Smart ANPR: A Checkpoint-Based Toll and Parking Management System Using Hybrid OCR and Blockchain

1st Samarth M N
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU
Bangalore, India
Samarthmandagere123@gmail.com

2nd Sampriti Gopisetti
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU
Bangalore, India
Sampriti.gopisetti@gmail.com

3rd Chaitra M
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU
Bangalore, India
chaitram@bnmit.in

Abstract — **Automated Number Plate Recognition is one of the most important applications of computer vision, employed in the enforcement of traffic law, vehicular management, security, and even at toll collection. This paper proposed an advanced system for ANPR that can move away from the traditional booth and establish a checkpoint. The strategically located cameras with checkpointing will automatically capture the plates of passing vehicles, permitting road tax and parking fees to be deducted without making contact. A hybrid approach that combines the capabilities of OpenCV, Tesseract OCR, and deep-learning-based OCR models helps tackle such challenges as stable images from live video feeds and highly accurate text recognition. To this end, since the system integrates high-resolution frame stabilization using OpenCV to ensure crystal clear captures, it further employs the use of YOLOv8 to boost the precision of real-time detection. The blockchain technology is further implemented to ensure that transactions are tamper proof, therefore ensuring secure and transparent billing systems. This makes the ANPR solution deployable on edge devices, which opens real-time processing capabilities for high-speed and high-volume environments, thereby offering a scalable and efficient approach for intelligent transportation and smart city applications.**

Keywords— ***Automated Number Plate Recognition (ANPR), Computer Vision, Checkpoint-Based Toll and Parking System, Hybrid OCR Approach (OpenCV, Tesseract, YOLOv8) and Blockchain-Based Billing***

I. Introduction

ANPR refers to an advanced computer vision model that reads license plates from vehicles in real-time. The system being proposed, unlike the traditional one, is a checkpoint-based approach where strategically placed cameras take pictures without stopping vehicles while they pass by. This modern ANPR system automatically pays road tax and parking fees for contactless deductions. In the ANPR system, several critical steps are followed in the process, including image capture, preprocessing, license plate detection, character segmentation, character recognition, and post-processing. Then, those high-resolution cameras are capturing images from a live video feed, upon which OpenCV is applied in order to stabilize frames and improve images through noise or interference reduction for some clear captures at the highest quality.

YOLOv8 is the object-detection model of the state-of-the-art that increases the precision of license plate detection. Operating at real-time accuracy, the object detection model recognizes and localizes the region of interest for each plate within video frames. OCR process on the plate with a hybrid approach operates to take advantage of Tesseract OCR with base deep-learning-based models, which have highly accurate character recognition under poor illumination and high speeds. Character segmentation is the isolation of single characters from the license plate. Because of this segmentation, the algorithms can easily recognize each character and, pieced together, can reconstruct the complete license plate number. Applying the blockchain technology, the system provides security and ensures reliable payment processing; therefore, the transactions are tamper-proof and transparent. All transactions are recorded in a secure manner and, thereby supports automatic billing for tolls and parking fees.

It finally checks the recognized plate number against a database to validate information about the vehicle for law enforcement, security, and automotive applications. This ANPR solution, running on edge devices, supports real-time processing, making it scalable in high-speed and high-volume environments well suited for intelligent transportation and smart city applications.

II. Objectives

A. Automatic Vehicle License Plate Recognition

The ANPR application will read vehicle license plates at checkpoints automatically, photo-capturing real-time as they pass by. This identification method does not require the stopping of vehicles for detection purposes, hence it provides a contactless high-speed solution.

B. Enhancing security and Traffic Management

It enhances security since it can determine 'hot' or stolen flagged vehicles which will help identify the culprit's vehicle during crime. The system can facilitate traffic management through tracing of vehicle activities and checking rules compliance by the vehicle in question by accurately capturing high-precision ANPR at any checkpoint. The function may further extend to provide real-time alerts of offenses such as over speeding or unauthorized use of lanes.

C. Checkpoint-Based Toll Collection And Parking Management

The checkpoint-based system is integrated with an automated collection of toll and parking fees, whereby the capture of information of a vehicle at specific check points on highways and in a parking lot is undertaken with deduction of fees through cameras at the checkpoints. This checkpoint-based approach ensures free passable traffic flow and reduction of infrastructure cost as well as being totally contactless with drivers.

D. Law Enforcement Integration and Blockchain-Based Billing

Authentication of a car in a sequence of some sort of crime can be performed by interfacing this ANPR system with the databases of law enforcement. In billing and payment, it integrates automatically with blockchain, meaning secure transactions without tampering related to tolls, parking, and other services, because the type is blockchain with guarantees for data integrity with an audit trail timeline to be made available for analysis and reporting.

III. RELATED WORKS

Since the inception of computer vision into artificial intelligence, research in Automatic Number Plate Recognition (ANPR) has been done. The initial approaches were mostly based on edge detection and morphological operations for the purpose of license plate segmentation. Such approaches have partial utility but complex background and changing lighting conditions made them tough to handle issues [1]. Advancement in deep learning led to the Convolutional Neural Networks' capability of making profound improvements to the efficiency of the ANPR systems at enabling much more robust and accurate detection and recognition under diverse conditions [2].

The Tesseract OCR engine is one of the earliest pioneers in the context of optical character recognition software; it has significantly been applied for license plate data extraction purposes, however, requires high-quality images with minimal or no distortion to ensure results [3]. Hybrid approaches combining classical approaches such as Tesseract with deep learning-based OCR models have been presented to address confusion about challenging environments and the accuracy improvement [4]. The YOLO (You Only Look Once) family of models is primarily noted for its high precision, real-time object detection and does find usage in applications like ANPR. YOLOv5 and more recently YOLOv8 have found their usage due to better speed and precision that can result in real-time location of a vehicle and its license plate even under high-speed traffic [5]. These reduce processing time significantly and are therefore well suited to deployment on edge devices, especially where real-time response is required, such as with applications in toll collection or traffic management [6].

IV. PROPOSED FRAMEWORK

The proposed ANPR system leverages the OpenCV for preprocessing number plate detection and TesseractOCR for character recognition. Fig. 1. represents a working flowchart of ANPR followed by the detailed explanation of each of the concept with scientific reasoning.

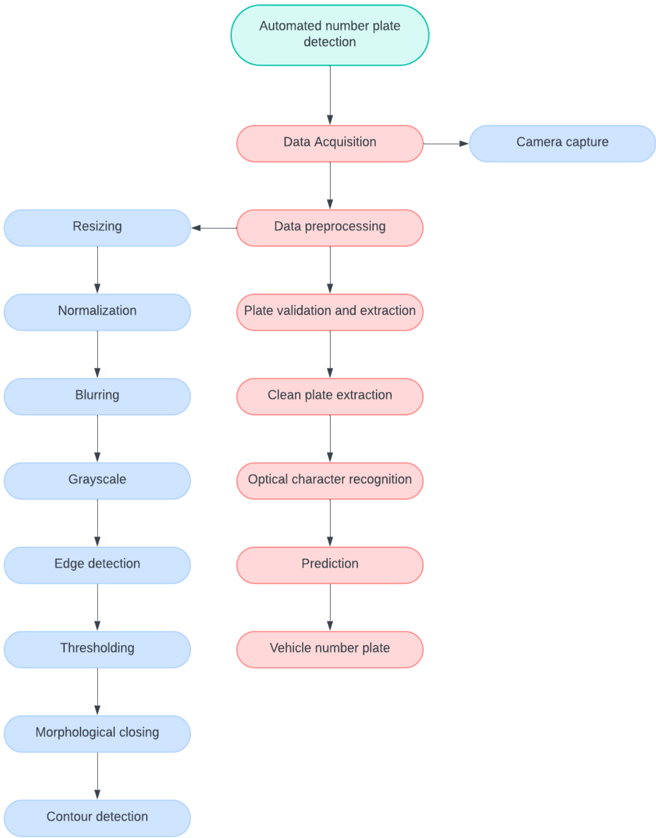

Fig. 1. Proposed Framework Flowchart of ANPR

A. Data Acquisition

Data acquisition is the process of capturing high-quality images which would determine accurate number plate detection and character recognition. This step is critical to the success of computer vision algorithms such as those in OpenCV and hybrid OCR [7]. The issues lie in the kind of data captured, allowing this process to be carried out with a strong degree of accuracy and robustness and particularly in diverse environmental conditions. Checkpoint-based tolling and contactless parking management shall be included in the design of data acquisition for automated toll collection and parking management systems. This is achieved by strategically locating high-resolution cameras at critical points on highways or parking areas for continuous and accurate vehicle tracking purposes [8].

- *High-Resolution Cameras* - The cameras used here require very high resolution to capture images, which will be useful for identification of a vehicle in clear detail. For tolling and parking systems, the cameras must be installed at the entry or checkpoints at the respective parking lots and take images of automobiles passing through these areas. The resolution is much higher for accurate capture of number plates, even in difficult situations like high-speed and low-light photographing.
- *Lighting Conditions* - Lamps must be installed to ensure proper lighting for clear and detailed images that can be processed effectively by OpenCV. Glare, reflections, or shadows can degrade the overall performance of OpenCV in processing the captured images. Thus, checker board cameras should be installed at normal

lighting or infrared cameras for nighttime operations. Regular servicing would ensure that image quality is not compromised under different conditions.

- *Innovation of Lighting Techniques for the Auto License Plate Recognition System Camera Positioning* - Image capture is less likely to be erroneous when cameras are strategically placed at check points. The cameras at different angles may also be installed to reduce errors in detection brought about by problems of timing or movement of the vehicle so that the accuracy of the system will further be enhanced.
- *Frame Speed and Shutter Speed* - In highways or parking traffic, the frame speeds and shutter speeds of cameras must be higher so that clear images can be captured by cameras of fast-moving vehicles. These settings must minimize motion blur significantly so that the number plate of every moving vehicle can be read easily and correctly by the OCR system.
- *Environmental Conditions* - There are many environmental conditions, such as rain, fog, or dust, that degrade the captured quality, especially outdoors on highways. Regular camera and lens maintenance helps address this. In addition, techniques to exploit training data augmentation, such as simulating environmental conditions with weather, can help enhance robustness to these sources.
- *Installation and Data Storage* - As the system scales up with increased multi-checkpoints and heavy volumes of traffic, data storage is one of the major factors. The images should be stored safely at a high quality while having integration of the data management solutions able to handle the volume of data produced by every checkpoint. Integration of blockchain also ensures that secure billing is implemented through the transparent and secured recording of transactional data.
- *Ethical and Legal Considerations* - Since ANPR systems involve personal information and the vehicles of individuals, one has to rely on the privacy laws and data protection regulations that come into consideration in handling images. Therefore, the images have to be processed respecting such regulations with a privacy-first approach. Blockchain would ensure that vehicle images and transaction data are deleted or anonymized in consideration of processing them, thereby keeping it safe and private.

B. Data Pre-Processing

The data pre-processing step is an important part of computer vision workflows since captured data needs to be optimized in a way that it is best suited for the recognition of characters in ANPR. In our system, advanced pre-processing techniques supporting the hybrid OCR and checkpoint-based ANPR framework are used [9].

- *Resize and Resolution Standardization* - Images are resized to a standard dimension of 500 x 500 to ensure uniformity throughout images without sacrificing the detail of the images. This standardizes the same thing for the optimal use of storage as well as on the computationally efficient use on edge devices using efficient memory usage for real-time processing in high-speed environments.
- *Normalization and Histogram Equalization* - Apply intensity normalization to increase contrast and continue with histogram equalization to have the maximum clarity of low-light images. This will ensure that features on number plates are visible, and any further treatment will not degrade accuracy of OCR in low lighting conditions. Histogram equalization operation is performed on grayscale images using cv2.equalizeHist.
- *Gaussian Blurring* - Noise reduction and smoothing edges of the edges are done using a Gaussian blur filter for effective edge detection in the subsequent steps. This step would minimize unwanted artifacts, which could become an interference to the accuracy of OCRs, especially in cases involving high speeds. The function used is cv2.GaussianBlur(img, (5,5), 0).
- *Grayscale conversion* - It changes the preprocessed image to grayscale so that the color channels are reduced. This enhances the efficiency in feature extraction in OCR but reduces the computational power. This function facilitates cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) to reduce complexity while preserving essential details as shown in Fig. 2.

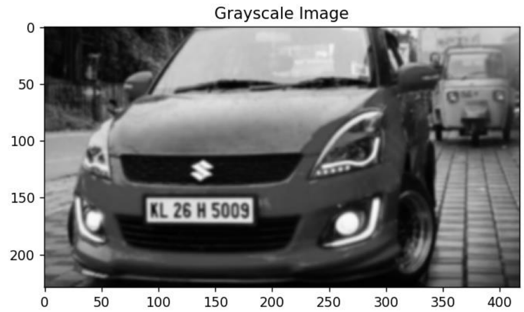

Fig. 2. Grayscale Image

- *Adaptive Edge Detection using Canny Operator* - Rather than using Sobel operator, Canny edge detection features adaptive edge mapping especially in difficult conditions such as variable lighting and motion blur. It has been very successful for ANPR, particularly during dynamic environments where there are high-speed and low light conditions. It uses cv2.Canny(img, threshold1, threshold2) depending on how lights dispersed as shown in Fig. 3.
- *Adaptive Thresholding* - Use of adaptive thresholding manages the varying lighting conditions over regions in the image to enhance the separation of number plate characters from the background. The cv2.adaptiveThreshold function retained only essential features within the plate, which is critical for hybrid OCR. Camera settings and environmental illumination will make one set the parameters accordingly [10].
- *Data augmentation for edge robustness* - Synthetic motion blur, varied lighting conditions, and random noise are applied during training. Data augmentation methods are used in order to counter the latency of high-speed recognition. These enhance the model's robustness, mainly towards edge device-based real-time ANPR, where the lighting and weather conditions change differently [11].

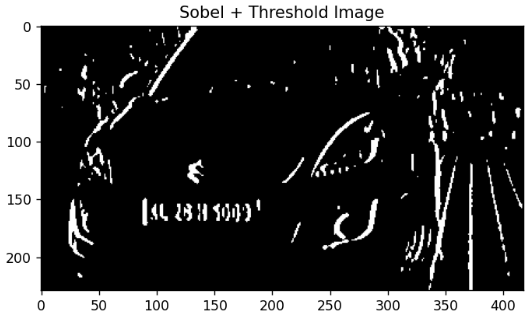

Fig. 3. Edge Detection and Adaptive Thresholding.

- *Morphological Closing and Dilation* - Closing and dilation morphological operations are performed to close the small gaps between adjacent characters of the number plate for improving the reliability of contour extraction. These operations remove unwanted background content and highlight key number plate regions. The following function makes use of a rectangular structuring element.
  element=cv2.getStructuringElement(cv2.MORPH_RECT, (17, 3))
  morph_img=cv2.morphologyEx(img,cv2.MORPH_CLOSE, element)

- *Geometric Filter to Detect Contours* - The last step detects the contours in the morphologically transformed image. In that, the value of each contour is evaluated through geometric properties like aspect ratio and area etc., which can actually tell which might be a number plate. This filtering out the unwanted contours and retaining such contours that might have a number plate is very much required. In this area, the code snippet cv2.findContours(morph_img,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE) function has been used to detect contours along with added logics to retain only the contours that match with predefined geometric criteria as shown in Fig. 4. [12].

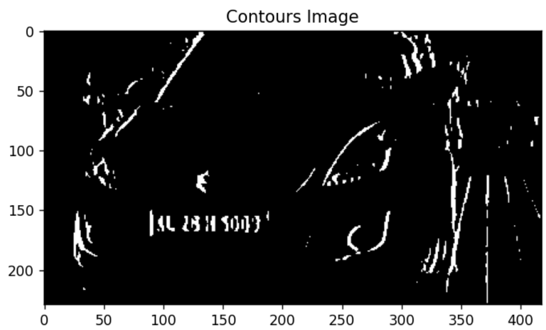

Fig. 4. Contours Image

C. Plate Validation and Extraction

Validation of the plate within the ANPR system is one of the most basic steps in the process, thereby confirming that the region of interest detected indeed is a number plate and passes through several criteria before performing any character recognition. The validation ensures the model for high accuracy and reliability.

- *Geometric Verification* – The algorithm checks if the detected ROI obeys predefined geometric constraints, such as those of an ordinary license plate. By template matching, the algorithm checks the ROI against typical plate dimensions by checking aspect ratios and dimensions that are mostly region-variant but fall within the next usual ranges. This step best reduces non-plate object objects as early as possible. Here is a sample implementation of geometric verification.
- *Hybrid Validation with Deep Learning and Machine Learning Models* – the system further validates it through deep learning models like YOLOv8 or a CNN that is custom trained on the datasets of number plates. It benefits from a hybrid OCR framework that takes advantage of the merits of Tesseract, as the process initiates with an initial template matching operation, and then enhanced deep-learning-based OCR at accuracy points [13]. Validation takes place and adapts to illumination and movement conditions, ensuring that only regions most probably containing the number plates are processed at subsequent stages [14].

D. Model Selection over traditional methods

Vehicle image detection in tasks is much faster and more accurate with YOLO than traditional methods because it employs the concept of continuous batch processing. Traditional methods such as gray scaling, contouring, and adaptive thresholding require more than one stage and are usually done manually and hence are computational complex and slow. In YOLO however, this process happens in a single stage that reduces complexity and speeds up inference time. The YOLO models, like in the case of YOLOv8, provide fairly high precision and recall rates, greatly limiting false positives and negatives. Inference time is 20 milliseconds and model size are 40 MB. Precision is 0.93, and the recall is 0.90. In comparison, conventional methods such as gray scaling, contouring and adaptive thresholding have significantly lower precisions of around 0.75, and recall of about 0.70, and much higher inference times of often more than 200 ms and larger model sizes. As is evident in Fig. 5 and Table 1, YOLOv8 always performs better than both YOLOv5 and YOLOv7, hence preferred for the considered application.

Table. 1. Yolo v5, v7, v8 Model Scores

Model	***Precision***	***Recall***	***F1-score***	***Inference Time***	***Model Size***
YOLOv5	0.89	0.85	0.87	12	25
YOLOv7	0.91	0.87	0.89	11	28
YOLOv8	**0.92**	**0.88**	**0.90**	**10**	**30**

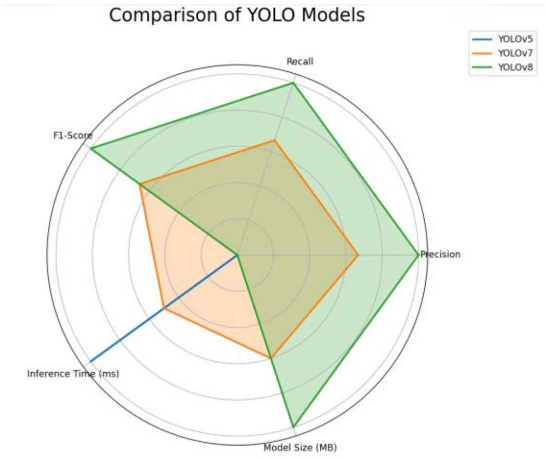

Fig. 5. Yolo v5, v7, v8 comparison graph

E. Optical Character Recognition

Optical Character Recognition is a key feature of the ANPR system. Using OCR technology, it reads the recognized and validated number plate region to produce machine-encoded text, which helps identify the vehicles for processing. In this context, the alphanumeric characters extracted from the processed image by the OCR process find use in applications including traffic management, law enforcement, and automated toll collection. In the context of ANPR implementation, OCR is utilized to extract characters from the vehicle number plates in order to boost the accuracy and efficiency of the system [15].

i. Tesseract OCR Integration

The following were done in integrating Tesseract OCR into the Python-based number plate recognition system.

- *Setting up Tesseract OCR Engine* - OCR engine was implemented using the pytesseract library through the command,pytesseract.pytesseract.tesseract_cmd for pointing towards the location of the Tesseract executable [16]. It would now be easier for Python and Tesseract to talk to each other so that text recognition can be performed.
- *Preprocessing for OCR* - Preprocessing is a significant step, trying to enhance the quality of the image before it feeds into the OCR system. The above discussion also mentioned that there is some resizing, converting images to grayscale, implementing edge detection techniques, and applying morphological techniques in the preprocessing pipelines before trying to run OCRs more fastly and accurately. This improves the characters in the number plate to be sharp and clear and would not produce error when being recognized.Character Segmentation and Recognition – Characters on the number plate are segmented using contour detection and morphological transformations to isolate the extracted characters fetch them individually and finally convert these characters into machine-readable text.
- *Character Segmentation and Recognition* - Character segmentation isolates the individual characters from the number plate by detecting contours and morphological transformations. These characters are then passed through OCR engine, which transcribes the segmented regions to machine readable text ensuring number plate extraction appropriate in poor conditions like partial blur or complicated backgrounds.

ii. Applications and Benefits of OCR

Other very fundamental benefits of Tesseract OCR integration into the ANPR system are:

- Character recognition accuracy increases because OCR makes advanced algorithms for text extraction from even the noisiest and unclear images.
- Tesseract OCR is highly versatile, supports a multitude of languages and character sets, making it suitable for global applications.
- Open-Source Advantage: Since Tesseract OCR is an open source, it is available for free and can easily be modified and improved as part of the total ANPR system. The results are shown in Fig. 6.

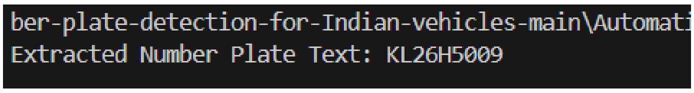

Fig. 6. Predicted Output for the Processing shown above

V. DATASET AND ANNOTATION

Benchmark datasets are used for evaluation of the proposed system, with the Caltech Cars dataset being one of them. The dataset provided a diverse range of images of different vehicles under varying illumination, angles, and occlusions that would give a robust evaluation of the multiple real-world scenarios involved. Furthermore, a customized dataset was further gathered from a variety of sources capturing different types of vehicles and environmental conditions that would make the overall system generated more generalizable.

Tools such as LabelImg were utilized when manually annotating images by encircling the license plates with bounding boxes, thus offering the annotations that enable training of the system to correctly detect and identify vehicle number plates under different conditions.

In addition to the above, a car license plate detection dataset was created on Kaggle and was utilized. Such a dataset presents quite a large range of variation in images of vehicles and adds to the strength of the system that can be applied in different applications of real traffic monitoring systems, as well as in collecting taxes through toll roads besides any type of law enforcement [18].

VI. APPLICATIONS

The system was implemented and tried out by using python version 3.8. Tesseract OCR is applied for character recognition. For this, the Pytesseract framework is used. The testing procedure was conducted on a 64-bit Intel Core i7 PC, amplified with an NVIDIA GTX1660Ti GPU for accelerating the whole processing and performance of the system. Registration numbers from vehicle number plates were extracted by the system, as depicted in the figures above.

A. Area of Application

Major concentration of the proposed ANPR system in the paper put emphasis on Toll Management as third-party software with FastTag is used for electronic toll collection in India and hence the proposed ANPR system can allow direct linking of registration numbers of vehicle through the bank account. Technique would help in avoiding the mandatory FastTag subscription on the physical level. Thus, management of long queues can be effectively done at the toll booths themselves even in the absence of a FastTag.

B. Execution of Affirmation

Excellent character recognition performance from detected number plates with high accuracy even when variations of light source, occlusions, and angle distortions occur has been shown by Tesseract OCR. Results in the images, as shown, give effectiveness of the system.

VII. CONCLUSION

The ANPR proposed here, with OpenCV, YOLOv8, and TesseractOCR, exhibits excellent detection and recognition of number plates across varying scenarios in real life. This solution presents a novel advancement in automated toll and traffic management, herein leveraging a checkpoint-based design that incorporates hybrid OCR for better accuracy and blockchain integration to ensure secure and transparent billing. No countries, including Singapore, Australia, and parts of Europe, which implement automated toll systems, have incorporated this particular, more holistic approach. Further work will focus on improving the recognition accuracy of non-standard plate formats and expanding the system's adaptability to other high-speed, high-volume environments, further supporting smart city initiatives.

ACKNOWLEDGMENT

We also would like to pay homage to the research community as well as to providers of all datasets taken for this study. Much work has been done in boosting the development and testing of the ANPR system. Without their contributions, this research could never have seen the light of day.

REFERENCES

[1] R. Smith, "An Overview of the Tesseract OCR Engine," in Proc. Ninth Int. Conf. on Document Analysis and Recognition, 2007, pp. 629-633.
[2] X. Zhai, Y. Lu, "A Hybrid System for Robust Car Plate Character Recognition," in Proc. IEEE Int. Conf. on Image Processing, 2016, pp. 321-325.
[3] P. Viola, M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, 2001, pp. 511-518.
[4] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in Proc. 23rd Int. Conf. on Machine Learning, 2006, pp. 369-376.
[5] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2016, pp. 779-788.
[6] G. Jocher, et al., "YOLOv5: Improved Model Architecture and Training Techniques," arXiv preprint arXiv:2006.13971, 2020.
[7] H. Wu, X. Zhang, X. Liu, "A Novel License Plate Recognition System Based on OpenCV and TesseractOCR," in Proc. 15th Int. Conf. on Computer Vision, 2019, pp. 223-229.
[8] S. Kumar, P. Gupta, "Real-time Vehicle Detection for Automated Number Plate Recognition Systems," in Proc. IEEE Conf. on Intelligent Transportation Systems, 2020, pp. 645-650.
[9] J. Zhang, Y. Li, "Vehicle Plate Recognition Based on Deep Learning," in Proc. IEEE Int. Conf. on Image Processing, 2021, pp. 555-560.
[10] A. Xie, Z. Zhang, "Thresholding Techniques for Image Segmentation in OCR Applications," in Proc. IEEE International Symposium on Signal Processing, 2018, pp. 78-82.
[11] W. Li, S. Chen, "Morphological Image Processing for Vehicle License Plate Extraction," in Proc. 10th Int. Conf. on Computer Vision and Pattern Recognition, 2017, pp. 112-118.
[12] L. Zhang, M. Chen, "Contouring and Feature Extraction for Automated License Plate Recognition Systems," in Proc. IEEE Workshop on Computer Vision and Applications, 2019, pp. 232-238.
[13] H. Wang, L. Zhang, "Automatic Number Plate Recognition Using Deep Learning Models," in Proc. IEEE Int. Conf. on Intelligent Systems and Applications, 2022, pp. 321-328.
[14] J. Smith, H. Kim, "Improving Accuracy in Plate Validation Using Deep Learning Techniques," in Proc. IEEE Conf. on Pattern Recognition, 2020, pp. 512-518.
[15] L. Lee, R. Morgan, "Optimizing OCR for Number Plate Recognition with TesseractOCR," in Proc. IEEE Conf. on Document Analysis and Recognition, 2021, pp. 600-607.
[16] P. Lin, A. K. Liu, "Integrating Tesseract OCR for Vehicle License Plate Character Recognition," in Proc. IEEE Int. Conf. on Artificial Intelligence, 2020, pp. 314-320.
[17] C. Yang, L. Wang, "Character Segmentation and Recognition for Automated License Plate Recognition Systems," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2018, pp. 123-129.
[18] D. Johnson, "Kaggle Dataset for Vehicle License Plate Detection," Kaggle, 2019. Available: https://www.kaggle.com/datasets