A Major Project Proposal on

AUTOMATIC LICENSE NUMBER PLATE DETECTION

Submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Software Engineering at Pokhara University

By

AJIT BANIYA

BIBEK BHATTARAI

DIPESH GAUTAM

Department of Research and Development

GANDAKI COLLEGE OF ENGINEERING AND SCIENCE

Lamachaur, Kaski, Nepal

(December, 2023)

A Major Project Proposal on

AUTOMATIC LICENSE NUMBER PLATE DETECTION

Submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Software Engineering at Pokhara University

By

AJIT BANIYA

BIBEK BHATTARAI

DIPESH GAUTAM

Supervisor

ER. KRISHNA KHADKA

Department of Research and Development

GANDAKI COLLEGE OF ENGINEERING AND SCIENCE

Lamachaur, Kaski, Nepal

(December, 2023)

ABSTRACT

Nepal has seen an exponential increase in the number of publicly and privately owned vehicles over the past decade but, a lack in the advancement of technology to keep track of these vehicles. As per current conditions, the task of tracking the vehicle number plate is a manual process which has proven to be time-consuming and tedious with sub-par results.

Keeping this in mind, we have created a system namely “Automatic License Number Plate Recognition” that automates the process of license plate identification and recognition. The system makes use of different algorithms for plate localization and character segmentation along with machine learning techniques for character recognition. This helps eliminate the manual process of having to identify the vehicle number plate.

Keywords: Machine learning, character segmentation, localization, character recognition.

TABLE OF CONTENT

ABSTRACT.. i
CHAPTER 1 4
INTRODUCTION 4
1.1 Introduction 4
1.2 Problem Statement.. 5
1.3 Objectives 5
1.4 Implications 5
CHAPTER 2 6
LITERATURE REVIEW 6
2.1 Literature Review.. 6
CHAPTER 3 9
TOOLS AND METHODOLOGY.. 9
3.1 Required Tools 9
3.2 Methodology.. 10
3.3 Use Case Diagram 11
CHAPTER 4 12
TIMELINE CHART FOR THE SYSTEM.. 12
REFERENCES 13

LIST OF FIGURES

Figure 1: Software Development Life Cycle 9

Figure 2: Use Case Diagram 10

Figure 3: Gantt Chart 11

CHAPTER 1
INTRODUCTION

1.1 Introduction

In recent years, the advancement of computer vision technologies has paved the way for innovative applications in various fields, including transportation, security, and automation. One such application is the automatic detection and recognition of license plates from images or video streams. This technology is crucial in enhancing efficiency and accuracy in tasks such as toll collection, parking management, and law enforcement.

The primary objective of this project is to develop a robust Automatic License Plate Detection System (ALPR) using computer vision techniques. The system will employ algorithms to automatically locate and recognize license plates within images or video frames. This involves several key tasks, including image preprocessing, license plate localization, and optionally, character segmentation and recognition.

Through this project, we aim to explore fundamental concepts in image processing, pattern recognition, and machine learning. By implementing and integrating these techniques into a cohesive system, we endeavor to achieve accurate and real-time detection of license plates under various environmental conditions and scenarios.

The outcomes of this project will not only demonstrate practical skills in computer vision and software development but also contribute to the broader discourse on the applications and implications of automated license plate detection systems in modern society.

1.2 Problem Statement

The traffic control and management system have undergone significant development in the past decade; however, it still exhibits shortcomings, particularly in leveraging modern technology. The current approach employed by the Nepal Traffic Police for tracking lost and stolen vehicles relies on a manual process, necessitating substantial human resources. Unfortunately, the outcomes have not met expectations relative to the resources invested, given the arduous nature of the vehicle tracking process. Our examination reveals that the Nepal Traffic Office has integrated approximately 3000 CCTV cameras across the country, complemented by high-resolution cameras strategically placed in traffic-congested areas to monitor daily traffic patterns. While this system effectively regulates unauthorized lane crossing, there is a notable absence of an automated license plate recognition system in the current landscape.

1.3 Objectives

- To identify the vehicle number plate.

1.4 Implications

The project has wide range of implications which are as:

- Utilizing the system for the automated detection and recognition of license plates to monitor vehicles.
- It has the potential to optimize the utilization of traffic resources effectively.

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Review

The major challenge of the traditional learning based and or tracking-by-detection methods
is the false positive matches that lead to wrong association of the tracks. The reason is that
those methods are based on applying an appearance model or object detector at all possible
windows around the target, and the object detection in the current frame as our base learner
because they have been demonstrated to be able to extract local visual features (structures)
and they are widely used in various visual recognition applications. Different from fully connected neural networks, CNNs force the extraction of local features by restricting the
receptive fields of hidden units to be local, since images have strong 2-D local structures.

From the conventional shift-invariant CNN we move to improved shift variant architecture.

Shift-invariant CNN can make them suitable for recognition or detection tasks but inappropriate for tracking tasks. [1]

Automatic Number Plate Recognition is comprised of four main sections: image preprocessing, number (license) plate extraction (localization), character segmentation and
character recognition. ANPR can be used to store the images captured by the cameras as
well as the text from the license plate, with some configurable memory. ANPR technology
tends to be region-specific, owing to plate variation from place to place. The ANPR system
that extracts a license plate number from a given image can be composed of four stages.

The first stage is to acquire the car image using a camera. The parameters of the camera, such as the type of camera, camera resolution, shutter speed, orientation, and light, have to

be considered. The second stage is to extract the license plate from the image based on some features, such as the boundary, the color, or the existence of the characters. The third

stage is to segment the license plate and extract the characters by projecting their color information, labeling them, or matching their positions with templates. The final stage is to

recognize the extracted characters by template matching or using classifiers, such as neural

networks. [2]

The number plates used in Nepal are usually of two formats, one containing all the characters in a single row and the other containing two rows of characters. Characters are

selected from Devnagari script. Here, we propose a complete number plate recognition pipeline that automatically localizes, normalizes and segments number plates from vehicle

images; segments characters from detected number plates and passes them to classification

system for labeling. Classification system implements SVM based machine learning algorithms for learning and prediction. Plate localization and segmentation are again not

researched much for handling all the situations. Nepali number plate characters are selected

from the pool of 29 characters in a specific order. The order defines various characteristics

of the number plates such as vehicle type and vehicle load. [3]

The earlier stages require higher accuracy since a failure would probably lead to another failure in the subsequent stages. YOLO is a realtime object detector that achieved impressive results in terms of speed/accuracy trade-off in the Pascal VOC and Microsoft

COCO detection tasks. We locate the vehicles in the input image and then their LPs within the vehicle bounding box. Considering that the bottleneck of ALPR systems is the LP recognition stage. Afterward, all LP characters are recognized simultaneously, i.e., the entire LP patch is fed into the network, avoiding the challenging character segmentation task. We eliminate various constraints commonly found in ALPR systems by training a

single network for each task using images from several datasets, which were collected under different conditions and reproduce distinct real-world applications. Moreover, we perform several data augmentation tricks and modified the chosen networks (e.g., we explored various models with changes in the input size, as well as in the number of layers,

filters, output classes, and anchors) aiming to achieve the best speed/accuracy trade-off at

each stage. [4]

CHAPTER 3

TOOLS AND METHODOLOGY

3.1 Required Tools

To develop our system, the following tools will be used accordingly:

Python:

Python, a versatile programming language, serves as the primary coding language for implementing the Vehicle Tracking System, offering a rich ecosystem of libraries and frameworks.

Django:

Django, a high-level Python web framework, facilitates the development of a robust backend for the VTS, providing an organized structure for handling data and user interactions.

NumPy:

NumPy, a fundamental library for numerical computing in Python, plays a crucial role in handling mathematical operations and array manipulations within the VTS codebase.

TensorFlow:

TensorFlow, a powerful open-source machine learning framework, is utilized for constructing and training deep neural networks, enabling the integration of advanced deep learning models into the VTS.

OpenCV:

OpenCV, a versatile computer vision library, is employed to process image and video data from cameras or sensors, facilitating tasks such as vehicle detection and tracking in the VTS.

YoloV8:

YOLOv8, also known as YOLOv4-tiny, is a compact yet powerful variant of the YOLO (You Only Look Once) object detection model. It is designed to deliver fast and efficient real-time object detection capabilities. Built upon the YOLOv4 architecture, YOLOv8 prioritizes speed and suitability for deployment on resource-constrained devices, making it ideal for applications requiring rapid detection and classification of objects in images or video streams.

3.2 Methodology

The project has followed the Software Development Life Cycle (SDLC), a structured process defining various stages in software development to ensure the delivery of a high-quality product. Utilizing the Incremental Model, we integrated the SDLC process.

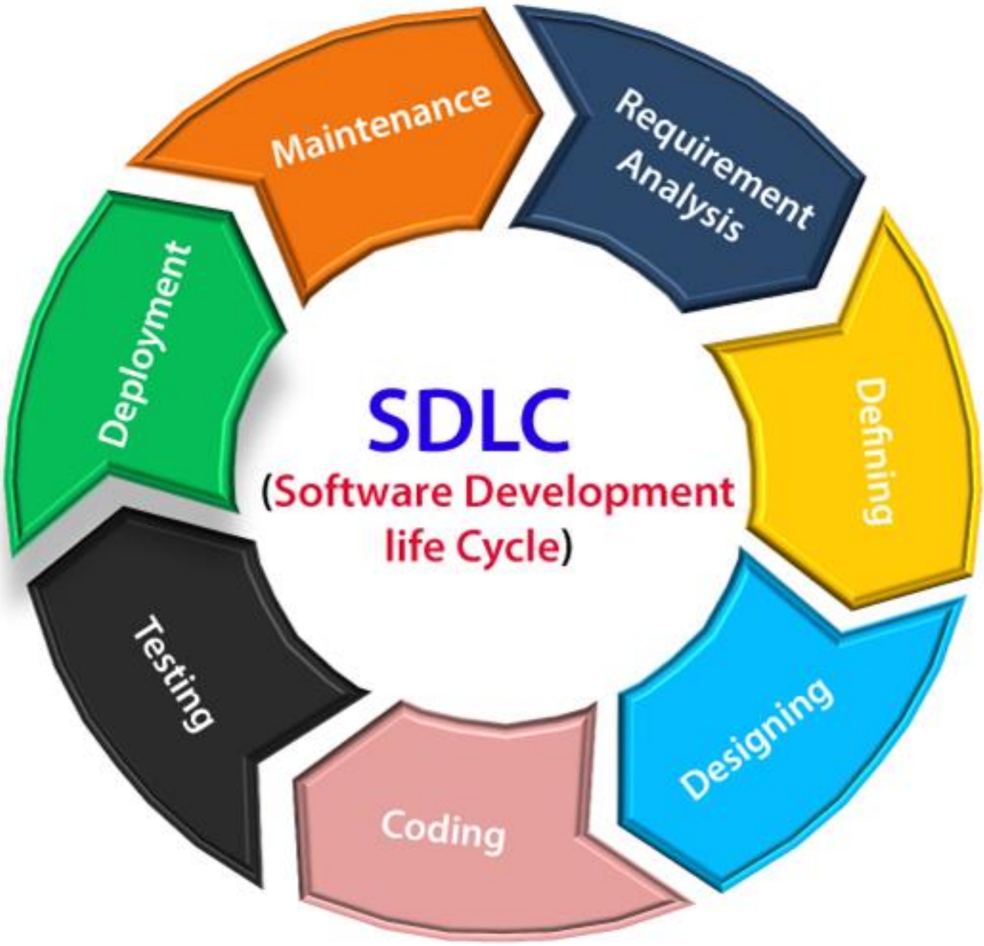

Figure 1: Software Development Life Cycle

3.3 Use Case Diagram

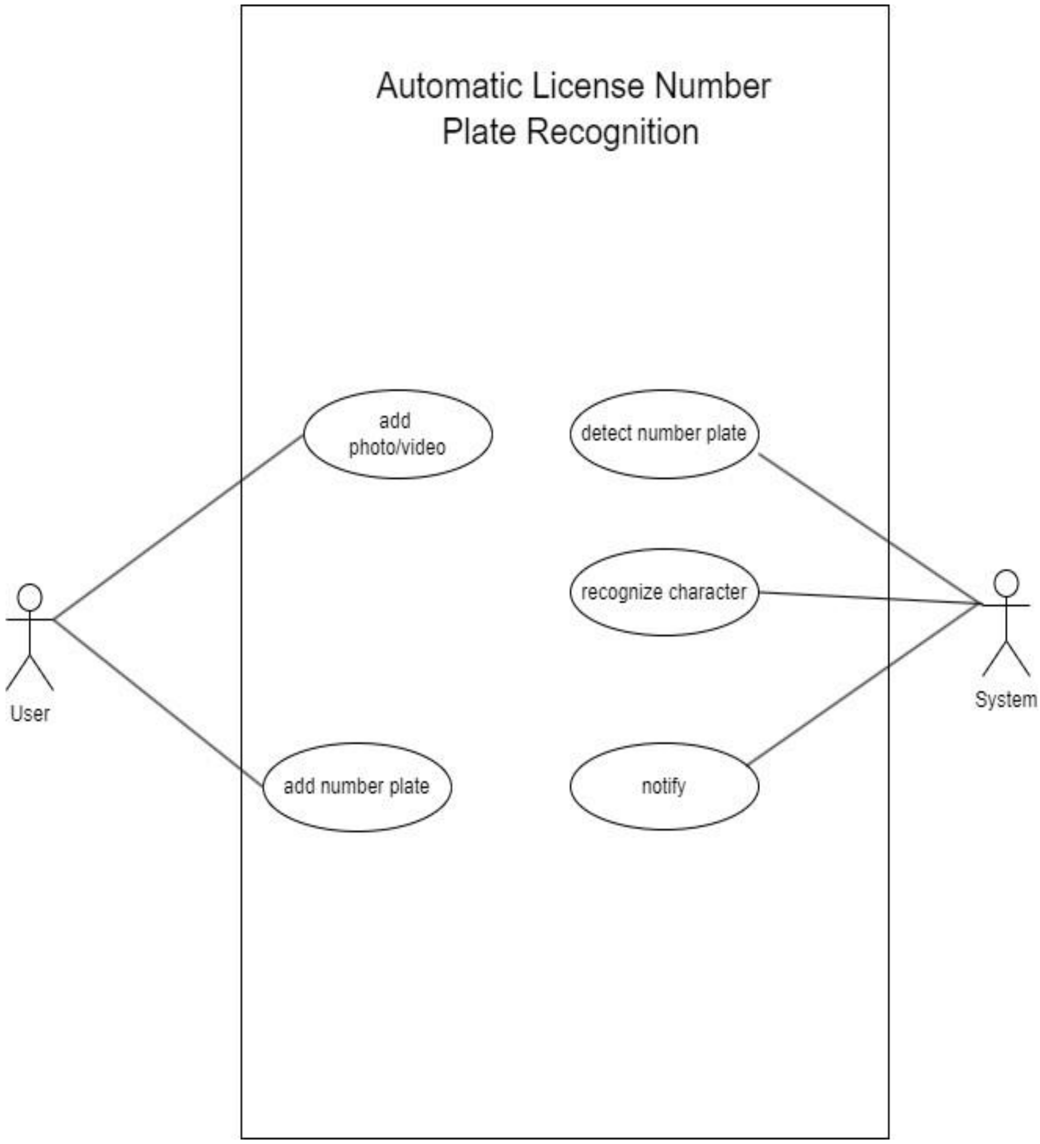

Figure 2: Use Case Diagram

CHAPTER 4
TIMELINE CHART FOR THE SYSTEM

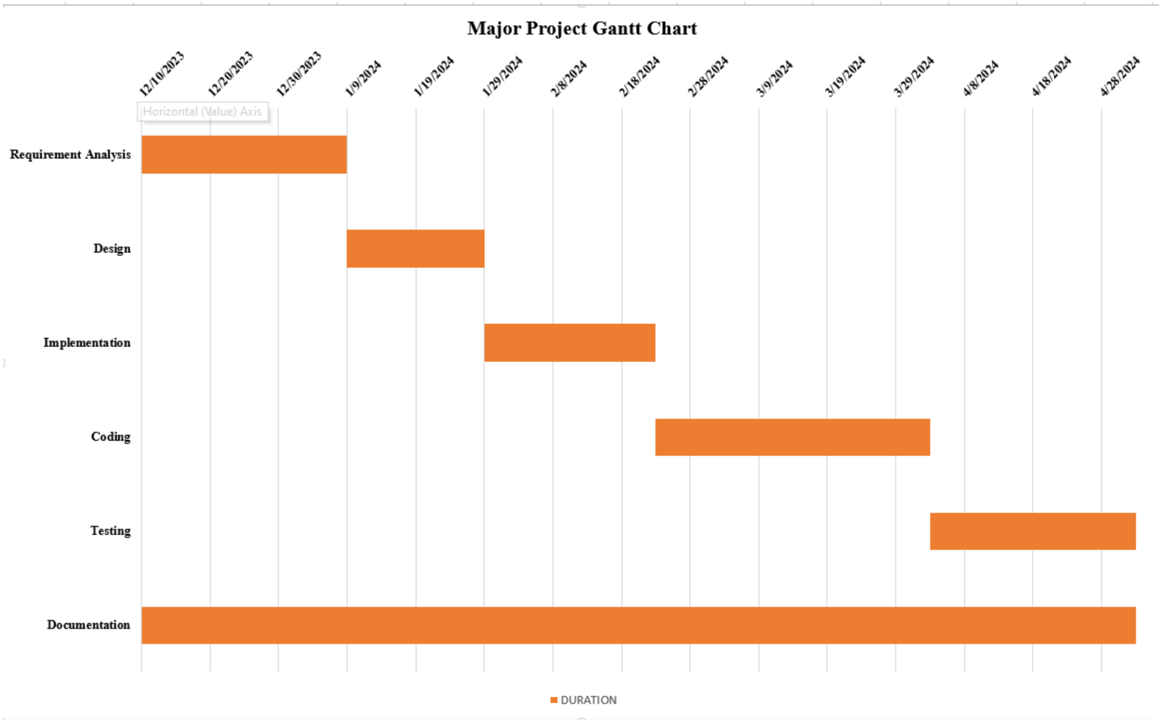

Figure 3: Gantt Chart

REFERENCES

S. Shruthi, "Vehicle Tracking using Convolutional Neural Network," World Congress on Engineering, London, 2011.

A. Jain and J. Kundargi, "Automatic Number Plate Recognition Using Artificial Neural Network," International Research Journal of Engineering and Technology (IRJET), vol. 2, 2015.

A. Pant, P. Gyawali, and S. Acharya, "Automatic Number Plate Recognition with Support Vector Machines," Kathmandu.

L. Rayson, L. Zanlorensi, G. Goncalves, E. Todt, W. Schwartz and D. Menotti, "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO Detector," Department of Informatics, Federal University of Paran´a, Curitiba, Brazil.

DrawIO: https://draw.io