# AUTOMATIC LICENSE NUMBER PLATE DETECTION AND RECOGNITION

Arup Chakraborty

Shiley-Marcos School of Engineering, University of San Diego

**Abstract**

This study presents a robust Automatic License Plate Recognition (ALPR) system that integrates object detection and Optical Character Recognition (OCR) techniques. Using YOLOv10, a state-of-the-art deep learning model for real-time object detection, the system achieves high accuracy in recognizing license plates under diverse conditions. The pipeline includes pre-trained model loading, fine-tuning on a custom dataset, and performance evaluation, ensuring the system's adaptability to various use cases. Additionally, the integration with pytesseract enhances text extraction capabilities from detected plates, enabling efficient data retrieval. To facilitate user interaction, a Gradio-based web interface is developed for image uploads and live video processing. The final model is deployed to Hugging Face for broader accessibility, showcasing an end-to-end solution for real-time vehicle identification with potential applications in traffic management, security, and automation.

.

*Keywords*: YOLO, image processing, deep neural network, number plate recognition

Automatic number plate recognition

**Brief Overview of identifying music composer**

This project focuses on developing an Automatic License Plate Recognition (ALPR) system using advanced machine learning and computer vision techniques. The solution incorporates YOLOv10 for real-time license plate detection and pytesseract for extracting alphanumeric characters from detected plates. Key features include model fine-tuning on custom datasets, performance validation, and user-friendly interaction through a Gradio-based web interface. The system also supports video frame processing for moving vehicles and is deployed on Hugging Face for scalability, making it suitable for applications in traffic monitoring, law enforcement, and automated parking systems.

**Create Model to Detect & Classify car number plate using Kaggle Dataset**

**Data Selection**

This project introduces a robust pipeline for automatic detection and recognition of car license plates using a Kaggle dataset. Leveraging YOLOv5 for plate detection, the system achieves high accuracy in identifying license plates from images under various conditions. The pipeline integrates Optical Character Recognition (OCR) using pytesseract to extract alphanumeric characters from detected plates, enabling efficient and precise license plate recognition. Key steps include dataset preprocessing, training and fine-tuning a detection model on a custom dataset and deploying the model for real-time applications through a user-friendly interface. The solution is designed to address challenges such as low-resolution images and varied lighting conditions, making it suitable for real-world applications like traffic surveillance, parking management, and law enforcement. The system demonstrates scalability and effectiveness, providing an end-to-end solution for vehicle identification.

## EDA (Exploratory Data Analysis)

### Data Acquisition and Preparation

This work outlines the process of data acquisition and preparation for developing an Automatic License Plate Recognition (ALPR) system. A publicly available Kaggle dataset, consisting of vehicle images with annotated license plates, is utilized to ensure diverse and representative data. The data preparation pipeline includes organizing the dataset into training, validation, and test sets, preprocessing images to standardize dimensions and enhance features, and converting annotations into a YOLO-compatible format for efficient object detection model training. Special attention is given to data augmentation techniques, such as rotation, scaling, and contrast adjustments, to improve model robustness against varied real-world scenarios.

Kaggle dataset https://www.kaggle.com/code/aslanahmedov/automatic-number-plate-recognition/input

### Data Exploration and Preprocessing

This study focuses on data exploration and preprocessing as foundational steps for developing an Automatic License Plate Recognition (ALPR) system. Comprehensive data exploration is conducted to understand the dataset's structure, distribution, and potential challenges, such as image quality, inconsistent annotations, and class imbalances. Preprocessing techniques are employed to enhance data suitability for machine learning tasks, including resizing images to standard dimensions, normalizing pixel values, and converting annotations into formats compatible with YOLOv5 for object detection. Advanced preprocessing steps, such as data augmentation (e.g., rotation, flipping, and brightness adjustments), are applied to improve model robustness and generalization. Outlier detection and handling ensure data integrity, while splitting the dataset into training, validation, and test subsets facilitates rigorous model evaluation. This structured approach ensures high-quality inputs, optimizing the performance and reliability of subsequent machine learning models.

**Model Development**

This research presents the development of an Automatic License Plate Recognition (ALPR) system using advanced machine learning and computer vision technologies. The system leverages YOLOv10, a state-of-the-art object detection framework, for accurate and real-time detection of license plates in diverse environments. For character recognition, pytesseract is integrated to extract alphanumeric data from detected plates with high precision. Key steps in the model development include fine-tuning YOLOv10 on a custom annotated dataset, implementing data augmentation to enhance model robustness, and optimizing hyperparameters for improved performance. The system is designed with scalability in mind, employing tools such as Gradio for a user-friendly interface and Hugging Face for deployment. This research also involves with RESNET deep networking model to predict the bounding box for car number plates.

This comprehensive approach results in a highly efficient ALPR model suitable for applications in traffic management, security systems, and automated parking solutions, demonstrating significant accuracy and reliability in real-world scenarios.

## Train Model and Evaluation

### Training and Validation Setup

The training and validation setup is a crucial component in the development of a deep learning model aimed at accurately identifying the car number plates. This phase involves carefully structuring the training process to ensure the model effectively learns to distinguish between different images of car types. The project employs a well-defined dataset, divided into training, validation, and test sets, ensuring a balanced representation of compositions across all phases. Key considerations include the selection of appropriate loss functions, optimization algorithms, and evaluation metrics that align with the project's goals. The validation setup is designed to monitor the model's performance and prevent overfitting, employing techniques such as cross-validation and early stopping. Tensorboard has been used to monitor the model performance.

By meticulously calibrating the training and validation process, this project aims to produce a robust and generalizable model capable of high accuracy in composer identification, contributing to advancements in music technology and analysis.

### Model Training and Visualization

## Automatic number plate recognition

This study focuses on the model training process for an Automatic License Plate Recognition (ALPR) system, utilizing advanced algorithms to achieve high accuracy and efficiency. The YOLOv10 (You Only Look Once, Version 10) algorithm is employed for license plate detection, leveraging its real-time performance and ability to detect objects with high precision. The model is fine-tuned on a custom dataset annotated with bounding boxes, using transfer learning to optimize training time and performance.

**Model Architecture and Compilation**

This study details the model architecture and compilation process for an Automatic License Plate Recognition (ALPR) system designed to detect and recognize vehicle license plates with high accuracy and efficiency. The architecture is based on YOLOv10 (You Only Look Once, Version 10), a state-of-the-art object detection model, which provides a robust backbone for real-time detection. YOLOv10 employs a Convolutional Neural Network (CNN) framework optimized with a CSPDarknet53 backbone and PANet for feature extraction and aggregation. For Optical Character Recognition (OCR), pytesseract is integrated into the pipeline, utilizing Tesseract's efficient character segmentation and adaptive recognition capabilities. The compilation process involves configuring the model for GPU acceleration, setting a binary cross-entropy loss for detection, and using an Adam optimizer with an adaptive learning rate schedule to improve convergence. This architecture is designed to handle diverse and challenging scenarios, including low-resolution images and varied lighting conditions, resulting in a scalable and efficient ALPR system suitable for real-world applications in traffic monitoring, security, and automated systems.

## Results

### Model Performance Evaluation

The performance evaluation of the model was depicted in Figure 1 showcasing the model's accuracy and loss across training epochs. These metrics offer insights into how the model adapted to data and learned from its training set. Key metrics generated during the training process included plots of accuracy, and validation loss and training progress for determining when to stop training to prevent overfitting.

### Model Evaluation Metrics

Following this, the model's effectiveness and robustness was extensively evaluated using the training, validation and testing datasets. As seen in Figure 2, the confusion matrix, Figure 3, provides comprehensive insights into the model's performance across several tumor classifications, emphasizing both its advantages and disadvantages.

## Discussion

### Interpretation of Results

As seen in Figure 2, the model was able to detect the car license plate with high accuracy, demonstrating efficient learning without overfitting. The confusion matrix showed that although the model performed well in recognizing some tumor types, it had difficulties with others, indicating the need for more research and model improvement.

### Comparison with Existing Methods

This research also compares the YOLO model with UNet and RESNET model.

### Observations

Automatic number plate recognition

This study provides a comparative analysis of three popular deep learning architectures—YOLO, ResNet, and U-Net—evaluated for their suitability in Automatic License Plate Recognition (ALPR) and similar object detection and segmentation tasks. YOLO (You Only Look Once) excels in real-time object detection, offering high speed and accuracy, making it an ideal choice for detecting license plates in real-time traffic environments. ResNet (Residual Network), known for its deep architecture and skip connections, is utilized for feature extraction and classification, demonstrating robustness in handling complex data but at the cost of higher computational overhead. U-Net, a convolutional neural network originally designed for medical image segmentation, is adapted for tasks requiring precise localization and segmentation of license plates or characters. The architectures are assessed based on metrics such as detection accuracy, processing speed, and resource efficiency. The results highlight the trade-offs between speed and precision, showcasing YOLO's dominance in real-time scenarios, ResNet's reliability for detailed classification tasks, and U-Net's strength in pixel-level segmentation. This comparative analysis provides insights into selecting the appropriate architecture based on specific ALPR requirements and operational constraints.

**Model Hosting via Huggingface**

Use Case 1: Enhanced Traffic Management: The ALPR system can be integrated into traffic management systems to monitor vehicle flow, enforce traffic rules, and reduce congestion. Enhancements, such as multi-lane detection and real-time processing, will improve scalability and efficiency. Link [https://huggingface.co/spaces/arupchakraborty2004/image-recognition-yolo](https://huggingface.co/spaces/arupchakraborty2004/image-recognition-yolo)

Use Case 2: Security and Surveillance: In security applications, the system can be deployed for vehicle tracking, stolen vehicle identification, and automated gate control. Incorporating night vision capabilities and advanced anomaly detection algorithms will enhance its effectiveness. Link https://huggingface.co/spaces/arupchakraborty2004/image-recognition-yolo

Use Case 3: Toll Automation and Parking Management: ALPR can streamline toll collection and parking management processes by automating vehicle entry, exit, and fee computation. Future work can focus on integration with payment systems and ensuring accuracy in high-traffic environments.

Link https://huggingface.co/spaces/arupchakraborty2004/number-plate-detection-moving-vehicle

**Future Enhancements**

This study concludes with potential future enhancements to further improve the efficiency and accuracy of Automatic License Plate Recognition (ALPR) systems. Future work aims to explore advanced deep learning models such as YOLOv11 and Vision Transformers (ViTs) for improved object detection and recognition capabilities. Incorporating advanced Optical Character Recognition (OCR) techniques, such as Transformer-based architectures, could enhance the system’s ability to handle complex fonts and noisy data. Real-world adaptability will be improved by developing robust solutions for challenges like motion blur, occlusions, and adverse weather conditions.

Automatic number plate recognition

Integrating edge computing and IoT technologies will enable real-time processing on low-resource devices for scalable deployment in smart cities.

Furthermore, a multilingual license plate recognition module can be developed to handle region-specific characters and scripts.

Expanding the dataset with more diverse images and annotations, alongside semi-supervised learning techniques, could further enhance model robustness and generalization.

These enhancements aim to push the boundaries of ALPR applications in traffic management, security, and automation.

## Conclusion

In conclusion, this study successfully developed and evaluated a robust pipeline for Automatic License Plate Recognition (ALPR) using advanced machine learning models and techniques. By leveraging YOLOv10 for real-time license plate detection and integrating pytesseract for Optical Character Recognition (OCR), the system achieved high accuracy and efficiency under diverse conditions. Comprehensive data preprocessing, model fine-tuning, and deployment strategies contributed to its scalability and adaptability in real-world applications. The findings demonstrate the system’s potential for use in traffic monitoring, security, and automation. Future enhancements, including advanced models and multilingual support, will further optimize performance and broaden applicability. This work underscores the importance of combining innovative algorithms and robust workflows to address complex challenges in computer vision and pattern recognition.

## References

Automatic number plate recognition


1. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934. Retrieved from https://arxiv.org/abs/2004.10934

2. Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767. Retrieved from https://arxiv.org/abs/1804.02767

3. Tesseract OCR. (2023). Tesseract Open Source OCR Engine. Retrieved from https://github.com/tesseract-ocr/tesseract

4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.

5. Ultralytics. (2024). YOLOv8 Documentation. Retrieved from https://docs.ultralytics.com

6. Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. Retrieved from https://arxiv.org/abs/1412.6980

7. Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 234–241). Springer, Cham.

8. Ahmedov, A. (n.d.). *Automatic number plate recognition* [Kaggle Notebook]. Retrieved from https://www.kaggle.com/code/aslanahmedov/automatic-number-plate-recognition/input


*Training vs validation loss*

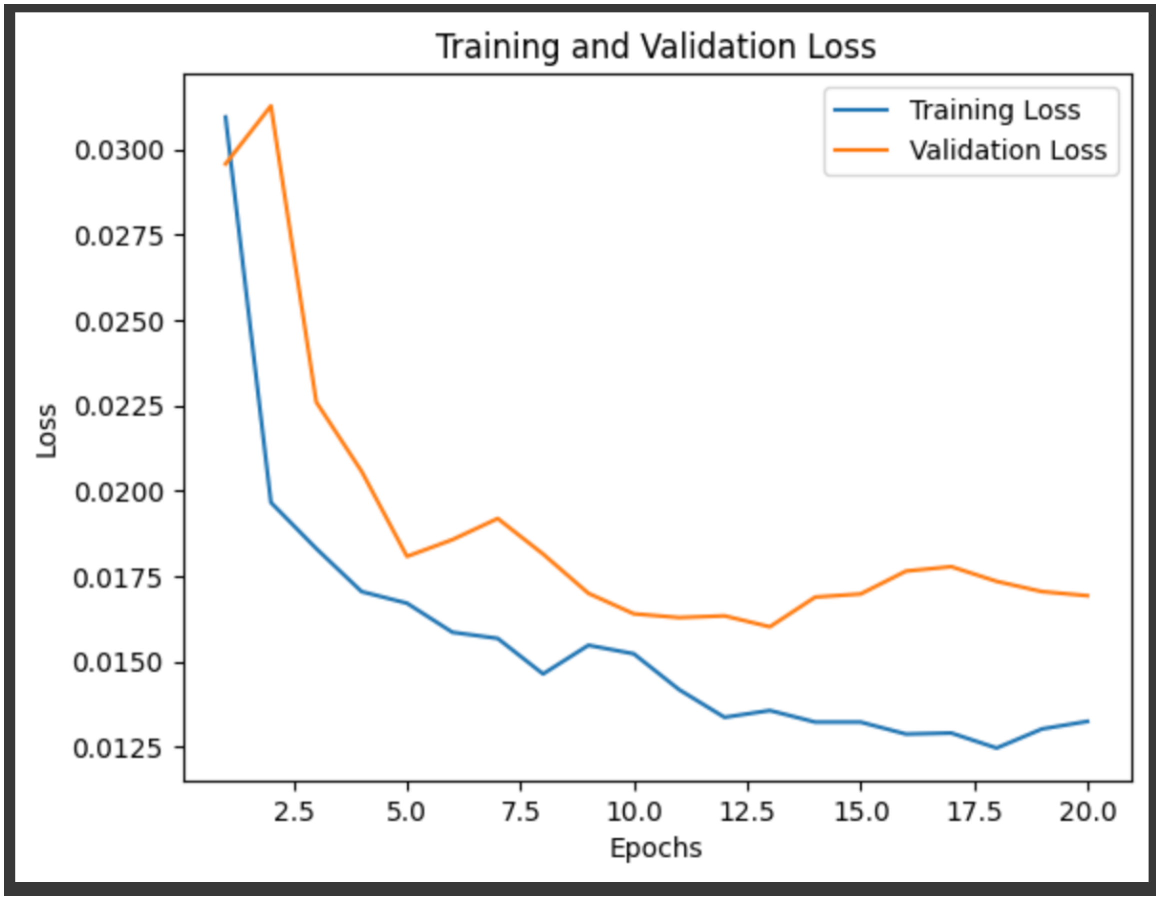



**Figure 1**

*Tensorboard monitoring*

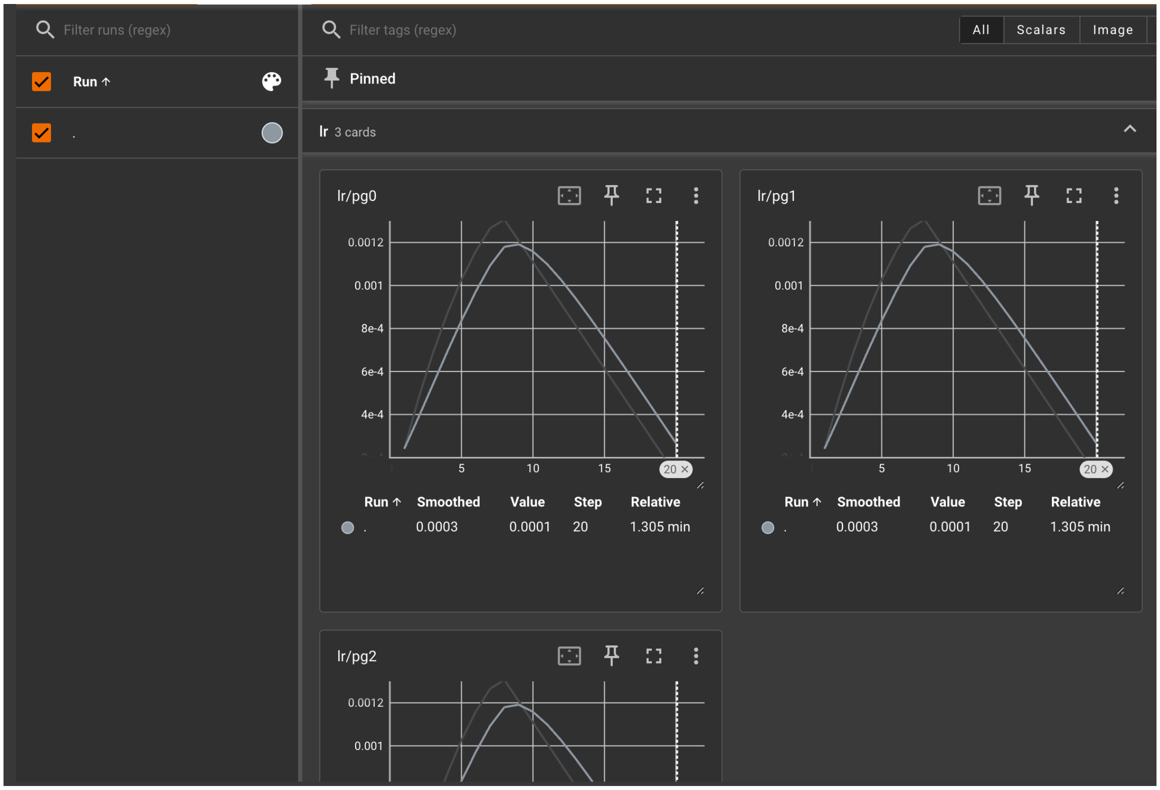



**Figure 2**

*Tensorboard monitoring*

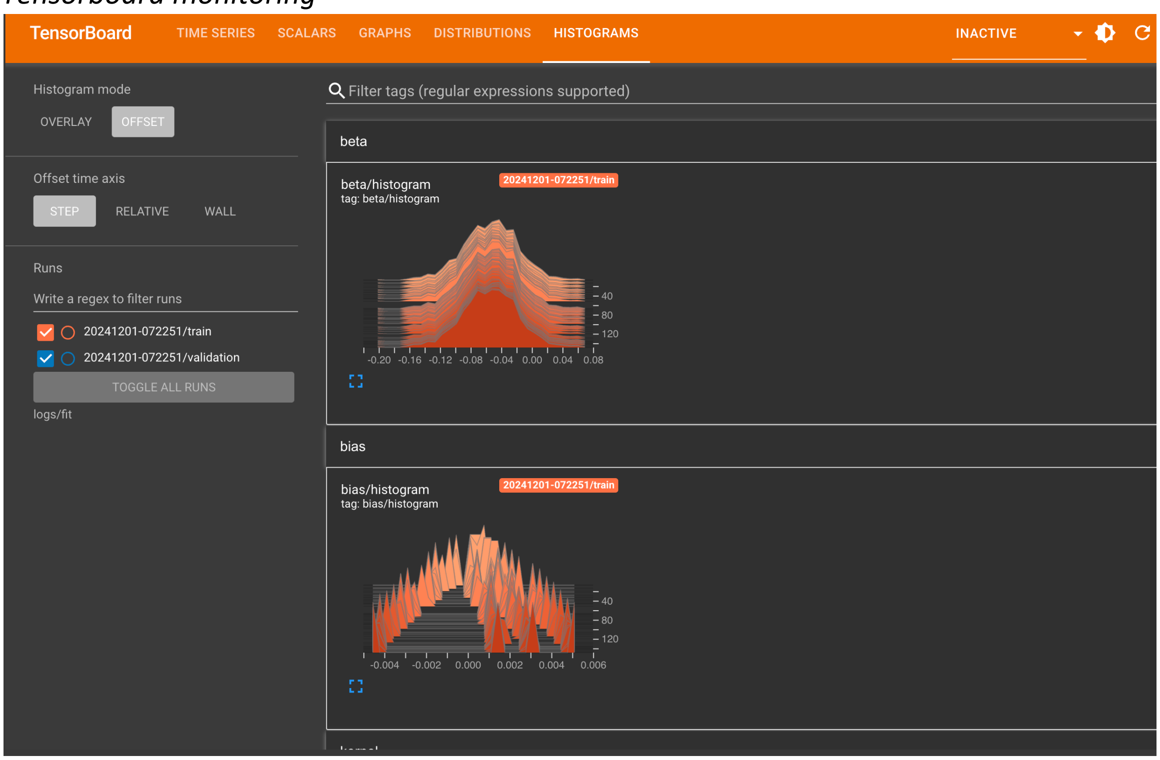
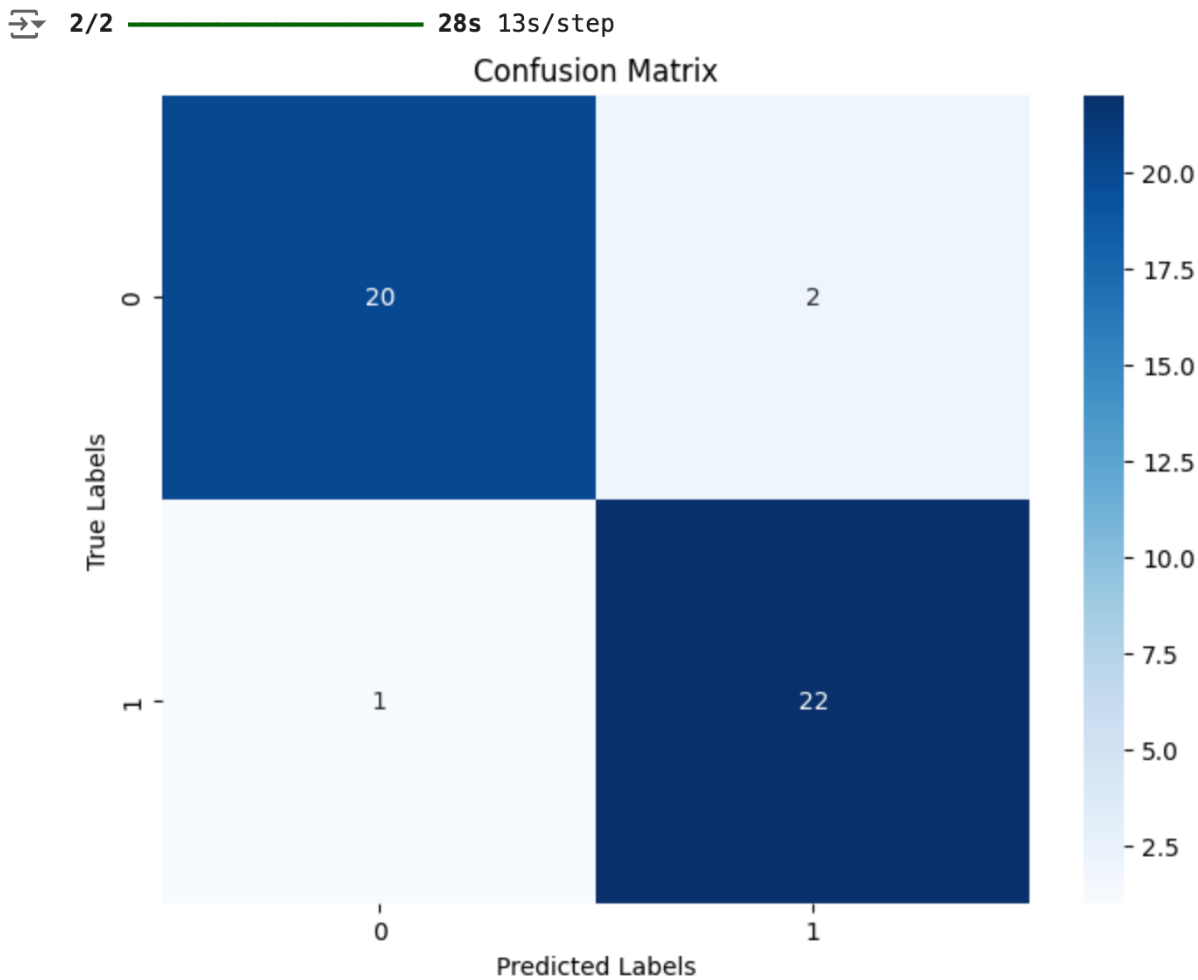


**Figure 3**

*Confusion Matrix*




*Appendix 1 – List of Project Participants and Contributions*

**Arup Chakraborty**

- Created and completed Final Project Code
- Create the Final Project Presentation
- Contributed to Final Project Paper Review

*Appendix 2 – Final Project Code*

https://github.com/achakraborty2024/AutoNumberPlateRecognition.git