Journal of Soft Computing Paradigm (ISSN: 2582-2640)
www.irojournals.com/jscp/

CNN based System for Automatic Number Plate Recognition

Gobinda Pandey*[1], Karun K C[2], Nirajan Lamichhane[3], Utsav Subedi[4]

[*1,2]Department of Electronics and Computer Engineering, Institute of Engineering, Pashchimanchal Campus, Pokhara, Nepal

[3,4]Department of Computer Science, St. Xavier's College, Maitighar, Kathmandu, Nepal

E-mail: govindapandey223@gmail.com*[1], karun92591@gmail.com[2], lcnirajan47@gmail.com[3], subediutsav8@gmail.com[4]

Abstract

This study presents a comprehensive approach to Automated Vehicle Number Plate Detection and Recognition, employing image processing and Convolutional Neural Networks (CNNs). The system encompasses two main stages: number plate detection and recognition. Utilizing a digital camera, the system employs image processing to segment the number plate region accurately. A super-resolution method is then applied via CNNs to enhance the image quality. Subsequently, a bounding box method isolates individual characters for precise recognition. In the recognition phase, CNNs extract features for effective classification. The study aims to advance automated vehicle identification systems for law enforcement and parking management applications, promising accurate and efficient number plate detection and recognition. The proposed work has also developed a user interface to ensure the successfulness of the objectives aimed.

Keywords: Number Plate Detection, Number Plate Recognition, Image Processing, Convolutional Neural Networks (CNNs), Digital Camera, Bounding Box Method, Character Segmentation.

DOI: https://doi.org/10.36548/jscp.2023.4.002
Received: 10.11.2023, received in revised form: 15.12.2023, accepted: 29.12.2023, published: 09.01.2024

1. Introduction

1.1 Background

Vehicles on the road are rising in extensive numbers, particularly in proportion to the industrial revolution and the growing economy. The significant use of vehicles has increased the probability of traffic rule violations, causing unexpected accidents, and triggering traffic crimes. To overcome these problems, an intelligent traffic monitoring system is required. The intelligent system can play a vital role in traffic control through the number plate detection of vehicles. In this proposed research, a system will be developed for detecting and recognizing vehicle number plates using a Convolutional Neural Network (CNN), a deep learning technique.

1.2 Problem Statement

The drastic increase in vehicular traffic on the roadways stimulates a huge demand for technology for traffic monitoring and management. In this scenario, manual tracking of vehicles running fast on the road is practically not feasible. There will be a wastage of manpower and time. Even if it is operated manually, that will reflect huge difficulties and enormous errors. With the increase in the number of automobiles, it has become increasingly challenging to track them and almost impossible to identify the owners of these vehicles in case of a violation of any traffic law. Traffic law violations have been recognized as a major cause of road accidents in most parts of the world with the majority occurring in developing countries. Even with the presence of rules and regulations stipulated against this, violators are still on the increase. This is because the rules are not properly enforced by the appropriate authorities. Several cases of kidnapping, hit-and-run, robbery, smuggling, and on-road fatalities are continuously reported, and this is because these vehicles cannot be easily recognized, especially when moving at high speed.

1.3 Research Objectives

The main objectives of the research is listed below:

- Develop a system to detect the number plate of vehicles.

- Create a system that can recognize number plate, such as the vehicle number, and state information from a detected number plate.

1.4 Research Scope and Limitations

The number plate detection system aims to detect and recognize number plates, commonly employed in global traffic monitoring systems. ANPR benefits law enforcement by identifying suspicious or criminal vehicles and providing both alibis and incriminating data. It also enhances security in various settings, such as managing parking in workplaces or restricting access. However, the use of ANPR raises privacy concerns due to data storage. Despite these concerns, ANPR is not considered a privacy infringement, as data is securely stored and accessed only for valid reasons.

1.5 Development Methodology

A thorough review of research articles was conducted to identify optimal methodologies, parameters, and hyperparameters for the research. Utilizing Kaggle, the relevant dataset was obtained for English number plate detection, and a custom Nepali dataset was created through procedures like preprocessing, labeling, and conversion to XML.

The methodology involved training the model using the labeled dataset, saving checkpoints during training, and ultimately saving the trained model. A separate dataset was used for testing, followed by validation until a satisfactory level of accuracy was achieved. In cases of model rejection during validation, the preprocessing procedure was revisited, retaining the model and repeating the validation process until meeting specifications.

Once validated, the model was preserved for creating a classifier capable of detecting and recognizing number plates. This classifier was then employed to identify number plate characters and display them accordingly (refer to Figure 1).

2. Literature Review

2.1 Background and Contextual Research

A vehicle registration plate, or license plate, is a unique identifier attached to a vehicle for official purposes [1]. Automatic Number Plate Recognition (ANPR) uses Optical Character

Recognition to read plates, aiding law enforcement and electronic toll collection [2]. ANPR systems store images and text, employing deep learning for classification tasks [4], often using Convolutional Neural Networks (CNN) [5]. Optical Character Recognition (OCR) technology converts scanned documents into editable data [6]. Image processing involves transforming images for information extraction [7].

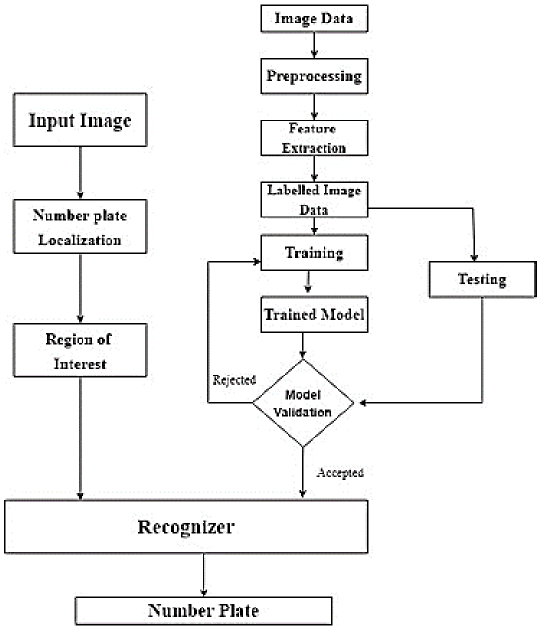

Figure 1. Development of Number Plate Detection System

2.2 Related Works

Several studies focus on license plate recognition using deep learning. Kilic and Aydin achieved 96.36% accuracy with a Turkish dataset using TensorFlow and Keras [20]. Cheang, and Teik Koon performed a “Segmentation-free vehicle license plate recognition using ConvNet-RNN”, they combined ConvNet-RNN for character recognition with 76% accuracy [12]. Henry and Chris, achieved a “Multinational license plate recognition using generalized character sequence detection” [14]. Other works applied CNN, k-means clustering, and genetic algorithms, reporting accuracies up to 98.86% [15, 16, 17]. Zhu used CNN for license plate recognition in urban roads with 82.5% accuracy [18]. Gupta introduced a high-resolution license plate generation method using GANs [10]. Akila used OCR for license plate recognition under various conditions [21]. Weihong achieved a segmentation rate of 82.6% and recognition precision of 87.3% [15]. Elihos used deep learning for character detection with 73.3% accuracy

[16]. Arafat proposed an edge-based CCA technique with 96.5%, 95.6%, and 94.4% accuracy [17]. Kashyap used OCR for character recognition with 82.6% accuracy [22]. Digital image processing emerged in the 1960s, with early applications in satellite imagery and medical imaging [8]. France introduced the first registration plate in 1893, with ANPR invented in 1976 [9, 10].

2.3 Related Works in Nepal

Pant, Gyawali, and Acharya implemented a Nepali number plate recognition system with 75.0% accuracy using Support Vector Machines [23]. Roy et al. developed a Vehicle License Plate (VLP) recognition and parking system using image processing and sensors [24].

3. Requirement Analysis

3.1 Software Requirements

i. Operating System: Windows or Linux or Mac OS
ii. Programming Language: Python
iii. Framework: Django framework
iv. Packages: Numpy, Pandas, OpenCV, Easy OCR

3.2 Hardware Requirements

i. Camera: An IP camera or webcam
ii. CPU: Pentium IV or higher
iii. RAM: At least 540 MB or more

3.3 Dataset Details

The automated vehicle number plate detection and recognition system is built on a dataset sourced from Kaggle for English number plate detection and a custom Nepali dataset created through preprocessing, labeling, and XML conversion. The training involved iterative model checkpoints and rigorous testing, ensuring accuracy. In cases of validation gaps, the pre-processing was refined until the criteria were met. The result is a system adept at detecting and recognizing number plates in both English and Nepali, displaying characters accurately.

3.4 Preprocessing, Labeling, and Conversion to XML

In preparing the dataset, a comprehensive three step process was followed involving preprocessing, labeling, and conversion to XML.

Preprocessing Step:

- Raw images underwent preprocessing to enhance quality and uniformity. Techniques included resizing to a standard dimension, pixel value normalization, and operations like grayscale conversion, histogram equalization, and noise reduction.
- Image augmentation techniques, such as rotation, translation, and flipping, were applied to increase model robustness against real-world image conditions.

Labeling Step:

- Each preprocessed image was manually labeled, marking the exact location of vehicle number plates with bounding boxes. A dedicated tool, labeling, facilitated this process.

Conversion to XML:

- Labeled data was converted to XML format, creating a well-structured dataset compatible with our training pipeline. XML files included filename, path, size, and bounding box coordinates, ensuring a standardized format for an effective model training.

3.5 Number Plate Detection Methods

The system incorporates sophisticated techniques for number plate detection [19]:

- **Number Plate Segmentation:** Utilizing image processing, we segmented number plates from the background by identifying distinct regions based on color, intensity, and shape characteristics.
- **Super Resolution Technique:** We applied super resolution methods to enhance image resolution, improving the clarity of number plate details for more accurate recognition.

- **Bounding Box Method for Character Segmentation:** For character-level segmentation within the number plate, a bounding box approach was employed. This ensured the precise isolation of characters for subsequent recognition steps.
- **CNN Architecture for Feature Extraction and Classification:** A Convolutional Neural Network (CNN) architecture was employed for feature extraction and classification. This deep learning model effectively learned hierarchical representations from input images, facilitating a robust number plate recognition.

3.6 System Components Interaction

The different components of the system interact seamlessly to achieve robust functionality. The user interface interacts with the preprocessing and labeling components during data preparation. The trained CNN model integrates with the number plate detection and recognition components for real-time processing. Finally, the results are presented through the user interface for user interaction.

3.7 Recognizer Components and Image Processing Techniques

The recognizer components and image processing techniques include:

- **Components:** The recognizer includes modules for number plate detection, character segmentation, and text recognition. Each module contributes to the overall recognition process.
- **Image Processing Techniques:** Techniques involve morphological operations for enhancing features, contour analysis for character segmentation, and Optical Character Recognition (OCR) for text extraction.
- **Mathematical Expression for Information Extraction:** Mathematical expressions, such as convolutional operations in the CNN architecture, are crucial for extracting meaningful information from images. These operations enable the network to learn hierarchical features essential for accurate recognition.

3.8 Algorithms Used for System Development

Various algorithms were employed in the development of the automatic number plate detection system. These algorithms are described briefly below:

i. **TensorFlow Object Detection:** Utilized for detecting, locating, and tracking objects in images or videos. It involves segment generation, feature extraction, and non-maximum suppression for bounding boxes.

ii. **Transfer Learning:** A machine learning method where pre-trained models are reused as starting points for new tasks. The research involved using early and central layers for computer vision tasks.

iii. **VGG Model:** Visual Geometry Group (VGG) model, a deep Convolutional Neural Network (CNN) architecture with 16 or 19 convolutional layers, is used for object recognition.

iv. **OpenCV:** A comprehensive open-source library for computer vision and image processing. Applied for real-time operations and image analysis, integrated with libraries like NumPy.

v. **Efficient Net:** Is a convolutional neural network architecture and scaling method that uniformly scales network dimensions using a compound coefficient.

vi. **ResNetV2:** Inception-ResNet-v2, a deep neural network architecture with residual connections, was trained on ImageNet for image classification.

vii. **SSD Mobile Net:** The single shot multibox detector algorithm, a faster alternative to Faster RCNN, used for object detection in the research.

viii. **Easy OCR:** The Optical Character Recognition (OCR) technology was employed to convert text in visuals to machine encoded text, useful for information extraction and text-to-speech applications.

4. Implementation and Testing

4.1 Implementation Details

The development of the research involved the use of various tools at different stages, including scheduling, planning, designing, coding, testing, and database handling.

i. **Python Programming Language:** Python an interpreted high-level programming language is used. Reasons for implementation: Readable and maintainable code, multiple programming paradigms, robust standard Library, many Open-Source frameworks and tools, simplify complex software development.

ii. **Django Framework:** Django high-level Python web framework. Reasons for choosing Django: Ridiculously fast, reassuringly secure, exceedingly scalable.

iii. **SQLite:** SQLite serverless, zero configuration, transactional SQL database engine. Reasons for choosing SQLite: No separate server required, zero configuration, single cross- platform disk file, Small and lightweight, Self-contained.

iv. **Figma:** Figma web-based vector graphics editor and prototyping tool. Features: Versatile across operating systems, collaborative design.

v. **Jupyter Notebook:** Jupyter Notebook open-source web application for live code, equations, visualizations, and text.

vi. **PyCharm:** PyCharm Integrated Development Environment (IDE) for Python. Features: Code analysis, debugger, unit tester, web development support.

vii. **Git and GitHub:** Git - distributed version control system. GitHub web-based platform for managing Git repositories.

viii. **Draw.io:** Draw.io free online diagram editor. Usage: Creating flowcharts, UML diagrams, Entity relation diagrams, etc.

ix. **MS Word:** Microsoft Word processor. Usage: documentation, text editing, formatting.

4.2 Testing

4.2.1 Unit Testing

Test Case 1: Number plate detection from an image (refer to Figure 2).

Figure 2. Test for Number Plate Detection

Test Case 2: Finding a region of interest in an image (refer to Figure 3).

Figure 3. Test for Finding Region of Interest

Test Case 3: Image recognition for English number plate (refer to Figure 4).

Figure 4. Test for Image Recognition of English Number Plate

Test Case 4: Image recognition for Nepali Number Plate (refer to Figure 5).

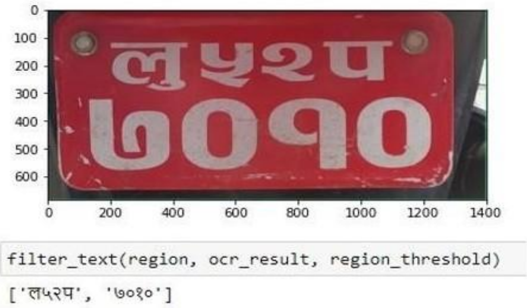

Figure 5. Test for Image Recognition of Nepali Number Plate

4.2.2 System Testing

Test Case 1: Invalid login (refer to Figure 6).

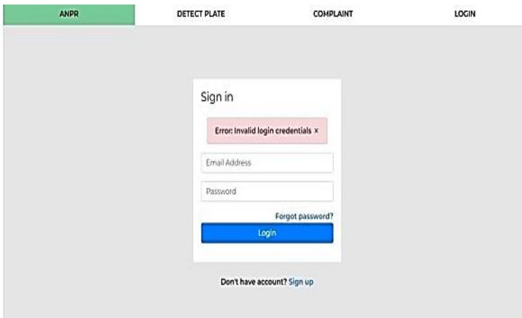

Figure 6. Test for Invalid Login

Test Case 2: Valid login and profile page display, Logout from the profile (refer to Figure 7).

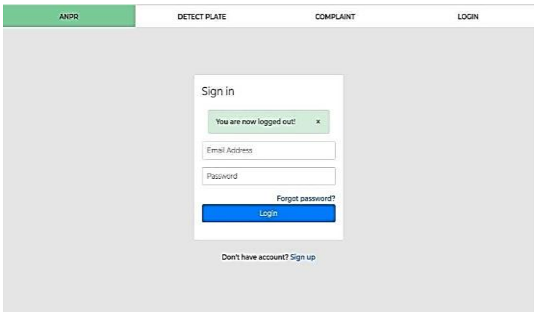

Figure 7. Test for Valid Login

Test Case 3: Password change (refer to Figure 8).

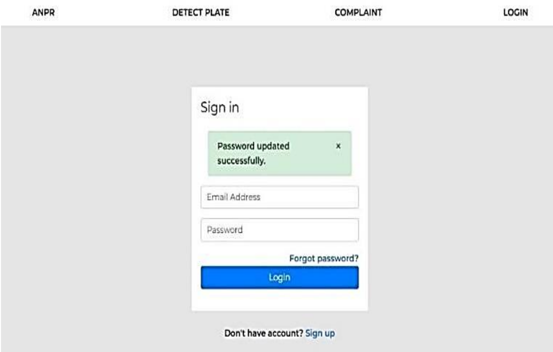

Figure 8. Test for Password Change

Test Case 4: Forgot password (refer to Figure 9).

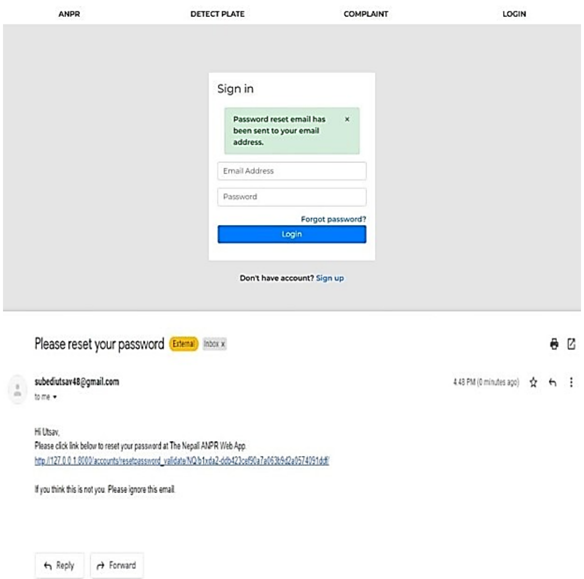

Figure 9. Test for Forgot Password

Test Case 5: Registering a complaint (refer to Figure 10).

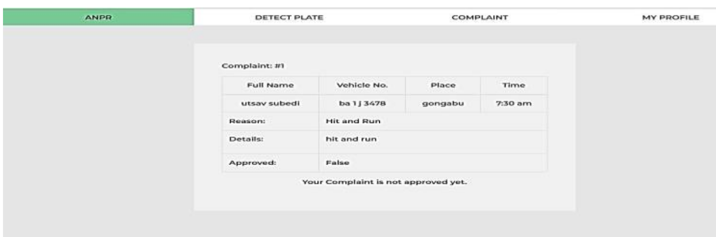

Figure 10. Test for Registering Complaint

5. Result Analysis

The performance analysis of the implemented system indicates successful achievement of objectives. Key functionalities, such as home screen navigation, number plate detection, complaint registration, and user profiles, that align with the research goals. The integration of algorithms and tools ensures robust and reliable performance in real-world scenarios.

Some user interface screenshots are listed as follows:

1. **Home Screen**: Navigation bar with menus including detect number plate, register complaint, and user's profile.
2. **Number Plate Detection**: Options to detect number plates via image upload or live cam (refer to Figure 11).
3. **Complaint Registration**: Form for registering complaints with details (refer to Figure 12).
4. **User's Profile**: Dashboard with registered complaints and user details (refer to Figure 13).
5. **Login and Register Page**: Pages for user authentication and registration (refer to Figure 14).
6. **Number Plate Detection Result**: Recognition of number plates with details (refer to Figure 15).

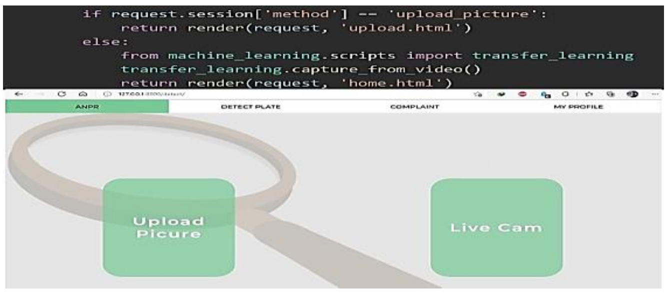

Figure 11. Detection Page of ANPR

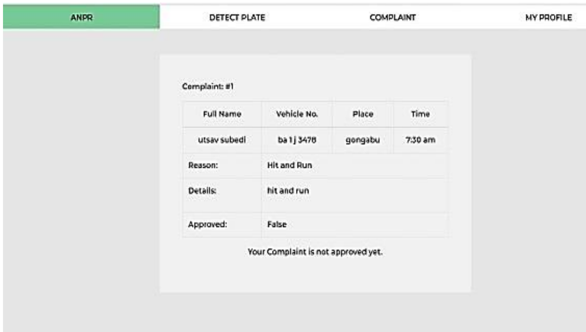

Figure 12. Registering Complaint on ANPR

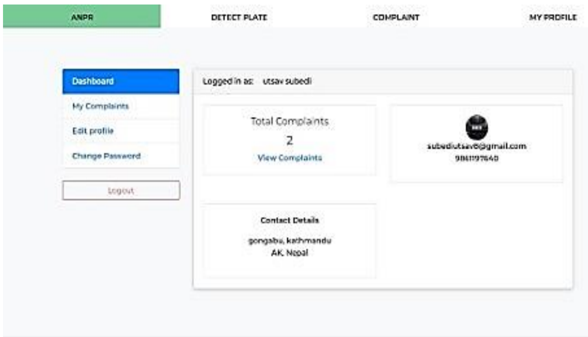

Figure 13. User’s profile on ANPR

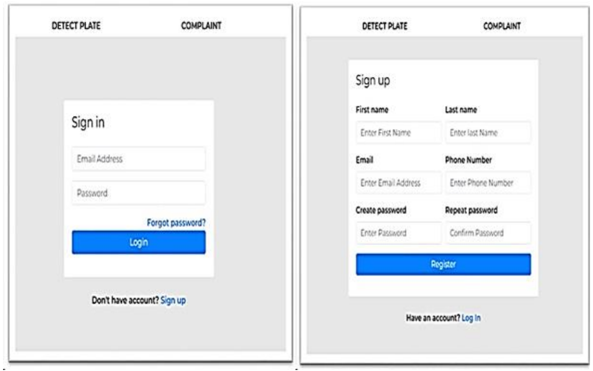

Figure 14. Login and Register Page of ANPR

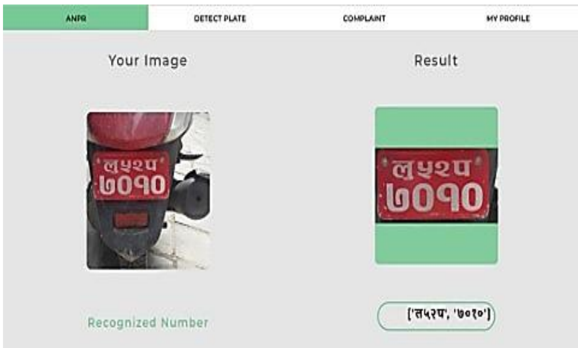

Figure 15. Recognition of Number Plate

6. Conclusion and Recommendation

6.1 Conclusion

The proposed system is implemented for the automatic vehicle license plate detection using Convolution Neural Network for character recognition. At first, it detects the plate area and extracts it. Then the characters are segmented from the extracted plate. All the characters are matched with the templates which are already created. Then the system recognizes the characters. The color images are used to test the system. The input images consist of different sizes, lighting, backgrounds, camera angles distance, etc. In most cases, number plates are successfully extracted and the characters were recognized. We achieved 93 percent accuracy with SSD Mobile Net, 83 percent with the Inception-ResNet model, and 75.5 percent with the Efficient Net approach. We also tried OpenCV, but it couldn't detect the license plate. We chose SSD Mobile Net as the model with the highest accuracy among these models. This model was integrated into the web application. The automatic vehicle number plate identification system plays an important role in detecting security threats. The operation of an automatic number plate detection system can be used by the police force to catch traffic law violators. If the vehicle breaks the law the image of the vehicle is captured and the ANPD process is started.

6.2 Future Recommendations

The total number of images to train the dataset can be increased. The collected images can be further increased to represent the variations in different weather conditions and different lighting conditions. For the recognition part, a more efficient OCR can be used or a custom model can be built to recognize Nepali fonts with more accuracy. The model can be further trained on different types of vehicles with an about equal sample of each type. Improvisation of the algorithms could be achieved to have a better performance even in different lighting conditions such as illumination and low light through the techniques of light reduction in illumination with the help of machine learning based approaches for the low light conditions.

References

[1] Baerwald, John Edward. "The functions and design of motor vehicle license plates." University of Illinois. Engineering Experiment Station. Bulletin; no. 457 (1960).

[2] Du, Shan, Mahmoud Ibrahim, Mohamed Shehata, and Wael Badawy. "Automatic license plate recognition (ALPR): A state-of-the-art review." IEEE Transactions on circuits and systems for video technology 23, no. 2 (2012): 311-325.

[3] Draghici, Sorin. "An Introduction to ANPR." University of Calgary, Alberta, Canada, 2017.

[4] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

[5] Saha, Sumit. "A comprehensive guide to convolutional neural networks—the ELI5 way." Towards data science 15 (2018): 15.

[6] Schantz, Hans. "The History of OCR." Recognition Technologies Users Association, Manchester, 2009.

[7] Shih, Frank Y. Image processing and pattern recognition: fundamentals and techniques. John Wiley & Sons, 2010.

[8] Gonzalez, Rafael C. "Digital Image Processing." Pearson Publications, New York, 2018.

[9] Robertson, Patrick. "The Book of The Firsts." New York: C. N. Potter, 1974.

[10] Gupta, Mayank, Abhinav Kumar, and Sriganesh Madhvanath. "Parametric Synthesis of Text on Stylized Backgrounds using PGGANs." arXiv preprint arXiv:1809.08488 (2018).

[11] Laroca, Rayson, Luiz A. Zanlorensi, Gabriel R. Gonçalves, Eduardo Todt, William Robson Schwartz, and David Menotti. "An efficient and layout-independent automatic license plate recognition system based on the YOLO detector." IET Intelligent Transport Systems 15, no. 4 (2021): 483-503.

[12] Cheang, Teik Koon, Yong Shean Chong, and Yong Haur Tay. "Segmentation-free vehicle license plate recognition using ConvNet-RNN." arXiv preprint arXiv:1701.06439 (2017).

[13] Parvin, Shahnaj, Liton Jude Rozario, and Md Ezharul Islam. "Vehicle number plate detection and recognition techniques: a review." Advances in Science, Technology and Engineering Systems Journal 6, no. 2 (2021): 423-438.

[14] Henry, Chris, Sung Yoon Ahn, and Sang-Woong Lee. "Multinational license plate recognition using generalized character sequence detection." IEEE Access 8 (2020): 35185-35199.

[15] Weihong, Wang, and Tu Jiaoyang. "Research on license plate recognition algorithms based on deep learning in complex environment." IEEE Access 8 (2020): 91661-91675.

[16] Elihos, Alperen, Burak Balci, Bensu Alkan, and Yusuf Artan. "Deep learning based segmentation free license plate recognition using roadway surveillance camera images." arXiv preprint arXiv:1912.02441 (2019).

[17] Arafat, Md Yeasir, Anis Salwa Mohd Khairuddin, and Raveendran Paramesran. "Connected component analysis integrated edge based technique for automatic vehicular license plate recognition framework<? show [AQ ID= Q1]?>." IET Intelligent Transport Systems 14, no. 7 (2020): 712-723.

[18] Zhu, Liping, Shang Wang, Chengyang Li, and Zhongguo Yang. "License plate recognition in urban road based on vehicle tracking and result integration." Journal of Intelligent Systems 29, no. 1 (2019): 1587-1597.

[19] Alam, Nur-A., Mominul Ahsan, Md Abdul Based, and Julfikar Haider. "Intelligent system for vehicles number plate detection and recognition using convolutional neural networks." Technologies 9, no. 1 (2021): 9.

[20] Kilic, Irfan, and Galip Aydin. "Turkish vehicle license plate recognition using deep learning." In 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), pp. 1-5. IEEE, 2018.

[21] Akila, K., B. Sabitha, R. Jayamurugan, M. Teveshvar, and N. Vignesh. "Automated license plate recognition system using computer vision." International Journal of Engineering and Advanced Technology (IJEAT-2019) (2019).

[22] Kashyap, Abhishek, B. Suresh, Anukul Patil, Saksham Sharma, and Ankit Jaiswal. "Automatic number plate recognition." In 2018 international conference on advances in computing, communication control and networking (ICACCCN), pp. 838-843. IEEE, 2018.

[23] Pant, Ashok Kumar, Prashnna Kumar Gyawali, and Shailesh Acharya. "Automatic Nepali number plate recognition with support vector machines." In Proceedings of the 9th international conference on software, knowledge, information management and applications (SKIMA), pp. 92-99. 2015.

[24] Roy, Er Prem Chandra, Arjun Bhandari Thapa, Kumar Shrestha, Prasanna Karmacharya, and Rajan Karna. "Vehicle Number Plate Recognition and Parking System." International Research Journal of Innovations in Engineering and Technology 2, no. 10 (2018): 18.