# Yawn Detection for Driver's Drowsiness Prediction using Bi-Directional LSTM with CNN Features

Sumeet Saurav[1,3], Shubhad Mathur[2], Ishan Sang[2], Shyam Sunder Prasad[1,3], Sanjay Singh[3]

[1]Academy of Scientific & Innovative Research (AcSIR), Ghaziabad, India
[2]Birla-Institute of Technology and Science (BITS, Pilani), Goa Campus, Goa, India
[3]CSIR - Central Electronics Engineering Research Institute (CSIR-CEERI), Pilani, India

sumeet@ceeri.res.in

**Abstract.** Drowsiness of drivers is a critical problem and has recently attracted a lot of attention from both academia and industry. A real-time driver's drowsiness detection system is often considered as a crucial component of the Advanced Driver Assistance System (ADAS). Although, there are a number of physical parameters associated with drowsiness like blink frequency, eye closure duration, pose, gaze, etc., yawing can also be used as an indicator of drowsiness. This work presents a novel deep learning-based framework for driver's drowsiness prediction based on yawn detection in a video stream. The proposed approach uses a combination of a convolutional neural network (CNN), 1D-CNN, and bi-directional LSTM (Bi-LSTM) network. In the first step, the pipeline extracts the mouth region from each frame of the video using a combination of face and landmark detector. In the subsequent step, spatial information from the mouth region is extracted using a pre-trained deep convolutional neural network (DCNN). Finally, sequential information which models the evaluation of yawn using the extracted mouth feature is learned using a blend of 1D-CNN and bi-directional LSTM (Bi-LSTM) network. Experiments were performed on manually extracted and annotated video clips obtained from two publically available drowsiness detection dataset namely YAWDD and NTHU-DD. Experimental results show the effectiveness of the proposed approach both in terms of recognition accuracy and computational efficiency. Thus, the proposed pipeline is a good candidate for real-time implementation on an embedded device.

**Keywords:** Drowsiness detection; convolutional neural networks; long short-term memory (LSTM); bi-directional LSTM (Bi-LSTM).

## 1 Introduction

According to a recent article [1], in India more than 150, 000 people are killed every year due to road accidents which is much higher than the developed countries like US, where the count was 40,000 in 2016. Another report of the American National Highway Traffic Safety Administration (NHTSA) [2], states that there were 803 causalities which were reported in 2016 due to drowsy driving. Therefore, drowsiness of drivers is considered as a critical problem and has attracted a lot of attention from both academia and industry. At present, many renowned multinational automobile companies like Nissan, Toyota, and Volkswagen are working in this area to create technologies which could mitigate the driver's drowsiness related issues by issuing a warning signal to the drivers.

There are a number of physiological parameters associated with drowsiness like eye closure duration, blink frequency, percent of times the eyes are close (PERCLOS), pose, gaze and nodding frequency which could be extracted using a visual sensor (Camera) [3]. Apart from these, another parameter which could also be a good indicator of driver's drowsiness is yawn. Yawning can be best defined as an involuntary act of gaping of mouth and deep inhalation followed by shallow exhalation. It is an important indicator to detect drowsiness and fatigue at an early stage.

During yawing, mouth opens wide and a change is observed in geometric features of mouth. Many approaches which use geometrical features extracted from mouth for yawn detection could be found in [4]-[6]. Various color-based approaches have also been proposed for mouth and lips detection [7]. In [8], the authors have used Gravity-Center template for face detection followed by mouth corner extraction using grey projection and Gabor wavelets. The authors in [9] segmented the image texture regions using stochastic region merging strategy, then skin color and texture models is used for classification. The work reported in [10] used two cameras to their advantage: a low-resolution camera for the face and a high-resolution one for the mouth. Haar-like features were then used to detect yawning based on the ratio of the height to width of mouth. Work presented in [11] used the static supervised classification based on log-polar signatures approach for open or closed mouth detection. Another work proposed in [12], used a combination of SVM based face detector, gradient edge detectors-based mouth region locator and circular Hough transform based yawn detector. Authors in [13] detected rate and amount of changes in mouth using back projection theory through a modified implementation of the Viola-Jones algorithm for face and mouth detection. Another method presented in [14] proposed to detect yawning using geometric and appearance features of mouth and eye regions and were also able to detect hand-covered and uncovered yawns. A combination of CNN and LSTM for yawn detection could be found in [15].

Our proposed method for yawn detection in some extent is similar to the work presented in [15]. However, there are a number of add-ons. First, in [15] the authors have used pre-trained CNN model trained on ImageNet, but in this work we pre-trained a deep convolutional neural network (DCNN) from scratch using FER+ database and used it for extracting spatial information from the mouth region. Secondly, the work in [15] have used stacked LSTM network for modelling temporal information related with yawn, but this work uses a combination of 1D CNN and Bi-LSTM for the same. In our method, a camera installed on the dashboard/side mirror of the car continuously captures video streams. From these video streams, face is first detected which is then passed to a landmark detector. Using coordinates of the facial landmarks the mouth region is extracted. From the mouth region, deep features are extracted using a pre-trained CNN. Finally, features extracted from 32 frames are fed to a combination of 1D CNN and Bi-LSTM to detect a yawn.

The remainder of this paper is divided as follows: Section II presents an overview of our proposed method. Experiments results and discussions are dealt in section III. We finally conclude in section IV.

## 2 Proposed Method

The proposed framework used for detection of yawn in a video stream is shown in Fig. 1.

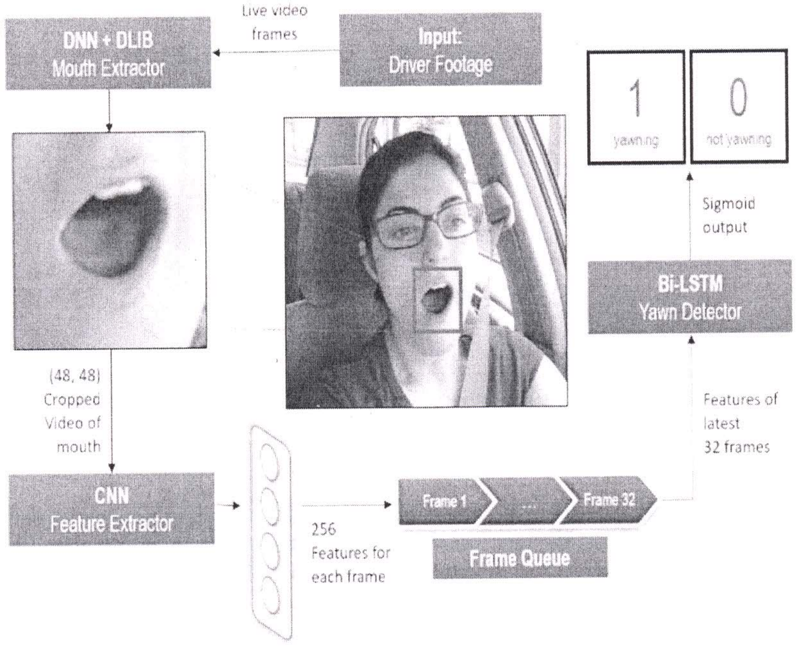


**Fig. 1.** Proposed framework used for yawn detection

As could be seen from the above figure, the framework consists of three main steps: mouth region extractor, deep feature extractor, and yawn detector. Each of these steps are briefly discussed below.

### 2.1 Mouth region extractor

In order to extract the mouth region from a given video stream, the mouth region extractor needs to detect the face region and then locate and track facial landmarks on the detected face. We used OpenCV's DNN model (ResNet-10 architecture) to detect the face region and Dlib's shape predictor [16] (68-face-landmarks) to get the facial landmarks. The points corresponding to the nose (33), lip corners (48 and 54) and near the mouth (48 to 67) (as shown in Fig 2.) are then used to find the mouth region. The model finds the center of the mouth region using the average of all the points near the mouth. Then using the points 33, 48 and 54 it estimates the size of the mouth region and creates a square bounding box around it. This box is then resized to size (48, 48) and passed on to the feature extractor.

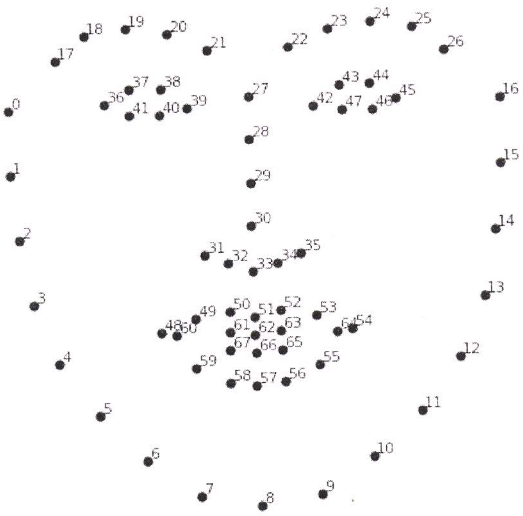


**Fig. 2.** Set of 68-points landmark locations available in [16]

### 2.2 Feature Extractor

To extract spatial information from the extracted mouth region a deep convolutional neural network (DCNN) has been used in this work. The proposed DCNN was trained from scratch using the FER+ database [17]. Details of the DCNN architecture has been shown in Fig. 3.

The network takes as input a 48 x 48 pixels grayscale image (one single frame of the mouth video). There are four convolutional layers, three max pooling layers, two dense layers, and finally a SoftMax classifier layer. Once the DCNN is trained, it was used for extracting features from the mouth images. The dimension of the features extracted using the network is 256.

### 2.3 Yawn Detector

A typical yawn is a dynamic and continuous process that lasts for around 2-3 seconds and has 3 phases: the person opening her/his mouth, the prolonged open state of the mouth and, then its closing. To find a relationship between the features of consecutive frames and determine if the person is yawning, a recurrent neural network (RNN) is used. RNNs use the features of the frame (it is currently processing) and also, the output for the previous frame to generate prediction for each frame. However, a vanilla RNN suffers from vanishing and exploding gradients problems and also struggles to learn long term dependencies. To counter these, special type of RNNs, Long Short-Term Memory Networks (LSTMs) are used instead [18]. LSTMs use memory cells to store information and "input" and "forget" gates to control its flow, which enables them to discover long range temporal relationships.

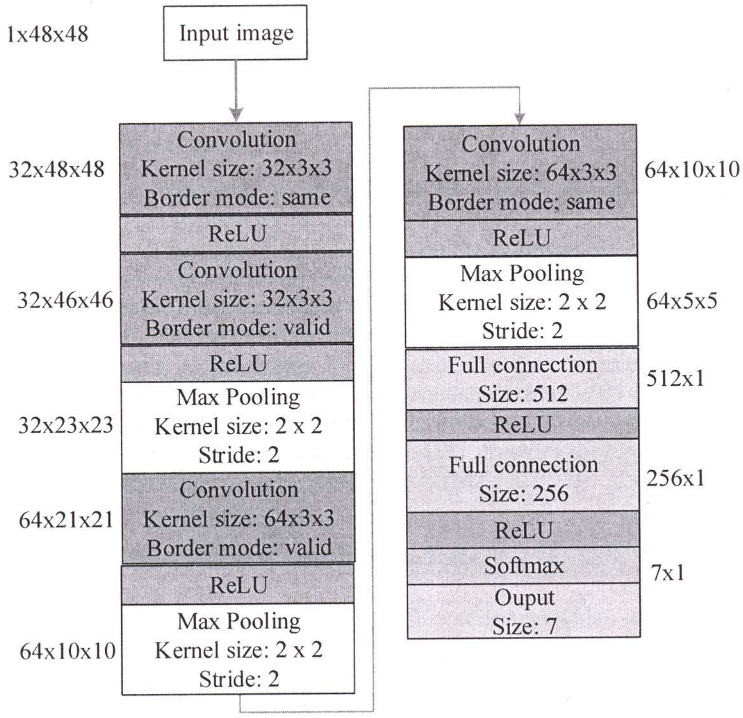
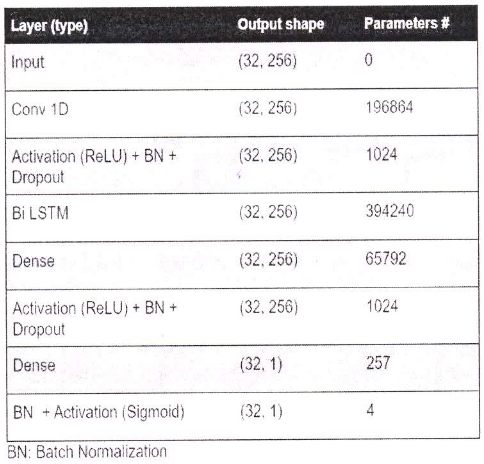


**Fig. 3.** Deep CNN architecture used for feature extraction

In our system, we used a combination of 1D-CNN and Bi-directional LSTM as shown in Fig. 4. The principle of a Bi-directional RNN is to split the neurons of a regular RNN into two parts: one for forward pass in the positive direction and another for the negative direction. This enables the neuron, processing the current frame, to gather information from both the past and the future frames, which leads to more robust learning.



| Layer (type) | Output shape | Parameters # |
|---|---|---|
| Input | (32, 256) | 0 |
| Conv 1D | (32, 256) | 196864 |
| Activation (ReLU) + BN + Dropout | (32, 256) | 1024 |
| Bi LSTM | (32, 256) | 394240 |
| Dense | (32, 256) | 65792 |
| Activation (ReLU) + BN + Dropout | (32, 256) | 1024 |
| Dense | (32, 1) | 257 |
| BN + Activation (Sigmoid) | (32, 1) | 4 |

BN: Batch Normalization

**Fig. 4.** Architecture used for modelling temporal information

1D-CNN layer (1*256, same padding) convolves on the 32 frames. The yawn detector takes as input 32 frames at a time and performs forward and backward propagation on them. For inference, the model maintains a queue of the latest 32 frames of the video feed and passes them into the yawn detector. It then takes the prediction for the latest frame (sigmoid output) as the output for the latest frame of the video. We kept a threshold value (A) for the sigmoid output and another threshold (B) for the number of continuous frames in which 'YAWNING' was predicted. If the sigmoid output is beyond A and the number of continuous 'YAWNING' frames is beyond B, the frame is classified as 'YAWNING', else as 'NOT YAWNING'.

## 3 Experimental Results and Discussions

This section gives details of various datasets and experiments which were performed in this study.

### 3.1 Dataset Preparation

Three different datasets namely FER+, YawDD, and NTHU-DDD were used in this study. The FER+ [17] dataset is the re-tagged version of the FER2013 database consisting of facial images of different expressions (happy, neutral, sad, surprise, disgust, contempt, anger, and fear). The dataset comprises of 35,541 facial images of resolution 48 x 48 pixels divided into training, validation, and test set. We used this database to pre-train the DCNN wherein only facial images from 7 expressions (anger, disgust, happy, fear, neutral, sad, and surprise) were considered and the training samples were created by merging the actual training and test set provided in the dataset. The validation set provided in the dataset was used as validation set during training. Sample images corresponding to different expressions from the dataset has been shown in Fig. 5.

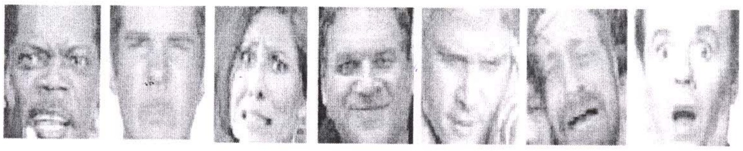


**Fig. 5.** Sample images from FER+ 7 expression dataset (left to right: anger, disgust, fear, happy, neutral, sad, and surprise)

The YawDD dataset [19] is an open-source dataset comprising of videos of drivers with various facial characteristics and targeted towards designing and testing algorithms and models for yawn detection. Two different variants of the database are available, one captured using a camera mounted on the dash board of the car and the other captured using camera mounted on the side mirror of the car. We used the later variant of the database which contains a total of 322 videos of male and female drivers of different ethnicities with/without glasses/sunglasses. The videos have been labelled into three classes: no talking (normal), talking, and normal. The videos have been captured using a RGB camera at 30 frames per second with a resolution of 640 x 480 pixels. Some of the sample frames extracted from the video clips of the database has corresponding to normal, talking, and yawning classes have been shown in Fig. 6, Fig. 7, and Fig. 8 respectively.

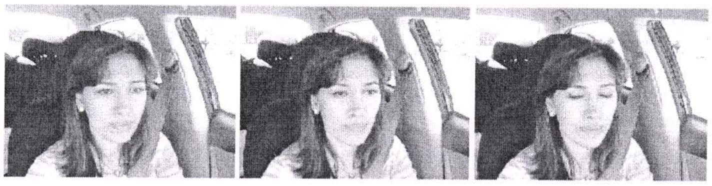


**Fig. 6.** Sample frames from the normal class video clip (YawDD dataset)

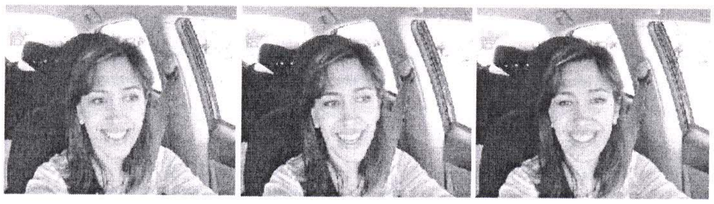


**Fig. 7.** Sample frames from the talking class video clip (YawDD dataset)

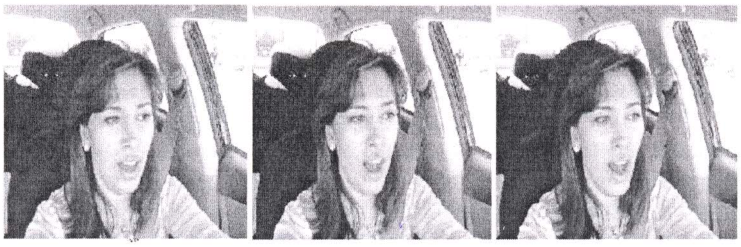


**Fig. 8.** Sample frames from the yawning class video clip (YawDD dataset)

The final database used is the NTHU-DDD database [20]. Videos in the dataset is divided into two splits: training and validation. The training dataset consists of 356 videos containing 18 subjects of different ethnicities recorded with and without wearing glasses/sunglasses under a variety of simulated driving scenarios, including normal driving, yawning, slow blink rate, falling asleep, burst out laughing etc., under day and night illumination conditions. The videos of the dataset were labelled either as sleepy/non-sleepy. However, in our experiments we manually extracted clips from the videos where the subject was either yawning or normal. Sample images extracted from the yawn video clip from the NTHU-DDD database has been shown in Fig. 9.

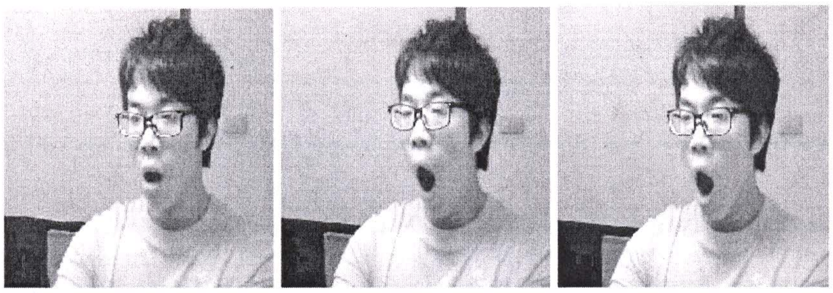


**Fig. 9.** Sample frames from the yawning class video clip (NTHU-DDD dataset)

Unfortunately, both the datasets (YawDD and NTHU-DDD) contained video clips which were labelled as yawning/sleepy, but much of the video content in them covered no yawning at all. Thus, it was necessary to crop out the yawning parts of the videos (a typical yawn lasting 3-5 seconds), otherwise the model would have wrongly learnt to classify people who were acting normal/talking as yawning. We also added the rest of the non-yawning parts of these 'yawning' videos in our 'non yawning' set. After cropping, we removed a few videos in which the participants' yawn appeared to be extremely fake. Finally, the total number of yawning clips obtained was 205 and 526 from YawDD and NTHU-DDD dataset respectively. The number of normal/talking clips obtained from these were 318 from YawDD and 178 from NTHU-DDD datasets respectively.

### 3.2 Model training and results

Training of the models was done on a GPU machine having Ubuntu 16.04 OS, 32 GB RAM, and NVIDIA GeForce 1080 GPU with 8 GB memory. We used Keras [21] with Tensorflow backend for conducting the experiments.

The feature extractor deep convolutional neural network (DCNN) was trained on the popular FER+ dataset. Different on-the-fly data augmentation techniques (horizontal flip, random rotation within the range of 10 degrees, random zooming by x0.1, and normalization) were used to avoid overfitting. These augmentations were performed using the in-built Keras ImageDataGenerator utility. The model was trained using a batch size of 64. The learning was kept fixed to 1e-5 and the Adam optimizer was used with categorical cross-entropy loss function. After training the model for 200 epochs, we got a validation accuracy of around 75%. The learning curve obtained after training the DCNN is shown in Fig. 10.

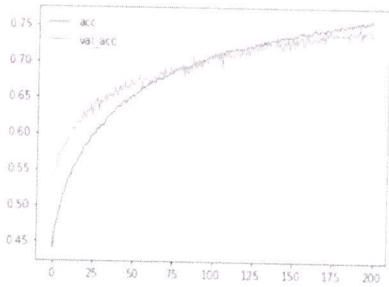


**Fig. 10.** Deep convolutional neural network learning curve (x-axis: epochs, y-axis: accuracy (%)

Once the feature extractor model got trained, we used it to extract features from each frame of the video clips and saved it on the disk. From all the videos in our dataset (YawDD and NTHU-DDD), we got a total of 1114924 non-yawning frames and 21919 yawning frames.

To train the yawn detector model the extracted saved features were loaded from the disk. We then used a sliding window approach on this consecutive array of frames to create numerous windows having frame size of 32. This array of windows was then randomly shuffled, and split into training and validation sets (70:30 ratio). After doing this, we ended up with 400227 windows (samples) for validation and 933860 windows (samples) for training. Each window had 32 frames and each frame had a 256-length feature vector. We trained this model with a learning rate of 1e-5, using Adam optimizer and a binary cross entropy loss function. We took a batch size of 64 and shuffled the dataset before each epoch. Batch Normalization and Dropout (with a rate of 0.4) is applied after every layer. We also used kernel and recurrent (L2) regularizes to prevent the model from overfitting. Our model achieved a final validation accuracy of 96.48% and training accuracy of 95.66% after being trained for 12 epochs as could be seen from the Fig. 11.

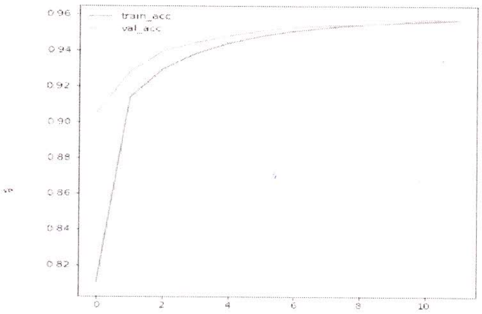
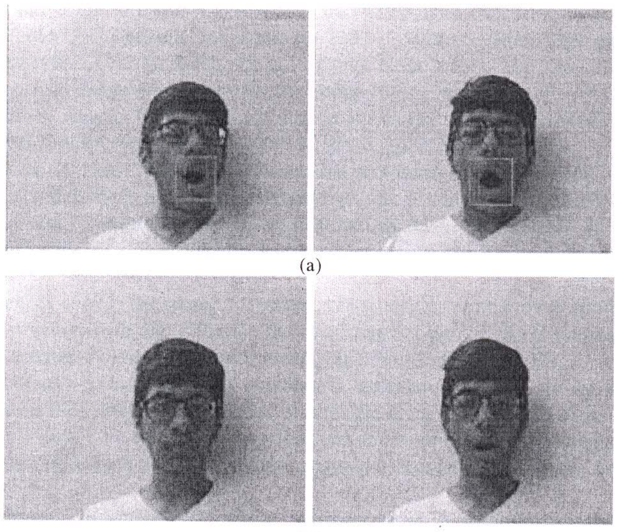


**Fig. 11.** Yawn detector learning curve (x-axis: epochs, y-axis- accuracy (%)

Once trained the model was tested on a laptop having 8 GB RAM, Intel i7 processor core, Ubuntu 18.04.2 LTS OS and Keras 2.2.4 deep learning library with TensorFlow backend. The inference speed achieved by the proposed yawn detector was ~25 FPS (frames per second) achieving real-time performance. Sample detection results obtained using our proposed framework for yawn detection are shown in Fig. 12.



(a)

(b)

**Fig. 12.** Prediction results of the trained model (a) Yawn case (b) Normal case

From the detection results, one can find that the proposed approach is quite robust towards lighting variations and is able to differentiate between different mouth states properly.

## 4 Conclusion

In this work, we presented an efficient deep learning-based approach for yawn detection targeted towards driver's drowsiness prediction. The proposed algorithmic pipeline consists of face detector, a landmark detector, a deep convolutional neural network (DCNN), and a combination of 1D-CNN and bi-directional LSTM (Bi-LSTM). Face and facial landmarks detector have been used for extracting the mouth region from the video frames. The DCNN pre-trained on FER+ database has been employed to extract the spatial information from the extracted mouth regions. Finally, a blend of 1D-CNN and Bi-LSTM has been used to model the temporal information associated with a yawn sequence. Experiments were done on a manually annotated yawing and normal clips cropped from the YawDD and NTHU-DDD datasets. From the experimental results, we found the effectiveness of the proposed approach both in terms of recognition accuracy and computational efficiency. Thus, the proposed pipeline is a good candidate for real-time implementation of a yawn detector on an embedded platform for driver's drowsiness prediction.

## References

1. https://timesofindia.indiatimes.com/india/over-1-51-lakh-died-in-road-accidents-last-year-up-tops-among-states/articleshow/72078508.cms (assessed on 18th Nov, 2019).
2. National Center for Statistics and Analysis. (2017, October). 2016 fatal motor vehicle crashes: Overview. (Traffic Safety Facts Research Note. Report No. DOT HS 812 456). Washington, DC: National Highway Traffic Safety Administration.
3. G. Sikander, S. Anwar, "Driver fatigue detection systems: A review", IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 6, pp. 2339-2352, 2018.
4. Q. Ji, Z. Zhu, P. Lan, "Real-time nonintrusive monitoring and prediction of driver fatigue", IEEE transactions on vehicular technology, vol. 53, no. 4, pp. 1052-1068, 2004.
5. T. Wang, P. Shi, "Yawning detection for determining driver drowsiness", In Proceedings of IEEE International Workshop on VLSI Design and Video Technology, pp. 373-376, May, 2005.
6. W. Rongben, G. Lie, T. Bingliang, J. Lisheng, "Monitoring mouth movement for driver fatigue or distraction with one camera", In Proceedings.of the 7th IEEE International Conference on Intelligent Transportation Systems, pp. 314-319, October2004.
7. Y. Lu, Z. Wang, "Detecting driver yawning in successive images", In 1st International Conference on Bioinformatics and Biomedical Engineering, pp. 581-583, July, 2007.
8. X. Fan, B.C. Yin, Y.F. Sun, "Yawning detection for monitoring driver fatigue", In IEEE International Conference on Machine Learning and Cybernetics, vol. 2, pp. 664-668, August, 2007.
9. R.S. Medeiros, J. Scharcanski, A. Wong, "Multi-scale stochastic color texture models for skin region segmentation and gesture detection", In IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1-4, July, 2013.
10. L. Li, Y. Chen, Z. Li, "Yawning detection for monitoring driver fatigue based on two cameras", In 12th IEEE International Conference on Intelligent Transportation Systems, pp. 1-6, October, 2009.

11. C. Bouvier,A. Benoit, A. Caplier, P.Y. Coulon, "Open or closed mouth state detection: static supervised classification based on log-polar signature", In International Conference on Advanced Concepts for Intelligent Vision Systems, pp. 1093-1102, Springer, Berlin, Heidelberg, October, 2008.
12. N. Alioua, A. Amine, M. Rziza, "Driver's fatigue detection based on yawning extraction", International journal of vehicular technology, 2014.
13. M. Omidyeganeh, S. Shirmohammadi, S. Abtahi, A. Khurshid, M. Farhan, J. Scharcanski, B. Hariri, D. Laroche, et. al., "Yawning detection using embedded smart cameras", IEEE Transactions on Instrumentation and Measurement, vol. 65, no. 3, pp.570-582, 2016.
14. Z. Jie, M. Mahmoud, Q. Stafford-Fraser, P. Robinson, E. Dias, L. Skrypchuk, "Analysis of yawning behaviour in spontaneous expressions of drowsy drivers", In 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), pp. 571-576, May, 2018.
15. W. Zhang, J. Su, "Driver yawning detection based on long short-term memory networks", In IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1-5, November, 2017.
16. D.E. King, "Dlib-ml: A machine learning toolkit", Journal of Machine Learning Research, vol. 10, pp.1755-1758, 2009.
17. E. Barsoum, C. Zhang, C.C. Ferrer, Z. Zhang, "Training deep networks for facial expression recognition with crowd-sourced label distribution", In Proceedings of the 18th ACM International Conference on Multimodal Interaction, pp. 279-283, ACM, October, 2016.
18. F.A. Gers, J. Schmidhuber, F. Cummins, "Learning to forget: Continual prediction with LSTM", 1999.
19. S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, B. Hariri, "YawDD: A yawning detection dataset", In Proceedings of the 5th ACM Multimedia Systems Conference, pp. 24-28, ACM, March, 2014.
20. C.H. Weng, Y.H. Lai, S.H. Lai, "Driver drowsiness detection via a hierarchical temporal deep belief network", In Asian Conference on Computer Vision, pp. 117-133, Springer, November, 2016.
21. F. Chollet, "Keras", 2015.