



Research article

Spotted hyena optimizer with deep learning enabled vehicle counting and classification model for intelligent transportation systems

Manal Abdullah Alohalil¹, Mashaal Maashi², Raji Faqih³, Hany Mahgoub⁴, Abdullah Mohamed⁵, Mohammed Assiri^{6,*} and Suhanda Drar⁷

¹ Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

² Department of Software Engineering, College of Computer and Information Science, King Saud University, Po box 103786, Riyadh 11543, Saudi Arabia

³ Department of Business Administration, Applied College, King Khalid University, Mohail Asser, Saudi Arabia

⁴ Department of Computer Science, College of Science & Art at Mahayil, King Khalid University, Saudi Arabia

⁵ Research Centre, Future University in Egypt, New Cairo, 11845, Egypt

⁶ Department of Computer Science, College of Sciences and Humanities- Aflaj, Prince Sattam bin Abdulaziz University, Aflaj 16273, Saudi Arabia

⁷ Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia

* **Correspondence:** Email: m.assiri@psau.edu.sa.

Abstract: Traffic surveillance systems are utilized to collect and monitor the traffic condition data of the road networks. This data plays a crucial role in a variety of applications of the Intelligent Transportation Systems (ITSs). In traffic surveillance, it is challenging to achieve accurate vehicle detection and count the vehicles from traffic videos. The most notable difficulties include real-time system operations for precise classification, identification of the vehicles' location in traffic flows and functioning around total occlusions that hamper the vehicle tracking process. Conventional video-related vehicle detection techniques such as optical flow, background subtraction and frame difference have certain limitations in terms of efficiency or accuracy. Therefore, the current study proposes to design the spotted hyena optimizer with deep learning-enabled vehicle counting and classification (SHODL-VCC) model for the ITSs. The aim of the proposed SHODL-VCC technique lies in accurate

counting and classification of the vehicles in traffic surveillance. To achieve this, the proposed SHODL-VCC technique follows a two-stage process that includes vehicle detection and vehicle classification. Primarily, the presented SHODL-VCC technique employs the RetinaNet object detector to identify the vehicles. Next, the detected vehicles are classified into different class labels using the deep wavelet auto-encoder model. To enhance the vehicle detection performance, the spotted hyena optimizer algorithm is exploited as a hyperparameter optimizer, which considerably enhances the vehicle detection rate. The proposed SHODL-VCC technique was experimentally validated using different databases. The comparative outcomes demonstrate the promising vehicle classification performance of the SHODL-VCC technique in comparison with recent deep learning approaches.

Keywords: traffic surveillance; intelligent transportation systems; deep learning; RetinaNet; Vehicle detection; Vehicle classification

1. Introduction

The fast evolution of motor vehicles and the increasing urban population have paved the way for traffic problems. The intelligent transportation system (ITS) is a potential tool to overcome the traffic issues [1]. With the growth of computer vision (CV), the Internet of Things and communication technologies, traffic surveillance has become a key technology for traffic parameter collection and it serves a vital role [2]. Traffic flow can be defined as a basic and a significant parameter in ITSs whereas counting and detecting the number of vehicles from traffic videos in a rapid and accurate fashion, is a general research area [3]. In the past, many vision methods were modelled for automatic counting of the vehicles in traffic videos. Several existing vehicle counting techniques perform the vehicle detection process on the basis of vehicle appearance and the attributes that can be positioned through foreground recognition and the vehicles are counted on the basis of vehicle detection outcomes [4]. In general, the vehicle counting methodologies rely upon traffic videos and are classified into two subtasks such as vehicle counting and vehicle detection [5].

With an exponential growth of CV technologies and artificial intelligence, the object detection techniques related to deep learning (DL) are commonly examined in recent years [6]. This technique can automatically extract the features with the help of machine learning (ML) technique; therefore, this technique is capable of powerful image abstraction and automated higher-level feature representation [7]. Nowadays, the vision-related vehicle object recognition process is classified into conventional machine vision techniques and the complicated DL techniques. The conventional machine vision techniques make use of vehicle movement to separate it from the fixed background images [8]. The utilization of deep convolutional networks (CNN) has achieved phenomenal success in the domain of vehicle object detection. CNN is highly potential in learning the image features and can execute many relevant tasks like bounding box regression and its classification [9].

The detection technique can be classified into two categories [10]. The two-stage technique generates a candidate box of objects through different techniques and categorizes the object by CNN. On the other hand, the one-stage technique does not produce a candidate box but directly converts the positioning issue of object bounding box into a regression issue for processing [11]. With the computability of hardware and the growth of DL object detection, like GPUs, it is likely to construct a DL vehicle counting method with high efficiency and accuracy though it has a few difficulties too [12].

The primary issue to be addressed is to develop a rapid vehicle detection method with optimum performance using TL without training data. Though the DL vehicle recognition methods can identify the vehicles with high accuracy levels [13], still there occurs unavoidable false detection or missing detection issues. Evading the errors, caused by such conditions, remains the primary difficulty to overcome so that the accuracy of vehicle counting can be enhanced, if the accurate of vehicle recognition is challenging to enhance [14].

The current study focuses on design and development of the Spotted Hyena Optimizer with Deep Learning Enabled Vehicle Counting and Classification (SHODL-VCC) model for ITSs. The proposed SHODL-VCC technique follows a two-stage procedure i.e., vehicle detection and vehicle classification. Primarily, the presented SHODL-VCC technique employs the RetinaNet object detector to identify the vehicles. Next, the detected vehicles are classified into different class labels with the help of the deep wavelet auto-encoder (DWAE) model. To enhance the vehicle detection performance, the spotted hyena optimizer algorithm is exploited as a hyperparameter optimizer, which considerably enhances the vehicle detection rate. The SHODL-VCC technique was experimentally validated using different databases and the results are discussed.

2. Related works

In the study conducted earlier [15], it has been recommended that the deep learning techniques can be enforced for vehicle counting in traffic videos. In this study, a technique was primarily devised for vehicle recognition related to TL to sort out the issue of lack of annotated dataset. Afterwards, depending on the vehicle recognition process, a vehicle counting technique was modelled related to merging the vehicle tracking and virtual detection processes. At last, owing to possible circumstances of false and missing detections, the authors modelled the missing alarm and false alarm suppression modules for enriching the precision of vehicle counting process. In literature [16], the authors discussed about a deep learning application for constituting a vehicle counting mechanism without tracking the movements of the vehicles. In this study, the pre-trained YOLOv3 technique was utilized to reduce the time taken for deploying the DL architecture and for enhancing the system performance. The presented technique took moderate computational hours in object detection and achieved a good performance.

Yin et al. [17] modelled a new approach for counting the vehicles in a human-like manner. The two major contributions of this study are as follows; firstly, the authors modelled a lightweight vehicle counting method called ST-CSNN with high potential. This counting technique compares the vehicles' identity to get rid of the duplicate samples. Integrated with the spatio-temporal data among the frames, it could hasten the speed and enhance the precision of counting process. Secondly, the authors strengthened the model's efficiency by including an enhanced loss function based on the Siamese neural networks. Youssef and Elshenawy [18] examined the utilization of cascade region-oriented R-CNN network for enabling an automatic vehicle tracking and counting method in aerial video streams. The devised approach integrates the cascade R-CNN architecture and the feature pyramid networks to accomplish precise vehicle detection and classification. Navarro et al. [19] explored the possibility of utilizing a low-cost embed mechanism for real-time vehicle recognition and counting with the help of DNNs. The study compared the efficiency of two distinct object tracking techniques such as the centroid tracking algorithm and the Kalman filter with Hungarian method.

Djukanović et al. [20] addressed the acoustic vehicle counting method utilizing one-channel audio.

The authors forecasted the pass-by instance of the automobiles in local minima for the clipped vehicle-to-microphone distances. The outcomes from the experimentation exhibited that the NN-related distance regression method outpaced the previously-devised SVR. In the study conducted earlier [21], a precise technique was modelled for vehicle counting in videos with the help of KLT tracker and Mask R-CNN technique. In this method, the vehicle detection process can be executed for every N frame utilizing mask R-CNN instance segmentation method. This method outpaced the rest of the DL techniques that use bounding box detection, since the former delivers a segmentation mask for all the detected objects. Further, an outstanding performance was achieved in case of occlusion. Once the objects were identified, their corner points were tracked and extracted. A potential technique was presented in this study for assigning the point trajectories to their respective detected vehicles.

3. Proposed model

In the current study, the authors have established a novel SHODL-VCC system for automated vehicle counting and classification in the ITS. The proposed SHODL-VCC technique involves different sub-processes namely, the RetinaNet vehicle detector, the SHO-based parameter tuning and the DWAE-based vehicle classification. These three modules are briefed in the following subsections. Figure 1 represents the workflow of the proposed SHODL-VCC method.

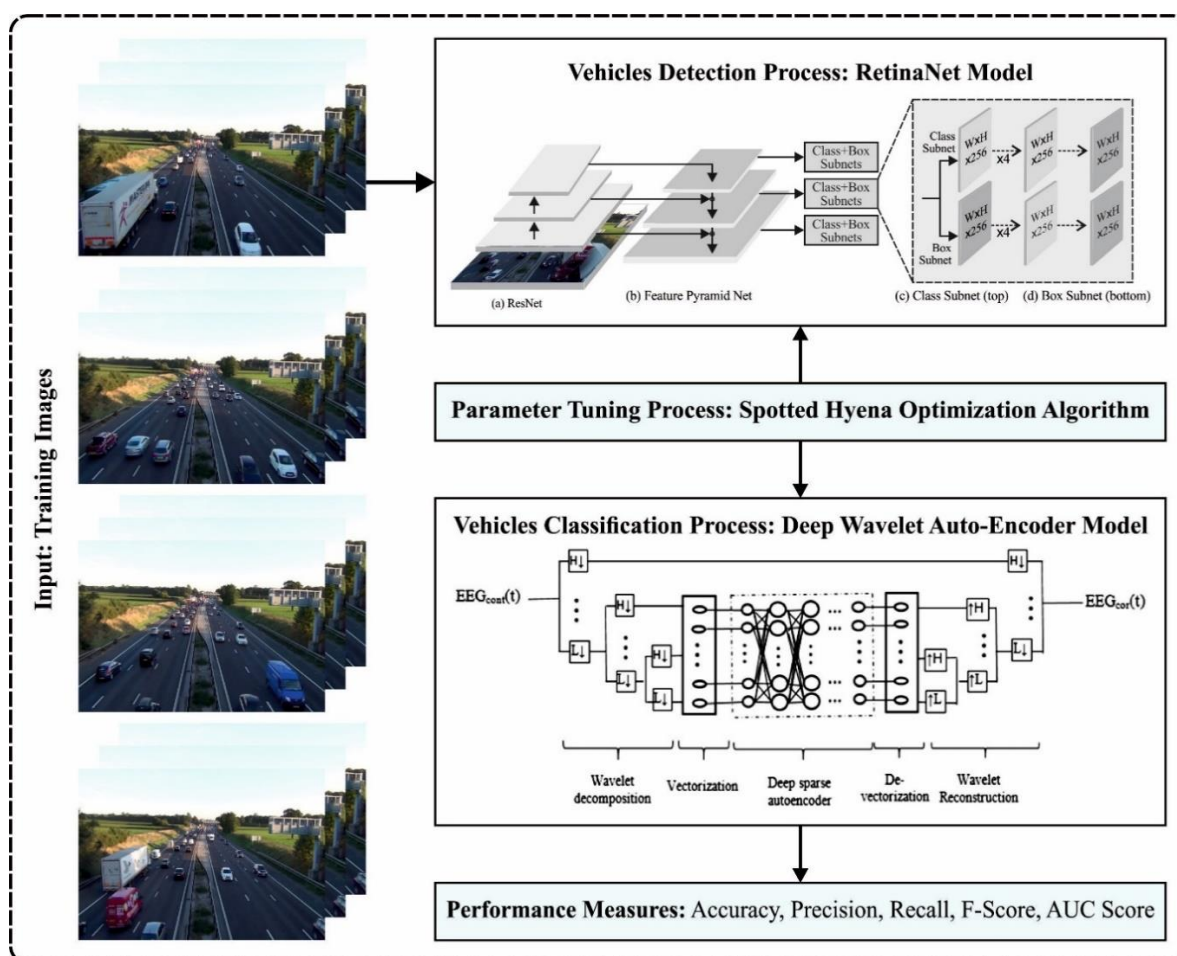


Figure 1. Workflow of the proposed SHODL-VCC system.

3.1. Vehicle detection module: RetinaNet model

Primarily, the presented SHODL-VCC technique employs the RetinaNet object detector to identify the vehicles. RetinaNet is one of the unified networks that is made up of two-task subnetworks with FPN as a backbone network [22]. Figure 2 shows the infrastructure of the RetinaNet method. During the candidate box extraction process, the concept of anchor frame is accepted unlike the RPN network. Further, the region of anchor boxes are amplified from 32×32 to 512×512 . The anchor box of all the layers contains three sizes $\{2^0, 2^{1/3}, 2^{2/3}\}$ and three aspect ratios $\{1:2, 1:1, 2:1\}$. Hence, the feature point of all the feature maps corresponds to nine anchor boxes. A feature pyramid is built on the top of the network whereas the FPN is built using the ResNet model. All the layers of the pyramid contain 256 channels. The two-task subnetworks are categorized into subnetwork and regression subnetwork. Here, the subnetwork forecasts the probability of K classes on all the anchor boxes. Subnetwork is a smaller FCN that is connected with all the layers of FPN and the parameters are shared in it. For the provided pyramid-level outcome with 256 channel characteristic graph, the subnetwork applies four 3×3 convolutional layers. The channel count from all the layers still remains 256, but there exists a ReLU activation layer here. After that, 3×3 convolutional layers are present with a channel number of Ka (K denotes the count of classes and a indicates the count of anchor boxes) and lastly it makes use of the sigmoid activation function. The regression subnetwork corresponds to the classifier subnetwork, and the smaller FCN is connected with all the layers of the FPN for border regression.

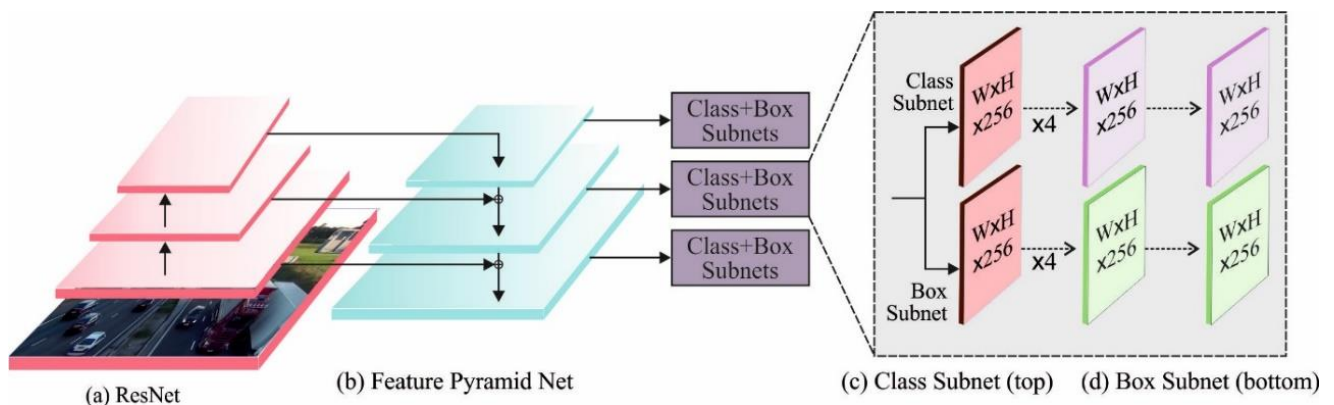


Figure 2. Structure of the RetinaNet model.

The RetinaNet model has a better recognition outcome in case of smaller objects since it employs a novel loss function i.e., focal loss (FL) [23]. The major concept of FL is to minimize the weight so as to categorize the samples and resolve the problems of imbalanced categories. Thus, in spite of large instance size, the contribution of the simply-categorized instances to loss function is smaller. For dual classification task, a typical Cross Entropy (CE) loss is formulated as follows.

$$CE(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1 - p) & \text{otherwise} \end{cases} \quad (1)$$

In Eq (1), $y \in \{\pm 1\}$ signifies the positive and negative instances whereas $p \in [0, 1]$ denotes the probability value of $y = 1$, anticipated by the model. p_t is determined using the following expression.

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{otherwise} \end{cases} \quad (2)$$

Next, the cross entropy loss is formulated as given below.

$$CE(p, y) = CE(p_t) = -\log(p_t) \quad (3)$$

The common solution to overcome the imbalance class is to add the weighted feature $\alpha \in [0, 1]$ i.e., the basis of FL function, as given below.

$$CE(p_t) = -\alpha_t \log(p_t) \quad (4)$$

α balances the negative and positive samples. However, it does not differentiate the easy samples from the default ones. The FL increases the moderating factor $(1 - p_t)^\gamma$ to minimize the weight of the sample and focuses on the training of difficult negative instances.

$$FL(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t) \quad (5)$$

Here, $\gamma \geq 0$ is the changeable focusing parameter that could smoothly fine-tune the proportion for easily-classified instance-weighted reduction and α_t indicates the balancing variable. Once the FL is applied for training and once the detector finds the trained sample to be inaccurate, the p_t becomes lesser and inclines to 0 whereas the adjustable factor approaches 1. Hence, the loss function has no effect; in case of easy-to-classify sample, p_t approaches 1 whereas the adjustable factor approaches 0. Hence, the loss weight of the easily-classified instances reduces considerably.

3.2. Hyperparameter tuning module: SHO algorithm

In order to improve the vehicle detection performance of the RetinaNet algorithm, the spotted hyena optimizer algorithm is used. Spotted Hyenas (SHy) are known as skilled pursuers and they are the largest among the hyaena species [24]. Further, they are well-known as ‘laughing hyenas’ since their vocals sound like human laughter. They are intelligent social creatures and their behaviors are extremely complicated to understand. The SHy traces the prey using its well-developed senses of hearing, smell and sight. Such behaviors of SHy led Dhiman et al. to develop a metaheuristic technique, i.e., spotted hyenas optimization algorithm. In this study, the authors constructed the mathematical approach based on Shy’s behavior and mutual dexterity for optimization. Three actions related to the SHO are catch, encircling and the noticeable catch.

Surrounding of catch: To progress these arithmetical archetypes, it is predictable that the present best challenger is the intended catch based on the fact that the hunt field is formally recognized [25]. In the study, the catch is proposed to the location that is familiar to the hunt mediator so as to attain the benefit and is mathematically processed in the following equations.

$$\vec{D} = |\vec{B} \cdot \vec{P}_p(x) - \vec{P}(x)| \quad (6)$$

$$\vec{P}(x + 1) = \vec{P}_p(x) - \vec{E} \cdot \vec{D}_h \quad (7)$$

Now \vec{D}_h indicates the stretch between SHy and the prey, χ denotes the existing count, \vec{B} and \vec{E}

represent the vector coefficients, \vec{P}_p shows the spot vector of the pursuit and \vec{P} denotes the locale vector of Shy as given below.

$$\vec{B} = 2r\vec{d}_1 \quad (8)$$

$$\vec{E} = 2hr\vec{d}_2 - \vec{h} \quad (9)$$

$$\vec{h} = 5 - \left(t * \left(\frac{5}{t_{\max}} \right) \right) \text{ where } t = 1, 2, 3 \dots, t_{\max} \quad (10)$$

Here, t represents the iteration count and t_{\max} indicates the maximal number of iterations. \vec{r}_{d1} and \vec{r}_{d2} denote the random vectors within $[0, 1]$.

Tricking: In order to arithmetically describe the demeanor of SHy, it is predictable that the best pursuit mediator has the data with regards to the position of the pursuit. The lasting hunt mediator forms an assemblage on the way to the best hunt mediator and stores the best consequence to regenerate the position as given below.

$$\vec{D}_h = |\vec{B} \cdot \vec{P}_h - \vec{P}(x)| \quad (11)$$

$$\vec{P}_k = \vec{P}_h - \vec{E} \cdot \vec{D}_h \quad (12)$$

$$\vec{C}_h = \vec{P}_h + \vec{P}_{k+1} + \dots + \vec{P}_{k+N} \quad (13)$$

Now, \vec{P}_h denotes the locale of the initial best Shy, \vec{P} denotes the locale of the supplementary SHy and N defines the number of SHy in the following equation.

$$N = \text{count}_{\text{nos}} \left(\vec{P}_h \vec{P}_{h+1} \vec{P}_{h+2}, \dots, (\vec{P}_h + \vec{M}) \right) \quad (14)$$

In Eq (14), \vec{M} denotes the arbitrary vector within $[0.5, 1]$, *nos* indicates the architecture of the number of consequences and the total number of competitor results. After adding \vec{M} , the value is compared with the best consequence in a specified quest field. \vec{C} indicates the assembly of N numbers of the best consequence.

Intruding quest (exploitation): In order to develop the prototype based on the equation and to attack the prey, \vec{h} mathematical measure is reduced. The discrepancy in \vec{E} gets similarly condensed from 5 to 0 with computation. $|E| < 1$ pressurizes the assemblages of SHy to outbreak, close to the hunt as given below.

$$\vec{P}(x+1) = \frac{\vec{C}_h}{N} \quad (15)$$

In Eq (15), $\vec{P}(x+1)$ stores the best product and revises the position of other hunt mediators and is reliable with the position of premium pursuit mediators.

Hunt for aim (exploration): SHy habitually pursues the prey based on the position of SHy that subsists in \vec{C}_h . It separately swings to hunt and pursues the prey [26]. Next, \vec{E} is exploited by a random standard > 1 or < -1 to coerce the hunt mediator and swing the distance from the prey. This strategy certifies the SHO method to pursue a wide attainment.

Fitness selection becomes a vital factor in the SHO technique. Solution encoding is used to assess

the goodness of the candidate solution. Nowadays, the accuracy value is the main condition used to design a fitness function.

$$Fitness = \max(P) \quad (16)$$

$$P = \frac{TP}{TP+FP} \quad (17)$$

In this expression, TP represents true positive whereas FP denotes the false positive value.

3.3. Vehicle classification module: DWAE model

Finally, the DWAE model is employed for automated vehicle classification process. A typical AE possesses an unsupervised feature learning ability, strong inference ability and robustness [27]. The property of the Wavelet Transform (WT) has time-frequency, localization and focal features. As a result, it is necessary to combine the WTs and a typical AE to resolve the real-world challenges. The study presents a novel type of unsupervised neural network named 'DWAE' that could catch the non-stationary vibration signals and characterize the complicated data. Equation (18) shows the decoding stage.

$$X = \xi(\kappa'Y + b') \quad (18)$$

Here X represents the result of the reconstructed vector, k indicates the kernel vector, b' signifies the bias value and ϵ indicates the error value more in the process of backpropagation [28]. The DWAE training method is shown herewith.

In the training sample $y = [y_1, y_2, \dots, y_n]^A$, the outcome of the hidden unit is i .

$$g_i(out) = \phi \left(\frac{\sum_{l=1}^n v_{il} y_l - e_i}{b_i} \right) \quad (19)$$

In Eq (19), ϕ signifies the wavelet activation function

$yl (l = 1, 2, \dots, n)$ represents the l th dimensional input of the trained samples

$v_{ij} (i = 1, 2, 3, \dots, g)$ indicates the weight connecting betwixt the input as well as hidden units.

b_i and e_i characterize $v_{ij} (i = 1, 2, 3, \dots, g)$ that directs the scale and shift factors of the wavelet activation function for the i th hidden unit.

$$\phi(a) = \cos(5a) \exp\left(\frac{a^2}{2}\right) \quad (20)$$

$$g_i(out) = \phi_{b,e}(i) = \cos\left(5 \times \frac{\sum_{l=1}^n v_{il} y_l - e_i}{b_i}\right) \times \exp\left(\frac{1}{2} \left(\frac{\sum_{l=1}^n v_{il} y_l - e_i}{b_i}\right)^2\right) \quad (21)$$

Like typical AE, the activation function of the resultant layer is selected as the sigmoid function. Next, the outcome of DWAE is evaluated by [29].

$$y = \text{sigm}\left(\sum_{i=1}^q v_{ri} \left(\cos 5 \times \frac{\sum_{l=1}^n v_{il} y_l - e_i}{b_i} \times \exp\left(\frac{-1}{2} \left(\frac{\sum_{l=1}^n v_{il} y_l - e_i}{b_i}\right)^2\right) \right)\right) \quad (22)$$

In Eq (22), y denotes i the reconstruction dimensional resultant of the trained sample and v_{ri} indicates the weight connecting between the hidden r and i .

4. Performance validation

In this section, the proposed model was validated using the vehicle dataset comprising 900 samples under three classes, as defined in Table 1. Figure 3 represents the sample images.

Table 1. Details of the dataset.

Class	No. of Images
Car	300
Bus	300
Truck	300
Total Number of Samples	900

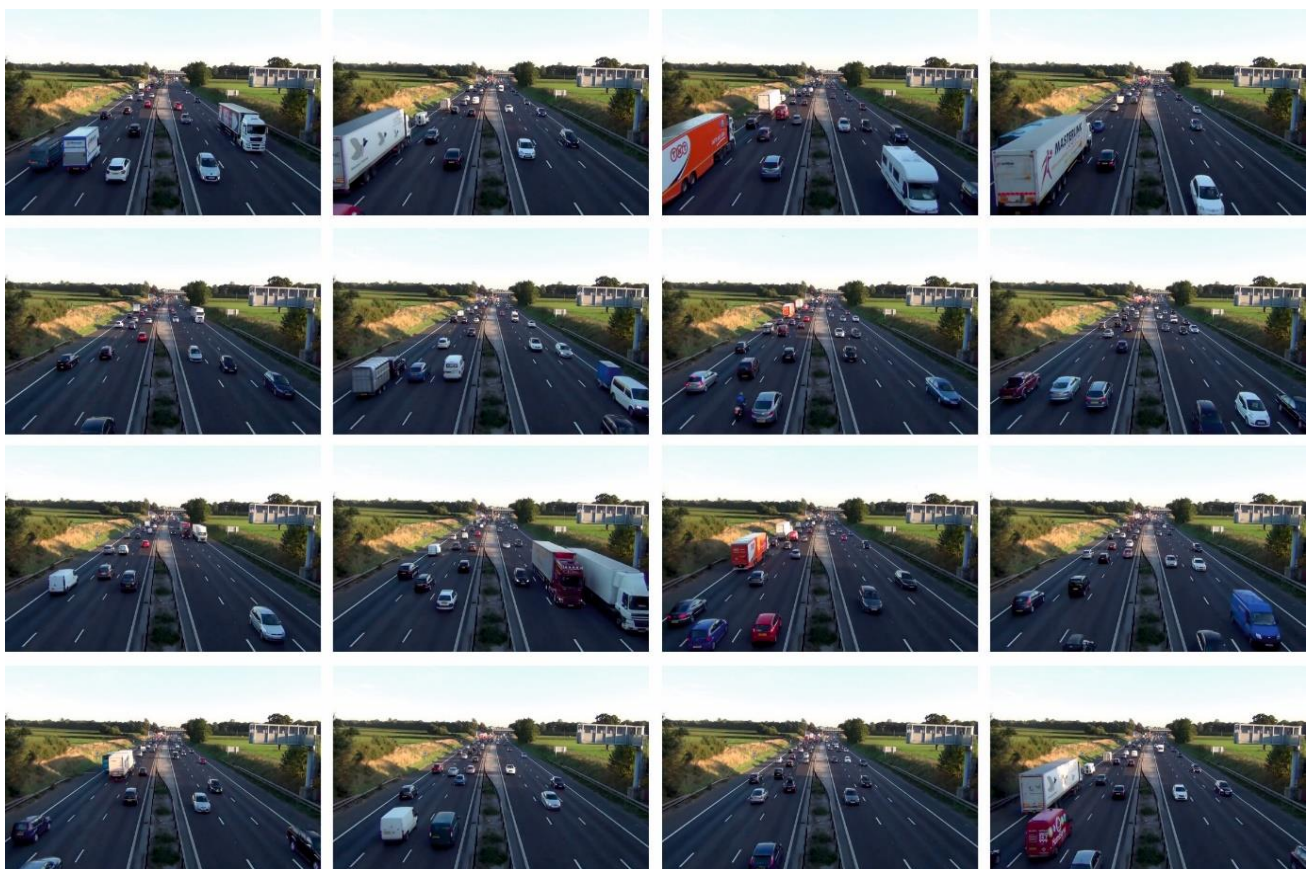


Figure 3. Sample images.

In Figure 4, the confusion matrices generated by the proposed SHODL-VCC technique are clearly shown in the vehicle classification process. The results found that the SHODL-VCC technique accurately recognized three different vehicles namely, car, bus and truck.

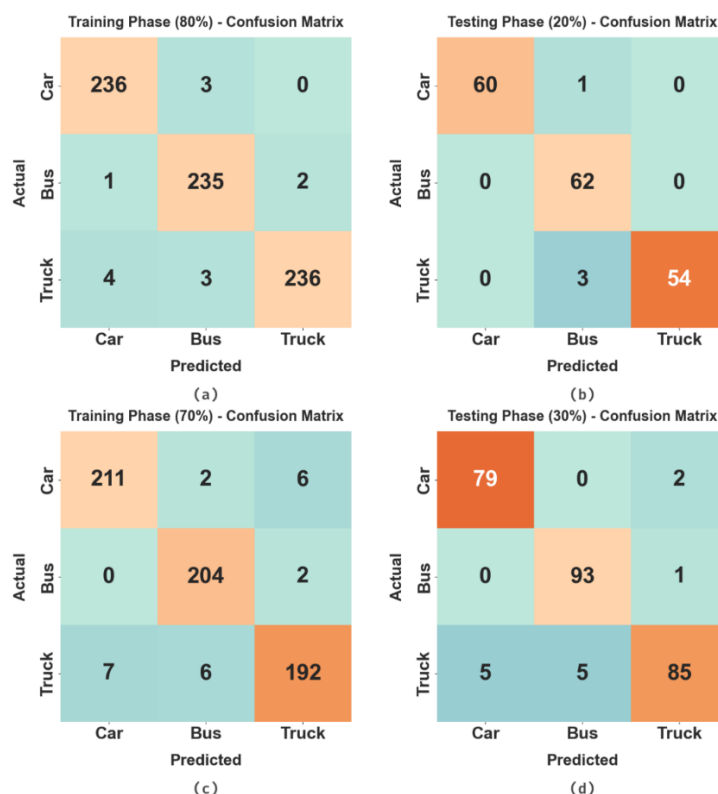


Figure 4. Confusion matrices of the SHODL-VCC system (a-b) TRS/TSS of 80:20 and (c-d) TRS/TSS of 70:30.

In Table 2, the vehicle classification results of the SHODL-VCC technique are provided. Figure 5 shows the vehicle classification outcomes of the SHODL-VCC technique on 80% of TRS. The results found that the SHODL-VCC technique identified all three types of vehicles accurately. Further, the SHODL-VCC technique was found to have attained an average $accu_y$ of 98.80%, $prec_n$ of 98.20%, $reca_l$ of 98.20%, F_{score} of 98.19% and an AUC_{score} of 98.65%.

Figure 6 provides the detailed vehicle classification outcomes of the SHODL-VCC system on 20% of TSS. The outcomes infer that the SHODL-VCC system identified all three types of vehicles in an accurate manner. Further, the SHODL-VCC methodology accomplished an average $accu_y$ of 98.52%, $prec_n$ of 97.98%, $reca_l$ of 97.70%, F_{score} of 97.78% and an AUC_{score} of 98.28%.

Table 2. Vehicle classification outcomes of the SHODL-VCC approach on 80:20 of TRS/TSS.

Labels	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	AUC_{Score}
Training Phase (80%)					
Car	98.89	97.93	98.74	98.33	98.85
Bus	98.75	97.51	98.74	98.12	98.75
Truck	98.75	99.16	97.12	98.13	98.35
Average	98.80	98.20	98.20	98.19	98.65
Testing Phase (20%)					
Car	99.44	100.00	98.36	99.17	99.18
Bus	97.78	93.94	100.00	96.88	98.31
Truck	98.33	100.00	94.74	97.30	97.37
Average	98.52	97.98	97.70	97.78	98.28

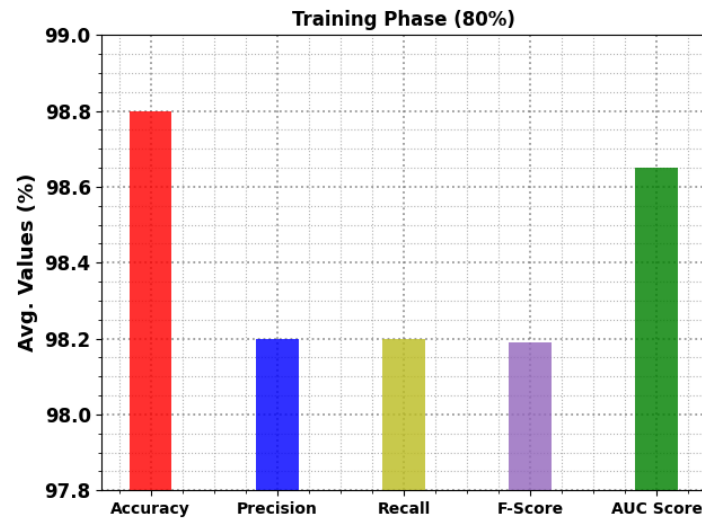


Figure 5. Average outcomes of the SHODL-VCC approach on 80% of TRS.

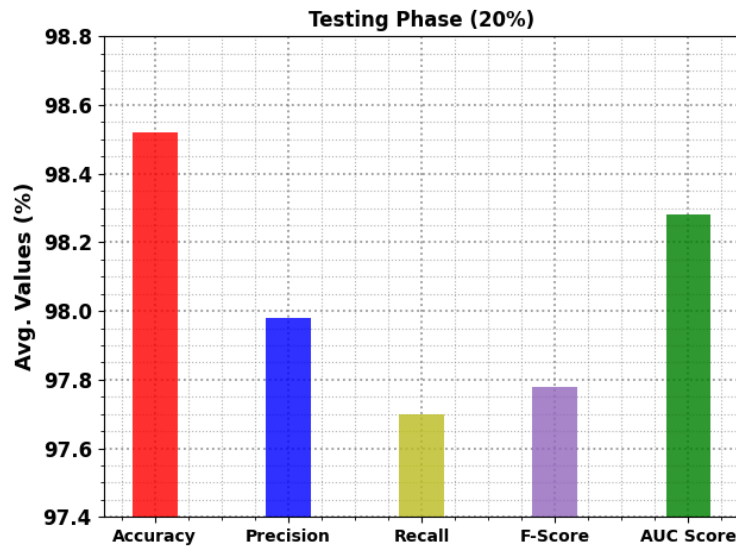


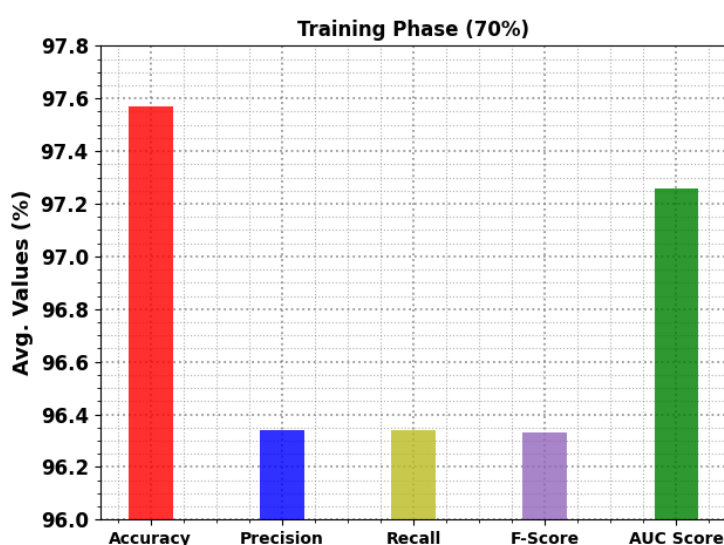
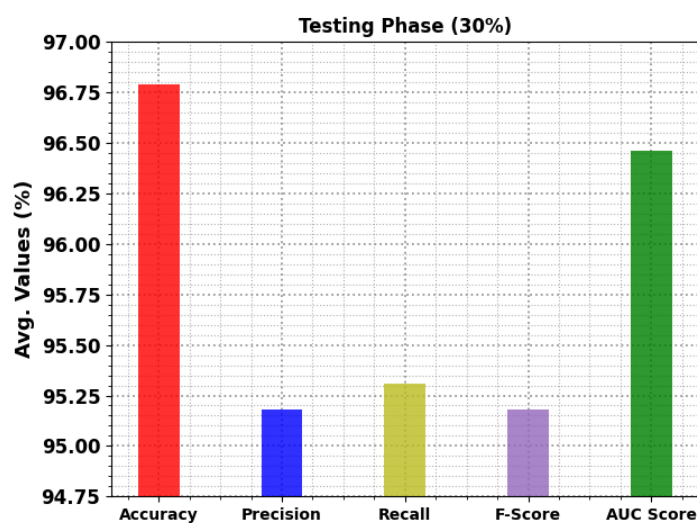
Figure 6. Average outcomes of the SHODL-VCC approach on 20% of TSS.

In Table 3, the vehicle classification outcomes of the SHODL-VCC methodology is portrayed. Figure 7 demonstrates the brief vehicle classification outcomes of the SHODL-VCC approach on 70% of TRS. The results infer that the SHODL-VCC method identified all three types of vehicles accurately. Further, the SHODL-VCC technique accomplished an average $accu_y$ of 97.57%, $prec_n$ of 96.34%, $reca_l$ of 96.34%, F_{score} of 96.33% and an AUC_{score} of 97.26%.

Figure 8 shows the brief vehicle classifier classification outcomes of the proposed SHODL-VCC technique on 30% of TSS. The outcomes imply that the SHODL-VCC system identified all three types of vehicles accurately. Further, the SHODL-VCC methodology accomplished an average $accu_y$ of 96.79%, $prec_n$ of 95.18%, $reca_l$ of 95.31%, F_{score} of 95.18% and an AUC_{score} of 96.46%.

Table 3. Vehicle classification outcomes of the SHODL-VCC approach on 70:30 of TRS/TSS.

Labels	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	AUC_{Score}
Training Phase (70%)					
Car	97.62	96.79	96.35	96.57	97.32
Bus	98.41	96.23	99.03	97.61	98.57
Truck	96.67	96.00	93.66	94.81	95.89
Average	97.57	96.34	96.34	96.33	97.26
Testing Phase (30%)					
Car	97.41	94.05	97.53	95.76	97.44
Bus	97.78	94.90	98.94	96.88	98.05
Truck	95.19	96.59	89.47	92.90	93.88
Average	96.79	95.18	95.31	95.18	96.46

**Figure 7.** Average outcomes of the SHODL-VCC approach on 70% of TRS.**Figure 8.** Average outcomes of the SHODL-VCC approach on 30% of TSS.

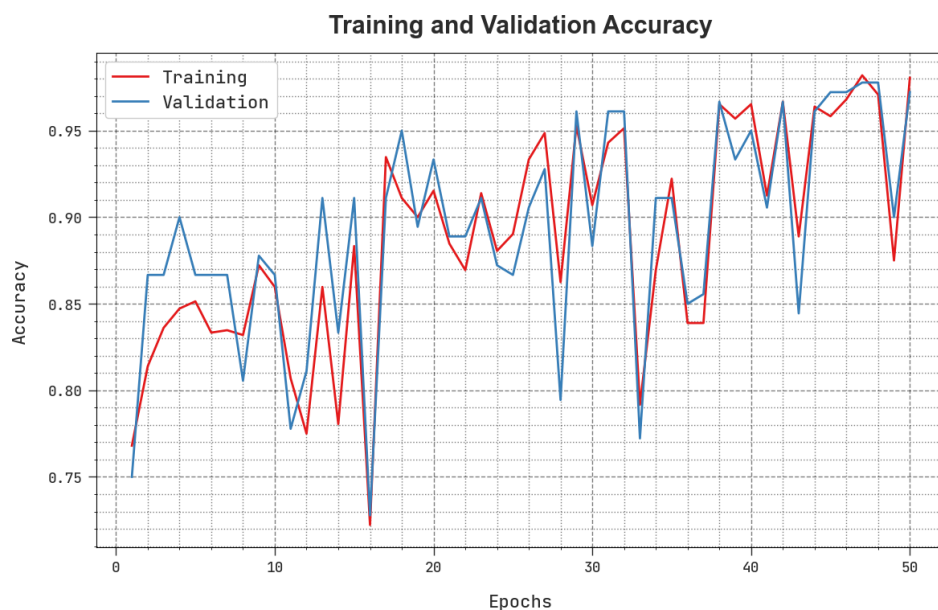


Figure 9. TACC and VACC analyses results of the SHODL-VCC approach.

The TACC and VACC values achieved by the proposed SHODL-VCC algorithm in terms of vehicle classification performance are shown in Figure 9. The figure infers that the SHODL-VCC method revealed a better performance with improved TACC and VACC values. Further, it can be noticed that the SHODL-VCC method reached the maximum TACC outcomes.

The TLS and VLS values, achieved by the SHODL-VCC technique in terms of vehicle classification performance, are portrayed in Figure 10. The figure infers that the SHODL-VCC method achieved an improved performance with minimum TLS and VLS values. Further, it can be understood that the SHODL-VCC approach resulted in minimal VLS outcomes.

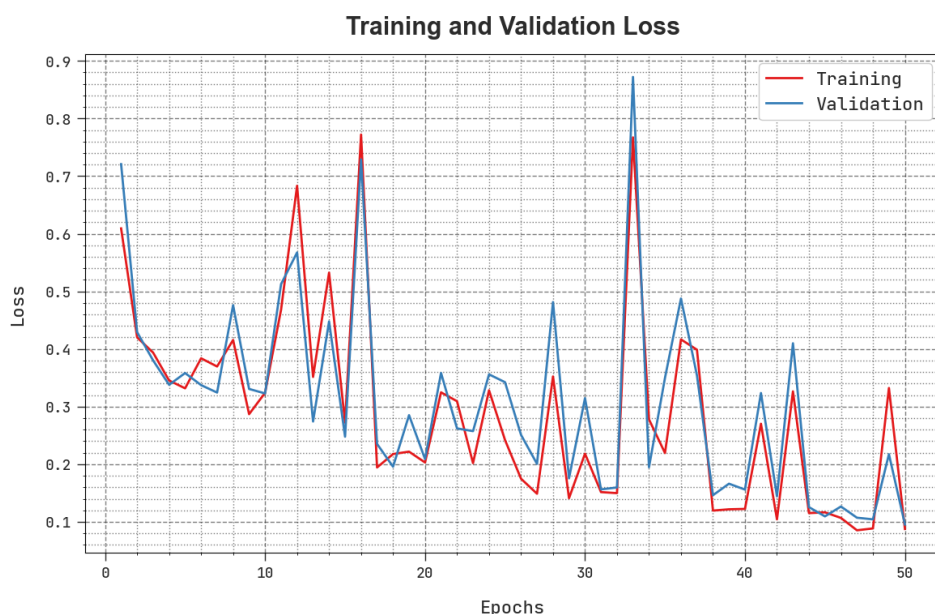


Figure 10. TLS and VLS analyses outcomes of the SHODL-VCC approach.

An evident precision-recall study was conducted upon the SHODL-VCC algorithm using the test database and the results are shown in Figure 11. The figure implies that the SHODL-VCC system led to improved precision-recall values in all three class labels.

A detailed ROC study was conducted upon the SHODL-VCC approach using the test database and the results are illustrated in Figure 12. The outcome infers that the SHODL-VCC algorithm revealed its capability in classifying the three class labels.

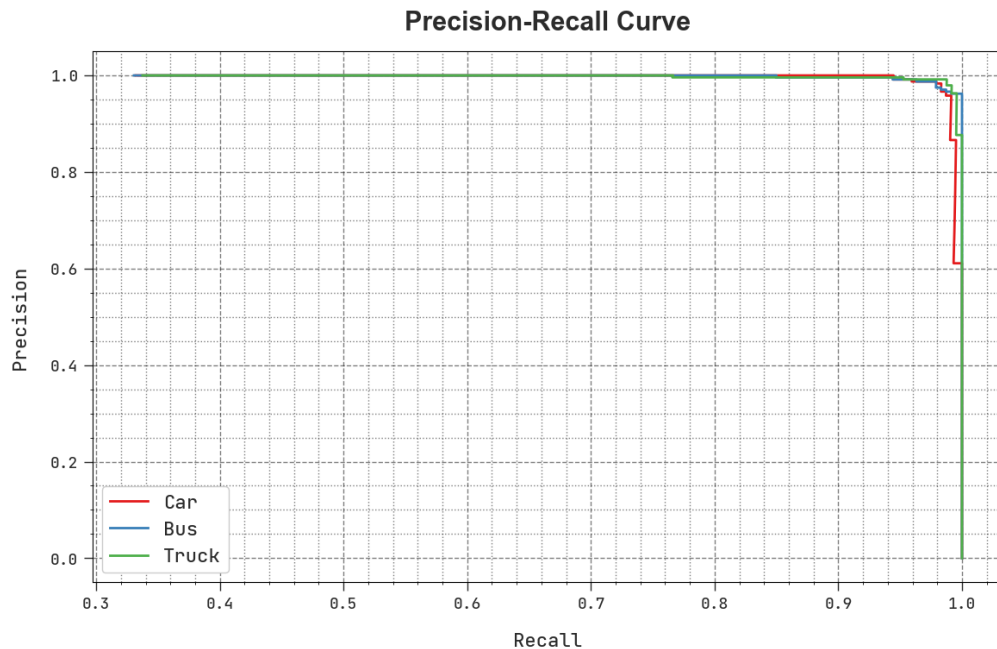


Figure 11. Precision-recall outcomes of the SHODL-VCC algorithm.

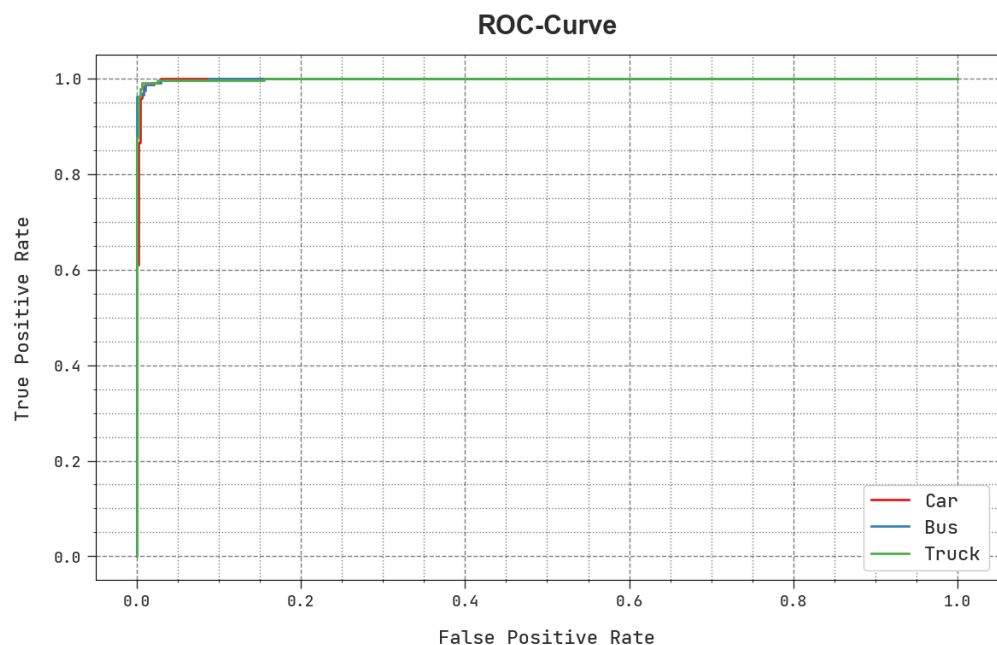
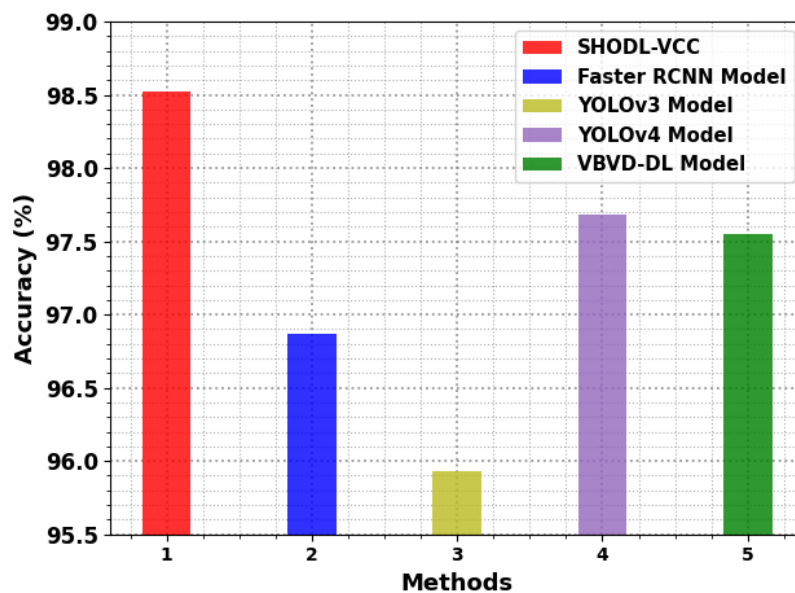


Figure 12. ROC curve outcomes of the SHODL-VCC algorithm.

Table 4. $Accu_y$ analysis outcomes of the SHODL-VCC system with other methods.

Methods	Accuracy (%)
SHODL-VCC	98.52
Faster RCNN Model	96.87
YOLOv3 Model	95.93
YOLOv4 Model	97.68
VBVD-DL Model	97.55

**Figure 13.** $Accu_y$ analysis outcomes of the SHODL-VCC approach with recent algorithms.

Finally, a detailed comparative $accu_y$ examination was conducted upon the SHODL-VCC technique on vehicle classification tasks and the results are reported in Table 4 and Figure 13 [30]. The results indicate that the YOLOv3 model achieved the least $accu_y$ of 95.93% while the Faster RCNN model produced a slightly improved $accu_y$ of 96.87%.

Moreover, the YOLOv4 and VBVG-DL models obtained reasonably closer $accu_y$ values such as 97.68% and 97.55% respectively. However, the SHODL-VCC technique reached a superior outcome with an $accu_y$ of 98.52%. These experimental results demonstrate the supreme performance of the SHODL-VCC algorithm on vehicle classification process.

5. Conclusions

In this study, the authors have established a novel SHODL-VCC system for automated vehicle counting and classification in the ITSs. The SHODL-VCC technique majorly follows a 2-stage procedure such as vehicle detection and vehicle classification. Primarily, the presented SHODL-VCC technique employs the RetinaNet object detector to identify the vehicles. Next, the detected vehicles are classified into different class labels using the DWAE model. To enhance the vehicle detection performance, the spotted hyena optimizer algorithm is exploited as a hyperparameter optimizer, which

considerably improves the vehicle detection rate. The proposed model can be used in several real-time applications such as traffic flow management, toll booth management, parking management, public safety and intelligent navigation. The simulation outcomes of the SHODL-VCC technique were validated on different databases. The comparative results demonstrate the promising vehicle classification performance of the SHODL-VCC technique than the rest of the deep learning approaches with a maximum accuracy of 98.52%. In the future, the vehicle classification performance of the SHODL-VCC technique can be improved with the help of fusion-based ensemble voting process.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through Large Groups Project under grant number (RGP2/95/44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R330), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Research Supporting Project number (RSPD2023R787), King Saud University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

Conflict of interest

The authors declare that they have no conflict of interest. The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

References

1. V. Kocur, M. Ftacnik, Multi-class multi-movement vehicle counting based on CenterTrack, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, 4009–4015.
2. J. Mirthubashini, V. Santhi, Video based vehicle counting using deep learning algorithms, in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, (2020), 142–147.
3. C. J. Lin, S. Y. Jeng, H. W. Lioa, A real-time vehicle counting, speed estimation, and classification system based on virtual detection zone and YOLO, *Math. Probl. Eng.*, **2021** (2021), 1–10. <https://doi.org/10.1155/2021/1577614>
4. A. Glowacz, Thermographic fault diagnosis of shaft of BLDC motor, *Sensors*, **22** (2022), 8537. <https://doi.org/10.3390/s22218537>
5. H. Xu, Z. Cai, R. Li, W. Li, Efficient citycam-to-edge cooperative learning for vehicle counting in ITS, *IEEE Trans. Intell. Transp. Syst.*, **23** (2022), 16600–16611. <https://doi.org/10.1109/TITS.2022.3149657>
6. Y. Y. Tseng, T. C. Hsu, Y. F. Wu, J. J. Chen, Y. C. Tseng, Efficient vehicle counting based on time-spatial images by neural networks, in *2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, IEEE, (2021), 383–391. <https://doi.org/10.1109/MASS52906.2021.00055>

7. M. Haris, A. Glowacz, Lane line detection based on object feature distillation, *Electronics*, **10** (2021), 1102. <https://doi.org/10.3390/electronics10091102>
8. Z. Xie, R. Rajamani, Vehicle counting and maneuver classification with support vector machines using low-density flash lidar, *IEEE Trans. Veh. Technol.*, **71** (2021), 86–97. <https://doi.org/10.1109/TVT.2021.3125919>
9. A. Glowacz, Ventilation diagnosis of angle grinder using thermal imaging, *Sensors*, **21** (2021), 2853. <https://doi.org/10.3390/s21082853>
10. C. Liu, D. Q. Huynh, Y. Sun, M. Reynolds, S. Atkinson, A vision-based pipeline for vehicle counting, speed estimation, and classification, *IEEE Trans. Intell. Transp. Syst.*, **22** (2020), 7547–7560. <https://doi.org/10.1109/TITS.2020.3004066>
11. A. M. Santos, C. J. Bastos-Filho, A. Maciel, Counting vehicle by axes with high-precision in brazilian roads with deep learning methods, in *International Conference on Intelligent Systems Design and Applications*, Springer, Cham, **418** (2021), 188–198. https://doi.org/10.1007/978-3-030-96308-8_17
12. A. Glowacz, Thermographic fault diagnosis of ventilation in BLDC motors, *Sensors*, **21** (2021), 7245. <https://doi.org/10.3390/s21217245>
13. O. E. A. Agudelo, C. E. M. Marín, R. G. Crespo, Sound measurement and automatic vehicle classification and counting applied to road traffic noise characterization, *Soft Comput.*, **25** (2021), 12075–12087. <https://doi.org/10.1007/s00500-021-05766-6>
14. A. Alsanabani, A. Ahmed, A. M. Al Smadi, Vehicle counting using detecting-tracking combinations: A comparative analysis, in *2020 The 4th International Conference on Video and Image Processing*, (2020), 48–54. <https://doi.org/10.1145/3447450.3447458>
15. H. Lin, Z. Yuan, B. He, X. Kuai, X. Li, R. Guo, A deep learning framework for video-based vehicle counting, *Front. Phys.*, **10** (2022), 32. <https://doi.org/10.3389/fphy.2022.829734>
16. M. Fachrie, A simple vehicle counting system using deep learning with YOLOv3 model, *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, **4** (2020), 462–468. <https://doi.org/10.29207/resti.v4i3.1871>
17. K. Yin, L. Wang, J. Zhang, ST-CSNN: a novel method for vehicle counting, *Mach. Vision Appl.*, **32** (2021), 1–13. <https://doi.org/10.1007/s00138-021-01233-2>
18. Y. Youssef, M. Elshenawy, Automatic vehicle counting and tracking in aerial video feeds using cascade region-based convolutional neural networks and feature pyramid networks, *Trans. Res. Rec.*, **2675** (2021), 304–317. <https://doi.org/10.1177/0361198121997833>
19. J. Navarro, D. S. Benítez, N. Pérez, D. Riofrío, R. F. Moyano, Towards a low-cost embedded vehicle counting system based on deep-learning for traffic management applications, in *2021 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 2021, 1–6. <https://doi.org/10.1109/CHILECON54041.2021.9702914>
20. S. Djukanović, Y. Patel, J. Matas, T. Virtanen, Neural network-based acoustic vehicle counting, in *2021 29th European Signal Processing Conference (EUSIPCO)*, 2021, 561–565. <https://doi.org/10.23919/EUSIPCO54536.2021.9615925>
21. Z. Al-Ariny, M. A. Abdelwahab, M. Fakhry, E. S. Hasaneen, An efficient vehicle counting method using mask r-cnn, in *2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*, 2020, 232–237. <https://doi.org/10.1109/ITCE48509.2020.9047800>

22. J. Liu, R. Jia, W. Li, F. Ma, H. M. Abdullah, H. Ma, et al., High precision detection algorithm based on improved RetinaNet for defect recognition of transmission lines, *Energy Rep.*, **6** (2020), 2430–2440. <https://doi.org/10.1016/j.egyr.2020.09.002>
23. M. Ahmad, M. Abdullah, D. Han, Small object detection in aerial imagery using RetinaNet with anchor optimization, in *2020 International Conference on Electronics, Information, and Communication (ICEIC)*, 2020, 1–3. <https://doi.org/10.1109/ICEIC49074.2020.9051269>
24. G. Dhiman, V. Kumar, Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications, *Adv. Eng. Softw.*, **114** (2017), 48–70. <https://doi.org/10.1016/j.advengsoft.2017.05.014>
25. A. Saha, P. Dash, N. R. Babu, T. Chiranjeevi, M. Dhananjaya, L. Knypinski, Dynamic stability evaluation of an integrated biodiesel-geothermal power plant-based power system with spotted hyena optimized cascade controller, *Sustainability*, **14** (2022), 14842. <https://doi.org/10.3390/su142214842>
26. M. Gafar, R. A. El-Sehiemy, H. M. Hasanien, A. Abaza, Optimal parameter estimation of three solar cell models using modified spotted hyena optimization, *J. Ambient Intell. Human. Comput.*, **2022** (2022), 1–12. <https://doi.org/10.1007/s12652-022-03896-9>
27. H. D. Shao, H. K. Jiang, X. Q. Li, S. P. Wu, Intelligent fault diagnosis of rolling bearing using deep wavelet auto-encoder with extreme learning machine, *Knowl. Based Syst.*, **140** (2018), 1–14. <https://doi.org/10.1016/j.knosys.2017.10.024>
28. I. Abd El Kader, G. Xu, Z. Shuai, S. Saminu, I. Javaid, I. S. Ahmad, et al., Brain tumor detection and classification on MR images by a deep wavelet auto-encoder model, *Diagnostics*, **11** (2021), 1589. <https://doi.org/10.3390/diagnostics11091589>
29. H. D. Shao, H. K. Jiang, K. Zhao, D. D. Wei, Dongdong, X. Q. Li, A novel tracking deep wavelet auto-encoder method for intelligent fault diagnosis of electric locomotive bearings, *Mech. Syst. Signal Process.*, **110** (2018), 193–209. <https://doi.org/10.1016/j.ymssp.2018.03.011>
30. H. Song, H. Liang, H. Li, Z. Dai, X. Yun, Vision-based vehicle detection and counting system using deep learning in highway scenes, *Eur. Transp. Res. Rev.*, **11** (2019), 1–16. <https://doi.org/10.1186/s12544-019-0390-4>



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)