e-ISSN:2582-7219

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

Volume 7, Issue 6, June 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.521

International Journal Of Multidisciplinary Research In Science, Engineering and Technology (IJMRSET)

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

Traffic Sense: An Integrated Approach to Traffic Management through Object Detection and CNN

Akshay Patil, Prof. Flavia Gonsalves

MCA Student, Mumbai Education Trust, Bandra, Mumbai, India

Assistant Professor, Mumbai Education Trust, Bandra, Mumbai, India

ABSTRACT: This paper introduces an Intelligent Traffic Management System (ITMS) powered by Convolutional Neural Networks (CNN) and YOLOv8 for object detection. ITMS accurately detects and classifies traffic scene objects like vehicles, pedestrians, and cyclists in real-time. Trained on extensive traffic datasets, the model analyzes complex traffic patterns, congestion, flow dynamics, and anomalies. Integration with existing infrastructure, such as traffic lights and surveillance cameras, enables scalable deployment. Leveraging CNN and YOLOv8, ITMS enhances urban traffic management by reducing waiting times and alleviating congestion, showcasing its potential for smarter, adaptive transportation networks.

KEYWORDS: YOLO (You Only Look Once), Computer Vision, Machine Learning, Object Detection, Convolutional Neural Network, Yolov8

I.INTRODUCTION

In the current era of rapid urbanization and technological advancements, the need for efficient and intelligent traffic management systems has become increasingly crucial. As the number of vehicles on the roads continues to rise, so do the challenges associated with traffic congestion, accidents, and environmental pollution. Real-time vehicle detection and counting play a pivotal role in addressing these issues by providing valuable insights into traffic patterns and enabling proactive measures to optimize traffic flow. This research paper aims to explore the potential of the YOLOv8 algorithm in real-time vehicle detection and its integration with a traffic signal model for intelligent traffic management. The YOLOv8 algorithm, a state-of-the-art object detection model, has shown remarkable performance in various applications, including vehicle detection. In this study, we have trained the YOLOv8 algorithm on a custom dataset of vehicle image data, comprising five different classes: car, bicycle, motorcycle, bus, and truck. The trained model demonstrates significantly accurate vehicle detection in real-time scenarios, showcasing its ability to identify and classify vehicles with high precision.

Compared to traditional object detection algorithms, YOLOv8 offers several advantages, such as faster processing speeds, higher accuracy, and the ability to detect multiple objects simultaneously. By leveraging the power of YOLOv8 and the custom dataset training, this research paper presents a novel approach to real-time vehicle detection and counting. The system is designed to process live video feeds from traffic cameras, accurately detect vehicles of different classes, and maintain a real-time count of vehicles on the road. The main contribution of this research paper lies in the development of a robust and reliable system for real-time vehicle detection and counting using YOLOv8. The system's integration with a traffic signal model enables data-driven decision-making, allowing for dynamic adjustments to traffic signal timings based on the detected vehicle count. This proactive approach aims to minimize congestion, reduce waiting times at intersections, and enhance overall traffic efficiency.

The paper is organized as follows, Section II covers a literature review of previous work done using a Convolutional neural network on Object Detection of Different Vehicles using CNN and Image Processing Methods. Section III shows information about data that are collected from Kaggle and its description. Sections IV, V, and VI present architecture,

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

methods used and experimental results performed on Object detection for different Vehicles. Finally, section VII presents the conclusion.

II.LITERATURE REVIEW

Recent advancements in deep learning have revolutionized object detection, particularly in vehicle detection. Nguyen (2019) [1] proposed an improved Faster R-CNN framework for fast vehicle detection, demonstrating the effectiveness of convolutional neural networks (CNNs) in this domain. Tang et al. (2017) [2] further explored the use of region-based CNNs and hard negative example mining for vehicle detection in aerial images, showcasing the versatility of these techniques. The You Only Look Once (YOLO) algorithm has gained significant attention due to its real-time performance and high accuracy. Nguyen et al. (2019) [3] presented a high-throughput and power-efficient FPGA implementation of YOLO for object detection, highlighting its potential for embedded systems. The YOLO algorithm has been widely adopted for various applications, including traffic detection projects, as evidenced by the availability of datasets such as the one hosted on Kaggle. The official documentation of Ultralytics[4], the creators of YOLOv8, provides comprehensive resources for understanding and implementing the YOLO family of models. Du (2018) [5] offers an insightful overview of the YOLO algorithm and its relation to the CNN family, further solidifying its position as a powerful tool for object detection. Real-time object detection is crucial for applications such as autonomous vehicles, where Prasanna et al. (2023) [6] demonstrated the use of deep learning techniques for this purpose. Motwani et al. (2022) [7] specifically focused on object detection and tracking for autonomous vehicles using the YOLO algorithm, showcasing its effectiveness in this domain. Prakash et al. (2023) [8] further explored the use of YOLO and CNN for multiple object identification in autonomous cars, highlighting the potential of combining these techniques for enhanced performance. The dataset available at https://www.kaggle.com/datasets/yusufberksardoan/traffic-detection-project [9] offers a comprehensive collection of traffic camera images from various countries worldwide. This meticulously curated dataset serves as a valuable resource for AI researchers, machine learning enthusiasts, and developers working on traffic management solutions.

III.DATA GATHERING

For our experimentation, we require a diverse range of images featuring various types of vehicles such as cars, motorbikes, buses, trucks, etc. Numerous platforms on the internet, such as Kaggle and Microsoft dataset, offer access to such datasets. In our research, we rely on Kaggle's Traffic Detection Project [9] dataset, renowned for its comprehensive collection of traffic camera images sourced from different countries. This dataset provides a global perspective on traffic monitoring and management.

Each image in the dataset is meticulously annotated using bounding boxes, enabling precise identification of various objects, including vehicles, pedestrians, and traffic signs. This annotation makes the dataset particularly suitable for object detection tasks. Moreover, the dataset encompasses images captured under diverse weather conditions, lighting scenarios, and traffic situations, enhancing its applicability to real-world scenarios.

The dataset is organized into three main folders: train, test, and validation. Within the training folder, images are further categorized into subfolders for images and labels. Each image is accompanied by a corresponding text file containing coordinates and class categories for the annotated objects.

Label	Images Count
Training Images	5805
Validation Images	549
Test Images	279

Table 1.1

Table 1.1 shows how many images we have in each phase of model training and evaluation, like in the training phase we have 5805 images of different vehicles.

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

Fig. 1.1 screenshot_898_jpg.rf.9a80a7ad8359c9d6673b5bdc70ab8a7c.jpeg

Each class associated with specific number, that numbered information given below:
Classes Information:
0 – bicycle
1 – bus
2 – car
3 – motorbike
4 – person

For example, Image 1.1 referred from the train images folder named: screen shot_898_jpg.rf.9a80a7ad8359c9d6673b5bdc70ab8a7c.jpeg for this file, In the label folder we have one text file named: screen-shot_898_jpg.rf.9a80a7ad8359c9d6673b5bdc70ab8a7c.txt. In this text file according to each vehicle location, the class and coordinates are mentioned. (2 is a class and the remaining coordinates separated with spaces).

2 0.59375 0.78671875 0.23671875 0.19609375
2 0.73828125 0.6203125 0.16171875 0.115625
2 0.86328125 0.57890625 0.13203125 0.1234375
2 0.31875 0.7546875 0.2421875 0.17578125
2 0.5375 0.6921875 0.18203125 0.13984375

screen-shot_898_jpg.rf.9a80a7ad8359c9d6673b5bdc70ab8a7c.txt

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

IV.MODEL ARCHITECTURE

YOLOv8, or You Only Look Once version 8, is a state- of-the-art object detection model that utilizes Convolutional Neural Networks (CNNs) for real-time object detection tasks. Here's a high-level overview of how YOLOv8 uses and how it internally works:

CNN Architecture: YOLOv8 is built on a CNN architecture, responsible for processing input images and extracting features relevant for object detection. The CNN backbone typically consists of multiple convolutional layers, pooling layers, and other types of operations designed to capture hierarchical features from the input image.

Feature Extraction: The CNN backbone processes the input image through a series of convolutional layers, gradually reducing the spatial dimensions while increasing the depth of feature maps. This process enables the network to extract meaningful features at different scales and levels of abstraction from the input image.

Object Detection Head: On top of the CNN backbone, YOLOv8 typically has a detection head consisting of additional convolutional and fully connected layers. This detection head takes the feature maps generated by the CNN backbone and produces bounding box predictions and associated confidence scores for multiple objects present in the image.

Bounding Box Regression and Classification: The detection head performs bounding box regression to predict the coordinates (e.g., x, y, width, height) of bounding boxes around objects of interest. Additionally, it performs classification to assign class probabilities to each bounding box, indicating the likelihood of the detected object belonging to different predefined classes (e.g., person, car, dog).

Output Generation: YOLOv8 generates output in the form of bounding boxes along with associated class probabilities and confidence scores. These bounding boxes are then post-processed using techniques such as non-maximum suppression (NMS) to filter out overlapping or redundant and produce the final set of objects .

Training: During the training phase, YOLOv8 learns to detect objects by minimizing a predefined loss function that penalizes errors in bounding box coordinates and classification probabilities. This involves feeding annotated training data into the network, adjusting its parameters (i.e., weights and biases) through , and updating them using optimization algorithms such as stochastic gradient descent (SGD) or Adam.

V. METHODOLOGY

In this study, we utilize the YOLOv8n architecture, an extension of YOLOv8 tailored to our research needs. Implemented through the library, YOLOv8n offers enhanced capabilities for object detection tasks.

We instantiate the YOLOv8n model using “yolov8n.yaml” a configuration file containing architecture specifications and hyperparameters. Training ensues with the “data.yaml” training, iterating for 20 epochs to refine the model's ability to detect and classify objects within images.

For validation, a separate images folder is employed, storing evaluation metrics such as precision, recall, and mean average precision.

Parameters for validation include:

- data: Path to validation configuration file.
- imgsz : Input image size during validation.

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

- batch: Batch size for validation.
- conf: Confidence threshold for filtering.
- iou : Intersection over union (IoU) threshold for non-maximum suppression.
- def: Device (e.g GPU) for validation.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

Training Phase: After training the model on the available dataset, we have achieved significantly good results, as evidenced by the confusion matrix shown below. The confusion matrix provides a comprehensive overview of the model's performance across different classes. From the total images of cars, 93% are correctly detected, indicating a high level of accuracy in car detection. Similarly, the model demonstrates good performance across other object classes as well:

- Accuracy for bicycles is 76%.
- Accuracy for buses is 83%.
- Accuracy for motorbikes is 77%.
- Accuracy for person detection is 68%.

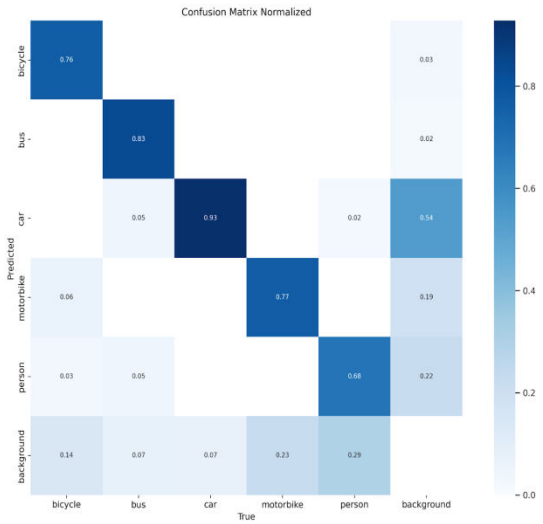

Fig 1.2 Confusion matrix

In the depicted PR curve, the steep initial rise followed by a gradual plateau demonstrates the model's ability to maintain high precision while capturing a significant proportion of positive instances. The calculated AUC-PR value for this curve is around 0.75, indicating excellent overall performance.

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

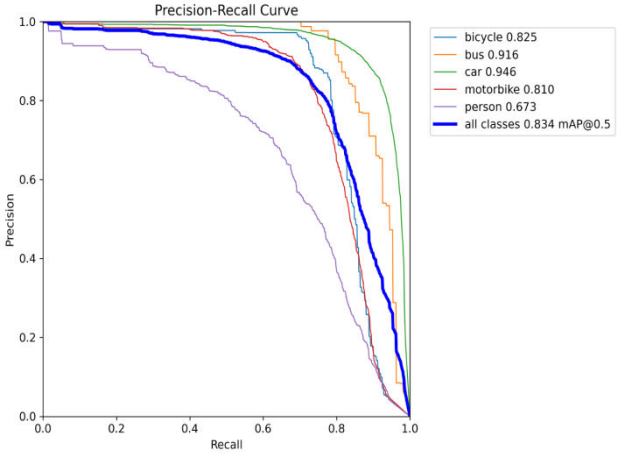

Fig 1.3 PR Curve

Below Image shows how Yolov8 object detection algorithm detected and annotated images with different classes.

Fig 1.4 Accurately Detected Objects

Validation Phase: During the validation phase, hyperparameter tuning is conducted to optimize the model's performance. Parameters such as learning rate, batch size, and regularization strength have a profound impact on the model's effectiveness. By systematically exploring various hyperparameter configurations and evaluating their impact on validation metrics, practitioners can identify the most effective parameter settings.

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

Following the validation process, we observed consistent model performance comparable to the training phase. However, given our limited dataset, future iterations of model evaluation on larger datasets are anticipated to further enhance precision and overall performance.

Testing Phase: After completing model training and hyper parameter tuning, we at last test model performance on image file and .mp4 video file. In both cases it has accurately detected vehicles and annotated them. For reference I have added image 1.5 below.

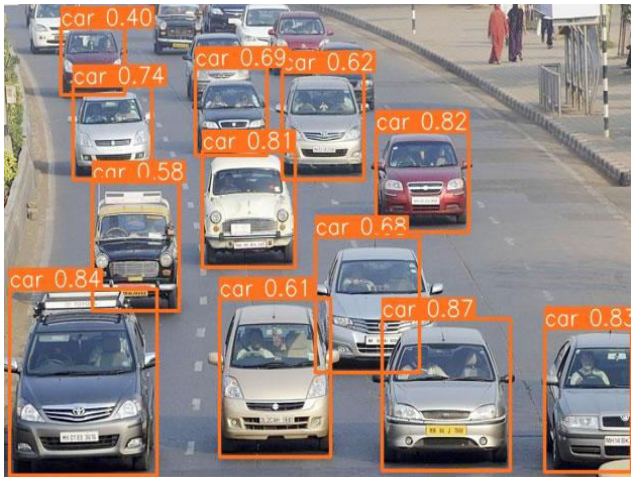

Fig 1.5

VII.CONCLUSION

Based on the experiment conducted, it can be inferred that the YOLO algorithm, relying on Convolutional Neural Networks (CNNs), exhibits remarkable capabilities in detecting various objects within the provided dataset. Despite encountering challenges associated with training the model on a limited dataset and fine-tuning its parameters, the YOLOv8 algorithm demonstrates its effectiveness in accurately detecting objects, thereby enhancing the overall performance of the model.

The implementation of YOLOv8's internal parameters results in notable improvements in object detection accuracy, even when working with sparse data. The model's ability to conduct real-time object detection and tracking is particularly noteworthy, offering the potential to mitigate traffic issues by providing timely insights to traffic signal models. By leveraging the vehicle counts detected by our trained model, the traffic signal models can dynamically adjust signal timings, thereby alleviating congestion in urban areas. This holistic approach not only addresses traffic congestion but also mitigates related challenges, thereby fostering a more efficient and sustainable urban environment.

In conclusion, the utilization of YOLOv8 demonstrates significant potential in tackling urban traffic problems. By harnessing the power of deep learning and real-time object detection, our research contributes to the development of innovative solutions for urban planning and traffic management, ultimately leading to improved quality of life in urban communities.

REFERENCES

[1] Nguyen, H. (2019). Improving Faster R-CNN Framework for Fast Vehicle Detection. Mathematical Problems in Engineering. https://doi.org/10.1155/2019/3808064.
[2] Tang, T., Zhou, S., Deng, Z., Zou, H., & Lei, L. (2017). Vehicle Detection in Aerial Images Based on Region Convolutional Neural Networks and Hard Negative Example Mining. Sensors (Basel, Switzerland), 17. https://doi.org/10.3390/s17020336.

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly Peer Reviewed & Referred Journal |

| Volume 7, Issue 6, June 2024 |

| DOI:10.15680/IJMRSET.2024.0706053 |

[3] Nguyen, D., Nguyen, T., Kim, H., & Lee, H. (2019). A High-Throughput and Power-Efficient FPGA Implementation of YOLO CNN for Object Detection. IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, 27, 1861-1873. https://doi.org/10.1109/TVLSI.2019.2905242.
[4] https://docs.ultralytics.com/
[5] Du, J. (2018). Understanding of Object Detection Based on CNN Family and YOLO. Journal of Physics: Conference Series, 1004. https://doi.org/10.1088/1742-6596/1004/1/012029.
[6] Prasanna, D., Annapurna, C., Yeshwanth, G., Shabina, G., & Tejalingam, P. (2023). Real-Time Object Detection. international journal of food and nutritional sciences. https://doi.org/10.48047/ijfans/v11/i12/207.
[7] Motwani, N., S, S., & Singh, U. (2022). Object Detection and Tracking for Autonomous Vehicles using Deep Learning Technique- YOLO. 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), 1-6. https://doi.org/10.1109/SMARTGENCON56628.2022.10083703.
[8] Prakash, M., Janarthanan, M., & Devi, D. (2023). Multiple Objects Identification for Autonomous Car using YOLO and CNN. 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), 597-601. https://doi.org/10.1109/ICICCS56967.2023.10142751.
[9] https://www.kaggle.com/datasets/yusufberksardoan/traffic-detection-project

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com