YOLO-based Computer Vision Algorithms for Road Traffic Monitoring in an Advanced Navigation System

John Loyd Maispag
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
johnloyd.masipag@tup.edu.ph

Jemimah Aniel Climaco
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
johnloyd.masipag@tup.edu.ph

Khen Kyrill Dado
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
khenkyrill.dado@tup.edu.ph

Jericho Martin Franco
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
jerichomartin.franco@tup.edu.ph

Julius Magayon
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
julius.magayon@tup.edu.ph

Jomel Manalastas
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
jomel.manalastas@tup.edu.p

Romalyn Manansala
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
romalyn.manansala@tup.edu.ph

Justice Colili
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
justice.colili@tup.edu.ph

Andrei Miguel Marquez
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
andreimiguel.marquez@tup.edu.ph

Christian Ceballos
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
christian.ceballos@tup.edu.ph

Angelyn Magsino
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
angelyn.magsino@tup.edu.ph

Carl Justin Viernes
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
carljustin.viernes@tup.edu.ph

***Abstract—* Computer vision was implemented utilizing the YOLOv8 Nano version integrated with ByteTrack, wherein Ultralytics' YOLOv8 was employed with a specialized detector to identify various types of vehicles in the project. The ByteTrack algorithm, in tandem with the detector, processed video sequences, with a threshold value being utilized to refine the classification of detections. The YOLOv8 Ultralytics library leveraged a YOLO-trained model based on the COCO dataset alongside the supervision library for ByteTrack loading. To facilitate live traffic monitoring, images were encoded using Base64, converted into blob images, and streamed to the website for real-time viewing.**

Keywords—YOLOV8 nano, ByteTrack, computer vision, Navigation, Traffic Monitoring, Real-Time Spanning Protocol

I. INTRODUCTION

Traffic congestion has been a severe case in the advancing urban environment of Metro Manila for decades. It worsens yearly as the number of private and public vehicles increases drastically. According to a 2017 study by the Japan International Cooperation Agency, the city incurs losses of approximately P3.5 billion per day due to this ongoing problem [1]. Currently, traffic traffic is managed through traffic lights, signals, and similar methods. However, these issues are addressed by constructing more new highways, making these approaches costly and impractical. The negative result of frequent road construction is constrained by the limited spaces in Urban areas, considering the people residing in the city. The researchers opt for a constant monitoring system among several solutions for road traffic.

Major urban cities in the Philippines have problems with congested roads, resulting in longer travel time, increased fuel consumption, and air pollution. According to Numbeo (a crowd-sourcing online database that deals with consumer prices, accurate property prices, and quality of life metrics), in the 2020 traffic index, the country secured the ninth spot over 81 participating countries. A higher score in the index translates to lower traffic quality for a country. With a score of 198.84 points, the country also topped among members of the ASEAN (Association of Southeast Asian Nations) for having the worst traffic quality.

metro - consuming a large percentage of DOTr's (Department of Transportation) 2024 budget. There is a need for more efficient and effective ways to monitor traffic situations to contribute to accounting for the traffic problem. With the help of modern technology, authorities and drivers can handle traffic situations better. Advanced monitoring and navigation systems offer more flexible and efficient solutions for traffic algorithms.

The system should be able to collect the data by using sensors to classify different categories of vehicles in a traffic flow on the road. Computer vision's primary function is for object detection, detecting the vehicles into digital images while optimizing coordinated plans [2]. YOLO (You Only Look Once) was introduced to the computer vision community in 2015 by the study of Joseph Redmon et al. [3]. It possessed speed and accuracy in its initial testing, predicting multiple bounding boxes and its class possibilities in object detection. In this paper, the latest version, YOL0v8 by Ultralytics [4], was utilized mainly to integrate artificial intelligence to detect vehicles for road monitoring and advanced navigation systems.

A. *The YOLO*

The YOLO (You Only Look Once) deep-learning computer vision algorithm is a well-suited choice in advanced navigation systems due to its speed and accuracy. With the help of YOLO, the navigation system can detect and track vehicles in real time and identify different types of vehicles passing through roads. The latest model in the YOLO family is the Ultralytics YOLOv8. It has "out-of-the-box" support for object detection, classification, and segmentation activities. It is accessible through a Python package and CLI (Command Line Interface). This version offers higher mAP (mean Average Precision) scores than previous versions. It is also faster than its predecessors - suitable for real-time projects.

The YOLOv8 provided five scaled versions: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large), and YOLOv8x (extra-large). Regarding accuracy, COCO (Common Objects in Context - the industry-standard benchmark for evaluating object detection models).

TABLE I: THE TABLE SHOWS THE ACCURACY OF YOLOV8 ON COCO

▾ Detection

Model	size (pixels)	mAP^{val} 50-95	Speed CPU (ms)	Speed T4 GPU (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	-	-	3.2	8.7
YOLOv8s	640	44.9	-	-	11.2	28.6
YOLOv8m	640	50.2	-	-	25.9	78.9
YOLOv8l	640	52.9	-	-	43.7	165.2
YOLOv8x	640	53.9	-	-	68.2	257.8

- **mAP^{val}** values are for single-model single-scale on COCO val2017 dataset.
  Reproduce by `yolo mode=val task=detect data=coco.yaml device=0`
- **Speed** averaged over COCO val images using an Amazon EC2 P4d instance.
  Reproduce by `yolo mode=val task=detect data=coco128.yaml batch=1 device=0/cpu`

Another, RF100 Accuracy, at Roboflow, the team drew 100 sample datasets from Roboflow Universe - a repository of over 100,000 datasets, to evaluate the YOLO models. The box plot below shows the evaluation of YOLOv8, YOLOv5, and YOLOv7.

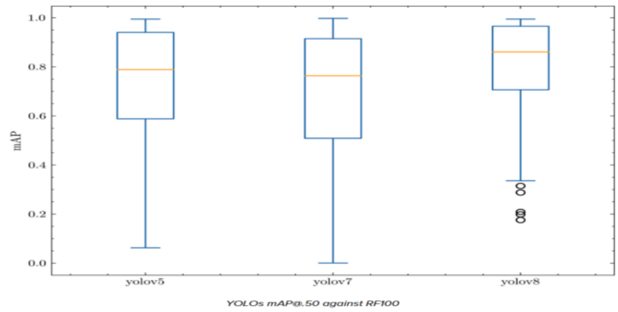

Figure 1: The box plot shows that YOLOv8 had fewer outliers and an overall better mAP when tested against the Roboflow 100 benchmark

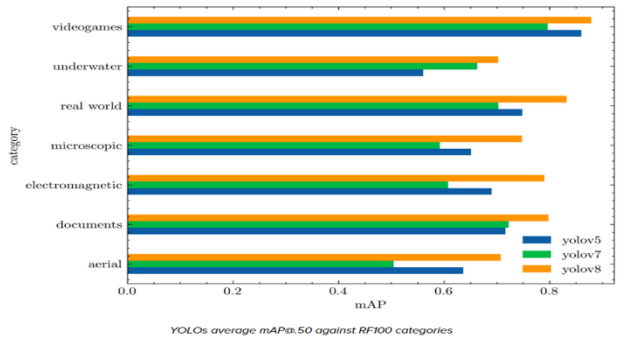

Figure 2: The bar plot tells the average mAP@.50 for each RF100 category - YOLOv8 outperforms the other versions.

The traffic sector is launching road widening and infrastructure projects aiming to address increased traffic in the

B. *The SORT and DeepSORT Algorithm*

YOLOv8 integrates well with SORT and DeepSORT algorithms. Running the leading computer vision algorithm with YOLOv8 on each video frame shows detections (including bounding boxes and labels) for the objects it detected. Then, the detections are fed from YOLOv8 into a separate SORT or DeepSORT algorithm. Tracker Link Detections analyze the detections across consecutive frames and associate them with the same object classifications.

The SORT algorithm relies on Motion Prediction. It utilizes the Kalman Filter to forecast the position of objects in the next frame based on their movement in previous frames. It also uses Bounding Box Overlap (IoU). The IoU (Intersection over Union) measures the overlap between the predicted and detected bounding boxes. Higher IoU suggests a higher chance they represent the same object. One drawback of SORT is that it struggles with occlusions (objects being hidden) and object identity changes (when two objects appear close and have the wrong assigned track).

On the other hand, DeepSORT builds upon the SORT by including a profound learning aspect. This algorithm can extract appearance features from the object detections via a pre-trained neural network. It offers a more robust algorithm as it can account for occlusions and object identity switching. One drawback of DeepSORT is that it is more complex and slower. It needs additional processing and computation power. It can affect the processing speed of the computer vision for real-time traffic monitoring.

II. OBJECTIVES

The objective of this study is to (1) integrate the YOLOv8 deep-learning computer vision algorithm with the SORT (Simple Online and real-time tracking) and DeepSORT (Deep Simple Online and Real-time Tracking) algorithms for real-time vehicle detection and tracking, (2) create an advanced flood and road monitoring system using the algorithms mentioned above for flood-prone areas in the City of Manila.

Integrating the YOLOv8 algorithm with SORT and DeepSORT algorithms serves a crucial purpose in enhancing the functionality and accuracy of the proposed flood and road monitoring system. YOLOv8, known for its speed and accuracy in object detection tasks, is utilized to detect and track vehicles in real time, providing valuable data for traffic monitoring and management within flood-prone areas. The latest version of YOLOv8 offers improved performance metrics, making it an ideal choice for real-time projects.

In this study, YOLOv8 is capable in detecting and classifying vehicles passing through roads, contributing to the system's ability to provide real-time traffic information to users. Additionally, the integration of SORT and DeepSORT algorithms allows for robust tracking of detected vehicles across consecutive frames, enabling the system to effectively analyze traffic flow patterns and identify potential congestion areas. The SORT algorithm utilizes motion prediction techniques, such as the Kalman Filter, to forecast the position of objects in the next frame based on their previous movement. On the other hand, DeepSORT incorporates deep learning features to extract appearance features from object detections, offering improved performance in handling occlusions and object identity switching.

The Real-Time Streaming Protocol (RTSP) is utilized to facilitate communication and control between the system and IP cameras for live video feed analysis. This protocol enables the system to access and process video streams from IP cameras in real-time, providing input data for the YOLOv8 algorithm to perform vehicle detection and tracking.

The integration of YOLOv8 with SORT and DeepSORT algorithms, along with the utilization of RTSP for live video feed processing, enhances the flood and road monitoring system's capability to accurately detect and track vehicles in real-time, thereby facilitating effective traffic management and navigation assistance in flood-prone areas of the City of Manila.

III. REVIEW OF RELATED LITERATURE

According to M. A. Al-qaness. et al. (2021), the suggested system combines image-based tracking, neural network architecture, and the You Only Look Once (YOLOv3) algorithm for tracking moving vehicles. The researchers trained the system using a variety of datasets. In order to evaluate the effectiveness of the suggested approach, they also exposed it to actual video clips showing traffic on the road. According to the evaluation results, the suggested system successfully detects, tracks, and counts vehicles even in dynamic scenarios. [5]

Moreover, the study by Li, H., Xu, C., and Luo, S. (2019) found that their suggested detection technique, which uses k-fold cross-validation and preprocessed datasets based on YOLOv3, performs better than regular YOLOv3, particularly in situations when there is a class imbalance. Furthermore, they claim that the Dense-YOLOv3 network structure reduces overfitting, improving the system's overall robustness and performance for unmanned vehicle applications [6].

Moreover, the study by Li, H., Xu, C., and Luo, S. (2019) found that their suggested detection technique, which uses k-fold cross-validation and preprocessed datasets based on YOLOv3, performs better than regular YOLOv3, particularly in situations when there is a class imbalance. Furthermore, they claim that the Dense-YOLOv3 network structure reduces overfitting, improving the system's overall robustness and performance for unmanned vehicle applications [6].

In another study by B. Arunmozhi [7], YOLOv8 simplifies counting and tracking vehicles with its advanced capability in object detection and tracking. YOLOv8's sophisticated object detection and tracking features render it a valuable asset for traffic monitoring and management. Its capacity to precisely and swiftly track vehicle speed and identify number plates holds substantial implications for law enforcement, traffic control, and public safety.

Most object detection studies are performed in the visible spectrum (RGB) domain. However, these images captured by the camera are affected by natural sunlight and low and high illumination conditions, causing work limitations. Infrared (IR) cameras can capture images with infrared radiation which is not visible to the human eye. It improves object detection and monitoring of vehicles under weather conditions. Significant progress is recognized in deep learning techniques for object detection in the visible spectrum domain. However, there is still a need to improve the detection of objects in the infrared spectrum. [8] Previous studies concluded that IR cameras performed well in public IR datasets using object detection methods such as FLIR ADAS [9] and KAIST Multispectral Pedestrian [10]. IR object detection architecture was implemented with the YOLO variant (YOLOR-P6), considering different parts of the image when detecting objects in real-time processing of vehicle detection [11].

In the study by L. Jianbing and C. Shuhui [11], a series of consecutive frames extracted from a video or image stream undergoes analysis, wherein the trained YOLOv8 model is applied to detect the presence and location of cars within each frame. Bounding boxes are generated around identified cars in each frame to delineate their spatial extent. Subsequently, tracking techniques are employed to monitor the movement of identified cars across successive frames. The Kalman filter, a recursive mathematical algorithm, is utilized to estimate the state of each tracked car. Additionally, the DeepSORT (Deep Simple Online and Real-time Tracking) algorithm is employed for object tracking, thereby enhancing tracking accuracy by associating detections with existing tracks and managing identity switches. Either the Kalman filter or the DeepSORT algorithm is selected as the preferred tracking technique to ensure continuity in tracking.

IV. METHODOLOGY

The FRENS system used the Yolov8 algorithm for computer vision that enabled the system to have data for geo-mapping for the users. Here are the sub processes of computer vision. First, the outdoor IR Bullet IP network camera served as the eye for road monitoring. The camera provided real-time footage of a specific road segment. This footage served as a dynamic input for processing and analyzing. Next, the board processing computers such as Raspberry Pi received the input from the camera via Real-Time Spanning Protocol (RTSP) on port 554.

The computer vision algorithm inside the board processed the input and yielded the total count for vehicle bounding in and out, and the count for different vehicles. Another, in the database, a PHP file called receive_json.ph served as the storage and transmitter of data sent by the Raspberry Pi connected to the Internet. Also, another PHP file called receive_blob.ph served as the translator of binary data into sequences of printable characters for live viewing of road traffic.

For the data visualization on the website, color scheme conditions like Green (Free Flow), Yellow (light), Orange (Moderate), and Red (Heavy) derived the Estimated Time of Arrival of users and road availability in addition to the data produced by flood sensors. The map finds rerouting options to lessen the journey of the drivers.

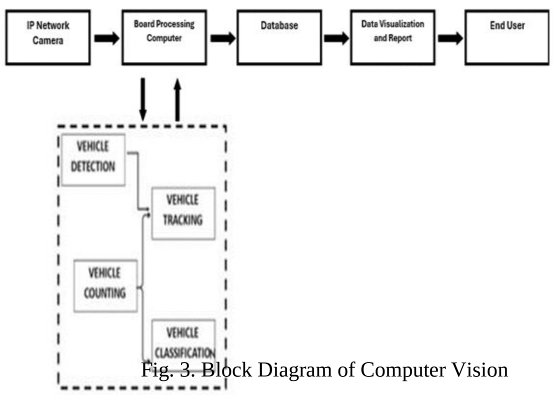

Fig. 3. Block Diagram of Computer Vision

V. RESULTS AND DISCUSSION

A. *System Integration and Testing*

TABLE II. AVERAGE PING (IN MS) OF THE IP CAMERAS FOR 5 DAYS – SPEED METRIC

05/21-25/2024	Camera_2	Camera_5	Camera_9
08:00 AM – 09:00 AM	23.19	31.67	18.52
11:00 AM – 12:00 PM	28.71	35.03	21.07
02:00 PM – 03:00 PM	30.13	37.94	22.46
05:00 PM – 06:00 PM	25.87	29.11	17.79
08:00 PM – 09:00 PM	26.69	30.25	18.36

Table II shows the average ping times, measured in milliseconds (ms), for three different cameras in the system. The average ping time is a critical metric, reflecting the responsiveness and efficiency of the network communication between the cameras and the central monitoring system. Lower ping times indicate faster data transmission, which is essential for real-time monitoring and timely decision-making. This figure provides a comparative analysis of the network performance for Cameras 2, 5, and 9, highlighting any potential latency issues and ensuring optimal functioning of the traffic monitoring system.

TABLE III AVERAGE INTERNET SPEED (IN MBPS) OF THE DEPLOYED ROUTERS FOR 5 DAYS – SPEED METRIC

05/21-25/2024	Router_2	Router_5	Router_9
08:00 AM – 09:00 AM	46	23	50
11:00 AM – 12:00 PM	33	18	44
02:00 PM – 03:00 PM	35	15	47
05:00 PM – 06:00 PM	30	26	39
08:00 PM – 09:00 PM	31	24	37

Table III presents the average internet speeds, measured in megabits per second (Mbps), for Routers 2, 5, and 9. This metric is essential for assessing the data transfer capabilities and overall performance of the network. Higher internet speeds indicate more efficient data transmission and quicker communication within the system. The figure provides a comparative analysis of the internet speeds for these three routers, identifying any performance variations and ensuring the network infrastructure is capable of supporting real-time traffic monitoring and data processing needs.

TABLE IV. AVERAGE PING (IN MS) OF THE PROCESSING COMPUTERS FOR 5 DAYS – SPEED METRIC

05/21-25/2024	PC_1	PC_2	PC_3
08:00 AM – 09:00 AM	6.12	8.16	10.0
11:00 AM – 12:00 PM	9.65	13.88	8.33
02:00 PM – 03:00 PM	10.12	15.91	9.47
05:00 PM – 06:00 PM	11.96	18.53	12.36
08:00 PM – 09:00 PM	13.04	13.19	9.01

Table IV shows the average ping times, measured in milliseconds (ms), for three different PCs in the network: PCs 2, 5, and 9. The average ping time is a critical measure of network latency, indicating the speed at which data packets travel between the PCs and the central server. Lower ping times signify faster and more efficient communication, which is vital for real-time applications. This figure provides a comparative analysis of the network performance for PCs 2, 5, and 9, highlighting any latency issues and ensuring optimal system responsiveness.

TABLE V. AVERAGE ACTUAL VEHICLE COUNT (IN | OUT) FOR 5 MINUTES.

05/21-25/2024	Traffic_2	Traffic_5	Traffic_9 (one – way)
08:00 AM – 09:00 AM	140 | 124	85 | 96	52
11:00 AM – 12:00 PM	136 | 105	91 | 87	61
02:00 PM – 03:00 PM	132 | 96	92 | 70	47
05:00 PM – 06:00 PM	178 | 156	110 | 109	63
08:00 PM – 09:00 PM	104 | 85	90 | 84	45

Table V illustrates the actual vehicle count over a 5-minute interval, detailing the number of vehicles entering (Count In) and exiting (Count Out) the monitored areas by three different cameras: Cameras 2, 5, and 9. The data reveals that between 5 and 6 pm, all three cameras detected a higher volume of vehicles, indicating this period as the peak of rush hour when many people are arriving home. This figure provides insight into traffic flow and density, capturing the effectiveness of each camera in monitoring and recording vehicular movements. By comparing the vehicle counts from these three cameras, the figure highlights traffic patterns and potential discrepancies in data, ensuring accurate and reliable traffic monitoring and management.

VI. CONCLUSION

The application has successfully addressed the challenges involved in implementing and integrating a flood and road monitoring system that utilizes computer vision and flood sensors with 92% accuracy through the utilization of advanced technologies, significant progress has been made towards achieving an efficient road monitoring system.

The camera utilized for vehicle detection has been able to count the number of vehicles within its field of view. This count serves as a classification metric, providing insights into the current traffic conditions at a specific node. Through computer vision, it can now classify and estimate additional travel time based on the number of vehicles detected.

The computer vision system for vehicle detection, utilizing IP cameras, has demonstrated promising performance with an 80% accuracy rate, enabling quick and reliable identification of vehicles. This enhances the overall monitoring and management of traffic conditions in the area, ensuring the provision of the best routes for users.

This project makes a significant contribution to the field of road monitoring systems. The developed system has demonstrated its potential to enhance transportation systems facing challenges posed by floods. The findings and recommendations presented here lay the foundation for future research and development in the domain of flood and road systems.

VII. RECOMMENDATION

The Flood and Road Eye Navigation System (FRENS) has demonstrated significant potential in providing real-time flood and traffic monitoring for the City of Manila. This research recommendation aims to explore and propose future endeavors and innovations for the enhancement and expansion of FRENS.

1. Explore the utilization of satellite APIs to enhance the precision of routing plans, particularly in addressing flood and traffic conditions on the map.
2. Consider expanding the deployment area to cover a broader geographical scope, ensuring a more comprehensive and inclusive flood and traffic monitoring system.
3. Develop a more robust and flexible database infrastructure along with an administrative site capable of efficiently handling a substantial volume of data from the deployed nodes and cameras.

REFERENCES

[1] B. O. de Vera, “JICA: Traffic congestion now costs P3.5 billion a day,” INQUIRER.net, https://newsinfo.inquirer.net/970553/jica-traffic-congestion-now-costs-p3-5-billion-a-day-metro-manila-traffic-jica-cost-of-traffic (accessed Jun. 14, 2024).

[2] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, “Object detection in 20 years: A survey,” Proceedings of the IEEE, vol. 111, no. 3, pp. 257–276, Mar. 2023. doi:10.1109/jproc.2023.3238524

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016. doi:10.1109/cvpr.2016.91

[4] Ultralytics, “Ultralytics/ultralytics: New - yolov8 in PyTorch > ONNX > OpenVINO > CoreML > TFLite,” GitHub, https://github.com/ultralytics/ultralytics (accessed Jun. 14, 2024).

[5] M. A. Al-qaness et al., “An improved YOLO-based Road Traffic Monitoring System,” Computing, vol. 103, no. 2, pp. 211–230, Jan. 2021. doi:10.1007/s00607-020-00869-8

[6] S. Luo, C. Xu, and H. Li, “An application of object detection based on yolov3 in traffic,” Proceedings of the 2019 International Conference on Image, Video and Signal Processing, Feb. 2019. doi:10.1145/3317640.3317657

[7] Vehicle Detection and Speed Tracking System Using YOLO v8. doi:https://ijarbest.com/journal/V9i5/2282

[8] N. Bustos et al., A systematic literature review on object detection using near infrared and thermal images, 2023. doi:10.2139/ssrn.4474503

[9] "Free Teledyne FLIR thermal dataset for algorithm training," FREE - FLIR Thermal Dataset for Algorithm Training | Teledyne FLIR, https://www.flir.com/oem/adas/adas-dataset-form/ (accessed Jun. 14, 2024).

[10] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and Baseline," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2015. doi:10.1109/cvpr.2015.7298706

[11] (PDF) vehicle tracking based on an improved DeepSORT algorithm and the Yolov4 Framework, https://www.researchgate.net/publication/356137079_Vehicle_Tracking_based_on_an_Improved_DeepSORT_Algorithm_and_the_YOLOv4_Framework (accessed Jun. 14, 2024).