# VISIPLATE

A PROJECT REPORT

BY

TEAM NO. 3

VRIDDHI JAIN (E23CSEU2042)
AADYA SINHA (E23CSEU2067)

SUBMITTED TO

SCHOOL OF COMPUTER SCIENCE ENGINEERING AND TECHNOLOGY, BENNETT UNIVERSITY

GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

April 2025

# DECLARATION

I/We hereby declare that the work which is being presented in the report entitled "VISIPLATE", is an authentic record of my/our own work carried out during the period from JAN, 2023 to April, 2023 at School of Computer Science and Engineering and Technology, Bennett University Greater Noida.

The matters and the results presented in this report has not been submitted by me/us for the award of any other degree elsewhere.

Signature of Candidate

Vriddhi jain

E23CSEU2042

Aadya sinha

E23CSEU2067

# ACKNOWLEDGEMENT

I/We would like to take this opportunity to express my/our deepest gratitude to my/our mentor, **Dr. Sanchali Das (Assistant Professor)** for guiding, supporting, and helping me/us in every possible way. I/we was/were extremely fortunate to have him as my/our mentor as he provided insightful solutions to problems faced by me/us thus contributing immensely towards the completion of this capstone project. I/We would also like to express my/our deepest gratitude to VC, DEAN, HOD, faculty members and friends who helped me/us in successful completion of this capstone project. Any other name you can mentioned here.

Signature of Candidate

Vriddhi Jain

Vriddhi jain
E23CSEU2042

Aadya Sinha
E23CSEU2067

**TABLE OF CONTENTS**

<<Right click on the heading and click update flied. Your heading will be pulled here. Same goes with all the table of content in the subsequent pages>>

## LIST OF TABLES

## LIST OF FIGURES

## LIST OF ABBREVIATIONS

| Abbreviation .................................................. | Explanation of the Abbreviation |
|---|---|
| CNN | Convolutional Neural Network |
| AI | Artificial Intelligence |
| OpenCV | open-source computer vision library |

## ABSTRACT

The increasing number of vehicles we see nowadays demands a better monitoring system which also involves innovation. this project presents a smart way of monitoring the traffic effectively during foggy weather which can also be integrated with the camera. Fog is major issue in northern and northeastern part of India as well most of the parts during winter season which causes accidents and recognizing number plates becomes tedious and hence the accidents can be left unsolved. so, our project here will help identify them and help law enforcements. this was just one situation. there are ton of issues which requires to recognize number plates, and our project was made by keeping them in mind.

Our project is combination of artificial intelligence (AI), and image dehazing and detection techniques for better recognition of number plates during low visibility environments. It can be integrated with the local cctvs. The AI-powered module is trained using convolution neural networks (CNNs). Advanced dehazing algorithms are used to improve the number plate visibility.

The system leverages OpenCV (cv2) for handling real-time image processing tasks. To further refine image clarity, the Pillow library is used for advanced image manipulations including resizing, filtering, and sharpening operations. Once images are pre-processed, a multi-layer Convolutional Neural Network (CNN) architecture is employed to detect and segment the number plate from the vehicle, even when visual obstructions due to fog are present and to ensure accuracy. We have used TensorFlow and PyTorch for deep learning model which allows us to experiment with both frameworks for training and evaluation and also ensures maximum flexibility and performance optimization.

A significant amount of data was collected, including both foggy and clear conditions, to train and test models. Through multiple training and optimization, the system achieved notable accuracy and reliability, in any foggy scenarios. The processed data is not stored in any database to avoid any legal issues, and the result will directly be shown to the authorized security personnel. The image can be captured both real time and static.

This solution has the potential to be integrated into real life use and will enhance the infrastructure and will solve public safety issues. This project also showcases the potential of AI by putting it to good cause and will help revolutionize monitoring systems in difficult environmental conditions.

# 1. INTRODUCTION

brief overview of current trends and the situation around our project-

- The use of transportation system has seen rapid growth in the recent years, increasingly with the AI advancement. however, law enforcement faces problem during adverse weather conditions due to low visibility.
- With the rise in climate unpredictability and increasing fog incidents in northern and hilly parts of India, there is a growing need for traffic monitoring systems. Recent studies have shown that AI-enhanced dehazing techniques and some real time image processing models can be used to make number plate detection accurate and less challenging.
- This opens an amazing opportunity to setup it into toll plazas, checkpoints, parking lots, and highways. It is a need in modern transportation systems.

## 1.1. Problem Statement

NTD and Technology In general the majority of the ANPR systems available on the market today are set-up for standard photographic conditions due to their initial design for traffic control and enforcement agencies. But in low visibility, particularly in areas with frequent fog, these systems don't perform very well. The ability to exhaustively accurately identify and recognize vehicle number plates in the stated environmental condition is hindering critical operations including those involved in traffic control, toll collection, vehicle surveillance, and criminal investigation. Current systems do not incorporate modern image enhancement algorithms, making it impossible to ensure continual monitoring or guarantee the public's safety while driving in foggy weather. Due to upcoming trends large number of road accidents have increased especially in hilly regions which we decided to address.

# 2. BACKGROUND RESEARCH

Literature review –
We found out that there were many research papers on topic of our project that is dehazing number plates and clearing fog from it but there were none who acted upon it .
**And below are the research paper contents attached.**
Over the last decade, many efforts have been made to solve the problem of detecting potential LP area from an image or a video. Different state-of-the-art methods apply different image processing approaches, techniques and algorithms to build their ALPD methods. To detect an LP area from an image, different features such as geometric feature, texture feature and colour feature of the LP are utilized individually or jointly [3]. Hazardous conditions have effect on these features of an LP.
Proposed method

The proposed ALPD method comprises of five stages:
1.Grayscale conversion, noise and rain effects removal.
2.Contrast enhancement and binarization.
3.Local counting filter and cropping connected components (CCC).
4.Tilt angle detection and correction.
5.Filtering non-LP regions.

INTRODUCTION-
With the rapid development of economy and the continuous enlargement of cities, the transportation system has become an indispensable component of modern society. To manage the numerous vehicles conveniently, the Intelligent Traffic System (ITS) [1] is proposed to provide innovative services relating to the transport and the traffic management. Each vehicle around the world has a license number as its principal identifier, as an essential stage in ITS; the technique of vehicle License Plate Recognition (LPR) [2] can recognise and distinguish the vehicles automatically. LPR is widely used by police forces for law enforcement purposes, and it also plays an important role in the applications of vehicle tracking, traffic congestion monitoring, management of city streets and electronic toll collection on pay-per-use roads. As shown in Figure 1, the license plate is blurred by the fogs or hazes, and the characters of this license plate are hard to be identified

PROPOSED SYSTEM -
An image-dehazing technique based on an CNN which takes into account the fog concentrations is proposed. The images are divided into several regions according to the fog concentrations, and a finer transmission map can be generated by the values of local atmospheric light rather than those of global atmospheric light, and then the fog-haze interference can be filtered from the original images as much as possible. A Joint Further-dehazing and Region extracting Model is provided. JFRM can dramatically reduce the distortion of image restoration caused by the cumulative errors and accurately detect the license plate regions of the preliminarily-dehazed images. In the proposed convolution- enhanced RCNN,the image quality can be significantly improved. OCR technique will take an image as input data then produces output in the form of text.

PREVIOUS WORK-

[1]

In this paper LPDR plays a significant role to recognize vehicles registration number at a certain distance. Over past few years, researchers have developed many techniques to detect vehicle license plate but still it remains a challenging task. Vehicle identification approach can be classified into four main steps such as pre-processing, License plate region extraction, characters segmentation and each character recognition in the licensed number plate. In this paper, first various vehicle images have been acquired through camera, then input colour image is converted to Gray scale image, brightness adjustment, contrast up to optimum values and removing noise using median filtering is done in order to get better quality image. Finding exact location of license plate is the most important processing step in vehicle detection system, because all others steps depend on exact extraction of license plate region. Exact location of license plate region is masked and extracted from image. Segmentation of each character is done for extracted region. Segmentation is a process of subdividing a digital image into the consequent parts. The main purpose of image subdividing into consequent parts or objects present in the image is that we sought for the analysis extract some meaningful information. Segmentation is the crucial step in recognizing the vehicle license plate. After segmentation, character
recognition is done. Each segmented character is compared with template matching, if characters are matched then it will display the output in text. First we select the image, remove noise and find the interested area of image, then the license plate location is extracted using edge detection then segmentation of each characters individually.

CONCLUSION-

This study explores the vehicle license plate recognition problem in the fog-haze environments. To reduce the fog-haze interference, a fog concentration dehazing method is first investigated, and in the proposed DCP method, the further-dehazing detection is optimised jointly, which dramatically reduces the distortion of image reconstruction caused by the cumulative errors and accurately extracts the license plates from the preliminarily dehazed images. Finally, the image super-resolution is accomplished with a convolution-enhanced RCNN, and hence the characters of license plates can be recognised successfully. Simulation results demonstrate our method can improve the accuracy of the vehicle license plate recognition in the fog- haze environments. In the near Future we will be able to detect the number plates in various climatic conditions. Future research focuses on the optimisation of transmission maps in the dehazed images. Extensive experiments have been conducted, and the results indicate that DCP can recognise the license plates accurately even in some severe fog-haze environments.

Inspiration for the Project -

A thorough analysis of existing technologies and their drawbacks revealed a sizable technological gap in number plate recognition systems designed for foggy conditions. Both a technical challenge and an opportunity for innovation are presented by the dearth of dependable, affordable, and AI-powered solutions. This project is driven by the practical need to improve road safety and surveillance in India, particularly in high-altitude areas where fog is a frequent hazard and during the winter. This project intends to improve the resilience, effectiveness, and safety of transportation infrastructure by creating a system that can precisely detect and recognize license plates in foggy conditions.

**References** :


- Hassan, R., Mahmud, M., & Hasan, M. (2019). Challenges in Vehicle License Plate Recognition: A Survey. *International Journal of Advanced Computer Science and Applications.*
- Zhang, Y., Wang, Y., & Liu, C. (2020). Deep Learning-Based Approach for Vehicle Plate Detection under Adverse Weather. *IEEE Transactions on Intelligent Transportation Systems*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *CVPR*.
- He, K., Sun, J., & Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Anwar, S., & Barnes, N. (2021). Dense Haze Removal with CNNs. *IEEE Access*.
- Patel, D., Joshi, K., & Shah, M. (2021). Vehicle Number Plate Recognition using Tesseract OCR with Preprocessing Techniques. *IJCSMC*.
- Ministry of Road Transport and Highways (2022). *Annual Report on Traffic Infrastructure and Smart Surveillance*.


## 2.1. Proposed System

- This project aims to develop an intelligent license plate recognition system that performs well in hazy conditions. The primary objective is to address the drawbacks of conventional ANPR systems, which struggle to generate accurate results in low-visibility environments. This system combines real-time optical character recognition, advanced image enhancement techniques, and deep learning-based object detection to provide reliable vehicle identification even in the face of challenging weather.
- The goal of the project is to improve the security, effectiveness, and responsiveness of traffic management and law enforcement entities. In places such as the northern and mountainous parts of India that are frequently affected by fog, this system will greatly improve road surveillance, toll operations, and vehicle tracking. Authorities are therefore better able to maintain control during fog seasons, which reduces accidents, increases monitoring, and guarantees continuous operations.
- This solution is in line with India's growing demand for weather-resilient infrastructure and the use of smart traffic systems. Once put into place, the system might be expanded to include highway monitoring systems and smart city initiatives, which would enhance the general customer experience for both transportation departments and commuters.

## 2.2. Goals and Objectives

**Table 1. Table of Goals and Objectives –**

| s.no. | Goals and objectives |
|---|---|
| **1.** | Improve number plate detection accuracy during foggy conditions by at least 85%. |
| **2.** | Integrate real time image enhancement. |
| **3.** | Use cv2 for image detection and regeneration. |
| **4.** | Use CNN and TensorFlow for defogging. |
| **5.** | Modification – while using the dehazed model, we have increased the number of CNN filters. |
| **6.** | Conduct at least 3 field tests in different visibility scenarios. |
| **7.** | Document all design and development phases clearly for future maintenance. |
| **8.** | Ensure compatibility with existing surveillance camera systems. |
| **9.** | Foster strong creativity and collaboration withing the project team. |
| **10.** | Design a system that can be upgraded for rain, snow, and night conditions. |

# 3. PROJECT PLANNING

Project Planning-

This project follows an Iterative Development Lifecycle, allowing for regular testing, feedback integration, and refinement of the system. The project is broken into key phases: requirement gathering, model training, system integration, testing, deployment, and user feedback analysis. Each phase allows for continuous improvement and ensures alignment with user needs and environmental constraints like fog conditions.

Project Stakeholders-

- Project Team: Computer science students (developers, testers, researchers)
- Academic Mentor: Guides on technical implementation and documentation
- Transport Authorities: Potential users and field testers of the system
- IT Consultants: Provide technical feasibility inputs End Users: Includes travelers, traffic police, toll operators

Resources Required-

- Hardware: HD camera for capturing vehicle images, GPU-enabled system/laptop
- Software: Python, CV2, pillow, pandas, pytorch, TensorFlow, CNN.
- Cloud Services: (Optional) For model training or dataset storage
- Dataset: Public vehicle license plate datasets and weather-specific imagery
- Testing Environment: Outdoor area with artificial fog (if needed) or fog simulation tools

Assumptions Made -

- Real-time implementation will be supported by sufficient processing power.
- Camera feed quality will be consistent and within the required resolution.
- Fog levels can be reasonably simulated for training and testing.
- End users (traffic department or monitoring stations) are familiar with basic tech operations.

## 3.1. Project Lifecycle

The project will adhere to a customized Agile-SCRUM lifecycle to ensure mobility, iterative development, and continuous feedback integration. In advance of developing a high-level plan involving technical components like real-time processing, deep learning integration, and image enhancement, the team will gather requirements. The project will be finished in four main sprints, each of that will concentrate on essential elements such as data collection, model training, system integration with camera hardware, and performance review. Each sprint will last roughly two to three weeks and include planning, execution, testing, and review.

The group will use a methodology like SCRUM, focusing on:

- Daily Standups: Short discussions to track progress and blockers.

- Sprint Reviews: Held at the end of each sprint to demonstrate the current build and receive mentor feedback.
- Retrospectives: To assess what went well and what could be improved in the next sprint.
- Backlog Grooming: Regular updates to tasks and features based on test results and feedback.

This method ensures a methodical and goal-oriented development process while enabling flexibility to address real-world issues like fluctuating fog levels, hardware constraints, and dataset refinement.

## 3.2. Project Setup

**Table 2. basic project decisions that will be used on this project are –**

| S.no. | Decision description |
|---|---|
| 1. | Python as primary language with CV2 for image detection and regeneration. |
| 2. | TensorFlow, CNN for defogging for the number plate. |
| 3. | The system will be tested locally on a GPU-enabled machine, simulating foggy conditions using datasets and artificial filters. |
| 4. | no real-time personal license plate data will be collected without consent. |
| 5. | No special access or NDAs required at this stage; however, data privacy will be considered if real footage is used. |
| 6. | Version control will be maintained using GitHub, and collaboration will happen via Google Docs and WhatsApp. |
| 7. | UI designs and flow will adhere to User-Centered Design principles to ensure minimal training for end-users. |

## 3.3. Stakeholders

**Table 3. stakeholder table-**

| Stakeholder | Role |
|---|---|
| **Sanchali Das** | Project mentor/guide |
| **Vriddhi jain** | Developer/model trainer/project lead/integration lead |
| **Aadya Sinha** | UI designer/Data collector/model trainer |
| **Traffic department** | Potential end user |
| **General public** | Indirect users |
| **Dataset providers** | Data source stakeholders |
| **university** | Project sponsor/evaluator |

## 3.4. Project Resources

**Table 4. resource table-**

| **resource** | **Resource description** | **quantity** |
|---|---|---|
| Development team | Core student team responsible for designing, coding, testing, and integration | 2 |
| Project Mentor | Faculty mentor providing technical advice and review | 1 |
| GPU-Enabled Laptop | Required for training and running machine learning models efficiently | 1 |
| Python Software Stack | Python with CV2, pytourch, pillow and other libraries | 2 |
| Internet Connection | Required for dataset download, cloud tools, and collaboration | 1 |

| | | |
|---|---|---|
| Collaboration Tools | Google Docs, WhatsApp, GitHub for team communication and version control | N/A |
| Image Enhancement Toolkit | Tools for dehazing images. | 1 |
| Dataset | Public datasets for number plate recognition, foggy image simulation | multiple |
| Fog Simulation Software | Tool or custom script to simulate foggy conditions on clear vehicle images | 1 |
| Cloud Storage | To store trained models, and testing results for remote access and backup | 1 |

## 3.5. Assumptions

**Table 5. Assumption table-**

| **S.no.** | **assumption** |
|---|---|
| **1.** | The project team and mentor will be able to meet at least once per week for regular progress updates. |
| **2.** | All required libraries and tools will be freely available and compatible. |
| **3.** | Team members will gain sufficient familiarity with machine learning and image processing within the timeline. |
| **4.** | Publicly available datasets will be sufficient for training and testing the model with acceptable accuracy. |
| **5.** | GPU-enabled systems will be available for model training during the critical implementation phase. |
| **6.** | No major hardware or software failures will occur that delay progress significantly. |

| | |
|---|---|
| **7.** | The system will be integrated and tested under simulated real-time conditions without needing government data. |
| **8.** | The fog simulation techniques will realistically mimic real-world conditions for effective testing. |

# 4. PROJECT TRACKING

## 4.1. Tracking

**Table 6. Project Tracking and Information Management table –**

| **information** | **description** | **links** |
|---|---|---|
| **Code storage** | All project code is stored in a private GitHub repository. | https://github.com/vriddhij/VisiPlate-number-plate-detection-in-foggy-weather |
| **Bug tracking** | Stored and maint | N/A |

| | | |
|---|---|---|
| | ained manually. | |
| **Project Documents and Assignments** | Project documents, reports, diagrams, and progress updates are maintained in Google Drive | https://drive.google.com/drive/folders/1BOSez5ryyBGZBkgP5ZPImniq6zmlfQv7?usp=share_link https://docs.google.com/document/d/1go4CCaZBFiHpdJP1FN9sMe81D9mGoVYe/edit?usp=sharing&ouid=113502162671526274912&rtpof=true&sd=true |

| | | |
|---|---|---|
| | folder s privat ely. | |
| | | |

## 4.2. Communication Plan

**Table 7. Regularly Scheduled Meetings –**

| Meeting type | frequency | Who attends |
|---|---|---|
| **Physical meeting** | Weekly | Project team and mentor |
| **Team meeting** | weekly | Project team |
| **Class status meeting** | Weekly un class | Team and mentor |

Table 8. Information To Be Shared Within Our Group –

| Who? | What information? | When? | How? |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **Project team** | Task assignments, progress, issues | weekly | Physical meeting |
| **Project leader** | Status updates, feedback from mentor | weekly | Share documents and update |

**Table 9. Information To Be Provided to Other Groups-**

| **Who?** | **What information?** | **When?** | **How?** |
|---|---|---|---|
| **Mentor and guide** | Final project | At completion of project | Github link, report and final presentation |
| **Mentor** | Weekly progress, updates | weekly | docs |

**Table 10. Information Needed from Other Groups-**

| **Who?** | **What information?** | **When?** | **How?** |
|---|---|---|---|
| **mentor** | Technical feedback, changes | weekly | Physical meeting |
| **Data providers** | Access to the number plate images | Before model training | Email or physically capturing the data |

## 4.3. Deliverables

**Table 11. Deliverables Table –**

| **s.no.** | **deliverable** |
|---|---|
| **1.** | Research and feasibility study document, including fog condition analysis |
| **2.** | Source code for fog removal, image processing, and number plate recognition |
| **3.** | Testing plan and results, including real-time foggy condition scenarios |

| | |
|---|---|
| **4.** | Build and deployment instructions for integration with the embedded system |
| **5.** | Installation guide for setting up software with the camera module |
| **6.** | User manual for operators using the vehicle detection system |
| **7.** | Postmortem document analysing project challenges, lessons learned, and outcomes |
| **8.** | Final submission: project report, PowerPoint presentation, and a demo video |

# 5. SYSTEM ANALYSIS AND DESIGN

## 5.1. Overall Description

The goal of this project is to create a reliable system for identifying license plates that can operate well in inclement weather, particularly heavy fog. To identify and read license plates in low-visibility situations, the system combines computer vision algorithms, machine learning models, and image enhancement techniques. Real-time vehicle images are captured by a high-resolution camera module, and pre-processing is carried out using dehazing and defogging algorithms based on deep learning-based image restoration models, contrast enhancement, and histogram equalization. These methods enhance the clarity of photos taken in fog and lessen atmospheric interference.

Following image enhancement, the number plate is located using object detection methods, such as CV2-based detection.  utilizing CNN and TensorFlow for defogging for on-field testing, this system is implemented using an edge-based microcontroller that is integrated with the camera. This project improves security and vehicle tracking capabilities by addressing real-world issues that law enforcement, toll booths, and smart city surveillance systems face during foggy seasons. The solution is scalable, adaptable, and future-upgradable for a variety of weather conditions or languages on plates thanks to its modular design and machine learning integration.

## 5.2. Users and Roles

**Table 12. description of the different types of users or roles -**

| **User** | **Description** |
|---|---|
| **System administrator** | Responsible for configuring, managing, and maintaining the hardware and software setup, ensuring data storage security, and overseeing overall system performance. |
| **Traffic officer** | Primary end user of the system. Uses the number plate identification tool to monitor vehicle activity, verify and generate reports in foggy conditions. |
| **Developer** | technical team members involved in designing, coding, training machine learning models, and integrating the image enhancement modules. |

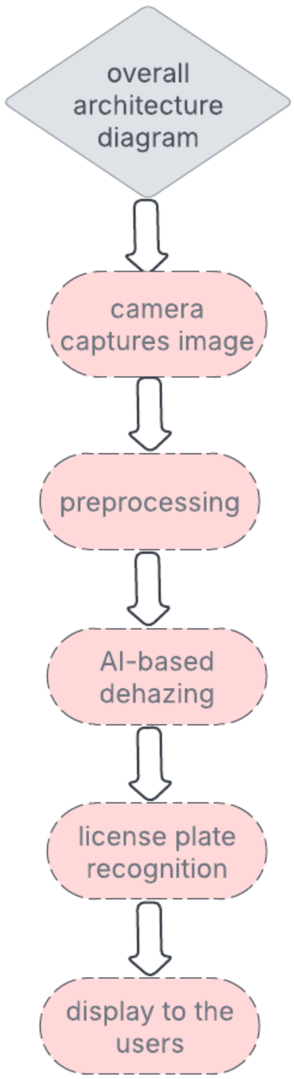
| | |
|---|---|
| **Law Enforcement** | Utilizes the processed number plate data for tracking stolen vehicles, enforcing traffic rules, and managing public safety. |
| **Surveillance Camera Unit** | An automated system or IoT component that captures and transmits images to the processing module. It acts as the primary data source in real-time. |

## 5.3. Design diagrams/Architecture/ UML diagrams/ Flow Charts/ E-R diagrams

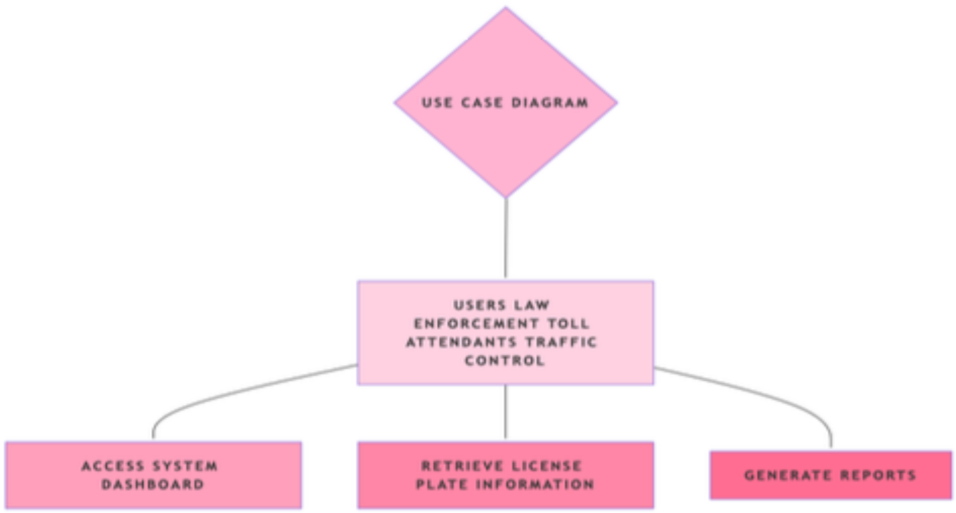
### 5.3.1. Product Backlog Items

1. In an effort to track vehicles accurately and uphold road safety, as a traffic cop, I want to be able to automatically recognize license plates in foggy conditions.
2. I want to supervise and handle camera feeds as an administrator to make sure that real time data is correctly recorded and processed.
3. In my capacity as a developer, I would like to add an image enhancement module to enable the system to enhance hazy car photos caused by fog.
4. To be able to take prompt action as a law enforcement officer, I would like to receive an alert whenever an unregistered or flagged license plate is detected.
5. As a way to make verifying data simple, as a user, I want the system to provide both textual and visual outputs of license plate numbers.
6. I want to continuously take images in fog as a surveillance unit to make sure no car passes by unnoticed.

### 5.3.1. Architecture Diagram




### 5.3.2. Use Case Diagram




### 5.3.3. Class Diagram

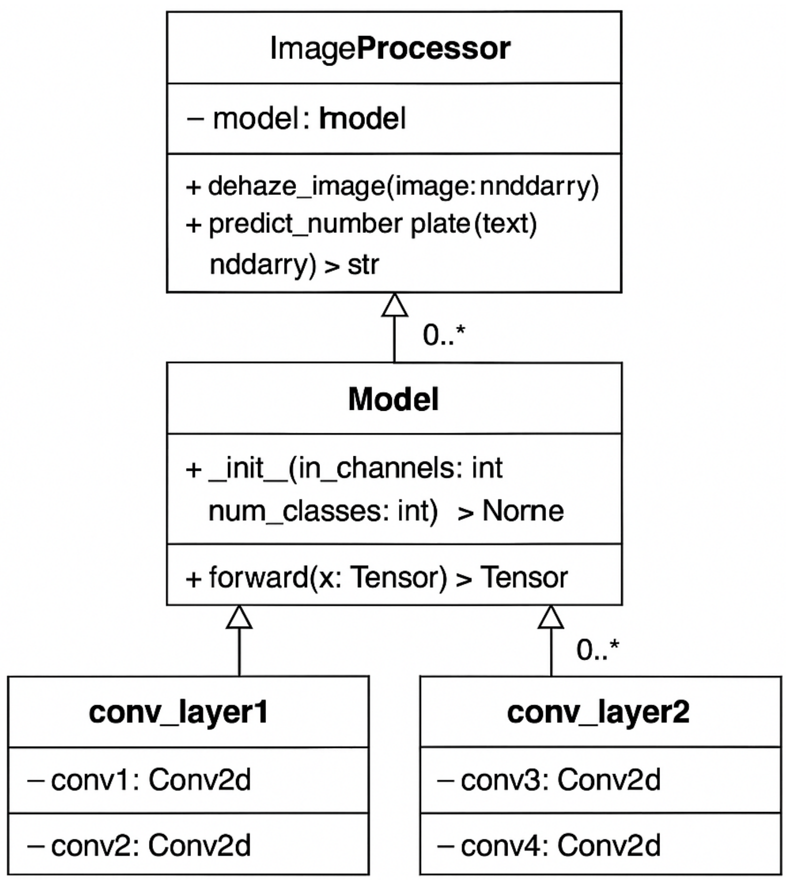
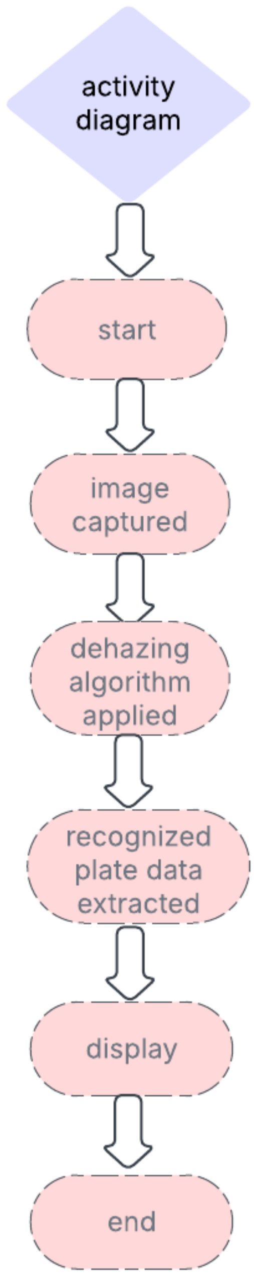



**Figure 1**

### 5.3.4. Activity Diagram



### 5.3.5.Sequence diagram

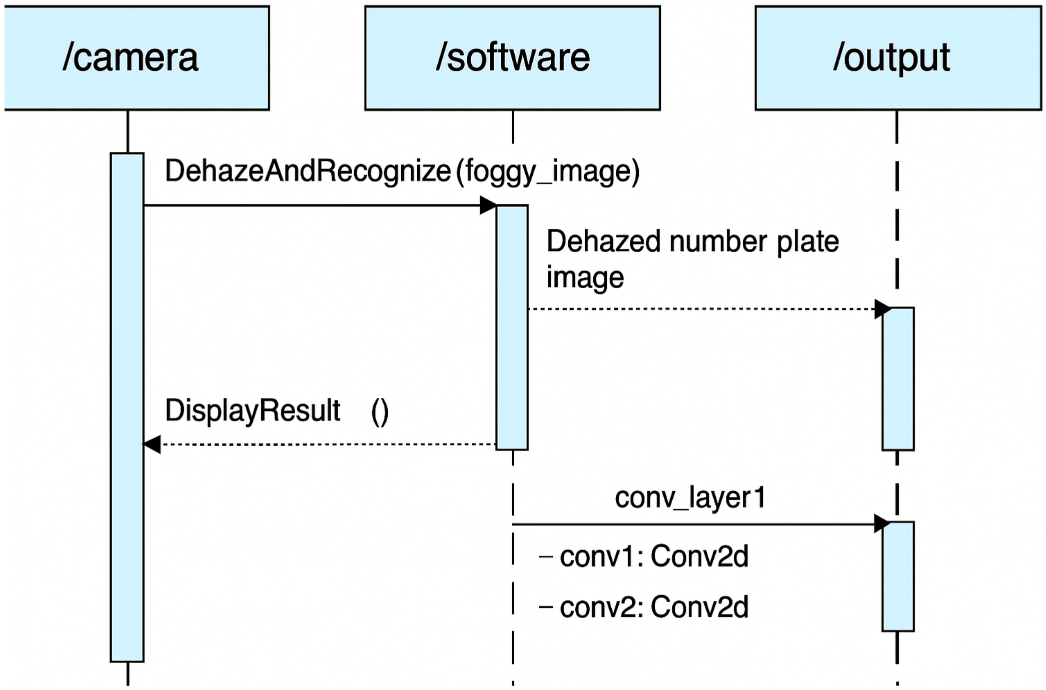



**Figure 2**

### 5.3.6. Data Architecture

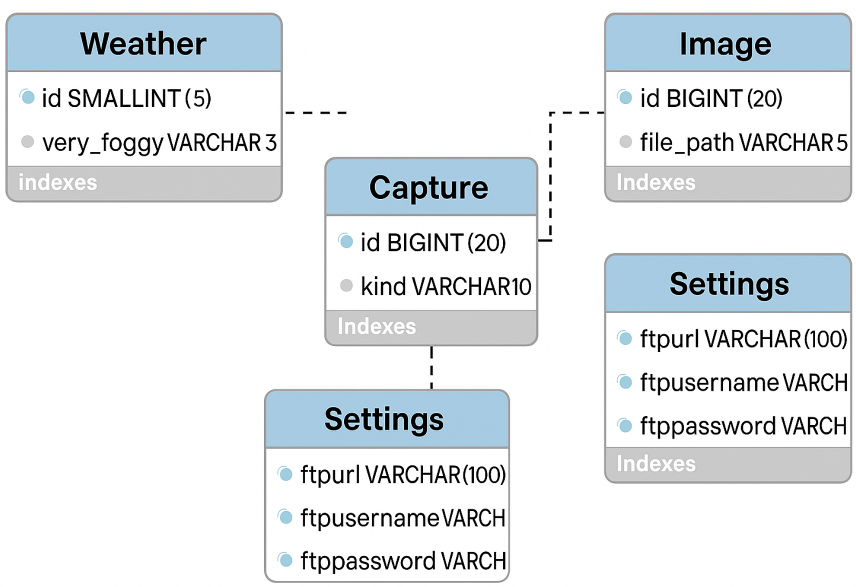



**Figure 3**

# 6. USER INTERFACE

## 6.1. UI Description

The UI consists of the following screens:

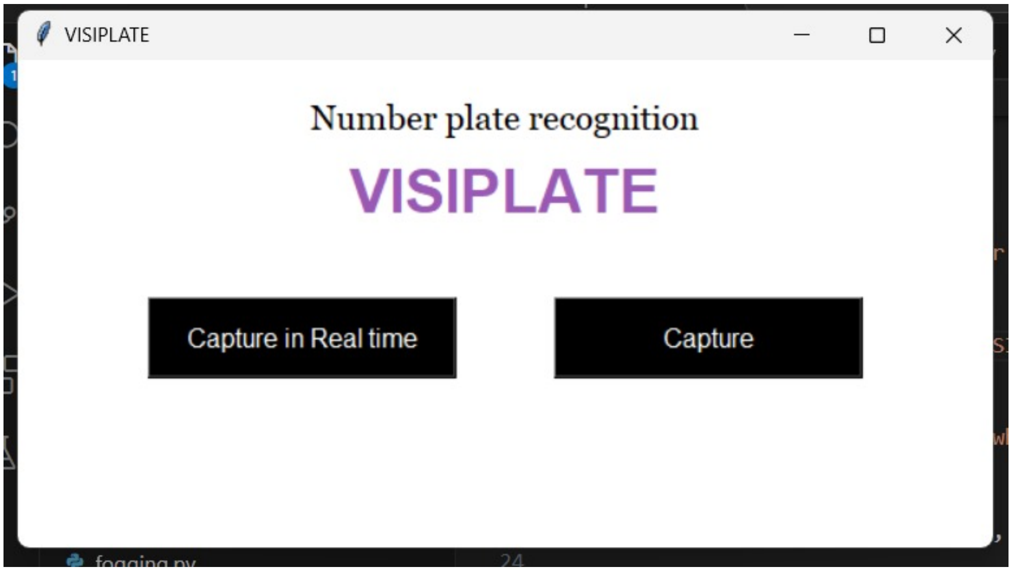
**1. Home Screen - VISPLATE Dashboard**

- **Title:** "Number Plate Recognition - VISPLATE"
- **Buttons:**
    - **Capture in Real-time:** This option activates the connected CCTV or webcam to capture live video footage. The system then processes the frames in real-time and detect number plate before extracting it.
    - **Capture:** This allows users to upload a static image instead of live footage.
- **User Interaction:** Users select either live or static input depending on their needs. It offers a clean and minimal design to avoid confusion and improve accessibility.

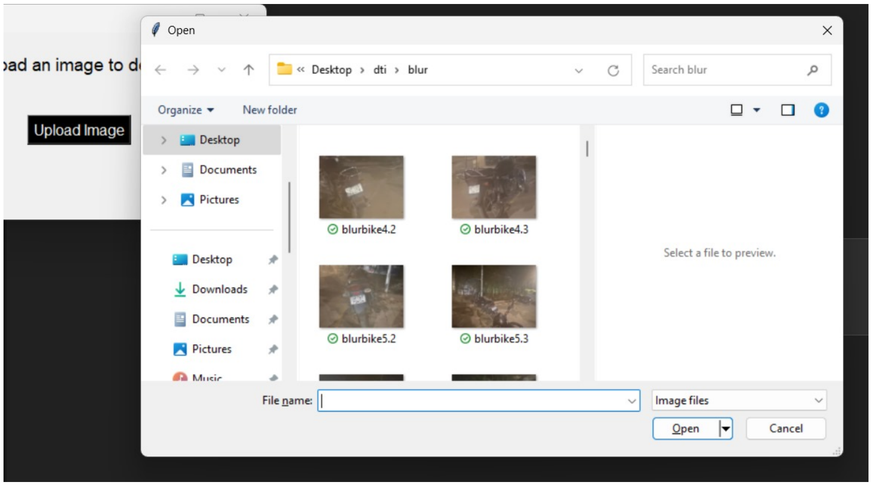
**2. Defogging Window**

- **Title:** "Upload an image to defog it"
- **Button:**
    - **Upload Image:** Allows users to select and upload an image affected by fog. Once uploaded, the system runs the dehazing algorithm to produce a clearer version of the image before passing it to the number plate recognition.
- **User Interaction:** Users click the button to browse and choose a foggy image. The system then processes the image using deep learning (CNN) and enhancement techniques (via libraries like cv2, Pillow, and PyTorch) and displays the improved result.

## 6.2. UI Mockup


VISIPLATE
Number plate recognition
VISIPLATE
Capture in Real time
Capture


Defog Image
Upload an image to defog it
Upload Image


Upload Image
Open
Desktop › dti › blur
Search blur
Organize
New folder
Desktop
Documents
Pictures
Desktop
Downloads
Documents
Pictures
blurbike4.2
blurbike4.3
blurbike5.2
blurbike5.3
Select a file to preview.
File name:
Image files
Open
Cancel


UP92A
J1894

**Figure 4:**

# 7. ALGORITHMS/PSEUDO CODE OF CORE FUNCTIONALITY

Algorithms used/pseudo code-

- Dehazenet
- Matplotlib
- Pytorch
- Pillow
- Transformers
- cv2
- numpy

Goal: Defog an input image using image processing techniques:

**Function: Defog Image**

1. Load the input foggy image.
2. Convert the image to dark channel:
    - For every pixel, take the minimum RGB value in a window around it.
3. Estimate atmospheric light:
    - Choose top bright pixels in the dark channel.
    - Calculate their average brightness in the original image.
4. Estimate initial transmission map:
    - Divide image by atmospheric light.
    - Compute dark channel again.
    - Use a constant factor to calculate transmission.
5. Refine the transmission map using guided filtering:
    - Smooth the transmission map using the original image as guidance.
6. Recover the scene radiance:
    - Use the refined transmission map and atmospheric light.
    - Apply the equation:
      $J = (I - A) / \max(t, t0) + A$
7. Return the clean, defogged image.

Model Components:

**DEHAZENET-**

**1. Convolutional Block (ConvBlock)**

- Input an image.
- Apply convolution → batch normalization → ReLU activation.
- Output the result.

**2. Residual Block (ResBlock)**

- Input feature map.
- Apply two ConvBlocks.
- Add original input to output (residual connection).
- Return the final result.

**3. Encoder**

- Input image.

- Stage 1: Apply ResBlock + downsample.
- Stage 2: Apply ResBlock + downsample.
- Stage 3: Apply ResBlock.
- Output features from each stage.

**4. Decoder**

- Input: Features from Encoder.
- Stage 1: Upsample and combine with encoder's last stage.
- Stage 2: Upsample and combine with encoder's middle stage.
- Stage 3: Upsample and combine with encoder's first stage.
- Final convolution to reconstruct the cleaned image.
- Return the defogged image.

# 8. PROJECT CLOSURE

## 8.1. Goals / Vision

- Our initial objective for this project was to use real-time camera feeds and sophisticated image processing techniques to create a system that could recognize license plates in foggy weather. Enhancing visibility through fog, extracting information from license plates, and storing it for use by law enforcement and analysis were the goals.

- As the project went on, our objectives changed to include creating a reliable fog removal model using deep learning, incorporating CV2 for precise image detection and creating a portable, lightweight system that can be integrated with roadside surveillance infrastructure. Building an extensible architecture that enables future updates, such as dust/smog handling or night vision support, was also emphasized.
- We focused on real-time performance, accuracy in a range of weather conditions, and user-friendly design in addition to creating a working prototype. By increasing vehicle traceability, lowering traffic crimes, and boosting public safety in low-visibility situations, we hope that this solution will greatly assist law enforcement and traffic control departments.

## 8.2. Delivered Solution

- Our project's original goal was to create a camera-integrated system that, by combining deep learning, computer vision, and image enhancement techniques, could recognize license plates even in foggy conditions. The suggested strategy called for removing fog in real time, identifying the area of the license plate, and taking pictures for later processing and archiving.
- We have produced a reliable and modular solution that consists of a fog-removal module that uses deep learning models based on TensorFlow, precise number plate detection using OpenCV (cv2), and smooth image capture in low visibility. Even in inclement weather, the system can efficiently detect and isolate vehicle number plates and process real-time input from a camera.
- Clean and organized code, documentation, test reports, installation guidelines, and a user manual are also among our deliverables. In addition to being scalable for future enhancements to handle additional environmental challenges like rain, snow, or low light, the solution is optimized for real-time performance and is simple to deploy with roadside cameras.

## 8.3. Remaining Work

- The current system uses OpenCV and TensorFlow to detect vehicle number plates in foggy conditions. Future developments, however, could boost scalability and overall performance. Integrating the system with cloud-based storage and an alert system to automatically record and report recognized license plates is one potential next step. Furthermore, it is strongly advised to increase model accuracy by training on a bigger and more varied dataset, which should include extreme weather conditions like intense rain and low light levels.
- Real-time notification systems, multilingual number plate recognition, and audio feedback integration are possible extra improvements. By using edge devices like the Raspberry Pi or NVIDIA Jetson Nano for real-time processing in resource-constrained environments, the hardware side could also be optimized.

# REFERENCES


- Hassan, R., Mahmud, M., & Hasan, M. (2019). Challenges in Vehicle License Plate Recognition: A Survey. International Journal of Advanced Computer Science and Applications.
- Zhang, Y., Wang, Y., & Liu, C. (2020). Deep Learning-Based Approach for Vehicle Plate Detection under Adverse Weather. IEEE Transactions on Intelligent Transportation Systems.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. CVPR.
- He, K., Sun, J., & Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Anwar, S., & Barnes, N. (2021). Dense Haze Removal with CNNs. IEEE Access.
- Patel, D., Joshi, K., & Shah, M. (2021). Vehicle Number Plate Recognition using Tesseract OCR with Preprocessing Techniques. IJCSMC.
- Ministry of Road Transport and Highways (2022). Annual Report on Traffic Infrastructure and Smart Surveillance.
- https://youtu.be/QzY57FaENXg?si=mwbcmvJD78OXSxX5
- https://youtu.be/0-4p_QgrdbE?si=LRrzoipnSPr9AE6f
- https://www.geeksforgeeks.org/introduction-convolution-neural-network/
- https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/
- https://www.researchgate.net/publication/377257838_CNN_based_System_for_Automatic_Number_Plate_Recognition