Tech Science Press

Hybrid Metaheuristics Based License Plate Character Recognition in Smart City

Esam A. AlQaralleh[1], Fahad Aldhaban[2], Halah Nasseif[2], Bassam A.Y. Alqaralleh[2,*] and Tamer AbuKhalil[3]

[1]School of Engineering, Princess Sumaya University for Technology, Amman, 11941, Jordan
[2]MIS Department, College of Business Administration, University of Business and Technology, Jeddah, 21448, Saudi Arabia
[3]Department of Computer Science, Faculty of Information Technology, Al-Hussein Bin Talal University, Ma'an, 71111, Jordan
*Corresponding Author: Bassam A.Y. Alqaralleh. Email: b.alqaralleh@ubt.edu.sa

Abstract: Recent technological advancements have been used to improve the quality of living in smart cities. At the same time, automated detection of vehicles can be utilized to reduce crime rate and improve public security. On the other hand, the automatic identification of vehicle license plate (LP) character becomes an essential process to recognize vehicles in real time scenarios, which can be achieved by the exploitation of optimal deep learning (DL) approaches. In this article, a novel hybrid metaheuristic optimization based deep learning model for automated license plate character recognition (HMODL-ALPCR) technique has been presented for smart city environments. The major intention of the HMODL-ALPCR technique is to detect LPs and recognize the characters that exist in them. For effective LP detection process, mask regional convolutional neural network (Mask-RCNN) model is applied and the Inception with Residual Network (ResNet)-v2 as the baseline network. In addition, hybrid sunflower optimization with butterfly optimization algorithm (HSFO-BOA) is utilized for the hyperparameter tuning of the Inception-ResNetv2 model. Finally, Tesseract based character recognition model is applied to effectively recognize the characters present in the LPs. The experimental result analysis of the HMODL-ALPCR technique takes place against the benchmark dataset and the experimental outcomes pointed out the improved efficacy of the HMODL-ALPCR technique over the recent methods.

1 Introduction

Continual urbanization possesses difficult problems on living quality and sustainable development of urban residents in smart cities [1]. The idea of smart cities is to make very effective usages of scarce resources, and enhance the quality of public services and citizen lives [2]. With the growth of embedded devices, Internet of Things (IoT), for example, mobiles phones sensors, Radio Frequency Identifications (RFIDs), and actuators, constructed into all the fabrics of urban environment and coupled together [3]. Several smart city applications were deployed and developed, e.g., smart healthcare, intelligent transportation, public safety, and environment monitoring, etc. License plate recognition (LPR) system is often a great advantage for parking, traffic, cruise control, and toll management applications [4]. Regarding security management and monitoring of any region or place, LPR system is utilized as tracing assistance to help eyes for the safety teams. In terms of law and safety enforcement, LPR system plays an important part in safeguarding, monitoring the borders, and physical intrusion [5]. Different types of LPR systems are introduced by utilizing many smart computation models to attain efficiency and accuracy.

Various recognition approaches were described to implement many intermediate processing phases at the time of Region of Interest (ROI) extraction. Nonetheless, fraud situations such as replacement and alteration, LPR system is related to intelligence method for effectiveness [6]. The first phase of LPR systems is plate localization that is related to a recognition method for license plates (LP) in the input image. Algorithms like threshold or edge detection [7] are utilized by the video sequence. But Gabor filter is taken into account as a promising method for plate recognition through RBG image [8], whereas the previous one employs grey scale conversion for binary images. As well, generate histogram by means of vertical and horizontal prediction on the input images to recognize ROI based histogram that identifies the plates through multiple objects. Also, Hough conversion is employed for finding the edges bounded by the number plate [9].

This paper develops an intelligent hybrid metaheuristic optimization based deep learning model for automated license plate character recognition (HMODL-ALPCR) technique that has been presented for smart city environments. The HMODL-ALPCR technique involves mask regional convolutional neural network (Mask-RCNN) model is applied and the Inception with Residual Network (ResNet)-v2 as the baseline network. In addition, hybrid sunflower optimization with butterfly optimization algorithm (HSFO-BOA) is utilized for the hyperparameter tuning of the Inception-ResNetv2 model. Finally, Tesseract based character recognition model is applied to effectively recognize the characters present in the LPs. The experimental result analysis of the HMODL-ALPCR technique takes place against the benchmark dataset.

2 Literature Review

Deep learning (DL), a comparatively young learning model in the CI family, has its source from Artificial Neural Networks (ANN). It enables computation model that is made up of multi-processing layers to learn representation of information with multi stages of abstraction, also it is capable of discovering complex structures from natural information in their new form without needing complex feature tuning and engineering [10]. In comparison with conventional ML models, DL method could develop exceptionally complex functions over layers of nonlinear conversion trainable from the start to the termination. In [11], proposed a cascaded DL method for constructing an effective Automatic license plate (ALP) recognition and detection method for the vehicle of northern Iraq. The LP in northern Iraq contains country region, plate number, and city region. Initially, the presented technique uses various pre-processing methods like adaptive image contrast enhancement

and Gaussian filtering for making the input image better suitable for additional processing. Next, a deep semantic segmentation network is utilized for determining the three LPs of the input images. Then, Segmentation is performed by using deep encoder-decoder network framework.

Chen [12] resolves the issues of car LP recognition through a YOLO darknet DL architecture. In the work, we employ YOLO seven convolution layers to identify an individual class. The recognition model is a sliding-window method. The object is to identify Taiwan car LP. Izidio et al. [13] introduced a method to engineer systems to recognize and detect Brazilian LP with CNN i.e., appropriate for embedded systems. The resultant systems detect LP in the captured image through Tiny YOLOv3 framework and recognize its character with second convolution networks trained on synthetic image and finetuned with actual LP image. Pustokhina et al. [14] proposed an efficient DL-based VLPR method with optimum K-means (OKM) cluster-based classification and CNN based detection method. The presented method works on three major phases such as LP segmentation, detection with OKM cluster method, and LP number detection with CNN method. In the initial phase, LP detection and localization method take place.

3 The Proposed Model

In this article, an automated HMODL-ALPCR technique has been presented to detect LPs and recognize the characters that exist in them for smart city environments. The HMODL-ALPCR technique involves Mask-RCNN for the detection of LPs and Inception with ResNet-v2 as the baseline network. Moreover, the HSFO-BOA is utilized for the hyperparameter tuning of the Inception-ResNetv2 model. Lastly, Tesseract based character recognition model is applied to effectively recognize the characters present in the LPs.

3.1 Phase I: Mask RCNN Based LP Detection Process

The Mask *R*-CNN technique is melioration dependent upon Faster *R*-CNN detection technique that presents the fully convolutional network (FCN) for generating masks. During the real time target detection procedure, the pixel of target are categorized accurately, and after that, the contour of target was judged. An image was primary input as to the backbone network consisting of Inception with ResNet v2 and FPN [15]. The structure of Mask RCNN is shown in Fig. 1. The backbone network removes any shared feature map (FM) which integrates the coordinate data of detection target place and the form texture data. Afterward, the RPN region offer network utilizes a sliding window for traversing this FM for generating many anchor frames with group of fixed scale and aspect ratio.
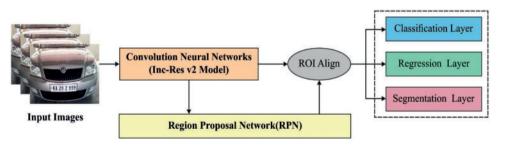

Figure 1: Mask RCNN structure

Afterward, the non-maximum suppression (NMS) technique was utilized for selecting the anchor box with superior score [16]. During the RoIAlign layer of Mask *R*-CNN technique, the quantization function from the feature aggregation procedure was changed by bilinear interpolation technique that

keeps the issue of mismatching and enhancing the accuracy of detecting and segmenting. In the trained procedure, the Mask R-CNN technique determines the multitask loss function to all sampled RoI as

$$L = L_{cls} + L_{box} + L_{mask}. \tag{1}$$

L_{cls} implies the classifier error, L_{box} refers the recognition error, and L_{mask} stands for the segmented error. L_{cls} and L_{box} from the Mask, R-CNN is determined as:

$$L' = \frac{1}{N_{cls}} \sum_i L_{cls}\left(p_i, p_i^*\right) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}\left(t_i, t_i^*\right). \tag{2}$$

where p_i signifies the forecasted probability of ith target on anchor point. p_i^* has referred to as the sign of anchor point samples. If the anchor point instance was positive, p_i^* is 1; else, it can be 0. Combined of t_i and t_i^* are vectors consisting of 4 translation and scaling parameters that correspondingly. The weight N_{cls}, N_{reg}, and λ control the 2 losses for keeping balance. The classification and regression losses are determined as:

$$L_{cls}\left(p_i, p_i^*\right) = -\log\left[p_i p_i^* + \left(1 - p_i^*\right)\left(1 - p_i\right)\right] \tag{3}$$

$$L_{reg}\left(t_i, t_i^*\right) = smooth_L\left(x\right)\left(t_i - t_i^*\right). \tag{4}$$

where smooth (x) refers the robust loss that is referred as the translation χ of modified frame on the horizontal axis at anchor points. It can be demonstrated as:

$$smooth_L\left(x\right) = \begin{cases} 0.5x^2 & |x| < l, \\ |x| - 0.5 & otherwise \end{cases}. \tag{5}$$

L_{mask} in Mask, R-CNN is the average binary cross entropy function which explains the loss of semantic segmentation branch. During the mask branch, an input FM is resultant as to $k \times m \times m$ formats then process, and k and m, correspondingly, controls the dimensional and scale of the FMs. The $|x| < l$, comparative entropy was reached by the pixel-by-pixel sigmoid computation of resultant FM, and the average entropy error is L_{mask}.

In the Mask RCNN model, the Inception with ResNetv2 is utilized as the baseline network. DL concentrates on effectiveness as a human mind. If the child was trained on distinct animals, an arbitrary image was created from the mind of child which is a dog as follows and cat as follows, and from the future, the child is identified as this animal. In DL work on a similar rule. Transfer learning (TL) is the next stage from DL. In trained a NN technique needs several times and various runs for capturing the accurate weight based on this model condition. It can be tedious works and could not be simple to student a novel to the field for entering TL. The TL manages the methods led by field experts to the public that skip the necessity of determining compatible weight and carry on to next stage of trained method on novel input data. An Inception ResNetV2 is introduced [17] by combining the 2 most famous DCNN, Inception and ResNet, and utilizing batch-normalization (BN) to the convention layer before summation. The leftover components are specially employed for enabling a superior amount of Inception block and consequence, deeper method. As already mentioned, the extremely noticeable complexity compared with highly deep network is the trained phase. It can be managed to utilize remaining connection. But, an enormous amount of filters were utilized from the system, the remaining was scaled down in an effectual manner for dealing with the trained complexity. If the amount of strainer surpasses 1000, the remaining variants encounter variability, and the network could not be trained. Thus the outcome, the remaining supports are scaled from network trained stabilization.

The sigmoid function was numerically measured which is the feature of transmitting some actual value to range amongst zero and one, shaped like the letter "S." The logistic function was another name to the sigmoid function. The sigmoid function is written as:

$$X = \frac{1}{1 + e^{-Y}}. \tag{6}$$

An important benefit of the sigmoid function is that it occurs amongst 2 points, 0 and 1. Thus the result can be most effective from this technique where it is required for anticipating probability as outcome. It can be selected this function as the possibility of something happening is only amongst zero and one.

3.2 Phase II: Design of HSFO-BOA Based Hyperparameter Tuning

In order to optimally adjust the hyperparameters involved in the Inception with ResNetv2 model, the HSFO-BOA is derived. A sunflower lifecycle is reliable: as they arise, accompany the sun daily and the needles of clock. Here, the inverse square law radiation is another key nature-based optimization. The heat quantity Q received by the plant is shown as follows [18]:

$$Q_i = \frac{P}{4\pi r_i^2} \tag{7}$$

While P indicates the source power and r_i represent the distance between the existing paramount and the plant i.

$$\vec{S}_i = \frac{X^* - X_i}{\|X^* - X_i\|}, i = 1, 2, \cdots, n_p, \tag{8}$$

The sunflower stride in the direction s can be evaluated as follows:

$$d_i = \lambda \times P_i \left(\|X_i - X_{i-1}\|\right) \times \|X_i - X_{i-1}\| \tag{9}$$

Here, λ shows the perpetual value that determines a "inertial" dislocation of the plant, $P_i(\|X_i - X_{i-1}\|)$ indicates the possibility of pollination as follows:

$$d_{\max} = \frac{\|X_{\max} - X_{\min}\|}{2 \times N_{pop}} \tag{10}$$

In the equation, $X_{\max}$ and $X_{\min}$ denotes the upper and lower limits, and N_{pop} indicates the overall amount of plants:

$$\vec{X}_{i+1} = \vec{X}_i + d_i \times \vec{s}_i \tag{11}$$

The process initiates with population generation that may be random or even. Corresponding individual ratings assist in choosing which one would be moved towards the sun. Next, each entity will position itself into the sun and move in a random manner. However, it is proposed to include the capacity to function with different suns in a future version, now it is restricted to the study. Paramount plants would pollinate around the sun.

For improving the efficacy of the SFO algorithm, the HSFO-BOA is derived by the integration of BOA to it. The BOA imitates the natural behavior of the butterflies on food sources finding and mating. This approach uses two distinct navigation patterns for searching the domains [19]. In the exploration stage ($r_1 \leq p$), butterflies move to the optimal butterfly of the colony whereas in the exploitation stage ($r_1 > p$), butterfly performs an arbitrary search within the searching space by

moving to a random butterfly in the colony. The mathematical expression of both patterns are given in the following:

When $r_1 \leq p$, the global search process becomes

$$^{t+1}X_i = {}^tX_i + \left(r_2^2 \times g^* - {}^tX_i\right) \times \varphi_i \tag{12}$$

When $r_1 > p$, the local search process becomes

$$^{t+1}X_i = {}^tX_i + (r_3^2 \times {}^tX_j - {}^tX_k \times \varphi_i \tag{13}$$

Here, t and $t+1$ indicate the present and upgraded states. As well, position of optimal butterfly in the colony has been demonstrated as g^*, and ${}^tX_i$ and ${}^tX_k$, are positions of two arbitrarily designated butterflies; r_1, r_2 and r_3 indicates three random scalars uniformly chosen within [0,1],$\cdot$ φ_i represent the fragrance factor and it is determined by the following equation:

$$\varphi_i = cI^a \tag{14}$$

Whereas, φ_i indicates the fragrance magnitude for *ith* butterfly; c denotes a coefficient, I, and a shows the intensity of the stimulus and the fluctuating absorption degree. I is related to the objective function, and for ith butterfly, it is considered $f(X_i)$, whereas f return objective function of the problem. The a and c coefficients are designated within [0,1],$\cdot$ p indicates the likelihood switch that describes the search behavior.

3.3 Phase III: Tesseract Based Character Recognition

Primarily, Adaptive Thresholding was implemented for changing the image as to binary version utilizing Otsu's technique [20]. The page layout analysis is the next stage and was implemented by removing the text block in the region. Afterward, the baselines of all lines were identified and the texts were separated as words with the application of finite as well as fuzzy spaces. During the next phase, the character summaries are removed in the words. The text detection was introduced as 2-pass technique. Primary pass, a word detection was implemented with the application of static classifier. All the words are passed suitably for adaptive classifying from the procedure of trained data. The secondary pass was run on the page utilizing a novel adaptive classifier technique where the words are not studied comprehensively for re-examining the modules.

4 Experimental Validation

The performance validation of the HMODL-ALPCR technique takes place using three benchmark datasets namely FZU Cars, Stanford Cars, and HumAIn 2019 dataset. Few sample images are depicted in Fig. 2.

Fig. 3 illustrates the sample results obtained by the HMODL-ALPCR technique. From the figure, it is clear that the HMODL-ALPCR technique has proficiently detected the LP and recognized the characters.

Tab. 1 offers the LP detection outcome analysis of the HMODL-ALPCR technique under distinct epochs. Fig. 4 examines the LP detection result analysis of the HMODL-ALPCR technique under distinct epochs on FZU Cars dataset. With 100 epochs, the HMODL-ALPCR technique has offered $prec_n, reca_l, F_{score}$, and mAP of 99.05%, 99.07%, 98.91%, and 98.56% respectively. Also, with 200 epochs, the HMODL-ALPCR technique has attained $prec_n, reca_l, F_{score}$, and mAP of 99.05%, 99.54%, 98.71%, and 98.42% respectively. Similarly, with 300 epochs, the HMODL-ALPCR technique has provided

$prec_n$, $reca_l$, F_{score}, and mAP of 99.05%, 99.42%, 98.73%, and 97.86% respectively. Likewise, with 400 epochs, the HMODL-ALPCR technique has exhibited $prec_n$, $reca_l$, F_{score}, and mAP of 99.00%, 99.40%, 98.76%, and 98.37% respectively.

Figure 2: Sample images

Figure 3: Sample visualization results of HMODL-ALPCR technique

Table 1: LP detection results of HMODL-ALPCR technique

No. of Epochs	Precision	Recall	F1-score	mAP
FZU Cars Dataset				
Epoch-100	99.05	99.07	98.91	98.56
Epoch-200	99.05	99.54	98.71	98.42
Epoch-300	99.05	99.42	98.73	97.86
Epoch-400	99.00	99.40	98.76	98.37

(Continued)

Table 1: Continued

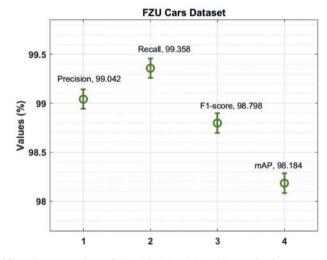| No. of Epochs	Precision	Recall	F1-score	mAP
Epoch-500	99.06	99.36	98.88	97.71
Average	99.04	99.36	98.80	98.18
Stanford Cars Dataset				
Epoch-100	98.99	99.00	97.86	96.95
Epoch-200	98.12	99.27	98.46	97.81
Epoch-300	97.74	98.96	98.54	96.32
Epoch-400	98.49	99.11	98.48	97.08
Epoch-500	98.94	99.11	97.71	97.34
Average	98.46	99.09	98.21	97.10
HumAIn 2019 Dataset				
Epoch-100	98.57	99.14	99.31	97.74
Epoch-200	98.47	98.97	98.72	98.32
Epoch-300	99.34	99.13	98.63	98.11
Epoch-400	98.94	99.25	98.82	98.40
Epoch-500	98.72	98.96	98.97	98.11
Average	98.81	99.09	98.89	98.14

Figure 4: Classification results of HMODL-ALPCR technique on FZU Cars dataset

Fig. 5 inspects the LP detection result analysis of the HMODL-ALPCR system under different epochs on Stanford Cars dataset. With 100 epochs, the HMODL-ALPCR approach has offered $prec_n$, $reca_l$, F_{score}, and mAP of 98.99%, 99.00%, 97.86%, and 96.95% correspondingly. Besides, with 200 epochs, the HMODL-ALPCR methodology has reached $prec_n$, $reca_l$, F_{score}, and mAP of 98.12%, 99.27%, 98.46%, and 97.81% respectively. In addition, with 300 epochs, the HMODL-ALPCR system has offered $prec_n$, $reca_l$, F_{score}, and mAP of 97.74%, 98.96%, 98.54%, and 96.32% correspondingly.

Moreover, with 400 epochs, the HMODL-ALPCR methodology has demonstrated $prec_n$, $reca_l$, F_{score}, and mAP of 98.49%, 99.11%, 98.48%, and 97.08% correspondingly.
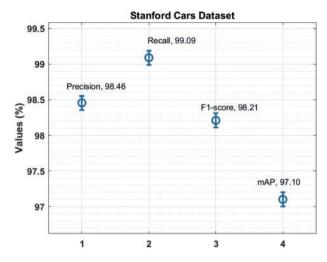

Figure 5: Classification results of HMODL-ALPCR technique on Standford Cars dataset

Fig. 6 demonstrates the LP detection result analysis of the HMODL-ALPCR system under distinct epochs on HumAIn 2019 dataset. With 100 epochs, the HMODL-ALPCR technique has obtainable $prec_n$, $reca_l$, F_{score}, and mAP of 98.57%, 99.14%, 99.31%, and 97.74% respectively. Along with that, with 200 epochs, the HMODL-ALPCR approach has reached $prec_n$, $reca_l$, F_{score}, and mAP of 98.47%, 98.97%, 98.72%, and 98.32% respectively. Similarly, with 300 epochs, the HMODL-ALPCR technique has accessible $prec_n$, $reca_l$, F_{score}, and mAP of 99.34%, 99.13%, 98.63%, and 98.11% correspondingly. At last, with 400 epochs, the HMODL-ALPCR system has exhibited $prec_n$, $reca_l$, F_{score}, and mAP of 98.94%, 99.25%, 98.82%, and 98.40% respectively.
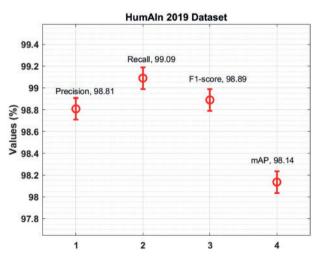

Figure 6: Classification results of HMODL-ALPCR technique on HumAIn 2019 dataset

Tab. 2 and Fig. 7 investigate the comparison study of the HMODL-ALPCR technique with recent methods on the test FZU Cars dataset. The results demonstrated that the CNN-VGG16 and DL-ResNet50 techniques have obtained lower performance with the minimal values of $prec_n$, $reca_l$, F_{score},

and *mAP*. In line with, the DL-ResNet101 and HT-SSA-CNN techniques have attained slightly enhanced values of *prec_n*, *reca_l*, *F_score*, and *mAP*. Next to that, the DL-VLPNR and OKM-CNN techniques have reached reasonable values of *prec_n*, *reca_l*, *F_score*, and *mAP*. However, the HMODL-ALPCR technique has outperformed the other methods with the maximum *prec_n*, *reca_l*, *F_score*, and *mAP* of 99.04%, 99.36%, 98.80%, and 98.18% respectively.

Table 2: Comparative LP detection results of HMODL-ALPCR technique on FZU Cars dataset

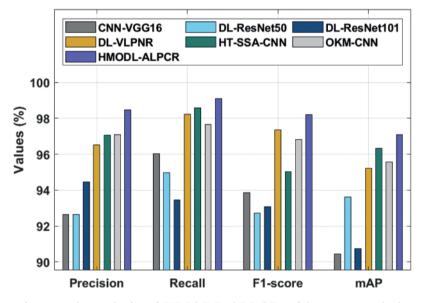| Model	Precision	Recall	F1-score	mAP
CNN-VGG16	93.97	96.56	94.43	92.43
DL-ResNet50	93.26	93.27	94.80	90.66
DL-ResNet101	96.15	96.88	97.03	93.70
DL-VLPNR	98.88	96.83	96.76	94.96
HT-SSA-CNN	96.81	99.08	96.83	97.38
OKM-CNN	97.49	98.95	96.89	96.68
HMODL-ALPCR	99.04	99.36	98.80	98.18

Figure 7: LP detection result analysis of HMODL-ALPCR with recent techniques on FZU Cars dataset

Tab. 3 and Fig. 8 examine the comparison study of the HMODL-ALPCR approach on the test Stanford Cars dataset. The outcomes exhibited that the CNN-VGG16 and DL-ResNet50 algorithms have reached lesser performance with the minimal values of *prec_n*, *reca_l*, *F_score*, and *mAP*. Likewise, the DL-ResNet101 and HT-SSA-CNN techniques have attained slightly enhanced values of *prec_n*, *reca_l*, *F_score*, and *mAP*. Followed by, the DL-VLPNR and OKM-CNN techniques have reached reasonable values of *prec_n*, *reca_l*, *F_score*, and *mAP*. At last, the HMODL-ALPCR system has outperformed the other methods with the maximum *prec_n*, *reca_l*, *F_score*, and *mAP* of 98.46%, 99.09%, 98.21%, and 97.10% respectively.

Table 3: Comparative LP detection results of HMODL-ALPCR technique on Stanford Cars dataset

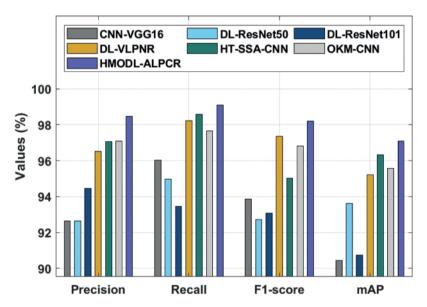| Model	Precision	Recall	F1-score	mAP
CNN-VGG16	92.65	96.04	93.85	90.44
DL-ResNet50	92.64	94.96	92.72	93.62
DL-ResNet101	94.47	93.46	93.07	90.74
DL-VLPNR	96.51	98.23	97.37	95.21
HT-SSA-CNN	97.06	98.58	95.02	96.32
OKM-CNN	97.09	97.67	96.83	95.57
HMODL-ALPCR	98.46	99.09	98.21	97.10

Figure 8: LP detection result analysis of HMODL-ALPCR with recent techniques on Stanford Cars dataset

Tab. 4 and Fig. 9 examine the comparison study of the HMODL-ALPCR approach on the HumAIn 2019 dataset. The outcomes exhibited that the CNN-VGG16 and DL-ResNet50 systems have obtained lesser performance with the reduced values of $prec_n$, $reca_l$, F_{score}, and mAP. Besides, the DL-ResNet101 and HT-SSA-CNN techniques have attained somewhat enhanced values of $prec_n$, $reca_l$, F_{score}, and mAP. Afterward, the DL-VLPNR and OKM-CNN algorithms have obtained reasonable values of $prec_n$, $reca_l$, F_{score}, and mAP. But, the HMODL-ALPCR system has outperformed the other algorithms with the maximal $prec_n$, $reca_l$, F_{score}, and mAP of 98.81%, 99.09%, 98.89%, and 98.14% correspondingly.

After examining the above mentioned tables and figures, it is obvious that the HMODL-ALPCR technique has outperformed the other techniques on all the datasets.

Table 4: Comparative LP detection results of HMODL-ALPCR technique on HumAIn 2019 dataset

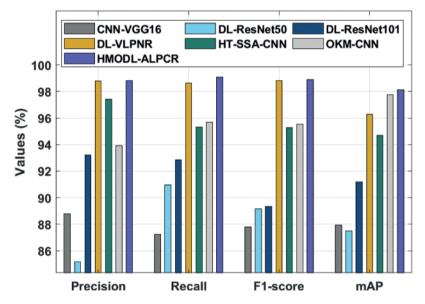| Model	Precision	Recall	F1-score	mAP
CNN-VGG16	88.80	87.24	87.79	87.95
DL-ResNet50	85.16	90.96	89.16	87.51
DL-ResNet101	93.20	92.84	89.34	91.19
DL-VLPNR	98.77	98.64	98.81	96.28
HT-SSA-CNN	97.43	95.33	95.27	94.71
OKM-CNN	93.91	95.69	95.54	97.76
HMODL-ALPCR	98.81	99.09	98.89	98.14

Figure 9: LP detection result analysis of HMODL-ALPCR with recent techniques on HumAIn 2019 dataset

5 Conclusion

In this article, an automated HMODL-ALPCR technique has been presented to detect LPs and recognize the characters that exist in them for smart city environments. The HMODL-ALPCR technique involves Mask-RCNN for the detection of LPs and Inception with ResNet-v2 as the baseline network. Moreover, the HSFO-BOA is utilized for the hyperparameter tuning of the Inception-ResNetv2 model. Lastly, Tesseract based character recognition model is applied to effectively recognize the characters present from the LPs. The experimental result analysis of the HMODL-ALPCR technique takes place against the benchmark dataset and the experimental outcomes pointed out the improved efficacy of the HMODL-ALPCR technique on existing techniques. In future, the detection performance can be improvised by the design of hybrid DL models for smart city environments.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] W. Wang, S. De, Y. Zhou, X. Huang and K. Moessner, "Distributed sensor data computing in smart city applications," in *2017 IEEE 18th Int. Symp. on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Macau, China, pp. 1–5, 2017.

[2] Y. Tian and L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," in *2015 IEEE Int. Conf. on Smart City/SocialCom/SustainCom (SmartCity)*, Chengdu, China, pp. 153–158, 2015.

[3] V. Gnanaprakash, N. Kanthimathi and N. Saranya, "Automatic number plate recognition using deep learning," *IOP Conference Series: Materials Science and Engineering*, vol. 1084, no. 1, pp. 012027, 2021.

[4] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.

[5] D. Zang, Z. Chai, J. Zhang, D. Zhang and J. Cheng, "Vehicle license plate recognition using visual attention model and deep learning," *Journal of Electronic Imaging*, vol. 24, no. 3, pp. 033001, 2015.

[6] W. Weihong and T. Jiaoyang, "Research on license plate recognition algorithms based on deep learning in complex environment," *IEEE Access*, vol. 8, pp. 91661–91675, 2020.

[7] Hendry and R. C. Chen, "Automatic license plate recognition via sliding-window darknet-yolo deep learning," *Image and Vision Computing*, vol. 87, no. 2, pp. 47–56, 2019.

[8] D. M. F. Izidio, A. P. A. Ferreira and E. N. S. Barros, "An embedded automatic license plate recognition system using deep learning," in *2018 VIII Brazilian Symp. on Computing Systems Engineering (SBESC)*, Salvador, Brazil, pp. 38–45, 2018.

[9] S. Lee, K. Son, H. Kim and J. Park, "Car plate recognition based on CNN using embedded system with GPU," in *2017 10th Int. Conf. on Human System Interactions (HSI)*, Ulsan, South Korea, pp. 239–241, 2017.

[10] S. Alghyaline, "Real-time Jordanian license plate recognition using deep learning," *Journal of King Saud University-Computer and Information Sciences*, pp. S1319157820305152, 2020. DOI 10.1016/j.jksuci.2020.09.018.

[11] N. Omar, A. Sengur and S. G. S. Al-Ali, "Cascaded deep learning-based efficient approach for license plate detection and recognition," *Expert Systems with Applications*, vol. 149, pp. 113280, 2020.

[12] Hendry and R. C. Chen, "Automatic license plate recognition via sliding-window darknet-yolo deep learning," *Image and Vision Computing*, vol. 87, pp. 47–56, 2019.

[13] D. M. F. Izidio, A. P. A. Ferreira and E. N. S. Barros, "An embedded automatic license plate recognition system using deep learning," in *2018 VIII Brazilian Symp. on Computing Systems Engineering (SBESC)*, Salvador, Brazil, pp. 38–45, 2018.

[14] I. V. Pustokhina, D. A. Pustokhin, J. J. P. C. Rodrigues, D. Gupta, A. Khanna *et al.,* "Automatic vehicle license plate recognition using optimal k-means with convolutional neural network for intelligent transportation systems," *IEEE Access*, vol. 8, pp. 92907–92917, 2020.

[15] D. H. Chen, Y. D. Cao and J. Yan, "Towards pedestrian target detection with optimized mask R-CNN," *Complexity*, vol. 2020, pp. 1–8, 2020.

[16] M. M. Karim, D. Doell, R. Lingard, Z. Yin, M. C. Leu *et al.,* "A region-based deep learning algorithm for detecting and tracking objects in manufacturing plants," *Procedia Manufacturing*, vol. 39, no. 4, pp. 168–177, 2019.

[17] C. A. Ferreira, T. Melo, P. Sousa, M. I. Meyer, E. Shakibapour *et al.,* "Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2," in *Int. Conf. Image Analysis and Recognition*, Portugal, pp. 763–770, 2018.

[18] G. F. Gomes, S. S. da Cunha and A. C. Ancelotti, "A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates," *Engineering with Computers*, vol. 35, no. 2, pp. 619–626, 2019.

[19] S. Arora and S. Singh, "Butterfly optimization algorithm: A novel approach for global optimization," *Soft Computing*, vol. 23, no. 3, pp. 715–734, 2019.

[20] T. Vetriselvi, E. L. Lydia, S. N. Mohanty, E. Alabdulkreem, S. A. Otaibi *et al.,* "Deep learning based license plate number recognition for smart cities," *Computers Materials & Continua*, vol. 70, no. 1, pp. 2049–2064, 2022.