Vehicle Detection and Traffic Data Generation

Using YOLOv8 in Metro Manila Highways

A Thesis Presented to the
Faculty of the College of Science
Technological University of the Philippines
Ayala Blvd., Manila

In Partial Fulfillment of the
Requirements for the Degree
Bachelor of Science in Computer Science

Imee Q. Compra

Simon Daniel M. Dela Cruz

Ronan M. Esguerra

Joshuel Ernest Q. Simbulan

Andrew James S. Tejerero

June 2024

INTRODUCTION

The study aimed to develop the project vehicle detection and traffic data generation using the YOLO algorithm. The project?s functionalities were limited to: Vehicle detection, classification, and data generation. The data generated by the application were stored in a database that was publicly available for online queries. The study could help authorities monitor traffic flow and flow in real-time. It could help drivers navigate congested areas, find parking spots, and avoid accidents. The system may benefit students, researchers, and people in the academic community. Vehicle Detection and Traffic Data Generation using YOLOv8 in Metro Manila. Real-world data is needed to verify and validate the accuracy of traffic models relative to the traffic system they simulate. This study enhanced the data gathering of MMDA, traffic engineers, researchers, and even hobbyists by enabling them to gather intensive traffic data with artificial intelligence. This, in turn, can be used as a tool to assist in planning and managing traffic within the scope of what traffic system was simulated. Vehicular Volume Reduction (UVVRP) policy of the Metro Manila Development Authority (MMDA) It is now projected that if no proper solutions are placed, the Philippines will lose ?5.4 billion a day by 2035. YOLO Detection SystemVehicle Detection and Traffic Data Generation using YOLov8 in Metro Manila. The traffic data report should contain information about the detected vehicle, specifically: address, class, date, time, and type (Public Vehicle, Private Vehicle)

METHOD

Vehicle Detection and Traffic Data Generation using YOLOv8 in Metro Manila. The web application was tested using manual testing by the governmental researchers. The generated traffic data was uploaded to the centralized database in MongoDB Atlas. The data was then used to track, detection, and count the vehicles using a live camera. The live camera was connected to the user?s device to display the live video. The video was then uploaded to a live video stream. The study uses YOLOv8 to detect, track, and count vehicles in a video stream. The traffic data is deployed and can be accessed and exported through the RESTful API. The dataconsuming user can view the traffic data using the same web application. This chapter outlines the research methodology of the study, which comprises segments on project design, project development, operational and testing procedures, and the evaluation process. The study was published in the open-source software journal, Cascades. OLO algorithm architecture The relentlessly Simplicity of the framework enables the researchers to set up endpoints for the application with ease. The ?Backbone? is the main body of the Google form that also contains a questionnaire that compasses the ISO 25010 software characteristics. The results were also evaluated using metrics, such as confusion mean average precision (mAP), accuracy, and loss. The web application also allows users to add their own live camera input. , and the parameters such as epochs, image size, Patience, and batch were adjusted. The ?Neck? utilizes ?SPPF? and ?New CSP-PAN? structures to connect the backbone and the head. ?SpPF?s? ?PANs?? ?CSP?S? are used to connect backbone and head, and ?CSPP? is used for ?PAN.?

RESULTS

The hosted API?s page can be visited at (https://traffic-data.onrender.com/) The hosted API can be used to generate traffic data using YOLOv8 in Metro Manila. A screengrab of the first frame captured by the live camera has been displayed alongside the required form fields. The API Explorer page was displayed properly with the form fields, the API calls, and buttons for View and Download CSV. The home page has been accurately displayed and the video page is displayed accurately. Vehicle Detection and Traffic Data Generation using YOLOv8 in Metro Manila. The ?/home? route serves as the landing page of the project ? /video? routes. The /video route displays a page in which users are asked to upload a video. The graph shows the count on the y-axis and the time on the x-axis. The web application allows the connection of a camera to feed the system with video input, which is used for detecting vehicles. Vehicle Detection and Traffic Data Generation using YOLOv8 in Metro Manila. The web application tracks vehicles as they cross the counting line and records the data in a centralized cloud database. The data can be filtered relentlessly with arguments such as address, class (kind of vehicle), date, and type (public or private vehicle) with its query. The API GET request to endpoint /find allows developers to fetch the recorded data from the MongoDB database. To test the API?s proper responses e.g., returning code 200, Postman was employed. The web application can detect vehicles and identify their class. It can also ask users to upload a video file. The data can be exported via a GET request API action /find. It has delivered its intended purpose, which is to ensure that users can expect accurate and reliable service from its functions. The project?s highest rating for a criterion was recorded in ?Functional Suitability,? with a total rating of 3.91, which evaluates to ?Highly Acceptable.?

DISCUSSION

The project titled ?Vehicle Detection and Traffic Data Generation using YOLOv8? successfully executed the test cases set in the paralleledoperational and testing of the system. It includes a summary of the project?s findings, conclusion, and recommendations. The project was developed using the following tools:. The custom-developed Object Detection Model and. The Gitv8 architecture has demonstrated high accuracy in the accurate classification of classes. The system was evaluated by 15 IT experts based on ISO/SEC 25010 in terms of terms of ISO/ SEC 25010. The project allows users to upload and submit a video file for detecting and counting vehicles. It tracks vehicles as they cross the counting line and records the data in a centralized cloud database. Various test cases were executed to test the application?s functionalities, while the API function in which the database can be loaded via an HTTP request was tested using a tool named Postman. The project exhibited excellent reusability, and users could just upload a new video or change the location of the live camera.