International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Comparative Effectiveness of Deep Learning Approaches for Drowsiness Detection in a Demographically Diverse Cohort

K R Sumana *[1], Dr. Rekha K S[2]

Submitted: 29/01/2024 **Revised**: 07/03/2024 **Accepted**: 15/03/2024

Abstract: Modern lifestyles often demand individuals to balance multiple responsibilities, leading to inadequate sleep and compromised alertness. This can impact various aspects of life, from productivity at work to safe operation of vehicles. This work acknowledges that drowsiness is not confined to particular situations such as driving; it can manifest during work, studying, or any activity demanding prolonged focus among all aged individuals. The proposed Artificial Intelligence Based Sleep Harness System (ASHS), featuring Deep Learning models, is not only a technological support but also a social solution aimed at caring for society's well-being. By integrating drowsiness detection models with remedial measures, contributes significantly to public safety and health across all age groups. ASHS utilizes deep learning methods such as Convolutional-Neural-Network (CNN), Faster-Region-Based CNN (FR-CNN), and Recurrent-Neural-Networks (RNN) to detect and address drowsiness. This research work monitors facial expressions, eye movements, and head tracking to comprehensively assess drowsiness. When drowsiness is detected, real-time alerts are issued, providing immediate corrective measures. Beyond individual well-being of all aged ones, this ASHS has a societal impact by enhancing safety in transportation, productivity in the workplace, and improved performance in educational settings. Additionally, it aids in elderly care by monitoring sleep quality and health, ensuring timely support.

Keywords: *AI-based Sleep Harness System (ASHS), CNN (Convolutional-Neural-Network), FR-CNN (Faster-Region-Based CNN), RNN (Recurrent-Neural-Networks)*

1. Introduction

In the realm of ensuring safety and well-being, the expertise in detecting drowsiness emerges as a paramount consideration. Beyond a mere concern for individuals of all ages, this expertise holds a pivotal role in addressing an array of societal challenges stemming from sleepiness. As a professional expert in the field, understanding the nuanced impact of drowsiness becomes crucial, as it not only transcends age demographics but stands at the forefront of mitigating broader issues associated with fatigue-induced lapses in alertness. There is an obvious need for a proactive response as occurrences of accidents and errors [1,2] caused by inattention keep increasing. Drowsiness not only results in diminished performance but also poses a threat to safety and overall quality of life. With advancements of our work, the system's capabilities could extend to include normal/tired person drowsiness detection.

2. Related Work

A machine vision methodology has been developed by Bharath Bharadwaj [1] et al. uses a webcam for continuous surveillance of driver fatigue, the system prioritizes the analysis of facial features, specifically eye and mouth movements in relation to head gestures. The core objective is to effectively address the critical need for early detection of drowsiness, introducing a progressive alert system designed to bolster driver safety.

Stephen Danny [2] et al. have developed a The Drowsiness Detection and Theft Prevention system, while innovative, lacks testing across a diverse range of age groups and occupational demographics. It employs facial landmark prediction to categorize drowsiness into three stages, triggering a gentle auditory alert, an urgent alarm, and notifications via WhatsApp and email. Additionally, it incorporates GPS location sharing through an Android app. The system also offers to observe theft prevention with real-time facial recognition using smartphone cameras and allows users to customize drowsiness sensitivity for flexibility and cost-effectiveness.

Face Detection-based Driver Fatigue Detection System proposed from Khan Furqan [3] et.al, continuously captures and analyzes images, to monitor the driver's eye state and issues warnings when drowsiness is detected. Persistent warnings trigger speed reduction and motor shutdown, coupled with email alerts. Research works of Viren Patel [4] et.al, reveals that over a quarter of road accidents stem from driver drowsiness, surpassing those caused by drunken driving. It is note taking that these systems proactively detect drowsiness and issue timely alerts, serving as a proactive measure to prevent road accidents caused by human error. Recent advancements in driver monitoring

[1] Research Scholar, Department of Computer Science and Engineering, The National Institute of Engineering, Mysuru-570008, Visveswaraya Technological University, Belgaum, Karnataka, India
ORCID ID : 0000-0002-1219-6050
[2] Faculty, Department of Computer Science and Engineering, The National Institute of Engineering, Mysuru570008, Visveswaraya Technological University, Belgaum, Karnataka, India
ORCID ID : 0000-0001-5269-8031
** Corresponding Author Email: krsumana@nie.ac.in*

systems from Arun Prakash [5] et.al, uses technology and AI to combat driver fatigue and drowsy driving accidents. It reviews drowsiness detection approaches, proposing a video-based system [12] with machine learning for accuracy assessment. The aforementioned studies on drowsiness detection have primarily concentrated on preventing driver drowsiness to mitigate road fatalities. Drowsiness detection primarily addressed the issue in the context of preventing driver fatigue and reducing road fatalities. However, it is essential to acknowledge that monitoring drowsiness holds significance for a wider demographic encompassing individual of all ages. This extends beyond mere road safety concerns, influencing the overall quality of life for individuals and contributing to enhanced personal performance and well-being.

3. Proposed Work

In the pursuit of enhancing safety through advanced drowsiness detection, the proposed system employs a sophisticated methodology centered around real-time video analysis. Positioned as a critical breakthrough, this methodology revolves around capturing video through a strategically positioned webcam, with a meticulous focus on the user's frontal face, particularly the mouth and eyes. The AI-Based Sleep Harness System, fortified by advanced Deep Learning models, transcends conventional detection methods. Rather than merely identifying drowsiness, it provides users with immediate insights into their physical well-being, empowering them to make informed decisions and proactively address the potential consequences of fatigue. Motivated by a commitment to cultivating a safer, more productive, and healthier lifestyle, this innovative solution serves as a versatile tool, adapting seamlessly to users' unique requirements. Its ultimate objective is to enhance the overall quality of life by addressing a fundamental issue that profoundly influences individuals' daily experiences.

4. Theory and Formula

Use The proposed system initiates by capturing video through a strategically positioned webcam, aimed at capturing the user's frontal face. From this video, frames are extracted to produce 2-D images. These frames are then put through face detection using an object recognition Neural Network. However, before proceeding with feature detection [1,2,12], it is crucial to normalize the face images. The goal of this normalization process is to reduce the influence of variables such as varying camera distances, uneven lighting conditions, and diverse image resolutions. Following a successful face identification, the system goes on to recognize facial landmarks [1,12] in these captured frames, such as the locations of the eyes, nose, and mouth as shown in figure 1.

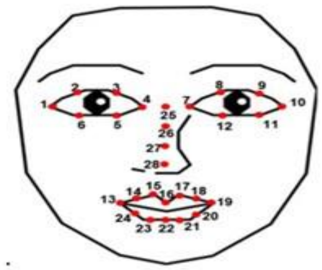

Fig 1. Facial Landmarking

Utilizing Nose-length-Ratio (NLR) data gathered from the facial markers, metrics including the Eye-Aspect-Ratio (EAR), Mouth-Opening-Ratio (MOR), Head-Bending-Angle (HBA), and Head-Position are calculated [1,6,12].

4.1. Eye-Aspect-Ratio (EAR): EAR is a measure of eye openness and helps to find eye-blinks and eye-closure [6,12,14]. It's often calculated using the horizontal and vertical distances between specific facial landmarks, typically the inner and outer corners of the eyes and is calculated as:

$$EAR = ((|P1 - P6|) + (|P3 - P5|) / (2 * |P1 - P4|) \quad (1)$$

Where, ‘P1’ to ‘P6’ are the coordinates of the following landmarks [1,6,12,14]: P1 (left-eye-outer-corner), P2 (left-eye-inner-corner), P3 (left-eye-inner-corner), P4 (left-eye-outer-corner), P5 (top of left eye), P6 (bottom-of-left-eye). Similarly, EAR of right eye shall be computed from P7 to P12 land markings.

$$MOR = ((|P16 - P22|) / (|P13 - P19|)) \quad (2)$$

4.2. Mouth-Opening-Ratio (MOR): MOR, serves the purpose of examining the extent of mouth opening [1,6,12,14] and can be effectively utilized for the detection of yawning, a characteristic indication of drowsiness and the same is calculated as:

$$MOR = ((|P16 - P22|) / (|P13 - P19|)) \quad (3)$$

In this context, P13 and P19 denote the coordinates representing the corners of the mouth on the left and right sides, respectively. Similarly, P16 and P22 indicate the coordinates corresponding to the middle of the upper and lower lips.

4.3. Head-Bending-Angle: The angle of head bending serves as a valuable indicator for identifying notable alterations in head posture, which could potentially signify drowsiness or lack of attentiveness. The formula for calculating the head bending angle [12,14] depends on the reference positions for the head. Assuming initial coordinates P0 and updated coordinates P1 of a specific head landmark can calculated as:

$$HBA = \arccos((P0 \cdot P1) / (|P0| * |P1|))$$

(4)

In this context, P0 represents the landmark's initial position, while p1 signifies the updated position of the same landmark.

4.4. Nose-Length-Ratio: NLR supports to detect user drowsiness, by examining the angle of head bending and inferred the length of the nose projected onto the focal plane of the camera. The NLR [12,14], is calculated from the facial landmarks and it is defined as:

$$NLR = (nose\_length\ (P28 - P25))/ average\_(nose\_length) \quad (5)$$

The above computed variables, amount of tiredness of the user is then determined using a machine learning approach. The DL models CNN, FRCNN and RNN were used to detect real time drowsiness among various category of population. These models use real time fresh images as input to detect sleepiness, signifying the person is either awake or drowsy [7,15,16].

5. CNN Model

To start with CNN model to detect sleepiness [7,6], it accompanies the following input, expected output, the working process, and the algorithm: Input: Train_x, Train_y features and labels of Training Set and represent the characteristics and corresponding labels of the testing dataset (pre-trained data model used). Output: wi, bj: weights and bias of Convolution and Pooling Neural Network (CPNN). Wjk bjk: weights and bias of Full Connection Neural Network (FCNN) with two layers. Required Parameters: CNN model works with max_time represents [7] the upper limit for the number of nACs (network access control) within each ISP (Internet Service Provider). target_error: the training is considered complete when the current training error is smaller than the target error. n_CPNN: the CPNN learning rate. Working Process: wi, bj, w b ik: weights and scaling parameters of CNN (CPNN+FCNN) are initialized with random values. The ongoing simulation time t is initially set to 'r-1' prior to entering the training loop, as Vo. With L(1) = 1>target_error, the mean square error is initialized at simulation time z, or L(t). Following that, the convolutional network architecture, illustrated in figure 2., is perfected as per the specifications outlined in the construction of the CNN model detailed below:

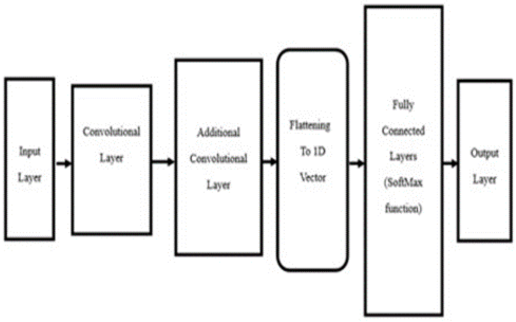

Fig 2. CNN Model Architecture of ASHS

The functionality of each layer was as follows, and the algorithmic process using the CNN model of ASHS is depicted in figure 3.

```
CNN Model Build
Begin
1. Complete the initialization function and set the necessary conditions.
2. while (Max_time and L( )) Target_error
3. for all TrainingSet
4.     According to Train_x, Train_p (prediction of label) is calculated and forward calculation
       based on 1-4 formula
5. end for
6. Re-calculated L(t) = E-1(Train_p(n)- Train_y(n))², where n signifies the overall count of training sets.
7. Δw, Δb, Δw', Δbj¹ are updated
8. w' (t), b' (t), wij(t), b¹(t) are adjusted
9. t++
10. end while
End
```

Fig 3. CNN Model Build Pseudocode

5.1. Convolutional Layer: A collection of flexible filters, referred to as kernels, are applied to the input picture in the structural design of a convolutional layer. Each of these filters is intricately crafted to detect precise features like edges and corners. This meticulous process involves systematic traversal of the image, executing element-wise multiplications, and aggregating the results. The outcome is a series of feature maps that illuminate distinct aspects of the input image.

5.2. Activation Function: An activation function such as ReLU (Rectified-Linear-Activation) is element-wise applied to the feature maps after the convolutional procedure. This infusion of nonlinearity into the network enables it to acquire knowledge of intricate patterns.

5.3. Pooling Layer: Pooling layers, which can be either max pooling or average pooling, serve the purpose of spatial dimension down-sampling in the feature maps while preserving critical information. Pooling contributes to the reduction of computational complexity and enhances the network's resilience to input variations.

5.4. Additional Convolutional Layer: The additional convolutional layer with progressively more intricate configurations layered together to acquire more advanced features from the input data. This layer captures more abstract features by building upon the features learned in previous layers.

5.5. Flattening: A sequence of convolutional and pooling layers transforms the feature maps into a one-dimensional vector. This stage prepares the data for the fully linked layers.

5.6. Fully Connected Layers: The flattened vector is forwarded through one or more fully connected layers, responsible for classification by acquiring intricate connections between the extracted features and the output categories. Typically, the final fully connected layer's output is input into a SoftMax activation function to generate probabilities for each class.

5.7. Output Layer: Within the scope of drowsiness detection, the output layer typically comprises two units: one for indicating the driver is alert and another for indicating the driver is drowsy. The SoftMax activation function converts the network's predictions into probability scores for each class.

6. Faster-Region-CNN Model

While FR-CNN customized for drowsiness detection, the pivotal phases of obtaining a properly labeled dataset and training the model appropriately are fundamental for attaining precise outcomes. To employ Faster R-CNN model [15,16] in drowsiness detection carried out the work portfolio as given below:

6.1. Data Preparation: Collected a dataset of images of faces with annotations indicating whether the driver is alert or drowsy. These annotations could be bounding boxes around the eyes or any other relevant facial regions.

6.2. Convolutional Backbone: The foundation of FR-CNN is our prepared convolutional neural network (CNN). From the input images, feature maps will be obtained through this network.

6.3. Region Proposal Network (RPN): In the context of drowsiness detection, faster R-CNN contains an RPN meant to indicate probable regions [14,15,16] inside the image that could encompass areas of interest, particularly the eyes and their surrounding areas. The RPN generates bounding box proposals and associated abjectness scores based on the extracted feature maps. These proposals are created by utilizing anchor boxes, which encompass predefined aspect ratios and scales and are slid across the feature maps.

6.3. Region-of-Interest (RoI) Alignment: Fixed-size feature maps are extracted for every suggested region by the ROI Align layer. This ensures that, regardless of the beginning sizes of the areas, the input sizes to succeeding layers are constant.

6.4. RoI Classifier and Regressor: Different branches lead the feature maps corresponding to Regions of Interest (RoI) in both the bounding box regression and classification processes. Whether the suggested areas have alert or sleepy face traits is predicted by the categorization branch. Regression branch: works on improving the suggested areas' bounding box coordinates.

6.5. Inference: When making an inference, the network takes a picture as input and performs the following actions: Extract feature maps and generate region proposals using the RPN using the CNN backbone. Use the ROI Align layer to extract uniformly sized feature maps for each proposal [12, 15, 16]. Forward the ROI feature maps through the classifier and regressor branches to anticipate bounding box coordinates and class probabilities.

6.6. Decision Threshold: A judgment threshold on the anticipated class probabilities can be established to assess whether or not an individual is sleepy. The individual may be regarded as sleepy if the chance of falling into the "drowsy" class is greater than the threshold for a certain area. Figure 4. shows the completed algorithmic model structure that is used to achieve sleepiness detection: the FR-CNN architecture. The information is given below:

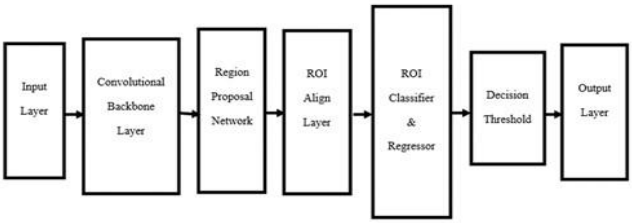

Fig 4. Faster Region-CNN Model architecture of ASHS

Each layer functioned as follows, and figure 5. shows the algorithmic procedure utilizing the FR-CNN model of ASHS.

```
Faster-Region-CNN Model Build
Begin
    while X belongs to seeds,
    do the following
        C ( X.data ) = ConExt ( X.data, thred, M )
        con = random.choice   ( Call − C ( X.data ) )
        if ( X.data is an image)   then loss = con
        perturbation = gradients ( loss, X.data )
        X' =X.data.copy( )
    for i from 1 to num of iterations do
         X.data+ = perturbation · i
         if ( M(X').result == X.label and M( X.data ) ! = X.label )
                  then add X.data to testInputs
         else for word in X do
                  out=EDA(X)
                  inputs+ = out
                  rate = Coverage Rate (inputs)
    if ( rate target Rate ) then add inputs to test Inputs
    return test Inputs
End
```

Fig 5. FR-CNN Model Build Pseudocode

7. RNN Model

Finally, in situations where the data exhibit a sequential or time-dependent connection, sleepiness detection is achieved through the use of Recurrent Neural Networks (RNNs) [10]. Data Preparation: Collected a data related sequences of facial landmarks as signals extracted from video.

7.1. Sequence Representation: Convert the sequences into a suitable format for RNNs. It is typical to represent each signal throughout a time window when working with physiological signals as a series of data points. That is for facial landmarks, can represent each frame's landmarks as a sequence.

7.2. RNN Architecture: At the heart of an RNN lies its recurrent layer, responsible for handling sequences of data while preserving a hidden state that encapsulates information from prior time steps. RNN cell variant called the Long Short-Term Memory (LSTM) cell engineered to address challenges related to the vanishing gradient problem.

7.2. Training: Throughout the training process, furnish the RNN with data sequences and their associated labels (either indicating drowsiness or alertness). The RNN systematically processes these sequences, updating its hidden state at each time step. Subsequently, the output generated at the final time step can be transmitted through a fully connected layer to formulate a prediction.

7.3. Loss Calculation: The forecast derived from the final time step is matched against the actual label using an appropriate loss function, such as binary cross-entropy for tasks involving binary classification. The loss is backpropagated through time to update the weights of the RNN.

7.4. Inference: The trained RNN is given a new data sequence during the inference phase, and it examines this sequence to make a prediction about the level of sleepiness. Depending on the particular requirements of the issue at hand, the forecast may be composed using data from several time steps or obtained from the result of the last time step. It's crucial to recognize that while RNNs are proficient at capturing sequential dependencies [12, 14] within data, they can encounter challenges in managing extended-range dependencies and may be susceptible to issues like the vanishing gradient problem. Whose architecture is as shown in Fig 6. and its model build are given below:

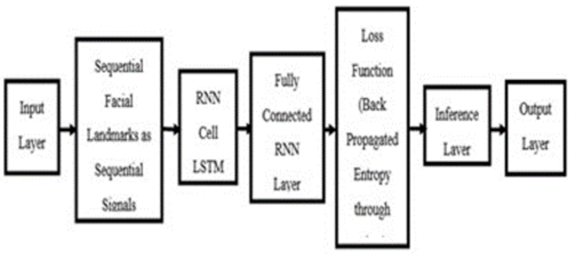

Fig 6. RNN Model architecture of ASH

Each layer operated in the manner described below, and Fig 7. illustrates the algorithmic process using the ASHS RNN model.

RNN Model Build

```
Begin
    Initialize RNN at random
    set NXENT NXE + R and for s=T, with each updation do
        if ( s == T ) then train RNN for N*XENT epochs using XENT only
        else train RNN for NXE + R epochs
    end for
End
```

Fig7. RNN Model Build Pseudocode

8. Experimental Setup

The user's level of fatigue is then assessed by a machine learning technique that makes use of calculated variables from EAR, MOR, and NLR [6,7,12,15,17]. The user is notified by an alarm when the system detects sleepiness. Once the face has been identified, the next stage is to identify individual facial characteristics, such the corners of the lips, the tip of the nose, and the borders of the eyes. This system serves the purpose of monitoring a user's fatigue level and has the capacity to detect the onset of sleep with a safety buffer and the architecture of the system is shown in Fig 8.

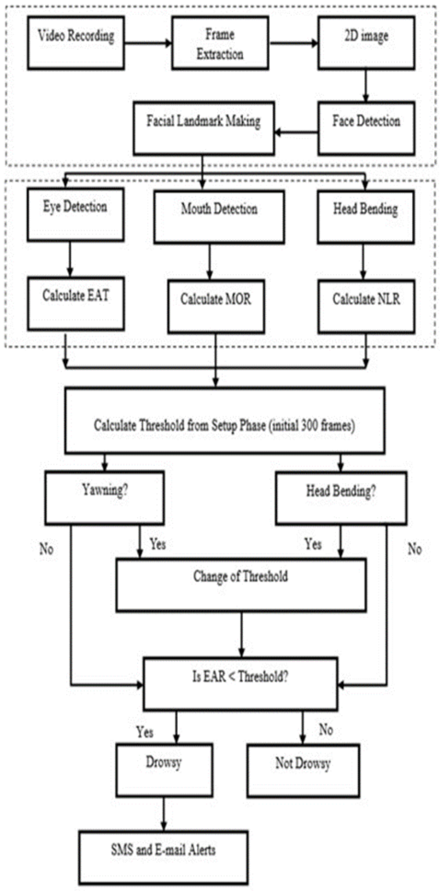

Fig 8. Proposed System Architecture of ASHS

However, before proceeding with feature detection, it is crucial to normalize the face images. The goal is to reduce the influence of variables such as varying camera distances, uneven lighting conditions, and diverse image resolutions.

This normalization is accomplished by converting the face picture to grayscale and scaling it to a fixed width of 500 pixels. A series of regression trees that anticipate the positions of facial landmarks using a predefined subset of pixel intensities come after this preprocessing. The sum of square error loss is decreased using the gradient boosting learning approach [8].

To differentiate between different types of face structures, different priors are applied. Using this procedure, the nose's center axis, the mouth, and the eyes' outlines are created in important spots on the face. These designated facial landmarks, identifiable by their distinct red markers, play a pivotal role as crucial reference points for subsequent stages [18-21] of data processing and analysis. They serve as anchor points for further computations and analyses in the system's workflow. Combination of facial landmarking, fatigue detection, and alert systems to enhance sleep monitoring and contribute to societal well-being [9,10,13], which effectively identifies and addresses fatigue-related risks, promoting safer and healthier sleep habits.

The alert unit is structured around the format (r, t), where r represents the resultant value (with $r \geq 0$ and $t \geq 0$, where t signifies time). The entire model's functionality hinges on the fatigue detection unit's output. This unit, powered by our designed models classifies images as either indicative of fatigue (+1) or not fatigue (-1). The classification output plays a critical role within the alert unit, as it becomes an input into a running summation process. This procedure retains a minimum value of zero while aggregating successive output values. Two threshold levels are utilized by the alert device. Situations with minimal or no weariness are identified by the first threshold. The second threshold distinguishes between low and high fatigue levels. When the cumulative value exceeds a predefined threshold, it indicates fatigue detection [8,12,16].

9. Experimental Results and Discussion

The challenges like position, orientation, lighting conditions are addressed with fixed location of the camera and its configuration towards the face detection of the person in calculating EAR, MOR, and NLR. Eye closure and frequent mouth opening action of the person of interest are checked with 50% of the threshold value, to verify whether drowsy or not. The angular posture of the head is compared with a degree of vertical line making ninety degree or perpendicular to a horizontal line, to compute HBA and then the NLR. Each parameter of the findings of the person from a video is treated as input to the ASHS. In image segmentation tasks, U-Net is utilized with CNN model. Two primary components make up the code: the encoder and the decoder. Taking hierarchical characteristics from the input image is the encoder's responsibility.

Following batch normalization, it consists of a sequence of convolutional layers. The quantity of filters and the spatial resolution of the feature maps decrease with each convolutional layer. An input layer receives the first input (either a picture or data) and begins the network. There are eight convolutional layers (`Conv2d`) applied in total, conv1 through conv8. Every convolutional layer modifies the feature maps from the preceding layer using a different set of filters. Following each convolutional layer, batch normalization is implemented ((`BatchNormLayer`) bn2 to bn8). Training may be stabilized and convergence can be enhanced via batch normalization. These layers employ a ReLU (`tf.nn.relu`) ith a slope of 0.2 as the activation function. To regulate the spatial dimensions of the feature maps, stride values and padding mode (`pad`) are assigned to each convolutional layer. The task of up sampling the feature maps and progressively restoring the spatial resolution falls to the decoder. Through concatenation levels, information from the encoder is included. The following procedures make up each up-sampling block:

Step-1: A series of deconvolutional layers (`DeConv2d`) are applied to up sample the feature maps. These layers (up7 to up0) increase the spatial resolution and decrease the number of filters. The `out_size` argument specifies the desired output size after up sampling.
Step-2: After each deconvolutional layer, batch normalization (`BatchNormLayer` dbn7 to dbn0) is applied.
Step-3: A concatenation layer (`ConcatLayer` concat6 to concat0) combines the up matched feature maps from the encoder with the sampled feature maps. This aids the network in regaining spatial data that was lost due to downsampling.

The final layer (`out`) is a convolutional layer with sigmoid activation (`tf.nn.sigmoid`). It produces the network's output, which is typically a segmentation map or image. Throughout the code, there are print statements that display information about the size of the feature maps and the final output. These print statements can be useful for debugging and monitoring the network's progress during training. Holistically, this model processes input images through a sequence of convolutional and deconvolutional layers, incorporating batch normalization, to meticulously produce a segmentation map as the ultimate output. The network adeptly acquires the capability to discern and segment objects or regions of interest within the input image, as illustrated in Fig 9(a). and Fig 9(b). across various age groups, particularly those without spectacles.

Fig 9(a). EAR, MOR and NLR determination of a person above 18 years for the Detection of Eyes Closure, Yawning and Head Bending with CNN without spectacles

Fig 9(b). EAR, MOR and NLR determination of a person below 18 years for the Detection of Eyes Closure, Yawning and Head Bending with CNN without spectacles

In Fig 9(a). and Fig 9(b)., the green reference lines within the frame serve to delineate landmarks on the eyes and mouth of the person of interest. The horizontal and vertical green lines, accompanied by a center rectangular marking, play a crucial role in facilitating calculations for EAR, MOR, HBA, and NLR. In parallel, the FR-CNN adheres to the procedural steps outlined in the CNN model, utilizing relevant libraries from the Python Imaging Library (PIL) for image processing. This process specifically focuses on drowsiness detection, centering around regions of interest such as the eye and mouth regions, utilizing face samples captured. Notably, this model is trained with a Classifier, representing an instance of a neural network model.

The `fit_generator` method is used for training the model and is typically employed where the data is loaded and preprocessed on-the-fly in batches. `training_set` and `test_set` are data generators or iterators. These objects provide batches of training and validation data for the model. The `epochs=100` parameter specifies to run for 100 iterations over the entire training dataset. `steps_per_epoch` and `validation_steps` parameters are set to `len(training_set)` and `len(test_set)`, respectively. These parameters indicate the number of batches to be drawn from the training and validation data generators in each epoch. They are essential for controlling the training process and ensuring that each batch is used exactly once in each epoch. Training process that consists of multiple epochs. In each epoch, the model is exposed to batches of training data (from `training_set`) and evaluated on validation data (from `test_set`). In order to minimize a predetermined loss function, the model's internal parameters—weights and biases—are modified throughout training. This process involves forward and backward passes through the neural network. The results of the training process are stored in the variable `r`, such as training and validation loss values, and possibly accuracy metrics.

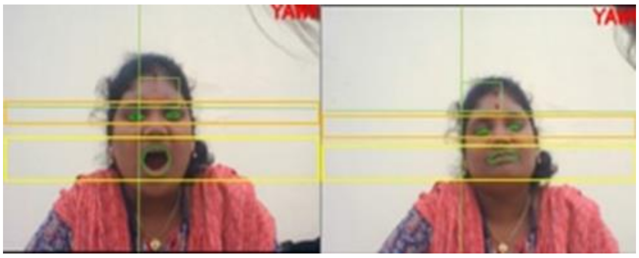

Fig 10(a). EAR, MOR and NLR determination of a person for the Detection of Eyes Closures, Yawning and Head Bending with FR-CNN model without spectacles

Fig 10(b). EAR, MOR and NLR determination of a person for the Detection of Eyes Closures, Yawning and Head Bending with FR-CNN model with spectacles

In Fig 10(a). and Fig 10(b)., the delineated orange and yellow line windows demarcate the Regions of Interest (ROIs) for eyes and mouth, crucial contributors to the substantial improvement in drowsiness detection within our selected models. These regions distinctly set our models apart from others. Furthermore, the conventional green line facial landmarks, consistent with the CNN model, are prominently featured in these figures.

The architecture of the RNN model incorporates an embedding layer designed to convert tokens into dense vectors, crucial for effective representation in a continuous vector space. Specifically configured with an output dimension of 100, each image is expressed as a 100-dimensional vector. This layer allows for initialization with pre-trained image embeddings. In order to successfully exclude padding values during training, the Masking Layer handles variable-length input sequences.

The Recurrent Layer makes use of Long Short-Term Memory (LSTM), a skilled variety of recurrent neural network layer that can effectively recognize complex sequential patterns in data. The LSTM layer, configured with 64 units in this version, makes it easier to capture complicated relationships in the input sequences. The `return_sequences` parameter set to `False` means that it only returns the final output of the sequence, suitable for various

sequence-to-sequence tasks. The LSTM layer's output gains non-linearity from the Fully Connected Layer, which has 64 units and ReLU activation. In the data, it aids the model in discovering intricate linkages. During training, a random subset of input units is set to zero using the Dropout regularization approach in the Dropout Layer. Adding noise to the model and minimizing dependence on any one unit also helps prevent overfitting. The output layer is the final dense layer with a softmax activation function produces a probability distribution over the input image.

Drowsiness detection with RNN model output is shown in the Fig 11(a). and Fig 11(b). for casual green line facial landmarkings with spectacle and without spectacle.

Fig 11(a). EAR, MOR and NLR determination of a person for the Detection of Eyes Closures, Yawning and Head Bending with RNN model with spectacles

Fig 11(b). EAR, MOR and NLR determination of a person for the Detection of Eyes Closures, Yawning and Head Bending with RNN model with beard

CNNs excel in precise facial recognition, while FR-CNNs and RNNs serve for comparative analyses within the ASHS framework. Our models effectively recognize diverse facial features, encompassing individuals with or without spectacles, varying age groups (including those below 18 years), different genders, and individuals with or without a beard. This collective integration of neural networks empowers ASHS for real-time drowsiness detection, considering parameters such as eyes, mouth, head bending, age groups, gender, and facial hair presence. The comparative results of these models are presented in table 1.

Table 1. The study contrasts values across ASHS framework.

Models	**EAR**	**MOR**	**NLR**	**Eye ROI**	**Mouth ROI**	**Training Time (ms)**	**Inference Time (ms)**	**Performance Accuracy (%)**
CNN	Yes	Yes	Yes	No	No	6	20	95.2
RNN	Yes	Yes	Yes	No	No	8	25	92.5
FR-CNN	Yes	Yes	Yes	Yes	Yes	12	30	97.8

Highlighting our commitment to precision, the CNN and RNN models exhibited accuracy percentages of 95.2% and 92.5%, respectively, in adeptly discerning sleepiness. Through meticulous optimization, we further refined the accuracy of drowsiness detection to an elevated 97.8%, surpassing the CNN by 2.6% and the RNN by 5.3%. This significant advancement was achieved by focusing input data primarily on the region of interest, specifically the mouth ROI and eye ROI, especially during instances of head bending.

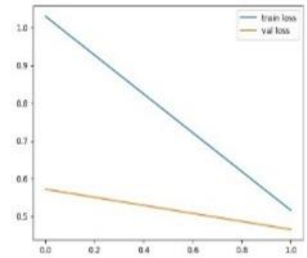

Fig 12(a). Graph showing output of Loss function of the models

Fig 12 (b). Graph showing the Training time of the models

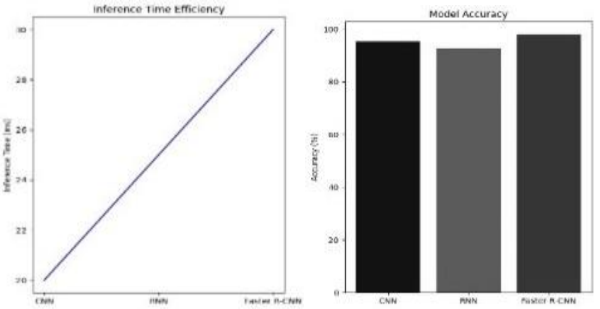

Fig 13(a). Graph showing the Inference time of the models

Fig 13(b). Graph showing the Accuracy of detection of drowsiness for real time data in terms of percentage with respect to the ASHS

The ASHS instantly notifies registered users of varying ages and genders of tiredness when drowsiness detection is confirmed. In our studies, we found variances in face characteristics, which the algorithm takes into account. In case of emergency, this notification is distributed by WhatsApp and email. By using an SMTP server with the Postman free software for email alerts, the ASHS guarantees quick communication. The user's position, which was acquired using the Google API, is included in a pre-written WhatsApp message that is sent out concurrently by the system using the user's contact details. In order to improve the warning system's overall effectiveness, the implementation smoothly incorporates strong neural network models for precise facial identification and landmark mapping.

10. Conclusion

Designed to meet the needs of a wide range of users, our suggested system offers a comprehensive set of benefits in the pursuit of improving safety via advanced sleepiness detection. This creative solution goes beyond traditional safety procedures by utilizing a sophisticated methodology based on real-time video analysis. Our technology provides increased safety for drivers as well as a wider range of age groups due to its thoughtful placement of the webcam to capture the frontal face and careful attention to the lips and eyes. This all-inclusive method for detecting sleepiness provides a proactive response to possible hazards, promoting health and enhancing performance in a variety of settings. When the system is used for driving, at work, or during everyday activities, it creates a universal safety net since it is good at quickly detecting indicators of tiredness.

A proactive and preventative attitude to lessening the effects of exhaustion is fostered by such inclusion, which enhances overall quality of life. In conclusion, a strong option for accurate sleepiness detection is the AI-based drowsiness detection system, or ASHS, which integrates sophisticated neural network models including CNNs, FR-CNNs, and RNNs. Increased user safety is ensured by the addition of a graded alert system. In addition to its technological capabilities, the system also systematically tackles a wide range of societal concerns, including driving, geriatric care, healthcare, education, and medical treatment. This thorough approach highlights the system's effectiveness in specific situations while also positioning it as a crucial factor in promoting social safety and overall well-being.

References

[1] Bharath Bharadwaj B S, Shashank B, Gayathri S, N S Dhanush, Darshan H (2022). “Real Time Drowsiness Detection”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, Volume: 09 Issue: 07, July 2022 www.irjet.net p-ISSN: 2395-0072.

[2] Stephen Danny Leo Xavier, K R Sumana, Dr. H. D. Phaneendra."Vehicular Safety Model: A Phase-Wise Vehicular Catastrophe Prevention Model", Volume 10, Issue VII, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: 3353-3358, ISSN : 2321-9653, https://doi.org/10.22214/ijraset.2022.45734.

[3] Khan Furqan, Khan Arif, Shaikh Rohaan, Dr. Ashfak Shaikh. “Driver Fatigue Detection System using Face Detection”. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, Volume: 08 Issue: 05, May 2021 www.irjet.net p-ISSN: 2395-0072.

[4] Viren Patel, Farman Khan, Aatif Qureshi, Sonali Suryawanshi. “Driver Drowsi-ness Detection System”. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, Volume: 08 Issue: 05, May 2021 www.irjet.net p-ISSN: 2395-0072.

[5] Arun Prakash, B Poojitha Reddy, Vishnu Dinesh, Amal Dasan P, Prof. Mohammed Za-beeulla (2022). “Human Driver’s Drowsiness Detection System”. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, Volume: 09 Issue: 06, Jun 2022 www.irjet.net p-ISSN: 2395-0072.

[6] Sri Mounika, T.; Phanindra, P.; Sai Charan, N.; Kranthi Kumar Reddy, Y.; Govindu, S. “Driver Drowsiness Detection Using Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and Driver Distraction Using Head Pose Estimation”. In ICT Systems and Sustain-ability; Springer: Berlin/Heidelberg, Germany, (2022); pp. 619–627.

[7] Dua, M.; Singla, R.; Raj, S.; Jangra, A.; Shakshi. “Deep CNN models-based ensemble approach to driver drowsiness detection”. Neural Comput. Appl. 2021, 33, 3155–3168.

[8] Kumar, A.; Patra, R. “Driver drowsiness monitoring

system using visual behaviour and machine learning". In Proceedings of the 2018 IEEE Symposium on Computer Applica-tions & Industrial Electronics (ISCAIE), Penang, Malaysia, 28–29 April 2018; pp. 339–344.

[9] Fatima, B.; Shahid, A.R.; Ziauddin, S.; Safi, A.A.; Ramzan, H. "Driver fatigue detection using viola jones and principal component analysis". Appl. Artif. Intell. (2020), 34, 456–483.

[10] Ed-doughmi, Y.; Idrissi, N.; Hbali, Y. "Real-Time System for Driver Fatigue Detection Based on a Recurrent Neuronal Network" J. Imaging (2020), 6, 8.

[11] Sheikh, A.A.; Mir, J. "Machine Learning Inspired Vision-based Drowsiness Detection using Eye and Body Motion Features". In Proceedings of the 2021 13th International Conference on Information & Communication Technology and System (ICTS), Sura-baya, Indonesia, 20–21 October 2021; pp. 146–150.

[12] Albadawi, Y.; AlRedhaei, A.; Takruri, M. "Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features". J. Imaging 2023, 9, 91. https://doi.org/10.3390/jimaging9050091.

[13] R. Ahmed, Kazi Emrul Kayes Emon and M. F. Hossain, "Robust driver fatigue recognition using image processing," 2014 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, Bangladesh, 2014, pp. 1-6, doi: 10.1109/ICIEV.2014.6850713.

[14] Chirra, V.R.R.; Uyyala, S.R.; Kolli, V.K.K. "Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State". Rev. D'Intell. Artif. 2019, 33, 461–466. [Google Scholar] [CrossRef].

[15] Girshick, Ross. "Fast r-cnn." Proceedings of the IEEE international conference on computer vision. 2015.

[16] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.

[17] Robert Gabriel Lupu, Florina Ungureanu, Valentin Siriteanu, "Eye Tracking Mouse for Human Computer Interaction", The 4th IEEE International Conference on e-Health and Bioengineering - EHB 2013.

[18] Amin Azizi Suhaiman; Zazilah May; Noor A'in A. Rahman. (2020). "Development of an intelligent drowsiness detection system for drivers using image processing technique". doi:10.1109/SCOReD50371.2020.9250948.

[19] Igor Lashkov; Alexey Kashevnik; Nikolay Shilov; Vladimir Parfenov; Anton Shabaev. (2019). "Driver Dangerous State Detection Based on OpenCV & Dlib Libraries Using Mobile Video Processing" Published in: 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Date of Conference: 01-03 August 2019, Date Added to IEEE Xplore: 05 December 2019, DOI: 10.1109/CSE/EUC.2019.00024, Publisher: IEEE, Conference Location: New York, NY, USA.

[20] Maliha Khan; Sudeshna Chakraborty; Rani Astya; Shaveta Khepra. "Face Detection and Recognition Using OpenCV", doi:10.1109/ICCCIS48478.2019.8974493. 2019.

[21] K. Satish; A. Lalitesh; K. Bhargavi; M. Sishir Prem; T Anjali. "Driver Drowsiness Detection". doi:10.1109/ICCSP48568.2020.9182237.