MJFAS
Malaysian Journal of Fundamental and Applied Sciences

Alsudani *et al.* | Malaysian Journal of Fundamental and Applied Sciences, Vol. 19 (2023) 379-388

RESEARCH ARTICLE

Automatic Car Number Plate Detection using Morphological Image Processing

Mustafa Qahtan Alsudani[a], Safa Riyadh Waheed[a], Karrar A Kadhim[a*], Myasar Mundher Adnan[b], Ameer Al-khaykan[c]

[a]Computer Techniques Engineering Department, Faculty of information Technology, Imam Jaafar Al-sadiq University, Iraq; [b]Islamic University, Najaf, Iraq; [c]Air conditioning and Refrigeration Techniques Engineering Department, Al-Mustaqbal University College, 51001 Hillah, Babylon, Iraq

Abstract One of the most common uses of computer vision, automatic number plate recognition (ANPR) is also a pretty well-explored subject with numerous effective solutions. Due to regional differences in license plate design, however, these solutions are often optimized for a specific setting. Number plate recognition algorithms are often dependent on these aspects, making a universal solution unlikely due to the fact that the image analysis methods used to develop these algorithms cannot guarantee a perfect success rate. In this research, we offer an algorithm tailor-made for use with brand-new license plates in Iraq. The method employs edge detection, Feature Detection, and mathematical morphology to find the plate; it was developed in C++ using the OpenCV library. When characters were found on the plate, they were entered into the Easy OCR engine for analysis.

Keywords: Car plate, car number detection, edge detection, feature extraction, OpenCV.

***For correspondence:**
mustafa.qahtan@sadiq.edu.iq

Received: 3 Nov. 2022
Accepted: 11 April 2023

Introduction

To enhance traveller protection, road security, and convenience, Intelligent Transportation Systems (ITSs) have grown widespread [1]. The technology known as Automatic Number Plate Recognition (ANPR) is used in ITSs to recognize cars by photographing their number plates and then deducing the registration information from the pictures [2]. In 1976, police in the United Kingdom created automatic number plate recognition. There are generally three phases to an ANPR process:

1. Detection of Plates
2. Segmentation of plates
3. Recognizing characters.

If the method fails at any point during the plate detection phase, the whole process is useless. At this point, the characteristics of the license plate are crucial [3]. Characteristics such as form, color, dimensions, symmetry, and spatial frequency are included. The nature of the picture and the circumstances under which it was acquired (including illumination, visibility, image skew, and camera quality) all affect the algorithm's ability to recognize the plate [4-7].

10.11113/mjfas.v19n3.2910

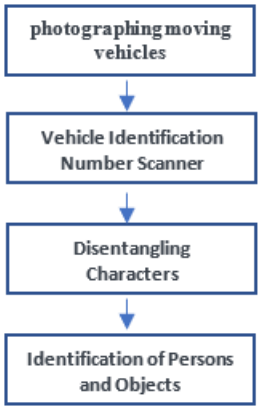

Figure 1. How ANPR works

When people speak about artificial intelligence, computer vision almost always comes up. Despite the hype, humans have always been fascinated by it since the human eye is the most important sensory organ [8]. Many different types of human subjects were used in computer vision experiments. Instead, we've settled on a study of license plates for our vehicles. The use of Optical Character Recognition is another motivation for our work on this issue; once a license plate is detected, we can quickly identify, extract, and display the number (OCR) [9]. Research into intelligent transportation systems is being actively used nowadays because to the miraculous advances made in science. The Automatic Vehicle License Plate Reader (AVLPR) is a computerized vision system used to spot license plates on moving vehicles. The cost of implementing any conventional AVLPR system is high since they all rely on proprietary technology. In a multiethnic context, people from different backgrounds work together to find answers to the endless issues that plague humanity. Python is one of the most important open-source projects in this field of science. The open computer vision (OpenCV) library is the result of Intel's research into computer vision and is essential to the field's further development [10]. Uploading photos taken from the front or rear of a car, followed by analyzing the picture to identify the license plate, is how number plate recognition works. A few examples of the three main stages are as follows:

In the first place, the processing of the scene benefits from the identification and placement of a license plate in this section.
The second stage is extracting the characters used for character segmentation from the observed numeric plates.
Third, optical character recognition (OCR) is used to convert the text into encoded text data.

Many institutions, including the police and local governments, now rely heavily on Automatic Number Plate Recognition (ANPR) technology as their principal security mechanism [11-13]. To put it simply, ANPR is a deterrent and a safety net. Knowing that one's license plate is being recorded and reviewed might deter illegal activity before it even begins. To that end, ANPR has applications in both public and private spheres. The best thing is that no human intervention is required, since this technology is totally automated and provides very accurate monitoring around the clock. Because of how efficient they are, you can trust them more than the typical individual. ANPR cameras are most effective when used by the police, who may filter through the data in search of vehicles suspected of being involved in illegal activities [14].

The real-time imaging advantages of ANPR have broad applicability. It would take much more time to warn offenders if their license plate was logged. However, ANPR makes it possible to scan a vehicle's license plate and instantly check its validity against a database [15].

OpenCV

To put it simply, (Open-Source Computer Vision Library) is a free, publicly available library for use in computer vision and machine learning [16]. Its primary goal is to offer a standardized framework for computer vision applications, with an emphasis on image and video processing, capturing, and analysis, including capabilities like face and object identification. The Open-Source Computer Vision Library, or OpenCV, is a free and open-source program for processing images and learning from machine data.

OpenCV was created to standardize the development of computer vision programs and hasten the incorporation of machine perception into consumer goods. OpenCV's BSD license facilitates code adoption and customization by commercial entities [17, 18].

The library's collection of over 2500 optimized algorithms cover the gamut from time-tested standards to cutting-edge methods in computer vision and machine learning. These algorithms may be used to do things like find similar images in a database, remove red eyes from images taken with a flash, follow eye movements, recognize scenery, and establish markers to overlay onto a 3D model of a person's face or body [19].

Haar Cascade XML File

Cascading is a kind of ensemble learning where many classifiers are chained together, with each classifier's output feeding into the next one. Cascading is a multistage multi-expert approach, as opposed to voting or stacking ensembles [20].

Multiple hundred "positive" sample views of the same item, as well as random "negative" photos of the same size, are used to train a cascading classifier. Once the classifier has been trained, it may be used to analyze a small section of a picture in order to identify the target item. Moving the search window over the picture allows the classifier to look in every possible position for the item. Object detection and tracking, especially face detection and identification, are two of the most popular applications of this method in the field of image processing [21].

Calculating Haar Features

As an initial step, we'll amass several Haar features. In a detection window, a Haar feature is the result of some math done on neighboring rectangular sections. Each region's pixel intensities are summed, and the differences between those sums are computed. A few instances of Haar characteristics are seen in Figure 2 below.

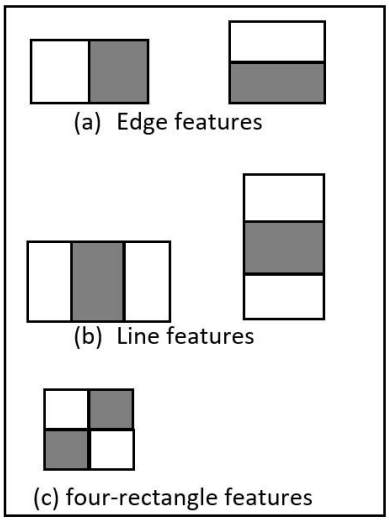

Figure 2. Haar characteristics

For a big picture, it might be tough to pinpoint these details. Because the integral image requires fewer processes, it is useful in this context [22].

Creating Integral Images

Using integral images greatly expedites the calculation of these Haar features. Instead of doing calculations at each each pixel, it divides the image into smaller rectangles and assigns each of those

rectangles its own array reference. The algorithm that creates the Haar features is then fed these inputs [23].

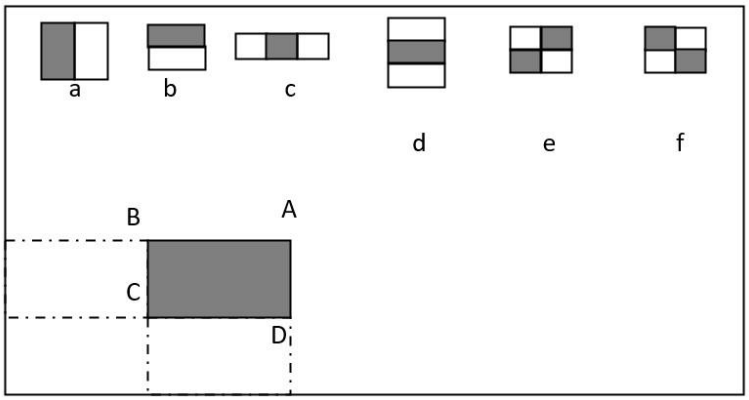

Figure 3. Haar features, divides the image into smaller rectangles

When doing object identification, just the properties of the object itself will matter, making almost all of the Haar features superfluous. However, among the tens of thousands of Haar features, how can we choose the best characteristics that accurately portray the object? Adaboost is useful because it allows you to do this.

Adaboost Training

To put it simply, Adaboost picks the top features and teaches the classifiers to utilize them. To improve the algorithm's object detection capabilities, it combines many "weak classifiers" into a single "strong classifier.

A weak learner is generated by iteratively sliding a window across an input picture and calculating Haar features for each resulting region. In order to distinguish between things and non-objects, this variation is measured against a trained threshold [24]. As a result of their status as "weak classifiers," a large collection of Haar features is required to successfully construct a robust classifier.

Implementing Cascading Classifiers

Every level of the cascade classifier is a set of somewhat incompetent learners. Training using boosting improves the accuracy of weak learners by utilizing their collective predictions to create a more robust model [25]. The classifier will then either report a successful item detection (positive) or skip over the area in question (negative) (negative). Because the vast majority of windows do not contain anything of interest, the stages are optimized to quickly discard negative samples. For an effective object detection method, it is crucial to minimize the number of times an item is incorrectly labelled as being something it is not [26]. Among the numerous available methods for object identification, Haar cascades is a popular choice. To minimize the model's false-negative rate during training using Haar cascades, it's crucial to find the optimal values for the hyperparameters.

Easy OCR

If you have a scanned document, a photograph of a document, a photograph of a scene (such as the text on signs and billboards in a landscape photograph), or a photograph with subtitle text superimposed on it, you can use optical character recognition (OCR) to convert the image into machine-encoded text (for example: from a television broadcast) [27].

Data entry from paper records like passports, invoices, bank statements, computerized receipts, business cards, mail, printouts of static data, and any suitable documentation is a common way to digitize printed texts for use in electronic editing, searching, compact storage, online display, and machine

processes like cognitive computing, machine translation, and (extracted) text-to-speech synthesis. Pattern recognition, artificial intelligence, and computer vision are all explored in OCR studies.

Early iterations required training photos of each character and could only process one typeface at a time. These days, it's not uncommon to come across sophisticated systems that can recognize a wide range of typefaces with a high degree of accuracy and that can accept input from a wide range of digital picture file formats. In certain cases, it is possible to generate an output formatted similarly to the original page, right down to the graphics, columns, and other non-textual elements [28].

As its name indicates, EasyOCR is a Python library that makes Optical Character Recognition a breeze for those working in computer vision, to implement OCR in its simplest form, EasyOCR is your best bet: Simply running the pip command will get you the EasyOCR package. Since the EasyOCR package has few prerequisites, setting up an OCR development environment is a breeze. When you have EasyOCR set up, you can import the package into your project with a single import line. After that, OCR can be accomplished with only two lines of code: one to start the Reader class, and another to OCR the picture using the read text method.

Pre-processing

Binarization allows for the conversion of color or grayscale images to black and white (called a "binary image" because there are two colors). Binarization allows the text (or any necessary image component) to be quickly and readily separated from its background. Since it is simpler to work with binary images, most commercial recognition algorithms only work with them. This makes binarization an important task in and of itself. As the quality of the binary output obtained by binarisation is highly reliant on the kind of input picture, much attention is taken in selecting the binarisation to be utilized for every specific input image. For better or worse, this affects the accuracy of the character recognition process (scanned document, scene text image, historical degraded document, etc.) [29].

Removes unwanted lines and boxes that aren't glyphs.
Analysis of the layout, often known as "zoning," identifies sections such as columns, paragraphs, captions, and so on. Particularly relevant in table and multi-column layouts.
The detection of lines and words creates a standard for the forms of words and characters and, if required, divides them apart.
Recognizing the script is essential for using the appropriate optical character recognition (OCR) system, since in multilingual publications, the script may vary at the word level.
Isolation of individual characters, or "segmentation," is required for per-character OCR when numerous characters are joined together as a result of picture artifacts or when a single character is fragmented into many fragments as a result of image abnormalities. Scale and aspect ratio should be normalized [30].

Aligning the picture to a uniform grid based on the points at which vertical grid lines would least commonly overlap dark regions makes fixed-pitch font segmentation a straightforward process. Because the white space between letters in a proportional font might be larger than the white space between words, and because vertical lines can cross over more than one character, more complex methods are required for designing fonts with proportional lettering.

Result and implementation

The suggested approach for adaptive number plate detection requires the import of five modules, including cv2(OpenCV), utils, NumPy, Pytesserract, and Easy OCR. These library components are often used to accurately recognize numbers or license plates and extract characters [31-34]. F. A Technique for Projecting Both Horizontal and Vertical Locations:

To guarantee proper placement of the license plate, it is necessary to assess the approximate location of the plate in order to segment and correctly identify the character. From this, we can get the equation for the horizontal projection:

$$IM(i,j)=| P I(i,j)- P I(i,j - 1)|,G$$

$$G = \{i = 1,2,3,...,M,j = 2,3,4,...,N\}$$

Here, IM(i, j) is the image IM pixels, P I(i, j) is the image PI pixel and M and N are the PI image pixels accordingly to the height and width. So the projection value of the row x named P1(x) can calculate the pixel value of an image IM per row, that is derived Thus, P1(x) row projection value can be calculated by calculation of the pixel of the picture IM picture per row. as:

$$P1(x) = \Sigma n\ y=2IM(x,y)$$

The rollout process will occur in stages.
Figure 4 depicts the process of using OpenCV to collect live video from any connected camera once the input automobile picture has been imported. Since OpenCV imports pictures in the BGR (Blue-Green-Red) format by default, we must first execute cv2.cvtColor to convert it to the RGB format that matplotlib requires for display.

The second action is to convert the picture to black and white. OpenCV's Cascade Classifier function will be used to import the Haar Cascade feature set (XML file) for license plates. As shown in Figure 4, we next use the Cascade Classifier's detect Multi Scale method to carry out the detection.

Figure 4. Car image captured by the camera

Our next step is to focus on the license plate and work towards extracting the numbers and text of the car plate using OCR capabilities as in Figure 5.

Figure 5. The car license plate has been successfully detected and bounded by a blue rectangle

Use EasyOCR to read the license plate from the car's photograph for free! It is necessary to carry out a number of image processing procedures to guarantee the accuracy of the OCR feature. First, let's take a look at just the license plate itself. The process involves putting up a function like to the one we used for license plate detection previously, only this time we will remove the license plate and return it as a separate picture. Figure 6 shows what we get when we run the function: a cropped view of only the license plate from the automobile. The function is called imgplate.

Figure 6. Showing the extracted image only

The picture will then be smoothed out. The smoothing method is useful for eliminating noise and letting the program concentrate on the broad outlines of a picture. This process, which we refer to as denoising, makes the image's text characters easier to make out. The median blur (usingcv2.medianBlur) was then used for the smoothing, which involves calculating the median of all the pixels beneath the kernel window and then using that value to replace the centre pixel. We found that median blur performed really well (far better than Gaussian blur) for this task, and thus we showcased it here.

Morphological image processing is a subfield of image processing that focuses on the non-linear manipulation of images based on the shape or morphology of their features. These operations place more emphasis on the relative ordering of pixels than on their absolute values, making them well-suited for processing binary images. Additionally, morphological techniques can be done to greyscale photographs to render their light transfer functions unknown, making the absolute pixel values irrelevant or of small relevance. Morphological methods examine an image by using a little shape or template. We compare the surrounding pixels in the image to the neighborhood of the structuring element placed in each of the possible positions in the image, depending on the type of operation, it may be determined if the element "fits" into the neighborhood or if it "hits" or intersects.

Perhaps only with careful inspection can you see that the letters' edges are less jagged and more rounded than in the original picture. After these adjustments, our picture is prepared for optical character recognition (OCR). Smoothing to a greater extent is necessary for other types of photographs. We utilized a standard CSV file as the database, with a search function that employs the capturing ad detecting function to capture the result text and then search for the text in the CSV file, after which text extraction will be carried out, as shown in Figure 7.

Figure 7. Text extracting of the plate number.

Discussion

That ANPR code up there detects license plates. In other words, the rectangular shape taken from the frame of number plate identification constitutes an accurate character generator. When the steps outlined below are used to implement the suggested method:

A picture of the license plate is first captured by the suggested model and then sent on to the python module for number plate extraction.
In Step 2, produced Grayscale and also canny pictures are used to demonstrate the license plate identification method (Localization). Third, the License Plate is identified, analyzed, and segmented. Finally, a message stating "not found!" is shown if the detected license plate does not match any in the CSV file containing information on the vehicle's owner. This study aims to show how advanced free and open-source software may be in the scientific community. Photographs of license plates are taken by an ANPR system's cameras, and the corresponding numbers are then used to get relevant information and data for the vehicle's owner. In our suggested approach, we have used technology to picture license plates. Reduced background noise allows for easier detection of progress. The process then proceeds to banalization and segmentation. With the aim of gathering the owner's information and data, the image collection system draws a picture of the number plate in the NPR system and recognizes the car license number. For the sake of our newspaper, we have developed a technology that automatically snaps photos of passing vehicles' license plates. Improvement is seen by a simultaneous decrease in noise. By utilizing the template as a matcher, we can separate the characters and determine who they are.

Text Recognition

There are two main categories of OCR algorithms, any of which may provide a prioritized list of potential characters.

Matrix matching, also known as "pattern matching," "pattern recognition," and "image correlation," is the process of comparing a picture to a database of recorded glyphs pixel by pixel. For this to work, it is necessary that the input glyph be properly separated from its surrounding picture, and that the stored glyph be of a comparable font and size. However, this method suffers when confronted with unfamiliar typefaces and works best with typewritten material. Directly, this is the method that the first OCR systems based on actual photocells used.

"Features" such as lines, closed loops, line direction, and line intersections are extracted from glyphs. To make the recognition procedure more computationally efficient, the characteristics extracted from the representation are used. An abstract vector representation of a character is compared to these details; this representation may be reduced to one or more glyph prototypes. This form of OCR is typical of "intelligent" handwriting recognition and, indeed, most cutting-edge OCR programs, and it makes use of general methods of feature identification in computer vision. Classifiers based on the concept of nearest neighbors, such as the k-nearest neighbor's algorithm, are used to find the closest match between an image's characteristics and the glyphs in a database. Cuneiform and Tesseract are two examples of character recognition software that employ a two-pass method. The second phase, called "adaptive recognition," makes advantage of the letter shapes identified with high confidence in the first pass to identify the remaining letters more accurately. Having this option is helpful when working with rare typefaces or deformed fonts caused by low-quality scans (e.g. blurred or faded). Google Docs OCR, ABBYY FineReader, and Transym are just a few examples of up-to-date OCR programs. While some OCR programs concentrate on individual letters, others, like Tesseract and OCRopus, employ neural networks that have been trained to detect whole lines of text. Iterative optical character recognition is a recent method that uses page layout to mechanically trim a text into pieces. Each part undergoes OCR separately, with character confidence level criteria that may be adjusted for optimal page-level OCR accuracy. The United States Patent and Trademark Office has issued a patent for this process. The ALTO format, a specific XML structure maintained by the US Library of Congress, may be used to save the OCR output. OCR and PAGE XML are two more prevalent file types. See OCR Software Comparison for a rundown of available programs that can decipher handwritten text.

Post-processing

A lexicon is a list of terms that are permitted to exist in a text, and using this list to confine OCR output improves accuracy. This might include the whole English vocabulary or it could be a more technical lexicon for a particular profession. In cases when the text in question includes proper nouns or other specialized terms that aren't part of the lexicon, this method may fail to provide the desired results. Tesseract improves the precision of the character segmentation process by using its dictionary. More advanced OCR systems may maintain the page's structure and generate an annotated PDF with both the original picture and a searchable textual representation, for example. The output stream may be a plain text stream or file of characters. Near-neighbor analysis" may employ co-occurrence frequencies to fix mistakes by taking into account how often specific words appear together. "Washington, D.C." is far more prevalent than "Washington DOC" in the English language. It is possible to improve accuracy by scanning accuracy by knowing the grammar of the language being scanned, and determining whether a word is more likely to be a verb or a noun. There have been applications of the Levenshtein Distance method in OCR post-processing to further enhance the quality of the data obtained from an OCR API.

Conclusion

Ultimately, this study aims to show how advanced open-source and free software may be in the scientific community. Python and OpenCV are great venues for researchers and students in Computer Vision to get their feet wet. The ANPR system works by taking a photo of a car's license plate and then using number recognition technology to send that information on to the car's owner. In our proposed approach, we have used technology to picture license plates. That's a good enough point where the progress signs are discernible from the background noise. As a next step, we perform segmentation and banalization. When a car pulls up, a picture of the front of the car is taken by the camera mounted on the door or in the police car, and the system then localizes the number plate and does further recognition. If the license

plate is valid, the door will open; otherwise, an alert will sound. In order to get the owner's identification and data, the image collecting system in the NPR system draws a picture of the number plate of the car in question. For the purpose of our newspaper, we have developed a technology that automatically snaps photos of passing vehicles' license plates. Improvement is demonstrated through a simultaneous reduction in noise. Template matching will be used to separate and identify the characters. No matter what, the method can only be applied to binary images and not RGB ones. Combining our software with mobile devices or the self-made gadgets employing a touch panel and RasberryPi is another enhancement we're considering making. As a result, it will be considerably easier to transport and operate, and it will be capable of performing a wider range of security-related duties.

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

Acknowledgment

This work is part of a research project, supported by the Ministry of Higher Education, Malaysia, and the University of Technology Malaysia. Also supported by Imam Ja'afar Al-Sadiq University, Baghdad, Iraq.

References

[1] Zhang, J., Wang, F. Y., Wang, K., Lin, W. H., Xu, X., & Chen, C. (2011). Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, *12*(4), 1624-1639.

[2] Tang, J., Wan, L., Schooling, J., Zhao, P., Chen, J., & Wei, S. (2022). Automatic number plate recognition (ANPR) in smart cities: A systematic review on technological advancements and application cases. *Cities*, *129*, 103833.

[3] Anagnostopoulos, C. N. E., Anagnostopoulos, I. E., Loumos, V., & Kayafas, E. (2006). A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems*, *7*(3), 377-392.

[4] Kadhim, K. A., Adnan, M. M., Waheed, S. R., & Alkhayyat, A. (2021). WITHDRAWN: Automated high-security license plate recognition system.

[5] Salim, A. A., Ghoshal, S. K., Shamsudin, M. S., Rosli, M. I., Aziz, M. S., Harun, S. W., ... & Bakhtiar, H. (2021). Absorption, fluorescence and sensing quality of Rose Bengal dye-encapsulated cinnamon nanoparticles. *Sensors and Actuators A: Physical*, *332*, 113055.

[6] Salim, A. A., Ghoshal, S. K., & Bakhtiar, H. (2022). Prominent absorption and luminescence characteristics of novel silver-cinnamon core-shell nanoparticles prepared in ethanol using PLAL method. *Radiation Physics and Chemistry*, *190*, 109794.

[7] Hathot, S. F., Abbas, S. I., AlOgaili, H. A. T., & Salim, A. A. (2022). Influence of deposition time on absorption and electrical characteristics of ZnS thin films. *Optik*, *260*, 169056.

[8] Kakani, V., Nguyen, V. H., Kumar, B. P., Kim, H., & Pasupuleti, V. R. (2020). A critical review on computer vision and artificial intelligence in food industry. *Journal of Agriculture and Food Research*, *2*, 100033.

[9] Lemley, J., Bazrafkan, S., & Corcoran, P. (2017). Deep Learning for Consumer Devices and Services: Pushing the limits for machine learning, artificial intelligence, and computer vision. *IEEE Consumer Electronics Magazine*, *6*(2), 48-56.

[10] Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: fast and flexible image augmentations. *Information*, *11*(2), 125.

[11] Patel, C., Shah, D., & Patel, A. (2013). Automatic number plate recognition system (anpr): A survey. *International Journal of Computer Applications*, *69*(9).

[12] Abbas, A. M., Abid, M. A., Abbas, K. N., Aziz, W. J., & Salim, A. A. (2021, April). Photocatalytic activity of Ag-ZnO nanocomposites integrated essential ginger oil fabricated by green synthesis method. *Journal of Physics: Conference Series*, *1892*(1), 012005. IOP Publishing.

[13] Salim, A. A., Ghoshal, S. K., Danmallam, I. M., Sazali, E. S., Krishnan, G., Aziz, M. S., & Bakhtiar, H. (2021, April). Distinct optical response of colloidal gold-cinnamon nanocomposites: Role of pH sensitization. *Journal of Physics: Conference Series*, *1892*(1), 012039. IOP Publishing.

[14] Tang, J., Wan, L., Schooling, J., Zhao, P., Chen, J., & Wei, S. (2022). Automatic number plate recognition (ANPR) in smart cities: A systematic review on technological advancements and application cases. *Cities*, *129*, 103833.

[15] Zhang, H., Chen, P., Zheng, J., Zhu, J., Yu, G., Wang, Y., & Liu, H. X. (2019). Missing data detection and imputation for urban ANPR system using an iterative tensor decomposition approach. *Transportation Research Part C: Emerging Technologies*, *107*, 337-355.

[16] Bradski, G., Kaehler, A., & Pisarevsky, V. (2005). Learning-based computer vision with intel's open source computer vision library. *Intel Technology Journal*, *9*(2).

[17] Dehghani, M., Gritsenko, A., Arnab, A., Minderer, M., & Tay, Y. (2022). Scenic: A JAX library for computer vision research and beyond. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*

Recognition (pp. 21393-21398).
[18] Hoydis, J., Cammerer, S., Aoudia, F. A., Vem, A., Binder, N., Marcus, G., & Keller, A. (2022). Sionna: An open-source library for next-generation physical layer research. *arXiv preprint arXiv:2203.11854*.
[19] Kaur, J., Sinha, P., Shukla, R., & Tiwari, V. (2021). Automatic Cataract Detection Using Haar Cascade Classifier. In *Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2020* (pp. 543-556). Springer Singapore.
[20] Carabe, L., & Cermeño, E. (2021). Stegano-morphing: Concealing attacks on face identification algorithms. *IEEE Access*, *9*, 100851-100867.
[21] Poojari, N. N., Sangeetha, J., & Shreenivasa, G. (2022). Automatic Student Attendance and Activeness Monitoring System. *Intelligent Systems and Sustainable Computing: Proceedings of ICISSC 2021* (pp. 405-415). Singapore: Springer Nature Singapore.
[22] Rahmad, C., Asmara, R. A., Putra, D. R. H., Dharma, I., Darmono, H., & Muhiqqin, I. (2020). Comparison of Viola-Jones Haar Cascade classifier and histogram of oriented gradients (HOG) for face detection. *IOP Conference Series: Materials Science Aand Engineering*, *732*(1), 012038. IOP Publishing.
[23] Kim, D., Hyun, J., & Moon, B. (2020, January). Memory-efficient architecture for contrast enhancement and integral image computation. In *2020 International Conference on Electronics, Information, and Communication (ICEIC)* (pp. 1-4). IEEE.
[24] Xing, H. J., & Liu, W. T. (2020). Robust AdaBoost based ensemble of one-class support vector machines. *Information Fusion*, *55*, 45-58.
[25] Nafees, A., Amin, M. N., Khan, K., Nazir, K., Ali, M., Javed, M. F., ... & Vatin, N. I. (2022). Modeling of mechanical properties of silica fume-based green concrete using machine learning techniques. *Polymers*, *14*(1), 30.
[26] Wang, L., & Yoon, K. J. (2021). Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
[27] Li, Y. Q., Chang, H. S., & Lin, D. T. (2022). Large-scale printed chinese character recognition for ID cards using deep learning and few samples transfer learning. *Applied Sciences*, *12*(2), 907.
[28] Phoenix, P., Sudaryono, R., & Suhartono, D. (2021). Classifying promotion images using optical character recognition and Naïve Bayes classifier. *Procedia Computer Science*, *179*, 498-506.
[29] Bijeesh, T. V., & Narasimhamurthy, K. N. (2020). Surface water detection and delineation using remote sensing images: A review of methods and algorithms. *Sustainable Water Resources Management*, *6*, 1-23.
[30] Waheed, S. R., Suaib, N. M., Rahim, M. S. M., Adnan, M. M., & Salim, A. A. (2021, April). Deep learning algorithms-based object detection and localization revisited. *Journal of Physics: Conference Series*, *1892*(1), 012001. IOP Publishing.
[31] Salim, A. A., Bidin, N., & Islam, S. (2017). Low power CO_2 laser modified iron/nickel alloyed pure aluminum surface: Evaluation of structural and mechanical properties. *Surface and Coatings Technology*, *315*, 24-31.
[32] Salim, A. A., Ghoshal, S. K., Suan, L. P., Bidin, N., Hamzah, K., Duralim, M., & Bakhtiar, H. (2018). Liquid media regulated growth of cinnamon nanoparticles: Absorption and emission traits. *Malaysian Journal of Fundamental and Applied Sciences*, *14*(3-1), 447-449.
[33] Salim, A. A., Bakhtiar, H., Shamsudin, M. S., Aziz, M. S., Johari, A. R., & Ghoshal, S. K. (2022). Performance evaluation of rose bengal dye-decorated plasmonic gold nanoparticles-coated fiber-optic humidity sensor: A mechanism for improved sensing. *Sensors and Actuators A: Physical*, *347*, 113943.
[34] Waheed, S. R., Rahim, M. S. M., Suaib, N. M., & Salim, A. A. (2023). CNN deep learning-based image to vector depiction. *Multimedia Tools and Applications*, 1-20.