Available *online* at : http://chipset.fti.unand.ac.id/

Journal on Computer Hardware, Signal Processing, Embedded System and Networking

| ISSN (Online) 2722-4422 |

Internet of Things

Single Board Computer Based Driver's Mobile Phone and Vehicle Speed Detection System with Real Time Processing

Anisha Shathya[1], *Dodon Yendri*[*2]

[1,2] *Department of Computer Engineering, Faculty of Information Technology, Andalas University, Limau Manis, Padang, West Sumatera, 25163, Indonesia*

ARTICLE INFORMATION

Received: October 30, 2025
Revised: August 20, 2026
Available online: October 31, 2026

KEYWORDS

Speed monitoring, Driver, You Only Look Once (YOLO), OBD-II, Phone, Jetson Nano

CORRESPONDENCE

E-mail: dodon@it.unand.ac.id

A B S T R A C T

A toll road is a type of public road that is part of the national road network in Indonesia, where users are required to pay a toll fee to access it. Toll roads are specifically designed for vehicles traveling at high speeds, so drivers are expected to always be alert and comply with the speed limits set by the toll road authorities. This research aims to design and build a mobile phone and vehicle speed detection system capable of monitoring the condition of the driver and vehicle. This system uses an NVIDIA Jetson Nano as the main processor connected to OBD-II and a Raspberry Pi Camera Module 2. This system design aims to reduce the number of accidents on toll roads by utilizing image processing using YOLO (You Only Look Once) to detect an object and pose for the driver.

INTRODUCTION

A toll road is a type of public road that is part of the national road network system in Indonesia, where users are required to pay a toll fee to access it. Based on Government Regulation No. 15 of 2005 concerning toll roads, it is explained that the definition of a toll road is a public road that is part of the road network system and as a national road whose users are required to pay a toll. A toll is a certain amount of money paid for the use of a toll road. The amount of the toll rate differs for each class of vehicle and these provisions have been determined based on a presidential decree. Meanwhile, a toll road section is a part or segment of a certain toll road whose management can be carried out by certain business entities [1]. Toll roads usually have better infrastructure and are designed to provide good, efficient, and fast travel. Toll roads are usually also used to connect cities or regions in Indonesia. Toll roads aim to reduce travel time, improve traffic safety, and support equitable development and regional balance [2].

However, many drivers still ignore their vehicle speed while driving on toll roads, even though this is crucial to prevent potential accidents and ensure the safety of all road users. Driving at high speeds is certainly dangerous for both themselves and other drivers. Driving on a toll road does not mean a driver can freely drive beyond the designated speed limit. If the speed of the vehicle being driven exceeds the designated speed limit, the potential for accidents is very high.

Driving at high speeds is a major cause of death and serious injury on highways, especially toll roads. This is reflected in all aspects of life, including driving. Quoted from the official website of the Ministry of Public Works and Public Housing's Toll Road Regulatory Agency, speed limits on toll roads have been regulated in the Regulation of the Minister of Transportation concerning Procedures for Determining Vehicle Limits, article 3 paragraph 4 in article 23 paragraph 4. In the regulation, it is stated that the speed limit for toll roads within the city is a minimum of 60 km/hour and a maximum of 80 km/hour, while the speed limit for toll roads outside the city is a minimum of 80 km/hour and a maximum of 100 km/hour [3].

The Indonesian National Police Traffic Corps (Korlantas Polri) recorded 27,791 vehicles violating speed limits on toll roads. This

https://doi.org/10.25077/chipset.7.02.102-108.2026
Attribution-NonCommercial-ShareAlike 4.0 International. Some rights reserved

figure is the accumulation of data from three toll road sections from the start of the implementation of electronic ticketing on toll roads on April 1 to April 12, 2022 [4].

The Central Statistics Agency (BPS) reported that the number of traffic accidents in Indonesia increased by an average of 4.87% per year between 2015 and 2019. This increase was accompanied by an increase in the number of fatalities and minor injuries (1.41% and 6.26%, respectively), as well as an increase in the value of material losses due to accidents (4.23% per year) [5]. According to data taken from MTI, the total number of traffic accidents in Indonesia throughout 2023 was recorded at 116,000 cases. This number increased by 6.8% compared to the previous year [6].

From the records of the Indonesian National Police Traffic Corps, it is also explained that the causes of traffic accidents in Indonesia are caused by several things, namely 61% of accidents are caused by human factors or human error such as problems with driving inability/skills and driver character, for example negligence, laziness, carelessness, and recklessness, then 9% are caused by vehicle factors (related to fulfilling roadworthy technical requirements), and 30% are caused by infrastructure and environmental factors [7]. (11)

Therefore, the main focus in this problem is the driver himself, because there are still many drivers who do not comply with traffic regulations on toll roads. Even though there are clear rules both on toll roads and in law, there are still many drivers who ignore them, such as violating the speed limit or being distracted by other things while driving, which will ultimately cause the risk of accidents for both the driver himself and other road users. Warnings when a driver exceeds the speed limit and when the driver is distracted while driving are crucial for preventing potential accidents for both the driver and other road users. These warnings encourage drivers to change their driving behavior to prevent potential accidents. This not only ensures their own safety but also protects other road users who could be affected by a driver's negligence in violating traffic regulations on toll roads.

Therefore, sophisticated and modern technology is needed to alert drivers whether their vehicle is exceeding the speed limit or whether the driver is distracted while driving.

METHOD

This study employs a device that is compatible with a car and does not interfere with use or placement on the dashboard. The device is designed like a mini robot that will be placed on the dashboard. This system uses the YOLO algorithm to detect objects and driver poses and uses the Jetson Nano as its main processing unit.

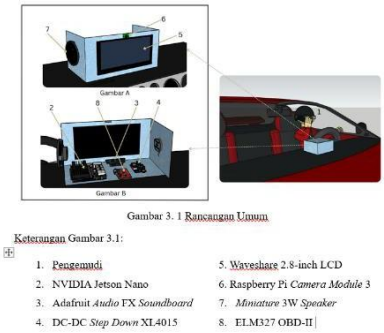


Figure 1. 3D System Design

Figure 1. Shows a 3D design where the system consists of 5 main components, namely the NVIDIA Jetson Nano as the main controller in the system, followed by the OBD-II as a reading of vehicle speed, thirdly there is a Raspberry Pi Camera Module 2 as a reading of the objects and driver pose, and there is a power bank as a supporting power source. All components are equipped with a charge module so that the power bank can be recharged so that it can be recharged again. For a clear overview of the hardware used in this system, here's a general overview:



Figure 2. General System Design

Figure 2. In general, this system is designed to monitor driver phone and vehicle speed in real time and send warning notifications to the driver via mini speaker. As shown in Figure 6, the camera layout is adjusted to the position on the top of LCD so that the data that collected like vehicle speed or driver's phone are shown on the LCD.

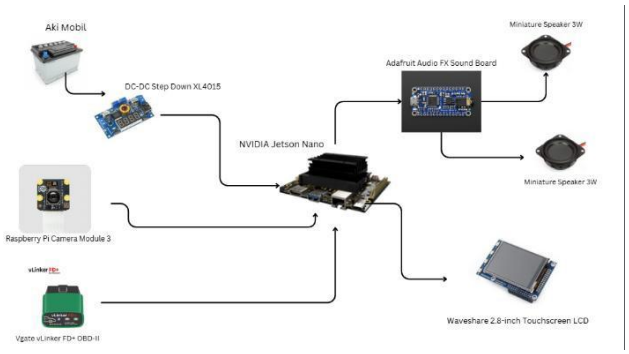


Figure 3. Hardware Desain

Figure 3. The above shows three system processes, with the NVIDIA Jetson Nano, OBD-II and Raspberry Pi Camera Module 2 serving as input to the NVIDIA Jetson Nano, allowing data to be processed and sent to the mini speaker and can be hear by the driver. This information is then sent to the driver directly. A power source is also required to maintain the operation of each component, using a 30000 mAh power bank and a type-c cable as a charger.

https://doi.org/10.25077/chipset.7.02.102-108.2026

The data obtained from these components, both in the form of visual images and vehicle speed data, is converted into digital information that depicts the driver and vehicle's condition in real time. This process allows the system to monitor two critical factors simultaneously. The captured data is then sent to and processed by the NVIDIA Jetson Nano, which serves as the central processing unit.

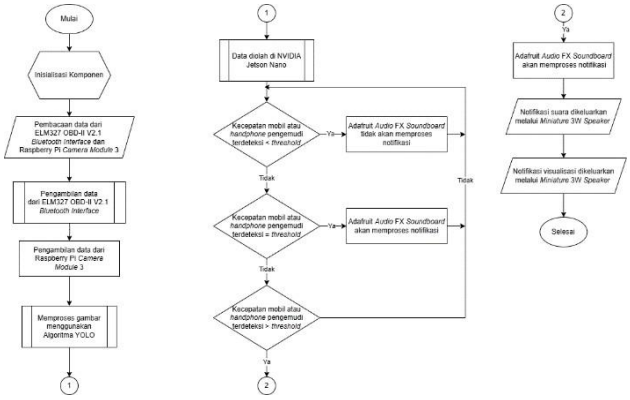


Figure 4. Flowchart System

Figure 4. is a flowchart depicting the process design of the system to be designed. This system functions to detect car speed and drivers using their mobile phones within a certain radius threshold.

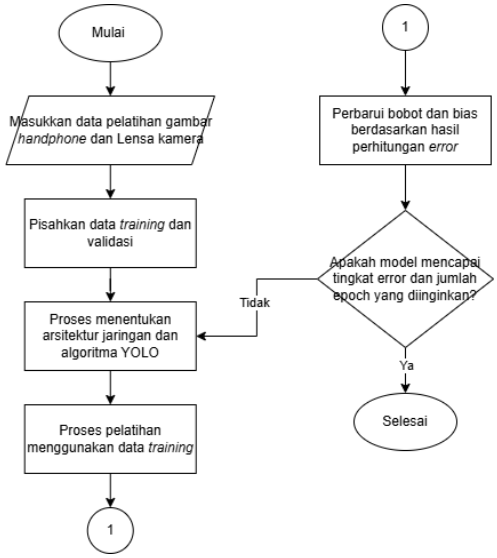
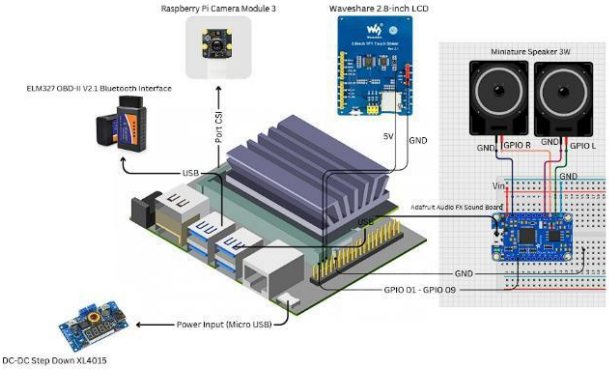


Figure 5. Flowchart YOLO Testing Process

Figure 5. shows a flowchart of the YOLO algorithm's data training process. This process begins with collecting data in the form of mobile phone images and camera lenses relevant to the needs. After the data is collected, preparation is carried out, such as cleaning the data, annotating the objects to be recognized, and ensuring the data is ready for use. Thedata is then divided into two parts: training data, which is used to train the model, and validation data, which is used to test the model's performance during training.



The trained model is then tested using completely new data that the model has never seen before. This is done to ensure that the model can effectively detect objects in situations different from those encountered in the training data.

This explanation provides a simple overview of how YOLO is trained to accurately recognize objects. This process ensures that the model performs well in detecting objects under a variety of conditions.

Predictions are made by referring to the following table of rules:

Table.1.Dataset Overview

Spesifikasi	Keterangan
Dataset	Consisting of 3,623 mobile phone images
Format	.JPG
Classes	1. *Handphone* 2. *Handphone Lens*
Pembagian Data	Of the 3,623 images, they were divided into 3,195 train data, 280 valid images, and 148 mobile phone images.

The following is the architecture of the ANN model that will be built:

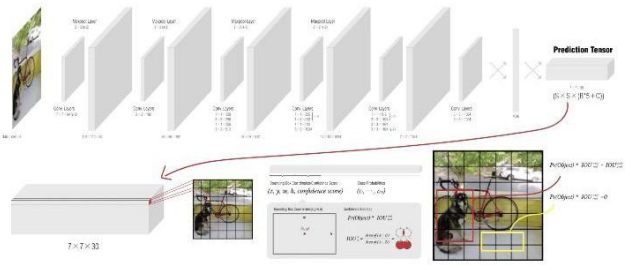


Figure 6. YOLO Structure Model

The figure illustrates the architecture and working process of the YOLO (You Only Look Once) object detection model. The model takes an input image of size 448×448×3, which passes through multiple convolutional and max-pooling layers to extract

important visual features while reducing spatial dimensions, resulting in a 7×7×1024 feature map. This output is then flattened and passed through fully connected layers to produce a prediction tensor of 7×7×30, where the image is divided into a 7×7 grid, and each grid cell predicts bounding boxes with coordinates (x, y, w, h), a confidence score, and class probabilities.

Each grid cell is responsible for detecting objects whose centers fall within it, and the confidence score represents both the likelihood of an object being present and the accuracy of the bounding box, measured using Intersection over Union (IoU). The final detections are obtained by multiplying the objectness probability with the IoU, allowing YOLO to detect multiple objects in a single forward pass efficiently and accurately.

In this design, the Raspberry Pi Camera Module 3 is connected to the NVIDIA Jetson Nano via the CSI (Camera Serial Interface) port interface, which allows for direct and fast image and video data transfer. The ELM327 OBD-II used to read vehicle speed data is connected to the NVIDIA Jetson Nano using a USB interface, facilitating data communication between vehicle devices and the NVIDIA Jetson Nano processing system. Meanwhile, the Adafruit Audio FX Sound Board, which functions to process and generate notification sounds, is also connected to the NVIDIA Jetson Nano via a USB interface.

RESULTS AND DISCUSSION

The implementation phase is the stage where the ideas or designs that have been developed are brought to life in the real world. At this stage, developers focus on realizing the previously developed concepts, methods, and designs to ensure they operate as planned. Implementation also includes a system testing process, where data is collected from each component to determine whether the collected data aligns with the design.

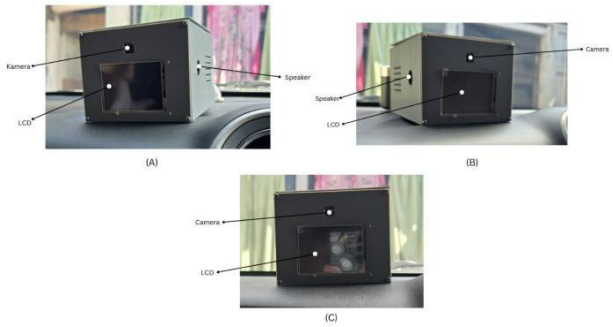


Figure 7. System Implementation Results

In Figure 7, shows a device designed to be mounted on the car's dashboard to provide a clear view of the driver and the surrounding environment. The dashboard was chosen as the installation location based on ease of access, stability, and optimal camera visibility for detecting driver activity in real time.

Figure (A) shows the device's appearance from the driver's perspective. From this angle, the camera is clearly visible at the top, the LCD is in the front, and the speaker is on the left side of the device. Figure (B) shows the device's appearance from the passenger sitting to the left of the driver. This angle further enhances the visibility of the camera at the top, the LCD is in the front, and the speaker is on the right side of the device. Figure (C) shows the device's appearance from the rear passenger's perspective.

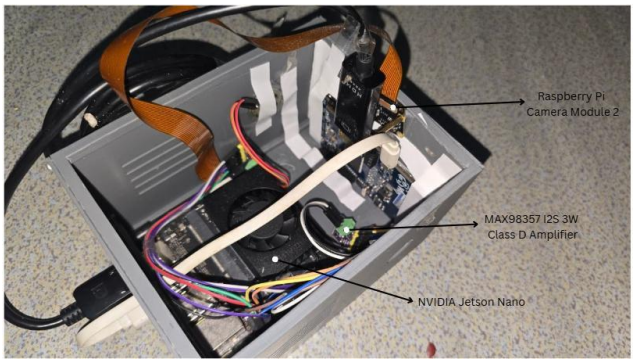


Figure 8. Hardware Implementation Results

In Figure 8. The NVIDIA Jetson Nano functions as a control center, reading data from the Raspberry Pi Camera Module 2 and the OBD-II, processing it using the YOLO model, then sending the classification battery is the main power source, with a larger capacity for up to ± hours of use. module is used to recharge the battery via USB type-C.

This system is designed to detect mobile phones used by drivers while driving. At this stage, a dataset is collected as a basis for training the detection model. The dataset is collected by capturing various images of drivers in different conditions, both with and without mobile phones, to ensure the model can accurately recognize patterns.

Table.2 YOLO Dataset

No	Data Types	Number of Images	Description
1	Train	3.195	Used for the YOLOv8 model training process.
2	Valid	280	Used to validate training results.
3	Test	148	Used to test the final performance of the model.
	Total	3.623	

The dataset contains a collection of images of mobile phone objects collected under various shooting conditions. Each image is grouped into three main sections: training data, validation data, and test data. Train data constitutes the largest portion of the dataset used to train the YOLO model to optimally recognize

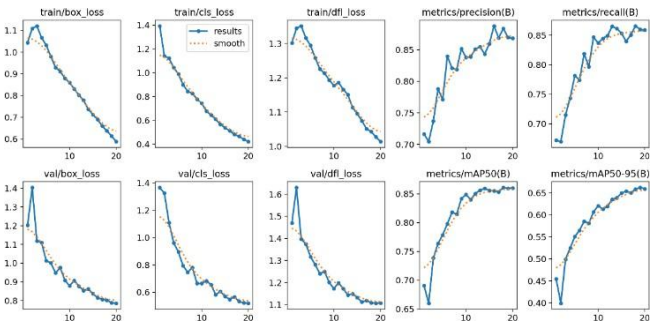


Figure 9. Training Results Graph

patterns and characteristics of mobile phone objects. Meanwhile, validation data is used to measure the model's ability to generalize

https://doi.org/10.25077/chipset.7.02.102-108.2026

to new data not directly trained on. The test data is used to evaluate the model's final performance after the training process is complete.

Figure 9. shown a Graph of YOLO model training results over 20 epochs. The graph shows changes in loss values (train and validation) and evaluation metrics such as precision, recall, mAP50, and mAP-95 at each epoch. It can be seen that the box loss, cls loss, and DFL loss values consistently decrease with increasing epochs, both on the training and validation data. This indicates that the model is increasingly able to adjust its weights to detect objects more accurately.

Based on the final training results, the YOLOv8 model successfully achieved a mean Average Precision (mAP50) value of 0.86 and mAP50-95 of 0.66. High precision and recall values indicate that the model is able to detect objects with a good level of accuracy and reliability. The training process was completed at the 20th epoch with a total training time of approximately 0.35 hours or equivalent to 21 minutes. These results indicate that the model configuration, number of epochs,

and input resolution used are optimal enough to produce a stable mobile phone detection model and are ready for use in the next testing stage. This best.pt model file is then used in the implementation stage of real-time detection using NVIDIA Jetson Nano.

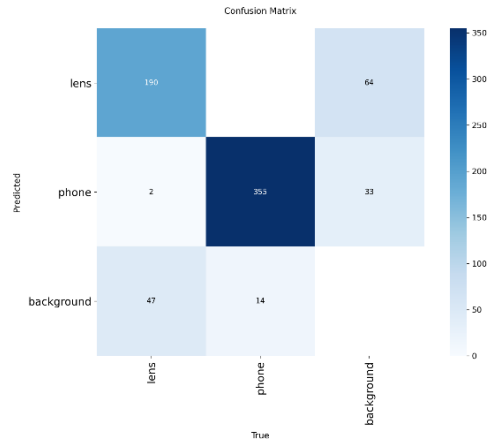


Figure 10. *Confusion Matrix*

The confusion matrix illustrates the performance of the object detection model in classifying three categories: lens, phone, and background. The diagonal values represent correctly classified samples, while off-diagonal values indicate misclassifications. The model successfully predicted 190 instances of *lens*, 355 instances of *phone*, and 14 instances of *background* correctly. However, there were some errors where *lens* images were misclassified as *background* (64 cases) and a few *background* instances as *lens* (47 cases). The *phone* class achieved the highest accuracy with minimal misclassification, showing only 2 cases wrongly labeled as *lens* and 33 as *background*. Overall, the model demonstrates strong performance, particularly in detecting *phone* objects, though there remains some confusion between *lens* and *background* classes that could be improved with further training or data augmentation.

Table 3. YOLO Model Accuracy Test Results

No	Parameter Evaluasi	Nilai	Deskripsi
1	*Precision*	0.97	The model's ability to correctly detect mobile phone objects compared to the total positive detections.
2	*Recall*	0.95	The model's ability to find all the mobile phone objects that actually exist in the image.
3	*mAP50-95*	0.96	Average overall accuracy of the model at various Intersection over Union (IoU) thresholds

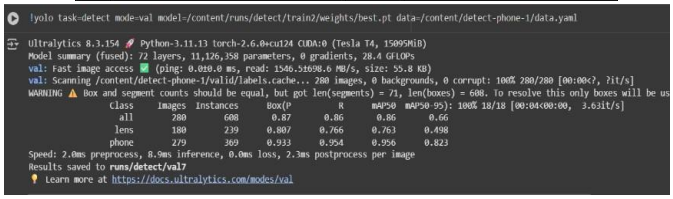


Figure 11. Final Training Results

Table 3. above shows the YOLO Object Detection model demonstrated excellent detection performance. A precision value of 0.97 indicates that the majority of positive detections were true positives. Meanwhile, a recall of 0.95 demonstrates the model's ability to recognize nearly all mobile phone objects in the test images. The mAP50-95 value of 0.96 reinforces these results, reflecting the model's average accuracy across various test scenarios. Based on these results, it can be concluded that the YOLO model has high accuracy, a low detection error rate, and good stability for use in a real-time driver mobile phone detection system.

Table.4 System testing on respondents

No	Kondisi Pengujian	Pencahayaan	Intensitas Cahaya (LUX)	Hasil Deteksi	Deskripsi
1	Siang Hari	Terang alami	12.000 Lux	Handphone terdeteksi	Deteksi stabil dan akurat
2	Sore Hari	Mendung	6.000 Lux	Handphone terdeteksi	Deteksi stabil dan akurat
3	Malam Hari	Lampu kabin	250 Lux	Handphone terdeteksi, kadang tidak stabil	Deteksi sedikit melambat, tetap akurat
4	Siang Hari	Terang alami	12.000 Lux	Handphone tidak terdeteksi	Tidak ada deteksi palsu (*false positive*)
5	Siang Hari	Hujan	2.000 Lux	Handphone Terdeteksi	Deteksi stabil dan akurat
6	Malam Hari	Gelap	50 Lux	Handphone terdeteksi, tidak stabil	Pengaruh pencahayaan minim

Based on the test results above, the YOLO Object Detection model is capable of good detection in bright, cloudy, rainy, and dark conditions, with a high level of stability. Detection begins to decline slightly when lighting is very low, but can still identify mobile phone objects with a low error rate. This indicates that the model is reliable enough to be used in a live driver monitoring system, especially in driving conditions during the day, evening, or night.

These results are consistent with light intensity (Lux) measurements in each condition, where optimal detection performance occurs at lighting levels above 5,000 Lux, such as during the day (±12,000 Lux) and evening (±6,000 Lux). In cloudy or rainy conditions with an intensity of around 2,000 Lux, the system is still able to detect objects accurately even though the lighting is reduced. However, in night conditions with cabin

lights (±250 Lux) or dark conditions without additional lighting (±50 Lux), the level of detection stability begins to decrease due to the limited light received by the camera.

Table 6. YOLO Performance Test Results

No	Resolusi Kamera	Rata-rata FPS	Waktu Respons	Deskripsi
1	480p (640×480)	28–30 FPS	Cepat	Gambar cukup jelas, deteksi sangat lancar
2	720p (1280×720)	18–22 FPS	Sedang	Gambar lebih tajam, sedikit lag
3	1080p (1920×1080)	5–6 FPS	Lambat	Gambar sangat tajam, tapi deteksi melambat

Based on the test results in Table 4, the system demonstrated that 720p resolution was the optimal setting. Although the FPS decreased slightly compared to 480p, the resulting images were significantly sharper and significantly improved object detection accuracy. Furthermore, detection performance remained stable and responsive, even when the system was running simultaneously with OBD-II data reading and warning sound playback via the MAX98357A DAC module. This demonstrates that the NVIDIA Jetson Nano has sufficient capabilities to run the YOLOv8 model in real time without experiencing significant performance degradation. Therefore, it can be concluded that the YOLOv8-based detection system on the NVIDIA Jetson Nano performs efficiently and reliably for application in real-time driver monitoring applications..

Table 5 YOLO Accuracy Test Results

No	Kondisi Pengemudi dan Penumpang	Jumlah Gambar Uji	Deteksi Benar	Deteksi Salah	Akurasi (%)
1	Pengemudi memegang handphone, penumpang tidak	200	192	8	96,0%
2	Pengemudi tidak memegang handphone, penumpang memegang handphone	200	185	15	92.5%
3	Pengemudi tidak memegang handphone, penumpang tidak memegang handphone	200	198	2	99.0%
		Rata-rata Akurasi			92,7%

Based on the test results in Table 5, the YOLOv8 Pose Detection model had an average accuracy of 92.7%. Misdetection errors generally occurred when the phone was in the center of the vehicle or too close to the passenger, making it difficult for the model to accurately determine whether the object was being held by the driver or the passenger. This occurred because the joint points (pose keypoints) between the driver and passenger were close together, especially when seated parallel, causing the system to sometimes incorrectly associate the passenger's hand with the driver's. Overall, the model was able to provide consistent and accurate detection results for use in a real-time driver alert system. This body pose detection was then combined with the phone object detection results from the YOLOv8 Object Detection model, allowing the system to logically determine that the driver, not the passenger, was using the phone.

CONCLUSIONS

Based on the tests conducted, the YOLO-based system on the NVIDIA Jetson Nano was proven to be able to detect mobile phones in both drivers and passengers with high accuracy, so that mobile phone users can be distinguished precisely. In addition, the YOLOv8 Pose Detection model is effective in detecting the driver's joint points, especially on the hands and head, to determine whether the driver is holding a mobile phone or not. The performance test results show that the Pose Detection model is able to process an average of 20.3 FPS with a latency of around 34 ms per frame, remaining stable and responsive, while the Object Detection model is able to process 720p resolution videos and still detect objects well, even in night conditions, with accuracies reaching around 92.7% and 97.0%, respectively. Overall, this system is proven to be reliable, responsive, and suitable for implementation as a real-time driver warning system, so that it can improve driving safety by providing early detection of risky behavior such as mobile phone use while driving.

REFERENCES

[1] Itenas Library, “Bab II Tinjauan Pustaka 2,” vol. Vol 2, p. 6.

[2] Darwanto, E. Yusuf Agung Gunanto, and B. Samadikun, “Analisis Dampak Pembangunan Jalan Tol Pejagan -,” vol. Vol. 2 No. 2, p. 45, Jun. 2021.

[3] Kumparan, “Ketahui Batas Kecepatan di Jalan Tol Sesuai Aturan Pemerintah.” Accessed: Jan. 23, 2025. [Online]. Available: https://kumparan.com/berita-hari-ini/ketahui-batas-kecepatan-di-jalan-tol-sesuai-aturan-pemerintah-224a9bevwwZ/2

[4] Tempo, “Korlantas Polri Catat 27.791 Pelanggaran Batas Kecepatan di Jalan Tol.” Accessed: Jan. 23, 2025. [Online]. Available: https://www.tempo.co/arsip/korlantas-polri-catat-27-791-pelanggaran-batas-kecepatan-di-jalan-tol-370759

[5] I. Zainafree, N. Syukria, S. Addina, and M. Z. Saefurrohim, “Epidemiologi Kecelakaan Lalu Lintas: Tantangan dan Solusi,” in Bookchapter Kesehatan Masyarakat Universitas Negeri Semarang, no. 1, Universitas Negeri Semarang, 2022, pp. 92–127. doi: 10.15294/km.v1i1.70.

[6] Administrator, “Angka Kematian Kecelakaan Transportasi Turun Sepanjang 2023.” Accessed: Jan. 23, 2025. [Online]. Available: https://indonesia.go.id/kategori/editorial/7879/angka-

 https://doi.org/10.25077/chipset.7.02.102-108.2026

kematian-kecelakaan-transportasi-turun-sepanjang- 2023?lang=1
[7] Biro Komunikasi dan Informasi Publik, "Tekan Angka Kecelakaan Lalu Lintas, Kemenhub Ajak Masyarakat Beralih ke Transportasi Umum dan Utamakan Keselamatan Berkendara." Accessed: Jan. 23, 2025. [Online]. Available: https://dephub.go.id/post/read/%E2%80%8Btekan-angka-kecelakaan-lalu-lintas,-kemenhub-ajak-masyarakat-beralih- ke-transportasi-umum-dan-utamakan-keselamatan- berkendara
[8] S. P. Demita, D. Yendri, and R. Suwandi, "Rancang Bangun Sistem Keamanan Helm dan Berkendara Berbasis Mikrokontroler," CHIPSET (Journal on Computer Hardware, Signal Processing, Embedded System and Networking), vol. 5, no. 01, pp. 20–34, Apr. 2024, doi: 10.25077/ chipset.5.01.20-34.2024.
[9] P. I. Hidayat, R. Eka Putri, and B. Rahmadya, "Rancang Bangun Sistem Pengawasan Pemakaian Alat Pelindung Diri Berbasis Single Board Computer," CHIPSET (Journal on Computer Hardware, Signal Processing, Embedded System and Networking), vol. 5, no. 01, pp. 65–75, Apr. 2024, doi: 10.25077/chipset.5.01.65-75.2024.
[10] G. Adnediel and D. Yendri, "Rancang Bangun Alat Pelatihan Parkir Mobil Berbasis Mikrokontroler," CHIPSET (Journal on Computer Hardware, Signal Processing, Embedded System and Networking), vol. 5, no. 02, pp. 116–125, Oct. 2024, doi: 10.25077/chipset.5.02.116-125.2024.
[11] Y. Zaita, K. Saddami, and Nasaruddin, "Integrating YOLOv7 with FixMatch for Enhancing Vehicle Detection Performance in Mixed Traffic Environments," Jurnal Nasional Teknik Elektro (JNTE), vol. 13, no. 3, pp. 153–160, Nov. 2024, doi: 10.25077/jnte.v13n3.1250.2024.
[12] M. Kezia and K. V. Anusuya, "IoT-based Web-Integrated OBD-II Telematics for Real-Time Vehicle Health Monitoring System," in 2023 International Conference on Intelligent Technologies for Sustainable Electric and Communications Systems (iTech SECOM), Coimbatore, India, 2023, pp. 1–6, doi: 10.1109/iTechSECOM59882.2023.10435307.
[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
[14] D. Maji, S. Nagori, M. Mathew, and D. Poddar, "YOLO-Pose: Enhancing YOLO for Multi Person Pose Estimation Using Object Keypoint Similarity Loss," in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 2022, pp. 2636–2645, doi: 10.1109/CVPRW56347.2022.00297.

AUTHOR(S) BIOGRAPHY

Anisha Shathya
Born in Pekanbaru, 17 July 2003, he attended SMAIT Al-Fityah and is currently pursuing a degree in Computer Engineering at the Faculty of Information Technology, Andalas University.