CUSTECH-Journal of Innovation, Management and Social Sciences (CJIMSS)
Vol. 1, No. 1, April, 23, pp.252-264

Development of an Automatic Vehicle Number Plate Recognition (AVNPR) System for Security Management in Minna, Niger State.

[1]Yusuf Ndagi Baba, [2]Solomon Ndace Jiya & [3]Abubakar Ibrahim Damana

[1&3]Department of Transport Management, Ibrahim Badamasi Babangida University,
Lapai, Niger State, Nigeria.
[2]Department of Geography, Ibrahim Badamasi Babangida University,
Lapai, Niger State, Nigeria.
Contact: babayusuf89@gmail.com,
+2349030095139.

Article history

Received March. 26, 2026
Revised April, 09, 2026
Accepted April, 15, 2026

KEYWORDS: Insecurity, AVNPR system, deep learning, Artificial Intelligence, and Minna.

Abstract

The deteriorating security situation in Nigeria, particularly in Niger State, has become a major concern for both national and international communities. They often move in large groups using motorcycles and vehicles to transport weapons and abducted victims. Their activities have caused a lot of harm to both their victims and the economy of Niger State, specifically Minna, and the country's image at large. Many measures have been implemented to manage insecurity in Minna, but none have yielded the required results, as most measures were passive. Hence, the development of an active system like the AVNPR system for security management in Minna, Niger State. The design utilizes hardware components such as a Raspberry Pi board, a Pi camera module, a microSD card, and a network interface for its design. While the software elements utilized for the design include Pi Imager, Python 3, and OpenCV, among others. YOLOv11 was used to detect the number plate, while Tesseract OCR was used to extract the alphanumeric write-ups on the number plate region. The results show that the developed system achieved over 90% detection accuracy. In conclusion, the ANPR system developed is effective in detecting vehicle and alphanumeric information on the number plate, assisting law enforcement agencies in monitoring activities of unregistered vehicles within Minna.

Recommended citation: Baba .Y.N, Jiya S .N. & Damana A.I. Development of an Automatic Vehicle Number Plate Recognition (AVNPR) System for Security Management in Minna, Niger State., Vol. 1, No. 1, April, 23, pp.252-264

1 INTRODUCTION

Nigeria's overlapping security challenges vary from kidnapping to extremist insurgencies, and violence and criminality have spread to practically every part of the country (Tanko, 2021). The origins of Nigeria's insecurity issues may be traced back to Boko Haram activities in Borno State. Over time, Nigeria's worsening economy has resulted in rising security challenges. Minna, the capital of Niger State, is not exempt from this worsening security situation. An insurgency is a violent, armed insurrection launched by small, weakly armed bands from primarily rural bases against a bigger authority (Lewis, 2020).

Kocher (2011) proposed that the primary descriptive element of insurgency is its asymmetric nature: tiny irregular forces fight a strong, well-equipped, regular military force. Insurgent actions include banditry, kidnapping, terrorism, and armed robbery. Many factors contribute to insecurity in Nigeria, specifically in Niger State. According to Anigbuogu (2019), factors like poor institutional capacity, weak security systems, porous borders, ineffective leadership, political instability, rising unemployment, and poverty may cause insecurity or crime.

Insecurity in Nigeria, including Niger State, has resulted in the loss of thousands of lives and property damage. For instance, between January and March 2023, about 1,230 persons have been killed across Nigeria, with banditry accounting for the loss of 360 people, while 272 deaths were caused by Boko Haram (Global Right Nigeria, 2023). Farming communities in Niger State, including areas close to Minna, have suffered repeated attacks where bandits burn homes, loot food stores, and destroy farm equipment (Leadership Nigeria, 2022). Apart from this, the psychological and economic effects of insecurity on motorists are profound. Victims and their families often face devastating consequences, including trauma and financial burdens. Psychological impacts, such as stress, anxiety, and Post-Traumatic Stress Disorder (PTSD), are common among survivors (Olaojo & Owoniyi, 2022).

Many efforts have been made by the government and cooperative society to reduce the insecurity problems in Nigeria, particularly in Niger State, through technology deployment, fostering community participation, and promoting cooperation between communication service providers and security agencies (Ashaolu, 2012; McKendrick, 2019). In Minna, specifically, the state government introduced CCTV cameras at every road intersection in the year 2022, aimed at managing the security of the state (The Eagle Online, 2022). However, all effort hasn't been yielding the required results, as there is still growing concern of insecurity within Minna and the neighborhoods around Minna. Insecure activities are supported by transportation. They use motorcycles and cars in convoys to convey their weapons and victims.

To address this gap, the development of an Automatic Vehicle Number Plate Recognition System (AVNPRS) can be used to monitor the activities of each vehicle entering or exiting Minna, and with proper surveillance, insurgent activities can be tackled. The AVNPR system is a device that reads vehicle registration plates using Optical Character Recognition (OCR) on images to generate vehicle location data (Hamed, 2016). It can be integrated into the existing closed-circuit television, traffic enforcement cameras, or cameras created particularly for the purpose. According to Norizam et al. (2013), automatic vehicle number plate detection systems are widely employed in the field of safety and security systems, particularly in automobile parking areas and border control.

Recent innovation has expanded automatic vehicle number plate recognition system capabilities through machine learning, deep learning techniques, and the Internet of Things (IoT) framework. Studies by Abubakar et al. (2022) point out that the Internet of Things enabled a transport system utilizing Artificial Intelligence (AI) and image processing to demonstrate a feasible solution to transportation challenges and security enforcement. In another study, Omdena (2023) highlights the importance of computer vision in developing scalable vehicle identification platforms that support inspection and recognition tasks.

Aside from safety, this technology is used to monitor road traffic, such as vehicle speed and identification of the car's owner. This technology will allow for the prevention of law infractions while road users are on the road. This research project aims to create an AVNPR system for managing insecurity in Minna. The specific objectives of the project are as follows:

i. To design a single database system that permits vehicle recognition in Minna.
ii. To develop a hybrid vehicle number plate recognition system.
iii. To examine a system that allows for the effective transmission of vehicle information from one junction to another.

2 LITERATURE REVIEW

Extensive literature exists on automatic number plate recognition systems developed and implemented in countries such as France, the United States, China, and Russia. for the purpose of managing state security. In Iran, Hamed (2016) developed an AVNPR) system for vehicle identification. The system uses unique cameras to photograph each passing car and sends the image to a computer to be processed by the License Plate Recognition software. Number plate recognition software employs a variety of methods, including location, orientation, normalization, segmentation, and optical character recognition (OCR).

Also, the Ministerial Council of Europe developed a border security and management concept for the Organization for Security and Co-operation in Europe (OSCE) in 2005, recognizing that border security and management are matters of national sovereignty and responsibility of states. Anish and Prathap (2022) created an automatic vehicle number plate recognition technique utilizing neural networks. Norizam et al. (2013) developed an automatic vehicle number plate identification system combining image processing techniques and OCR to achieve accurate vehicle number plate recognition for vehicles in Malaysia. In a different project, Hai Kelia et al. (2020) created an automatic vehicle number plate recognition system in POLIMAS for an efficient automatic vehicle identification system by recognizing the vehicle plate number.

In 2003, the Washington State Department of Transportation (WSDOT) completed three integrated Intelligent Transportation Systems (ITS) deployments in collaboration with US Customs, the US Department of Agriculture, the ports of Seattle and Tacoma, marine carriers, trucking firms, and Canadian transportation agencies. These technologies are intended to identify, monitor, and preprocess cargo in intermodal containers trucked across the Washington, DC, border. Two of the systems are believed to use a normal, window-mounted vehicle transponder to offer commercial vehicle information, while the other uses a container door seal as a disposable transponder to provide security and cargo information.

Utilizing Raspberry Pi and open-source libraries, Eke et al. (2023) studies aimed at improving road safety and law enforcement through the development and implementation of a real-time automatic license plate number recognition system. The results show a 97.33% plate region extraction accuracy, while segment accuracy and recognition accuracy account for 95.33% and 98.67%, respectively. Though the design utilized affordable hardware, it had limited scalability and real-time performance under high vehicle speed conditions. Also, the research was mainly a design without real-life deployment.

Similarly, Dad et al. (2025) utilized Deep Learning (DL) technology, Raspberry Pi, Servo Motor (SM), an authorized plate number database, and a camera to develop a system with the capacity to detect and implement an Automated Vehicle Access Control System (AVACS). The essence of the design is to build a system that enhances security and optimizes vehicle entry in restricted areas. The findings reveal that the system recognized plate numbers under different conditions. The research differs from past designs, as it offers practical integration of hardware and software to solve real-life problems. The paper fails to address scalability and model performance issues under dense traffic conditions.

In a separate project, Shaikh et al. (2023) use the YOLO-V3 and V4 models created on the Darknet-53 CNN architecture for designing real-time item verification. The overall goal for the study is to build a system that automatically detects vehicle plate numbers with the goal of promoting vehicle tracking and law enforcement. The study used a data set of 1000 images from social media. The results point out that YOLOv3 has a mAP of 93.6% with an average loss of 24.43% over 4000 iterations, while YOLOv4 achieved a mAP of 97.7% with an average loss of 38.4% over 6000 iterations. Though the study demonstrates stronger utilization of Deep Learning (DL) for image detection and recognition, it fails to discuss deployment challenges.

Naveen (2023) utilizes MySQL to design and implement a vehicle management system, which is aimed at solving the problems of inefficiency in traditional vehicle management. The findings indicate that the system, built successfully, organizes and acquires vehicle-related data. Also, the SQL queries allow drivers to make decisions, particularly about vehicle fuel usage, maintenance cost, repairs, and sales. The research provides a comprehensive insight into vehicle operations, making it suitable for real-world decision-making. However, the lack of scalability is due to the restriction of the system to a single-user desktop environment.

Utilizing Optical Character Recognition (OCR), Tabhane et al. (2024) studies aimed at designing a real-time system capable of identifying vehicle plate numbers. The findings uncover that the built system successfully captures license plate numbers in real time. Though the research has a clear and simple architecture design, it lacks quantitative performance metrics like precision and accuracy of the design to detect the plate numbers.

Ugale et al. (2024) utilize gate environments like colleges, offices, and residential communities as possible areas where their design of an automotive vehicle identification system may be implemented for the purpose of enhancing security.

The study utilizes Arduino, optical character recognition, a real-time database, and a Flutter-based mobile app for the design. The design demonstrates high accuracy and detection in vehicle plate number detection in real-time, reducing manual checks. Also, the system concluded that real-life implementation may enhance control efficiency and improve security within the gate environment.

Dalvi et al. (2024) utilize computer vision and deep learning (DL) to develop and examine the automated plate number recognition system. The purpose of the system was to accurately identify and verify vehicle license plate number images under different conditions. Also, the system will enhance traffic enforcement, toll collection, and parking management. The results point to 85-90% accuracy being recorded on the front view of the still image, while about 80% accuracy was recorded in the angled image.

Latif and Yaba's (2022) project aimed at building a hybrid system for automatic number plate recognition in Iraqi vehicles. The system integrates image processing, machine learning (ML), and OCR to detect Arabic numbers and background color on the plate numbers. The outcome demonstrates that the background color detection of the plate number recorded about 97.8% success. Rao and Srinivas (2024) built a hybrid real-time vehicle plate number recognition system capable of image identification in diverse conditions, particularly poor lighting, dust, and skewed angles. The model shows accurate identification of vehicle plate numbers obscured by dust or containing written text. The hybrid system enhanced reliability and adaptability compared to the traditional system. The model may solve real-time challenges if implemented, but the restricted scalability and non-incorporation with a real-time database are the key weaknesses of the design. Improving the weakness could enhance the accuracy and real-life application of the developed model.

Yash and Neetha (2024) developed a robust and efficient vehicle plate number recognition system using ML techniques, specifically convolutional neural networks. The purpose of the design is to build a system with the capacity to capture accurately under various environmental conditions. The outcome of the convolutional neural network shows 92% accuracy, 90% precision, and 91% recall in the hybrid model.

This research is different from the existing system because the foregoing project will have the ability to recognize vehicle number plates, photograph the vehicle number plates, save the acquired data to the cloud, and transfer it from one point to another.

3 RESEARCH METHODOLOGY

3.1 Software requirement

The software tools and environments used to develop and deploy the car plate number recognition system on the Raspberry Pi are summarized and explained below:

i. Raspberry Pi OS (Buster): This is a Debian-based Linux distribution optimized for Raspberry Pi boards. It provides both a graphical desktop interface and a command-line environment for development. The Buster version offers stable support for Python 3, OpenCV, and TensorFlow-Lite—all essential for image processing and machine learning tasks used in this project.
ii. SD Card Formatter: This is Windows-based software used to properly format the microSD card before installing the Raspberry Pi operating system. It ensures the SD card is free from partition errors and ready for flashing, which helps prevent boot or corruption issues.
iii. Raspberry Pi Imager: This official Raspberry Pi Foundation tool was used to write the Raspberry Pi OS image onto the microSD card. It provides a simple graphical interface to select the OS, choose the storage device, and flash the image. It also allows pre-configuration of Wi-Fi credentials, SSH access, and hostname before the first boot.
iv. Python 3 (Thonny IDE): Python was the core programming language used for implementing the AVNPR logic. Thonny, the lightweight Python IDE preinstalled on Raspberry Pi OS, was used for writing, running, and debugging the code. It provides a user-friendly environment suitable for real-time system execution.

v. OpenCV (Open-Source Computer Vision Library): OpenCV was used for capturing video frames from the Pi camera, preprocessing images (grayscale conversion, noise reduction, and cropping), and visualizing detection results. It forms the foundation of the image handling process in the system.
vi. YOLOv11 License Plate Detection Model: The YOLOv11 (You Only Look Once, Version 11) model was employed for detecting number plates from captured images. Its deep-learning architecture allows real-time object detection with high accuracy and minimal latency on the Raspberry Pi 4 hardware.
vii. Tesseract OCR (Optical Character Recognition): Tesseract was used to extract alphanumeric text from the detected plate region. The OCR engine converts the cropped /plate image into readable text, which is then formatted and validated before being sent to the cloud database.
viii. Flask (Python Web Framework): Flask served as the communication bridge between the Raspberry Pi and the cloud web server. It handles HTTP requests and allows seamless transmission of recognized plate numbers, timestamps, and images to the online dashboard.
ix. VNC Viewer & SSH: VNC Viewer was used for remote desktop access to the Raspberry Pi without the need for a direct monitor. SSH (Secure Shell) was also enabled to provide command-line access for advanced configuration and debugging over the same network.
x. Advanced IP Scanner: This Windows-based utility was used to detect the Raspberry Pi's IP address on the network. The IP address is then used to establish a remote connection through VNC or SSH.
xi. Minna Plate Spotter Web Application: The processed results (number plate, image, timestamp) are transmitted to the Minna Plate Spotter Dashboard via the Flask API. The web platform displays all recognized vehicle entries and allows users to monitor or review past detections. Website link: https://minna-plate-spotter.vercel.app/

3.2 Hardware specification:

These are physical components or devices required for running the Automatic Vehicle Recognition System.

i. Raspberry Pi 4 Board: The main microprocessor responsible for running the ALPR system and executing the detection and OCR models.
ii. Raspberry Pi Camera Module: Captures real-time vehicle images for license plate recognition.
iii. MicroSD Card (32GB): Used as storage for the Raspberry Pi OS, application files, and captured image data.
iv. Network Interface (Wi-Fi/Ethernet): Enables communication between the Raspberry Pi and the remote web dashboard.
v. Power Supply Unit (5V/3A): Provides stable power to the Raspberry Pi and connected peripherals.
vi. External Display/Keyboard (optional): Used initially for configuration and debugging before enabling remote access

3.3 AVNPRS System Workflow

Figure 1 shows the step-by-step operational process of the designed AVNPRS. The flowchart describes how the developed system captures, processes, recognizes, and transmits vehicle data from image identification to final information storage and internet display. The major process in the flowchart is as follows:

i. System initialization (i.e., start): this stage represents the first step in the automatic number plate recognition, which starts by setting up the Raspberry Pi and camera module, allowing the hardware responsible for real-time image capturing to be activated. This setup enables the system to prepare itself for obtaining a vehicle image from a live scene at different road intersections or parking areas.

ii. Capture vehicle image: In the second stage, the camera module endlessly captures vehicle images passing at the road junctions. Conditional checks are carried out to confirm that vehicle images are successfully captured. However, if successful, the system proceeds to image preprocessing, while if not successful, the developed system automatically repeats image capturing until a clear edge of the plate number is obtained. The feedback mechanism shown in Figure 1 improves the consistency of data collection under different lighting or motion conditions.

iii. Image preprocessing: In this stage, the obtained image is subject to preprocessing activities such as resizing, grayscale conversion, and noise minimization. In this step, image qualities are standardized and detection accuracy is improved by abating distortion, reflection, and unnecessary background data.

iv. License Plate Detection (LPD) using YOLOv11 Model: After the preprocessing of the obtained vehicle image, the captured image is passed through the YOLOv11 Object Detection Model (ODM), allowing identification and localization of the number plate region within the vehicle image. The flowchart in Figure 1 indicates that if a number plate is detected, the edge is cropped for further processing, and if not, it discards and recaptures the image, ensuring only important features of the image are analyzed. This stage forms the key aspect of the recognition process, relying on artificial intelligence using deep learning in achieving high precision and speed.

v. Plate region cropping: Once the vehicle number plate is detected, the recognized region is cropped from the entire image, and the isolated areas are only analyzed. This allows a decline in computational overhead and removes unnecessary parts of the image.

vi. Apply OCR for text extraction: In this stage, the cropped image is further processed to extract text using an OCR engine, particularly using Easy OCR to extract the alphanumeric text on the number plate. However, if the number plate texts are successfully extracted, the system proceeds to formatting of text, and if unsuccessful, it logs the image as unreadable and requests a manual review or reprocessing of the image.

vii. Format extracted text and metadata attachment: In this stage, the successfully extracted number plate text in the previous stage is cleaned and standardized by removing noise and inconsistency, such as spacing and special characters. The system now attach metadata such as timestamp, road junction, and device ID.

viii. Send data and image via Flask API: In this stage, the process number plate information and corresponding images are sent through a Flask API to the cloud database. When the data and image transmission is successful, the information is uploaded, and if not, the system stores the data in a local record for retrying once internet connectivity is restored. Thereby preventing data loss due to internet failure.

ix. Upload to cloud database: Once the Flask API is received, the information stored in the cloud database is the number plate, timestamp, junction ID, image URL, and confidence score. The centralized storage underpins efficient data handling and remote monitoring.

x. Information display and web integration: after successful uploading of the data on the cloud, the recognized number plate is automatically displayed on the website (i.e., the Minna Plate Spotter Website), which serves as the web dashboard. In this stage, the detected number plate number, captured image, location, and time of capture are recorded. The web platform enabled real-time vehicle number plate viewing and monitoring, allowing security agencies to effectively manage security challenges in Minna.

xi. Allow users to search and filter: The web dashboard is designed with search and filter tools to allow users to quickly search a particular record based on number plates, time, or junction. These functions support assessment, surveillance, and authentication of suspicious or stolen cars.

xii. End: Once the processing and displaying of the recorded number plate was successful, the designed system automatically returned to the start stage, ready to capture the next vehicle information and repeat the cycle continuously uninterrupted.

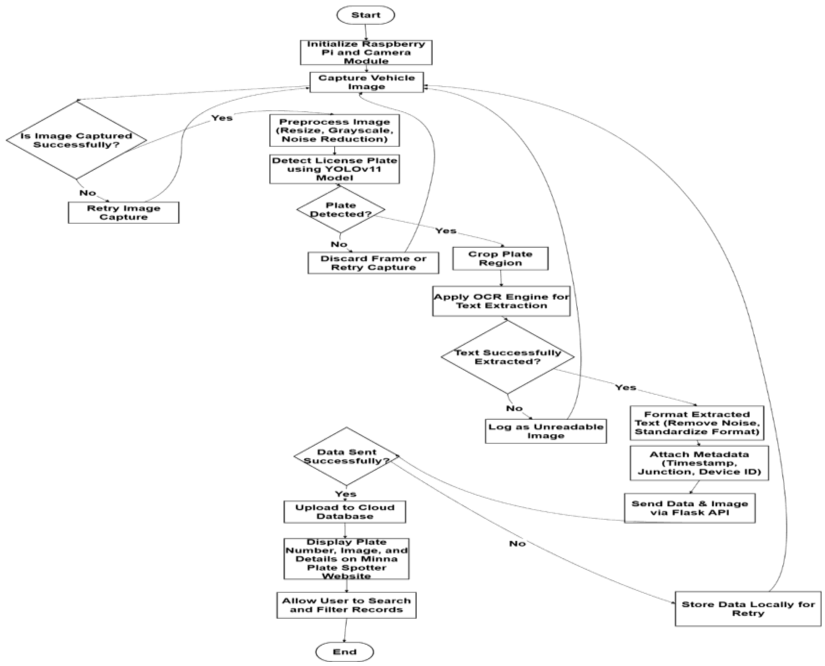

Figure 3.1: System architecture diagram illustrating the workflow from camera input to web visualization.
(**Source:** Authors' Survey, 2025.)

4 **RESULTS AND DISCUSSION**

4.1 System setup and preliminary configuration

The project execution stage involves the establishment of a working environment for Raspberry Pi 4 (8 GB RAM). The Pi camera module used for the design assists in real-time image acquisition. The operating environment for the system designed is as follows:

a) Successfully connected to Raspberry Pi via SSH and enabled VNC for GUI access.
b) Enabled the Camera Interface using raspi-config.
c) Updated and upgraded Raspberry Pi OS.
d) Installed essential tools: git, cmake, build-essential, pkg-config, and Python libraries opencv-python, EasyOCR, numpy, and imutils; installed Tesseract OCR for text recognition.
e) Verified camera functionality with test images.

However, the Pi camera was accessed remotely utilizing SSH and VNC, while the camera interface was aided by the raspi-configuration. In addition, the camera test was carried out to confirm proper image acquisition and frame stability. The configuration provides a framework for successive data collection and model deployment.

4.2 Dataset preparation

Figure 4.1 shows the various number plates acquired for the model. In order to achieve a reliable and diverse dataset to aid the recognition model in generalizing effectively to different Nigerian number plate styles, colors, and lighting conditions.

Over 10,000 images of Nigerian vehicle license plates were collected from multiple sources, including locally captured images and open datasets. The dataset was divided into three main categories: 7,000 images for training, 2,000 for testing, and 1,000 for validation. Each image was carefully annotated using Roboflow, where bounding boxes were

CUSTECH-Journal of Innovation, Management and Social Sciences (CJIMSS)
Vol. 1, No. 1, April, 23, pp.252-264

drawn around each license plate region. The dataset was then exported in YOLO format, which contains the class ID and normalized coordinates for every bounding box.

Data preprocessing was performed to enhance the quality and variability of the dataset. Each image was resized to 640 × 640 pixels, and adjustments were made to improve contrast and brightness. Random rotations, flips, and blurring augmentations were applied to ensure robustness of the model under real-world conditions such as glare, motion blur, or varying distances.

The training environment was set up using Google Colab with GPU acceleration on a Tesla T4 processor. Required packages such as ultralytics, torch, torchvision, and opencv-python were installed. The dataset was uploaded to Google Drive and connected to Colab to facilitate easy access during the training phase. Initial test scripts were executed to verify that all data files were correctly loaded and labeled without corruption or errors.

The authors concluded with a fully prepared and verified dataset ready for model training. The complete dataset was prepared, annotated, and verified for accuracy (see figure 4.4). The training environment was successfully configured, providing a solid foundation for model training in subsequent section.

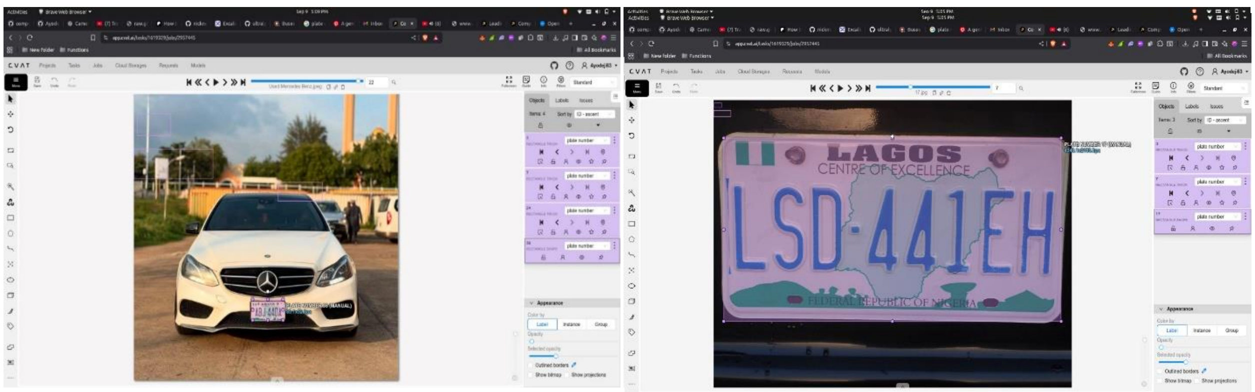

Figure 4.1: Samples of collected Nigerian license plate images showing variations in plate design.
Source: Authors' Survey, 2025.

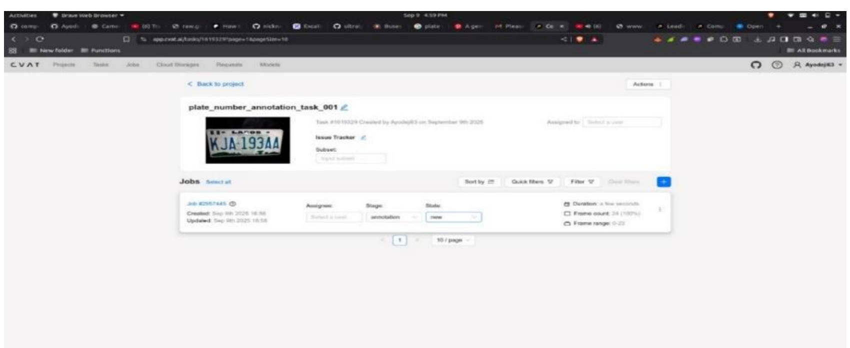

Figure 4.2: Annotated dataset in Roboflow highlighting bounding boxes around plates.
Sources: Authors' Survey, 2025.

4.3 Model training

In this section, YOLOv11 was trained for object detection accuracy, particularly adequate detection of vehicle plate numbers. The training parameters include image size of 640 × 640 pixels, batch size of 16, Learning Rate (LR= 0.01), AdamW optimizer, momentum (0.937), Training Epochs of 3, and GPU mixed precision. The model training was done on the prepared data sheet, and the weight file was saved at the end of each epoch. The most performing weights were stored as best.pt file following the system integration. In order to monitor training performance, YOLO's built-in visualization tools were used for tracking convergence and stability. However, the figure 4.4 shows the model training.

CUSTECH-Journal of Innovation, Management and Social Sciences (CJIMSS)
Vol. 1, No. 1, April, 23, pp.252-264

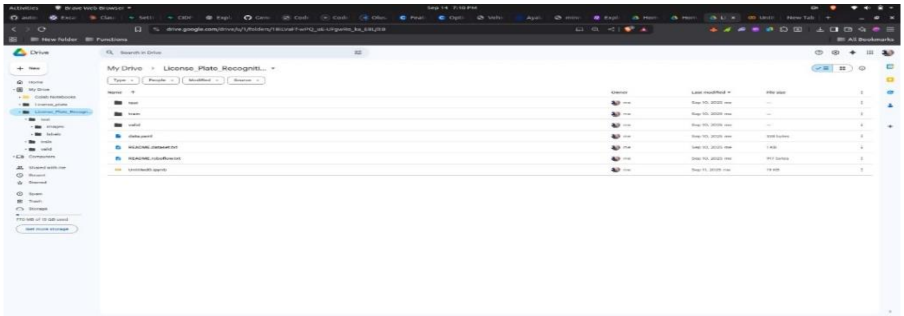

Figure 4.3: Folder structure of training, testing, and validation sets in Google Drive.
Source: Authors' Survey, 2025.

4.4 Training performance

The model performance improved with each epoch, showing a consistent decrease in loss values and an increase in accuracy (see figures 4.4 and 4.5). The recorded metrics were as follows:

i. Epoch 1: mAP = 0.871, runtime = 1 hour 30 minutes
ii. Epoch 2: mAP = 0.903, runtime = 20 minutes
iii. Epoch 3: mAP = 0.940, runtime = 2 minutes 28 seconds

The gradual reduction in training time reflected effective optimization and GPU acceleration. The best model weights were saved automatically as best.pt, representing the optimal version of the trained model. After training, the model was validated using the reserved dataset, achieving a precision of 95.6%, recall of 90.8%, and an overall mAP of 94%. Performance visualization from the YOLO logs confirmed stable training behavior and accurate predictions. The model showed clear convergence, with box loss and classification loss significantly minimized. The model achieved over 90% detection accuracy for Nigerian license plates, demonstrating strong generalization ability. The trained model weights were exported for further testing and real-world deployment.

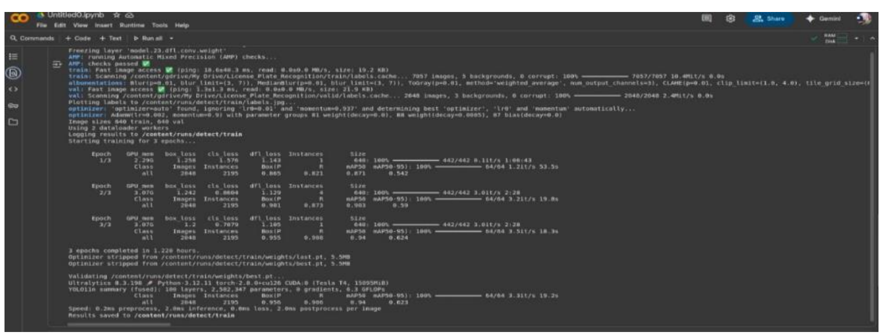

Figure 4.4: Screenshot of YOLOv11 training progress in Google Colab
Source: Authors' Survey, 2025.

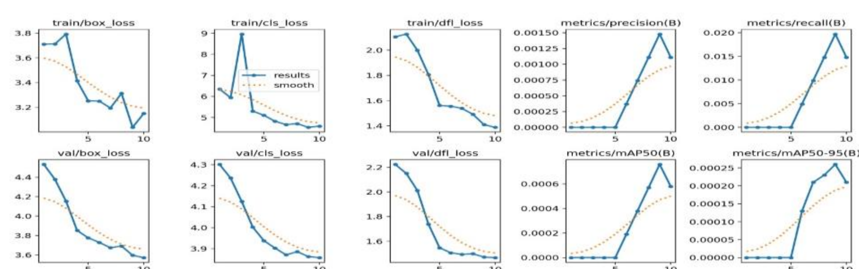

Figure 4.5: Training loss and accuracy curves showing model improvement over epochs, GPU performance, and

CUSTECH-Journal of Innovation, Management and Social Sciences (CJIMSS)
Vol. 1, No. 1, April, 23, pp.252-264

memory utilization during training.
(**Source:** Authors' Survey, 2025.)

4.5 Object Character Recognition (OCR) integration, deployment, and web dashboard design

Following the model training in the previous section, OCR integration, deployment, and a web dashboard were developed. Using the EasyOCR, the alphanumeric text on the plate number was extracted (see figure 4.7). Also, data visualization apparatus for real time were developed. The procedures used in achieving this stage are as follows:

i. Integration of EasyOCR to extract alphanumeric text from a detected plate number.
ii. Implementing python program where each apprehended frame was handled chronologically: vehicle image capture → plate number detection → text Abstraction
iii. Deployment of the trained YOLO models and object character recognition module into the Raspberry Pi for edge processing, allowing real-time plate number detection and recognition.
iv. Designing a flake-based backend Application Programming Interface (API) to transmit recognized data to an online database.
v. Developing and hosting a web dashboard on vercel: titled Minna Vehicle Recognition (MVR) portal enabling users to view captured data including timestamp, location, junction details and snapshots.
vi. Conducting system performance test to confirm proper vehicle information transmission, synchronization and dashboard updates.

Figure 4.6: Showing text extraction from detected plate region
Source: Authors' Survey, 2025.

4.6 Web dashboard development

Figure 4.7 shows the screenshot of Minna Vehicle Recognition Portal showing detected plate numbers and capture details. The web dashboard developed allows the authorized users to visualized recognized vehicle data on the webpage titled ''Minna Vehicle Recognition Portal'' (see figure 4.7). The dashboard was designed and hosted on Vercel and serves as central point for vehicle number plate recognition and monitoring. The main features for the developed dashboard include real-time display of detected plate numbers, date and time of detection, capture junction or location, corresponding cropped image of the vehicle and search and filter tools for querying vehicle records. Also, the web application was design to connect with Flash backend API which send the recognized vehicle data from raspberry pi to the cloud. The system therefore, automatically update for newly detected image and displays in real time for security management.

CUSTECH-Journal of Innovation, Management and Social Sciences (CJIMSS)
Vol. 1, No. 1, April, 23, pp.252-264

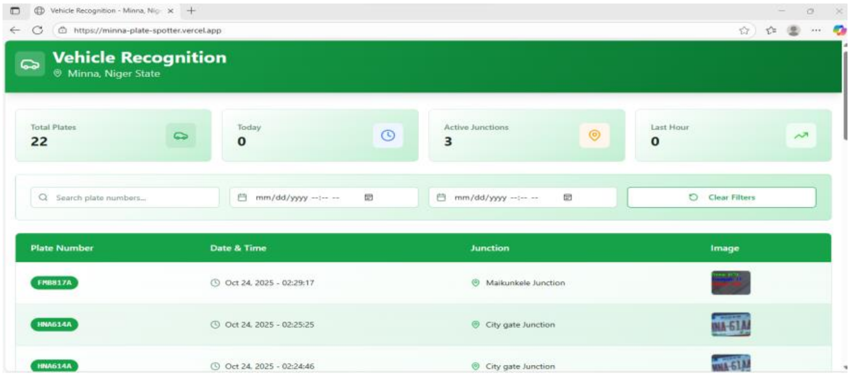

Figure 4.7: Screenshot of Minna Vehicle Recognition Portal showing detected plate numbers and capture details
(**Source:** Authors' Survey, 2025.)

4.7 System architecture and database design

In implementing AVNPRS, the study used a five-layer framework, which includes the following:

i. Image capturing layer: This stage utilized raspberry pi and a camera module to capture a motor vehicle image.
ii. Detection layer: The study used the YOLOV11 model to detect the vehicle license plate number region.
iii. Recognition layer: The third layer uses EasyOCR to interpret alphanumeric texts on the plate number.
iv. Communication layer: In this layer, the study used Flask API to transmit vehicle data to the cloud.
v. Visualization layer: This is the last layer of the system architecture, which uses a web dashboard to display detected plate numbers and their details.

4.8 System Testing and Validation

The entire system developed was subject to system testing in order to confirm its operational performance. The test was conducted at different road intersections/junctions in Minna, specifically Chanchaga, City Gate, Kpakungu, and Dutsen Kura Junctions. This enabled the authors to confirm the functionality of the camera, model detection, object character recognition accuracy, and network transmission. Similarly, the testing ensured that all the vehicle images acquired by the modules are visualized on the web and are functioning as designed.

5 CONCLUSIONAND RECOMMENDATION

This study on the development of an automatic number plate recognition system for security management involves developing a system with a single database system that permits vehicle recognition, developing a hybrid vehicle number plate recognition system, and building a system that allows for the effective transmission of vehicle information from one junction to another. The results shows that the developed system have the ability to detect and extract alphanumeric digits in the plate number region with a mAP of 94%. The study concluded by recommending that key road intersections in Minna, particularly City Gate, Kpakungu Roundabout, and Dutsen Kura Junction, among others, should be equipped with this system. This will assist in monitoring vehicles in and out of Minna, and vehicle activities with unrecognized number plates can be closely monitored in and out of Minna.

5.1Acknowledgement

This research was supported by Tertiary Education Trust Fund (TETFund) through Institution Based Research (IBR Batch 9-1: 2023/2024) Grant of Ibrahim Badamasi Babangida University, Lapai, Niger State, Nigeria..

REFERENCES

Abubakar, M., Abdulkadir, A., & Ibrahim, A. (2022). Artificial intelligence and Internet of Things for smart transportation systems: Applications and security enforcement. In Proceedings of the 2022 International

Conference on Artificial Intelligence of Things and Crowdsensing (AIoTCs) (pp. 1–6). IEEE. https://doi.org/10.1109/AIoTCs55057.2022.00009

Anish, T.P., Prathap, P.M.J. (2022). Automated Vehicle Number Recognition Scheme Using Neural Networks. In: Singh, M., Tyagi, V., Gupta, P.K., Flusser, J., Ören, T. (eds) Advances in Computing and Data Sciences. ICACDS 2022. Communications in Computer and Information Science, vol 1613. Springer, Cham. https://doi.org/10.1007/978-3-031-12638-3_23

Ashaolu, D. D. (2012). Solving security challenges in Nigeria through intelligence gathering and surveillance. Available at http://dx.doi.org/10.2139/ssrn.2275986. Retrieved March 3, 2023

Dad, S., Pandit, N., Pardeshi, A., Shaikh, A., & Sumbe, O. (2025). Design and Implementation of Vehicle Access Control Using License Plate Recognition. *International Research Journal of Modernization in Engineering Technology and Science*, 7(1). https://doi.org/10.56726/IRJMETS66038

Dalvi, A., Fernandes, J., Toms, C., Saju, B., & Patil, U. (2024). Automatic Number Plate Recognition System. *International Journal of Creative Research Thoughts*, 12(4). https://www.ijcrt.org/papers/IJCRT24A4424.pdf

Eke, R. A., & Orakwue, S. I. (2023). Automatic License Plate Recognition System. *IRE Journals*, 6(10). ISSN: 2456-8880.

Global Rights Nigeria. (2023). *Mass atrocities casualties tracking report: Q1 2023*. Retrieved from https://globalrights.org

Hai Kelia V., Din Keraf, N., Darus, H., Mohd Noor, N., Loke Yue Nie, E., and Ehkan, P., (2020). Proposal for automatic vehicle number plate recognition system in POLIMAS. Journal of Engineering and Science Research 4 (1): 20-27

Hamed S. (2016). Proposal for automatic license and number plate recognition System for vehicle identification. International Conference on New Research Achievements in Electrical and Computer Engineering, 1, pp 1-5

Kocher, Matthew Adam; Pepinsky, Thomas B.; Kalyvas, Stathis N. (2011). "Aerial Bombing and Counterinsurgency in the Vietnam War". *American Journal of Political Science*, 55 (2): 201–218. doi:10.1111/j.1540-5907.2010.00498

Latif, H. O., & Yaba, H. H. (2022). Plate number recognition based on hybrid techniques. *UHD Journal of Science and Technology*, 6(2), 39–48. https://doi.org/10.21928/uhdjst.v6n2y2022.pp39-48

Leadership Nigeria. (2022, August 15). *Rising wave of banditry: Niger State farming communities under siege*. Leadership Newspaper. https://leadership.ng/rising-wave-of-banditry-niger-state-farming-communities-under-siege/

Lewis, Janet I. (2020). How Insurgency Begins: Rebel Group Formation in Uganda and Beyond. Cambridge University Press. pp. 17–18. doi:10.1017/9781108855969

McKendrick, K. (2019). Artificial intelligence prediction and counter terrorism. London, England: Institute of International Affairs Chatham House.

Naveen, R. (2023). Creating a Vehicle Management System through MySQL. *International Journal of Research Publication and Reviews*, 4(8), 3259–3266. https://www.ijrpr.com

Norizam S., Sri Nor Hafidah M. J., Mahfuzah M., and Kamarul H., (2013). Development of automatic vehicle plate detection system. 3rd International Conference on System Engineering and Technology, 19 – 20

Olaojo B. O. & Awoniyi G. F. (2022). Analysis of the impact of armed banditry on Nigeria's economic growth and national development. *Counselling Implication, Journal of African Social Studies (JASS)*, 3(1), 149-162.

Omdena. (2023). Building a vehicle recognition and inspection system using computer vision. Omdena AI Project. Retrieved from https://www.omdena.com/projects/buiding-a-vehicle-recognition-and-inspection-system

Rao, P. N., & Srinivas, Y. (2024). Real time number plate recognition system using hybrid models. *International Journal of Scientific Development and Research*, 4(8). https://www.researchgate.net/publication/377151627

Shaikh, M. S., Nilekani, Y., Shinde, H., & Varma, S. L. (2023). Automatic number plate detection and recognition using deep learning. *International Journal of Research and Analytical Reviews*, 10(2). https://www.ijrar.org/viewfull.php?&p_id=IJRAR23B1081

Tabhane, S. S., Khodaskar, A. A., Sonkusare, T. J., Sinkar, I. R., Sapkal, S. G., & Zagade, Y. N. (2024). Moving Vehicle Registration Plate Detection System Using OCR. *International Research Journal of Engineering and Technology (IRJET)*, 11(4). https://www.irjet.net

The Eagle Online. (2022, June 14). *Niger State Government installs CCTV cameras at road intersections to boost security.* The Eagle Online. Retrieved from https://theeagleonline.com.ng/tag/cctv-cameras/

Ugale, S., Dange, S., Patil, U., Jarhad, K., & Jadhav Patil, A. (2024). Real time vehicle detection and authentication system. *International Journal of Engineering Technology Research & Management*, 8(11), 59–62. https://www.ijetrm.com/

Yasha, B. P., & Neetha, S. S. (2024). Enhanced vehicle identification: A machine learning approach to number plate recognition. *International Journal of Research Publication and Reviews*, 5(6), 2501–2505. https://doi.org/10.55248/gengpi.5.0624.1502