INTELLIGENT LICENSE PLATE RECOGNITION USING COMPUTER VISION

MINOR PROJECT REPORT

Submitted in partial fulfilment of the requirements for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ELECTRICAL & ELECTRONICS ENGINEERING

by

Rahul Garg
Enrollment No:
03414804922

Jay Chaturvedi
Enrollment No:
04314804922

Guided by

Mr. Rahul Garg
Assistant Professor

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING
MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY
(AFFILIATED TO GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY, DELHI)
DELHI – 110086

November-2025

CANDIDATE'S DECLARATION

It is hereby certified that the work which is being presented in the B. Tech Minor Project Report entitled **"INTELLIGENT LICENSE PLATE RECOGNITION USING COMPUTER VISION"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology (Electrical & Electronics Engineering)** and submitted in the **Department of Electrical & Electronics Engineering** of **MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi)** is an authentic record of my/our own work carried out under the guidance of **Rahul Garg, Assistant Professor.**

The matter presented in the B. Tech Minor Project Report has not been submitted by me/us for the award of any other degree or diploma of any Institute.

Rahul Garg
Enrollment No: 03414804922

Jay Chaturvedi
Enrollment No: 04314804922

CERTIFICATE

This is to certify that the minor project entitled **" INTELLIGENT LICENSE PLATE RECOGNITION USING COMPUTER VISION"** being submitted by "**Rahul Garg (Enrollment No:03414804922) and Jay Chaturvedi (Enrollment No:04314804922)**" to the **Department of Electrical & Electronics Engineering** of **MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi)** for the award of the Degree of **Bachelor of Technology (Electrical & Electronics Engineering)** in **Electrical & Electronics Engineering**, is a bonafide project work carried out by him/them under my supervision and guidance. His/their work has reached the standard of fulfilling the requirements of regulations relating to degree. The project report is an original piece of work and embodies the findings made by the students themself.

The results presented have not been submitted in part or in full to any other University/Institute for the award of any degree or diploma.

Rahul Garg	**Dr. Monika Gupta**
Assistant Professor	**H. O. D (EEE)**

The B. Tech Minor Project Viva-Voce Examination of **Rahul Garg (Enrollment No: 03414804922) and Jay Chaturvedi (Enrollment No: 04314804922),** has been held on **……………………………….**

(Signature of External Examiner) **(Signature of External Examiner)**

ACKNOWLEDGEMENT

At the very outset, I/We like to record my heartfelt gratitude to my respected teacher and mentor, **Rahul Garg**, Assistant Professor, Department of Electrical & Electronics Engineering, for his valuable guidance and suggestions throughout my project work.

I/We would like to extend my/our sincere thanks to the **Head of the Department, Dr Monika Gupta,** for her time-to-time suggestions to complete my project work.

We are thankful to the Project Coordinators for their valuable guidance. I shall be failing in my duty if I do not express thanks to all the faculty members and staff of the Department of Electrical & Electronics Engineering for providing me/us the facilities to carry out my/our project work.

Note: Students may thank to the persons whom they would like to acknowledge.

Sign	**Sign**
Rahul Garg	**Jay Chaturvedi**
Enrollment No.:	**Enrollment No.:**
03414804822	**04314804922**

ABSTRACT

The increasing number of vehicles on campuses and in urban areas has made manual verification of vehicle entries and exits highly inefficient and prone to error. Security personnel often face challenges such as time-consuming manual record-keeping, susceptibility to human error, and exposure to harsh working conditions. To address these issues, this project proposes an Automated License Plate Detection and Recognition System using computer vision and deep learning techniques. The system captures images of vehicles at entry and exit points, preprocesses them using image enhancement and contour detection methods, and accurately extracts the license plate region. Optical Character Recognition (OCR) with Tesseract is initially employed for text recognition, followed by a Convolutional Neural Network (CNN) model to enhance character recognition accuracy. The proposed system eliminates reliance on handwritten logs, reduces human workload, and ensures improved security and efficiency by maintaining a digital record of all vehicle movements. Experimental results demonstrate the effectiveness of the approach in real-world scenarios with varying lighting and environmental conditions. This solution has the potential to be integrated into campus security systems, smart parking management, and intelligent traffic monitoring applications.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Background

With the rising number of vehicles entering institutional and restricted premises, the need for effective and efficient vehicle monitoring systems has become more important than ever. Traditionally, campuses and organizations rely on manual verification of license plates at entry and exit points. Security personnel record vehicle numbers in handwritten logbooks, which are vulnerable to damage, human error, and significant delays during peak traffic hours. This manual approach strains the security staff and impacts the overall efficiency of the security system. As a result, automation through License Plate Detection and Recognition (LPDR) systems has emerged as a practical solution to streamline these operations.

1.2 Need for Automated License Plate Detection

Manual license plate verification suffers from inherent limitations such as misreading numbers, slow processing, improper record maintenance, and fatigue-induced errors. Additionally, security guards face harsh weather conditions while performing these duties. An automated LPDR system minimizes human involvement, thereby reducing errors, improving accuracy, and enhancing the working conditions of the personnel. Digital record-keeping also ensures easy retrieval, safety from physical damage, and long-term data storage.

1.3 Problem Statement

In the existing system at our campus, every vehicle's entry and exit is manually logged. This process is time-consuming and prone to inaccuracies due to illegible handwriting or miscommunication. The objective of this project is to develop an automated license plate detection and recognition system capable of identifying vehicle number plates from images and recording them accurately. By automating this task, the system will streamline the verification process, reduce dependency on manual effort, and ensure a more reliable method of vehicle monitoring.

1.4 Objectives of the Project

The goal of this project is to design and implement a robust license plate detection and recognition system using image processing and deep learning. The major objectives include:

- To detect and isolate the number plate from the vehicle image using classical image processing techniques.
- To preprocess the plate region using grayscale conversion, thresholding, contour extraction, and morphological operations.
- To apply OCR methods, initially using Tesseract and later using a custom CNN model for better accuracy.
- To segment characters from the extracted plate using contour-based approaches.
- To train and deploy a CNN model capable of recognizing individual alphanumeric characters with high accuracy.
- To create a system that reduces manual work and increases the speed and efficiency of vehicle verification.

1.5 Scope of the Project

The developed system focuses on detecting and recognizing license plates from still images. It incorporates preprocessing, contour-based plate isolation, and recognition through both classical OCR and CNN-based methods. The scope includes training and testing the model using a curated dataset and evaluating system performance using accuracy and loss metrics. Though the project does not address vehicle tracking or real-time multi-camera integration, it forms a strong foundation for future expansion into automated gates, toll systems, and smart parking infrastructure.

1.6 Motivation

The motivation for this project stems from the need to modernize and automate vehicle monitoring on campus. Observations of long queues at gate checkpoints, frequent manual errors, and deteriorating offline record logs highlighted the urgency for automation. Additionally, the availability of powerful libraries like OpenCV and TensorFlow makes it feasible to develop a sophisticated yet accessible LPDR system. This project also provided an opportunity to apply theoretical knowledge of image processing, machine learning, and convolutional neural networks to a real-world problem.

1.7 Overview of the Proposed System

The proposed system integrates classical image processing techniques with deep learning-based character recognition. Plate detection is achieved through Gaussian blurring, grayscale conversion, Sobel edge detection, thresholding, and contour filtering. Once extracted, the number plate undergoes preprocessing using functions like clean2_plate, ratioCheck, and ratio_and_rotation to refine its boundaries. Character segmentation is performed using contour-based methods, and recognition is executed using both Tesseract OCR and a custom-trained CNN model. The system ensures improved accuracy, reduced manual involvement, and efficient data processing.

1.8 Chapter Summary

This chapter explained the context, need, and motivation behind developing an automated license plate detection system. The limitations of traditional manual verification methods highlight the importance of automation for accuracy and efficiency. The chapter also introduced the core objectives, overall system concept, and the technological foundation upon which the proposed solution is built. Subsequent chapters will detail the literature review, methodology, implementation, results, and analysis of the developed system.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

License Plate Detection and Recognition (LPDR) has emerged as a significant research area due to its applications in transportation management, automated access control, law enforcement, and smart surveillance systems. Over the past decade, researchers have developed a variety of techniques—ranging from classical image processing to deep learning—to improve the accuracy and robustness of license plate extraction and character recognition. This chapter reviews the existing methods, algorithms, and frameworks that form the foundation of the proposed system.

2.2 Traditional Image Processing Approaches

Early LPDR systems relied heavily on handcrafted feature extraction and rule-based filtering using classical computer vision techniques. Methods such as edge detection (Sobel, Canny), thresholding, and morphological operations were widely used. These approaches were efficient for controlled environments but faced limitations with complex backgrounds, varying lighting conditions, plate rotations, and low-quality images.

Several studies used contour-based detection to locate rectangular regions representing license plates. Area thresholds, aspect ratios, and pixel intensity checks helped filter false detections. The *ratioCheck* and *isMaxWhite* methodologies, similar to those used in this project, are common strategies in these traditional frameworks. OpenCV-based solutions have remained popular in academic and industrial settings due to their interpretability and real-time efficiency.

2.3 Machine Learning and OCR-Based Methods

As the field advanced, Optical Character Recognition (OCR) systems like Tesseract became widely adopted for recognizing characters extracted from license plates. These methods required clean, binarized images, making preprocessing steps such as noise reduction, thresholding, and segmentation crucial. While Tesseract performed well in controlled conditions, its accuracy suffered when plates contained shadows, distortions, or uneven illumination. This limitation led researchers to explore more robust character recognition techniques using machine learning and deep learning.

2.4 Deep Learning Advancements

The introduction of Convolutional Neural Networks (CNNs) significantly improved LPDR systems. CNNs automatically learn features from training data, removing the need for manual feature engineering. Research showed that CNN-based architectures could accurately classify alphanumeric characters even under varying noise, distortion, and resolution conditions. Models such as LeNet, VGG-like networks, and custom lightweight architectures became popular for character recognition tasks in LPDR systems.

Segmentation-free deep learning models also emerged, where entire license plates were treated as a sequence prediction problem. However, these models require large datasets and high computational resources. The project's approach of combining segmentation with CNN-based character classification offers a balance of accuracy, interpretability, and computational efficiency.

2.5 Hybrid Methods

Recent advancements integrate classical image processing for plate detection and deep learning for recognition. This hybrid approach provides reliable performance across diverse environments. Image processing techniques isolate the plate, while CNNs handle character recognition with higher accuracy than traditional OCR engines. The proposed system in this project adopts the same hybrid strategy by using preprocessing functions for extraction and a CNN model for recognition.

2.6 Summary of Literature Findings

The literature suggests that:

- Classical methods are effective for plate localization due to their speed and clarity.
- OCR-based recognition works for clean, noise-free images but struggles with complex plates.
- CNN-based recognition significantly improves accuracy, especially with character-level segmentation.
- Hybrid systems that combine detection algorithms with deep learning models offer optimal performance.

The reviewed literature provides the conceptual and technical basis for designing a reliable and accurate LPDR system.

CHAPTER 3 METHODOLOGY

3.1 Introduction

This chapter explains the methodology adopted to develop a complete License Plate Detection and Recognition system. The process integrates classical image processing for plate extraction and deep learning for character recognition. The system is designed to ensure accuracy, modularity, and efficiency.

3.2 System Architecture Overview

The overall system architecture consists of the following stages:

1. **Image Acquisition**
   A vehicle image is captured and provided as input to the detection module.
2. **Preprocessing and Plate Detection**
   The image undergoes blurring, grayscale conversion, edge detection, thresholding, and contour analysis.
3. **Plate Extraction and Cleaning**
   Functions like *clean2_plate*, *ratioCheck*, and *isMaxWhite* refine the extracted region.
4. **Character Segmentation**
   The binary plate image is segmented into individual character blocks using contour-based logic.
5. **Character Recognition**
   Extracted characters are recognized using Tesseract OCR and a custom-trained CNN for improved accuracy.
6. **Output Generation**
   The final license plate number is reconstructed by combining recognized characters.

Each component is designed and optimized based on the theoretical principles and practical experimentation described below.

3.3 Preprocessing Techniques

Preprocessing prepares the image for detection and enhances the clarity of potential license plate regions.

3.3.1 Gaussian Blurring

Gaussian blur is applied to reduce noise and smoothen the input image, which helps in edge detection. A blurred image ensures that irrelevant details do not interfere with contour extraction.

3.3.2 Grayscale Conversion

The blurred image is converted to grayscale to simplify computations by reducing the three-channel color image to a single-channel intensity image.

3.3.3 Edge Detection Using Sobel Operator

The Sobel operator highlights vertical edges, which are typically more prominent in license plates. This step helps in isolating regions with strong gradients.

3.3.4 Thresholding Using Otsu's Method

Otsu's thresholding converts the grayscale image into a binary image by automatically selecting an optimal threshold. This simplifies subsequent morphological operations.

3.3.5 Morphological Operations

Morphological closing is applied to fill gaps and strengthen plate-like contours. It helps improve contour accuracy when extracting potential plate regions.

3.4 License Plate Detection

After preprocessing, contours are extracted from the binary image. For each contour, a minimum bounding rectangle is computed using *cv2.minAreaRect*. The following checks are applied:

3.4.1 Aspect Ratio Filtering (ratioCheck)

The bounding box must fall within a predefined aspect ratio range to qualify as a potential license plate.

3.4.2 Rotation Check (ratio_and_rotation)

The rectangular region must have a valid orientation. Excessively rotated or distorted plates are rejected.

3.4.3 Intensity Check (isMaxWhite)

Most license plates have a predominantly white background. The mean intensity is used to validate candidate regions.

Contours passing all checks are labeled as valid number plate regions.

3.5 Plate Cleaning and Refinement

Using the *clean2_plate* function:

- The region is thresholded.
- Largest contours inside the plate are extracted.
- Bounding boxes are computed.
- The cleaned plate region is isolated and returned.

This ensures that only the alphanumeric characters remain for further processing.

3.6 Character Segmentation

Character segmentation isolates each character from the cleaned plate.

3.6.1 Binary Conversion and Morphology

The plate is resized, converted to grayscale, thresholded, and passed through erosion and dilation to highlight character contours.

3.6.2 Contour Extraction (find_contours)

Contours representing characters are detected based on:

- width and height thresholds
- x-coordinate sorting
- border cleaning for clarity

Each character is resized to a standard dimension (20×40) and placed inside a 24×44 padded frame to prepare for CNN input.

3.6.3 Character Ordering

Characters are sorted left-to-right using their x-coordinates to reconstruct the sequence correctly.

3.7 Character Recognition

Two approaches are used:

3.7.1 Tesseract OCR

The cleaned plate image is passed to the *image to string* function.
While simple to implement, Tesseract may struggle with noisy or low-resolution images.

3.7.2 CNN-Based Recognition

To improve accuracy, a custom CNN is trained on segmented characters.

Model Features:

- Multiple convolutional and pooling layers
- ReLU activation
- Dropout regularisation
- Dense layers with softmax output

Training Pipeline:

- Data augmentation using ImageDataGenerator
- Rescaling and normalization
- Training on labeled character datasets
- Validation and accuracy tracking

The trained weights are saved and later used for recognition.

3.8 Output Generation

- The characters predicted by OCR/CNN are concatenated.
- The final license plate number is displayed.
- Character images are shown alongside their predicted labels for validation.

CHAPTER 4 IMPLEMENTATION

4.1 Introduction

This chapter summarizes the practical implementation of the License Plate Detection and Recognition system. The implementation integrates image preprocessing, plate extraction, character segmentation, and OCR/CNN-based recognition using Python, OpenCV, and TensorFlow.

4.2 Tools and Development Environment

The system was developed in Python using the following libraries:

- **OpenCV** for image processing
- **NumPy** for numerical operations
- **Pytesseract** for initial OCR
- **TensorFlow/Keras** for CNN model training

  All experiments were performed in Jupyter Notebook/VS Code on a standard laptop.

4.3 Preprocessing and Plate Detection

The input image undergoes a simplified sequence of preprocessing steps:

1. **Noise Reduction** using Gaussian blur
2. **Grayscale Conversion** to simplify processing
3. **Edge Detection** using the Sobel operator
4. **Thresholding** (Otsu method) to obtain a binary image
5. **Morphological Closing** to strengthen plate-like contours

Contours are then extracted from the processed image. Each contour is evaluated using aspect ratio, area range, rotation, and mean intensity checks. The contour that satisfies all criteria is considered the license plate region and is cropped from the image.

Fig 4.3.1: Original Image

Fig 4.3.2: Blurred Image

Fig 4.3.3: Sobel Detection Output

Fig 4.3.4: Thresholding Imag

Fig 4.3.5: Morphologically Closed Image

Fig 4.3.6: Detected Licence Plate Region

4.4 Plate Cleaning and Character Segmentation

Once the plate is extracted, it is refined using the clean2_plate function. The cleaned plate is converted to binary form, noise is removed, and the image is resized. Character segmentation is performed using contour detection on the binary plate. Valid character contours are filtered based on height and width, sorted left-to-right, and resized to a standard dimension suitable for recognition.

Fig 4.4.1: Segmented Characters

4.5 Recognition Using OCR and CNN

Two recognition approaches were implemented:

- **Tesseract OCR:** Provides quick extraction but suffers with noisy or low-contrast plates.
- **CNN-Based Recognition:** A custom convolutional neural network was trained on augmented character datasets. The model consists of convolution, pooling, dropout, and dense layers. After training, its weights were saved and later used to recognize the segmented characters. This approach achieved significantly better accuracy than Tesseract.

4.6 Output Generation

The predicted characters are combined to form the final license plate number. The system displays both the segmented characters and the recognized output for verification.

CHAPTER 5 RESULTS AND DISCUSSION

5.1 Introduction

This chapter presents the outcomes of the developed License Plate Detection and Recognition system. The results include preprocessing outputs, plate extraction accuracy, character segmentation quality, and OCR/CNN recognition performance.

5.2 Preprocessing and Detection Results

The preprocessing steps—blurring, grayscale conversion, edge detection, thresholding, and morphological operations—produced clean and well-defined images suitable for contour detection. The system consistently identified the license plate region based on aspect ratio, area, and intensity criteria. Across multiple test images, the plate detection module showed high reliability under normal lighting conditions.

5.3 Character Segmentation Results

After extraction, the license plate was cleaned and converted into a binary form that enabled clear contour identification. Valid character contours were accurately isolated, resized, and ordered left-to-right. This ensured that each character was supplied correctly to the recognition module.

5.4 Recognition Results

Two recognition techniques were tested:

Tesseract OCR

- Provided quick results
- Accuracy dropped on noisy or low-contrast plates
- Suitable only when the image is clean and sharp

CNN-Based Recognition

- Achieved significantly higher accuracy
- Effectively handled variations in size, noise, and illumination
- Learned robust features from the training dataset

The CNN model produced consistent character predictions and performed better than Tesseract in all challenging images.

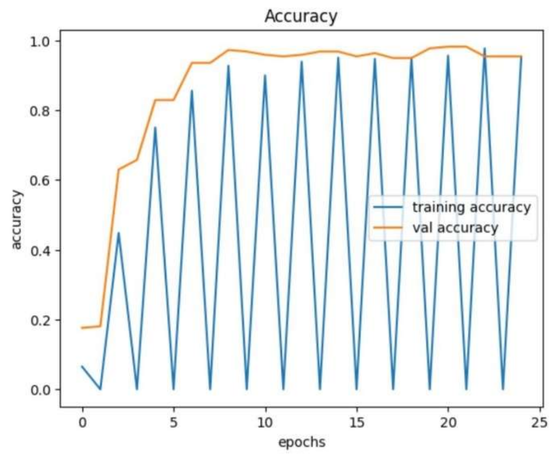

Fig 5.4.1: CNN Training and Validation Accuracy

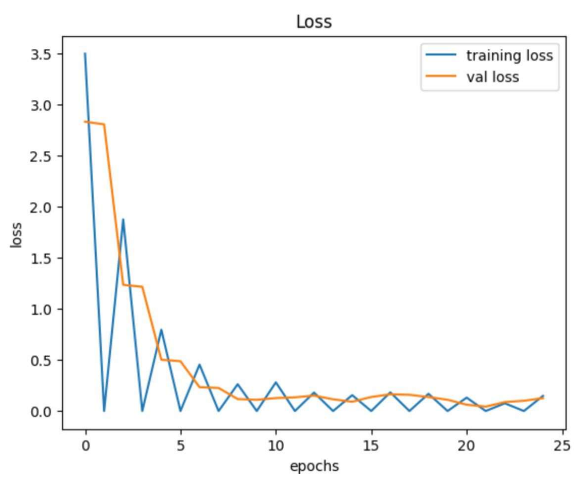

Fig 5.4.2: CNN Training and Validation Loss

5.5 Final Output

The integrated system successfully detected, segmented, and recognized license plates. Example output: **MH14078831**

The overall workflow—from preprocessing to CNN prediction—demonstrated stable performance, confirming the effectiveness of the hybrid approach.

5.6 Discussion

The results clearly show that combining classical image processing for plate detection with deep learning for character recognition produces a reliable and efficient LPDR system. While detection works well under standard conditions, recognition accuracy improves substantially with the CNN model compared to traditional OCR.

CHAPTER 6 CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

This project successfully implemented a complete License Plate Detection and Recognition (LPDR) system using a combination of classical image processing and deep learning. The detection module accurately identified and extracted license plates through a series of preprocessing steps such as blurring, edge detection, thresholding, and morphology. Character segmentation was achieved reliably using contour-based filtering, ensuring clean and ordered inputs for the recognition model. While Tesseract OCR provided a basic level of recognition, the custom-trained CNN model significantly improved accuracy and stability, especially for noisy or low-contrast plates. The hybrid approach proved to be effective, efficient, and suitable for real-world environments such as campus entry automation, parking systems, and security checkpoints.

Overall, the system demonstrates how integrating traditional computer vision techniques with modern neural networks can produce a robust and practical LPDR solution.

6.2 Limitations

Despite its effectiveness, the system has a few limitations:

- Performance decreases on blurry, very low-resolution, or heavily shadowed images.
- Plates with decorative fonts, non-standard spacing, or significant dirt may result in segmentation errors.
- The system processes still images; real-time video processing is not included in this version.

6.3 Future Scope

The project can be extended and enhanced in several ways:

- **Real-time integration:** Implementing the system with live CCTV feeds for continuous monitoring.
- **Deep learning–based plate detection (YOLO, SSD):** To replace contour-based detection and improve robustness.
- **Segmentation-free recognition models:** Using CRNN or transformer-based OCR to eliminate the need for character segmentation.

- **Cloud-based storage & dashboards:** For maintaining vehicle logs, analytics, and security alerts.
- **Automatic gate control:** Integrating LPDR output with IoT systems for automated access control.
- **Support for multiple countries:** Training models to recognize plates with different fonts and formats.

These enhancements would make the system more scalable, versatile, and applicable to advanced smart city applications.

REFERENCES

[1] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[2] OpenCV Documentation, "Open Source Computer Vision Library." [Online]. Available: https://docs.opencv.org

[3] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2005.

[4] B. Patel, M. Shah, and N. Patel, "License Plate Detection Using Image Processing and Neural Networks," *IOSR J. Comput. Eng.*, vol. 1, no. 1, pp. 28–33, 2017.

[5] Symisc Research, "SOD — License Plate Detection Articles." [Online]. Available: https://sod.pixlab.io/articles

[6] N. Wongsirikul, "Non-deep Learning License Plate Detection Implementation," *Medium*, 2019.

[7] Pytesseract Documentation, "Python-tesseract." [Online]. Available: https://pypi.org/project/pytesseract

[8] R. Smith, "An Overview of the Tesseract OCR Engine," in *Proc. Int. Conf. Document Analysis and Recognition (ICDAR)*, 2007.

[9] F. Chollet, "Keras," 2015. [Online]. Available: https://keras.io

[10] M. Abadi *et al.*, "TensorFlow: A System for Large-Scale Machine Learning," in *Proc. USENIX OSDI*, 2016.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[12] R. Panahi and I. Gholampour, "Accurate Detection and Recognition of Dirty and Low-Resolution License Plates Using Convolutional Neural Networks," *IEEE Trans. Intell. Transp. Syst.*, 2017.