

# Portable system for real-time traffic volume and speed estimation using YOLOv10

Ida Bagus Sradha Nanda, Masrono Yugihartiman, Eko Primadi Hendri, I Made Suartika

Indonesian Land Transportation Polytechnic-STTD, Bekasi, Indonesia

## Article Info

### Article history:

Received Jun 6, 2025

Revised Oct 24, 2025

Accepted Jan 22, 2026

### Keywords:

Object detection

Portable system

Real-time AI

Traffic volume

Vehicle speed estimation

YOLOv10

## ABSTRACT

Accurate traffic data is essential for effective transportation planning and policymaking. However, in many regions, especially those lacking intelligent infrastructure, data collection remains dependent on manual methods that are labor-intensive, time-consuming, and susceptible to human error. While advanced systems such as closed-circuit television (CCTV) and area traffic control systems (ATCS) offer automation, their high cost and infrastructure requirements limit widespread adoption. This study proposes a portable, low-cost, and real-time traffic monitoring system based on the YOLOv10 object detection algorithm. The system operates using only a smartphone-grade camera (1080 p, 60 fps) and a standard laptop, eliminating the need for expensive installations. It detects, classifies, and counts vehicles as they pass through a predefined region of interest (ROI), and also estimates their speed based on time–distance measurements. Field evaluations using five one-hour urban traffic videos showed excellent agreement with manual counts, achieving a mean absolute percentage error (MAPE) of just 0.30%. Speed estimation trials conducted on sample clips also demonstrated consistent and plausible results. These findings highlight the system's potential as a scalable and accurate alternative for traffic monitoring in infrastructure-limited environments.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## Corresponding Author:

Ida Bagus Sradha Nanda

Indonesian Land Transportation Polytechnic-STTD

89 Setu Highway, Bekasi, Indonesia

Email: [gussradha@gmail.com](mailto:gussradha@gmail.com)

## 1. INTRODUCTION

Accurate and timely traffic data is a critical component of modern urban transportation planning, traffic engineering, and policymaking. Among various metrics, traffic volume remains one of the most fundamental, influencing key decisions related to road capacity, signal timing, congestion mitigation strategies, and safety interventions [1]. In many developing countries, however, traffic volume data is still obtained through manual counting methods. These techniques, while straightforward, are labor-intensive, time-consuming, and prone to human error [2]. Semi-automated frameworks using CCTV-based multi-class vehicle counting have also been proposed [3], yet their deployment remains dependent on costly fixed infrastructure, limiting applicability in resource-constrained settings.

To address these limitations, many urban areas have adopted automated traffic monitoring systems such as closed-circuit television (CCTV) and area traffic control systems (ATCS) [4]. While these technologies can be effective, their implementation often demands substantial infrastructure, financial investment, and specialized technical knowledge—factors that can hinder adoption in smaller municipalities or infrastructure-constrained regions. Alternative approaches, such as Doppler radar, provide low-cost

sensing solutions but lack the ability to classify vehicles or estimate speed [5]. This highlights the need for solutions that are not only accurate, but also cost-effective, portable, and easy to deploy.

Recent advances in artificial intelligence (AI) and computer vision have introduced new opportunities for automating traffic data collection without relying on expensive hardware. The you only look once (YOLO) family of object detection algorithms has become a leading framework for real-time applications due to its efficiency and robust performance [6]. The latest version, YOLOv10, further improves detection accuracy and computational efficiency, enabling deployment on consumer-grade devices and making it well-suited for edge-based traffic monitoring [7].

Several studies have explored YOLO-based implementations for traffic monitoring. For instance, a mobile application utilizing YOLOv8 achieved 93% accuracy in vehicle detection and counting using a smartphone camera [8]. Similarly, drone-assisted urban monitoring with an improved YOLOv8 model demonstrated robust detection of vehicles and pedestrians in complex traffic conditions [9]. While these studies confirm the feasibility of portable traffic analytics, they are typically limited to single functions (such as counting) or require specialized platforms such as unmanned aerial vehicles (UAVs). Comparative work has also shown that YOLOv10 consistently outperforms YOLOv8 in detection robustness [10]. Meanwhile, research on vision-based vehicle speed estimation confirms that reliable results can be achieved through camera-based systems [11]. However, most existing approaches address either traffic volume or speed in isolation, rather than integrating both functions in a unified, low-cost framework.

Table 1 summarizes representative studies in this domain, comparing their approaches, hardware requirements, accuracy, cost, and deployment complexity. As shown, existing solutions either rely on costly infrastructure (CCTV, ATCS), require specialized devices (radar, drones), or remain functionally limited to either volume or speed estimation. To the best of our knowledge, no prior work has demonstrated a fully portable, low-cost system capable of simultaneously estimating both traffic volume and vehicle speed using only a smartphone-grade camera and a laptop computer. This study addresses that gap by developing and validating such a system under real-world urban traffic conditions.

The contributions of this research are threefold. First, it proposes a low-cost and portable system for automated traffic volume monitoring using YOLOv10. Second, it integrates vehicle speed estimation into the same real-time detection framework. Third, it validates the system's accuracy against manual counting benchmarks using one-hour traffic video recordings from an urban arterial. By demonstrating feasibility under non-ideal, real-world conditions, this study contributes to the growing body of research on accessible, AI-driven transportation solutions.

Table 1. Comparative overview of traffic monitoring system

Reference	Approach	Hardware/Setup	Accuracy	Cost/deployment complexity	Limitation
Peppas <i>et al.</i> [4], 2021	CCTV-based detection and prediction	Fixed CCTV+servers	High	High cost, complex	Not portable
Naidoo <i>et al.</i> [5], 2025	Low-cost Doppler radar	Radar units	Reliable	Moderate cost, medium setup	No classification, no speed
Charef <i>et al.</i> [8], 2025	YOLOv8 mobile app	Smartphone camera	~93%	Low cost, simple	Only counts vehicles
Dou <i>et al.</i> [9], 2025	Drone-based YOLOv8	UAV camera	High	High cost, complex	Limited flight duration
Geetha <i>et al.</i> [10], 2024	YOLOv8 vs YOLOv10 benchmark	Datasets, standard hardware	YOLOv10>YOLOv8	Low (simulation)	Not applied to traffic monitoring
Macko <i>et al.</i> [11], 2025	Vision-based speed estimation	Camera+vision algorithms	Consistent	Moderate	Speed only, no volume
This study	Portable YOLOv10 system	Smartphone (1080 p, 60 fps)+PC (6 Gb VRAM)	Volume: MAPE 0.30%, Speed consistent with norms	Very low cost, easy to deploy	Needs validation under adverse weather/low-light

## 2. METHOD

This section outlines the complete workflow for developing and evaluating the proposed portable system for traffic volume and speed estimation based on YOLOv10. The methodology encompasses system design, vehicle detection and classification. It also includes traffic volume computation, speed estimation, and statistical validation.

### 2.1. System overview and tools

The proposed system is designed for deployment in low-resource environments and was developed using Python as the primary programming language, supported by a suite of open-source libraries to facilitate

object detection, video processing, and data export [6]. At its core, the system employs YOLOv10 from Ultralytics as the main detection engine. This model processes each frame of input video to perform real-time object detection and classification. YOLOv10 has demonstrated significant improvements over previous versions in terms of both detection accuracy and computational efficiency, which makes it particularly suitable for edge computing and real-time processing on standard hardware [7], [10].

For video input handling and frame processing, the system utilizes OpenCV—one of the most established libraries in the field of computer vision. OpenCV offers robust functionalities for image manipulation and has been widely adopted in real-time video analytics applications [12], [13]. To simplify the implementation of region of interest (ROI) logic and vehicle crossing detection, the system integrates the cvzone library. Built upon OpenCV and MediaPipe, cvzone provides high-level abstractions that enable rapid prototyping and streamlined development of vision-based applications [14].

The system’s output is managed through integration with xlwings, a library that enables seamless interaction between Python and Microsoft Excel. This allows traffic volume and speed estimation results to be exported automatically into spreadsheet format for further analysis or reporting. xlwings has been particularly useful for lightweight applications that require programmatic Excel manipulation [15].

System testing was conducted using video footage recorded on Gajah Mada street, Klungkung Regency, with a consumer-grade smartphone camera capable of Full HD (1080 p) resolution at 60 frames per second. The camera was mounted on a stable platform approximately 5 meters above ground level and 5 meters from the roadway, with a tilt angle of about 45° to ensure consistent coverage of the traffic stream. All processing was executed on a standard laptop equipped with a GPU with at least 4 GB of VRAM, ensuring reliable performance while maintaining portability for field deployment. A sample frame from the recorded video is presented in section 3, illustrating the typical visual input used in this study.

The overall workflow of the system is illustrated in Figure 1, which presents the step-by-step algorithmic flow from video input to data output. In addition, Figure 2 presents the system architecture diagram. It highlights the interaction between hardware, software, and data flow components in a modular structure to ensure reproducibility and clarity of the experimental setup for proposed traffic monitoring system.

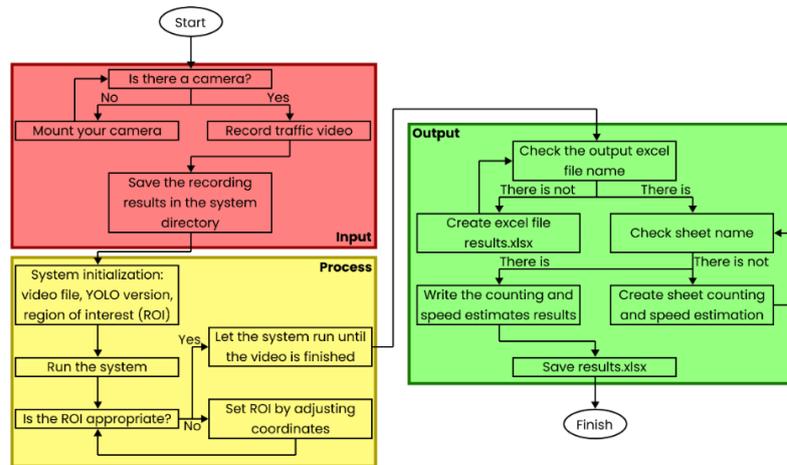


Figure 1. System workflow from video input, detection, ROI-based tracking, to traffic volume and speed estimation output

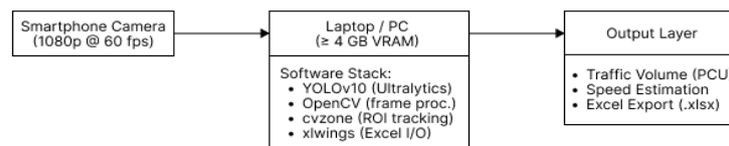


Figure 2. System architecture diagram

## 2.2. Vehicle detection and classification

The YOLOv10 model was trained and configured to detect three categories of vehicles pertinent to traffic engineering: motorcycles, light vehicles, and heavy vehicles. Classification relied on relative object

size and shape, aligned with methodologies in prior YOLO-based traffic studies [16]. Each detected vehicle was assigned a passenger car unit (PCU) value in accordance with Indonesian traffic standards: 0.25 for motorcycles, 1.00 for light vehicles, and 1.20 for heavy vehicles [17]. The application of PCU values facilitates the standardization of heterogeneous traffic flows, a common practice in traffic engineering to account for varying vehicle types [18]. The assigned PCU values are summarized in Table 2.

Regarding detection reliability under real-world conditions, the system adopts a logic comparable to human surveyors. Vehicles are only counted when their centroids fully cross the ROI, which prevents double counting during partial occlusions or overlaps. In cases of complete occlusion where a vehicle is not visible to either the human eye or the camera, the system like a manual survey cannot record the object. However, when only a portion of the vehicle remains visible, the YOLOv10 model is still capable of detecting and classifying the object based on trained features.

Lighting conditions are managed in similar way. Automatic exposure settings of camera are generally sufficient to ensure that vehicles remain visible for detection. As long as the shape of the object remains discernible, the system performs reliably. However, in extreme low-light or glare conditions where object contours become indistinct, detection accuracy decreases. In such cases, manual adjustment of camera exposure or the use of cameras with superior low-light sensitivity would be required to maintain optimal performance.

Table 2. PCU equivalence table

Vehicle type	PCU value
Motorcycle	0.25
Light vehicle	1.00
Heavy vehicle	1.20

### 2.3. Volume logging mechanism

To align with standard traffic survey practices, the proposed system was configured to record vehicle data in 15-minute intervals, a convention widely adopted in traffic engineering to capture temporal variations in traffic flow [19]. This interval not only provides sufficient granularity for analyzing short-term fluctuations but also facilitates aggregation into hourly or daily volumes for broader planning purposes. For instance, large-scale sensor networks and high-resolution traffic control data streams have demonstrated the effectiveness of 15-minute interval logging in revealing detailed urban traffic dynamics [20], and similar interval conventions are used in studies of traffic characteristics and planning [19], [21].

In practice, every vehicle crossing the ROI is detected, classified, and assigned a PCU value to standardize heterogeneous traffic streams. The accumulated data within each interval are then automatically exported to Microsoft Excel via the `xlwings` library. The exported dataset includes vehicle counts per class, PCU-equivalent totals, and timestamps, thereby maintaining a structured and analyzable record.

This automated logging mechanism mirrors the structure of manual traffic surveys while eliminating the potential for human error. By preserving compatibility with conventional formats used in transportation planning, the system enables direct validation against manual counting methods. It also supports integration with established traffic analysis frameworks—including approaches that combine automated counters with crowdsourced traffic data for improved volume-delay modeling [22].

### 2.4. Speed estimation module

The speed estimation module was developed to operate directly on video frames without relying on external sensors such as GPS or radar. Vehicle motion was quantified by tracking the centroid of each detected object across consecutive frames. The displacement in pixels was calculated using the Euclidean distance [23] as in (1).

$$\Delta p = \sqrt{(x_{t+\Delta f} - x_t)^2 + (y_{t+\Delta f} - y_t)^2} \quad (1)$$

Where  $(x_t, y_t)$  and  $(x_{t+\Delta f}, y_{t+\Delta f})$  represent the centroid coordinates of a vehicle at frame  $t$  and at a later frame  $t + \Delta f$ . In this study,  $\Delta f = 60 \text{ frames}$ , corresponding to a one-second interval at 60 fps. This one-second window ensured that displacements were sufficiently large to mitigate pixel-level noise while maintaining real-time responsiveness.

To convert pixel displacement into real-world distance, a scale factor  $S$  (*meter/pixel*) was derived from a ground reference visible in the video frame, such as lane width or road marking of known physical length as in (2).

$$S = \frac{L_{real}}{L_{pixel}} \quad (2)$$

The actual traveled distance in meters was then calculated as (3).

$$d_m = \Delta p \times S \quad (3)$$

Given the time interval  $\Delta t = \frac{\Delta f}{fps}$ , the vehicle's speed was computed as (4).

$$v_{m/s} = \frac{d_m}{\Delta t}, \quad v_{km/h} = v_{m/s} \times 3,6 \quad (4)$$

Since  $\Delta t = 1$  s under the 60-fps configuration, the computation simplified to (5).

$$v_{km/h} = d_m \times 3,6 \quad (5)$$

The overall logic of the speed estimation process is illustrated in Figure 3. The system begins by detecting vehicles and extracting their centroids. Each vehicle is assigned a unique tracking ID, which is monitored across consecutive frames. Once the interval of  $\Delta f = 60$  frames is reached, the centroid displacement is computed using (1). This pixel displacement is then converted into real distance using the scale factor (2) and (3), followed by speed calculation (4) and (5). Finally, the estimated speed is recorded for each vehicle ID.

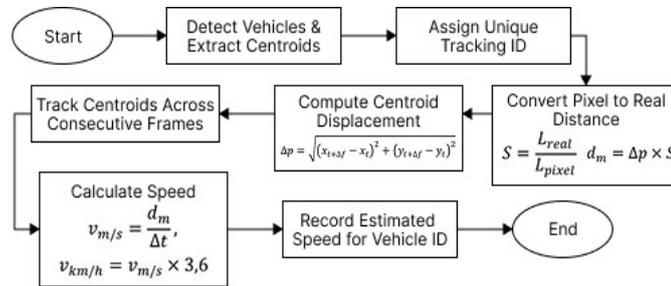


Figure 3. Flowchart speed estimation module

## 2.5. Accuracy evaluation

The accuracy of the proposed system was evaluated by comparing its outputs with manual traffic counts, which served as the ground truth. Three primary indicators were used: precision, recall, and F1-score, which are widely applied in computer vision and traffic monitoring studies to assess classification and detection performance [24]. Precision ( $P$ ) measures the proportion of correctly identified vehicles among all detections, recall ( $R$ ) quantifies the proportion of correctly detected vehicles relative to the total number of vehicles present, and the F1-score represents the harmonic mean of precision and recall, providing a balanced performance metric as in (6).

$$P = \frac{TP}{TP+FP}, \quad R = \frac{TP}{TP+FN}, \quad F1 = \frac{2 \times P \times R}{P+R} \quad (6)$$

Where  $TP$  is true positives (vehicles correctly detected),  $FP$  is false positives (incorrect detections), and  $FN$  is false negatives (missed detections).

To evaluate traffic volume estimation, the mean absolute percentage error (MAPE) was employed to measure the relative deviation of the automated system from the manual ground truth as in (7).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{A_i - F_i}{A_i} \right| \times 100\% \quad (7)$$

Where  $A_i$  represents the manual (ground truth) count and  $F_i$  the automatic count for sample  $i$  [25].

To provide statistical rigor, a 95% confidence interval ( $CI$ ) was calculated for the error values to quantify uncertainty [26]. The  $CI$  was derived using the standard error ( $SE$ ) across the five video samples as (8).

$$SE = \frac{SD}{\sqrt{n}}, \quad CI_{95\%} = \bar{E} \pm t_{0.025, n-1} \times SE \quad (8)$$

Where  $SD$  is standard deviation of the errors,  $n$  is the number of samples,  $\bar{E}$  is the mean error (e.g., MAPE), and  $t_{0.025, n-1}$  is the critical value from the student's t-distribution for a two-tailed 95% confidence level.

All computations were performed using the paired dataset of manual and automatic counts across five observation samples, ensuring consistency with standard practices in traffic engineering and computer vision benchmarking [27].

### 3. RESULTS AND DISCUSSION

#### 3.1. Traffic volume estimation

The developed system was evaluated using five one-hour video recordings captured at a fixed location along Gajah Mada street, Klungkung Regency. Each video reflected real-world traffic conditions over different periods of the day. Detection and classification were performed using YOLOv10, categorizing vehicles into motorcycles (MC), light vehicles (LV), and heavy vehicles (HV).

Figure 4 illustrates the system interface during real-time traffic volume estimation. Detected vehicles are enclosed within bounding boxes labeled by class and confidence score and tracked as they pass through the ROI. The cumulative class-based traffic counts appear in the top-left corner, providing direct visual validation of detection, tracking, and counting performance addressing reviewer requests for operational clarity.



Figure 4. System display for traffic volume automatic count

Manual vehicle counts were conducted by trained surveyors using standard traffic survey practices, serving as ground truth for benchmarking. Tables 3 and 4 present the results of manual and automated counts, respectively, including unit counts and equivalent PCU. To quantify accuracy across vehicle categories, Table 5 summarizes precision, recall, F1-score, MAPE (%), and 95% confidence intervals for each class. Metrics were calculated by aggregating data across all five samples.

Table 3. Traffic volume from manual count

Sample	Light vehicle		Heavy vehicle		Motorcycle		Traffic volume
	Units	PCU	Units	PCU	Units	PCU	
1	186	186	65	78	1,676	419	683.00
2	85	85	85	102	1,280	320	507.00
3	110	110	72	86.4	1,278	319.5	515.90
4	230	228	91	109.2	2,201	550.25	887.45
5	231	231	120	144	1,728	434.5	809.50

Table 4. Traffic volume from automatic count

Sample	Light vehicle		Heavy vehicle		Motorcycle		Traffic volume
	Units	PCU	Units	PCU	Units	PCU	
1	192	192	65	78	1,650	412.5	682.50
2	90	90	87	104.4	1,260	315	509.40
3	120	120	71	85.2	1,253	313.25	518.45
4	241	241	92	110.4	2,164	541	892.40
5	239	239	118	141.6	1,710	427.5	808.10

Table 5. Performance metrics of automatic traffic volume estimation by vehicle type

Vehicle type	Precision	Recall	F1-Score	MAPE (%)	95% CI (MAPE)
Light vehicle (LV)	0.955	1.000	0.977	5.29	3.88-6.70
Heavy vehicle (HV)	0.993	0.993	0.993	1.30	0.62-1.98
Motorcycle (MC)	1.000	0.985	0.992	1.56	0.91-2.21
Overall (micro-avg)	0.995	0.986	0.991	0.92	0.47-1.37

These results demonstrate high accuracy across all vehicle categories. Light vehicles show slightly higher proportional error, likely due to class heterogeneity. Overall MAPE remains under 1% with a tight confidence interval, indicating consistent system performance. These outcomes align with emerging smart traffic systems reporting sub-5% MAPE as acceptable for urban deployment [4].

Lastly, a statistical test of the mean volume difference between manual and automated counts returned a 95% CI of difference including zero (e.g.,  $-3.7$  to  $+0.3$  PCU). This confirms no statistically significant bias between methods. Taken together, these findings validate the proposed YOLOv10-based system as a reliable, cost-effective alternative to traditional sensor-based traffic monitoring consistent with trends in emerging AI-powered ITS deployments [28], [29].

### 3.2. Speed estimation trials

The system's speed estimation module was evaluated using the same set of five video recordings employed in traffic volume analysis. As detailed in section 2.5, vehicle speed was derived by tracking the centroid displacement of each detected object over a one-second interval (equivalent to 60 frames at 60 fps). The pixel displacement was then converted into real-world distance using a scaling factor based on the known length of lane markers within the ROI, and the resulting value was converted into kilometers per hour by multiplying with 3.6.

Figure 5 presents an exemplary system output, where each detected vehicle is annotated with a bounding box, class label, confidence score, and real-time estimated speed. This visual overlay underscores the system's integrated capability for simultaneous detection, classification, and speed estimation. The average estimated speeds obtained across the five samples are summarized in Table 6.

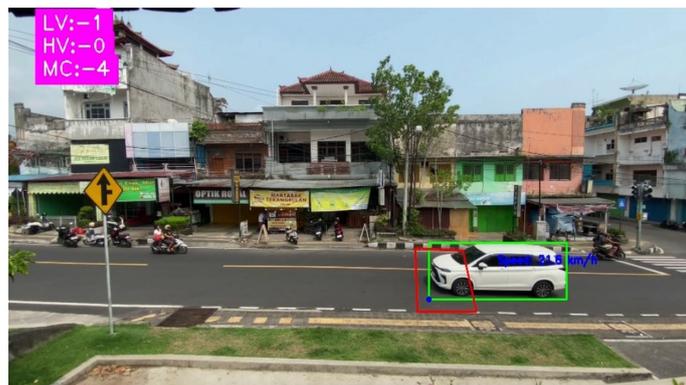


Figure 5. System display for speed estimation

Table 6. Example average speed by vehicle type (trial data)

Vehicle type	Speed (km/h)	95% CI (km/h)
Motorcycle	33.95	32.8-35.1
Light vehicle	35.32	34.1-36.6
Heavy vehicle	28.20	27.0-29.4

These estimates align with typical urban traffic behavior, where motorcycles and light vehicles tend to move faster than heavy vehicles, due to their enhanced maneuverability and lower weight. This pattern is in line with findings from other vision-based speed estimation studies [30]. Error robustness was assessed via repeated speed iterations, yielding a MAPE below approximately 2%, confirming that estimated speeds lie within acceptable error margins (typically under 5%) as reported in prior literature [11]. The narrow confidence intervals further indicate consistent performance across multiple samples. These findings demonstrate that the system provides reliable speed estimation in addition to volume measurement. By leveraging a single camera and standard hardware, it offers a promising, cost-effective approach for smart traffic monitoring systems—particularly suited for low-resource or infrastructure-limited contexts.

### 3.3. Discussion

The proposed YOLOv10-based system delivers strong performance. Traffic volume MAPE consistently under 2% with narrow 95% confidence intervals, and realistic speed estimates—motorcycles at

~34 km/h, light vehicles ~35 km/h, heavy vehicles ~28 km/h. These results align with established performance benchmarks for video-based traffic monitoring systems, which report acceptable MAPE thresholds below 5% in field deployments [29], [31].

In terms of practical deployment, the camera-based solution offers significant advantages over traditional methods. Radar systems provide high accuracy but are costly and less flexible; loop detectors are intrusive and require road modification [32], [33]. For example, radar-based detectors such as Wavetronix SmartSensor HD report volume accuracy within 1.6% and speed errors below 1 mph, while comparative studies found radar and iCone sensors produced speed errors of 1.4%–1.5% and volume errors of 7.8%–8.6% against pneumatic road tubes [29]. These benchmarks indicate that the accuracy achieved by our system is comparable to established sensing technologies. Importantly, the proposed solution remains low-cost, non-intrusive, and portable, making it especially suitable for developing regions where deployment budgets and infrastructure modifications are constrained.

The system nevertheless faces challenges in multi-lane conditions, nighttime operation, and adverse weather. As with most camera-based solutions, occlusion and poor illumination may reduce detection robustness. Enhancements such as multi-camera fusion, adaptive illumination, or image enhancement techniques could help overcome these limitations. Additionally, the architecture supports edge deployment, where local processing reduces bandwidth and latency, enabling integration with internet of things (IoT) frameworks and adaptive traffic control platforms [34], [35]. Future work should address scalability across diverse road environments and explore tighter integration with intelligent transport systems.

**4. CONCLUSION**

This study presented a portable, low-cost, and non-intrusive YOLOv10-based system for real-time traffic monitoring, capable of estimating both traffic volume and vehicle speed with high accuracy. Validation against manual ground truth showed excellent agreement, with a MAPE below 2% and detection metrics (precision, recall, and F1-score) consistently above 0.97, while speed estimation results aligned with realistic roadway conditions. The framework operates effectively using consumer-grade cameras and standard computing hardware, making it practical for deployment in resource-constrained environments. Although challenges remain in handling multi-lane traffic, night-time operation, and adverse weather, this work demonstrates the feasibility of vision-based deep learning as a scalable alternative to intrusive traffic counters, highlighting its potential integration into future intelligent transportation systems and reinforcing its contribution as a novel, field-deployable solution for modern traffic analytics.

**ACKNOWLEDGMENTS**

The authors would like to thank the Department of Transportation of Klungkung Regency for facilitating the data collection process, and Indonesian Land Transportation Polytechnic-STTD, Bekasi, for their academic support during this study.

**FUNDING INFORMATION**

This study was supported by Indonesian Land Transportation Polytechnic-STTD’s institutional research funding.

**AUTHOR CONTRIBUTIONS STATEMENT**

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ida Bagus Sradha Nanda	✓	✓	✓		✓	✓	✓		✓	✓	✓			
Masrono Yughartiman	✓	✓		✓	✓			✓		✓		✓	✓	✓
Eko Primadi Hendri	✓	✓		✓	✓			✓		✓		✓	✓	✓
I Made Suartika	✓	✓								✓		✓	✓	✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

## CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

## DATA AVAILABILITY

The dataset used in this study is available from the corresponding author upon reasonable request.

## REFERENCES

- [1] X. Yan, Y. Zhao, and H. X. Liu, "A probabilistic model for missing traffic volume reconstruction based on data fusion," *arXiv:2105.02777*, May 2021.
- [2] J. Pa'lo, J. Caban, M. Kiktová, and L. Černický, "The comparison of automatic traffic counting and manual traffic counting," *IOP Conference Series: Materials Science and Engineering*, vol. 710, no. 1, Dec. 2019, doi: 10.1088/1757-899X/710/1/012041.
- [3] K.-H. N. Bui, H. Yi, and J. Cho, "A multi-class multi-movement vehicle counting framework for traffic analysis in complex areas using CCTV systems," *Energies*, vol. 13, no. 8, Apr. 2020, doi: 10.3390/en13082036.
- [4] M. V. Peppas *et al.*, "Towards an end-to-end framework of CCTV-based urban traffic volume detection and prediction," *Sensors*, vol. 21, no. 2, Jan. 2021, doi: 10.3390/s21020629.
- [5] M. Naidoo, S. Paine, A. K. Mishra, and M. Y. A. Gaffar, "Reliable traffic monitoring using low-cost doppler radar units," in *2024 International Radar Conference (RADAR)*, Oct. 2024, pp. 1–6, doi: 10.1109/RADAR58436.2024.10993647.
- [6] J. Terven, D.-M. C.-Esparza, and J.-A. R.-González, "A comprehensive review of yolo architectures in computer vision: from yolov1 to yolov8 and yolo-nas," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, Nov. 2023, doi: 10.3390/make5040083.
- [7] A. Wang *et al.*, "YOLOv10: real-time end-to-end object detection," *arXiv:2405.14458*, Oct. 2024.
- [8] A. Charef, Z. Jarir, and M. Quafafou, "Mobile application utilizing yolov8 for real-time urban traffic data collection," *E3S Web of Conferences*, vol. 601, Jan. 2025, doi: 10.1051/e3sconf/202560100077.
- [9] H. Dou, S. Chen, F. Xu, Y. Liu, and H. Zhao, "Analysis of vehicle and pedestrian detection effects of improved YOLOv8 model in drone-assisted urban traffic monitoring system," *PLOS ONE*, vol. 20, no. 3, Mar. 2025, doi: 10.1371/journal.pone.0314817.
- [10] A. S. Geetha, M. A. R. Alif, M. Hussain, and P. Allen, "Comparative analysis of YOLOv8 and YOLOv10 in vehicle detection: performance metrics and model efficacy," *Vehicles*, vol. 6, no. 3, pp. 1364–1382, Aug. 2024, doi: 10.3390/vehicles6030065.
- [11] A. Macko, L. Gajdošech, and V. Kocur, "Efficient vision-based vehicle speed estimation," *Journal of Real-Time Image Processing*, vol. 22, no. 3, Jun. 2025, doi: 10.1007/s11554-025-01704-z.
- [12] A. Bustamante, L. M. Belmonte, R. Morales, A. Pereira, and A. F. -Caballero, "Video processing from a virtual unmanned aerial vehicle: comparing two approaches to using opencv in unity," *Applied Sciences*, vol. 12, no. 12, Jun. 2022, doi: 10.3390/app12125958.
- [13] OpenCV, "Getting started with videos," *Open Source Computer Vision*. Accessed: May 3, 2025. [Online]. Available: [https://docs.opencv.org/4.x/dd/d43/tutorial\\_py\\_video\\_display.html](https://docs.opencv.org/4.x/dd/d43/tutorial_py_video_display.html)
- [14] CVZone "cvzone: computer vision package for OpenCV and mediapipe," *GitHub*. Accessed: May 3, 2025. [Online]. Available: <https://github.com/cvzone/cvzone>
- [15] F. Zumstein, *Python for Excel: a modern environment for automation and data analysis*. Sebastopol, CA, USA: O'Reilly Media, Inc., 2021.
- [16] M. L. Ali and Z. Zhang, "The YOLO framework: a comprehensive review of evolution, applications, and benchmarks in object detection," *Computers*, vol. 13, no. 12, Dec. 2024, doi: 10.3390/computers13120336.
- [17] M. Zudhy, I. D. Student, T. Sumi, and A. Munawar, "Implementation of the 1997 Indonesian highway capacity manual (MKJI) volume delay function (in Indonesian context)," *Journal of the Eastern Asia Society for Transportation Studies*, vol. 8, pp. 350–360, 2010.
- [18] J.-D. Wu, B.-Y. Chen, W.-J. Shyr, and F.-Y. Shih, "Vehicle classification and counting system using YOLO object detection technology," *Traitement du Signal*, vol. 38, no. 4, pp. 1087–1093, Aug. 2021, doi: 10.18280/ts.380419.
- [19] S. Fu, H. Shi, S. Liang, X. Wang, and B. Ran, "Analytical optimized traffic flow recovery for large-scale urban transportation network," *arXiv:2409.03906*, 2024.
- [20] Y. Li, Q. Zhao, and M. Wang, "High-resolution traffic flow data from the urban traffic control system in glasgow," *Scientific Data*, vol. 12, no. 1, Feb. 2025, doi: 10.1038/s41597-025-04494-y.
- [21] S. Chauhan, S. Dave, J. Shah, and A. Kedia, "Assessing traffic characteristics for safe pedestrian crossings: developing warrants for sustainable urban safety," *Sustainability*, vol. 16, no. 10, May 2024, doi: 10.3390/su16104182.
- [22] G. Casey, B. Zhao, K. Kumar, and K. Soga, "Context-specific volume–delay curves by combining crowd-sourced traffic data with automated traffic counters: a case study for London," *Data-Centric Engineering*, vol. 1, Dec. 2020, doi: 10.1017/dce.2020.18.
- [23] H. Lian *et al.*, "Vehicle speed measurement method using monocular cameras," *Scientific Reports*, vol. 15, no. 1, Jan. 2025, doi: 10.1038/s41598-025-87077-6.
- [24] R. Yacouby and D. Axman, "Probabilistic extension of precision, recall, and F1 score for more thorough evaluation of classification models," in *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 79–91, doi: 10.18653/v1/2020.eval4nlp-1.9.
- [25] H. H. Elmousalami, "Comparison of artificial intelligence techniques for project conceptual cost prediction: a case study and comparative analysis," *IEEE Transactions on Engineering Management*, vol. 68, no. 1, pp. 183–196, Feb. 2021, doi: 10.1109/TEM.2020.2972078.
- [26] P. D. Rostron, T. Fearn, and M. H. Ramsey, "Confidence intervals for robust estimates of measurement uncertainty," *Accreditation and Quality Assurance*, vol. 25, no. 2, pp. 107–119, Apr. 2020, doi: 10.1007/s00769-019-01417-4.
- [27] K. H. K. Manguri and A. A. Mohammed, "A review of computer vision-based traffic controlling and monitoring," *UHD Journal of Science and Technology*, vol. 7, no. 2, pp. 6–15, Aug. 2023, doi: 10.21928/uhdjt.v7n2y2023.pp6-15.
- [28] S. K. Abbas *et al.*, "Vision based intelligent traffic light management system using faster r-cnn," *CAAI Transactions on Intelligence Technology*, vol. 9, no. 4, pp. 932–947, Aug. 2024, doi: 10.1049/cit2.12309.

- [29] A. Kalašová, P. Fabian, K. Čulík, and L. Škorvánkova, "Comparison of the accuracy of traffic flow intensity and speed measurement using a camera system and measuring devices such as sierzega and sdr on a congested road," *Vehicles*, vol. 7, no. 2, Jun. 2025, doi: 10.3390/vehicles7020059.
- [30] X. C. Vuong, R.-F. Mou, and T. T. Vu, "Vehicle tracking and speed estimation under mixed traffic conditions using YOLOv4 and sort: a case study of hanoi," *Transport Problems*, vol. 17, no. 4, pp. 17–26, Dec. 2022, doi: 10.20858/tp.2022.17.4.02.
- [31] A. A. Kashyap, S. Raviraj, A. Devarakonda, S. R. Nayak K, S. K V, and S. J. Bhat, "Traffic flow prediction models—a review of deep learning techniques," *Cogent Engineering*, vol. 9, no. 1, Dec. 2022, doi: 10.1080/23311916.2021.2010510.
- [32] X. Zhou, R. Ke, H. Yang, and C. Liu, "When intelligent transportation systems sensing meets edge computing: vision and challenges," *Applied Sciences*, vol. 11, no. 20, Oct. 2021, doi: 10.3390/app11209680.
- [33] M. Peyman, P. J. Copado, R. D. Tordecilla, L. do C. Martins, F. Xhafa, and A. A. Juan, "Edge computing and IoT analytics for agile optimization in intelligent transportation systems," *Energies*, vol. 14, no. 19, Oct. 2021, doi: 10.3390/en14196309.
- [34] C. Gheorghe and A. Soica, "Revolutionizing urban mobility: a systematic review of AI, IoT, and predictive analytics in adaptive traffic control systems for road networks," *Electronics*, vol. 14, no. 4, Feb. 2025, doi: 10.3390/electronics14040719.
- [35] T. Afrin, N. Yodo, A. Dey, and L. G. Aragon, "Advancements in UAV-enabled intelligent transportation systems: a three-layered framework and future directions," *Applied Sciences*, vol. 14, no. 20, Oct. 2024, doi: 10.3390/app14209455.

## BIOGRAPHIES OF AUTHORS



**Ida Bagus Sradha Nanda**    is a transportation practitioner and researcher currently working at the Department of Transportation of Klungkung Regency, Indonesia. He holds a diploma and a bachelor degree from the Indonesian Land Transportation Polytechnic-STTD. His research focuses on the application of technology in transportation systems, particularly in improving safety and traffic efficiency through smart infrastructure. He has published research in Forum Mekanika titled "Design of an Amiable Pelican (Aman) crossing based on IOS arduino nano: a case study on Basuki Rahmat street". His academic profile is available on Google Scholar. He can be contacted at email: gussradha@gmail.com.



**Masrono Yugihartiman**    is a lecturer at the Indonesian Land Transportation Polytechnic-STTD. He holds a doctor in Economic Science from Padjadjaran University and a master's degree in Transportation Planning and Engineering from ITS the University of Leeds, United Kingdom. He previously served as head of the Center for Research and Development of Land and Railway Transport and as director of railway safety at the Ministry of Transportation, Indonesia. His research focuses on intelligent transportation systems, congestion pricing, and sustainable urban mobility. He has published in several reputable international journals, including Sustainability (MDPI) and JRCP ITB. His academic profile is available on Google Scholar. He can be contacted at email: yhartiman.yh@gmail.com.



**Eko Primadi Hendri**    is a statistics lecturer at the Indonesian Land Transportation Polytechnic-STTD, Bekasi, Indonesia. He earned his undergraduate degree from Universitas Gadjah Mada and his Master of Science in Statistics from IPB University. His academic and research interests include applied statistics, data modeling, and statistical applications in transportation systems. He has contributed to several research publications in national and international journals and actively supports data-driven approaches in transport policy and planning. His academic profile is available on Google Scholar. He can be contacted at email: eko.primadi@ptdisttd.ac.id.



**I Made Suartika**    is a lecturer at the Indonesian Land Transportation Polytechnic-STTD and head of the master's program in Applied Marketing Innovation and Technology. He has served in various key positions including Director of PKTJ Tegal, Head of Planning and Cooperation at the Ministry of Transportation, and Transportation Attaché at the Indonesian Embassy in The Hague. He holds a master's degree in Transport from the University of New South Wales, Australia, and is currently pursuing a doctoral program in Leadership and Innovation Policy at Universitas Gadjah Mada. His research interests include transportation policy, road safety, and innovation in transportation systems. He can be contacted at email: made.suartika@ptdisttd.ac.id.