Journal of Smart Energy and Sustainability, 1(2): 105-115 Spring 2022

ORIGINAL RESEARCH PAPER

IoT-based Automated Vehicle Plate Detection Algorithm for Urban Surveillance Systems by A New Hybrid Optimized CNN

Dong Sun

Department of Computer Science and Technology, Henan Institute of Technology, Xinxiang 453003, China

Received: 2021-12-25 Accepted: 2022-03-05 Published: 2022-04-01

ABSTRACT

One of the important and more challenging categories in the smart cities and IoT is to monitor the vehicles plate licenses. This system is a key factor in most of the traffic monitoring in the IoT based smart city applications. In this research, a method for plate license recognition based on optimal training of the CNN is proposed. To do this, the configuration and the hyperparameters of the CNN were optimized by a new hybrid optimization including world cup optimizer, whale optimizer, and chaotic theory to obtain a better result with high convergence. Simulations are applied to the UFPR-ALPR dataset and are compared with six popular techniques in terms of accuracy and time. Experimental achievements indicated that the proposed method gives superiority toward the other comparative techniques and is an efficient method for vehicles plate licenses detection.

Keywords: *Vehicle license plate recognition; CNN; whale optimization algorithm; world cup optimization algorithm; IoT; Logistic Mapping*

How to cite this article

Sun D. IoT-based Automated Vehicle Plate Detection Algorithm for Urban Surveillance Systems by A New Hybrid Optimized CNN J. Journal of Smart Energy and Sustainability, 2022; 1(2): 105-115.

DOI: 10.22034/SES.2022.02.01

1. INTRODUCTION

In a smart city, all parts of the city, such as transportation systems, schools, libraries, hospitals, power plants, water networks, garbage collection systems, and many other urban amenities, can be integrated and, by aggregating their information, improve the quality of service provision for each sector [1].

The Internet of Things (IoT) is a key technology in smart cities. Estimates show that by 2020, this technology will cover more than 30 billion objects.

The Internet of Things is a platform for connecting different intelligent objects with each other and the user in a unified network [2, 3]. This network can include different objects from smart different sensors, TVs, HVAC systems, air conditioning systems and their command parts like smartphones, and remote controls which can be used in different locations like smart homes, urban surveillance system, and traffic monitoring.

Urban surveillance and smart transportation systems are significant parts of the IoT applications in today's world [4, 5].

An important structure in the smart transportation systems is imaging sensors (i.e. cameras) which are used for automatic identifying the considered vehicles (cars) based on image processing and machine vision techniques.

In most applications, the performance of the suggested technique was decreased because of the improper selection of the technique of object detection.

Meanwhile, deep neural networks are powerful and widely used subject in computer science and machine learning that is one of the most accurate methods in the present years [6, 7]. This

* Corresponding Author Email: *13938761140@hait.edu.cn*

type of neural network is capable of learning complex patterns which are also used for its power and accuracy in many real-world issues (like vehicle license plate recognition (VLPR), vehicle recognition, disease prediction, automatic categorization of texts and images).

There are various classification applications for the deep learning-based system in image processing [8].

For example, in 2015, Rajput et al. proposed an automated system for the recognition of the vehicle license plate [9]. The system contained three stages including image acquisition, plate localization, and OCR. The simulation results were implemented on forty different vehicle license plates.

In 2016, Kolosnjaji et al. presented a deep structure for the classification of the malware system call sequences. They designed a CNN for achieving the best features in the classification. Simulation results confirmed that the presented system has a high superiority toward the used methods in malware classification [10].

In 2017, Lee et al. implemented a built-in configuration based on GPU for the license plate recognition without line detection. The proposed method was based on a simple AlexNet deep neural network. Simulation achievements indicated that the proposed method gives a better ability in the detection of the Jetson TX1 board system.

In 2018, Fu et al. presented a plate detection system for Chinese vehicles based on Deep Learning [11]. Their method contained two parts of localizing the license plate and detecting the car plate characters based on leveraging CNN without segmentation. The achievements showed that the suggested technique gives a better operation in detecting of the Chinese vehicle plates.

In 2019, Zhong et al. proposed a multiple-temporal method for crop classification. The proposed deep neural network obtained proper 85.54% accuracy while the best results for the non-deep learning classifier were 84.17% accuracy [12].

In 2019, Khan et al. introduced a deep network configuration for the breast cancer diagnosis in the medical images. They analyzed the method based on three different deep frameworks including ResNet, GoogLeNet, and VGGNet. The results illustrated the system better efficiency for the classification of breast cancer [6].

There are also some different works in the field of VLPR. For instance, in 2018Mohamed Admi et al. presented a technique using the MSER for detecting of the license plates text [13].

For eliminating the false recognition, character classifier was used. The achievements indicated that the suggested technique give more proper results toward another method which is compared in that paper.

From the above explanations, it is clear that VLPR is an interesting subject in the worldwide that is turned into a hot topic in the IoT based utilizations of smart towns like automatic charging system for the parking, looking for robbed vehicles, traffic control, and monitoring the traffic data.

The number of a license plate in each vehicle is unique which can be used to identify the considered vehicle. Generally, in VLPR systems, the main target is to identify the plate location in the vehicles and then to extract the numbers for the monitoring applications [14].

In this research, a new efficient method to direct recognition of the vehicle license plates is proposed. The proposed method can be utilized in smart cities as a real-time system for urban monitoring systems.

In the following, in section 2, the method of the recognition system is presented which includes two main parts CNN and the proposed optimization algorithm. Section 3 is about simulations of the proposed system on a standard dataset. Afterward, in section 4 the discussions about the simulation results are mentioned. The conclusions is the final section of the paper.

2. METHODOLOGY

2.1. Convolutional neural networks

Since the last years when the computational optimizers are performed by computers, much research work has begun by computer scientists, engineers, and mathematicians for the simulation of the computational behavior of the human brain. This is a branch of artificial intelligence (AI) being categorized in the subclass of computational intelligence (CI) under the heading "Artificial Neural Networks" (abbreviated as "ANNs") [15-17].

In the context of artificial neural networks, many software and mathematical techniques which are an inspiration of the human brain have been presented, being applied in various fields for solving various engineering, scientific, and utilized problems.

Convolution Neural Networks (CNN) is a significant learning method where different layers have been trained in an effective way. The technique

is so useful and is a broadly used technique in various vision applications of computer [18].

In general, a CNN network includes the pooling layer, the fully connected layer and the Convolutional layer.

Various layers implement different assignments. In each CNN, two parts of training exists. The feedforward (FF) stage and the backpropagation (BP) or post-propagation (PP) step.

At first, the input image is fed to the network, which is nothing but a point multiplication among the input and the variables of each neuron, and finally the imposition of Convolutional performances on each layer. Then the output of the network is evaluated.

Here, to set the network's variables, the output is employed for calculating the network error rate. To do this, compare the network output using a loss function with the correct solution, and this kind of error is evaluated. The later phase is defined by the evaluated error rate of the BP phase.

In this part, the derivation of the parameters is obtained based on the chain rule and every single parameter is changed based on the effect on the error created. After updating the variables, the later phase is the FF. The training of the network finishes after completion of the true amount of these stages [19].

The main configuration of a CNN and its interior connectivity pattern among neurons is inspired by the main configuration of the visual ability in the animals.

For providing the invariance in rotation, translation, and scaling, three configurations have been considered including spatial or temporal sub-sampling, shared weights, and local receptive fields. The raw image (input) has been fed to the network that is centered and size normalized.

Afterward, some units in a small neighborhood of the former layer send inputs to the neuron in the intermediate layer (IL).

The achieved neurons exploit significant visual features like corners, oriented edges, and endpoints by utilizing local receptive fields (RF). The exploit characteristics have been connected with the next deeper ILs for exploiting the higher-level features. Finally, the prediction output is obtained based on distinct features exploited by the changing layers of the network.

a) Convolutional layers (CL): in this study, three CLs including Conv1, Conv2, and Conv3 have been applied to the inputted image convolution from the prior layer with various two-dimensional weights for exploiting the features.

The outputted activation map in l layer can be defined as follows:

$$x_j^l = f\left(\sum_i x_i^{l-1} \otimes W_{ij}^l + b_j^l\right) = \max\left\{0, \left(\sum_i x_i^{l-1} \otimes W_{ij}^l + b_j^l\right)\right\} \quad (1)$$

where, $\otimes$ describes the convolution operator, the activation function of the CNN (here ReLU) is denoted by f, x_i^{l-1} is the i^{th} inputted feature map (FM) of $l-1$ layer, W_{ij}^l is the weight filter connecting the j^{th} FM of the outputted layer to the i^{th} FM of the inputted layer and b_j^l convoluted with the inputted signal to produce the outputted characteristic as a bias.

The reason for selecting the ReLU is that it is more efficient for training rather than other activation functions like hyperbolic tangent and sigmoid.

Like other neural networks configuration, weights values are first selected randomly and then optimized based on backpropagation and gradient descent technique.

Here, the only difference between Conv1 and Conv2 and the Conv3 is the size of Convolutional filters and their number.

b) Pooling layers: The process of reducing the size of feature maps called pooling. This process generates stable output for distortions and shifts. Furthermore, this step will also decrease the number of training parameters and avoiding the overfitting problem.

Max pooling is deployed for sub-sampling the feature maps in Pool1 and Pool2 which is applied to the considered FM in the prior layer as follows:

$$x_j^l = down(x_j^{l-1}) \quad (2)$$

here, $down(.)$ defines a kind of Convolutional pooling.

Here, a rectangular dimension region $L_x \times L_y$ is applied to find the maximum response from the inputted FMs of the prior layer, in which the down sampling factor $d_x = d_y = 2$ and the step size is 1.

As a result, the outputted FM's size in the present layer can be halved than the previous layer.

As a result, the final output is the multi-scale features which are extracted from the input plate images. The pooling step for Pool2 will be repeated

similar to Pool1.

To obtain a multi-layer configuration for the feature extraction, the Convolutional layers and the max-pooling layers have been repeated.

C) Fully connected layer: neurons that are fully connected are associated with all the neurons in the previous layer, exactly as they are seen in the normal neural networks. Here, *Softmax* categorizer is employed in the eventual layer to manipulate the various categorizations of the vehicle plates.

Softmax classifier is fed by the feature vectors of the plate that is the predecessor layers' output. The classifier output evaluates the possibilities of the vehicle plate information classes for the images, where the maximum amount describes the expected vehicle plate information category.

For a two-fold categorization, we have:

$$P(y=1 \mid x;w)=\frac{1}{1+e^{-w^T x}} \tag{3}$$

where, x and w describe the feature vector with K dimensions and the weight vector variable, respectively where they belong to $\Re^{(K+1)\times 1}$, y is the label of the class.

To perform a multi-class vehicle plate recognition which the output variable y can have various N values, therefore two-fold recognition is used to make the output probability:

$$P(y=r \mid x;w)=\frac{e^{w_i^T x}}{\sum_{i=1}^{N} e^{w_i^T x}} \tag{4}$$

where, w represents the corresponding class for the weight parameters.

D) Cross-entropy loss function: at the final step, the generalized error can be evaluated using the final Softmax layer. The loss function is given below:

$$E_r=-\left|\sum_{i=1}^{M}\sum_{j=1}^{N} f\left(\log\frac{e^{w_r^T x^{(i)}}}{\sum_{j=1}^{N} e^{w_j^T x^{(i)}}}\right)\right| \tag{5}$$

when the class of the plate is true, otherwise $E_r=0$.

2.2. Hybrid Whale Optimizer (HWO) based on World Cup Optimizer (WCO) and Chaotic Theory (CT)

Meta-heuristic is the science of achieving the optimum solution for optimization problems. Applications of this science are exponentially increasing due to increasing of the number of NP-hard and complicated problems [20-22]. In most cases, meta-heuristic algorithms have inspired from different processes in the real world from human social behavior to animals instincts, or other phenomena. For instance, the genetic algorithm is inspired by Darwin's evolution theory [23-25], particle swarm optimization which is simulated based on fish swarms or the birds motion [26-30], whale optimization algorithm that is derived from the whales prey hunting process [18, 31, 32], and *WCO* which is inspired from the human society football competitions [33-37].

From the literature and the aforementioned explanations, it is clear that the whale optimizer (WO) is a newly introduced bio-inspired optimizer which is inspired by the whales hunting process [32].

WO is an optimization algorithm that begins with a vector of solution parameters randomly as the solution candidates. After generating the vector solution, WO testifies the value of the fitness function based on random solutions. After finding the initial cost of each value, it was renewed by the optimizer formulas until the desired value is satisfied.

The main idea in WO is to simulate the prey hunting process of the whales by generating a trap in a torsion movement around the prey and then catch them all; this process is named *bubble-net feeding behavior*.

This part forms the main contribution of the WO.

The encircling process is modeled as follow:

$$X(t+1)=\begin{cases} X^*(t)-AD & p<0.5 \\ D'e^{bl}\cos(2\pi t)+X^*(t) & p\geq 0.5 \end{cases} \tag{6}$$

$$A=2ar-a \tag{7}$$

where, t describes the present iteration, D' is the present best candidate, and b is the logarithmic form of the torsion movement.

Afterward, the process of bubble-net hunting can be defined as follows:

$$D'=\left|CX^*(t)-X(t)\right| \tag{8}$$

$$C=2r \tag{9}$$

and:

$$-1\leq l\leq 1$$

$$0<p\leq 1$$

$$0<r\leq 1$$

$$0<a\leq 2$$

The above values are random values in the mentioned interval [32].

It is important to note that to make sure about the algorithm convergence, the best candidate develops the agent's position if $|X|>1$. Otherwise, the best candidate will perform as the pivot point. The pseudo-code of the WO algorithm has been defined below.

Start
Initialization: *population* (X_i), *a, C, and A*
Assess the value of cost function for the agent
X^* *mentions the optimum candidate of the present iteration*
Apply WO:
t=1
while *t* ≤ *maximum iteration:*
for *all agents*
if $|A| \le 1$ ***then***
Update the location of each agent
else
Find a random value for the agent X_{rand}
Update the location of each agent
end if
end for
Update a, C, and A
Update X^*
t = t + 1
end while
return X^*
End

Generally, in some cases, the global optimization for the WO doesn't work properly. So, we need a strong exploration term to improve this shortcoming.

In this research, a hybrid structure including *Logistic Mapping* and a recently defined optimizer, which is WCO, has been applied to improve the exploration part of the WO. To do this, the random parameters vector of the system, i.e. $P_{woa}=[l,p,r,a]$ are achieved by combining *WCO and* the *Logistic Mapping*.

Using the WCO makes the optimizer's consuming time and the convergence rate increased that will enhance the diversity of population to get out of the local optima.

As mentioned before, the world cup optimizer has been utilized to select the best values for the P_{woa} un the determined intervals.

Meaning that, the inputted solution vector for the combined WO is *n* × 4,where *n* refers to the initial population's number:

$$[x_1^m, x_2^m, x_3^m, x_4^m]=[l,p,r,a] \tag{10}$$

and

$$P_{teams}=\begin{bmatrix} x_{c1,1} & x_{c2,1} & \cdots & x_{cM,1} \\ x_{c1,2} & x_{c2,2} & \cdots & x_{cM,2} \\ x_{c1,3} & x_{c2,3} & \cdots & x_{cM,3} \\ x_{c1,4} & x_{c2,4} & \cdots & x_{cM,4} \end{bmatrix} \tag{11}$$

where, x_i^m is the WO parameters which is chosen optimally, P_{teams} defines population of all continent, *M* is the value of the continents, and $x_{i,j}$ is the i^{th} team of the j^{th} country.

Rating parameters in world cup optimizer defined based on a *Ranking* system which is formulated as follows:

$$Rank=\frac{(\beta\times\sigma+\overline{X})}{2} \tag{12}$$

$$\overline{Y}=\frac{1}{n}\sum_{i=1}^{n}Y_i \tag{13}$$

$$\sigma=\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(Y_i-\overline{Y})^2} \tag{14}$$

where, $\overline{Y}$ and σ are respectively the mean and the standard deviation of the *Y*, β is the weight of the σ limited in the interval [0, 1], *n* defines the teams' number.

The next parameter of the WCO is the *Play-Off*, which has been formulated as below.

$$Pop=[Y_{Best},Y_{Rand}] \tag{15}$$

here, *Pop* is the later algorithm population by the size $N\times M$, Y_{Rand} defines a value randomly, and:

$$\frac{1}{2}\times p_r\times(Ub-Lb)<Y_{Best}<\frac{1}{2}\times p_r\times(Ub+Lb) \tag{16}$$

here, *Lb* and *Ub* are the lower and the higher problem limits and p_r is a coefficient of *Lb* and *Ub*. Further information of WCO is achieved by [21].

If you pay attention to the previous sections, Y_{Rand} and p_r from the WCO algorithm are random values in a determined interval.

In this part, the logistic mapping function (LMF) has been used to improve the original WCO by generating several small perturbation into it [38, 39]. Here, the LMF has been used to generate the free parameters of WCO, which are Y_{Rand} and p_r. The chaotic equivalent for the mentioned variables

is defined as below:

$$[L_1^m, L_2^m] = [Y_{Rand}, p_r] \quad (17)$$

where, L_i^m is the i^{th} parameter of the LMF of Eq. (17) and m defines the number of iteration.

The LMF of the algorithm is achieved as follows:

$$L_i^{m+1} = \eta L_i^m (1 - L_i^m) \quad (18)$$

here, η define the coefficient of adjusting, in which $\eta \in [0,1] - \{0.25, 0.5, 0.75\}$ and $L_i^0 \in [0,1]$ is the first number randomly.

The achieved optimizer has been depicted in Fig. (1).

2.3. Hyper-parameter optimization using LMSSOWCO

The purpose of this part is to minimize the error function to obtain the optimal parameters values (x^*) for CNN.

Here, each candidate solution describes a certain point in the search space with D dimensions, in which D mentions the CNN hyper-parameters' number that should be optimized.

3. SIMULATION RESULTS

3.1. Dataset description

The dataset images are plate images of the vehicle from UFPR-ALPR Dataset [40]. This dataset contains 4500 fully annotated images which are extracted from 150 vehicles from the roads and streets where both cameras and vehicle are moving.

The images were acquired by the size 1920 × 1080 pixels from 3 different cameras in the ".PNG" format [41]. We divide the data collection into 3 sections including 50% for training, 20% for testing, and 30% for validation. Fig. (2) depicts several instances of the UFPR-ALPR data collection.

3.2. Training of CNN

Fig. (3) shows the general configuration of the presented optimized CNN based on LMSSOWCO Algorithm.

Various hyperparameters have been searched and optimized based on LMSSOWCO Algorithm from the points of the spatial size of convolutional kernels (Sconvi), number of convolutional kernels (Nconvi), and learning rate (α) in the convolutional layers. Table 1 illustrates the lower and upper limitations for the hyperparameters.

For increasing the speed of the convergence, in all LMSSOWCO iterations, narrower search space domain for the learning is considered. Here, the mean value of the best solution is selected for the hyperparameters.

In this study, the *MatConvNet* library is employed for CNN optimization [42]. Backpropagation (BP) and Stochastic gradient descent (SGD) are deployed for extracting the hierarchical features of the vehicle plate images.

Stochastic gradient descent is applied to develop the weight parameters in the iterations. This technique has the ability to extend very large datasets by a satisfying convergence in CNN [43].

The parameters gradient can be obtained by a mini-batch of the training samples. The new update is formulated below:

$$W_{i+1} = W_i - \alpha \nabla_{W_i} E_r \quad (19)$$

where, α is the learning rate.

As before mentioned, the value of the parameters in the CNN is obtained by evaluating the fitness function and gradient based on the mini-batch size of the training set. MATLAB R2017b is employed to simulate the method with an Intel(R) Core(TM) i7–4720 GHz.

4. RESULT AND DISCUSSION

The main purpose of this research is to place and to segment the vehicle plates from the input images acquired from the cameras.

To do so, a new improved CNN based on a new optimizer, called LMSSOWCO, has been introduced for the optimization of the hyper-parameters of the optimized CNN configuration.

UFPR-ALPR dataset is utilized for the training and the testing of the vehicle plate images. Optimized selection for the proposed CNN configuration by some different LMSSOWCO experiments is illustrated in Table 2.

Different configurations have been introduced for CNN by multiple LMSSOWCO experiments as illustrated in Table 3.

Several numbers of epochs have been employed for minimizing the running time without destroying the network convergence.

Fig. (3) depicts the workflow of the presented technique. The main architecture includes three convolutional layers (conv1, conv2, and conv3) along with ReLU as the activation function.

Based on CNN, all the input images have been convoluted with the convolutional kernels by a step

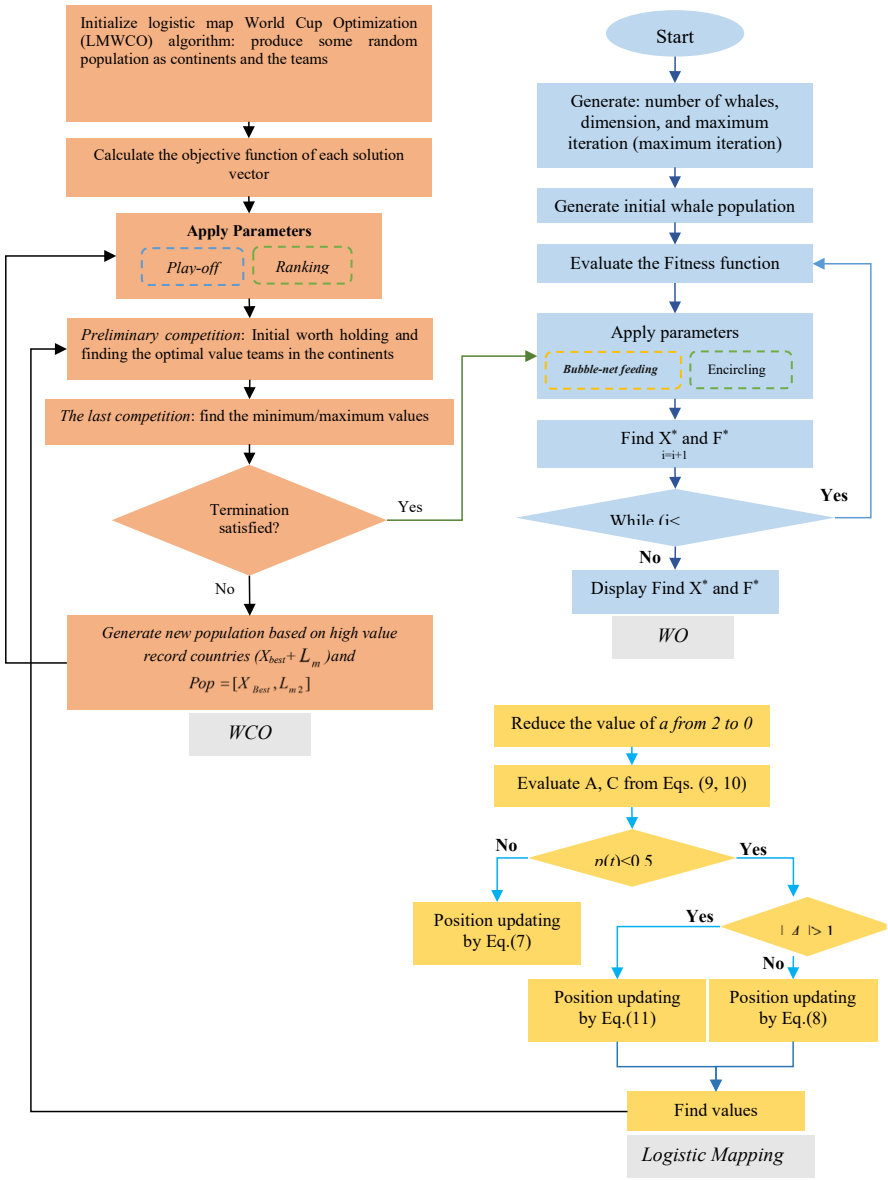


Fig. 1. The workflowof the LMSSOWCO optimizer

Fig. 2. several sample images of the UFPR-ALPR data collection

Table 1. upper and the lower bounds for the hyperparameters for LMSSOWCO

Search boundaries	α	Conv 1		Conv 2		Conv 3	
		Number	size	Number	size	Number	size
Upper and lower bounds	$1e^{-6}$	1	3	1	3	1	3
Upper and lower bounds	$1e^{-3}$	9	12	10	22	10	31

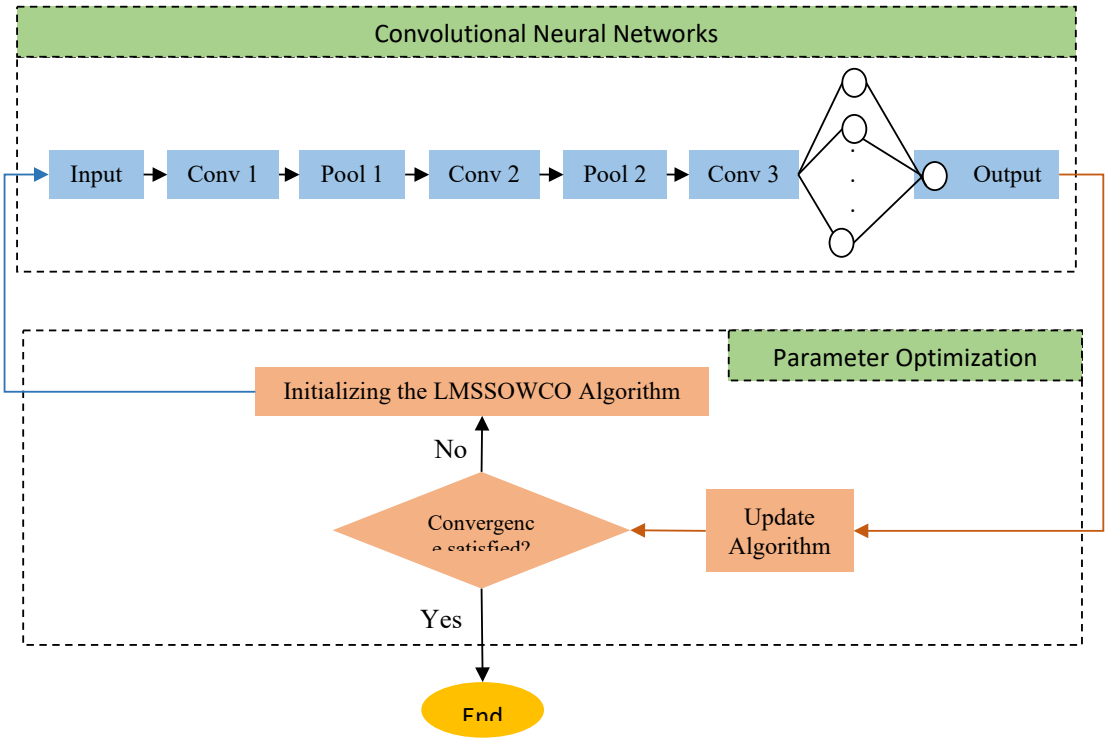
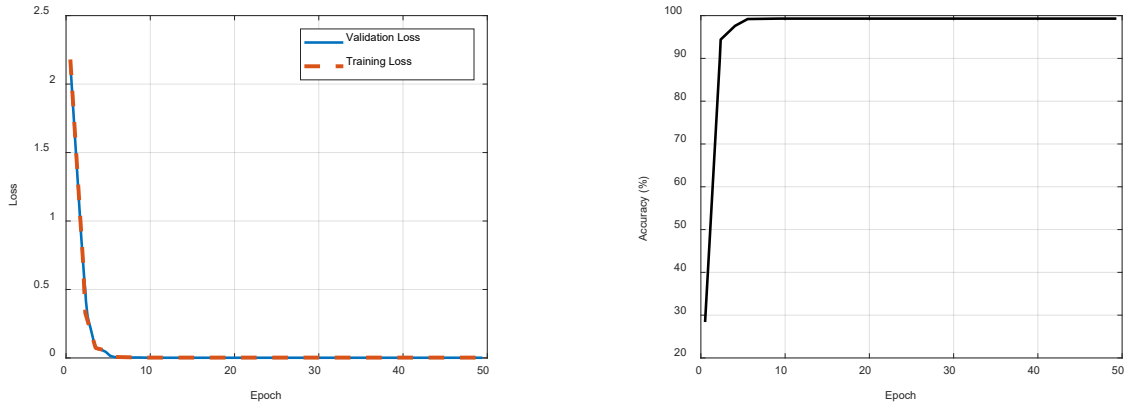


Fig. 3. Block diagram of the parameter optimization of the CNN configuration based on LMSSOWCO Algorithm.

Table 2. Optimized configuration of the presented CNN for UFPR-ALPR dataset experiment

Layer	Parameter	Configuration	Layer	Parameter	Configuration
Conv 1	Kernel size	10×10	Pool 2	Kernel size	2×2
	Num. kernels	9		Step	2
	Step	1		Output size	12×12×5
	Input size	60×60	Conv 3	Kernel size	5×5
	Output size	50×50×9		Num. kernels	7
Pool 1	Kernel size	2×2		Step	1
	Step	2		Output size	9×9×7
	Output size	25×25×9	FC	Output size	567
Conv 2	Kernel size	5×5	Output	Output size	1
	Num. kernels	5			
	Step	1			
	Output size	20×20×5			



(A) (B)

Fig. 4. Training curve of using UFPR-ALPR Dataset: (A) algorithm Loss and (B) algorithm accuracy

Table 3. choice of CNN configuration by several LMSSOWCO tests

Iteration	Pop. Size	Time of LMSSOWCO training	Max. Accuracy
10	10	10.13	99.95
10	15	15.25	99.68
10	20	19.36	99.75
15	10	09.15	99.75
15	15	08.74	99.40
15	20	20.64	99.74
20	10	12.73	99.84
20	15	17.64	99.84
20	20	23.49	99.84

Table 4. time of training and the precision of the optimized CNN for different number of epochs

Epoch	10	20	30	40	50
time of training	10.2	13.4	19.23	24.32	29.35
Accuracy (%)	99.62	99.48	99.48	99.48	99.48

size 1 as the outputted FMs. We have also employed two layers for pooling of size 2 × 2 kernels for both of them with step size one.

In the proposed network for UFPR-ALPR dataset, nine kernels of size 10×10 have been selected for the conv1 layer, five kernels of size 5 × 5 for the conv2 layer, and 7 kernels of size 5 × 5 for the conv3 layer.

Afterward, joint high-ranking characteristics are trained with the CLs in the FC layer and then for the output of the network, a single neuron has been evaluated. Finally, Softmax classifier is used for the vehicle plate recognition with its feature vector.

The value for learning rate in here is employed 0.0076 for simulations using UFPR-ALPR datasets. Fig. (4) shows the training loss which is converged in the earlier stage sixth epoch. Algorithm accuracy is also shown in this figure which declares the proposed method reach to a good accuracy after the fifth epoch.

Table 4 illustrates the time of training and the precision of the optimized CNN based on LMSSOWCO for UFPR-ALPR Dataset.

A comparison of the presented technique with some other popular procedures for plate letters recognition purpose is illustrated in Table 5. The detection accuracy of the presented

technique outperforms the other feature extraction techniques.

From the results, it is clear that using LMSSOWCO technique obtains the automatically optimized combination for the hyper-parameters in the CNN configuration.

The main goal of the presented technique is to direct detecting of the plate letters based on a new optimization technique based on deep learning.

5. CONCLUSIONS

In the current study, a new technique for direct identification of the vehicle license plate letters is proposed. The method is designed to employ as a real-time system in digital cameras located at urban surveillance in smart cities. The proposed method is an accurate method based on an optimized design of the CNN. The main purpose of optimization here is to select the optimal hyperparameters (structure) for the deep network. The optimizer is a new hybrid whale optimizer and world cup optimization along with the logistic mapping to increase the system convergence. Simulations are applied to the UFPR-ALPR dataset and the achievements of the presented technique have been put in comparison with six popular techniques. Experimental showed that the presented technique has a promising result toward the other compared methods.

REFERENCES

[1] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems,* vol. 82, pp. 395-411, 2018.

[2] B. Farahani, F. Firouzi, V. Chang, M. Badaroglu, N. Constant, and K. Mankodiya, "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare," *Future Generation Computer Systems,* vol. 78, pp. 659-676, 2018.

[3] A. Dehghantanha, *Handbook of Big Data and IoT Security*: Springer, 2019.

[4] F. Saghafi and H. Kordsalari, "Suggesting the Driving Forces Behind the Effective Implementation of the Internet of Things in the IRI Railway System with Focus on Improving Safety," in *2018 9th International Symposium on Telecommunications (IST)*, 2018, pp. 375-380.

[5] R. Jalali and S. Zeinali, "Smart Flight Security in Airport Using IOT (Case Study: Airport of Birjand)," *International Journal of Computer Science and Software Engineering,* vol. 7, pp. 142-147, 2018.

[6] S. Khan, N. Islam, Z. Jan, I. U. Din, and J. J. C. Rodrigues, "A Novel Deep Learning based Framework for the Detection and Classification of Breast Cancer Using Transfer Learning," *Pattern Recognition Letters,* 2019.

[7] I. M. Baltruschat, H. Nickisch, M. Grass, T. Knopp, and A. Saalbach, "Comparison of deep learning approaches for multi-label chest X-ray classification," *Scientific reports,* vol. 9, p. 6381, 2019.

[8] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, *et al.*, "A survey on deep learning in medical image analysis," *Medical image analysis,* vol. 42, pp. 60-88, 2017.

[9] H. Rajput, T. Som, and S. Kar, "An automated vehicle license plate recognition system," *Computer,* vol. 48, pp. 56-61, 2015.

[10] B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, "Deep learning for classification of malware system call sequences," in *Australasian Joint Conference on Artificial Intelligence*, 2016, pp. 137-149.

[11] M. Fu, N. Chen, X. Hou, H. Sun, A. Abdussalam, and S. Sun, "Real-Time Vehicle License Plate Recognition Using Deep Learning," in *International Conference On Signal And Information Processing, Networking And Computers*, 2018, pp. 35-41.

[12] L. Zhong, L. Hu, and H. Zhou, "Deep learning based multi-temporal crop classification," *Remote sensing of environment,* vol. 221, pp. 430-443, 2019.

[13] M. Admi, S. El Fkihi, and R. Faizi, "A Novel MSER based Method for Detecting Text in License Plates," in *Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications*, 2018, p. 33.

[14] K. Tejas, K. A. Reddy, D. P. Reddy, K. Bharath, R. Karthik, and M. R. Kumar, "Efficient License Plate Recognition System with Smarter Interpretation Through IoT," in *Soft Computing for Problem Solving*, ed: Springer, 2019, pp. 207-220.

[15] O. Abedinia, N. Amjady, and N. Ghadimi, "Solar energy forecasting based on hybrid neural network and improved metaheuristic algorithm," *Computational Intelligence,* vol. 34, pp. 241-260, 2018.

[16] P. Moallem and N. Razmjooy, "A multi layer perceptron neural network trained by invasive weed optimization for potato color image segmentation," *Trends in Applied Sciences Research,* vol. 7, p. 445, 2012.

[17] N. Razmjooy, B. S. Mousavi, and F. Soleymani, "A hybrid neural network Imperialist Competitive Algorithm for skin color segmentation," *Mathematical and Computer Modelling,* vol. 57, pp. 848-856, 2013.

[18] H. M. Kanoosh, E. H. Houssein, and M. M. Selim, "Salp Swarm Algorithm for Node Localization in Wireless Sensor Networks," *Journal of Computer Networks and Communications,* vol. 2019, 2019.

[19] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, and H. Adeli, "Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals," *Computers in biology and medicine,* vol. 100, pp. 270-278, 2018.

[20] O. Abedinia, N. Amjady, and A. Ghasemi, "A new metaheuristic algorithm based on shark smell optimization," *Complexity,* vol. 21, pp. 97-116, 2016.

[21] N. Razmjooy, M. Khalilpour, and M. Ramezani, "A New Meta-Heuristic Optimization Algorithm Inspired by FIFA World Cup Competitions: Theory and Its Application in PID Designing for AVR System," *Journal*

of Control, Automation and Electrical Systems, vol. 27, pp. 419-440, 2016.
[22] N. Razmjooy and M. Ramezani, "An Improved Quantum Evolutionary Algorithm Based on Invasive Weed Optimization," *Indian J. Sci. Res,* vol. 4, pp. 413-422, 2014.
[23] N. Ghadimi, "Genetically tuning of lead-lag controller in order to control of fuel cell voltage," *Scientific Research and Essays,* vol. 7, pp. 3695-3701, 2012.
[24] S. Mirjalili, "Genetic Algorithm," in *Evolutionary Algorithms and Neural Networks*, ed: Springer, 2019, pp. 43-55.
[25] B. S. Mousavi and F. Soleymani, "Semantic image classification by genetic algorithm using optimised fuzzy system based on Zernike moments," *Signal, Image and Video Processing,* vol. 8, pp. 831-842, 2014.
[26] N. Ghadimi, M. Afkousi-Paqaleh, and A. Nouri, "PSO based fuzzy stochastic long-term model for deployment of distributed energy resources in distribution systems with several objectives," *IEEE Systems Journal,* vol. 7, pp. 786-796, 2013.
[27] A. Jalili and N. Ghadimi, "Hybrid harmony search algorithm and fuzzy mechanism for solving congestion management problem in an electricity market," *Complexity,* vol. 21, pp. 90-98, 2016.
[28] J. C. Bansal, "Particle Swarm Optimization," in *Evolutionary and Swarm Intelligence Algorithms*, ed: Springer, 2019, pp. 11-23.
[29] P. Moallem and N. Razmjooy, "Optimal threshold computing in automatic image thresholding using adaptive particle swarm optimization," *Journal of applied research and technology,* vol. 10, pp. 703-712, 2012.
[30] N. Razmjooy and M. Ramezani, "Training Wavelet Neural Networks Using Hybrid Particle Swarm Optimization and Gravitational Search Algorithm for System Identification."
[31] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Computing,* vol. 22, pp. 1-15, 2018.
[32] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software,* vol. 95, pp. 51-67, 2016.
[33] P. S. Bandaghiri, N. Moradi, and S. S. Tehrani, "Optimal tuning of PID controller parameters for speed control of DC motor based on world cup optimization algorithm," *parameters,* vol. 1, p. 2, 2016.
[34] N. Razmjooy, A. Madadi, and M. Ramezani, "Robust Control of Power System Stabilizer Using World Cup Optimization Algorithm."
[35] N. Razmjooy and M. Shahrezaee, "Solving Ordinary Differential Equations using World Cup Optimization Algorithm."
[36] N. Razmjooy, F. R. Sheykhahmad, and N. Ghadimi, "A hybrid neural network–world cup optimization algorithm for melanoma detection," *Open Medicine,* vol. 13, pp. 9-16, 2018.
[37] M. Shahrezaee, "Image segmentation based on world cup optimization algorithm," *Majlesi Journal of Electrical Engineering,* vol. 11, 2017.
[38] Y. Luo, J. Yu, W. Lai, and L. Liu, "A novel chaotic image encryption algorithm based on improved baker map and logistic map," *Multimedia Tools and Applications,* pp. 1-21, 2019.
[39] C. Schymura and D. Kolossa, "Learning Dynamic Stream Weights for Linear Dynamical Systems Using Natural Evolution Strategies," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 7893-7897.
[40] UFPR-ALPR. (2019). *UFPR-ALPR* Available: https://web.inf.ufpr.br/vri/databases/ufpr-alpr/
[41] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, *et al.*, "A robust real-time automatic license plate recognition based on the YOLO detector," in *2018 International Joint Conference on Neural Networks (IJCNN)*, 2018, pp. 1-10.
[42] S.-K. Chou, M.-K. Jiau, and S.-C. Huang, "Stochastic set-based particle swarm optimization based on local exploration for solving the carpool service problem," *IEEE transactions on cybernetics,* vol. 46, pp. 1771-1783, 2016.
[43] L. Bottou and O. Bousquet, "The tradeoffs of large scale learning," in *Advances in neural information processing systems*, 2008, pp. 161-168.
[44] A. P. U. Siahaan, "Vehicle Plate Recognition using Template Matching," 2018.
[45] S. Yu, B. Li, Q. Zhang, C. Liu, and M. Q.-H. Meng, "A novel license plate location method based on wavelet transform and EMD analysis," *Pattern Recognition,* vol. 48, pp. 114-125, 2015.
[46] K. M. A. Yousef, M. Al-Tabanjah, E. Hudaib, and M. Ikrai, "SIFT based automatic number plate recognition," in *2015 6th International Conference on Information and Communication Systems (ICICS)*, 2015, pp. 124-129.
[47] D. F. Llorca, R. Arroyo, and M. A. Sotelo, "Vehicle logo recognition in traffic images using HOG features and SVM," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013, pp. 2229-2234.
[48] T. K. Cheang, Y. S. Chong, and Y. H. Tay, "Segmentation-free vehicle license plate recognition using ConvNet-RNN," *arXiv preprint arXiv:1701.06439,* 2017.
[49] C. Gou, K. Wang, Y. Yao, and Z. Li, "Vehicle license plate recognition based on extremal regions and restricted Boltzmann machines," *IEEE transactions on intelligent transportation systems,* vol. 17, pp. 1096-1107, 2015.