

## Deep Learning Based Vehicle Number Plate Detection Based on Advance Sequential Long Short-Term Memory with Convolutional Neural Network



Sree Southry Singaravelu\*, Sabeenian Royappan Savarimuthu

Department of Electronics and Communication Engineering, Sona College of Technology, Salem 636005, India

Corresponding Author Email: sreesouthry@sonatech.ac.in

Copyright: ©2024 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/ts.410620

## ABSTRACT

Received: 27 February 2024 Revised: 13 August 2024 Accepted: 29 August 2024 Available online: 31 December 2024

#### Keywords:

vehicle plate, segmenting the images, enhanced feature scaled social spider optimization, advance sequential long short-term memory with convolutional neural network, feature weights Day by day, increasing roadside unit vehicles create more traffic, accidents, high speed, theft, and serious problems. Identifying the number plates automatically in vehicle boards is difficult because various angles of projection, number plate types, positions, and character styles are tough. Many existing systems formalize automatic number plate recognition systems based on computer-aided solutions with image processing support. The video is fragmented, and pictures are caught at the right point with appropriate lighting and clarity, and standard textual styles are improperly handled. The point is to plan a proficient robotized authorized vehicle recognizable identification system utilizing vehicle plates by analyzing features. Due to increasing pixel intensity noise illumination during segmentation, feature scaling creates more dimension, leading to improper detection accuracy. By addressing this problem, they proposed an Advance Sequential Long Short Term Memory method with a Convolutional Neural Network (ASLSTM-CNN) approach for a vehicle number plate detection and recognition method that can help detect number plates of vehicles. Initially, the number plate video frames will be collected and converted into images from the Standard UCI repository for training and testing, classification, and detection of the images. The next step is pre-processing the images using Sobel's filtering method; canny filters can help reduce the images. Use the Sobel method to find the approximate absolute gradient scale for each point in the image. The canny filter method can detect each edge first, reducing the noises from the images and finding the images to detect the gradient regions. The second step is segmenting the images using enhanced region-based Convolutional Neural Segmentation (ER-CNS) for segment input images based on the areas and then extracting the features based on the segmenting Region using Enhanced Feature Scaled Social Spider Optimization (EFS3O) analysis of the feature weights based on its threshold values and evaluating the maximum support range. ASLSTM-CNN uses the SoftMax Neural Network (ASLSTM-CNN-SN<sup>2</sup>) to recognize the image region and check the layers estimations. Finally, characters are identified by ASLSTM-CNN; each feature can be efficient in evaluating the images and improve the detection accuracy by up to 95.6%, with a precision rate of up to 9.1% best rate which is better than previous approaches.

#### **1. INTRODUCTION**

With increasing vehicles, modern cities must install effective and efficient automated traffic systems to manage traffic restrictions. In this case, number plate recognition plays a vital role. License plate recognition is an image processing technology that uses digital cameras, color or grayscale digital cameras, and infrared cameras to extract video frames from a vehicle's license plate image. It identifies a car using its number plate. Image processing is a crucial technique used in LPR recognition systems. Developing a Number Plate Recognition (NPR) system using image processing is challenging due to its limited ability to handle multiple scales. This is because LPR video frames for images may look dirty, have motion blur, low resolution, poor lighting, low contrast, etc.

Number plates may fade, and movement may appear blurry.

Number Plate Recognition consists of five main stages. They initially used localization techniques to detect and isolate the number plates in the input images. It is followed by plate orientation, which adjusts the plate tilt, and resizing, which changes the dimensions to the desired size. Image normalization is then performed to adjust the image brightness and contrast. Letter separation aims to separate individual letters from number plates. Character in image detection makes detecting license plate recognition systems difficult. Character identification determines the success or failure of character splitting and recognition. In most cases, convolutional neural networks (CNNs) have attracted much research interest in recent years due to their many capabilities in solving detection, optical character recognition, and classification problems.

The number plate recognition system combines various technologies and mechanisms such as image pre-processing,

target detection, character segmentation, and recognition to recognize number plate characters. It consists of a camera that detects number plate objects and a processing unit that processes and extracts characters and interprets pixels into digitally readable characters. Automatic Number Plate Recognition (ANPR) systems are used in traffic enforcement, including speed cameras, traffic light cameras, stolen vehicle detection, and border patrol. It can also be used for building management, such as parking lot management and access control.

ANPR assumes a significant role in video vehicle observation and frameworks, such as leaving-the-board frameworks, cost installment handling frameworks, and frameworks that require confirmation. Mechanizing cycles can save time for security faculty. Tag recognition can lead to many issues, like the vast number of vehicles on a regular street. The vehicle types incorporate vehicles, cruisers, bikes, vans, transports, auto carts, SUVs, smaller-than-expected trucks, vans, and farm haulers. Each class has an alternate tag shape and style. The utilization of different textual types and altered sheets added to the intricacy. Plates come in various shapes and sizes. This implies that not all plates are rectangular; some are trapezoidal, and some are irregular.

Figure 1 shows Many applications for automatic number plate recognition systems, calculating programming cost categories for toll booths, traffic monitoring, line control, vehicle location acquisition, programming vehicle tagging, and access control. The working process of the Automatic Number Plate Recognition System consists of 4 main phases: 1) image pre-processing and segmentation, 2) Feature Extraction, 3) Character Segmentation and extracting the plate extraction, and 4) Character Recognition phase for classification. They were finally detecting the plate number detection.

This research uses a deep feature selection and classification model to improve character detection accuracy. The laver of the ASLSTM-CNN model is primarily used to detect text regions that occur in the input image. Apply a classification model to differentiate number plates from familiar characters. Advanced Sequential Long-Short-Term Memory with Convolutional Neural Networks (ASLSTM-CNN) is considered a unique architecture of artificial neural networks currently used by researchers in Number Plate Recognition (NPR) systems. We propose an integrated approach that integrates the segmentation and recognition steps using an ASLSTM-CNN that directly manipulates image pixels. Technical advice for image acquisition, image pre-processing, number plate detection, text segmentation, and recognition. Recognition results show a significant improvement in recognition performance after classification compared to existing classification models.



Figure 1. Block diagram

#### 2. RELATED WORKS

Systems for detecting number plates (LPD) are indispensable in many traffic-related applications. Gaussian filters enhanced cumulative histogram-based techniques and contrast-limited adaptive histogram equalization techniques are a few suggested approaches [1]. Computer Vision (CV) and Deep Learning (DL) techniques are crucial to improving ANPR and meeting the objectives of Intelligent Transport Systems (ITS). A new ANPR pipeline based on deep Learning that can be used with heterogeneous number plates and an intelligent vehicle access control system that considers various plate shapes and styles are available in many Asian and European nations [2].

Due to fluctuations in viewpoint, shape, color, multipatterns, and sporadic lighting conditions during picture collecting, the Vehicle Number Plate Identification (VLPI) procedure is complex [3]. Monitoring based on physical traffic enforcement cannot simultaneously track damages and monitor such high traffic [4]. The size of the number plate, which fluctuates depending on how close the car is to the camera, is one of the main issues these systems encounter. Systems typically employ single-scale detectors in arrays within picture pyramids to solve this issue [5].

It can be required to operate, for instance, on a mobile device or cloud server or in dim light or inclement weather. To address these requirements, many number plate recognition methods have been created [6]. Advanced image processing algorithms and genetic algorithms (GA) based on improved neutrosophic synthesis (NS) have been used to offer a novel method for recognizing Licence Plates (LPs) [7].

A popular method for automatically acquiring car number plates using artificial vision is number plate recognition [8]. ALPR takes pictures using a color camera, a black-and-white camera, or an infrared camera. In a practical setting, ALPR must process number plates swiftly and effectively no matter the weather, whether indoor, outdoor, day or night [9]. Algorithm adjustments are necessary for these ALPR systems to function with LPs in other nations. A prior study on crossborder LP authentication examined data from several countries using the same LP system [10].

This technology deals with unclear number plates, shifting

weather and lighting circumstances, shifting traffic conditions, and fast-moving cars [11]. Numerous applications exist for automatic number plate identification, essential to developing intelligent transportation systems. However, most recent work on recognizing number plates uses pictures of the front of the plate. It is challenging to recognize number plates in natural settings and during impromptu eyesight [12]. It was often 35% or less of number plates that two different platforms could accurately detect and match [13]. Available methods to enhance number plate matching for systems that recognize number plates rely entirely on manual data reduction, wherein incorrect number plates are manually inserted.

The approach described here uses an altered templatematching algorithm to analyze target color pixels to locate a number plate [14]. These methods cannot handle complex real-world capture conditions, such as varying lighting or oblique camera angles. A robust number plate detection network is suggested to increase the robustness of identifying number plates in difficult capture [15]. Most methods only function in an acceptable range of circumstances, like fixed lighting, restricted vehicle speed, predetermined routes, and fixed training. Many LPR approaches were developed in still photos or video sequences [16].

An efficient segmentation network-based multimodal technique for reading license plates that converts the segmentation and optical character recognition steps from traditional methods to object event detection [17]. Current target detection and number plate recognition methods and an improved YOLOv5m and number-based LPRNet model were developed, considering the requirements of number plate recognition systems for recognition accuracy and real-time performance in the current complex scenarios it is being designed. It was suggested that plate recognition be used [18].

Currently, most techniques under controlled conditions positively affect recognizing number plates, and most such number plates are photographed at favorable viewing angles and lighting conditions [19]. There are four observations on how ALPR was created: the layer structure based on resampling improves accuracy and speed [20].

A mathematical model based on the BLEU score operator is suggested to enhance the detail of edge information in photos of number plates and the effectiveness of text detection and identification techniques [21]. Traditional location-aware algorithms are unsuitable for real-world applications because they are sensitive to lighting, shadows, background complexity, and other elements. Profound learning advancements have made it possible for algorithms that recognize number plates to extract deeper characteristics, dramatically increasing the accuracy of detection and recognition [22].

A novel mathematical model to enhance the detail of edge information in photos of number plates and enhance the effectiveness of text detection and identification systems [23]. Traditional location-awareness algorithms are not ideal for actual applications because they are quickly influenced by lighting, shadows, background complexity, or other factors [24]. Profound learning advancements have made it possible for algorithms to recognize number plates to extract deeper characteristics, considerably increasing the accuracy of detection and recognition [25].

Vertical edge detection, morphological adjustments, and other validations are used to carry out Number Plate Detection (LPD). The number plate candidate character regions are created by extracting the character-specific ERs [26]. Despite recent excellent performance, two challenges still need to be solved in most related studies due to the rapid development of Deep Learning (DL)-based methodologies.

License plate detection is performed with vertical edge detection and morphological operations. Then, remove the potential letter region of the license plate and include the specific character [27]. End-to-end irregular Number Plate Recognition (EILPR) is one of the two challenges that still need to be addressed in most relevant studies despite their recent high performance [28] and the rapid growth of deep learning (DL)-based algorithms.

Due to many factors, detecting the vehicle number plate and identifying the letters inside the car can take time and effort. These circumstances include difficult situations, including erratic lighting and weather, noise from data capture that cannot be avoided, and real-time performance demands of cutting-edge Smart Transportation System (STS) applications [29]. Automated number plate identification is crucial for many applications connected with intelligent transportation systems. The majority of the currently used techniques concentrate on specific methods (such as toll management) or single Number Plate (LP) zones, which restricts their usefulness [30].

The study [31] used deep learning-based CNNs to improve home alarms and remove unwanted signals using CCTV cameras. This method often needs significant computation and memory, which restricts its use in surveillance networks.

The paper [32] used ML, DL, and AI cameras to count people entering and exiting areas. A centroid tracking and detection method estimated the number of people and their direction. This algorithm replaces manual security and traffic management in stores using computer vision and DL. Likewise, the DL-based CNN algorithm was developed by [33] for printed circuit board (PCB) defect identification. However, Long-lasting PCB defects are a growing concern.

Counting vehicles on busy roads helps authorities gather data for better traffic management [34]. However, managing traffic and delivering enough parking takes much work. The authors [35] focused on suggested crops for different soil nutrient levels with specific fertilizer suggestions. However, there are challenges in selecting, cultivating, and fertilizing for high yields. The novel [36] modified CNN and pseudo-CNN methods to eliminate the noise in the digital images. The method rejects distorted images and effectively removes impulse noise in digital images. The study [37] discusses automatic vehicle number plate detection using Mirrored EAST, which improves localization performance by utilizing the distance between an image and its mirrored counterpart. The study [38] introduced a DL-based scheme using MobileNet-V2 and YOLOx for vehicle identification and number plate detection, achieving an efficient detection rate. A study [39] analyzed a car-following dataset utilizing ML methods for automated vehicle identification, extracting trajectories and traffic streams. Obtaining effective training models for multiple vehicle identification in real traffic scenarios is challenging [40].

## 2.1 Problem identification factors

From the review, the problem was identified and considered as follows:

• Existing methods failed to concentrate highillumination images in real-time entity progression to character patterns.

- Complex images with different sizes, backgrounds, camera angles, and distances cause pixel variation and angle tendencies, and character projects are not segmented.
- Taking higher feature dimensions and scalar values degrades the segmentation process, leading to poor accuracy.
- Mutual dependencies of character patterns and edge mapping are non-scalable, which reduces detection accuracy. By the feature dimension, lower precision rate, recall rate, f1, and dice coefficient formation degrade the detection accuracy.

## **3. MATERIALS AND METHOD**

Deep learning-based Automatic Number Plate Recognition is an image processing method detecting the Vehicle's number plate using an ASLSTM-CNN sliding over an image to detect characters. The Number plate characters extraction module follows the ASLSTM-CNN method, sets the convolution kernel size to 1, and integrates different features without changing the feature size.



Figure 2. Proposed diagram

The proposed reduced the number plate feature size from  $(38 \times 14 \times 256)$  to  $(38 \times 14 \times 1)$ . Feature weighting optimizes the features of the license plate characters and extends the original license plate feature extractor. The Number plate character classifier transforms the extracted character feature dimensions into predictive categories and uses the SoftMax function to find license plate numbers. Figure 2 shows the proposed diagram using ASLSTM-CNN to detect the image's vehicle number plate video frames. Initially, vehicle frames to images were collected from the UCI repository, and the second stage was based on Sobel and canny filters for detecting the photos in the pre-processing step. Then, the images are segmented based on the Region using the ER-CNS and feature

weights, detecting the maximum support range based on the EFS3O algorithm. Before classification, the weights are evaluated based on the SoftMax Neural Network (SN2) using the layers estimation. Finally, classification using ASLSTM-CNN using the detection of the characters for vehicle plate images.

## 3.1 Capture vehicle video

To capture a vehicle video, the camera must be fixed at least 3 feet above the number plate and pointed at the number plate. The camera captures video from either the front or rear of the Vehicle. Vehicle videos include number plates. The camera sends input video to the computer. Cameras can be mounted at various positions. However, this method works best for 15-to 20-second video clips. A 20-s video has 480 frames/image (24 fps), and operations are performed on 240 images to extract number plates.

## 3.1.1 Convert video frames to images

Capture frames from a video and convert them into images. This change is necessary because converting video frames into images requires using a different algorithm to detect the number plate. A video of 50 seconds can generate 1,000 images. These images are then stored in an arbitrary folder and further processed to detect vehicle number plates.

### 3.2 Vehicle number plate image collection

The number plate detection dataset trains a model detecting license plate numbers in images collected from UCI repositories. A detected number plate is used for the next stage of number plate images. This Dataset consists of 1000 images of vehicle number plates annotated with bounding boxes.

#### 3.2.1 Dataset features

- Size of Dataset: 1000+
- Resolution: 100% HD images and higher (1920×1080 and higher).
- Places: Capture over 700 towns and villages in India.
- Range: Different light conditions like day, night, distances, and viewing angles.
- Applications: Number plate detection, ANPR, recognition, automated driving system, etc.

Figure 3 shows the vehicle number plate sample images with Boundary Box and image sizes. These images are saved as color JPEGs in the camera. Implementing the language Python and anaconda tool can be used as input to this system to process JPEG images of the vehicle number plate.



Figure 3. Sample images from dataset

#### 3.3 Sobel and canny filter for pre-processing

A pre-processing step is used to improve the number plate localization and letter segmentation performance. A camera captures an image of the vehicle with a 13-megapixel resolution, and the number plate number is recognized. Converts the image from grayscale to black and white, removing all objects smaller than 100 pixels.

$$I = N * \frac{T}{s} \tag{1}$$

where, I is the required images, N is the number of images, Ttime, and S-size images. The size of images can be 1200\*1600 or 270\*180. First, convert the RGB image to a grayscale image to extract the features of images more efficiently. In the reprocessing stage, there are two kinds of filters: one is Sobel, and the other one is a canny filter. The Sobel filter detects the edges and canny to reduce the noises from the images.

#### 3.3.1 Sobel filter

The Sobel filter method finds the edge using covers of  $3\times 3$  size. One estimates the slope in the *x* direction, and the other calculates the *y* direction. The mask slides over the image one square pixel at a time. This algorithm computes the image intensity gradient at each point, telling each point to increase in image intensity from bright to dark. Edges regions represent darker or lighter intensity variation.

A gradient is a vector whose elements measure how quickly pixel values change with distance in the x and y directions.

$$\frac{\partial i(a,b)}{\partial a} = \nabla a = \frac{i(a+da,b)-i(a,b)}{da}$$
(2)

$$\frac{\partial i(a,b)}{\partial a} = \nabla b = \frac{i(a+b+db) - i(a,b)}{db}$$
(3)

In separate images, db and da- distance along with a and b image edges can be considered db and da regarding the number of pixels between two points.

$$db = da = 1 \tag{4}$$

Pixel values coordinate are (x, y),

$$\Delta a = i(x+1,y) - i(x,y) \tag{5}$$

$$\Delta b = i(x, y+1) - i(x, y) \tag{6}$$

The Sobel convolution kernels are designed to accommodate vertical and horizontal edges. Image angles are calculated ( $\alpha$ ) is 0, and the direction of the maximum contrast from black to white moves from left to right across the image.

$$G_a = \frac{\partial i}{\partial a}; G_b = \frac{\partial i}{\partial b} \tag{7}$$

$$S(a,b) = |G_a| + |G_b|$$
 (8)

$$\alpha = \tan^{-1} \left[ \frac{G_a}{G_b} \right] \tag{9}$$

Each of these edges is associated with an image. Calculate the horizontal and vertical slopes  $(G_a \text{ and } G_b)$  and integrate

them to find the total size and direction of the slope at each point.

#### 3.3.2 Canny filter

The Canny edge detection algorithm is introduced to improve the edge detection process. The main goal is to reduce minor errors. Canny edge detection is an operator and an algorithm that helps detect sharp edges in noisy images.

Using filter for reducing the noise C(x, y), smoothening the images

$$C(m,n) = c\sigma(m,n) * f(m,n)$$
(10)

$$C(x, y)$$
 Is given by  $C\sigma(m, n) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left[-\frac{u^2+v^2}{2\sigma^2}\right]}$  (11)

Evaluate gradient of g(u, v)

$$x(m,n) = \sqrt{g_u^2(m,n) + g_v^2(m,n)}$$
(12)

$$\theta(m,n) = \tan^{-1} \left[ \frac{g_u(m,n)}{g_v(m,n)} \right]$$
(13)

Threshold limits (T) are given as:

$$T_{c}(a,b) = \begin{cases} c(m,n) \ if(a,b) > T \\ 0 \end{cases}$$
(14)

If (image pixel grade > higher Threshold)

Accept the pixels as the edges Else if (image pixel grade < lesser threshold) Discards the pixels

Else if (the image pixel grade is associated with pixels above the upper Threshold)

Accept pixels as edges.

T-Chosen, a, b – values of edges, (m, n)-along the edges, Canny uses two thresholds (upper and lower). Edge points detected by the operator must accurately locate the edge center. Canny is an essential method for isolating noise in an image and detecting edges before detecting edges in an image. The canny method is the best way to find edge values and thresholding without the edge features in the image.

## 3.4 Enhance region-based convolutional neural segmentation (ER-CNS)

RCNS combines CNN with region proposal segmentation to classify the image regions into detected vehicle number plates. Compared with traditional sliding window-based detection methods, selecting (recommended) regions can reduce the search space and thus reduce the detection.

The localization procedure numbers the pattern image to each location of the vehicle image. For each segment pixel, calculate the numerical index to ensure that the pattern matches the picture. Finally, the most significant similarities are identified as characters' valid images.

#### 3.4.1 Character segmentation

The most essential steps in license plate recognition for image segmentation. Without segmentation, a character could be split into two characters. Use the bounding box method to measure the features of the image region. Create a bounding box for all characters on the number plate to extract and recognize each character and number. Use the bounding box method to divide the exact area of each character. The character Segmentation process separates the number plate.

$$C(a) = \sum_{P \in \beta} vP * a_p + \sum_{\{aP\}} V^{pq} X |a_p - a_q| a_q$$
  

$$\in \{0, 1\}$$
(15)

where,  $\beta$  is an image with the pixel values of  $P \in \beta$ ,  $a_p$  individual pixel labels values of 0 and 1. Sets of the image pixels defined by  $\in vP$  and  $V^{pq}$ .

**Steps: Bounding box method for vehicle plate region** Begin

## For a = 1: n(each video segmentation), do

For  $b = 1: n_f$  (each segmenting frames), do

Recognized the character as center objects with target bounding box;

Image region  $(R_g)$  is a targeting boundary box;

Normalization Region of Interest (ROI)

Compute the segmenting of the images with normalized ROI using below equation.

$$C(a) = \sum_{P \in \beta} vP * a_p + \sum_{\{aP\}} V^{pq} X |a_p - a_q| a_q \in \{0,1\}$$

Normalized image values;

End

Collect the set of images edge point  $(\{V_i\}_{T=1}^{x_i})$ Compute feature vector  $(x_i)$ 

End

Training set  $T_s = (\{a_1, b_2\}_{x=1}^N)$ 

Train RCNS the classifier using  $T_s$  with validation for segmenting images

End

The steps are a bounding box algorithm that describes the object's (Vehicle's) position. A bounding box is a rectangular box defined by the x-axis and y-axis coordinates of the rectangle's upper-left corner and the x-axis and y-axis coordinates of the rectangle's lower-right angle.

3.4.2 Enhance region-based convolutional neural segmentation

The number plates are detected by passing each zone to the RCNN model. The Model predicts types and assigns confidence values to each Region. All areas predicted by the Model to be plate areas are selected with greater than 95% confidence of segmentation. The most accurate Region is chosen for all selected regions using the non-maximal suppression function. This part is considered the detected number plate part.

Evaluate the pixel of the images for each column

$$S_i = \sum_{i=1}^n s_{i,j}, j = 1, 2, \dots w$$
(16)

Detection of Region Segment Columns (RSC)

$$RSC = (sc \in s_i | s_i \le 1, j = 1, 2, ...w)$$
(17)

Initiate the m, sum=0  
For (n=1, SC,-1, n++)  
{  
If 
$$(sc_{n+1} - RSC_n \le TH)$$
  
 $sum = sum + sc_n, n = n + 1$   
Else  
If  $m \ge 1$  then  $sc_n = Region\left(\frac{sum}{n}\right)$   
Else  $sc_n = RSC_n$   
 $Sum = 0, n = 0$   
}

Each segmented column is estimated using a neural network to handle additional features and region segmentation of character extraction. Thus, the pixels of each segmented column are calculated, and the nearest neighbors of the segmented Region are analyzed.

## 3.5 Enhanced feature scaled social spider optimization (EFS3O)

In the proposed method, each solution in the search space represents a spider's position in the public network. All spiders are weighted according to the best or worst solution represented by the community spider. The algorithm models two search agents (spiders), male and female.

Depending on their gender, each is subject to a different evolutionary operator, reflecting cooperative behaviors commonly believed in populations. An attractive aspect of sociable spiders is that their population is heavily skewed toward females. The algorithm first defines the number of female and male spiders described as individuals in the modeling search space. The count of  $c_f$  females is randomly chosen within the *c* range of 75%~85% of the total population.

$$c_f = floor[(0.8 - random \ 0.25).c]$$
 (18)

 $c_f$  is evaluated by the random numbers between (0, 1), and the count of male spiders  $(c_m)$  is computed as the match between *c* and  $c_f$ .

$$c_m = c - c_f \tag{19}$$

The entire population (p), collected by N number of elements, is divided into two sub-fields, F and M.

$$(M = \{m_1, m_2, \dots, m_n\})$$
(20)

$$(F = \{f_1, f_2, \dots, f_n\}) \ p = F \cup M(p = \{p_1, p_2, p_3, \dots, p_n\})$$
(21)

$$p = \{p_1 = f_1, p_2 = f_2, \dots, p_{nf}\}$$
(22)

$$p = \{p_1 = m_{1_1}, p_2 = m_2, \dots, p_{n_m}\}$$
(23)

• Fitness evaluation

In the proposed method, each spider characteristic (c) receives a weight  $(w_x)$  that reflects the quality of the solution associated with spider  $s_i$  (irrespective of gender) in the population (p).

Calculate the weights of every spider

$$w_{x} = \frac{worst_{p} - s(c)}{worst_{p} - best_{s}}$$
(24)

s(c)-Fitness values obtained by the estimation of the spider position  $(f_s)$ , the values worst<sub>s</sub> and best<sub>s</sub> is considering the minimizing problems.

$$best_s = \min_{p \in \{1,2,\dots,N\}} \left( s(c_p) \right) \tag{25}$$

$$worst_s = \max_{p \in \{1,2,\dots,N\}} \left( s(c_p) \right) \tag{26}$$

### Spiders' vibrations through the web

The Vibrations depend on the weight of the spider and the distance from which the shock is generated. The distance between the spider that excited the vibration and the member that detects it is related, with members closer to the vibrating individual in the network vibrating more strongly than members further away.

$$Vibration_{x,y} = w_y \cdot e^{-d_{x,y}^2}$$
(27)

where, the  $d_{x,y}$  is the Euclidian distance between the spiders x and y,  $d_{x,y} = |P_x - p_y|$ .

It is possible to calculate the perceived vibrations considering any individual pair.

Vibrations  $(vib_x)$  are observed by the individual  $x(p_i)$  of

result information transmitted by member count  $c(p_i)$ , who is an individual that has two essential attributes configured in Figure 4.



Figure 4. Relation between spiders (a) and (b)

$$w_b = \max_{c \in \{1, 2, \dots N\}} (w_c)$$
(28)

$$vib_x = w_b. e_{xb}^{-d^2} \tag{29}$$

The vibrations  $vibf_x$  observed by the individual  $I(p_I)$ result from the information transmitted by the member  $F(p_f)$ , with f being the nearby female count to x.

The resulting attraction depends on some random event whose choice is designed to be a random outcome. This function generates the same random number  $(n_r)$  in the range [0, 1].  $n_f$  limit is less than  $n_m$ , movement occurs,

$$f_x^{n+1}f_x^1 \begin{cases} f_x^n + \alpha. vib_x. (p_x - f_x^n) + \beta. vib_x. (p_x - f_x^n) + \delta. (rand - 1/2) \text{ with probability Th} \\ f_x^n - \alpha. vib_x. (p_x - f_x^n) - \beta. vib_x. (p_x - f_x^n) + \delta. (rand - \frac{1}{2}) \text{ with probability } 1 - Th \end{cases}$$
(30)

where,  $\alpha$ ,  $\beta$ ,  $\delta$  and random numbers between (0,1) whereas n represents the iteration numbers.  $p_c$  and  $p_b$  represent the nearby member to x, who holds a maximum weight and is the best individual in the entire web population (p). Therefore, this algorithm has a global search capability to improve the exploit behavior of the proposed approach.

$$M_{x}^{n+1} = \begin{cases} m_{x}^{n} + \alpha. vib_{x}. \left(p_{f} - m_{x}^{n}\right) + \delta. \left(rand - \frac{1}{2}\right) w_{f} + x \\ > w_{nf} + m \\ m_{x}^{n} + \alpha. \left(\frac{\sum_{M=1}^{N_{m}} m_{x}^{n}. w_{n_{f}+M}}{\sum_{M=1}^{M_{n}} w_{nf+M}} - m_{x}^{n}\right) if w_{n_{f}}^{x} \\ \le w_{n} f + m \end{cases}$$
(31)

where, the separate  $P_f$ , the nearby female separates to the male member X, as  $\frac{\sum_{M=1}^{N_m} m_{x}^n \cdot w_{n_f} + m}{\sum_{M=1}^{M_n} w_{n_f} + M}$  match to the weighted mean of the male population  $M_p$ .

Calculate the weights of all spiders in  $S_p$ 

for 
$$(x = 1, x < n + 1, x + +)$$
 (32)

$$w_x = \frac{worst_p - s(c)}{worst_p - best_p}$$
(33)

where, c(.) -represent sub-group function,  $best_p =$  $\min_{c \in \{1,2,3.n\}} (p_n) \text{ and } worst_p = \max_{c \in \{1,2,3.n\}} (p_n).$ 

End for

Move the male spider towards the male partner Find the intermediate male individual  $(w_{n_{f+m}})$  from M

$$for (x = 1, x < n + 1, x + +)$$
  
Evaluate  $vib_x$   
If  $(w_{n_{f+x}} + w_{n_{f+m}})$   
 $m_x^{n+1} = m_x^n + \alpha . vib_x, (p_f - m_x^f) + \delta . (rand - \frac{1}{2}))$   
Else if

$$m_x^n + \alpha. \left( \frac{\sum_{M=1}^{N_m} m_x^n . w_{n_f + m}}{\sum_{M=1}^{M_n} w_{n_f + M}} - m_x^n \right)$$

End if End for

Maximum vibrations communicate essential information through the web. This information is considered local information and is used by each member to carry out cooperative activities while prompting the social parameter of the group.

#### 3.6 SoftMax Neural Network (SN2)

A SoftMax Neural Network is a mathematical function that

converts numbers to probabilities. The probability of all values is relative to the relative section of each value.

SN2 is implemented through weight optimization, changing the weights to minimize the loss function. The SoftMax function is defined as:

$$F(a_{x}) = \frac{e^{a_{x}}}{\sum_{y=1}^{n} e^{a_{x}}}$$
(34)

where,  $a_x$  is the  $x^{th}$  dimensional output, and N is the number of dimensions, usually equal to the number of classes, let us think of  $F(a_x)$  as the probability of the ith class.

$$F(a_x) = \max_{x \in (1,n)} F(a_x) \tag{35}$$

SoftMax function has a learning problem when N classes are large. First, the significant parameters of the last layer perform only forward and backward propagation. Images in this category have a predicted score lower than a classifier containing. Mathematically, this can be formalized as:

 $F(a_{xi}) < F(a_{xj})$  Where  $i \neq j$  and  $i, j \in [1,]$  F. The input value is estimated as positive or negative of the input value, and the function output creates a network or Model for multiclass classification. The network's output layer contains as many neurons and classes as there are targets.

The SoftMax activation feature prevents neurons from strengthening and makes learning more efficient. However, for more complex problems, the calculations are more efficient.

# 3.7 Advance sequential long short-term memory with convolutional neural network (ASLSTM-CNN)

The first step is converting the video clip into images and detecting the cars from each frame. The next step is to find number plates from detected vehicles. In the final stage, the reading of the license plate letters is recognized from the detected license plates using ASLSTM-CNN. The proposed deep learning model uses the Image library to simplify the training process.

First, the CNN layer is selected to extract features from cropped images. The resulting feature array is fed into ASLSTM layers, which recognize letters in a left-to-right sequence, with multiple features identifying a single note. Finally, the results are processed through the translation layer, and the authentication results are defined according to the number plate character recognition.

#### • Sequential Features in Vehicle Numbers

In an ASLSTM-CNN architecture, the output of the CNN is not used directly as an output. Instead, it is sent to an RNN with 36 hidden layers for further ranking. By learning a nonlinear function, the recurrent layer derives the label inserting of the predicted labels and the co-occurrence biases in the Model's latent recurrence stage.

$$S(t) = H[s(t-1), w_l(t)]$$
(36)

$$Op(t) = H[s(t-1), w_l(t)]$$
 (37)

where, H[.] -transforming functions S(t) and Op(t) are the hidden layers and output of the recurrent layer based on the time (t).  $w_l(t)$  is the prediction label.

## Image features Training function

The ASLSTM-CNN model is trained using cross-entropy loss and stochastic gradient descent with SoftMax scores and backpropagation (for the CNN-LSTM model) with time normalization. Although it is possible to fine-tune our upper VGGNet layers, we must keep the network constant in our implementation for simplicity.

The first convolutional layers of ASLSTM-CNN usually learn simple image features like edges and selecting features. Since these features are the same for different objects, the Model needs to know only the weights of new convolutional layers. For this reason, during the training of the VGGNet layer, the backpropagation is stopped after the final convolutional layer weights are changed.

Figure 5 describes the CNN-LSTM Model, which combines a CNN layer, which provides sequence prediction, with an LSTM layer that extracts features from input data. CNN-LSTMs are commonly used for plate recognition and image labeling. Their common feature is that they have been developed to apply visual time series prediction problems and generate text annotations from image sequences.



Figure 5. LSTM-CNN layers training function

Each point corresponds to the center point of the original image Region after the feature map output by convolution. Based on this, two default boxes with different aspect ratios are created. The default box is meant to match the ground truth box of the number plate. Finally, default boxes with category probabilities below the Threshold (0.7) are excluded.

Table 1 shows the dataset details, which contain vehicle number plate images based on the training, valid, and testing processing using vehicle number plate images, which are augmented by collecting roadside mapping units. It evaluates Characters as the kind of all the characters shown on the number plate.

Table 1. Number plate detection train, test, and validation

Dataset	<b>Images Count</b>	Characters
Training	4000	65
Validation	1000	65
Testing	1000	65

• Loss function

The loss function has two parts: Confidence calculations correspond to default boxes, target categories, and regression results for associated positions. Reliability is achieved by position regression with SoftMax Loss.

$$LF(a, c, R, m) = \frac{1}{N} (Lf_{con}(a, c) + \alpha Lf_{Rel}(a, R, m))$$
(38)

where, N-number of positive images, a, c, R, m - features weights, lf-loss function

$$Lf_{con}(a, R, m) = \sum_{x \in positive}^{N} \sum_{m \in \{a, Rb, w, m\}} L^{n}_{ab} \ softmax_{lf}(lf^{n}_{a} - R_{b})$$
(39)

In the validation process, the samples are split into the training set, which gets the first 70% of the frames, and the test set receives the rest of the edges. This approach is better than the loss function for the random selection of structures used in the video section.

## • ASLSTM-CNN

Advance Sequential Long Short Term, Memory with Convolutional Neural Network (ASLSTM-CNN) model comparing the target and training images, provides a large dataset with training images. An active neural network performs it. Need an encoder to extract image features. Use CNN to segment and extract the generated image using LSTM. Accuracy is assessed using the ranked BLEU index.

For X = 1: N (All both segmentation) do

For A = 1:  $F_x$  (All frames), do Detecting each character with a target bounding box;

Image region  $(R_x)$  with a target bounding box;

 $R(x) = \sum_{a \in \beta} i^a X x_a + \sum_{\{a,b\}} j^{ab} X |x_a - x_b|$ 

where,  $\beta$ -image pixels,  $a \in \beta$ ,  $x_a$  – individual pixels, in the pixel values assume 0 and 1 weights. Pairs of neighboring pixels defined  $i^a$ ,  $j^{ab}$ .

Region of Interest (ROI) with Aspect Ratio Preservation;

End

Collect the set of various points  $\{V_p\}_{a=1}^{f_x}$ ;

Compute feature vector  $x_a$ 

End

 $x_1 = RNN(a_1, x_{i-1})$ 

 $x_1$ -Image size,  $a_1$ -current images pixels,  $x_{i-1}$ -previous images values.

 $\begin{aligned} x_i^f &= ASLSTM_f(a_1, x_{i-1}^f) \\ x_i^b &= ASLSTM_b(a_1, x_{i+1}^b) \\ \text{Evaluating Training set } \chi &= (\{x_a, y_b\})_{x=1}^N \\ \text{CNN classifier using cross-validation} \end{aligned}$ 

End

where, f-Advancing LSTM, b-backward LSTM,  $label = \arg \max L(x)$  Arguments Extract feature vector of a target

image. The training data is matched with the target image features for image segmentation. ASLSTM-CNN extracts feature from these images and transforms the image text into text to detect category and vehicle number, respectively.

#### 4. RESULT AND DISCUSSION

This section tests the proposed method ASLSTM-CNN using features trained from a vehicle number plate dataset. Performance assessments are performed in stages to test accuracy, compliance, and repeatability. The text case metric is calculated based on the actual/error status of the execution error rate. The performance value is a combination of positive and negative values.

Table 2 describes A dataset of vehicle images processed to test the efficiency of the proposed system. Several training and test images are evaluated for the number plate segments to estimate the numbers for the classification.

Table 3 presents the precise analysis, also called positives or the proportion of related measures. For the imbalanced bias classification problem, divide the accuracy by the number of true and false positives.

$$Precision(P) = TP/(TP + FP) * 100$$
(40)

Figure 6 compares methods for precision values of the actual positive rate, and the proposed implementation achieves higher performance than the others. Existing methods have 86% of convolutional neural network (CNN), 92% of End-to-End Irregular Number Plate Recognition (EILPR), and 83% of YOLOv5. Advanced Sequential Long Short-Term Memory (ASLSTM-CNN) has a high accuracy of 95%, which is better than previous methods.

Table 2. Simulation parameters for the proposed method

Using Parameters	Values
Name of the Dataset	Vehicle Number Plate dataset
Tool	Anaconda
Language	Python
No. Of. images	1000
Trained images	700
Testing images	300

Table 3. Precision performance

Number of Images	YOLOv5 %	CNN %	EILPR %	ASLSTM- CNN %
50	46	51	60	69
100	54	66	73	78
150	62	74	80	84
200	75	79	86	90
250	78	82	88	92
300	83	86	92	95



Figure 6. Analysis of precision rate

Table 4. Analysis of recall performance

Number of	YOLOv5	CNN	EILPR	ASLSTM-
Images	in %	in %	in %	CNN in %
50	46	52	58	68
100	51	60	67	75
150	64	71	76	85
200	72	76	84	89
250	75	79	86	90
300	80	84	89	91

Table 4 shows the actual positive recall performance number divided by the total number of elements in the positive class.

$$Recall(R) = TP/(TP + FN) * 100$$
(41)



Figure 7. Analysis of recall values

Figure 7 compares the recall of true positives and false negatives between different methods. The proposed implementation performs better than other algorithms. In the existing techniques, Convolutional Neural Network (CNN) is 84%, End-To-End Irregular Number Plate Recognition (EILPR) is 89%, and YOLOv5 is 80% but the proposed method of Advance Sequential Long Short-Term Memory with Convolutional Neural Network (ASLSTM-CNN) is 91% is high recall better than previous methods.

Table 5. Analysis of image detection accuracy

Number of Images	YOLOv5 %	CNN %	EILPR %	ASLSTM- CNN %
50	52	57	62	67
100	59	65	72	75
150	70	78	80	84
200	76	81	85	88
250	79	82	86	89
300	84	88	90	96

Table 5 describes how this will create different levels of users and improve the accuracy of number plate image detection. The proposed system significantly impacts number plate detection performance compared to other methods.

$$Accuracy(A) = TP/(TP + TN) * 100)$$
(42)

$$F - score(F) = \frac{\left(2 * Precision(P) * Recall(R)\right)}{\left(Precision(P) + Recall(R)\right)} \quad (43)$$

Figure 8 compares the image detection accuracy values of the various methods, and the proposed implementation performs better than other algorithms. In existing approaches, convolutional neural network (CNN) is 88%, end-to-end irregular number plate recognition (EILPR) is 90%, and YOLOv5 is 84%. In contrast, our proposed method uses Advance Sequential Long Short-Term Memory convolutional neural network (ASLSTM)-CNN), and the recall is 96% higher, which is better than previous methods.

Table 6 describes the false error rate performance of different classes of number plate images to reduce errors. The

proposed method reduces the error in image training and test datasets.



Figure 8. Analysis of image detection accuracy

Table 6. False rate performance

Number of Images	YOLOv5 %	CNN %	EILPR %	ASLSTM- CNN %
50	59.1	55.2	52.1	50.2
100	50.1	48.2	45.2	42.1
150	49.2	46.2	44.3	43.4
200	46.2	43.2	42.4	40.6
250	44.2	42.4	41.8	39.2
300	40.5	39.5	38.8	34.8



Figure 9. Analysis of false score

Figure 9 shows the value of the false rate by comparing different methods, finding that the proposed implementation has error rate performance compared to other algorithms. Existing approaches have 39.5% for convolutional neural networks (CNN), 38.8% for end-to-end irregular number plate recognition (EILPR), and 40.5% for YOLOv5. In contrast, our proposed method's Advance Sequential Long Short-Term Memory convolutional neural network (ASLSTM-CNN) false rate is 34.8% lower than the previous process.

## **5. CONCLUSION**

To conclude, Vehicle number plate detection and recognition are performed using image processing techniques. Compared with image processing techniques, deep learningbased methods provide more reliable results. Propose an automatic number plate detection and recognition method that extracts number plate features using a decomposable region model and trains a number plate detector-using Advance Sequential Long Short-Term Memory with Convolutional Neural Network (ASLSTM-CNN). To improve the accuracy of optical character recognition, number plates extracted from the scenes captured by the camera are further enhanced. Recognize the number plate characters and get the vehicle status and detailed information. A record of detected vehicles is also maintained by storing the vehicle number plate text. The proposed method for number plate detection and recognition using deep Learning achieves an accuracy of 96% in the ASLSTM-CNN model, respectively, which gives reliable results. As a result, the proposed method finds a solution to recognize characters.

### ACKNOWLEDGMENT

The authors wish to express their heartfelt gratitude to the Department of Electronics and Communication Engineering, Sona College of Technology, Salem, Tamil Nadu, India, for providing the essential resources and infrastructure to carry out this research. We extend our sincere thanks to our colleagues and mentors for their invaluable feedback and guidance, which significantly contributed to the success of this work.

#### **AUTHORS' CONTRIBUTIONS**

**Sree southry:** Conception and design of the study, development of the methodology, and supervision of the overall research project. Led the data collection and analysis, drafted the manuscript, and critically revised the content. Also provided substantial contributions to the interpretation of results.

**Sabeenian:** Provided guidance on the study design, contributed to the critical revision of the manuscript, and offered expert insights during the interpretation of data. Oversaw the methodological approach and validated the final draft of the manuscript.

## REFERENCES

- Al-Shemarry, M.S., Li, Y. (2020). Developing learningbased preprocessing methods for detecting complicated vehicle licence plates. IEEE Access, 8: 170951-170966. https://doi.org/10.1109/ACCESS.2020.3024625
- [2] Khan, M.G., Saeed, M., Zulfiqar, A., Ghadi, Y.Y., Adnan, M. (2022). A novel deep learning based ANPR pipeline for vehicle access control. IEEE Access, 10: 64172-64184. https://doi.org/10.1109/ACCESS.2022.3183101
- [3] Pustokhina, I.V., Pustokhin, D.A., Rodrigues, J.J., Gupta, D., Khanna, A., Shankar, K., Seo, C., Joshi, G.P. (2020). Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems. IEEE Access, 8: 92907-92917.

https://doi.org/10.1109/ACCESS.2020.2993008

- [4] Charran, R.S., Dubey, R.K. (2022). Two-wheeler vehicle traffic violations detection and automated ticketing for Indian road scenario. IEEE Transactions on Intelligent Transportation Systems, 23(11): 22002-22007. https://doi.org/10.1109/TITS.2022.3186679
- [5] Molina-Moreno, M., González-Díaz, I., Díaz-de-María,

F. (2018). Efficient scale-adaptive license plate detection system. IEEE Transactions on Intelligent Transportation Systems, 20(6): 2109-2121. https://doi.org/10.1109/TITS.2018.2859035

- [6] Shashirangana, J., Padmasiri, H., Meedeniya, D., Perera, C. (2020). Automated license plate recognition: A survey on methods and techniques. IEEE Access, 9: 11203-11225. https://doi.org/10.1109/ACCESS.2020.3047929
- Yousif, B.B., Ata, M.M., Fawzy, N., Obaya, M. (2020). Toward an optimized neutrosophic K-means with genetic algorithm for automatic vehicle license plate recognition (ONKM-AVLPR). IEEE Access, 8: 49285-49312. https://doi.org/10.1109/ACCESS.2020.2979185
- [8] Valdeos, M., Velazco, A.S.V., Paredes, M.G.P., Velásquez, R.M.A. (2022). Methodology for an automatic license plate recognition system using convolutional neural networks for a Peruvian case study. IEEE Latin America Transactions, 20(6): 1032-1039. https://doi.org/10.1109/TLA.2022.9757747
- [9] Du, S., Ibrahim, M., Shehata, M., Badawy, W. (2012). Automatic license plate recognition (ALPR): A state-ofthe-art review. IEEE Transactions on Circuits and Systems for Video Technology, 23(2): 311-325. https://doi.org/10.1109/TCSVT.2012.2203741
- [10] Henry, C., Ahn, S.Y., Lee, S.W. (2020). Multinational license plate recognition using generalized character sequence detection. IEEE Access, 8: 35185-35199. https://doi.org/10.1109/ACCESS.2020.2974973
- [11] Panahi, R., Gholampour, I. (2016). Accurate detection and recognition of dirty vehicle plate numbers for highspeed applications. IEEE Transactions on Intelligent Transportation Systems, 18(4): 767-779. https://doi.org/10.1109/TITS.2016.2586520
- [12] He, M.X., Hao, P. (2020). Robust automatic recognition of Chinese license plates in natural scenes. IEEE Access, 8: 173804-173814. https://doi.org/10.1109/ACCESS.2020.3026181
- [13] Oliveira-Neto, F.M., Han, L.D., Jeong, M.K. (2013). An online self-learning algorithm for license plate matching. IEEE Transactions on Intelligent Transportation Systems, 14(4): 1806-1816. https://doi.org/10.1109/TITS.2013.2270107
- [14] Ashtari, A.H., Nordin, M.J., Fathy, M. (2014). An Iranian license plate recognition system based on color features. IEEE Transactions on Intelligent Transportation Systems, 15(4): 1690-1705. https://doi.org/10.1109/TITS.2014.2304515
- [15] Fan, X., Zhao, W. (2022). Improving robustness of license plates automatic recognition in natural scenes. IEEE Transactions on Intelligent Transportation Systems, 23(10): 18845-18854. https://doi.org/10.1109/TITS.2022.3151475
- [16] Anagnostopoulos, C.N.E., Anagnostopoulos, I.E., Psoroulas, I.D., Loumos, V., Kayafas, E. (2008). License plate recognition from still images and video sequences: A survey. IEEE Transactions on Intelligent Transportation Systems, 9(3): 377-391. https://doi.org/10.1109/TITS.2008.922938
- [17] Huang, Q., Cai, Z., Lan, T. (2020). A new approach for character recognition of multi-style vehicle license plates. IEEE Transactions on Multimedia, 23: 3768-3777. https://doi.org/10.1109/TMM.2020.3031074
- [18] Luo, S., Liu, J. (2022). Research on car license plate recognition based on improved YOLOv5m and LPRNet.

IEEE Access, 10: 93692-93700. https://doi.org/10.1109/ACCESS.2022.3203388

- [19] Zou, Y., Zhang, Y., Yan, J., Jiang, X., Huang, T., Fan, H., Cui, Z. (2020). A robust license plate recognition model based on bi-LSTM. IEEE Access, 8: 211630-211641. https://doi.org/10.1109/ACCESS.2020.3040238
- [20] Wang, Y., Bian, Z.P., Zhou, Y., Chau, L.P. (2021). Rethinking and designing a high-performing automatic license plate recognition approach. IEEE Transactions on Intelligent Transportation Systems, 23(7): 8868-8880. https://doi.org/10.1109/TITS.2021.3087158
- [21] Raghunandan, K.S., Shivakumara, P., Jalab, H.A., Ibrahim, R.W., Kumar, G.H., Pal, U., Lu, T. (2017). Riesz fractional based model for enhancing license plate detection and recognition. IEEE Transactions on Circuits and Systems for Video Technology, 28(9): 2276-2288. https://doi.org/10.1109/TCSVT.2017.2713806
- [22] Wang, W.H., Tu, J.Y. (2020). Research on license plate recognition algorithms based on deep learning in complex environment. IEEE Access, 8: 91661-91675. https://doi.org/10.1109/ACCESS.2020.2994287
- [23] Pan, X., Li, S., Li, R., Sun, N. (2022). A hybrid deep learning algorithm for the license plate detection and recognition in vehicle-to-vehicle communications. IEEE Transactions on Intelligent Transportation Systems, 23(12): 23447-23458. https://doi.org/10.1109/TITS.2022.3213018
- [24] Wang, W., Yang, J., Chen, M., Wang, P. (2019). A light CNN for end-to-end car license plates detection and recognition. IEEE Access, 7: 173875-173883. https://doi.org/10.1109/ACCESS.2019.2956357
- [25] Zhang, L., Wang, P., Li, H., Li, Z., Shen, C., Zhang, Y. (2020). A robust attentional framework for license plate recognition in the wild. IEEE Transactions on Intelligent Transportation Systems, 22(11): 6967-6976. https://doi.org/10.1109/TITS.2020.3000072
- [26] Gou, C., Wang, K., Yao, Y., Li, Z. (2015). Vehicle license plate recognition based on extremal regions and restricted Boltzmann machines. IEEE Transactions on Intelligent Transportation Systems, 17(4): 1096-1107. https://doi.org/10.1109/TITS.2015.2496545
- [27] Seo, T.M., Kang, D.J. (2022). A robust layoutindependent license plate detection and recognition model based on attention method. IEEE Access, 10: 57427-57436.

https://doi.org/10.1109/ACCESS.2022.3178192

- [28] Xu, H., Zhou, X.D., Li, Z., Liu, L., Li, C., Shi, Y. (2021). EILPR: Toward end-to-end irregular license plate recognition based on automatic perspective alignment. IEEE Transactions on Intelligent Transportation Systems, 23(3): 2586-2595. https://doi.org/10.1109/TITS.2021.3130898
- [29] Tourani, A., Shahbahrami, A., Soroori, S., Khazaee, S., Suen, C.Y. (2020). A robust deep learning approach for automatic iranian vehicle license plate detection and recognition for surveillance systems. IEEE Access, 8: 201317-201330.

https://doi.org/10.1109/ACCESS.2020.3035992

- [30] Silva, S.M., Jung, C.R. (2021). A flexible approach for automatic license plate recognition in unconstrained scenarios. IEEE Transactions on Intelligent Transportation Systems, 23(6): 5693-5703. https://doi.org/10.1109/TITS.2021.3055946
- [31] Malarvizhi, C., Dass, P., Karthikeyani, P., Sudha, V., Iniyan, S. (2023). Machine learning and advanced technology based fire detection. In 2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), Bengaluru, India, pp. 1232-1235. https://doi.org/10.1109/IITCEE57236.2023.10090923
- [32] Chinnusamy, J.R., Ranganathan, K.K.P., Sekar, V., Balasundaram, M.B. (2023). People flow management using computer vision & deep learning. In AIP Conference Proceedings. AIP Publishing, 2857(1). https://doi.org/10.1063/5.0164341
- [33] Vijayalakshmi, S., Kavitha, K.R., Sangeetha, D.P., Thilak, S. (2020). Robust defect detection on PCB based on deep learning. Journal of Advanced Research in Dynamical and Control Systems, 12(5): 937-949.
- [34] Ayub khan, A., Sabeenian, R.S., Janani, A.S., Akash, P. (2022). Vehicle classification and counting from surveillance camera using computer vision. In Inventive Systems and Control: Proceedings of ICISC 2022. Singapore: Springer Nature Singapore, pp. 457-470. https://doi.org/10.1007/978-981-19-1012-8\_31
- [35] Suriyakrishnaan, K., Kumar, L.C., Vignesh, R. (2022). Recommendation system for agriculture using machine learning and deep learning. In Inventive Systems and Control: Proceedings of ICISC 2022. Singapore: Springer Nature Singapore, pp. 625-635. https://doi.org/10.1007/978-981-19-1012-8\_42
- [36] Paul, E., Sabeenian, R.S. (2022). Modified convolutional neural network with pseudo-CNN for removing nonlinear noise in digital images. Displays, 74: 102258. https://doi.org/10.1016/j.displa.2022.102258
- [37] Yin, G., Huang, S., He, T., Xie, J., Yang, D. (2023). Mirrored EAST: An efficient detector for automatic vehicle identification number detection in the wild. IEEE Transactions on Industrial Informatics, 20(3): 4594-4605. https://doi.org/10.1109/TII.2023.3322158
- [38] Mustafa, T., Karabatak, M. (2024). Real time car model and plate detection system by using deep learning architectures. IEEE Access. https://doi.org/10.1109/ACCESS.2024.3430857
- [39] Li, Q., Li, X., Yao, H., Liang, Z., Xie, W. (2023). Automated vehicle identification based on Car-Following data with machine learning. IEEE Transactions on Intelligent Transportation Systems, 24(12): 13893-13902. https://doi.org/10.1109/TITS.2023.3304607
- [40] Sun, W., Xu, F., Zhang, X., Hu, Y., Dai, G., He, X. (2023). A dual-branch network for few-shot vehicle reidentification with enhanced global and local features. IEEE Transactions on Instrumentation and Measurement, 72: 1-12. https://doi.org/10.1109/TIM.2023.3285978