5 but introduces modifications to the CSPLayer, now called the C2f module. The C2f module, the cross-stage partial bottleneck with two convolutions, integrates high-level features and contextual information to enhance detection accuracy. YOLOv8 employs an anchor-free model incorporating a decoupled head to handle objectness, classification, and regression tasks independently. This design facilitates task specialization within each branch and enhances the model's accuracy. The output layer of YOLOv8 employs the sigmoid function as the activation function for the objectness score. This score represents the probability of an object being present within the bounding box. The softmax function is employed to calculate the class probabilities, which indicate the likelihood of objects belonging to each potential class.

The YOLOv8 model incorporates the CIoU [6] and DFL [7] loss functions to calculate the bounding box loss, while binary cross-entropy is employed for classification loss. These losses have enhanced object detection performance, specifically in scenarios involving diminutive objects.

The YOLOv8 framework includes a semantic segmentation model known as YOLOv8-Seg. The backbone utilized in this context is a CSPDarknet53 feature extractor. It is then accompanied by a C2f module, which deviates from the conventional YOLO neck architecture. The C2f module is subsequently accompanied by two segmentation heads, which acquire the ability to forecast the semantic segmentation masks for the given image input. The model exhibits detection heads that resemble those found in YOLOv8. These detection heads are comprised of five detection modules and a prediction layer. The YOLOv8-Seg model has demonstrated superior performance on multiple object detection and semantic segmentation benchmarks while maintaining notable speed and efficiency.

YOLOv8 is executable through the command line interface (CLI) or can be installed as a PIP package alternatively. Furthermore, it has numerous integrations for labelling, training, and deploying.

When tested on the MS COCO dataset test-dev 2017 [8], YOLOv8x demonstrated an average precision (AP) of 53.9% using images of size 640 pixels. In comparison, YOLOv5 achieved an AP of 50.7% on the same input size. YOLOv8x also exhibited a processing speed of 280 frames per second (FPS) on an NVIDIA A100 with TensorRT. [5]

2) ***Training****:* In the context of employing YOLOv8 for object detection, specifically in vehicular surroundings, the training procedure assumes a pivotal role in guaranteeing that the model is meticulously calibrated to the distinct requirements and dynamics of the designated environment.

- *Data Preparation*: The initial phase entails gathering a comprehensive dataset encompassing various vehicle-specific objects across different environmental conditions; in this case, a part of 10k images from the BDD100k dataset. The dataset encompasses annotations for various classes: rider, truck, traffic light, car, traffic sign, bike, motor, bus, train, and person. It is imperative to ensure that the dataset encompasses a wide range of scenarios, lighting conditions, and backgrounds to train a model with solid resilience.
- *Configuration of Model*: Before the commencement of training, the architecture of YOLOv8 is adjusted to identify the particular classes pertinent to environments involving vehicles. The procedure entails the establishment of the C2f module, implementing a decoupled head, and preparing the output layer to accommodate the specified number of classes.
- *Transfer Learning*: Exploiting a pre-trained YOLOv8 model expedites the training procedure. The model that has undergone pretraining on a comprehensive dataset such as MS COCO is subsequently fine-tuned using a dataset specifically focused on vehicles (BDD100K). This methodology enables the model to maintain its overall object detection capabilities while focusing on ten vehicle-related object classes mentioned in Table I.
- Training Strategy: The model is subjected to intensive training on a high-performance GPU to accommodate the demanding computational demands. Various augmentation techniques are utilized during training to improve the model's generalization ability. The CIoU and DFL loss functions are employed to assess the accuracy of bounding boxes, whereas binary cross-entropy is utilized for classification loss.
- *Hyperparameter Optimisation*: Fine-tuning key hyperparameters, such as learning rate, batch size, and epochs, is conducted to discover the optimal trade-off between

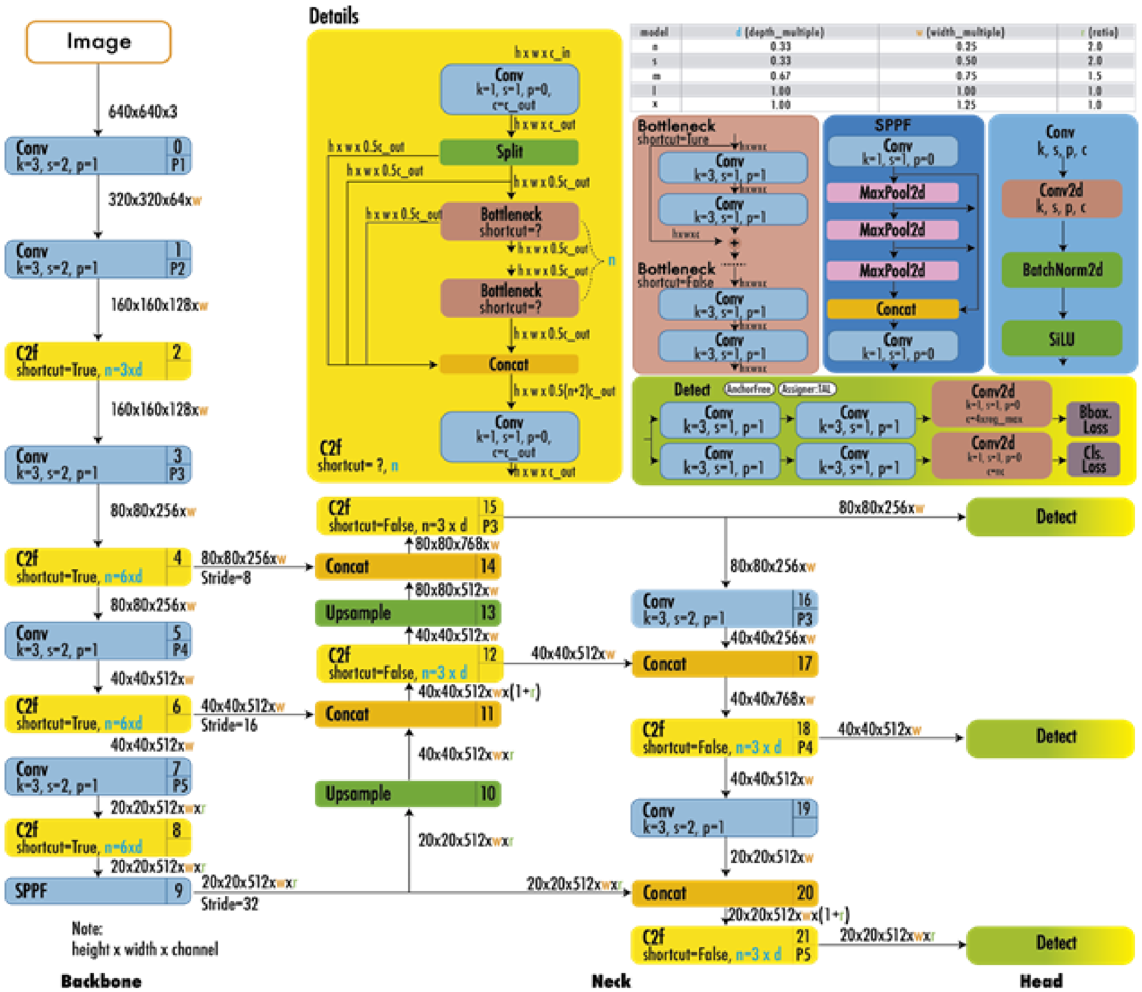

Fig. 1. YOLOv8 Architecture. Diagram based in [5].

0	rider
1	truck
2	traffic light
3	car
4	traffic sign
5	bike
6	motor
7	bus
8	train
9	person

TABLE I
VEHICLE SPECIFIC OBJECT CLASSES USED

computational efficiency and predictive performance. The objective is to attain high precision (AP) when detecting objects specific to vehicles while maintaining the crucial real-time detection capabilities necessary in vehicular environments.

- *Evaluation and Iteration*: Following the training process, the model undergoes evaluation using an independent validation set distinct from the dataset specific to vehicles. Each class's metrics, such as average precision (AP), are carefully examined. Based on the obtained results, it is recommended to carry out subsequent iterations of training by modifying the parameters or incorporating supplementary data to enhance the model's performance.
- *Deployment Readiness*: After the model attains acceptable performance metrics and demonstrates resilience under diverse circumstances, it is primed for integration with object-tracking algorithms for further analysis.

By implementing this systematic training methodology, YOLOv8 undergoes fine-tuning to enhance its capabilities as a robust instrument for detecting objects specific to vehicles. This significantly improves computational efficiency and accuracy, essential attributes for effectively navigating the dynamic and occasionally unpredictable settings encountered in vehicular environments.

*3) **Justification**:* : The choice is grounded in the established track record of YOLOv8 in effectively managing real-time limitations while upholding a notable level of detection precision. This characteristic renders it well-suited for the dynamic and frequently uncertain circumstances in vehicular environments.

B. ByteTrack and BoT-SORT for Object Tracking

Following object detection, objects are accurately monitored and traced across frames using ByteTrack and BoT-SORT algorithms. Both algorithms are implemented separately to preserve the identities and trajectories of the detected objects.

ByteTrack is a object tracking (MOT [9]) system that has been specifically designed to enhance the accuracy of tracking particularly in challenging scenarios, like occlusion or cases with low object detection scores. The tracker has proven its effectiveness by performing on a range of demanding datasets such, as MOT17, MOT20, HiEve and BDD100K. This indicates that it excels in tracking situations involving crowds, complex events and driving scenarios.

The BoT-SORT system is a tracking system designed to track and identify multiple pedestrians. It uses state of the art techniques to analyze motion and appearance compensates, for camera movement. Employs Kalman filter state estimation for tracking. In the tracking by detection approach the system predicts the positions of objects. Connects new detections with existing tracks. The BoT SORT algorithm shows improvements compared to methods by effectively addressing limitations found in similar algorithms, like SORT.

1) ByteTrack Implementation: The ByteTrack system utilizes a tracking unit called the Basic Unit of Tracking Element (BYTE) to track and analyze data. At its core the BYTE methodology is employed for data association, which involves categorizing detection boxes into score or low score detections. The first step is to establish a correlation, between high score detection boxes and tracks based on motion or appearance similarity. This process often includes making use of Kalman Filter predictions. For tracks that do not find a match they are then connected to detection boxes with scores in a round of association. This process aims to identify and retrieve objects with low detection scores while eliminating background detections.

- In the ByteTrack algorithm, a detection score threshold is implemented to classify detection boxes into two categories based on their scores: high scores and low scores.
- For every frame, the system utilizes predictive algorithms to estimate the positions and scores of detection boxes. It then links detection boxes with high scores to existing tracklets and subsequently associates remaining tracklets with detection boxes that have low scores.
- The system employs the Kalman filter to make predictions of the positions of tracklets. It then utilizes the Hungarian Algorithm or other methodologies to perform matching based on metrics such as Intersection over Union (IoU) or similarities in appearance.
- The system maintains associations and updates track identities by considering each detection box, resulting in a significant reduction in identity switches and an enhancement in tracking robustness.

Fig. 2 demonstrates the procedure mentioned above.

2) BoT-SORT Implementation: The tracking algorithm analyses video sequences frame-by-frame, initially identifying objects and extracting visual characteristics. The detections are classified into two groups, high and low, based on a predetermined threshold. The Kalman filter [10] is utilized to forecast the new positions of established tracks for every frame while accounting for camera motion compensation to correct any camera movements.

Linking detections to pre-existing tracks combines Intersection over Union (IoU) and appearance data. The system initially endeavours to establish correspondences between detections and tracks by considering both criteria associated with highly confident detections. It exclusively employs Intersection over Union (IoU) as the evaluation metric for the remaining detections. Following the association process, the system updates the matched tracks by incorporating new detections and making necessary adjustments to their Kalman filter states. Tracks are effectively managed by eliminating those not meeting specific duration criteria and creating new tracks for detections with a high confidence level that do not match existing tracks.

The iterative procedure iterates for every frame, guaranteeing the robust tracking of each object across the entire video. The implementation follows a sequential process consisting of detection, prediction, association, and update steps. This iterative approach allows for continuously improving track positions and identities across multiple frames.

3) Justification: The selection of ByteTrack and BoT-SORT is based on their demonstrated resilience and effectiveness in tracking. The justification for utilizing both methods is to conduct a comparative analysis and gain a comprehensive understanding of the respective benefits in the specific domain of vehicle-based tracking.
The methodology is designed to offer a thorough comprehension of the performance of various object-tracking algorithms within a vehicular setting and to ascertain the optimal approaches and tactics for real-time object detection and tracking. The system is formulated to follow an iterative and adaptable approach, enabling ongoing enhancements and updates following the most recent scientific research and technological progress.

IV. Results

A. Object Detection

The YOLOv8 model has been optimized to detect vehicles with results. The precision rate of the model, which stands at 56.2% demonstrates its effectiveness, in identifying positives. This achievement is particularly notable considering the complexity of the task. However the observed recall rate of 34.7% suggests that the model struggled to identify a portion of positive cases. This indicates a need to improve the sensitivity of the model. The mean Average Precision (mAP) at an Intersection Over Union (IoU) threshold of 50% was 40.4% indicating a level of accuracy, across various classes demonstrated by the model.

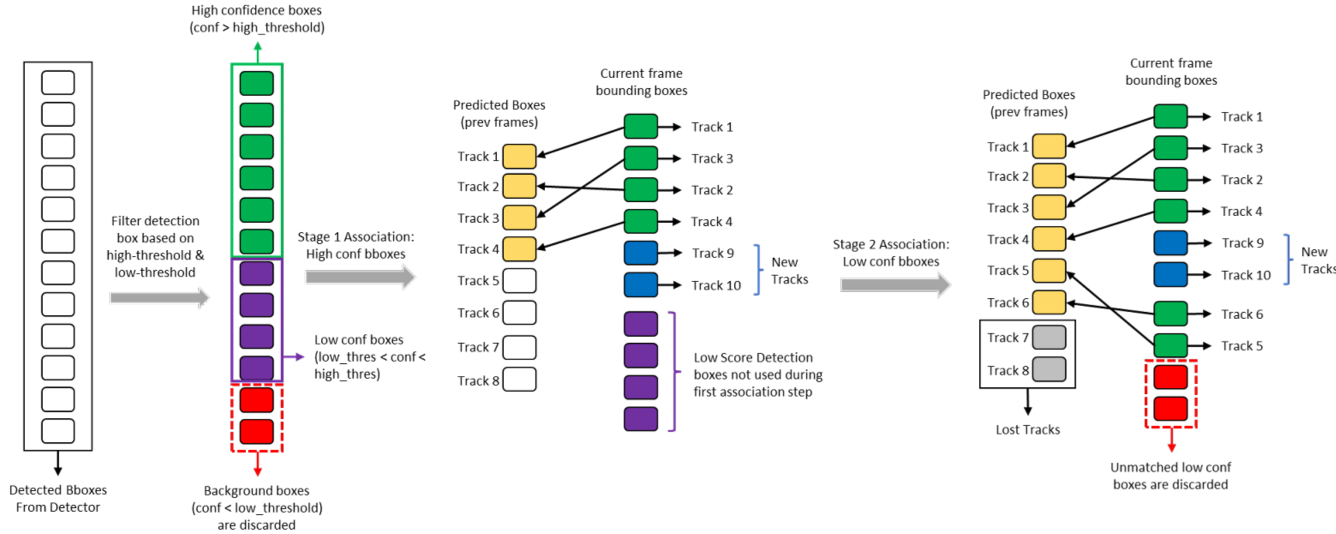

Fig. 2. BoT-SORT Process

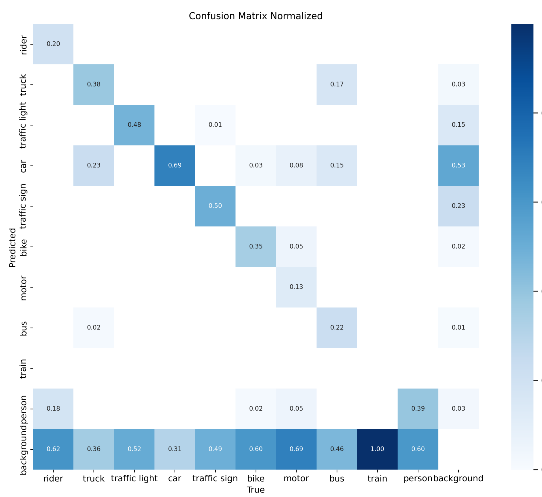

Fig. 3. Normalized Confusion Matrix of Object Categories

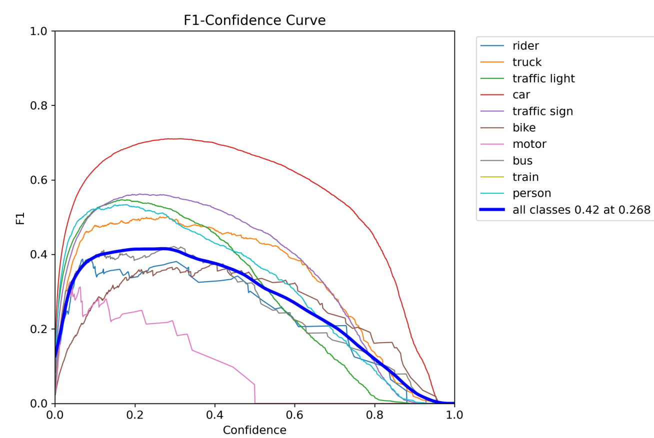

Fig. 4. F1 Confidence Curve

After analyzing the normalized confusion matrix in Fig. 3, several significant observations were identified. The model exhibited a robust capacity for identifying 'cars', attaining the utmost normalized value of accurate predictions within this category. The observed success can be ascribed to the regular occurrence of automobiles in the dataset, which enables the model to acquire resilient characteristics for detecting cars. Nonetheless, the confusion matrix also exposes noteworthy misclassifications, such as where 'bikes' were frequently mistaken for 'motor' categories, suggesting a potential convergence in the feature space or an absence of discernible attributes acquired by the model for these classes.

The F1-Confidence Curve, which represents the geometric mean of precision and recall as a function of various confidence thresholds, as observed in Fig. 4, exhibited a maximum F1 score at a particular threshold, beyond which the score exhibited a decrease. This peak corresponds to the optimal trade-off attained by the model in terms of precision and recall. The model's performance exhibits a notable decline at elevated confidence levels, potentially attributable to an excessively cautious threshold resulting in the omission of numerous true positive instances.

The utilization of the Precision-Recall Curve played a pivotal role in conducting a more detailed analysis of the model's performance. The curves, as seen in Fig. 5, representing categories such as 'traffic lights' and 'cars' exhibited higher proximity to the upper right corner, which suggests superior performance. In contrast, categories such as 'bikes' and 'trains' tended to the lower right quadrant, indicating diminished precision and recall. Specifically, the category 'train' demonstrated a precision and recall value of zero, indicating a complete failure in detection. This could be attributed to a need for sufficient training examples or the intricate features of this

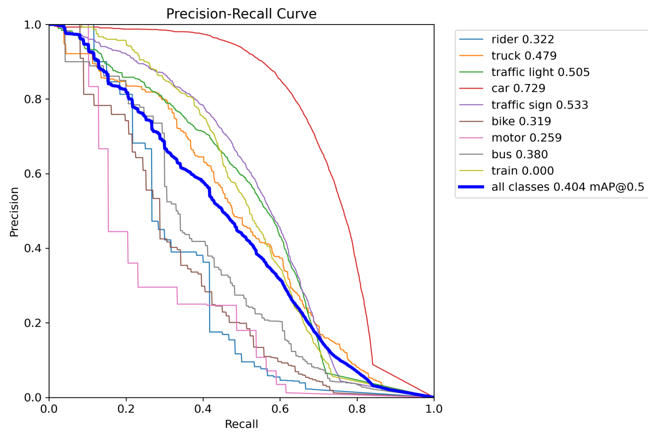

Fig. 5. Precision-Recall Curve

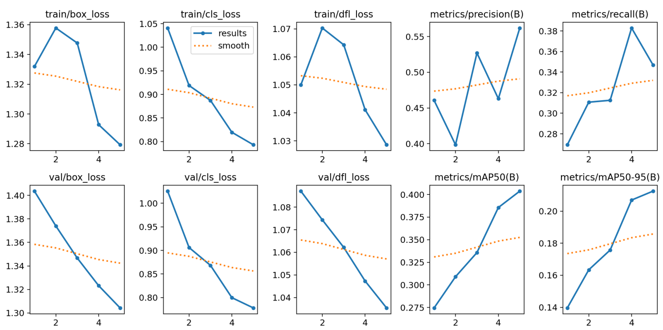

Fig. 6. Graphs of loss during training and validation

particular category.

Finally, the graphs depicting the loss during training and validation and the metrics for precision and recall (given in Fig. 6) were examined across the training epochs. The observed decline in loss values suggests that the model has effectively learned and improved. Nevertheless, variations in precision and recall throughout epochs indicated instability in the learning process, which could be enhanced by implementing supplementary regularisation techniques or an improved training strategy. ‘

The novelty in our findings arises from the successful fine-tuning of YOLOv8 to detect objects specific to vehicles. This led to achieving a mean Average Precision (mAP) of 40.4% at an Intersection over Union (IoU) of 50% while utilizing a smaller dataset size. This highlights the model's efficacy in various and dynamic settings by utilizing a more minor but representative training dataset, thereby questioning the belief that extensive datasets are necessary for achieving high accuracy in object detection. The results demonstrate a novel framework for enhancing data utilization efficiency in real-time applications within vehicles and intelligent traffic systems.

Additionally, a comparison of the predicted and labeled frames is displayed in the Fig. 7 and Fig. 8 respectively.

In brief, the YOLOv8 model that underwent fine-tuning

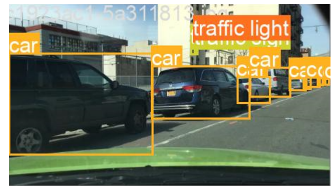

Fig. 7. Labelled Frame

Fig. 8. Predicted Frame

exhibited moderate effectiveness in detecting objects specific to vehicles. The findings emphasize the significance of class representation in the dataset and the necessity of precise calibration of model confidence. Additional investigation into the optimization of model parameters and the augmentation of data is necessary to mitigate the discrepancies in detection accuracy among different classes and improve the overall performance of the model.

B. Object Tracking

In addition we have conducted an evaluation of the ByteTrack and BoT SORT object tracking algorithms in a controlled setting that is specifically tailored for tracking vehicles. After analyzing the videos it is evident that ByteTrack outperforms BoT SORT in terms of maintaining the identity of tracked objects over a period. ByteTrack demonstrates a heightened ability to track objects across successive frames while maintaining their identities consistently. This is particularly important when vehicles and other objects are in continuous motion, and occlusions occur frequently.

The video analysis results indicate that ByteTrack demonstrated a more remarkable ability to maintain object identities even when faced with interruptions caused by occlusion or rapid movement (Fig. 12 and Fig. 14). In contrast, BoT-SORT tended to experience a higher frequency of object misplacement under comparable circumstances (Fig. 11 and Fig. 13). This phenomenon was notably apparent in congested traffic situations where numerous objects were nearby, resulting in frequent occlusion. The discrepancy in performance observed between the two tracking algorithms can be ascribed to the superior association metrics employed by ByteTrack. These metrics demonstrate a more remarkable ability to handle object interactions and occlusions effectively than BoT-SORT.

Moreover, the empirical data is substantiated by quantitative metrics. ByteTrack exhibited greater consistency in the Intersection over Union (IoU) metric for vehicles across consecutive frames. Furthermore, the durability of monitored identities, quantified by the count of successive frames in which an object was effectively tracked, exhibited a preference for ByteTrack. The BoT-SORT system exhibited reduced efficacy in the highly congested vehicular setting, resulting in shorter durations of identity tracking and an increased frequency of identity switches. Also, at times, BoT-SORT failed to identify certain objects that ByteTrack could (Fig. 9 and Fig. 10).

The findings emphasize the significance of reliable tracking in intricate surroundings and propose that ByteTrack is more

Fig. 9. BoT-SORT (Frame 271)

Fig. 10. ByteTrack (Frame 271)

Fig. 11. BoT-SORT (Frame 368)

Fig. 12. ByteTrack (Frame 368)

suitable for tasks that demand accurate maintenance of object identity over time, such as driving and urban traffic analysis systems. Our Research suggests that ByteTrack, for keeping tabs on vehicles, could be really effective. We may use and enhance it later.

V. Conclusions and Future Work

This study systematically examines how the YOLOv8 object detection algorithm can be applied and improved specifically for vehicles. It also compares two object tracking algorithms, ByteTrack and BoT SORT to analyze their effectiveness. The results, from using YOLOv8 show that it can accurately identify objects related to vehicles with a level of precision. The precision score is 56.2% the recall score is 34.7% and the mean average precision (mAP) is 40.4% when using an Intersection over Union (IoU) threshold of 50%. These metrics were obtained by selecting a subset of the BDD100K dataset emphasizing the importance of data quality and relevance in achieving reliable detection performance alongside data volume.

Regarding object tracking, our investigation reveals that ByteTrack outperforms BoT-SORT in preserving object identities for periods in challenging situations with frequent occlusions and dynamic interactions among objects. This improved performance is crucial for time applications, like navigation and traffic monitoring, where continuous object tracking plays a role.

The study showcases novel findings by illustrating that successful object detection and tracking in a vehicular setting can

Fig. 13. BoT-SORT (Frame 500)

Fig. 14. ByteTrack (Frame 500)

be accomplished without requiring large datasets. Additionally, the study identifies ByteTrack as a highly promising tool for preserving object identity in complex and ever-changing environments. The discoveries mentioned above significantly contribute to the continuous advancement of intelligent transportation systems and vehicle technologies.

Subsequent research endeavours will prioritize the resolution of the identified constraints, specifically by augmenting the YOLOv8 model's ability to accurately retrieve information and maintain consistency and advancing the tracking algorithms to enhance their robustness in the face of intricate, real-life circumstances. Furthermore, the investigation of hybrid tracking methodologies that integrate the capabilities of ByteTrack and BoT-SORT may result in additional progress in this domain. The findings emphasize the significance of precise data selection and the potential of sophisticated tracking algorithms to fulfil the requirements of real-time object detection and tracking in dynamic vehicular settings

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2016.

[2] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," 2022.

[3] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "Bot-sort: Robust associations multi-pedestrian tracking," 2022.

[4] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," 2020.

[5] J. Terven and D. Cordova-Esparza, "A comprehensive review of yolo: From yolov1 and beyond," 2023.

[6] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, pp. 12 993–13 000, Apr. 2020. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/6999

[7] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," 2020.

[8] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015.

[9] P. Dendorfer, A. Ošep, A. Milan, K. Schindler, D. Cremers, I. Reid, S. Roth, and L. Leal-Taixé, "Motchallenge: A benchmark for single-camera multiple target tracking," 2020.

[10] R. G. Brown and P. Y. C. Hwang, *Introduction to random signals and applied kalman filtering: with MATLAB exercises and solutions, 3rd ed.* New York: Wiley, 1997.