

A Live-Video Automatic Number Plate Recognition (ANPR) System Using Convolutional Neural Network (CNN) with Data Labelling on an Android Smartphone

Abd Gani, S. F.¹, Miskon, M. F.², Hamzah, R. A.³, Mohamood, N.⁴, Manap, Z.⁵, Zulkifli, M. F.⁶,
Md Ali Shah, M. A. S.⁷

^{1,3,4,5,6}Fakulti Teknologi Kejuruteraan Elektrik dan Elektronik, Universiti Teknikal Malaysia Melaka, Malaysia

²Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka, Malaysia

⁷MASH Construction Engineering Services Sdn. Bhd., Malaysia

Abstract—Automatic Number Plate Recognition (ANPR) combines electronic hardware and complex computer vision software algorithms to recognize the characters on vehicle license plate numbers. Many researchers have proposed and implemented ANPR for various applications such as law enforcement and security, access control, border access, tracking stolen vehicles, tracking traffic violations, and parking management system. This paper discusses a live-video ANPR system using CNN developed on an Android smartphone embedded with a camera with limited resolution and limited processing power based on Malaysian license plate standards. In terms of system performance, in an ideal outdoor environment with good lighting and direct or slightly skewed camera angle, the recognition works perfectly with a computational time of 0.635 seconds. However, this performance is affected by poor lighting, extremely skewed angle of license plates, and fast vehicle movement.

Keywords—computer vision, image processing, automatic number plate recognition, vehicle identification.

I. INTRODUCTION

Automatic Number Plate Recognition (ANPR) and its various complex algorithms are actively being researched especially alongside the rising interests in autonomous driving and intelligent transportation systems, where the need for human intervention is minimized by the system to determine vehicle identity [1]. ANPR system combines computer hardware and software algorithms in computer vision to recognize a license plate number which can then be verified against a database of registered users and confirm the identity of the vehicle owner. Image capture, image pre-processing, extracting the region of interest (ROI), and translating the pixels into mathematically comprehensible letters utilizing Optical Character Recognition (OCR) are usually the common steps involved [2].

With the advent of digital camera technology and improved computational processing within handphones and microcomputers, ANPRs are more portable and handheld in recent years [3]–[5].

One of the applications that utilizes ANPR is campus security gate control which automatically monitors and controls vehicle entry into university compound, and this article focuses the development of the ANPR within this scope. In auto-gate systems, the ANPR identifies the vehicle license plate number and compares it against a database of registered campus members, and the respective information such as the name of the vehicle owner, student or staff identification number, faculty name and telephone number can be retrieved and displayed automatically. This can be used as a cheaper access control method instead of using access cards or checking campus stickers which is costly, time consuming, and may lead to traffic congestion especially during peak hours [6]. With the aid of Google Cloud Vision, an API that uses machine learning models trained on a huge dataset of pictures that is accessible via API queries, licence plate numbers identified by the system may be captured and logged for future retrieval if the need arises. This API's engine intelligently classifies pictures, identifies objects, and recognises printed text inside them [7].

Other than government facilities, private sectors and industries are also implementing ANPR solutions to address their security concerns, for examples in resorts, museums, hospitals, parking services, and casinos. Because private facilities generally do not have access to government or police databases to check licence plates, they must create and match their own labelling databases to identify their clients, VIPs, key staff, or those on the blacklist [8].

Beyond providing perimeter protection, ANPR can also be utilised to improve service applications such as parking valet or regular customer versus VIP recognition, logistics and key staff tracking, sales and advertising, and parking management.

Although various research on ANPR has been done, not many focuses on implementing the ANPR on a smartphone, demonstrating the portability and advances in computing power of a mobile device. In this paper, the ANPR system discussed is developed purely on an HTC U19e Android smartphone which was introduced in 2019 running on a Snapdragon 710 Chipset, and Adreno616 GPU. The development is done using Android Studio IDE on an Apple MacBook Pro. The following sections will discuss the design and implementations in detail.

II. LITERATURE REVIEW

An ANPR system is typically comprised of the subsequent 5 stages [9] as depicted in Figure 1.

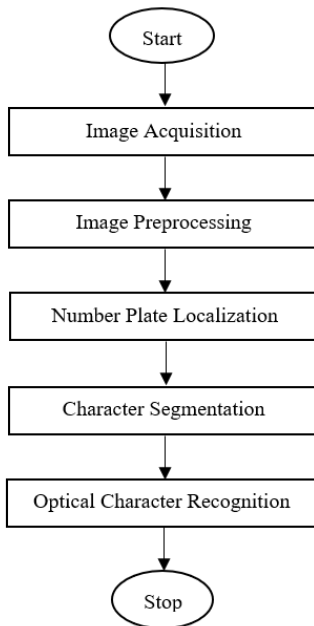


Figure 1 ANPR Flowchart.

First, an image of a vehicle of interest is captured using a camera with good resolution, alignment, shutter speed, and lighting condition. The image is then improved by applying image pre-processing techniques such as normalisation, brightness and contrast adjustments, and skewness correction.

Number plate localization is the process of extracting the licence plate from a vehicle image using some features such as the boundary, colour, or presence of characters. The individual characters on the licence plate are then located and identified during character segmentation stage. The last stage is optical character recognition, which uses pattern matching or classifiers such artificial neural networks (ANN), Hidden Markov Model (HMM), fuzzy logic, and template matching to recognise segmented characters [5]. The final output, which is decipherable text, may then be linked to other apps or databases.

ANPR detects, normalises, and improves a licence plate picture using a series of image processing algorithms, as well as optical character recognition (OCR) to recognise the alphanumeric of the licence plate. Some of the popular methods are Hidden Markov Models (HMM), Template Matching, Neural Networks, K-Nearest Neighbour (KNN), fuzzy logic, and Support Vector Machine (SVM).

A. Template Matching

The simplest technique for character recognition is called Template Matching, which is a cross-correlation method in which the extracted character's resemblance to a collection of template characters is compared or assessed. The character with the most similarities to one of the candidates in the template set is picked. The resultant image's grey level intensities are noticeably affected by changes in lighting conditions, which is why these approaches are often employed on binary pictures [1]. Khalil [10] used Template Matching and test his system on license plates in Saudi Arabia and Egypt. The system doesn't utilise any segmentation to skew or de-skew the input picture for improved character identification, and works right after the licence plate extraction phase, which is known as the Information Recognition Phase (IRP). The IRP number plates are divided into two sections: the upper portion is the county name, and the lower portion is the identifying numbers. Together with Template Matching the author also used Moving Window to find specified object within the image. Template Matching technique is also discussed in [11]–[13].

B. Sliding Concentration Window and Artificial Neural Network

Sliding Concentric Window (SCW), a novel segmentation approach, was presented to discover an effective licence plate candidate region by Kaushik [14].

The study also recommended utilising a method called Least Square Fitting with Perpendicular Offsets (LSFPO) to deal with picture tilt correction. The SCW retrieves the licence plate from the pictures by locating the vehicle region's vertical and horizontal boundaries. The geometric parameters of the licence plate, such as aspect ratio, licence plate number area, and bounding box, are computed to improve extraction efficiency. False positives from the list of candidates are eliminated by utilising these parameters. LSFPO is applied to rectify the tilted image in successful candidates. A parameter to associate with the reaction data and a prognosticator data factor are required by the LSFPO fitting technique. The coefficient is a quantity that can be calculated by minimising the sum of squared residuals. The resulting positive contender is then supplied into the Artificial Neural Network (ANN) for recognition.

C. Support Vector Machine

Support Vector Machines (SVM) can also be used for character recognition, and it is a supervised learning method commonly used for regression and classification. Each successful candidate's recognition will be handled by SVM, which will test against a library of trained licence plate number models. The accuracy is calculated by classifying the SVM outputs based on the maximum value. SVM, however, must be trained using valid number plate sample database prior to use. The 2 types of SVM classifier are One Against All (OVA) and One Against One (OVO). This method produces an accuracy of 89.77%, 82.86%, and 65.22% for 1 meter, 3 meters, and 5 meters distance of capture, respectively [15].

D. Convolutional Neural Network

A Convolutional Neural Network (CNN) based object detector called You Only Look Once (YOLO) is developed in [16]. It is a 2-step process that uses simple data enhancement techniques like reversed letters and inverted licence plate numbers. The CNNs are tweaked and trained at each phase. The resultant model produces good results for two datasets that are distant. The first database; SSIG-SegPlate Database by Smart Surveillance Interest Group (SSIG), contains 2000 frames from nearly 101 vehicle videos. The system attains 93.53% recognition precision at 47 FPS, performing better than both the OpenALPR commercial systems and Sighthound (with corresponding recognition precision of 93.03% and 89.80%), and significantly better results than the past techniques, which attain an accuracy of 81.80% [1].

III. THE PROPOSED METHOD

A. The Hardware

A PC or laptop is needed to design and develop the ANPR, and for this project, a 13.3-inch 2015 MacBook Pro with 8GB of LPDDR3 1866MHz RAM and 2.7GHz dual-core Intel Core i5 processor is used.

A smartphone is needed to deploy, run, and test the app. A 2019 HTC U19e Android smartphone running on Snapdragon 710 Chipset, and Adreno616 GPU is used for this purpose. The app is called "ANPR CNN" inside the Android environment and during the development phase, the app is constantly refined and updated by tethering it to the MacBook Pro.

B. The Software

Android Studio IDE is the development software used to setup the appropriate environment for app development, as displayed in Figure 2. The initial setup for Android Studio is done by first configuring the necessary development environments such as the Android API, Android SDK, and Android Virtual Device (AVD). A specific target device or smartphone is then configured, and the developer can then start to write the code and add assets such as images, icons and sounds to the app. To test the app, the developer can either run the app from the actual smartphone, or alternatively run the app using the built-in emulator device which is known as the AVD. Advanced developers can utilize features such as application profiling for tweaking better performance and debug the application in real-time. Once development is done, the app can be published by packaging it into the Android Package (APK) format.

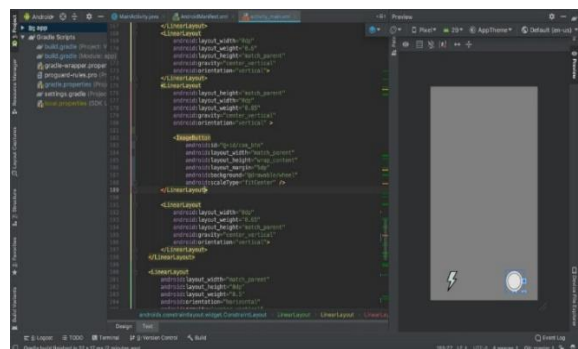


Figure 2 Android Studio IDE.

Image classification and regression work much like classification and regression on standard structured data, but the neural network is built using elements of CNN, like convolutional layers, max pooling, and flattening. All parameters are learned. Neural Architecture Search (NAS) is proven to be better than human-designed CNN in accuracy and latency, and it is used to mechanise HW-CNN codesign by incorporating factors from both the CNN model and the HW throttle, and then searching for the optimum model-accelerator pair that improves correctness and effectiveness [17]. AutoML deep learning system based on NAS is incorporated into the app. Dataset are then presented and fed into AutoML and the trained weight is then generated. As a result, developers can comprehend the information in an image by condensing sophisticated pre-trained machine learning models inside a user-friendly vision client library. It can rapidly classify a picture into hundreds of predetermined categories and recognise specific objects or faces within photos, as well as read printed text.

Figure 3 shows the system flow of the text recognition handling done inside the ANPR CNN mobile app. The device's camera sensor is initialised by CameraSourcePreview class, and it will start to get screening frames from the image sensor. Text that are spotted within the preview frames will be isolated using the label boxes produced by GraphicsOverlay class. The class responsible for doing the text recognition is TextRecognitionProcessor hence the resulting preview frames captured is fed into class for further processing.

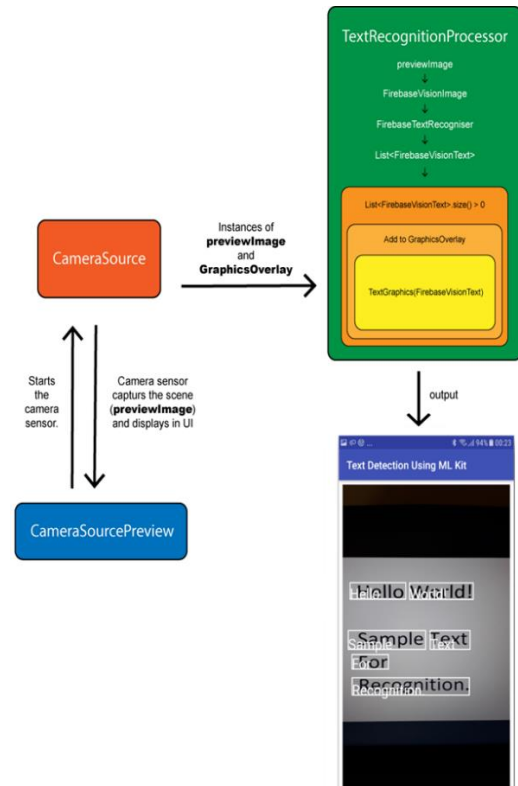


Figure 3 ANPR CNN Text Recognition Workflow.

Figure 4 shows the Firebase Realtime Database containing the verified license plate numbers, and once a new license plate number text is recognized, it will be authenticated against this database to determine its legitimacy. Entries in this database is using JavaScript Object Notation (JSON) system and each item will have its own metadata such as last detected timestamp, plate number, student name and id, and faculty name.

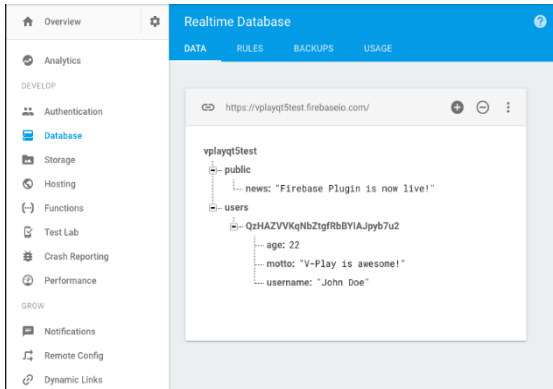


Figure 4 Firebase Realtime Database.

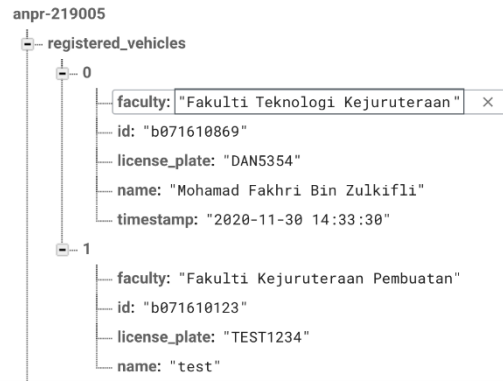


Figure 5 JSON Data Entry in Firebase.

Firestore Database is a realtime cross-platform database of choice, and since it is cloud-hosted, data can be synchronized seamlessly and queried easily using NoSQL. For data storage, it is arranged using a key-value JSON tree with numerous nodes format. This real-time database, on the other hand, should never be used to replace a Relational Database Management System (RDBMS), as it lacks tables and columns, making complicated search queries more difficult although it is seemingly adequate for the purpose of this project. Before query, the JSON data must first be retrieved and flattened. An instance of entries with the distinctive keys produced by the SDK when composing new data to the database is presented in Figure 5. The objective was to replace old data with fresh data using the exact same keys. When a user does a search query, they must query all the unique keys that have been saved, denormalize them, and flatten the JSON accordingly. The data can be saved locally when offline to keep real-time events responsive, and then synchronised with the remote database afterwards. The JSON format allows users to create their own database structures. It employs the WebSocket protocol, which responds to data changes within milliseconds, rather than a stateless network protocol like Hypertext Preprocessor Protocol (HTTP).

IV. RESULTS AND DISCUSSION

Figure 6 shows the ANPR CNN app splash screen. The app's main window is designed to be minimalistic and user-friendly by providing simple and clear user interface. Once the app finishes loading, the splash screen disappears, and the live-video scanning frame is activated as illustrated in Figure 7.

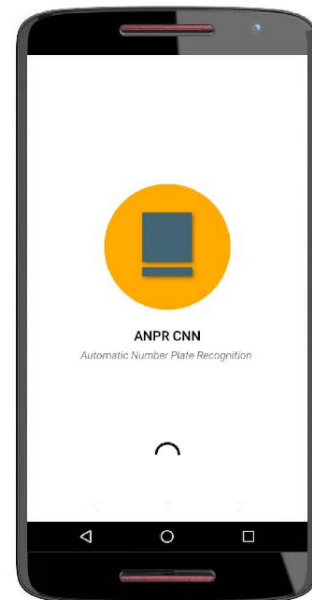


Figure 6 ANPR CNN App Splash Screen.

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 10, October 2021)

There is only 2 control available in the app main window, which is the lightning icon on the bottom left of the screen, and the snapshot icon on the bottom right of the screen. The lightning icon enables additional illumination in the form of an LED flashlight in case the surrounding is dark. The snapshot icon takes a snapshot of the current frame and saves it into the photo gallery.

Note that the user does not need to snap a still image in order to recognize license plate numbers. The ANPR CNN app will examine the live picture frame by frame in realtime. The user only needs to direct the camera in the direction of potential vehicles. The live-video scanning frame is always on, and recognition is constantly done to find any potential license plate numbers in the camera feed. Once a license plate number is found, a green box will appear along with the deciphered text in red colour on the screen. If the found license plate number exists in the Firebase database, additional owner data label pops up on the screen using Android Toast Message, showing vehicle owner's details such as name, faculty, student or staff id, and last detected time. If the found license plate number does not exist in the Firebase database, it means that the user is not registered, hence no further message is displayed.



Figure 7 ANPR CNN main app window. Note that the recognition immediately starts once the live-video scanning frame is loaded, and in the event a registered license plate number is recognized, the detailed info label pops up.

In Figure 8 an outdoor test run around the university campus is conducted and the output is observed. Since the camera is directed straight at the license plate numbers and with proper lighting during the day, the system performs the recognition accurately and correctly. A built-in timer java.util.Timer is used to get the time difference of the recognition process and it is found that in an ideal angle and lighting it takes an average of 0.635 seconds to complete the task.



Figure 8 Recognition of license plate numbers that are registered in the database (data label pops up displaying owner identity and details).

In Figure 9 the ANPR CNN app was taken for a drive test around the city of Melaka and the output is observed. The recognition works perfectly given that the video manages to capture a well-lit and direct angle or slightly skewed image. Images that are extremely skewed, however, takes a long time to process, and even come out with the wrong results especially when observing character pairs like (O,0), (P,B), (Z,2), (S,5), (B,8), (D,O), (1,I), etc.

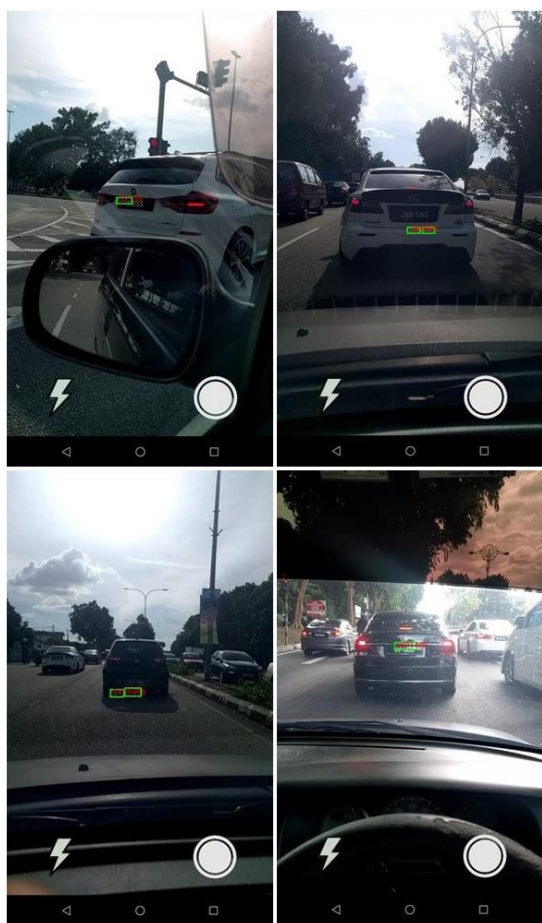


Figure 9 Recognition of license plate numbers that are not registered in the database (no additional data is displayed).

V. CONCLUSION

This paper discusses an ANPR system developed on an Android smartphone based on Malaysian license plate standards.

In an ideal outdoor environment with good lighting and direct or slightly skewed camera angle, the recognition works perfectly with a computational time of 0.635 seconds. This performance is however affected by poor lighting, extremely skewed angle of license plates, fast vehicle movement causing mismatches in the recognized output such as (O,0), (P,B), (Z,2), (S,5), (B,8), (D,O), (1,I), etc which can deceive the OCR mechanism. It is recommended these factors must be tested and addressed into the system to ensure the accuracy of the suggested algorithms in the future.

Acknowledgements

The authors would like to thank Universiti Teknikal Malaysia Melaka (UTeM) and MASH Construction Engineering Services Sdn. Bhd. for providing the support needed to complete the work herein via grant number MTUNC/2019/FTKKE-CACT/MC0016.

REFERENCES

- [1] Lubna, N. Mufti, and S. A. A. Shah, "Automatic Number Plate Recognition: A Detailed Survey of Relevant Algorithms," *Sensors* 2021, Vol. 21, Page 3028, vol. 21, no. 9, p. 3028, Apr. 2021, doi: 10.3390/S21093028.
- [2] N. L. Yaacob, A. A. Alkahtani, F. M. Noman, A. W. M. Zuhdi, and D. Habeeb, "License plate recognition for campus auto-gate system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 128–136, Jan. 2021, Accessed: Jul. 07, 2021. [Online]. Available: <http://ijeecs.iaescore.com/index.php/IJEECS/article/view/21997>
- [3] A. G. S. Fakhar, H. M. Saad, K. A. Fauzan, H. R. Affendi, and A. M. Aidil, "Development of portable automatic number plate recognition (ANPR) system on Raspberry Pi," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 3, 2019, doi: 10.11591/ijece.v9i3.pp1805-1813.
- [4] M. Y. Aalsalem and W. Z. Khan, "CampusSense - A smart vehicle parking monitoring and management system using ANPR cameras and android phones," in *International Conference on Advanced Communication Technology, ICACT*, Mar. 2017, pp. 809–815. doi: 10.23919/ICACT.2017.7890230.
- [5] T. S. Gunawan, A. Mutholib, and M. Kartiwi, "Design of Automatic Number Plate Recognition on Android Smartphone Platform," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 5, no. 1, pp. 99–108, Jan. 2017, doi: 10.11591/ijeecs.v5.i1.pp.
- [6] F. C. Chazanga, J. Phiri, and S. Namukolo, "Development of a Two Factor Authentication for Vehicle Parking Space Control based on Automatic Number Plate Recognition and Radio Frequency Identification," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 1, pp. 588–597, 2019, doi: 10.14569/IJACSA.2019.0100174.

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 10, October 2021)

- [7] K. Naik, N. Sawant, G. Kamat, S. Kandolkar, and N. Marchon, "IRIS: An Application for the Visually Impaired Using Google Cloud API," *Lecture Notes in Electrical Engineering*, vol. 703, pp. 29–43, 2021, doi: 10.1007/978-981-15-8391-9_3.
- [8] K. Alexander, A. Wicaksana, and N. M. S. Iswari, "Labeling Algorithm and Fully Connected Neural Network for Automated Number Plate Recognition System," *Studies in Computational Intelligence*, vol. 847, pp. 129–145, May 2019, doi: 10.1007/978-3-030-25217-5_10.
- [9] A. Roy and D. P. Ghoshal, "Number plate recognition for use in different countries using an improved segmentation," *Proceedings - 2011 2nd National Conference on Emerging Trends and Applications in Computer Science, NCETACS-2011*, pp. 179–183, 2011, doi: 10.1109/NCETACS.2011.5751407.
- [10] M. I. Khalil, "1Car Plate Recognition Using the Template Matching Method," *International Journal of Computer Theory and Engineering*, pp. 683–687, 2010, doi: 10.7763/IJCTE.2010.V2.224.
- [11] I. S. Ahmad, B. Boufama, P. Habashi, W. Anderson, and T. Elamsy, "Automatic license plate recognition: A comparative study," *2015 IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2015*, pp. 635–640, Jan. 2016, doi: 10.1109/ISSPIT.2015.7394415.
- [12] K. Kanayama, Y. Fujikawa, K. Fujimoto, and M. Horino, "Development of vehicle-license number recognition system using real-time image processing and its application to travel-time measurement," *IEEE Vehicular Technology Conference*, pp. 798–804, May 1991, doi: 10.1109/VETEC.1991.140605.
- [13] O. Hommos, A. Al-Qahtani, A. F. A. Al-Zawqari, F. Bensaali, A. Amira, and X. Zhai, "HD Qatari ANPR system," *2016 International Conference on Industrial Informatics and Computer Systems, CIICS 2016*, Apr. 2016, doi: 10.1109/ICCSII.2016.7462420.
- [14] K. Deb, I. Khan, A. Saha, and K.-H. Jo, "An Efficient Method of Vehicle License Plate Recognition Based on Sliding Concentric Windows and Artificial Neural Network," *Procedia Technology*, vol. 4, pp. 812–819, Jan. 2012, doi: 10.1016/J.PROTCY.2012.05.133.
- [15] Noprianto, S. Wibirama, and H. A. Nugroho, "Long distance Automatic Number Plate Recognition under perspective distortion using zonal density and Support Vector Machine," *Proceeding - 2017 3rd International Conference on Science and Technology-Computer, ICST 2017*, pp. 159–164, Aug. 2017, doi: 10.1109/ICSTC.2017.8011871.
- [16] R. Laroca et al., "A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector," *Proceedings of the International Joint Conference on Neural Networks*, vol. 2018-July, Oct. 2018, doi: 10.1109/IJCNN.2018.8489629.
- [17] M. S. Abdelfattah, Ł. Dudziak, T. Chau, R. Lee, H. Kim, and N. D. Lane, "Best of Both Worlds: AutoML Codesign of a CNN and its Hardware Accelerator," *Proceedings - Design Automation Conference*, vol. 2020-July, Feb. 2020, Accessed: Jul. 07, 2021. [Online]. Available: <https://arxiv.org/abs/2002.05022v2>