**📘 Project Report**

**Title: License Plate Detection using YOLOv8**

**Presented by:**

* Hanzala Haroon
* Qamar ul Hasnain

**Department: Computer Science**

**University: University of Management and Technology (UMT)**

**Supervisor: Dr. Syed Farooq Ali**

**Abstract**

This project introduces an automated system for detecting vehicle license plates using a deep learning-based object detection model — YOLOv8. The system is designed to enhance surveillance and traffic monitoring by accurately identifying license plates in real-time footage. A custom dataset was obtained from Roboflow, consisting of annotated images of vehicle license plates. The YOLOv8n model was trained for 100 epochs, utilizing essential parameters such as learning rate and image resolution tuning. The model’s performance was evaluated using metrics such as precision, recall, F1-score, and mAP. The results demonstrate high accuracy and prove the feasibility of deploying such systems in real-world environments.

**Table of Contents**

1. Introduction
2. Literature Review
3. Dataset Description
4. Proposed Methodology
5. YOLOv8 Architecture
6. Hyperparameter Tuning
7. Evaluation Metrics
8. Results and Analysis
9. Conclusion
10. References

**1. Introduction**

License plate recognition (LPR) is a crucial part of intelligent traffic monitoring systems. It enables law enforcement and city traffic authorities to automate the tracking of vehicles, helping reduce traffic violations and improve safety. Traditional image processing methods are limited in complex environments. Therefore, this project utilizes YOLOv8, a real-time object detection algorithm, to detect license plates effectively from video footage.

**2. Literature Review**

Earlier systems relied on classical techniques like edge detection, contour detection, and morphological operations, which were sensitive to lighting and noise. With the rise of deep learning, object detection models like YOLO (You Only Look Once), SSD, and Faster R-CNN have gained popularity. YOLOv8 is the latest version from Ultralytics, optimized for real-time performance and high accuracy, making it suitable for detecting small objects like license plates.

**3. Dataset Description**

The dataset was obtained from [Roboflow](https://universe.roboflow.com), named “license-plate-detector-ogxxg”.

* Total Images: ~500
* Classes: 1 (LicensePlate)
* Dataset Split:
  + Training Set: 70%
  + Validation Set: 20%
  + Test Set: 10%
* Each image is labeled with a bounding box around the license plate.

**Figure 1**: Sample annotated image  
**Figure 2**: Dataset distribution bar chart (Train/Val/Test)

**4. Proposed Methodology**

The following steps were followed in the project:

1. Dataset collection and annotation from Roboflow
2. Installing YOLOv8 environment and dependencies
3. Model training using yolov8n.pt weights
4. Model evaluation using test video footage
5. Performance analysis via CSV logs and visual detection output

**Figure 3**: Methodology pipeline diagram

**5. YOLOv8 Architecture**

YOLOv8 consists of three main components:

* **Backbone**: Extracts features from the input image (CSPDarknet).
* **Neck**: PANet/FPN used to combine and refine features.
* **Head**: Predicts bounding boxes and class probabilities.

**Figure 4**: YOLOv8 architecture block diagram

**6. Hyperparameter Tuning**

| **Parameter** | **Value** |
| --- | --- |
| Model | YOLOv8n |
| Epochs | 100 |
| Image Size | 640×640 |
| Batch Size | 16 (default) |
| Learning Rate | 0.01 (default) |
| Confidence Threshold | 0.25 |

* Optimizer: Adam/SGD (default YOLO setting)
* Activation Function: **Leaky ReLU**

**Figure 5**: Leaky ReLU activation graph

**7. Evaluation Metrics**

The following metrics were used to assess performance:

* **Precision**: Proportion of correctly predicted plates
* **Recall**: Proportion of actual plates detected
* **F1-Score**: Harmonic mean of precision and recall
* **mAP50**: Mean Average Precision at IoU threshold of 0.5
* **Confusion Matrix**: To visualize TP, FP, FN, etc.

**8. Results and Analysis**

| **Metric** | **Value** |
| --- | --- |
| Precision | 0.9671 |
| Recall | 0.9095 |
| F1 Score | 0.9374 |
| mAP50 | 0.9756 |
| Total Epochs | 100 |

* The model successfully detected license plates in the test video with high confidence.
* **Figure 6**: Confusion matrix (visual)
* **Figure 7**: Final frame detection with bounding boxes

**9. Conclusion**

The system successfully implemented YOLOv8 for accurate and efficient license plate detection in video footage. The model showed high precision, recall, and F1-score, validating its effectiveness. With further enhancements like OCR for license plate recognition, the system can be deployed for real-time traffic management and law enforcement applications.

**10. References**

1. Roboflow Dataset: <https://universe.roboflow.com/snu-i6ovv/license-plate-detector-ogxxg>
2. Ultralytics YOLOv8 GitHub: <https://github.com/ultralytics/ultralytics>
3. Ultralytics Docs: https://docs.ultralytics.com
4. Papers with Code: YOLO Object Detection