

The 9th International Conference on Culture Technology
October 29 - October 31, 2026 Shanghai - China

Design of Blind Spot System as Collision Warning on Truck Vehicle Based on Internet of Things

I Putu Agung Bayupati

Made Dimas Aldi Dwi Pramana

Kadek Suar Wibawa

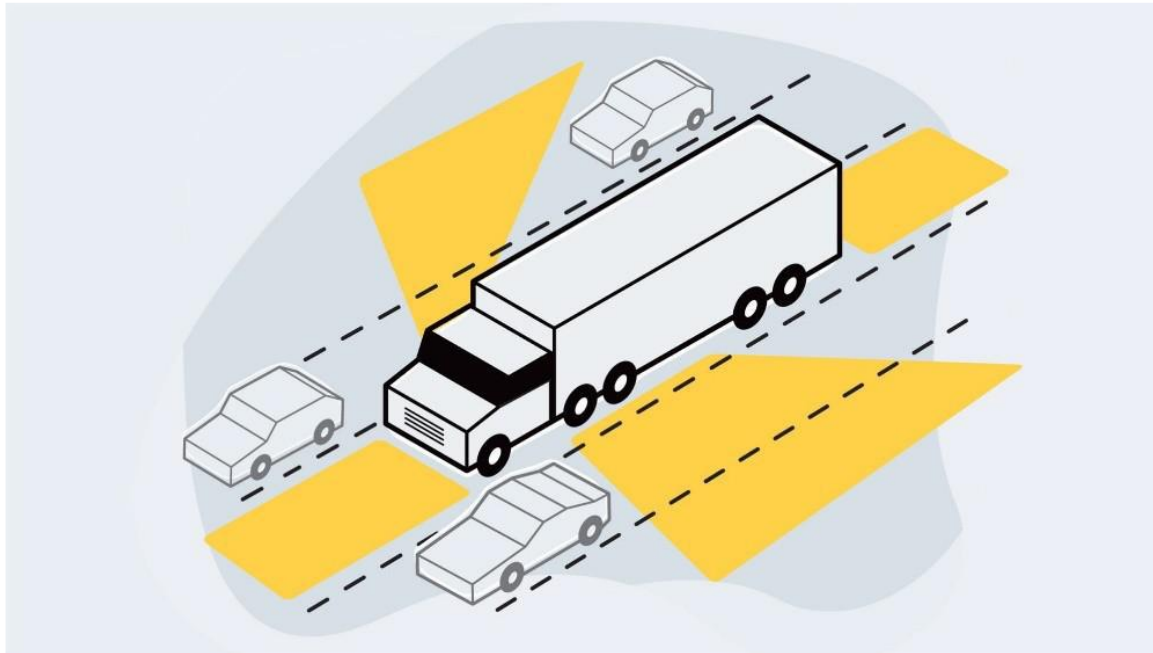
Putu Wira Buana, Information

Desy Purnami Singgih Putri

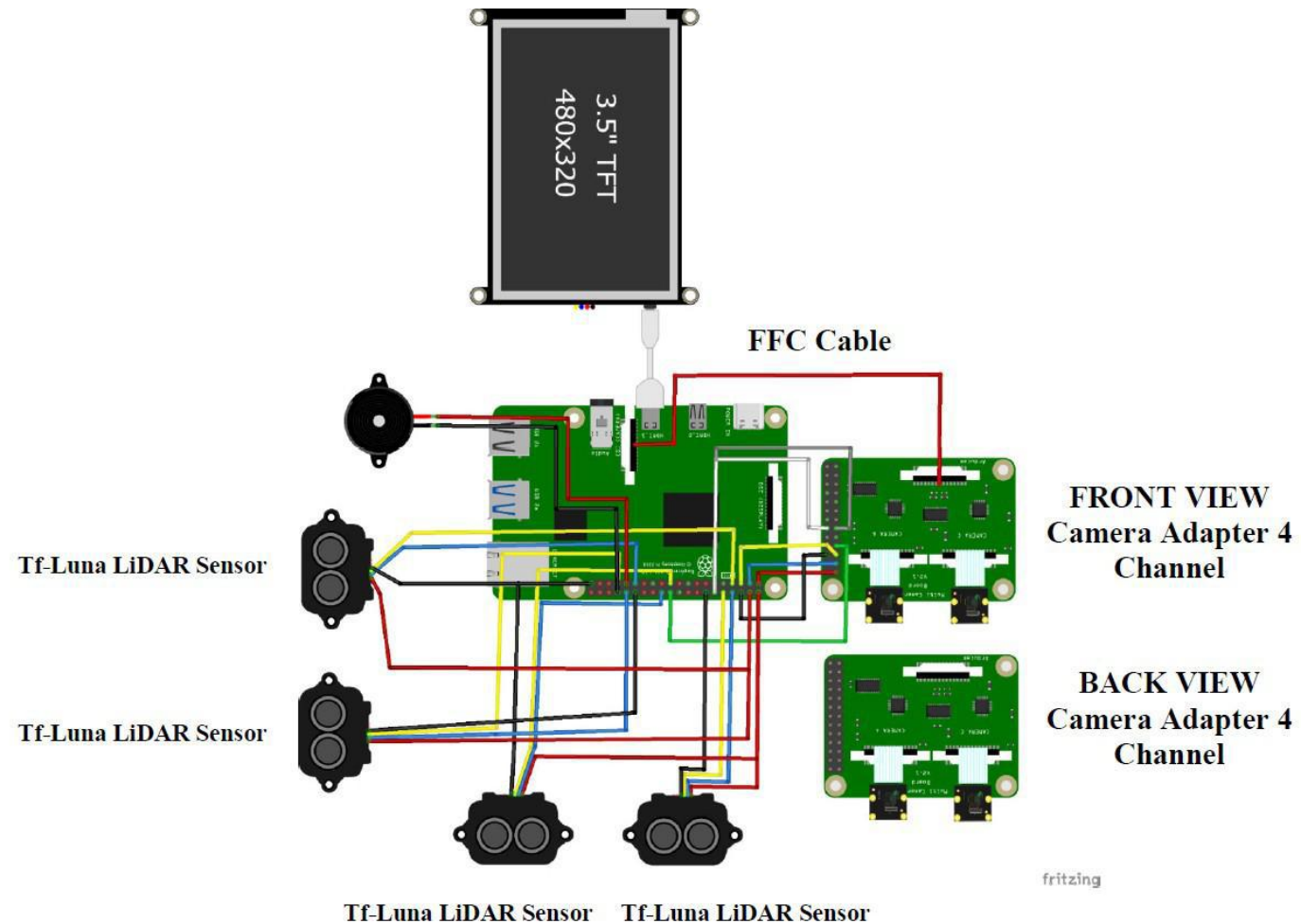
Ida Ayu Gde Suwiprabayanti Putra

Research Problem

- ▶ How to develop IoT-based sensor technology offers a promising solution for improving vehicle safety
- ▶ Performance in detecting objects or other vehicles within blind spots



Schematic of Design IoT Board



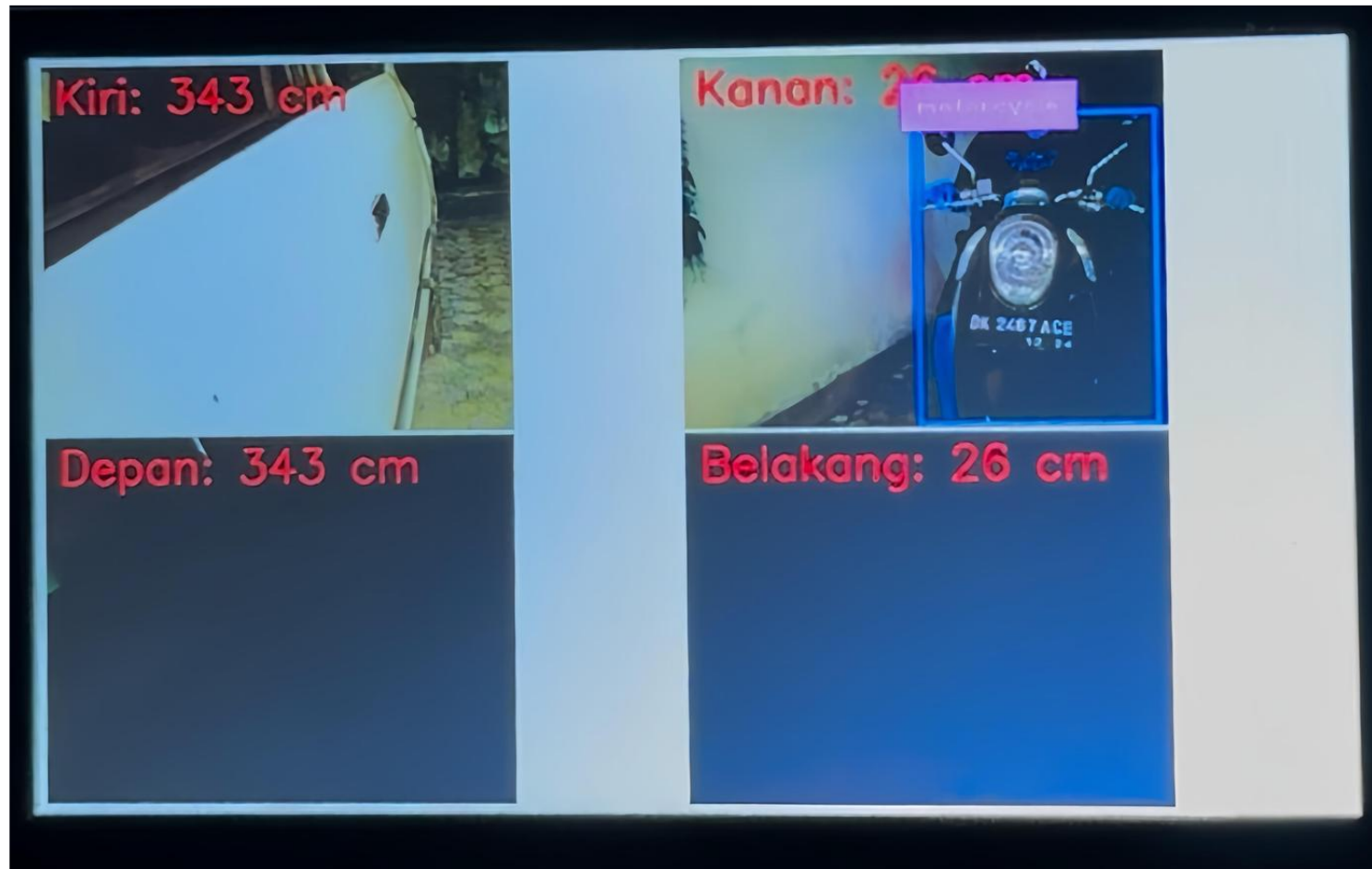
Results And Discussion

Raspberry Pi Camera Test

- ▶ The Raspberry Pi Camera OV5647 indicate that the camera operated properly and successfully displayed video streams from the truck's different blind-spot areas
- ▶ The video streams displayed smoothly with satisfactory image quality, showing that the camera could capture and transmit visual information for blind spot monitoring
- ▶ YOLOv8 processed the captured images and identified objects by generating bounding boxes around them. Each bounding box included a label indicating the detected object's class, such as a car or motorcycle

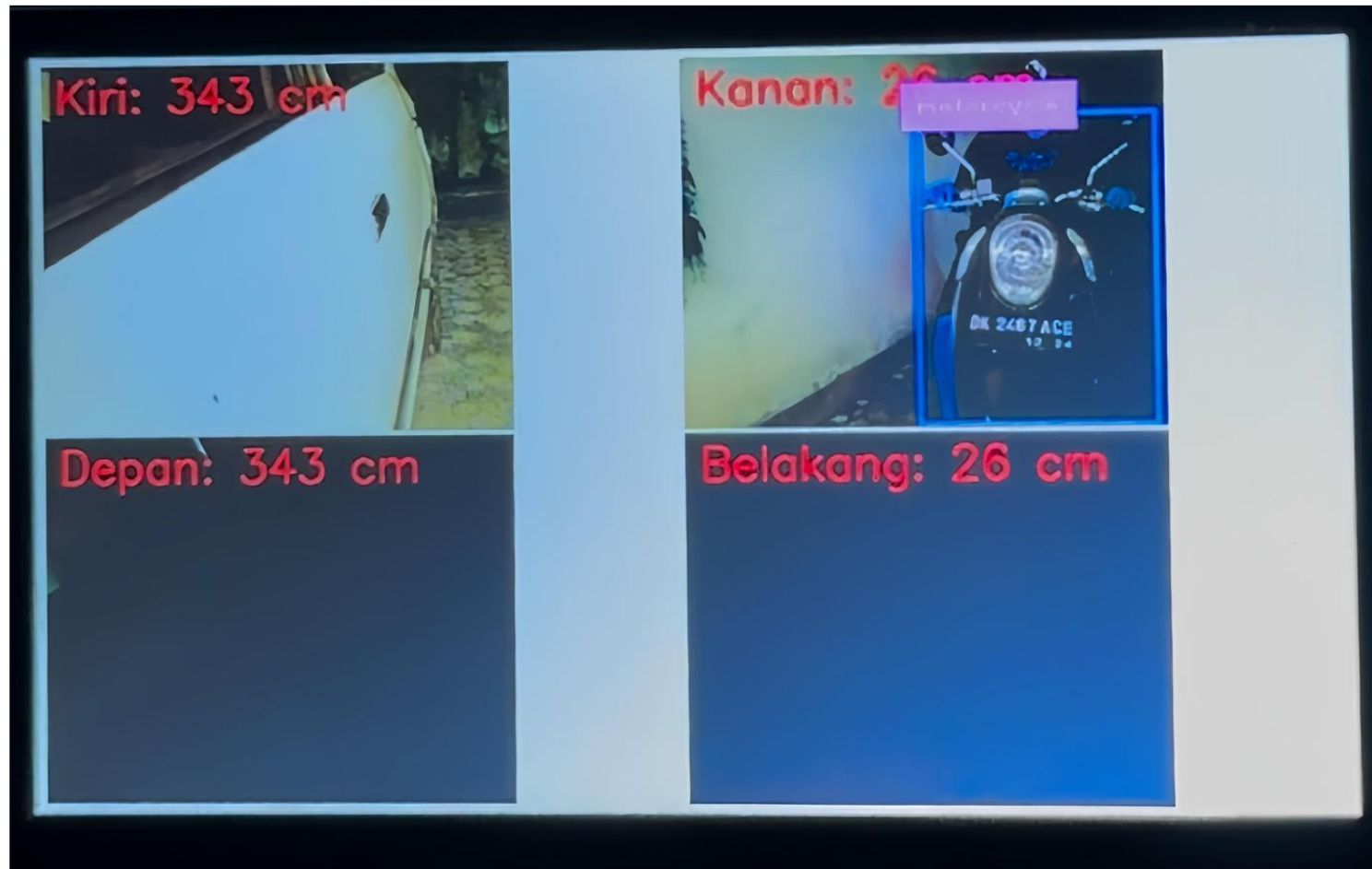
Cont'd

Raspberry Pi Camera OV5647



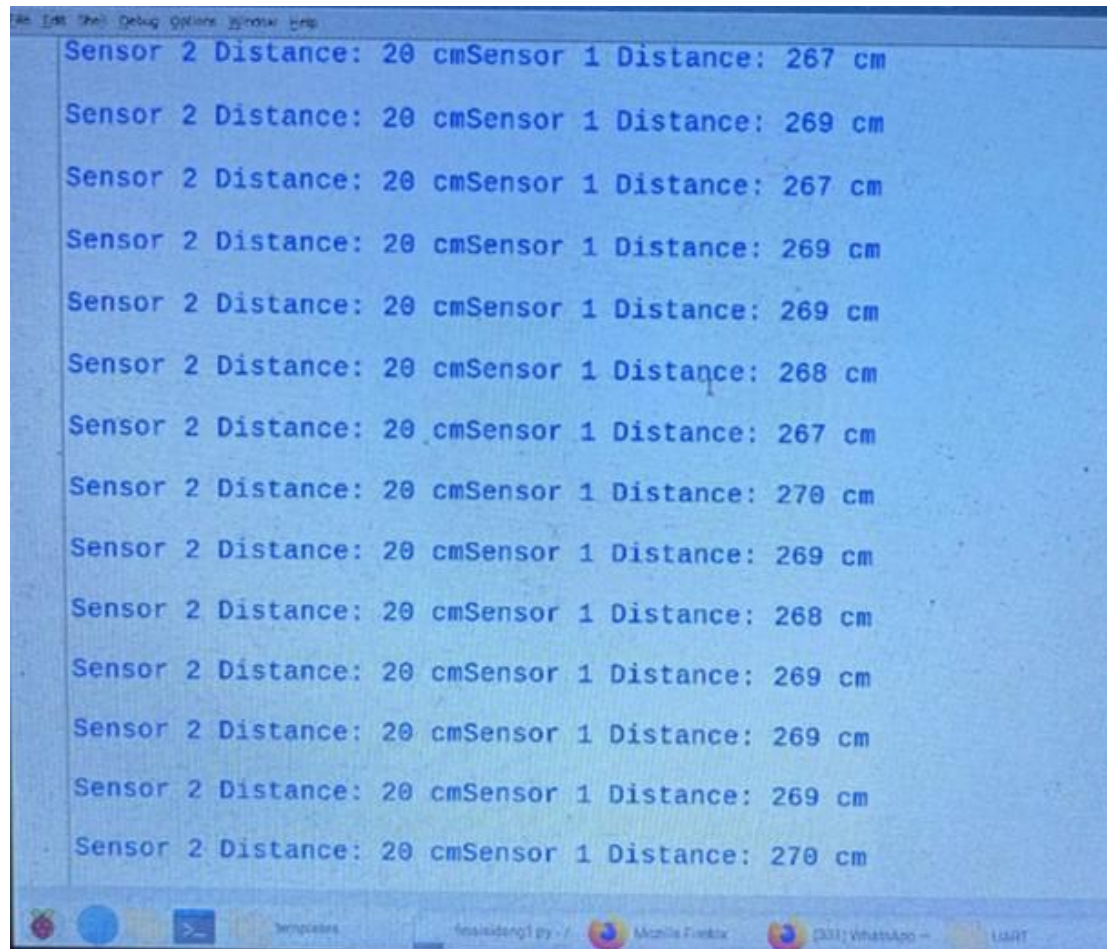
Cont'd

Raspberry Pi Camera OV5647



Cont'd

Display of Distance Data in Shell Kernel



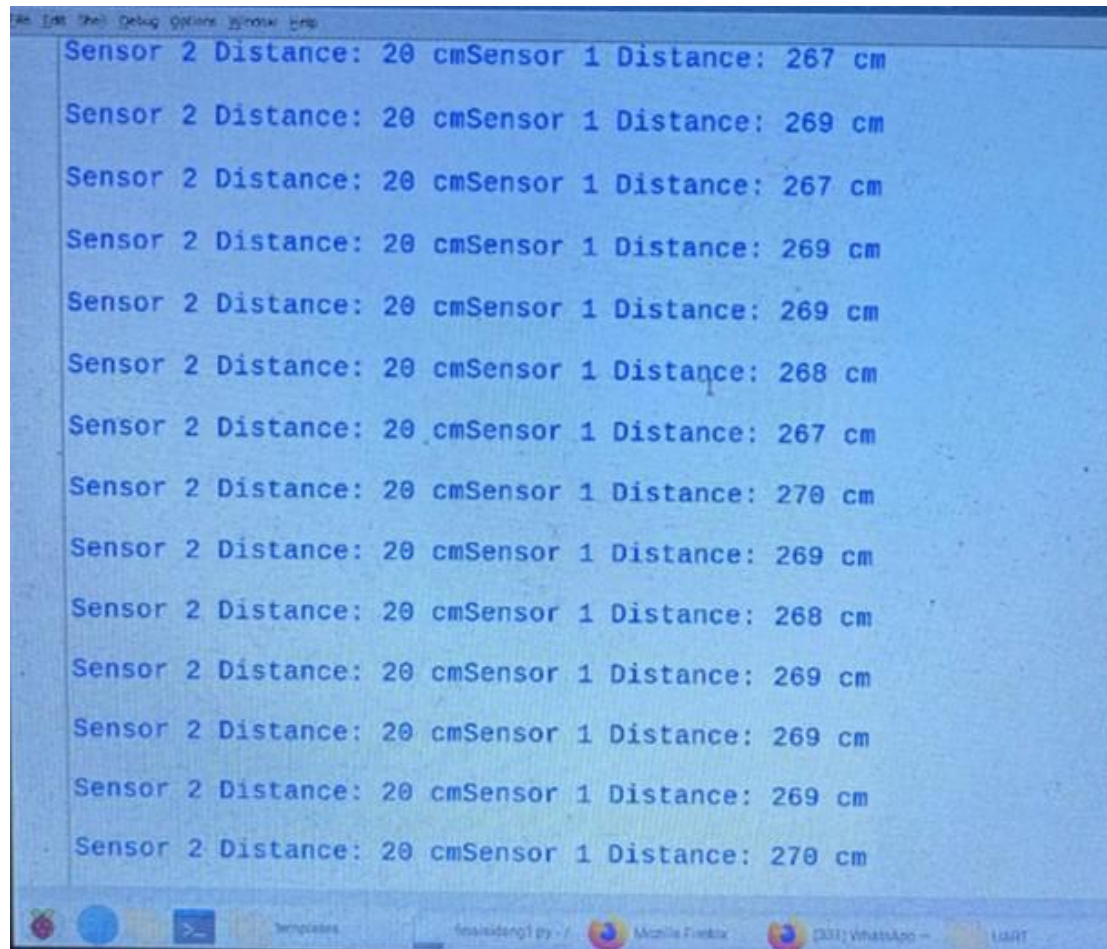
A screenshot of a terminal window with a blue background and white text. The window title bar at the top reads "File Edit Shell Debug Options Window Help". The terminal displays 15 lines of data, each showing the distance from two sensors. The text is as follows:

```
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 268 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 270 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 268 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 270 cm
```

The terminal window is part of a desktop environment. At the bottom, there is a taskbar with several icons: a Raspberry Pi logo, a blue circle icon, a terminal icon, a folder icon labeled "Templates", a file icon labeled "raspiideng1.py - /", a Mozilla Firefox icon, a WhatsApp icon labeled "(331) WhatsApp -", and a taskbar icon labeled "LIGHT".

Cont'd

Display of Distance Data in Shell Kernel



A screenshot of a terminal window with a blue background and white text. The window title bar at the top reads "File Edit Shell Debug Options Window Help". The terminal displays 15 lines of data, each showing the distance from two sensors. The text is as follows:

```
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 268 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 267 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 270 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 268 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 269 cm
Sensor 2 Distance: 20 cmSensor 1 Distance: 270 cm
```

The terminal window is part of a desktop environment. At the bottom, there is a taskbar with several icons: a Raspberry Pi logo, a blue circle icon, a terminal icon, a folder icon labeled "Templates", a file icon labeled "raspiideng1.py - /", a Mozilla Firefox icon, a WhatsApp icon labeled "(331) WhatsApp -", and a taskbar icon labeled "11:47".

Cont'd

TF-Luna LiDAR Sensor Test

- ▶ Real-world testin→During testing, the sensor successfully measured object distances, and the corresponding distance data were displayed on the LCD screen.
- ▶ When the measured distance meets the programmed warning condition, the system activates the buzzer to alert the driver to a potential collision hazard

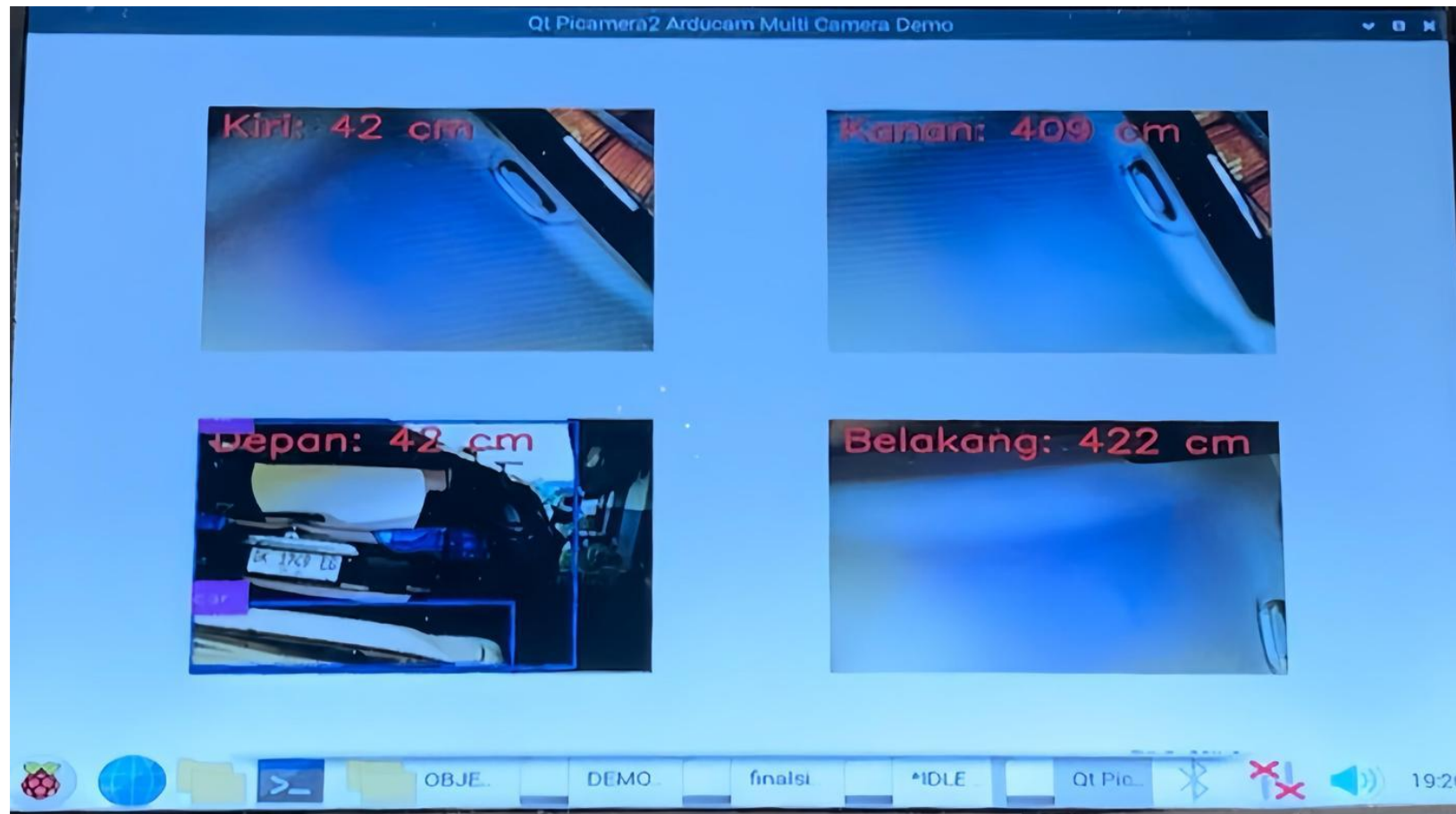
Cont'd

TF-Luna LiDAR Sensor Test

- ▶ Real-world testin→During testing, the sensor successfully measured object distances, and the corresponding distance data were displayed on the LCD screen.
- ▶ When the measured distance meets the programmed warning condition, the system activates the buzzer to alert the driver to a potential collision hazard

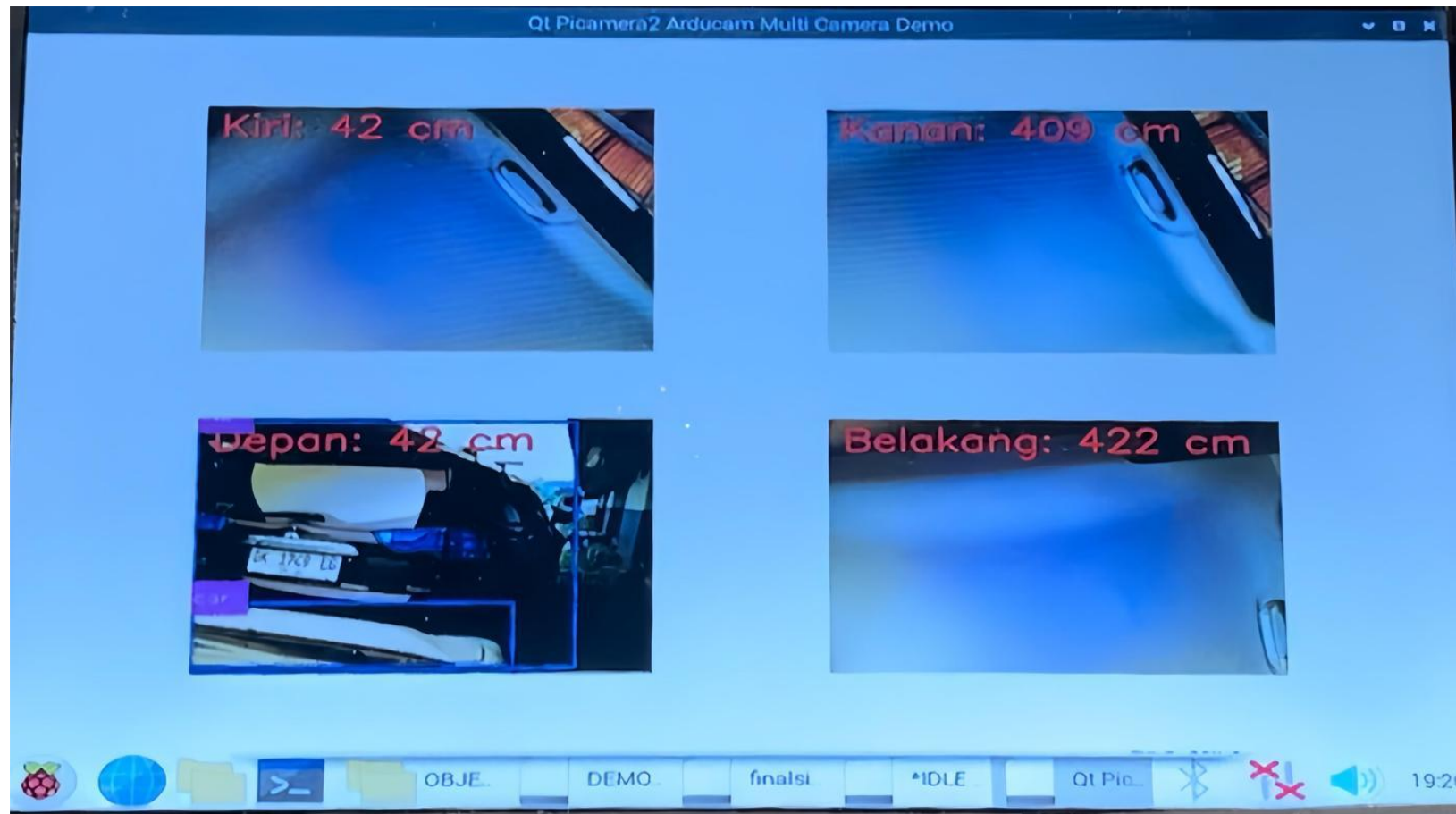
Cont'd

Sensor Tf-Luna LiDAR Module Result



Cont'd

Sensor Tf-Luna LiDAR Module Result



CONCLUSION

- ▶ The Raspberry Pi Camera, integrated with the YOLOv8 object detection model, was able to detect objects within the truck's blind spot areas, display bounding boxes, and assign labels to detected objects, such as motorcycles and cars.
- ▶ The TF-Luna LiDAR sensor was also able to measure object distances and provide distance data for the collision warning system. When an object entered the predefined danger zone, the system activated the buzzer to alert the driver to a potential collision hazard.
- ▶ The prototype demonstrated its ability to provide visual and audible warnings to support driver awareness of surrounding objects