Integration of Ensemble Variant CNN with Architecture Modified LSTM for Distracted Driver Detection

Zakaria Boucetta[1], Abdelaziz El Fazziki[2], Mohamed El Adnani[3]
Computer Science Engineering Laboratory
Faculty of Sciences, Cadi Ayyad University
Marrakech, Morocco

***Abstract*—Driver decisions and behaviors are the major factors in on-road driving safety. Most significantly, traffic injuries and accidents are reduced using the accurate driver behavior monitoring system. However, the challenges occur in understanding human behaviors in the practical environment due to uncontrolled scenarios like cluttered and dynamic backgrounds, occlusion, and illumination variation. Recently, traffic accidents are mainly caused by distracted drivers, which has increased with the popularization of smartphones. Therefore, the distracted driver detection model is necessary to appropriately find the behavior of the distracted driver and give warnings to the driver to prevent accidents, which need to be concentrated as serious issues. The main intention of this paper is to design and implement a novel deep learning framework for driver distraction detection. First, the datasets for driver distraction detection are gathered from public sources. Furthermore, the Optimal Fusion-based Local Gradient Pattern (LGP) and Local Weber Pattern (LWP) perform the pattern extraction of the images. These patterns are inputted into the new deep learning framework with Ensemble Variant Convolutional Neural Network (EV-CNN) for feature learning. The EV-CNN includes three different models, like Resnet50, Inceptionv3, and Xception. The extracted features are subjected to the architecture-optimized Long Short-Term Memory (LSTM). The Hybrid Squirrel Whale Optimization Algorithm (HSWOA) performs both the pattern extraction and the LSTM optimization. The experimental results demonstrate the effective classification performance of the suggested model in terms of accuracy during the detection of distracted driving and are helpful in maintaining safe driving habits.**

Keywords—Distracted driver detection; ensemble variant convolutional neural network; hybrid squirrel whale optimization algorithm; local gradient pattern; local weber pattern; optimal fusion-based pattern descriptors; long short term memory

I. Introduction

The complicated traffic road systems have four different elements like environment, roads, cars, and people. If these four elements are not coordinated, then it leads to road accidents [1]. Recent research reveals that more number of accidents nearly 90% of accidents are caused due to driver error owing to their drunkenness, distraction, and fatigue. Mostly, car accidents highly occur because of the reason of driver distraction. Some of the actions of the drivers like eating or drinking, switching to a favorite radio station and using mobile devices lead to accidents as they only need to focus on the driving towards road without any distraction [2] [3]. Distracted driving mainly causes the wrong lane changes those results in severe traffic accidents. National Highway Traffic Safety Administration (NHTSA) defines distracted driving as "any activity that diverts attention from driving, including talking or texting on the phone, eating and drinking, etc.". The rapid development of onboard electronics like smartphones and navigation systems are designed as supplementary factors that influence the distraction of the driver [4]. Hence, it is essential for developing in-depth research on determining their occurrence mechanisms, distracted driver attitude, and developing their respective solutions. This can be done for enhancing driving safety and reducing the frequency of distracted driving behaviors [5] [6], [7].

Many research works have been made for identifying the behavior of distracted drivers at different levels. The Gaussian Mixture Model (GMM) model is developed by collecting the parameters of vehicle motion through in-car GPS for judging whether the distraction occurred or not in the driver [7]. The collected vehicle data in the running state is utilized in the existing model based on support vector machine technology and has differentiated them into three classes like high cognitive load distraction, low cognitive load distraction and no distraction [8]. The dynamic parameters regarding the brake baffle position and acceleration baffle position are gathered and involved five diverse machine learning approaches for distinguishing the visual distraction of the driver, lane lateral deviation and longitudinal/lateral acceleration, where the vehicle speed is used in the model. This model achieves an accurate identification of the distraction state of the driver. Recently, the detection approaches of distracted driving relies on the dynamic estimation of vehicle operating parameters like lane deviation, steering wheel angle, acceleration, and vehicle speed for determining the distraction [9] [10]. On the other hand, this methodology requires the installation of additional exclusive data acquisition tools in the vehicle [11]. Concerning the different kinds and specifications, the accuracy and completeness of data acquisition devices are varying which is not beneficial for the application in the practical driving environment.

Various vision and physiological sensors are vastly utilized in monitoring the driver's status. Physiological sensors are commonly limited to estimating particular driving behavior. In order to evaluate the fatigue and somnolence of the driver, EEG and EOG [12] are mostly used in the existing model. However, these sensors are cost-ineffective and need certain prior information like gaze direction. Some approaches are developed for detecting distracted driver behaviors using vision, vehicle driving status, and physiological parameters. According to the development in computing hardware, neural networks, and camera technology, complete features are allowed to be obtained from complex images. Convolutional Neural Networks (CNNs) [13] are commonly used for certain complex tasks in image processing. Recently, different approaches have been developed in the deep learning concept for solving the problems related to image recognition and classification [14] [15] [16]. These approaches are used for extracting the features from the images and utilized for performing the classification. Yet, the distracted behavior is detected by involving a single pre-trained model, which leads to an overfitting problem that further results in detection failures in practical applications. Therefore, a new distracted driver detection model needs to develop using a deep learning approach.

The main contributions of the research work are described as follows:

- To design a new distracted driver detection model with the support of the suggested hybrid optimization algorithm using the developed ensemble learning for feature extraction through the improved deep learning model for distracted driver detection to reduce traffic accidents by warning the distracted drivers.
- To develop an optimal fusion-based pattern extraction approach by combining the extracted patterns of LGP and LWP for maximizing the entropy of the developed HSWOA-based pattern images.
- To integrate an ensemble variant-CNN for extracting more significant features from the ResNet50, Inceptionv3, and Xception networks for ensuring the highly essential features for the proposed model.
- To develop an improved deep learning model named O-LSTM for detecting distracted driver behavior with the help of suggested HSWOA by optimizing certain parameters in LSTM for enhancing the accuracy and precision of the detection in the proposed distracted driver detection model.
- To introduce the hybrid optimization algorithm named HSWOA for optimizing the fusion weights of the optimal fused patterns and for tuning the hidden neurons and number of epochs of the LSTM to achieve accurate detection performance and to improve the overall performance of the proposed model.
- To evaluate the efficiency of the proposed model by comparing it with diverse heuristic algorithms and also with the different classifiers under various quantitative measures.

The rest of the sections in the proposed model are stated as follows. The Section II discusses the exciting works and their challenges. The Section III explains the proposed distracted driver detection model using deep learning approaches. The Section IV conveys the optimal fusion-based pattern extraction of the proposed model. The Section V narrates the ensemble variant-CNN-based feature extraction phase for the proposed model. The Section VI describes the detection phase and suggested HSWOA of the proposed model. The Section VII depicts the achieved results of the proposed distracted driver detection model. The Section VII concludes the developed distracted driver detection model.

II. Literature Survey

A. Related Work

In 2020, Omerustaoglu et al. [17] have developed the distracted driver detection approach based on vision data for improving the generalization ability of the proposed model. The dataset was obtained as the sensor data and driver images that were acquired from the practical drives. The distraction of the driver was identified through the two-stage model with the help of deep learning approaches. This proposed model was validated and shown that proposed model was enhanced the overall performance of both the fused approaches based on the public dataset. In 2021, Huang et al. [18] have implemented a hybrid form of deep learning framework for finding the distracted behaviors of the drivers by processing the image features. The co-operative pre-trained model was developed for enhancing the accuracy while detecting the driving activity in the system with the help of combined deep learning architectures. The simulation analysis has shown that the proposed model has secured better classification accuracy in the detection of distracted driver behavior and also confirmed the potential capability of maintaining the safety driving without any accidents.

In 2021, Jegham et al. [19] have proposed a Kinect sensor for recognizing the safe driver with the multimodal signals from the sensors. Then, the attention-based network was developed for performing the cognitive process towards the deep network by concentrating on the motion and silhouette of the driver that was acquired from the correlated driver scenes. The analysis results have revealed that the proposed model was highly effective in terms of providing high classification accuracy when compared with the conventional methods. In 2020, Ou and Karray [20] have suggested a novel driver distraction detection model using an improved deep learning approach for determining distracted behavior in different driving conditions. Initially, the generative models were designed for generating the images related to diverse driving conditions and then, the discriminative models for classifying the generated images. The collected dataset was correlated with newly generated training samples, and then, the deep learning approach was trained for recognizing the distraction. The suggested framework has proved that the generative models have a high ability to produce driver images under diverse driving conditions.

In 2021, Pal et al. [21] have investigated a deep learning model for classifying and identifying the distracted behavior of drivers to alert them at the right time and also has ensured an effective solution for the problem. The integrated model has divided the driver activity into ten different categories that have also included with the safe driving class. When the distracted driver was observed, the proposed model has detected the event and eliminated the accidents by alerting the drivers. In 2021, Rao et al. [22] have integrated a detection model for identifying the distraction behavior of the driver using the driving images that were captured using an in-vehicle camera. Here, the whitening of the image has occurred for minimizing the correlation and redundancy of the pixel matrix. The analysis results have shown that a higher accuracy was acquired through the proposed model in detecting the distraction behavior when compared with the existing machine learning models.

In 2021, Kumar et al. [23] have proposed a deep learning-based detection model for finding driver distraction behavior. This introduced model has provided superior performance based on the practical environmental inputs, which was determined by comparing it with the other state-of-the-methods. The suggested approaches have secured high scores in all the quantitative measures for classification. In 2021, Dunn et al. [24] have implemented the data from the two different driving studies by incorporating the automatic-equipped vehicles with the help of a driving automation system in order to analyze the driver behaviors. The analysis results have ensured with significant insights of the driving automation systems under diverse operational phases. Here, the overreliance and overtrust on the advanced technologies have reduced certain safety benefits of these proposed systems.

B. Problem Statement

How to correctly identify whether the driver is in a distracted driving state and provide the necessary warnings for the driver to avoid potential safety risks has become one of the most concerning issues. Many techniques have been developed earlier, and some of them are given pros and cons as mentioned in Table I. CNN and LSTM-RNN [17] reduce the overfitting problem and improve the overall accuracy rate. However, it shows the performance loss owing to the viewpoint of the car and camera are not the same as in the public datasets as their collection setup. HCF [18] achieves a good identification rate and also achieves better classification performance. Yet, when the number of training samples is increased, the performance rate can be affected. DSA [19] enhances the classification capability by choosing the highly essential regions from the images. However, it is slightly suffered from high illumination variation particularly focusing at the front view. CNN [20] efficiently reduces the computational time using the feature selection process. Still, it does not be applicable for real-time applications as it needs to be enhancing the effectiveness of the controller. CNN [21] efficiently detects the distracted regions. Yet, it is difficult to train the data and the computation of the network is slow. Deep CNN [22] attains a better accuracy rate at a faster speed. However, more memory is required to train the data. CapsNet [23] attains better accuracy, precision and recall values even with the utilization of less model parameters. On the other hand, it does not consider the detection of drowsiness, which is a significant non-visual feature. VCC NDS and DAF NDS [24] show effective performance on studying the Over-trust and overreliance on driving automation. But, it includes a very less number of samples for investigation. Hence, owing to the consideration of these existing challenges, it is well-determined to develop a new driver distraction model for avoiding traffic accidents caused by drivers.

TABLE I. FEATURES AND CHALLENGES OF EXISTING DISTRACTED DRIVER DETECTION MODELS

Author [citation]	**Methodology**	**Features**	**Challenges**
Omerustaoglu *et al.* [17]	CNN and LSTM-RNN	It reduces the overfitting problem and improves the overall accuracy rate.	It shows the performance loss owing to the viewpoint of the car and camera are not the same as in the public datasets as their collection setup.
Huang *et al.* [18]	HCF	It achieves a good identification rate. It achieves better classification performance.	When the number of training samples is increased, the performance rate can be affected.
Jegham *et al.* [19]	DSA	It enhances the classification capability by choosing the highly essential regions from the images.	It is slightly suffered from high illumination variation particularly focussing at the front view.
Ou and Karray [20]	CNN	It efficiently reduces the computational time using the feature selection process.	It does not be applicable for real-time applications as it needs to be enhancing the effectiveness of the controller.
Pal *et al.* [21]	CNN	It efficiently detects the distracted regions.	It is difficult to train the data. The computation of the network is slow.
Rao *et al.* [22]	Deep CNN	It attains a better accuracy rate at a faster speed.	More memory is required to train the data.
Kumar *et al.* [23]	CapsNet	It attains better accuracy, precision and recall values even with the utilization of less model parameters.	It does not consider the detection of drowsiness, which is a significant non-visual feature.
Dunn *et al.* [24]	VCC NDS and DAF NDS	It shows effective performance on studying the Over-trust and overreliance on driving automation.	It includes a very less number of samples for investigation.

III. Architectural View of Distracted Driver Detection with Data Collection

A. Architectural Description

Before, the classification of a distracted driver is performed in two different ways. In the first way of method, wearable sensors are used for computing physiological and biomedical signals like heart rate, muscular and vascular activities and brain activity. However, the above-mentioned method has certain disadvantages like user involvement and hardware cost. In the second way of method, the cameras are used for classifying distracted drivers, where diverse vision-based approaches are involved for monitoring the distracted behaviors in practice such as fatigue-related features from the driver's face, body postures, gaze and head pose detection. Mostly, the vision-based techniques are incorporated as the two-step structure, where the significant features are obtained from the original data based on the hand-crafted techniques, and next, the classifiers are considered according to the hand-crafted features. If the two-step structure is used in the approach, then the approach was not able to acquire an optimal trade-off among the hand-crafted features and robustness of the trained classifier. The vision-based approaches are involved with the decision trees and Support Vector Machines (SVMs) for detecting distracted drivers, which seeks most of the research. CNNs are generally utilized for complex tasks related to image processing. Deep learning models have been developed rapidly into speech recognition, natural language processing, and computer vision and ensured promising outcomes for solving the distracted driver detection problem. Recently, various deep learning approaches like DenseNet, ResNet, Inception, VGGNet, and AlexNet have been developed for handling the problems related to image recognition and classification. These techniques are used for extracting the significant features from the images and are considered for image classification. But, the recognition of distracted driver behavior based on the single pre-trained model may lead to overfitting issues that further result in detection failure. Therefore, it is required to develop a new distracted driver detection model using ensemble learning approaches that are diagrammatically represented in Fig. 1.

A new distracted driver detection model is developed and processed under various steps for detecting distracted driver behaviors while driving to ensure safe driving and to avoid traffic accidents with the help of deep learning and a hybrid optimization algorithm. The input driver images with both safe driving and distracted behavior images are obtained from publicly available resources. The collected input images were considered for the optimal fusion-based pattern extraction phase. This phase concatenates the pattern extracted from two different pattern extraction approaches like LGP and LWP, where the optimized weights for respective patterns are incorporated for maximizing the entropy of the extracted patterns. The fusion weights are optimized with the help of a suggested hybrid algorithm named HSWOA, which is implemented by combining the features of SSA and WOA. Then, the optimized patterns are given into the developed EV-CNN, in which the deep features are extracted from the pooling layers of the ensemble variants of CNN like ResNet 50, inceptionv3 and xception for obtaining the most accurate features of the proposed model to the detection phase. Further, the extracted features are subjected into the developed O-LSTM, where the suggested HSWOA is used for optimizing the number of hidden neurons and number of epochs of LSTM to improve the accuracy and precision of the proposed model. At last, the distracted driver behaviors are detected using the implemented O-LSTM.

B. Data Collection

The proposed distracted driver detection model collects the required images from the "https://www.kaggle.com/c/state-farm-distracted-driver-detection/data: access date: 2021-12-27". Here, the dataset consists of the driver images of doing some actions such as reaching behind, doing makeup, talking on the phone, eating and texting. The dataset is comprised of ten classes to predict the driver distraction behavior with both the test and train data. The input images of the proposed distracted driver detection model are represented as IM_p^{indriv}, where $p = 1, 2, \ldots, P$ and P show the total number of collected driver images from the dataset. The sample images for ten classes are listed in Fig. 2.

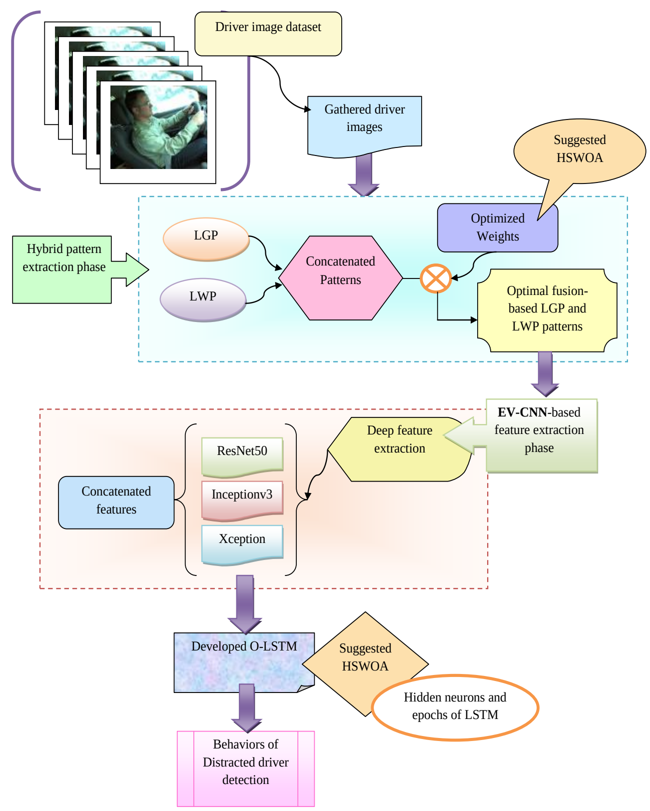


Fig. 1. The Proposed Architecture of a New Distracted Driver Detection Model.

Class description	Image 1	Image 2	Image 3
Class 1: Safe driving			
Class 2: Texting-right			
Class 3: Talking on the phone-right			
Class 4: Texting-left			
Class 5: Talking on the phone-left			
Class 6: Operating the radio			
Class 7: Drinking			
Class 8: Reaching behind			

Class 9: Hair and makeup			
Class 10: Talking to passenger			

Fig. 2. Sample Distracted Driver Images considered for the Proposed Distracted Driver Detection Model

IV. Primary Steps of Distracted Driver Detection: Optimal Fusion Approach

A. Optimal Fusion-based Pattern Extraction

The proposed distracted driver detection model utilizes the LGP and LWD for extracting the patterns from the input images IM_p^{indriv} to attain the accurate performance while detecting distracted drivers while driving. LGP [25] is used for providing the transformation of output image, which is independent of the global intensity variations. This approach is employed for ensuring the facial features but, it is highly sensitive to the local variations along with the edge components in the human face. Similarly, LWP [26] is utilized in the proposed model as it provides superior performance over other descriptors. However, this model considers only four pixels on both the vertical and horizontal directions that are insufficient to represent the local information. Therefore, the pattern extracted from the LGP and LWP is integrated with the optimized weights, respectively using the suggested HSWOA for enhancing the detection performance and to overcome the existing challenges. Here, the optimized fusion weights with the LGP-based pattern is denoted by α, and the optimize weights with the LWP-based pattern is indicated by β. The optimal fusion-based patterns are obtained through in Eq. (1).

$$Pt_z^{of} = \left(Pt_x^{LGP} * \alpha\right) + \left(Pt_x^{lwd} * \beta\right) \tag{1}$$

The main objective FF_1 of optimizing the weights of the extracted patterns is to maximize the entropy of the proposed model that is given in Eq. (2).

$$FF_1 = \underset{\{\alpha,\beta\}}{\arg\max}\left(\frac{1}{EN}\right) \tag{2}$$

Here, the term EN denotes the entropy, which is computed through the Eq. (13). Entropy EN gives "the amount of loss of data in a given input which provides the required information for feature extraction". This is shown in the Eq. (3).

$$EN = \sum_{x=0}^{X-1} Pt_x \log Pt_x \tag{3}$$

The total number of patterns is expressed as X. The entropy needs to be maximized for acquiring the efficient optimized pattern for the distracted driver detection model. Finally, the Optimal fusion patterns are denoted by Pt_z^{of}. The optimal fusion-based pattern extraction with LGP and LWP is represented in Fig. 3.

B. LGP

The input images IM_p^{indriv} are subjected to the LGP technique for extracting the local gradient pattern in the neighboring by involving the intensity gradient profile of the neighborhood measure. Therefore, the LGP value of IM_p^{indriv} is computed for each pixel (a_d, b_d) using Eq. (4).

$$LGP(a_d, b_d) = \sum_{i=0}^{I} t\left(h_I - \bar{h}\right) 2^i,$$
$$t(a) = \begin{cases} 1, & a \geq 0 \\ 0, & otherwise \end{cases} \tag{4}$$

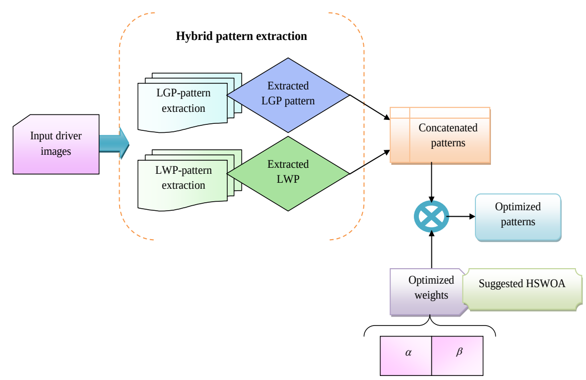


Fig. 3. Optimal Fusion-based Pattern Extraction for the Proposed Distracted Driver Detection Model.

Here, the term h_I represents the gradient of the neighboring pixel I and the average gradient of the neighboring pixels are denoted as $\bar{h}$ that are calculated as given in Eq. (5) and Eq. (6).

$$h_I = |T_I - T_C| \tag{5}$$

$$\bar{h} = \left(\frac{1}{8}\right)\sum_{i=0}^{I} h_I \tag{6}$$

The extracted patterns are obtained from the LGP technique that is expressed as Pt_x^{LGP}.

C. LWP

The proposed distracted driver detection model employs the input images IM_p^{indriv} for extracting the patterns to determine the distracted driver. LWP is the pattern extraction technique by determining the ratio of variations present in the pixel intensity. This is said to be stimulus information under the visual perception. Differential excitation computes "the ratio of change in pixel intensity between center pixels against its neighbors". It identifies the local salient visual patterns. The differential excitation is estimated at central pixel $\varepsilon(y_s)$ using Eq. (7).

$$\varepsilon(y_s) = \arctan\left\{\sum_{j=0}^{q-1}\left(\frac{y_j - y_s}{y_s}\right)\right\} \tag{7}$$

Here, the term q denotes the number of neighbors on a circle of radius B and y_s indicates the intensity value of the central pixel. If $\varepsilon(y_s)$ is determined as positive, then it influences the surroundings pixel to be lighter than the present pixel. In contrast, if $\varepsilon(y_s)$ is determined to be negative, it influences the surrounding pixels are darker than the present pixel. The orientation component of LWD is estimated through Eq. (8).

$$\theta(y_s) = \arctan\left\{\frac{y_{\left(\frac{q}{2}-\mathrm{B}\right)} - y_{(\mathrm{B})}}{y_{(q-\mathrm{B})} - y_{\left(\frac{q}{2}-\mathrm{B}\right)}}\right\} \tag{8}$$

The extracted patterns using LWD are denoted as Pt_x^{lwd}.

V. Ensemble Learning-based Feature Extraction for Distracted Driver Detection

A. Ensemble Learning Model

The proposed distracted driver detection model extracts diverse features from the optimized patterns Pt_z^{of} for performing efficient detection performance in the developed model using the suggested HSWOA. CNN-based techniques [27] are selected in the proposed model as it provides better recognition features with the utilization of its more layers and it is independent of the human efforts when implementing its functionalities. Then, the extracted deep features from the pooling layer of the ensemble variant model, where the ResNet50 [28], inceptionv3 [29] and xception [29] approaches are used. ResNet50 in the proposed model contains huge number of layers, which can be trained without enhancing the training error percentage. Similarly, inceptionv3 in the proposed model gains high performance and effectively utilizes the computing resources with less computation load. Xception provides superior performance when utilizing in the larger image classification dataset and ensures higher computational efficiency. The significant features are obtained from the ensemble variant model from the extracted deep features for the distracted driver detection phase. Finally, the extracted features from the three approaches are concatenated and put forwarded to the O-LSTM-based detection. The developed EV-CNN-based feature extraction is depicted in Fig. 4.

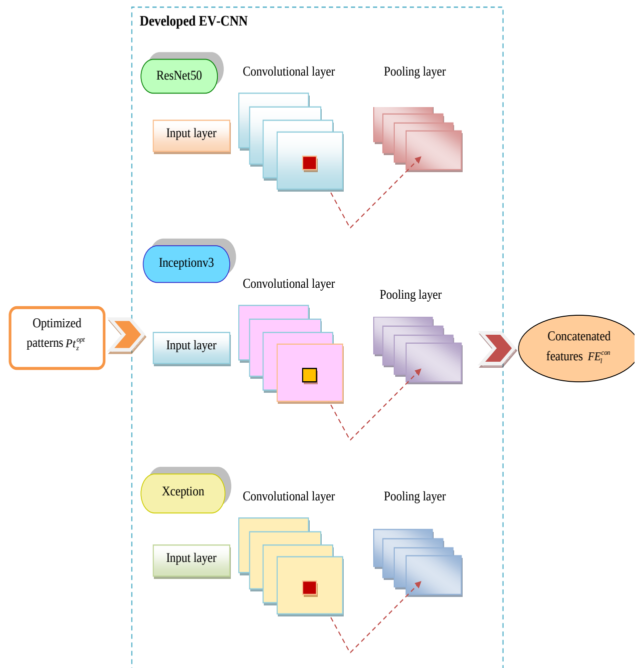


Fig. 4. Feature Extraction of Proposed Distracted Driver Detection Model using EV-CNN.

B. ResNet Model

The proposed distracted driver detection model uses the ResNet50 technique for extracting the significant features from the optimal fusion-based pattern images. The ResNet [30] ensures improved performance even when more number of images is given into it. This model is developed using the skipping connections that is made on the two to three layers with ReLU and batch normalization. The residual learning is applicable to the multiple layers of architecture. The residual block of the ResNet is computed using the Eq. (9).

$$Ol = J(in, U + in) \tag{9}$$

Here, the term Ol denotes the output layer, in shows the input layer, and the residual map are represented by the function J. The residual block on the ResNet is processed when the dimensions of the input layer and output layer are the same. Additionally, each block of the ResNet contains two or three layers. The first two layers of the ResNet express the "GoogleNet by doing convolution 7×7" and max-pooling at the size of 3×3. The extracted features from the pooling layer of ResNet50 are represented as FE_g^{RES} that is computed with the number of 400 features.

C. Inception Model

The extracted deep features DF_f^{cnn} are used for obtaining the essential features from Inceptionv3 approach for the proposed distracted driver detection model. Inception model is used for clustering the sparse convolution kernel structure into the diverse dense sub-convolution kernel combinations. Convolutional filters with various sizes are employed for acquiring the diverse receptive fields. The training procedure of the inception is described initially with the input deep features into the three convolution layers of the inception model. Three convolution layers are performed with kernels of 3×3 along with the one max pooling layer for extracting the low latitude features from the input deep features. Further, two convolution layers are used with the help of kernels of 1×1 and 3×3 and also with the one max pooling layer are utilized for extracting the features. Then, the three pooling layer is used for obtaining the high-dimensional features from the input deep features. The extracted features from the pooling layer of inception model are counted to be 400, which are indicated by FE_h^{ine}.

D. Xception Model

The proposed distracted driver detection model uses the xception technique for extracting the significant features from the deep features DF_f^{cnn}. Xception model employs 36 convolutional layers for performing the feature extraction based on the CNN. These convolutional layers are sorted accordingly into the 14 modules, which are covered by the linear residual connections. The input deep features DF_f^{cnn} are considered to the Convolutional kernels for extracting the features. The output of the convolution layer is computed using Eq. (10).

$$B_P = f\left(CK_P * B_{P-1} + c_P\right) \tag{10}$$

Here, the term B_P denotes the output of P^{th} convolution layer; the activation function is indicated by $f(\cdot)$, the convolution kernel is represented by CK_P, convolution operation is denoted by "$*$" and the offset parameter is shown by c_P. The feature maps are given into the separable convolution for feature extraction. Here, the separable convolution is utilized for minimizing the parameters count and computation complexity in the model. The extracted features from the pooling layer of the Xception model are counted to be 400 that are denoted by FE_k^{xcep}.

Finally, the extracted features from ResNet50, Inceptionv3 and Xception are concatenated into a single set of features, which is expressed by $FE_l^{con} = FE_g^{RES} + FE_h^{ine} + FE_k^{xcep}$, where $l = 1,2,\ldots,L$ and L denotes the total number of concatenated features for the proposed distracted driver detection model.

VI. Hybrid Meta-heuristic-based Distracted Driver Detection

A. LSTM-based Detection Model

The concatenated features FE_l^{con} are utilized in the O-LSTM [31] for classifying distracted driver behaviors to avoid traffic accidents. The classification phases are made stronger by the recurrent structures of the deep learning algorithm and so, the external memories are needless for storing the output. The recurrent structures of the LSTM classifiers provide less complexity in computation. The four components such as "cells, input gate, output gate and forget gate" are presented in the LSTM network. The cell carries the data and passed to the input and output gate. The forget gate is initially used for determining the information passed through the network that is shown in Eq. (11).

$$c_t = \sigma\left(B_c \cdot \left[k_{t-1}, \left(FE_l^{con}\right)_t\right] + w_c\right) \tag{11}$$

Here, the terms σ and k_t are correspondingly shown as sigmoid activation function and the output of the hidden state. The weight matrices are described as B_c, B_g, B_h, B_q and the input variable is given as FE_l^{con}. Then, the cell output, output gate and forget gate are represented as g_t, q_t and G_t, respectively. The biased values of these gates are portrayed as w_c, w_f, w_h, w_q. The input gate is formulated in Eq. (12).

$$g_t = \sigma\left(B_g \cdot \left[k_{t-1}, \left(FE_l^{con}\right)_t\right] + w_f\right) \tag{12}$$

Further, it updates a new cell states using sigmoid function that generates the new vector $\widehat{G}_t$ that is shown in Eq. (13).

$$\widehat{G}_t = \tan k\left(B_h \cdot \left[k_{t-1}, \left(FU_{g*}^{opt}\right)_t\right] + w_h\right) \tag{13}$$

For updating the old cell into new cell, the earlier state is integrated with forget gate and added more parameters that is given in Eq. (14).

$$G_t = g_t * G_{t-1} + c_t * \widehat{G}_t \tag{14}$$

Finally, the output gate provides the cell state using the output of the sigmoid of output gates that are given in Eq. (15) and Eq. (16).

$$q_t = \sigma\left(B_q \cdot \left[k_{t-1}, \left(FE_l^{con}\right)_t\right] + w_q\right) \tag{15}$$

$$k_t = q_s * \tan k\left(d_t\right) \tag{16}$$

The sigmoid activation function of the LSTM classifier is represented as σ with hyperbolic tangent $\tanh$. The distracted driver behaviors are detected using the developed O-LSTM model that is depicted in Fig. 5.

B. Objective Model

The detected distracted behavior of the driver shows the accurate detection outcomes based on the developed O-LSTM with the suggested HSWOA. The detected results of the proposed distracted driver detection model are obtained as the various classes of safe driving and distracted driving behaviors. The objective function of the proposed distracted driver detection model is to maximize the detection accuracy and precision that is given in the Eq. (17).

$$Ofun_{(2)} = \underset{\left\{HN_{lstm}^{hn}, Ep_{lstm}^{ep}\right\}}{\arg\min}\left(\frac{1}{ay + pr}\right) \tag{17}$$

Here, the term HN_{lstm}^{hn} and Ep_{lstm}^{ep} are represented as hidden neurons and the number of epochs of the LSTM, respectively. The developed HSWOA optimizes the number of hidden neurons in the range of [5,255] and number of epochs in the interval of [5, 20]. Accuracy ay is measured as the "closeness of the measurements to a specific value" as given the Eq. (18).

$$accy = \frac{\left(t^p + t^n\right)}{\left(t^p + t^n + f^p + f^n\right)} \tag{18}$$

Here, the true positive and true negative values are shown as t^p and t^n, respectively and false positive and false negative values are given as f^p and f^n, respectively. Precision pr is explained as "the fraction of relevant instances among the retrieved instances" as given in Eq. (19).

$$pr = \frac{t^p}{t^p + f^p} \tag{19}$$

The solution encoding of the suggested HSWOA is shown in Fig. 6.

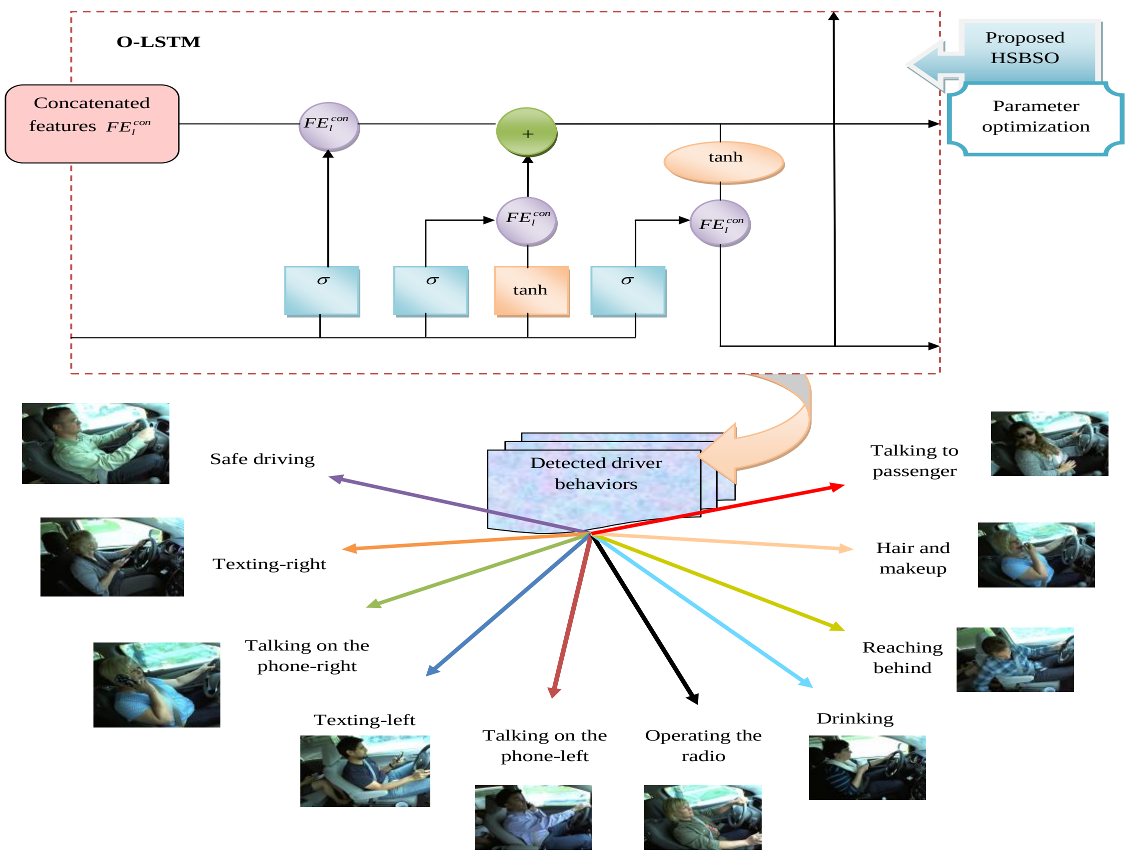


Fig. 5. Proposed O-LSTM-based Distracted Driver Detection Model

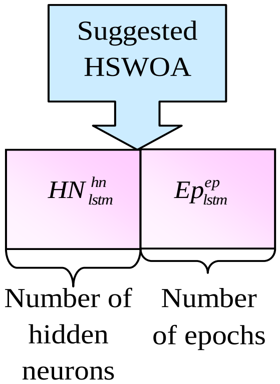


Fig. 6. Solution encoding of the Proposed Distracted Driver Detection Model using the Developed HSWOA.

C. Proposed HSWOA

The proposed distracted driver detection model employs the suggested HSBSO for reducing the training complexity in LSTM by optimizing its hidden neurons and epochs and also tuning the weights of the extracted deep features from the developed EV-CNN. WOA [31] algorithm is chosen for the proposed model due to the ability to solve the real-time optimization problems, which is also simple and easy to implement. But, it does not contain the capability to balance the exploration and exploitation phase. Also, it is easily falls into the local optimum problem. To overcome these challenges of WOA, a meta-heuristic algorithm named SSA [32] is adopted into WOA. Thus, the HSBSO algorithm is developed by adopting the features of SSA into the WOA algorithm. The SSA algorithm is capable of generating the optimal solution for high-dimension optimization problems at the limited time period. In the proposed HSBSO, two random variables c and d are introduced, which are determined by the fitness-based computations that are shown in Eq. (20) and Eq. (21).

$$c = abs\left(f\left(j\right) - bestfit\right) \tag{20}$$

$$d = abs\left(worstfit - f\left(j\right)\right) \tag{21}$$

Here, the term $f\left(j\right)$ denotes the fitness of the current solution and the best fitness value and worst fitness value are denoted by $bestfit$ and $worstfit$, respectively. If the condition $\left(c > d\right)$ is satisfied, then the position update takes place using the SSA otherwise, the position update takes place according to the WOA algorithm.

SSA [26] is implemented based on the motivation of the jumping mechanism and gliding strategies of the flying squirrels. The entire optimization process was carried out in the summer and winter phase. Initially, considered the count of the population is PP and the upper bound and lower bound of the search space are regarded as z_U and z_L . Every individual in the population is generated according to Eq. (22).

$$z_a = z_L + rnd\left(1, dd\right) \times \left(z_U - z_L\right) \tag{22}$$

Here, the term rnd denotes the random number at the range of 0 to 1, z_a indicates the a^{th} individual and the dimension of the search space is expressed by dd . The SSA algorithm needs only one squirrel at the tree and so, equal number of trees and squirrels are present in the search space. Here, the squirrels are categorized based on the fitness of the population into three diverse varieties such as individuals located at hickory trees f_{hk} with the minimum fitness, individuals located at acorn trees f_{ac} with the ranking of second fitness and individuals located at normal trees f_{nt} . Then, the foraging behavior is designed mathematically as follows.

The position of the flying squirrels in the acorn nut trees f_{ac} that jumps to hickory nut tree. Here, the new position of the squirrels is computed in Eq. (23).

$$z_{ac}^{new} = \begin{cases} z_{ac}^{old} + rd_G GC\left(z_{hk}^{old} - z_{ac}^{old}\right) & if\ r_1 \geq PR_{pd} \\ random\ position & others \end{cases} \tag{23}$$

Here, the gliding constant is indicated by GC and the random function r_1 is determined from the interval $\left[0,1\right]$. The random gliding distance is shown by rd_G that is computed in Eq. (24).

$$rd_G = \frac{hh_G}{\tan\left(\phi\right) \times cv} \tag{24}$$

The terms cv and hh_G are considered as the constant values and the gliding angle $\tan\left(\phi\right)$ is calculated in Eq. (25).

$$\tan\left(\phi\right) = \frac{F}{M} \tag{25}$$

The drag force is represented by F and the lift force is expressed by M and these two forces are computed as follows.

$$F = \frac{1}{2\rho v^2 s C_{dd}} \tag{26}$$

$$M = \frac{1}{2\rho v^2 s C_{ll}} \tag{27}$$

Here, the terms ρ , v , s , C_{ll} and C_{dd} are indicated as the constants. Several numbers of squirrels are located on the normal trees that are migrated to the acorn nut trees for getting food resources, where the new position is decided by Eq. (28).

$$z_{nt}^{new} = \begin{cases} z_{nt}^{old} + rd_G GC\left(z_{ac}^{old} - z_{nt}^{old}\right) & if\ r_2 \geq PR_{pd} \\ random\ position & others \end{cases} \tag{28}$$

Here, the term r_2 denotes the random function that is determined from the interval $\left[0,1\right]$. Similarly, some squirrels in the normal tree are changed its location to the hickory nut tree, in which the new position of the squirrel is formulated in Eq. (29).

$$z_{nt}^{new} = \begin{cases} z_{nt}^{old} + rd_G GC\left(z_{hk}^{old} - z_{nt}^{old}\right) & if\ r_3 \geq PR_{pd} \\ random\ position & others \end{cases} \tag{29}$$

Here, the term r_3 indicates the random function that is determined from the interval $\left[0,1\right]$.

Estimate the seasonal monitoring condition: This step is to avoid the local minima problem. Here, the seasonal constant SE and the respective minimum values are correspondingly computed in Eq. (30) and Eq. (31).

$$SE_c^m = \sqrt{\sum_{t=1}^{q}\left(z_{ac,t}^m - z_{hk,t}\right)^2}, m = 1,2,3 \tag{30}$$

$$SE_{cMIN} = \frac{10E-6}{365^{\frac{j}{(j_{\max})}\over 2.5}} \tag{31}$$

Here, the number of iterations is indicated by j and maximum number of iterations is denoted by $j_{\max}$. The season is determined through the condition $SE_c^m < SE_{cMIN}$. When $SE_c^m < SE_{cMIN}$ is satisfied, the winter season is started or else the season will be unchanged. When the summer season arises, all individuals update their position to f_{hk} that is shown in Eq. (32).

$$z_{nt}^{new} = z_L + LEVY\left(nn\right)\times\left(z_U - z_L\right) \tag{32}$$

The levy distribution is known to be strong tool for enhancing the global exploration capability for most of the optimization algorithms that is given in Eq. (33).

$$LEVY\left(y\right) = 0.01\times\frac{dr_p \times \sigma}{\left|dr_q\right|^{1/\beta}} \tag{33}$$

Here, the two different functions denoted by dr_p and dr_q that is in the range of $\left[0,1\right]$ and the constant is indicated by β and also the term σ is computed through Eq. (34).

$$\sigma = \left(\frac{\Gamma\left(1+\beta\right)\times\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{\left(1+\beta\right)}{2}\right)\times\beta\times 2^{\left(\frac{(\beta-1)}{2}\right)}}\right) \tag{34}$$

Finally, the algorithm stops when the maximum count of iterations is fulfilled.

WOA [31] is encouraged by the hunting behavior of humpback whales. These whales usually catch the pack of small fishes in their close surfaces for their food sources. This foraging happens by generating various bubbles in the circular direction around their prey. This unique behavior is known to be “bubble-net feeding method of humpback whales”.

1) “Encircling prey”: Every whale is indicated as the search agent. The abilities of the humpback whales are intended to identify the prey location and encircle them. WOA is initiated the searching without knowing the position of the optimal solution in the search space and then, the target food source location is considered as the current optimal solution. If the best candidate is identified, then all other candidates will update their position based on the best solution. The encircling behavior of the whales is represented as given in the Eq. (35).

$$\vec{B} = \left|\vec{A}\cdot\vec{z}^*\left(j\right) - \vec{z}\left(j\right)\right| \tag{35}$$

$$\vec{z}\left(j+1\right) = \vec{z}^*\left(j\right) - \vec{C}\cdot\vec{B} \tag{36}$$

Here the current iteration among the search agent is given as j, the position of the current solution and the best solution are defined as z and z^*, respectively and the coefficient vectors are denoted as $\vec{C}$ and $\vec{A}$ that are formulated in the Eq. (37) and Eq. (38).

$$\vec{C} = 2\cdot\vec{b}\cdot\vec{V} - \vec{b} \tag{37}$$

$$\vec{A} = 2\cdot\vec{V} \tag{38}$$

Here, the random vector is given as V that lies in the interval of $\left[0,1\right]$ and the convergence factor is shown as b, which simultaneously reduces from 2 to 0 with maximizing the number of iterations.

2) “Bubble net attacking method”: This hunting behavior of whales is classified into two phases namely “shrinking and encircling of the prey” and “spiral upward encirclement and suppression”. The shrinking and encircling method is attained by decreasing the value of $\vec{V}$. If $\left|C\right| \leq 1$ satisfied, then the new position is updated for the search agent using the Eq. (36). The spiral-based position updating takes place for reaching their food destination which is equated in Eq. (39).

$$\vec{z}\left(j+1\right) = \vec{B}^{'}\cdot e^{sr}\cdot\cos\left(2\cdot\pi\cdot r\right) + \vec{z}^*\left(j\right) \tag{39}$$

Here, the random number is denoted as r in the range of $\left[-1,1\right]$ and the constant s is utilized for indicating the shape of logarithmic spiral. Term $\vec{B}^{'} = \left|z^* - z\right|$ shows the distance between the search agent and best solution. WOA selects the hunting mechanism among these two behaviors by assuming the same probability for updating the positions of the search agents that is shown in the Eq. (40).

$$\vec{z}(j+1)=\begin{cases}\vec{z}^*(j)-\vec{C}\cdot\vec{B} & if\ P<0.5\\ \vec{B}'\cdot e^{sr}\cdot\cos(2\cdot\pi\cdot r)+\vec{z}^*(j) & if\ P\geq 0.5\end{cases} \quad (40)$$

3) Searching for prey: WOA contains the ability to balance the "exploitation and exploration phase" due to the size of vector C. If $C \geq |1|$ satisfied, then the position of search agents is updated through the random vector for establishing the global optimal solution by reducing the local minima. This is depicted in the Eq. (41).

$$\vec{B}=\left|\vec{A}\cdot\vec{z}_{RN}-\vec{R}\right| \quad (41)$$

$$\vec{z}(j+1)=\vec{z}_{RN}-\vec{C}\cdot\vec{B} \quad (42)$$

Here, the random vector is indicated as z_{RN} and the best optimal solution that is position of prey is identified as z^*. The pseudo code of the proposed HSBSO is given in Algorithm 1.

Algorithm 1: Proposed HSBSO

Initialize the population along with its parameters

Evaluate the fitness of every individuals

Compute two variables c and d using Eq.(20) and Eq. (21), respectively

Determine the best and worst fitness values

While (until reaching the stopping criterion) do

If $(c>d)$

Solution updating based on SSA

Else

Solution updating based on WOA

End

End while

Obtain the best optimal solution

End

The hybrid optimization algorithm is used for enhancing the overall performance of the proposed model especially in the pattern extraction phase and detection phase, which reduces the overall burden of the proposed model. The flowchart of the proposed HSWOA is depicted in Fig. 7.

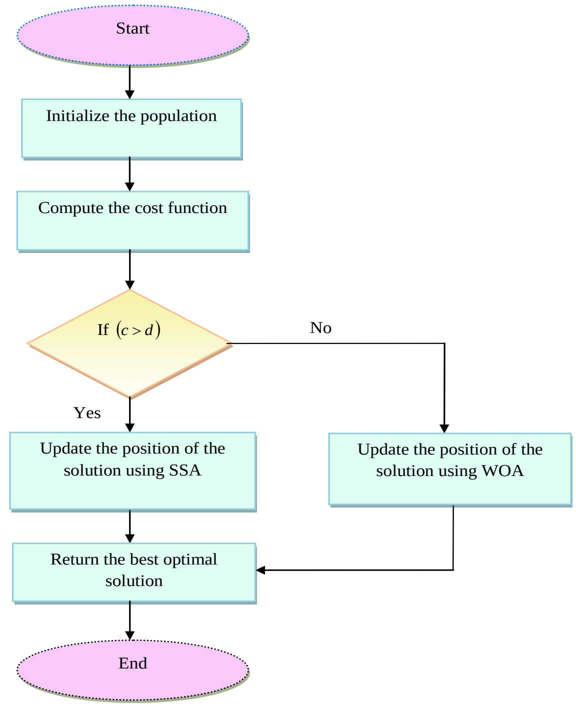


Fig. 7. Proposed HSWOA for Distracted Driver Detection Model.

VII. Result and Discussion

A. Experimental Setup

The platform used for implementing the proposed distracted driver detection model was Python and certain experimental analysis was performed for testing the suggested model using several quantitative measures. The experimental analysis was mainly focused on comparing the performance between the proposed model and conventional meta-heuristic algorithms and also with different classifiers. The involved quantitative measures were classified as positive or Type I measures and negative measures or Type II measures. This experimental analysis was undergone with a population count as 10 and maximum iterations count as 10 for the proposed distracted driver detection model. The proposed HSWOA was compared with other meta-heuristic algorithms like "Particle Swarm Optimization (PSO) [33], Grey Wolf Optimizer (GWO) [34], SSA [32] and WOA [31] and machine learning algorithms like Neural Network (NN) [30], Convolutional Neural Network (CNN) [27], Long Short Term Memory (LSTM) [35], and Ensemble Learning Model (ELM) [17]".

B. Performance Metrics

The performance of proposed distracted driver detection model is evaluated using various quantitative measures that are given as follows.

1) MCC (mc) is "a measure of the quality of binary classifications of testing" as given in the Eq. (43)

$$mc=\frac{t^P\times t^N-f^P\times f^N}{\sqrt{\left(t^P+f^P\right)\left(t^P+f^N\right)\left(t^N+f^P\right)\left(t^N+f^N\right)}} \quad (43)$$

2) Specificity (spt) is "the proportion of negatives that are correctly identified" as represented in the Eq. (44):

$$spt = \frac{t^N}{t^N + f^P} \tag{44}$$

3) NPV (nv) is described as "the sum of all persons without disease in testing" as denoted in Eq. (45):

$$nv = \frac{t^N}{t^N + f^N} \tag{45}$$

4) F1-score (F1) is determined as "the measurement of the accuracy in the conducted test" as given in Eq. (46):

$$F1 = 2 \times \frac{2t^P}{2t^P + f^P + f^N} \tag{46}$$

5) FDR (fdr): is "a method of conceptualizing the rate of errors in testing when conducting multiple comparisons" as denoted in Eq. (47):

$$fdr = \frac{f^P}{f^P + t^P} \tag{47}$$

6) Sensitivity (sen): is "the proportion of positives that are correctly identified" as denoted in Eq. (48):

$$sen = \frac{t^P}{t^P + f^N} \tag{48}$$

7) FPR (frp): is defined as "the ratio between the numbers of negative events wrongly categorized as positive (false positives) and the total number of actual negative events" as given in Eq. (49):

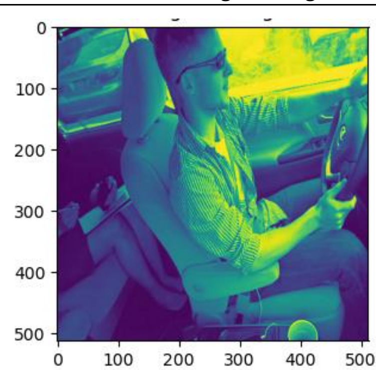
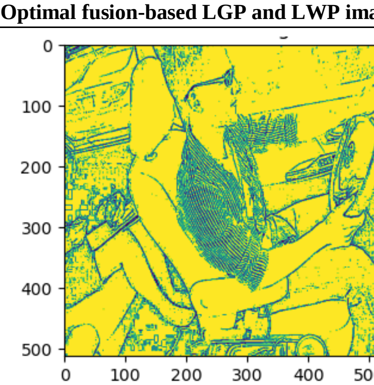
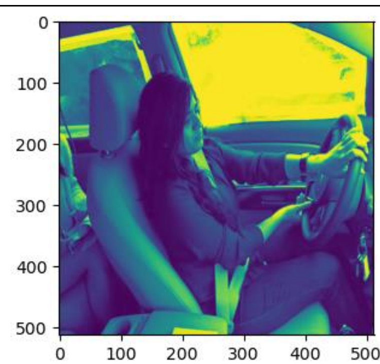
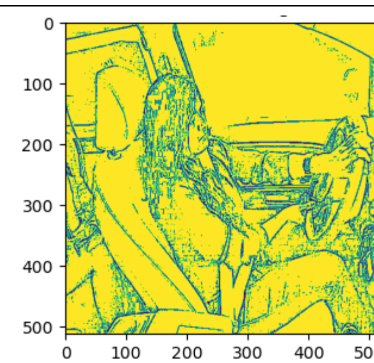
$$fpr = \frac{f^P}{f^P + t^N} \tag{49}$$

8) FNR (fnr): is "the proportion of positives which yield negative test outcomes with the test" as given in Eq. (50):

$$fnr = \frac{f^N}{f^N + t^P} \tag{50}$$

C. *Optimal Fusion-based Pattern Extracted in the Proposed Distracted Driver Detection Model*

The resultant image of the optimized patterns from the LGP and LWP in the proposed model is given in Fig. 8.

Image description	**Original images**	**Optimal fusion-based LGP and LWP images**
Image 1		
Image 2		

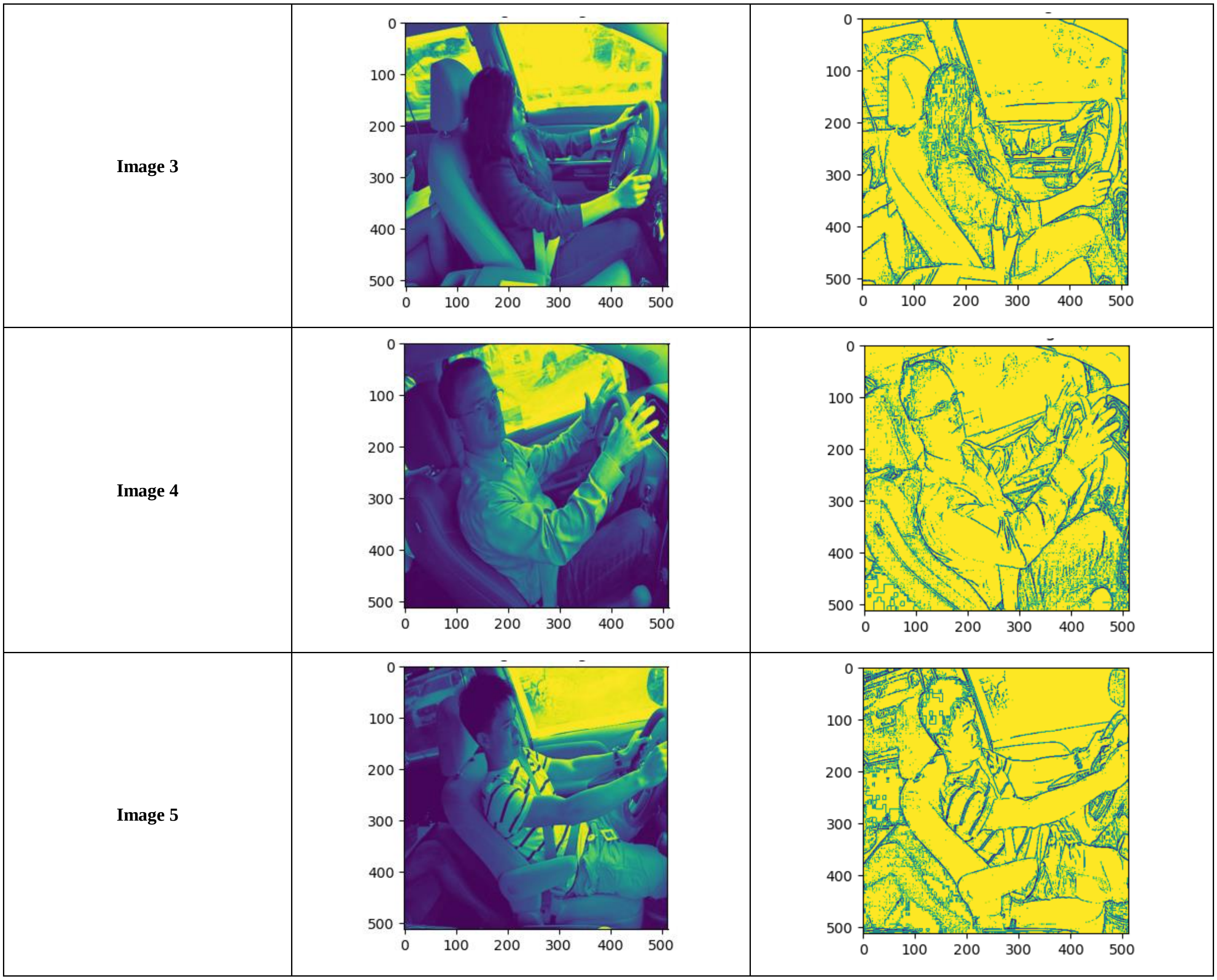


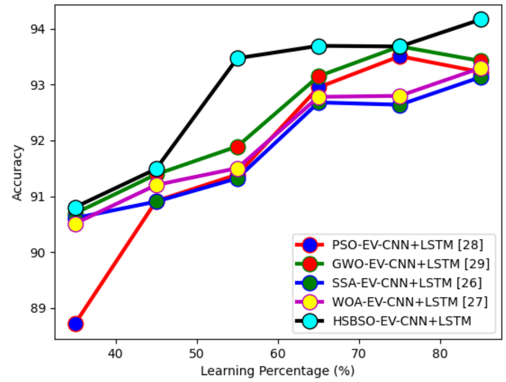
Fig. 8. Original and Optimal Fusion-based LGP and LWP Images of the Safe Drivers and Distracted Drivers.

D. Performance Analysis on Accuracy based on different Meta-heuristic Algorithms

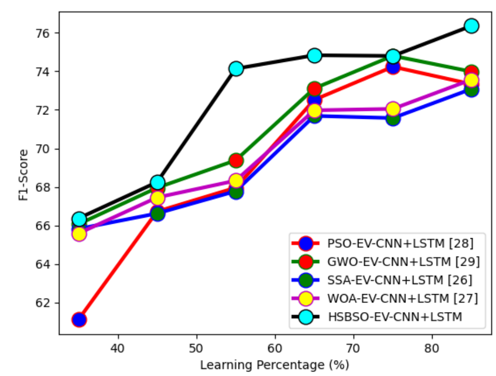
The performance analysis on accuracy for the proposed distracted driver detection model is tested by comparing with the different meta-heuristic algorithms as shown in Fig. 9 at varying learning percentages. The performance of the proposed HSWOA-EV-CNN+LSTM shows 0.21%, 0.64%, 0.53%, and 0.21% improved accuracy than the PSO-EV-CNN+LSTM, GWO-EV-CNN+LSTM, SSA-EV-CNN+LSTM and WOA-EV-CNN+LSTM, respectively at the learning percentage as 70 on dataset 2. While observing the accuracy and F1-Score of the proposed model, it shows a slight low performance at the iteration of 40, which has been increased at the iterations of 50, 60 and 70 that performance is maintained with a slight increase in performance until the iteration of 80. Thus, the overall performance of the proposed algorithm for the developed distracted driver detection model secures enhanced performance than other existing methods.

E. Performance Analysis on Proposed Model with different Classifiers

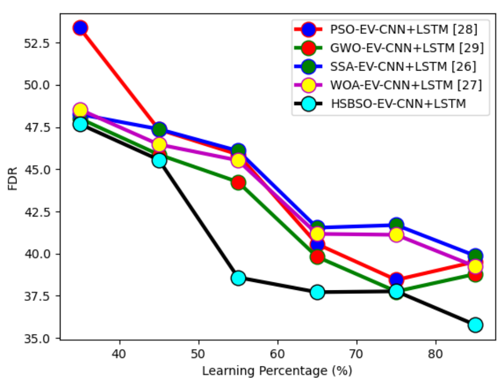
The performance analysis for the proposed distracted driver detection model is evaluated by comparing with different classifiers is depicted in Fig. 10 at varying learning percentages. The performance of the proposed HSWOA-EV-CNN+LSTM contains 6.66%, 6.66%, 7.86% and 4.34% higher accuracy than the NN, CNN, LSTM and ELM, respectively at the learning percentage of 35 on dataset 3. When considering the FNR of the proposed model, it ensures an increased performance rate on increasing the number of iterations. In the MCC analysis on the proposed model, it secures 0.6 values at iteration 35 that is increased to 0.7 at the iteration of 55 and in the further two iterations of 65 and 75, the MCC value has been enhanced than in the prior iterations. Thus, the overall performance of the proposed HSWOA-EV-CNN+LSTM for the proposed distracted driver detection model shows higher performance than other conventional methods of distracted driver detection.


Accuracy
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

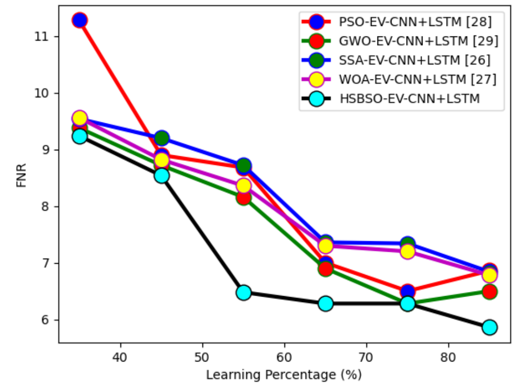
(a)


F1-Score
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

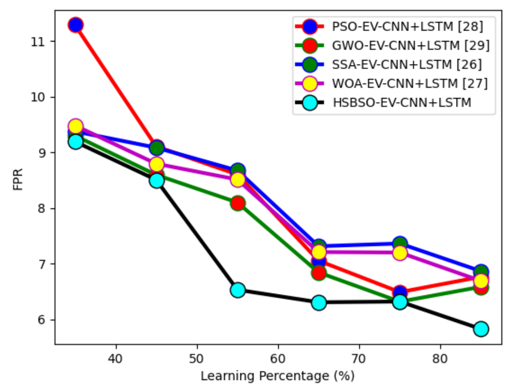
(b)


FDR
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

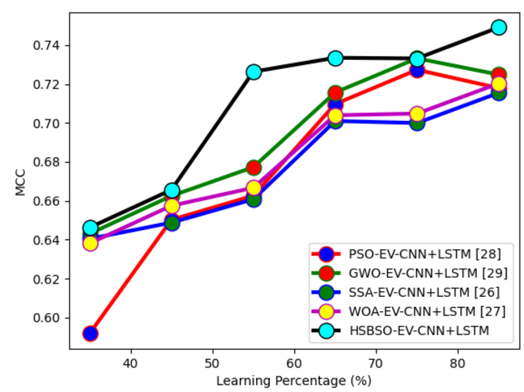
(c)


FNR
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

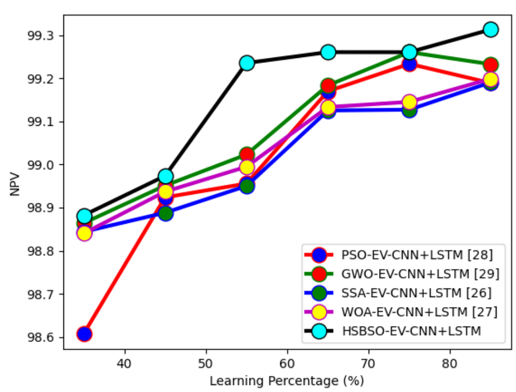
(d)


FPR
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

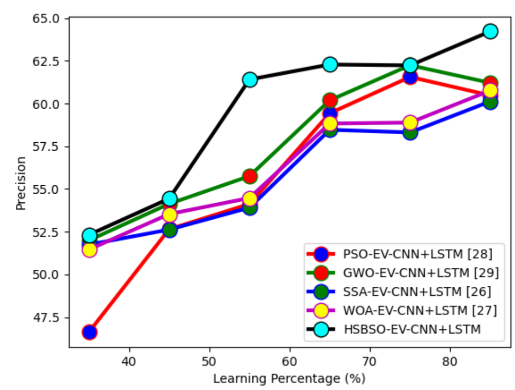
(e)


MCC
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

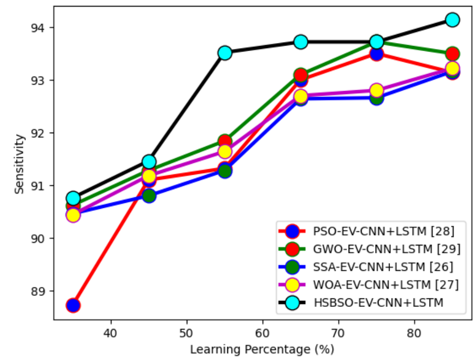
(f)


NPV
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

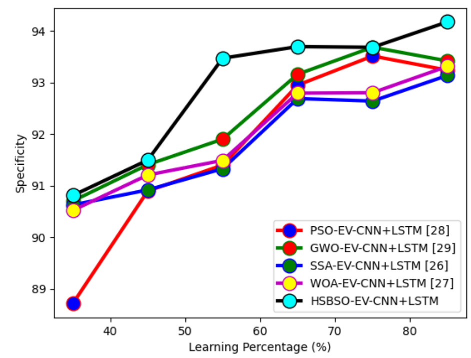
(g)


Precision
Learning Percentage (%)
PSO-EV-CNN+LSTM [28]
GWO-EV-CNN+LSTM [29]
SSA-EV-CNN+LSTM [26]
WOA-EV-CNN+LSTM [27]
HSBSO-EV-CNN+LSTM

(h)

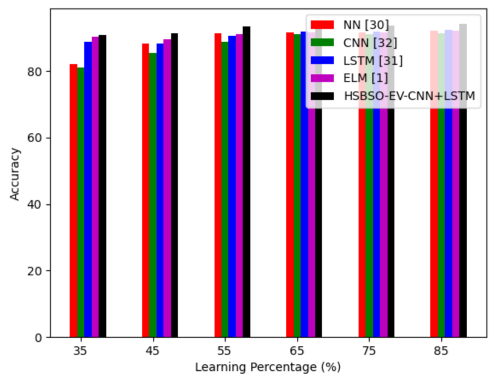


(i)

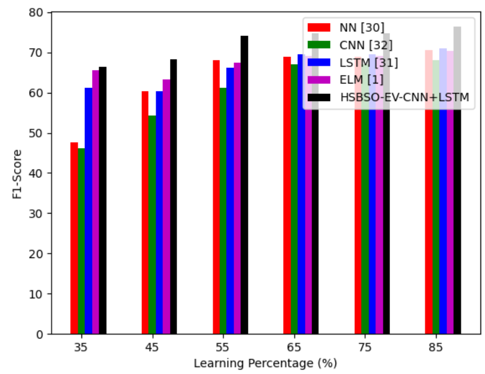


(j)

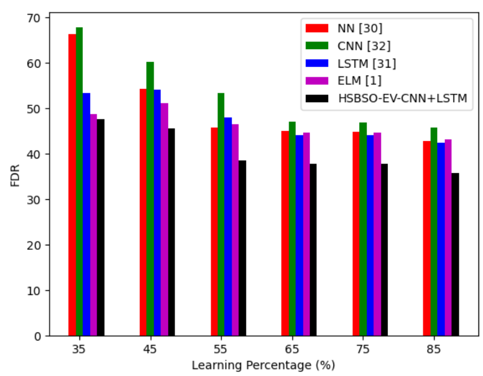
Fig. 9. Performance Analysis on Suggested Distracted Driver Detection Model with Conventional Meta-heuristic Algorithms in Terms of "(a) Accuracy, (b) F1-Score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Presicion, (i) Sensitivity, (j) Specificity".



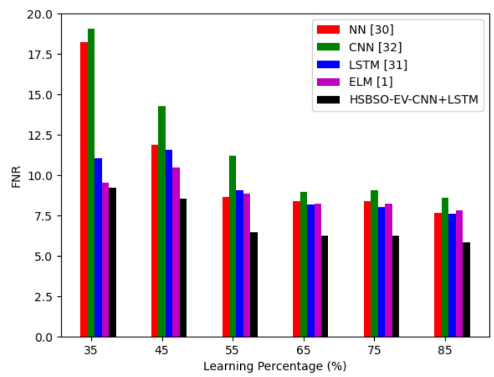
(a)



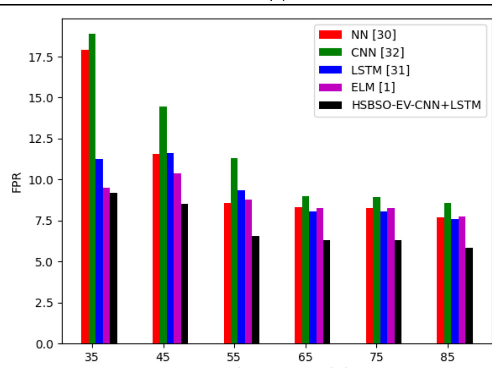
(b)



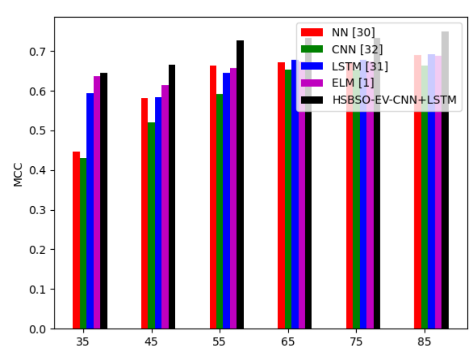
(c)



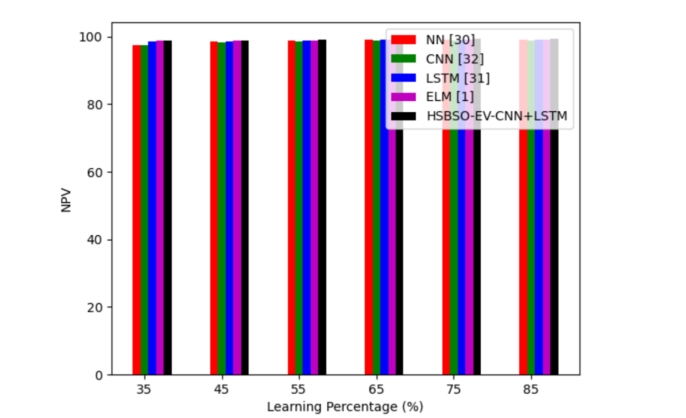
(d)



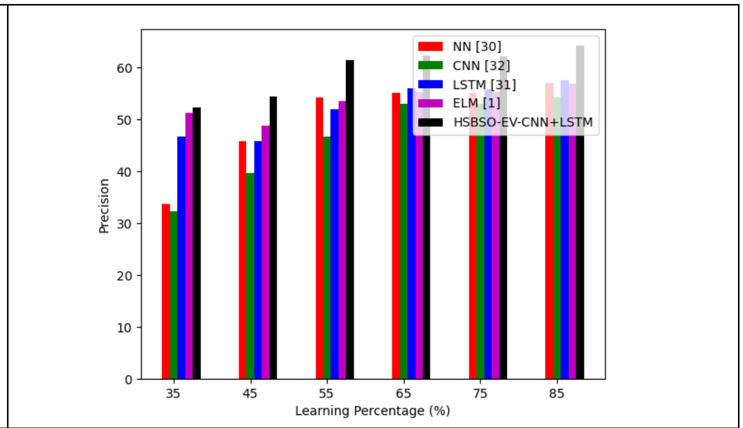
(e)



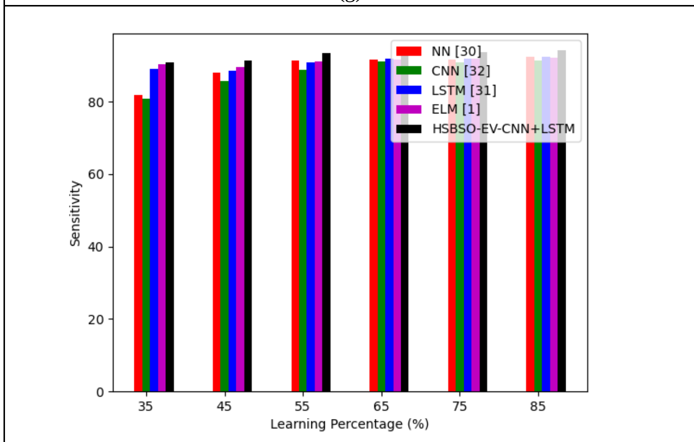
(f)



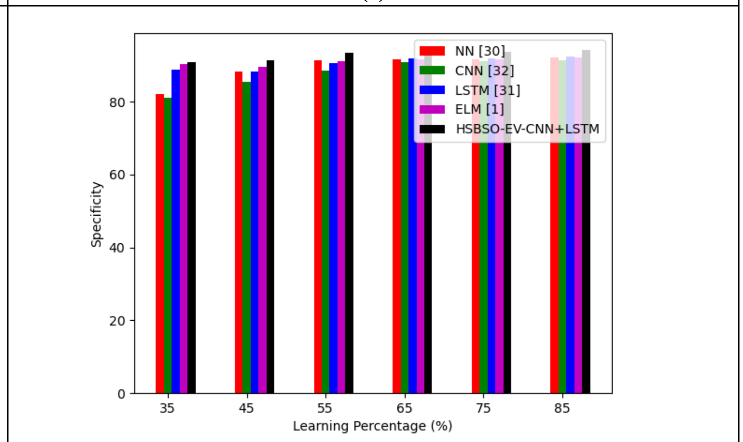
(g)



(h)



(i)



(j)

Fig. 10. Performance Analysis Accuracy of the Suggested Distracted Driver Detection Model with Existing Classifiers in Terms of "(a) Accuracy, (b) F1-Score, (c) FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Presicion, (i) Sensitivity, (j) Specificity".

F. Overall Performance Analysis for the Proposed Model based on different Meta-heuristic Algorithms

The overall performance analysis for the proposed distracted driver detection model is estimated by comparing it with the different meta-heuristic algorithms as depicted in Table II. The performance of the proposed HSWOA-EV-CNN+LSTM is 0.36%, 0.59%, 0.62% and 0.38 superior to the PSO-EV-CNN+LSTM, GWO-EV-CNN+LSTM, SSA-EV-CNN+LSTM, and WOA-EV-CNN+LSTM, respectively when the observing the performance of precision. In all the performance measures, the proposed model shows enhanced performance in detecting distracted driver behaviors through the given images. Therefore, the proposed distracted driver detection model enhances its performance more than the existing methods.

G. Overall Performance Analysis for the Proposed Model based on different Classifiers

The overall performance analysis of the proposed distracted driver detection model is tested by comparing it with the different classifiers as expressed in Table III. The performance of the proposed HSWOA-EV-CNN+LSTM provides 1.88%, 3.55%, 4.4% and 1.06% enhanced MCC than the NN, CNN, LSTM and ELM, respectively. The proposed distracted driver detection model using the suggested HSBSO shows enhanced performance in all quantitative measures particularly provides higher values in the accuracy, precision, specificity and MCC than the conventional techniques. Therefore, the proposed distracted driver detection model provides improved performance than the existing methods.

TABLE II. OVERALL PERFORMANCE ANALYSIS OF SUGGESTED DISTRACTED DRIVER DETECTION MODEL WITH DIFFERENT META-HEURISTIC ALGORITHMS

Measures	*PSO [28]*	*GWO [29]*	*SSA [26]*	*WOA [27]*	*HSWOA-EV-CNN+LSTM*
"Accuracy"	0.9351	0.93686	0.9264	0.928	0.93684
"Sensitivity"	0.935	0.9372	0.9266	0.928	0.9372
"Specificity"	0.935111	0.936822	0.926378	0.928	0.9368
"Precision"	0.615537	0.622393	0.583061	0.588832	0.622311
"FPR"	0.064889	0.063178	0.073622	0.072	0.0632
"FNR"	0.065	0.0628	0.0734	0.072	0.0628
"NPV"	0.992336	0.992607	0.991273	0.991453	0.992607
"FDR"	0.384463	0.377607	0.416939	0.411168	0.377689
"F1-Score"	0.742358	0.748025	0.715742	0.720497	0.747965
"MCC"	0.727266	0.73316	0.699924	0.704787	0.733101

TABLE III. OVERALL PERFORMANCE ANALYSIS OF SUGGESTED DISTRACTED DRIVER DETECTION MODEL WITH DIFFERENT CLASSIFIERS

Measures	**NN [30]**	**CNN [32]**	**LSTM [31]**	**ELM [1]**	**HSWOA-EV-CNN+LSTM**
"Accuracy"	0.9173	0.91058	0.91936	0.91762	0.93684
"Sensitivity"	0.916	0.9094	0.92	0.9178	0.9372
"Specificity"	0.917444	0.910711	0.919289	0.9176	0.9368
"Precision"	0.55214	0.530881	0.558795	0.553091	0.622311
"FPR"	0.082556	0.089289	0.080711	0.0824	0.0632
"FNR"	0.084	0.0906	0.08	0.0822	0.0628
"NPV"	0.989929	0.989067	0.990423	0.990145	0.992607
"FDR"	0.44786	0.469119	0.441205	0.446909	0.377689
"F1-Score"	0.688981	0.670402	0.695284	0.690231	0.747965
"MCC"	0.672149	0.653005	0.678935	0.673661	0.733101

VIII. CONCLUSION

This paper has presented a new distracted driver detection model using suggested HSWOA along with the developed EV-CNN and improved LSTM approach. The gathered input images were initially given into the optimal fusion-based pattern extraction, where the two methods like LGP and LWP were used for generating the optimally fused patterns. These patterns were integrated with the optimized weights based on the proposed HSWOA. Then, the optimized patterns were acquired and considered for the developed EV-CNN approach for extracting the most significant features of the patterns. Further, the extracted features were subjected into the developed O-LSTM, which has been improved by optimizing certain parameters using the developed HSWOA. At last, the distracted driver behaviors were detected using the developed O-LSTM. Through the performance analysis, the proposed model using the developed HSWOA-EV-CNN+LSTM has secured 1.33% enhanced than NN, 1.83% improved than CNN, 2.2% better than LSTM and 0.84% elevated than ELM in terms of accuracy. Therefore, the proposed distracted driver detection model has ensured enhanced performance using developed O-LSTM along the suggested HSWOA than the existing methods for future work; we intend to increase the efficiency of the proposed solution by addressing additional driving style data along with road surface information, which will help improve the generalization of the distracted driver detection model.

REFERENCES

[1] M. Gjoreski, M. Z. Gams, M. Luštrek, P. Genc, J. U. Garbas, and T. Hassan, "Machine Learning and End-to-End Deep Learning for Monitoring Driver Distractions from Physiological and Visual Signals," IEEE Access, vol. 8, pp. 70590–70603, 2020, doi: 10.1109/ACCESS.2020.2986810.

[2] Z. Li, S. Bao, I. v. Kolmanovsky, and X. Yin, "Visual-Manual distraction detection using driving performance indicators with naturalistic driving data," IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 8, pp. 2528–2535, Aug. 2018, doi: 10.1109/TITS.2017.2754467.

[3] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F. Y. Wang, "Driver activity recognition for intelligent vehicles: A deep learning approach," IEEE Transactions on Vehicular Technology, vol. 68, no. 6, pp. 5379–5390, Jun. 2019, doi: 10.1109/TVT.2019.2908425.

[4] C. Y. Chiou, W. C. Wang, S. C. Lu, C. R. Huang, P. C. Chung, and Y. Y. Lai, "Driver Monitoring Using Sparse Representation with Part-Based Temporal Face Descriptors," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 1, pp. 346–361, Jan. 2020, doi: 10.1109/TITS.2019.2892155.

[5] M. Rezaei, M. Terauchi, and R. Klette, "Robust Vehicle Detection and Distance Estimation Under Challenging Lighting Conditions," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 5, pp. 2723–2743, Oct. 2015, doi: 10.1109/TITS.2015.2421482.

[6] S. Wang, Y. Zhang, C. Wu, F. Darvas, and W. A. Chaovalitwongse, "Online prediction of driver distraction based on brain activity patterns," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 1, pp. 136–150, Feb. 2015, doi: 10.1109/TITS.2014.2330979.

[7] N. Vinayaga-Sureshkanth, A. Maiti, M. Jadliwala, K. Crager, J. He, and H. Rathore, "A practical framework for preventing distracted pedestrian-

related incidents using wrist wearables," IEEE Access, vol. 6, pp. 78016–78030, 2018, doi: 10.1109/ACCESS.2018.2884669.

[8] Y. Xing et al., "Identification and Analysis of Driver Postures for In-Vehicle Driving Activities and Secondary Tasks Recognition," IEEE Transactions on Computational Social Systems, vol. 5, no. 1, pp. 95–108, Mar. 2018, doi: 10.1109/TCSS.2017.2766884.

[9] T. Liu, Y. Yang, G. bin Huang, Y. K. Yeo, and Z. Lin, "Driver Distraction Detection Using Semi-Supervised Machine Learning," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 4, pp. 1108–1120, Apr. 2016, doi: 10.1109/TITS.2015.2496157.

[10] C. Craye, A. Rashwan, M. S. Kamel, and F. Karray, "A Multi-Modal Driver Fatigue and Distraction Assessment System," International Journal of Intelligent Transportation Systems Research, vol. 14, no. 3, pp. 173–194, Sep. 2016, doi: 10.1007/s13177-015-0112-9.

[11] A. Gumaei et al., "A deep learning-based driver distraction identification framework over edge cloud," Neural Computing and Applications, 2020, doi: 10.1007/s00521-020-05328-1.

[12] M. Zhu, J. Chen, H. Li, F. Liang, L. Han, and Z. Zhang, "Vehicle driver drowsiness detection method using wearable EEG based on convolution neural network," Neural Computing and Applications, vol. 33, no. 20, pp. 13965–13980, Oct. 2021, doi: 10.1007/s00521-021-06038-y.

[13] M. Shahverdy, M. Fathy, R. Berangi, and M. Sabokrou, "Driver behavior detection and classification using deep convolutional neural networks," Expert Systems with Applications, vol. 149, Jul. 2020, doi: 10.1016/j.eswa.2020.113240.

[14] A. K. Dey, B. Goel, and S. Chellappan, "Context-driven detection of distracted driving using images from in-car cameras," Internet of Things (Netherlands), vol. 14, Jun. 2021, doi: 10.1016/j.iot.2021.100380.

[15] J. Xie, K. Hu, G. Li, and Y. Guo, "CNN-based driving maneuver classification using multi-sliding window fusion," Expert Systems with Applications, vol. 169, May 2021, doi: 10.1016/j.eswa.2020.114442.

[16] B. Baheti, S. Talbar, and S. Gajre, "Towards Computationally Efficient and Realtime Distracted Driver Detection with MobileVGG Network," IEEE Transactions on Intelligent Vehicles, vol. 5, no. 4, pp. 565–574, Dec. 2020, doi: 10.1109/TIV.2020.2995555.

[17] F. Omerustaoglu, C. O. Sakar, and G. Kar, "Distracted driver detection by combining in-vehicle and image data using deep learning," Applied Soft Computing Journal, vol. 96, Nov. 2020, doi: 10.1016/j.asoc.2020.106657.

[18] C. Huang et al., "HCF: A Hybrid CNN Framework for Behavior Detection of Distracted Drivers," IEEE Access, vol. 8, pp. 109335–109349, 2020, doi: 10.1109/ACCESS.2020.3001159.

[19] I. Jegham, A. ben Khalifa, I. Alouani, and M. A. Mahjoub, "Soft Spatial Attention-Based Multimodal Driver Action Recognition Using Deep Learning," IEEE Sensors Journal, vol. 21, no. 2, pp. 1918–1925, Jan. 2021, doi: 10.1109/JSEN.2020.3019258.

[20] C. Ou and F. Karray, "Enhancing Driver Distraction Recognition Using Generative Adversarial Networks," IEEE Transactions on Intelligent Vehicles, vol. 5, no. 3, 2020, doi: 10.1109/TIV.2019.2960930.

[21] A. Pal, S. Kar, and M. Bharti, "Algorithm for Distracted Driver Detection and Alert Using Deep Learning," Optical Memory and Neural Networks (Information Optics), vol. 30, no. 3, 2021, doi: 10.3103/S1060992X21030103.

[22] X. Rao, F. Lin, Z. Chen, and J. Zhao, "Distracted driving recognition method based on deep convolutional neural network," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 1, pp. 193–200, Jan. 2021, doi: 10.1007/s12652-019-01597-4.

[23] D. K. Jain, R. Jain, X. Lan, Y. Upadhyay, and A. Thareja, "Driver distraction detection using capsule network," Neural Computing and Applications, vol. 33, no. 11, pp. 6183–6196, Jun. 2021, doi: 10.1007/s00521-020-05390-9.

[24] N. J. Dunn, T. A. Dingus, S. Soccolich, and W. J. Horrey, "Investigating the impact of driving automation systems on distracted driving behaviors," Accid Anal Prev, vol. 156, p. 106152, Jun. 2021, doi: 10.1016/j.aap.2021.106152.

[25] B. Jun and D. Kim, "Robust face detection using local gradient patterns and evidence accumulation," in Pattern Recognition, Sep. 2012, vol. 45, no. 9, pp. 3304–3316. doi: 10.1016/j.patcog.2012.02.031.

[26] L. Zhang, F. Liu, and J. Tang, "Real-time system for driver fatigue detection by RGB-D camera," ACM Transactions on Intelligent Systems and Technology, vol. 6, no. 2, Mar. 2015, doi: 10.1145/2629482.

[27] M. Wollmer et al., "Online driver distraction detection using long short-term memory," in IEEE Transactions on Intelligent Transportation Systems, Jun. 2011, vol. 12, no. 2, pp. 574–582. doi: 10.1109/TITS.2011.2119483.

[28] C. Ou, Q. Zhao, F. Karray, and A. el Khatib, "Design of an end-to-end dual mode driver distraction detection system," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2019, vol. 11663 LNCS, pp. 199–207. doi: 10.1007/978-3-030-27272-2_17.

[29] J. M. Mase, P. Chapman, G. P. Figueredo, and M. T. Torres, "Benchmarking Deep Learning Models for Driver Distraction Detection." [Online]. Available: https://www.researchgate.net/publication/341625555.

[30] R. Sayed and A. Eskandarian, "Unobtrusive drowsiness detection by neural network learning of driver steering," 2001.

[31] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," Advances in Engineering Software, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[32] M. Jain, V. Singh, and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm," Swarm and Evolutionary Computation, vol. 44, pp. 148–175, Feb. 2019, doi: 10.1016/j.swevo.2018.02.013.

[33] Institute of Electrical and Electronics Engineers, Particle swarm optimization of feedforward neural networks for the detection of drowsy driving.

[34] X. Zhang, Z. Liu, Q. Miao, and L. Wang, "An optimized time varying filtering based empirical mode decomposition method with grey wolf optimizer for machinery fault diagnosis," Journal of Sound and Vibration, vol. 418, pp. 55–78, Mar. 2018, doi: 10.1016/j.jsv.2017.12.028.

[35] S. Masood, A. Rai, A. Aggarwal, M. N. Doja, and M. Ahmad, "Detecting distraction of drivers using Convolutional Neural Network," Pattern Recognition Letters, vol. 139, pp. 79–85, Nov. 2020, doi: 10.1016/j.patrec.2017.12.023.