

## Research Article

# A Machine Learning Algorithm to Automate Vehicle Classification and License Plate Detection

S. R. Srividhya<sup>1</sup>, C. Kavitha<sup>1</sup>, Wen-Cheng Lai<sup>2,3</sup>, Vinodhini Mani<sup>1</sup>,  
and Osamah Ibrahim Khalaf<sup>4</sup>

<sup>1</sup>Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai, India

<sup>2</sup>Bachelor Program in Industrial Projects, National Yunlin University of Science and Technology, Douliu 640301, Taiwan

<sup>3</sup>Department Electronic Engineering, National Yunlin University of Science and Technology, Douliu 640301, Taiwan

<sup>4</sup>Al-Nahrain University, Al-Nahrain Nanorenewable Energy Research Center Baghdad, Iraq

Correspondence should be addressed to S. R. Srividhya; [vidhyasrinivasan1890@gmail.com](mailto:vidhyasrinivasan1890@gmail.com)  
and C. Kavitha; [kavitha4cse@gmail.com](mailto:kavitha4cse@gmail.com)

Received 3 February 2022; Revised 1 March 2022; Accepted 9 March 2022; Published 6 June 2022

Academic Editor: Kuruva Lakshmanna

Copyright © 2022 S. R. Srividhya et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the field of intelligent transportation systems (ITS), video surveillance is a hot research topic; this surveillance is used in a variety of applications, such as detecting the cause of an accident, tracking down a specific vehicle, and discovering routes between major locations. Object detection and shadow elimination are the main tasks in this area. Object detection in computer vision is a critical and vital part of object and scene recognition, and its applications are vast in the fields of surveillance and artificial intelligence. Additionally, other challenges arise in regard to video surveillance, including the recognition of text. Based on shadow elevation, we present in this work an inner-outer outline profile (IOOPL) algorithm for detecting the three levels of object boundaries. A system of video surveillance monitoring of traffic can be incorporated into this method. It is essential to identify the type of detected objects in intelligent transportation systems (ITS) to track safely and estimate traffic parameters correctly. This work addresses the problem of not recognizing object shadows as part of the object itself in-vehicle image segmentation. This paper proposes an approach for detecting and segmenting vehicles by eliminating their shadow counterparts using the delta learning algorithm (Widrow-Hoff learning rule), where the system is trained with various types of vehicles according to their appearance, colors, and build types. An essential aspect of the intelligent transportation system is recognizing the type of the detected object so that it can be tracked reliably and the traffic parameters can be estimated correctly. Furthermore, we propose to classify vehicles using a machine learning algorithm consisting of artificial neural networks trained using the delta learning algorithm, a high-performance machine learning algorithm, to obtain information regarding their travels. The paper also presents a method for recognizing the number plate using text correlation and edge dilation techniques. In regard to video text recognition, number plate recognition is a challenging task.

## 1. Introduction

The field of intelligent transportation systems (ITS) uses video surveillance in different ways, including finding out what caused an accident, tracking down a specific vehicle, and discovering routes between major locations. This area focuses primarily on detecting objects and eliminating their shadows[41]. It plays a crucial role in recognizing objects and scenes in the world of computer vision, and its applications in artificial intelligence and surveillance are endless. In

addition to video surveillance, the recognition of text is another challenge. An algorithm for detecting three levels of object boundaries is presented that enhances the segmentation of satellite imagery using the inner-outer outline profile (IOOPL). Incorporating a video surveillance system to monitor traffic can be incorporated into this method. To maximize traffic safety and predict traffic parameters accurately, it is critical to identify the type of detected objects in intelligent transportation systems (ITS). A problem with vehicle image segmentation arises from the failure to recognize shadows on

objects as part of the actual object. Using the Widrow-Hoff learning rule (delta learning algorithm) [1], this paper presents a method for detecting and segmenting vehicles with the help of their shadow counterparts. Vehicles of various appearance, colors, and build types are used to train the system. The identification of the type of detected object is an integral part of intelligent transportation system (ITS) to track it with reliability and estimate traffic parameters. A high-performance machine learning algorithm [2] is intended to be used to classify the vehicles using an artificial neural network-based algorithm [3] to generate logs that represent the vehicles' travel details. A method of number plate recognition using edge dilation and text correlation is demonstrated in this paper. Video text recognition presents a challenging challenge in regard to number plate recognition. For the traffic parameters to be correctly estimated, these things must be considered. Calculate traffic rates, for example, before designing flyovers. The congestion of transport infrastructure leads to pollution and time wasted by users, so it has an economic effect as well. The use of this ITS can minimize such problems.

It is the integration of information technology and communications (ICT) into vehicles and highways so that traffic conditions can be monitored, mobility can be increased, congestion can be reduced, and traffic security can be enhanced [4]. Moreover, they provide the user with traffic forecasts and alternative routes or ways to reach specific areas or places in different regions. Classification [5], shadow elimination, object detection, and classification are proposed in this research to minimal congestion, traffic prediction, avoid accidents, for investigations, etc. The detecting edges, maiming shadow, detecting vehicles, and delta learning algorithms for classification of vehicles, log creation, and license plate detection are handled with these algorithms. A deep learning algorithm is used [6] for detecting and classifying vehicles in a complex background environment, eliminating shadows, and then identifying the vehicle. To interpret a set of random variables, text correlation is used, along with edge dilation, as a statistical method of finding the relation between the variables. In edge dilation, the actual and background objects in a picture are detected and categorized. Among the factors that could adversely affect this system's results are weather conditions, lighting, plate placement, vehicle movement, and mechanical plate damage. All the above factors can be eliminated by utilizing proper lighting, specialist video equipment, and adequate computer image processing. Otherwise, the system may become complex. This will be enhanced in our future work.

The following factors contribute to difficulties encountered during the detection and extraction of number plates:

- (1) Scene complexity influences the effectiveness of extraction
- (2) The plates on different vehicles are positioned differently
- (3) Noise can be generated during the camera capture process

- (4) Noise is caused by weather conditions
- (5) Due to the time of day, lighting can affect contrast
- (6) Adding unnecessary characters, frames, and screws creates confusion
- (7) The position of the camera or the plate results in distortions that affect the efficiency of the plate extraction
- (8) Lighting conditions that result in low or uneven illumination, blurred images, low-resolution images, reflections, and shadows affect the accuracy of number plate area extraction

*1.1. Motivation.* In the field of computer vision for humans and vehicles, the most active research topic is traffic surveillance. As the number of cars grows rapidly, license-plate recognition has become essential. Vehicles have facilitated human life, but have also caused various problems, such as congestion, parking problems, and accidents. Surveillance systems typically consist of traffic camera networks that process video captured at the location and transmit the dimensions in real time. As part of this study, the work examines the algorithmic aspects of such a system.

*1.2. Contribution.* The major contribution of this system is organized as follows:

- (1) The work designs a system that uses a camera for continuous video recording in real time
- (2) The research improve the traffic flow prediction system, avoid accidents, and estimate the amount of congestion
- (3) It designs two algorithms to identify the middle of the road, remove shadows, and classify vehicles
  - (i) Shadow maiming and detecting vehicles using IOOPL algorithm
  - (ii) Vehicle classification, license plate detection, and log generation using a delta learning algorithm
- (4) The shadow of a vehicle is eliminated, followed by the detection and classification of the vehicle in a complex background environment

The rest of this paper is organized as follows. In Section 2, the related works are discussed. Section 3 explains the proposed methodology and Section 4 reports the experimental results and the comparative evaluation. Finally, Section 5 concluded this paper.

## 2. Related Works

The foreground and background objects are differentiated via Gaussian distributions, which are modeled mathematically (background elimination). Traffic images are the only ones for which a fixed thresholding value can be applied. Objects and background are the only things that can be differentiated. In regard to traffic monitoring and shadow

elimination[38], multilayer object detection is impossible. The only methods for eliminating shadows without log generation are shadow elimination and shadow classification [7]. Gaussian distribution [7] in image processing is one of the methods for determining the threshold value of the background. The number of pixels occurrences is maximum. The paper requires high-resolution videos for maiming shadows and classifying vehicles in the traffic video surveillance context [7], and manual searches are required. An intelligent transportation system has been developed to resolve this issue, which uses log queries to identify a vehicle. Objects can also be extracted from video or moving images in addition to threshold comparison, pixel-by-pixel processing, edge detection, interframe differencing, and background subtraction. These image processing methodologies are described in [8]. A study in [9] demonstrated that background subtraction is the most accurate method for tracking traffic as well as uses it in many other applications, such as motion capture, handwriting recognition, and video surveillance [7].

The Gaussian distribution is modeled on each pixel of a video frame in the background subtraction process. There are several methods for doing this [10]. Natural environments with multimodal backgrounds do not work well with the model. The mixture of Gaussians models (GMM) [11] is a popular technique for modeling complex backgrounds. By utilizing video surveillance, it can be applied to the problem of traffic monitoring over complex backgrounds. A key algorithm for background modeling, GMM, is an effective method for detecting motion and identifying objects from video sequences. As the author's state in [12], the GMM is irrelevant for outdoor scenes. Based on the results, they replaced the GMM probability density function (PDF) with a kernel-based method of estimating pixel intensity distribution in outdoor scenes. However, the memory requirement and the time it takes to calculate the kernel density per pixel are both quite high. The codebook is another efficient technique described in [13]. The first step before the identification of license plates is the detection of license plates. The system uses dynamic images to identify the license plate region based on the characteristics of the sensors. It uses two types of dynamic images, MMADR and NDDR, to locate the cars on the screen [14].

An ITS (intelligent transportation system) needs to be able to detect and recognize license plates. Using vertical boundary pairs and geometric relationships, we propose a robust method of detecting license plates. By removing the noise simultaneously with recognizing the type of license plate from the vertical Sobel edge image, it is possible to identify [15]. Using this method, a variety of complexities can be handled, including bad illumination, blurring, and tilt [15].

Using a random forest, Bai et al. [16] propose an object-oriented approach to detecting change. To represent the features, the training samples are combined. Random forest classifiers are used for identifying changes in classification. According to Maulidia et al. [17], the accuracy of Otsu and K-nearest neighbor (KNN) can be obtained for converting an RGB image to a binary image, extracting attributes of

the image. Pattern recognition uses feature extraction to convert pixels into binary form. The Otsu method was used to extract features, after which the KNN was used to classify the image by comparing the neighborhood test and training data. Learning algorithm was used to classify test data into classes, which was based on the learning algorithm.

The authors of Liu et al. [18] presented a supervised K-means algorithm for separating number plate characters into subgroups which were classified further by the support vector machine (SVM). By identifying blurred number plate images, their system improved the accuracy of classification. By evaluating the angle of the camera, vehicle speed, and surrounding light and shadow, the system was able to distinguish the obstacles in character recognition. Images of unrecognizable and faint characters were captured. A large number of samples increased the workload of the SVM classifier, decreasing accuracy. [19] Quiros et al. used the KNN algorithm for analyzing number plate characters. The proposed system consisted of a camera installed on a highway, which analyzed the feed and captured images of vehicles. Based on the detected contours within the number plates, and their sizes, the plates were segmented using the detected contours. A KNN algorithm was used to classify each contour, which was trained using different sets of data containing 36 characters, comprising 26 alphabets and 10 numerical digits. Analyzing previously segmented characters and comparing the algorithm with character recognition techniques, such as artificial neural networks, was also carried out. Compared with the literature, their proposed system did not provide comparable character recognition results.

The authors argue in [12, 13] that the GMM is not effective for outdoor scenes. According to their findings, the distribution of pixel intensity over a long period of time covers a wide range of intensity, and therefore, the GMM probability density function (PDF) is replaced by a kernel-based density estimation method. As a result, the memory and time requirements to compute the kernel density at each pixel are extremely high. Using the codebook model proposed in [20, 21] the authors model each pixel based on a codebook of code-words. According to the authors, GMM and kernel methods cannot handle rare background pixel values. As a solution to this problem, they propose to use a training phase to model these rare pixel values, and every pixel value occurring during this phase must pass a periodical test. Pixel values that pass the background test are considered background.

**2.1. Discussion.** For predictable license plate styles, some automatic number plate recognition (ANPR) systems[42] may use simple image processing techniques, performed under controlled conditions. In contrast, advanced ANPRS use dedicated object detectors, such as HOG, CNN, SVM, and YOLO. ANPR software using state-of-the-art neural networks with AI capabilities is used in the more advanced and intelligent ANPR systems. ANPR also has applications in computer vision and machine learning, just like many other fields. It is challenging to implement ANPR due to the sheer variety of license plate types across states, territories, and countries. Number plate identification is further complicated by the fact that any ANPR algorithm will need

to work in real time. Several previous reviews from literature have been presented in Table 1, which shows the summary of ANPR system techniques. Various techniques were reviewed for each phase of the ANPR system in the cited work. A good collection of references is provided by the authors in [22, 23] for the new researchers in the license plate detection field. Despite this, the study did not compare the accuracy rates of different recognition techniques. Rather, the authors compared the performance of the various algorithms employed in the past. For the years 1999-2015, the efficiency of the ANPR system was presented in percentage terms. In the conclusion, we concluded that the efficiency of the ANPR system is not stable and that performance varies due to various factors like noise, environmental conditions, and algorithm and model selection.

### 3. Proposed Methodology

The IOOPL allows for shadow elimination and vehicle classification, and the delta learning rule is used to determine the classification of the vehicle, and the log is generated for future use. In the log, all details about classified vehicles are recorded for future reference. The workflow of the work is explained as in Figure 1.

**3.1. Traffic Surveillance Camera.** In general, surveillance refers to the monitoring of behavior, activities, or other changing information concerning people for influencing, managing, directing, or protecting them. Traffic surveillance cameras are video cameras[40] that monitor vehicular traffic on the road. As in urban areas, they use electrical power from main supplies or solar panels to provide consistent imagery without the threat of power outages during inclement weather. These systems are commonly found along major roads like highways, freeways, motorways, autoroutes, expressways, and arterial roads and are connected with optical fibers buried alongside or even under the road.

**3.2. Denoising.** The next step in the process of shadow elimination and vehicle detection is video denoising, in which the video signal is used to remove the noise present in every frame so that the vehicles can be easily identified.

**3.3. Image Encoding.** The encoder writes the image data to a stream. Prior to writing the image pixels to the stream, encoders can compress, encrypt, or alter the pixels in various ways.

**3.4. Background Elimination.** As part of image processing and computer vision, background elimination is also known as foreground detection, and it allows for detecting objects in the foreground (humans, cars, etc.). Areas of interest in an image are usually objects (humans, cars, text, etc.) in the foreground.

**3.5. Edge Detection.** In mathematical terms, edge detection refers to the identification of points in a digital image where the brightness changes sharply, more precisely discontinuities. A set of curved line segments referred to as edges, are typically created at points where image brightness changes sharply.

**3.6. Shadow Vehicle Extraction.** Extract the shadow of the vehicle to feed the next stage of the process. In the process of detecting the object, extracting this image simplifies the problem.

**3.7. IOOPL Implementation.** It incorporates a multilayered boundary that layers the object into three layers after extracting the image with a shadow, so that the exact boundaries of the object can be detected or classified. Additionally, this IOOPL aids in detecting or classifying the vehicle type after detecting the object with shadows.

**3.8. Shadow Less Vehicle Extraction.** Implementing IOOPL allows the object to be multilayered to detect the exact edges of the object as well as eliminate the shadows associated with it. Such an image can be used to rate the vehicles based on their type.

**3.9. Vehicle Recognition.** As part of the vehicle recognition[37], this phase plays an important role. Furthermore, included is a machine learning algorithm called delta rule (Widrow-Hoff learning rule). Using this rule, the vehicle type was identified. Once the vehicle type was identified, it was saved in the database.

**3.10. License Plate Detection and Log Generation.** In the back end of the application, a log is generated that can be used in the future and is also available when searching for a vehicle detail. License plate detection is also done in this phase using text detection.

Detection of vehicles and elimination of shadows are challenging aspects of computer vision. This work extends to the classification of vehicles and to the generation of logs. Moreover, they provide the user with traffic forecasts and alternative routes or ways to reach specific areas or places in different regions. Using IOOPL algorithm to detect and eliminate shadows, avoid accidents, minimize congestion, predict traffic, minimize traffic congestion, predict traffic, etc. In this work, edge detection, shadow elimination, and object detection is accomplished.

Developing a good shadow elimination algorithm is challenging for many reasons:

- (1) The system must be able to withstand changes in illumination
- (2) The algorithm should not detect nonstationary objects such as falling leaves, rain, snow, and shadows cast by moving objects
- (3) This should react to any changes made to the static background

**3.11. IOOPL Algorithm.** A work in this field implements the IOOPL filtering algorithm (Figure 2), in which a boundary is split into multiple boundaries and the object (vehicle) is detected. As part of the IOOPL algorithm [4], shadow elimination[34] and vehicle detection are combined in a subprocess that involves edge detection, background removal, and denoising the image. As well as the vehicle type, the time and the place at which the vehicle crossed a highway or



TABLE 1: Summary of ANPR system techniques.

From	Extraction	Procedure segmentation	Recognition
Slimani, I. et.al (2019) [20]	Otsu adaptive thresholding, CCA, edge detection-Canny	Closed curves	Template matching
Kashyap, A. et.al (2018)	Edge statistics and morphology techniques	Bounding box	Template matching
Vaishnav, A. et.al (2018) [21]	Morphology techniques	Region props bounding box using MATLAB	Template matching
Laroca, R. et.al (2018) [22]	Object detection, CNNs (YOLO detector)	Character segmentation CNNs, bounding box	Data augmentation, distant CNNs for letter and digits
Yogheedha, K. et.al (2018) [24]	—	Preprocessing techniques, Otsu's thresholding	Bounding box feature and template matching OCR
Ashtari et.al (2014) [25]	Color features (Hue and shape) with vertical/slope sweep	Image processing filter (histogram, Laplacian, morphology), connected component/projection analysis	Decision tree and SVM

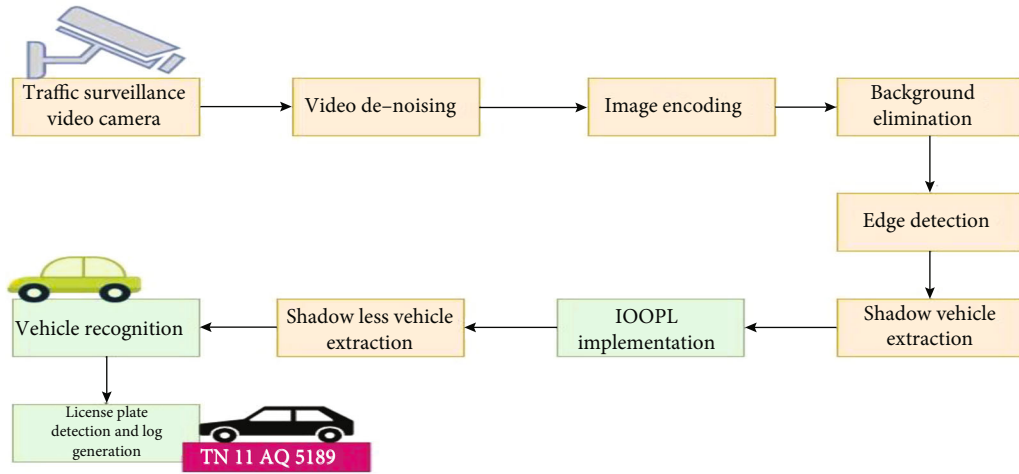


FIGURE 1: Workflow of the system.

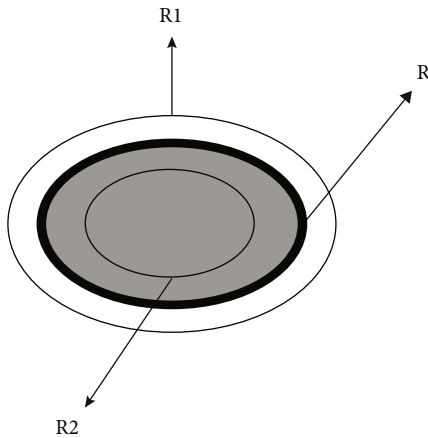


FIGURE 2: IOOPL multilayered boundary.

region can also be detected and logged in the database. The advantage of the ITS is traffic prediction and also facilitates in finding out the perfect place to lay a road and build a fly-over or bridge.

This work ensures image accuracy by detecting the edges and by eliminating the shadows[35]. This algorithm detects objects (vehicles) and eliminates their shadows. Shadow areas within an image can be recovered using an algorithm that uses IOOPL matching. Both shadow and non-shadow areas lying near the shadow boundary can represent the same type of object. Shadow boundaries can contract when they are contracted and expand when they are expanded. The inner and outer outline profile lines are generated along the inner and outer outline lines to determine the radiation features of the same type of object on both sides. As illustrated in Figure 2, R is the vector line that defines the shadow boundary. In the unshadowed view, R1 represents the outer perimeter and R2 represents the inner perimeter after contracting R inward. There is a one-to-one correspondence between R1 and R2. The correlation between R1 and R2 indicates that this location belongs to the same type of object.

By collecting the grayscale values of the corresponding nodes along with R1 and R2, IOOPL[32] is calculated. A shadowed outer profile line (OPL) is considered an inner

one; a nonshadowed one is considered an outer one. Normally, the objects that are linked to the building on both sides of the shadow boundary are not homogeneous. The inner and outer surfaces of the outline should also be ruled out for possible abnormalities. Similarity matching must be applied to the IOOPL section by section to rule out the two types of nonhomogeneous sections discussed previously. Shadow removal factors can be determined by examining grayscale distributions within and between homogeneous IOOPL sections.

Below is a pseudocode illustrating how IOOPL works:

First, images are acquired from video surveillance cameras, and whether they are captured by a static camera is also checked. Overlapping shadows can cause objects to be incorrectly detected.

The video is smoothed using Gaussian smoothing in the second step. By smoothing out the edges of an object, we can obtain a detailed outline that is then multilayered to determine the object's exact shape. Next, the object is divided into many levels of boundaries as illustrated in Figure 2. This method divides an object into layers for detection of the exact boundaries and allows the elimination of shadows from the object as a result of which the object has its exact shape.

Lastly, an object matching process is applied on overlapped objects to determine whether they are a part of the detected object or not. Resulting extra parts of the acquired objects will be eliminated as background objects if they are not part of the detected objects.

Figure 2 shows an example of an IOOPL multilayered object. In this case, R1 is the no shadow outline after expanding R outward, while R2 is the inner outline after expanding R inward. Even though the objects are linked together in one picture, this allows it to be distinguished into three parts R1, R2, and R3.

In computer vision, the hardest thing is to eliminate shadows and to classify vehicles. It is extended to license plate recognition and log generation in the proposed work. Shadows can be eliminated using IOOPL [4] (inner-outer outline profile), which uses different boundaries to mark the object, and these boundaries help to distinguish between background, shadow, and objects. Pixel values also serve as a means of separating the objects in the background from those in the foreground. Foreground objects are detected by canny edge detection method, and background objects are detected by threshold frequency. Moreover, they provide the user with traffic forecasts and alternative routes or ways to reach specific areas or places in different regions. This work proposes vehicle classification, license plate recognition, and traffic prediction to avoid accidents, as well as the generation of logs for later use. Based on the following algorithms, we can classify vehicles and recognize license plates.

**3.12. Delta Algorithm.** In Figure 3, it explains the clear understanding of how the delta learning rule is training the system, and it also explains the prediction of vehicles using the trained dataset in Figure 4.

To process the data, this hidden layer was created because perceptron rules[43] were not easily applied across

```

Begin
[1]: Obtain video sequence
[2]: Compute Gaussian smoothing ( $\alpha=2$ ,  $n=11$ )
[3]: Decrement the video into subsections/parts
[4]: For video
[5]: Similarity matching is done for each object
[6]: End For
[7]: If co-relation is high
[8]: Image merges
[9]: Else
[10]: Images need to be separated
[11]: End if
[12]:End

```

ALGORITHM 1: IOOPL algorithm.

multiple layers. Delta learning is an alternative to perceptron learning. By the name of LMS (least mean square), this rule was developed by Widrow and Hoff. This method is most commonly used to produce accurate results. For instance, if there is a difference of zero between the output vector and the correct answer, no learning will occur; otherwise, the weights will be adjusted to reduce it. It is determined by the following formula:  $\Delta w_{ij} = r * a_i * e_j$ , where  $r$  represents the learning rate,  $a_i$  represents the activation of  $u_i$ , and  $e_j$  represents the difference between the expected output and the actual output of  $u_j$ . The unsupervised competitive learning model [26] is one of these models.

The delta rule, also known as the gradient descent algorithm[45], is a machine learning algorithm for updating the input values of artificial neurons in a single layer neural network. It is a special case of the more general back propagation algorithm. For a neuron  $a$  with activation function  $g(x)$ , the delta rule for  $a$ 's  $b$ th weight  $w_{ab}$  is given by

$$\Delta w_{ab} = \alpha(t_a - y_a)g'(h_a)x_b \quad (1)$$

where

$\alpha$  is a small constant called *learning rate*

$g(x)$  is the neuron's activation function

$t_a$  is the output targeted

$h_a$  is the neuron's inputs (weighed input)

$y_a$  is the actual output, and

$x_b$  is the  $i$ th input.

It holds that  $h_a = \sum x_b w_{ab}$  and  $y_a = g(h_a)$ .

When a neuron has a linear activation function, the delta rule is commonly stated in a simplified form.

$$\Delta w_{ab} = \alpha(t_a - y_a)x_b \quad (2)$$

The delta rule's derivation[39] is different from that of the perceptron's update rule. As per the perceptron,  $g(h)$  is activated by the Heaviside step function. This means that  $g'(h)$  will not be at zero, and it will be equal to zero elsewhere, preventing it from being applied directly. A delta rule (DR) is also known as a perceptron learning rule (PLR) [27] and only applies to continuous activation functions. Widrow-Hoff learning rule is another name for this delta

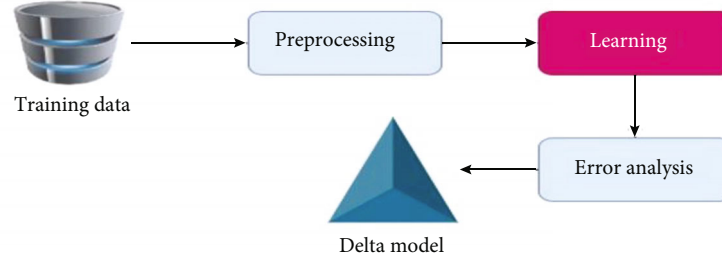


FIGURE 3: Flow of learning process using delta learning model.

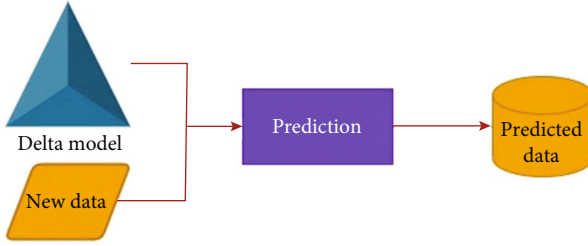


FIGURE 4: Flow of prediction using delta algorithm.

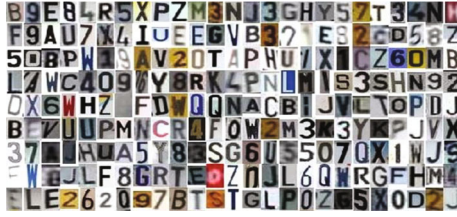


FIGURE 5: Alphanumeric values.

rule. The perceptron learning rule (PLR) has some similarities; however, there are some differences:

- (1) It is not necessary that the error ( $\delta$ ) in DR have values of 0, 1, or -1, as it is in PLR, but it may also have any other value
- (2) PLR works for threshold output functions only, but DR works for any differentiable function  $f$

Note that the rule will be different for not linear  $f$ .

*Step 1.* Add at least 10 images of vehicles, along with the model and make of the vehicle to your Neuron table.

*Step 2.* Based on the vehicle information (e.g., means, color distribution, and shapes), assign a weight to groups of vehicles of the same type.

*Step 3.* Use several input vehicles to train the table, utilize the already stored information, and calculate the weight values corresponding to each output.

*Step 4.* As you train, you should adjust the weighting function to match the target vehicle value.

*Step 5.* Incorporate the neuron activation function into weight values, so that each time the input vehicle is correlated with the neural weight value, the corresponding neuron functions are triggered.

**3.12.1. Text Correlation.** To match the input with the learned database, one uses a text correlation technique[44]. Both license plate recognition and vehicle classification use correlation techniques.

A delta learning rule is used in vehicle classification to learn the type of vehicle. By applying the delta rule, weights are also assigned by considering the vehicle's perspective, length, and model when the vehicle is captured either in the still or as a video; the weight is calculated by recalculating the image and then comparing that weight to the database to determine its exact weight.

Databases learn the set of alphanumeric values corresponding to license plate[33] formats, and when a license plate is detected simultaneously, the alphanumeric characters are compared to the characters in the database. Figure 5 shows a set of alphanumeric values that are learned by the database

#### (1) Correlation Steps in Classification.

- (1) Use Otsu's gray threshold method to remove the background
- (2) Use Frame Subtraction to detect moving objects
- (3) Use horizontal and vertical scanning to determine the inner and outer profiles
- (4) Use cross-correlation to determine the degree of match between the input vehicle and the database vehicle
- (5) The maximum matched index is updated in the rule and in the log after comparing all DB images with the current segmented vehicle based on correlation

#### (2) Correlation Steps in License Plate Recognition.

- (1) Segmenting vehicle using edge dilation and preprocessing
- (2) Apply color filters to the segmented vehicle to find the yellow pixels
- (3) Calculate the difference between car and license plates using the linear distribution area algorithm. This is due to the bordered license plate area

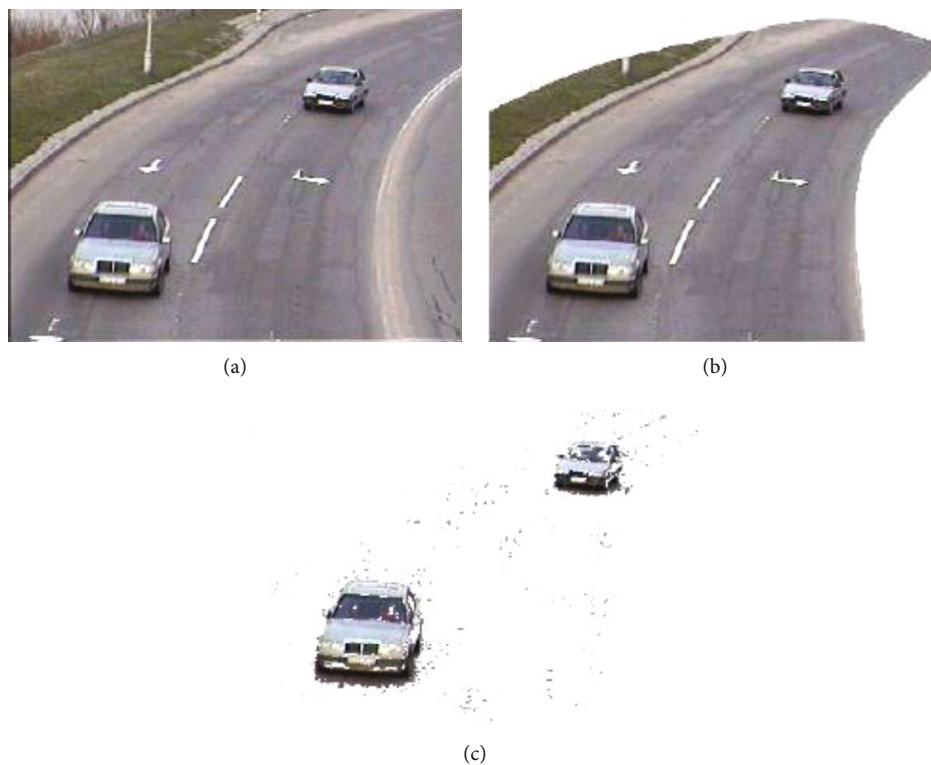


FIGURE 6: (a) Image with background, (b) IOOPL applied image, and (c) background eliminated image.

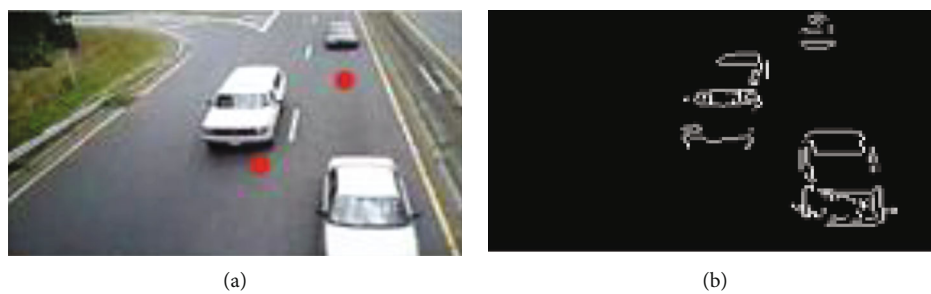


FIGURE 7: (a) Traffic image and (b) IOOPL applied image.

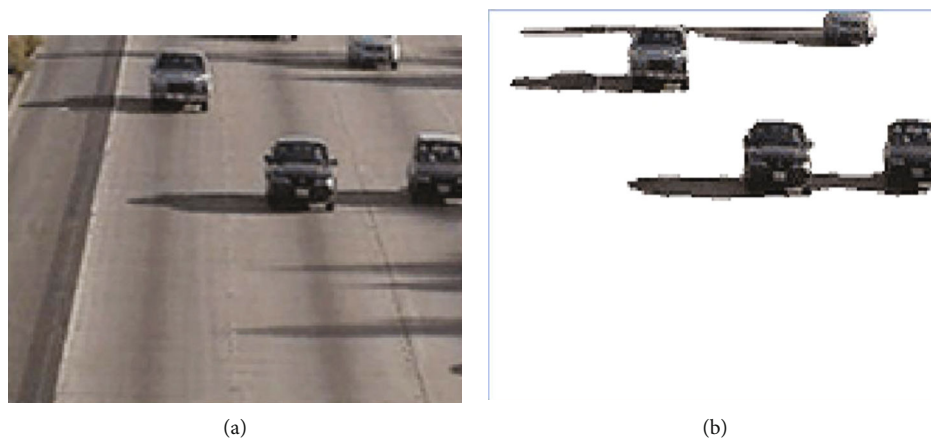


FIGURE 8: (a) Image Frame with shadowed image (b) object detection with shadow.



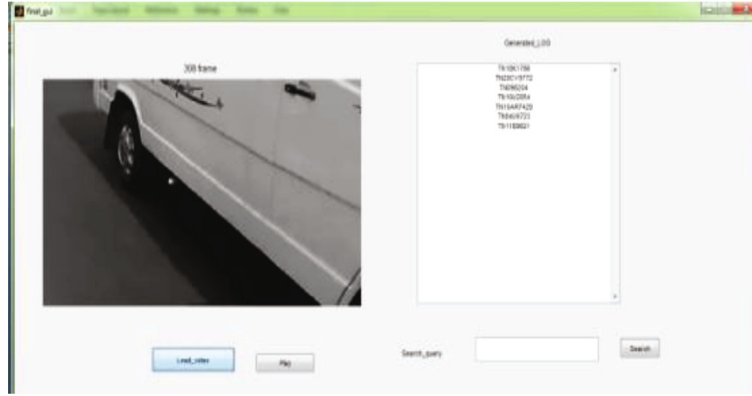


FIGURE 9: License plate detection in moving vehicles.

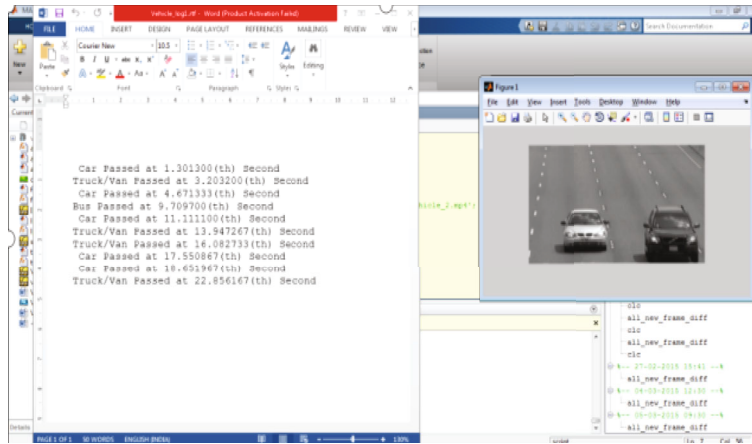


FIGURE 10: Vehicle classification and log generation.

TABLE 2: Image database.

Samples	Camera	Description	Average plate size	Tilt	Images
Sample set 1	Digital	Road entrance, day light, moving vehicles	30 × 70 pixels	15°	300
Sample set 2	Digital	Outdoor, daylight	30 × 70 pixels	15°	450
Sample set 3	Analog	Outdoor, daylight, moving vehicles	30 × 70 pixels	30°	450
Sample set 4	Digital	Nonmoving vehicles	Various	15°	560
Sample set 5	Digital	Parking or road, daylight	Various	15°	600
Sample set 6	Digital	Shadow parking	30 × 70 pixels	25°	750
Sample set 7	Analog	Indoor, daylight	40 × 80 pixels	15°	800
Total number of images					3910

TABLE 3: Accuracy calculation for number plate detection.

Frames per video (approx.)	Images per video (approx.)	Number plate extracted	Accuracy
240	240	220	92%
300	300	240	80%
450	450	380	84%
560	560	450	80%

- (4) Separate the alphanumeric contents of the LP by horizontal scanning while you locate the LP area
- (5) Find the matching index value by using text correlation between the input image and the template image
- (6) With the help of a maximum matched index, construct a switch table to determine the ASCII value of an alphanumeric. Update the log file

**3.12.2. Edge Dilation.** The dilation technique detects and categorizes the actual objects in the input image based on the morphological process. Two types of data are passed into

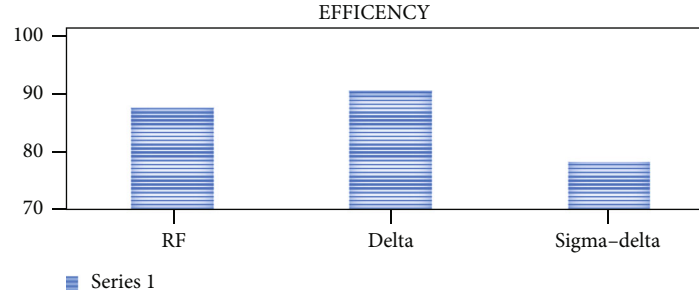


FIGURE 11: Comparison results.

the dilation operator. Dilating image begins with the image which needs to be enlarged. This second form is a (usually small) set of coordinates known as a structuring element, which determines how precisely the dilation effects the input image.

Using edge Dilation, you can detect where the edge lining is flowing in a digital image, as well as fill in missed (gaps) and noised edge linings. It is able to detect and extract objects from complex digital images in an environment with noise/movement. If you capture a moving vehicle using a still or moving camera, the edge linings of the input video frames are quite dynamic and disoriented. Edge dilation lets you morph those exact boundaries by removing unnecessary background objects.

The technique is used to detect the edges of license plates, where it is used in license plate recognition. Using dilation, the edges of the car and the number plate can be detected, allowing us to determine the number plate edges precisely.

In license plate recognition, the following steps are used to dilate the edge:

*Step 1.* Enlarge and color filter the segmented and filtered vehicle image, then apply the `imerode` MATLAB function to find the edge values on the eight directional edges.

*Step 2.* Follow the edges in all directions by using the region-growing technique.

*Step 3.* Text edges on license plates can be identified by the dilation value of the edges. Natural shapes dilate a great deal, while text shapes dilate less.

*Step 4.* Introduce threshold mechanism to find the minimum edge dilation value by using the area finding formula on the tracked vehicle edges.

In Figures 6 and 7, you can see the many phases of object detection and background elimination using the IOOPL algorithm to detect a particular vehicle in a highway traffic surveillance video by eliminating the other objects in the background. In Figure 8, the original image with background is shown. The initial focus is on two cars in the video; edges are detected as well as the background is removed simultaneously. Object edges are smoothed by Gaussian smoothing to identify the object type. According to the IOOPL algo-

rithm, if the correction result is high, then the two subsections belong to the same object and should be merged, else object shadows are present in the subsections and need to be separated. These images illustrate how the background is eliminated and objects are detected sequentially. In addition to monitoring traffic conditions, enhancing mobility, and reducing congestion, this research also enhances security in the transportation sector. Moreover, they assist users in drawing traffic forecasts [28] and in advising about departure times for different regions as well as alternative routes or ways to major locations. The aim of this work is to reduce congestion, predict traffic, avoid accidents, etc., using shadow elimination and vehicle classification.

Based on this research, we implemented IOOPL algorithm to detect particular vehicles[36] in highway traffic surveillance videos by removing other background objects, as well as delta learning algorithm to classify the vehicle types by learning their attributes. There are several phases to the process of detecting objects and removing background noise. Figure 9 shows that the user can load the video and that the license plate has been detected. Searching for a particular vehicle number has another advantage. In Figure 10, vehicles are classified and logs are generated.

The intelligent transportation system is not capable of classifying or logging vehicles. It is saved for further use later to generate the log and to detect license plates[46]. Searching for specific vehicles based on the license plate is one of the easiest methods. The main advantage of this work is that it helps monitor traffic conditions, reduce congestion, and improve mobility, as well as improve traffic security [29]. Moreover, they assist users in drawing traffic forecasts and in advising about departure times for different regions as well as alternative routes or ways to major locations. We propose the shadow elimination and vehicle classification here as a means of reducing congestion, predicting traffic, avoiding accidents, etc.

To demonstrate the accuracy of this work, real-time video has been used. This is shown in Figure 10. This work also allows vehicle tracking by recognizing license plates. To make vehicle tracking simpler, search query options are used.

## 4. Results and Discussion

In this section, the results of the experiments are discussed. The IOOPL algorithm is moreover used in satellite images and is applied for traffic surveillance to capture videos in this

work. Using the proposed IOOPL algorithm, the shadow of the image is maimed by detecting the edges of the object. Delta learning rule is used for vehicle classification and to give an optimized type of vehicle. Text detection is used to identify a license plate in real-time video. Table 2 shows the experimental environment, vehicles to be observed, plate size, and number of images extracted per sample. Table 3 shows the accuracy percentage for number plate detection based on the frames per video

The datasets consist of real-time video samples with respect to the parameters as shown in figure. The software tool used for implementation is MATLAB, which is one of the important image processing tools

As a result, this shows that the implemented delta algorithms have 92% of accuracy in classifying the vehicle shown in Figure 11. It shows the comparison results of the classification of vehicles from the training datasets.

## 5. Conclusions

A vision-based traffic monitoring system for highway road safety analyzes vehicles in video sequences and eliminates moving shadows, which is important for estimating traffic parameters. Although the models are typically used in traffic scenes of highways, they can also be used in other types of traffic scenes. Through their integration into vision systems, these algorithms will help in improving traffic parameter estimation and vehicle tracking due to the satisfactory results they produce in suppressing shadows and detecting vehicles; on the other hand, they are low cost and highly accurate, which will help in improving traffic parameters estimation and vehicle tracking due to their low cost. Highway traffic surveillance performs the critical job of vehicle detection and classification in ITS. Any outdoor environment can be used with the proposed model. In moving cast video, license plates are recognized by a combination of text correlation and edge dilation methods. Additionally, there is a log produced in a read-only format so that later use can be made. Shadow detection, vehicle classification, license plate recognition, and logging are excellent examples of how these approaches provide satisfactory results, as well as being highly cost-efficient and accurate. In other words, integrating these algorithms and methods in ITS can improve its performance. Vehicle tracking is the primary use of license plate recognition. The video used to demonstrate the work's accuracy is a real-time video. In the future, we will try to enhance our proposed approach to focus on night surveillance, occlusion handling, 3D modeling, and tracking vehicles.

## Data Availability

The data used to support the findings of this study are real-time videos and are available from the corresponding author on reasonable request.

## Conflicts of Interest

The authors of this manuscript declared that they do not have any conflict of interest.

## Acknowledgments

This research has been funded by the National Yunlin University of Science and Technology, Douliu.

## References

- [1] R. Surendran, O. I. Khalaf, and C. Andres, "Deep learning based intelligent industrial fault diagnosis model," *CMC-Computers, Materials & Continua*, vol. 70, no. 3, pp. 6323–6338, 2022.
- [2] S. Rajendran, O. I. Khalaf, Y. Alotaibi, and S. Alghamdi, "MapReduce-based big data classification model using feature subset selection and hyperparameter tuned deep belief network," *Scientific Reports*, vol. 11, no. 1, p. 24138, 2021.
- [3] M. Rajalakshmi, V. Saravanan, V. Arunprasad, C. A. Romero, O. I. Khalaf, and C. Karthik, "Machine learning for modeling and control of industrial clarifier process," *Intelligent Automation & Soft Computing*, vol. 32, no. 1, pp. 339–359, 2022.
- [4] H. Zhang, K. Sun, and W. Li, "Object-oriented shadow detection and removal from urban high-resolution remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 11, pp. 6972–6982, 2014.
- [5] S. N. Alsubari, S. N. Deshmukh, A. A. Alqarni et al., "Data analytics for the identification of fake reviews using supervised learning," *CMC-Computers, Materials & Continua*, vol. 70, no. 2, pp. 3189–3204, 2022.
- [6] P. Perez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proceeding of the European Conference on Computer Vision*, pp. 661–675, Copenhagen, Denmark, 2002.
- [7] A. H. Asaidi, A. Aarab, and M. Bellouki, "Shadow elimination and vehicles classification approaches in traffic video surveillance context," *Journal of Visual Languages and Computing*, vol. 25, no. 4, pp. 333–345, 2014.
- [8] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, no. 4, pp. 359–381, 2003.
- [9] N. A. Mandellos, I. Keramitsoglou, and C. T. Kiranoudis, "A background subtraction algorithm for detecting and tracking vehicles," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1619–1631, 2011.
- [10] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: real-time tracking of the human body," *Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.
- [11] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real time tracking," *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 246–252, 1999.
- [12] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151–1163, 2002.
- [13] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. S. Davis, "Real-time foreground-background segmentation using codebook model," *Real-Time Imaging*, vol. 11, no. 3, pp. 172–185, 2005.
- [14] Y. Xia, J. Zhi, M. Huang, W. Li, and R. Ma, "A simple on-road object segmentation approach in ITS system," in *2009 International Asia Conference on Informatics in Control, Automation and Robotics*, pp. 282–286, Bangkok, Thailand, 2009.

- [15] Liangli, "License plate detection method using vertical boundary pairs and geometric relationships," *IEEE Conference*, 2010.
- [16] T. Bai, K. Sun, S. Deng, D. Li, W. Li, and Y. Chen, "Multi-scale hierarchical sampling change detection using random forest for high-resolution satellite imagery," *International Journal of Remote Sensing*, vol. 39, no. 21, pp. 7523–7546, 2018.
- [17] M. R. Hidayah, I. Akhlis, and E. Sugiharti, "Recognition number of the vehicle plate using Otsu method and K-nearest neighbour classification," *Science Journal Information*, vol. 4, pp. 66–75, 2017.
- [18] G. Suryanarayana, K. Chandran, O. I. Khalaf, Y. Alotaibi, A. Alsufyani, and S. A. Alghamdi, "Accurate magnetic resonance image super-resolution using deep networks and Gaussian filtering in the stationary wavelet domain," *IEEE Access*, vol. 9, pp. 71406–71417, 2021.
- [19] A. R. F. Quiros, R. A. Bedruz, A. C. Uy et al., "A kNN-based approach for the machine vision of character recognition of license plate numbers," in *TENCON 2017 - 2017 IEEE Region 10 Conference*, pp. 1081–1086, Penang, Malaysia, 2017.
- [20] C. Kavitha, X. Anita, and S. Selvan, "Improving the efficiency of speculative execution strategy in hadoop using amazon elasticache for redis," *Journal of Engineering Science and Technology (JESTEC)*, vol. 16, no. 6, pp. 4864–4878, 2021.
- [21] I. Slimani, A. Zaarane, A. Hamdoun, and I. Atouf, "Vehicle license plate localization and recognition system for intelligent transportation applications," *Proceedings of the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, 2019, pp. 1592–1597, Paris, France, 2019.
- [22] A. Vaishnav and M. Mandot, "An integrated automatic number plate recognition for recognizing multi language fonts," in *Proceedings of the 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pp. 551–556, Noida, India, August 2018.
- [23] N. A. Khan, O. I. Khalaf, C. A. T. Romero, M. Sulaiman, and M. A. Bakar, "Application of intelligent paradigm through neural networks for numerical solution of multiorder fractional differential equations," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 2710576, 16 pages, 2022.
- [24] R. Laroca, E. Severo, L. A. Zanlorensi et al., "A robust real-time automatic license plate recognition based on the YOLO detector," in *Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, Rio de Janeiro, Brazil, July 2018.
- [25] K. Yogheedha, A. Nasir, H. Jaafar, and S. Mamduh, "Automatic vehicle license plate recognition system based on image processing and template matching approach," in *Proceedings of the 2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA)*, pp. 1–8, Kuching, Malaysia, August 2018.
- [26] R. Javed, S. Ur Rehman, M. U. Khan, M. Alazab, and T. Reddy, "CANintelliIDS: detecting in-vehicle intrusion attacks on a controller area network using CNN and attention-based GRU," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1456–1466, 2021.
- [27] N. Deepa, Q.-V. Pham, D. C. Nguyen et al., "A survey on blockchain for big data: approaches, opportunities, and future directions," 2020, <https://arxiv.org/abs/2009.00858>.
- [28] V. Mani, C. Kavitha, S. S. Band, A. Mosavi, P. Hollins, and S. Palanisamy, "A recommendation system based on AI for storing block data in the electronic health repository," *Frontiers in Public Health*, vol. 9, p. 831404, 2022.
- [29] T. R. Gadekallu, Q. V. Pham, D. C. Nguyen et al., "Blockchain for edge of things: applications, opportunities, and challenges," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 964–988, 2022.
- [30] B. Qin, Z. Ma, Z. Fang, and S. Wang, "Fast Detection of Vehicles Based-on the Moving Region," in *10th IEEE International Conference on Computer-Aided Design and Computer Graphics*, pp. 202–207, Beijing, China, 2007.
- [31] H. Kuo, J. Lee, and S. Kao, "An Autonomous License Plate Detection Method," in *Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 110–113, Kyoto, Japan, 2009.
- [32] P. Auer, H. M. Burgsteiner, and W. Maass, *The p-Delta Learning Rule for Parallel Perceptrons*, Institute for Theoretical Computer Science, 2001.
- [33] Y. Li, P. Gong, and T. Sasagawa, "Integrated shadow removal based on photogrammetry and image analysis," *International Journal of Remote Sensing*, vol. 26, no. 18, pp. 3911–3929, 2005.
- [34] S.-C. Chen, M. L. Shyu, and C. Zhang, "An Intelligent framework for spatio-temporal vehicle tracking," in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, pp. 213–218, Oakland, Calif, USA.
- [35] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, pp. 179–187, 1962.
- [36] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara, "Detecting moving shadows: algorithms and evaluation," *Machine Intelligence*, vol. 25, no. 7, pp. 918–923, 2003.
- [37] Y. Ding, B. Banitalebi, T. Miyaki, and M. Beigl, "FTraffic: a study of passive traffic awareness using emitted RF noise from the vehicles," *EURASIP Journal on Wireless Communications and Networking*, vol. 1, 2012.
- [38] H. Asaidi, A. Aarab, and M. Bellouki, "Shadow detection approach combining spectral and geometrical properties in highway video-surveillance," *International Journal of Computers and Applications*, vol. 53, no. 17, pp. 40–44, 2012.
- [39] H. Lin, J. Zhao, S. Li, and G. Qiu, "License plate location method based on edge detection and mathematical morphology," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 853–857, Chongqing, China, 2020.
- [40] A. H. Ashtari, M. J. Nordin, and M. Fathy, "An Iranian license plate recognition system based on color features," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 4, pp. 1690–1705, 2014.
- [41] T. R. Gadekallu, D. S. Rajput, M. P. K. Reddy et al., "A novel PCA-whale optimization-based deep neural network model for classification of tomato plant diseases using GPU," *Journal of Real-Time Image Processing*, vol. 18, no. 4, pp. 1383–1396, 2021.
- [42] P. Durgadevi and S. Selvaraj, "Resource allocation in cloud computing using SFLA and cuckoo search hybridization," *International Journal of Parallel Programming*, vol. 48, no. 3, pp. 549–565, 2020.
- [43] C. Kavitha, V. Mani, S. R. Srividhya, O. I. Khalaf, and C. A. Tavera Romero, "Early-stage Alzheimer's disease prediction using machine learning models," *Frontiers in Public Health*, vol. 10, p. 853294, 2022.
- [44] A. R. Khaparde, F. Alassery, A. Kumar et al., "Differential evolution algorithm with hierarchical fair competition model," *Intelligent Automation & Soft Computing*, vol. 33, no. 2, pp. 1045–1062, 2022.