International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

AN APPROACH FOR DETECTING DROWSY DRIVER USING KEY FRAME SELECTION FROM VIDEO

G N Sharath Kumar[1], Hanumanthappa J[2], Chethan. Raj.C[3,*], P Naveen Kumar[4]

[1]Research Scholar, Department of Studies in Computer Science, University of Mysore (UOM), Mysore, India, sharugn@gmail.com.

[2]Professor, Department of Studies in Computer Science, UoM, Mysore, India.

[3,*]Associate Professor, Department of Computer Science & Engineering, CCE, VTU, Mysore, India, cchethan005@gmail.com.

[4]Department of Electronics Yuvaraja's College, UOM, Mysore, India

Abstract

Drowsy driving is a leading cause of many road accidents. Existing methods for detecting drowsy drivers need to process all the frames; it requires a huge amount of computation in both space and time complexities. To reduce the complexities, we proposed a keyframe-based method for detecting drowsy drivers. It processes a smaller number of frames to detect the drowsy state of the driver than other existing methods present nowadays. The method initially converts video to frames and uses a 2-stage filtering strategy to select keyframes from all the frames. The method selects the first frame as the keyframe. In the first stage, it finds the color histogram difference between the keyframe and the next frame using Bhattacharya distance. The frame is selected as a keyframe in the first stage if the distance crossed the threshold value. In the second stage, we determine the structural similarity index between the key frame and the selected frame. Finally, the frame is selected as a keyframe if the value of the structural similarity index is less than the predefined threshold. This process is repeated until the completion of the video. We extracted Histogram of Oriented Gradient (HOG) and Local Binary Pattern (LBP) features and trained with Support Vector Machine Linear (SVM), SVM (Residual Based Functions-RBF), Random Forest (RF), and K-Nearest Neighbor (KNN). The model with HOG and LBP with trained SVM (RBF) detects the drowsy state of the driver. The model outperformed other classifiers and achieved an average accuracy of 82.5%. The experiment is performed on the National Tsing Hua University Drowsy Driver Detection (NTHUDDD) dataset.

Keywords: Keyframe, HOG, LBP, EAR, MAR, SVM(RBF), NTHU-DDD.

1. Introduction

Nowadays, video processing is a basic requirement for any option that monitors real-time circumstances. To determine whether an event or action has taken place, we must process all the frames, which requires significant computational time and storage space. To decrease this

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

complexity, we explored extracting and selecting a few frames that provide additional information about that occurrence, known as keyframes. Keyframe extraction is essential for video analysis. As a result, extensive study has been conducted on keyframe extraction. This research aims to identify the most representative frames that capture significant changes or events within a video. By focusing on these keyframes, we can enhance the efficiency of video processing while maintaining the integrity of the information conveyed. This approach not only streamlines the analysis process but also allows for better indexing and retrieval of video content. Consequently, advancements in machine learning and computer vision are increasingly being integrated into keyframe extraction techniques, leading to more accurate and automated systems. These developments are paving the way for innovative applications in various fields, such as surveillance, content creation, and education. As technology evolves, the potential for real-time analysis and improved user experiences continues to grow.

There are four major ways to keyframe extraction: sampling-based [1], object-based [2], segment-based [3], and shot-based [4]. In the sampling-based technique, keyframes are chosen at random without regard for the video's content. The keyframes chosen using the segment-based technique can effectively depict the video's content. Shot-based techniques are useful and crucial since they select keyframes from each shot. Object-based techniques are efficient and meaningful, but they prioritize the foreground and are only useful in specific applications. We considered detecting the driver's drowsy state utilizing keyframes, which minimizes computing complexity in a minimum number of frames. This approach allows for a more focused analysis, enhancing the accuracy of drowsiness detection while ensuring that the system remains responsive and efficient. By concentrating on the most relevant frames, we can significantly improve the effectiveness of the monitoring system in real-time scenarios.

Drowsiness can be identified using several parameters; based on the parameter, it is divided into three methods. These methods include physiological parameters, vehicle-based parameters, and behavioral parameters. Physiological parameters [5], [6], [7] use sensors attached to the driver’s body to gather information about their drowsy state. Generally, drivers are not comfortable with physiological sensors like EEG, ECG, and EOG. Vehicle-based parameters [8], [9], [10] include driving parameters, such as a camera mounted near the steering wheel to monitor the angular displacement of the steering wheel. A camera is mounted near the steering wheel to check the angular displacement of the steering wheel within a short period of time, while another camera is mounted on the rear side of the vehicle to monitor lane departure. Initially, behavioral parameters [11], [12], [13] include facial movement observation, which is performed by mounting a camera facing towards the driver and watching facial expressions such as closing of eyes, opening of mouth, and head nodding. The system can alert the driver if he feels drowsy, while ensuring that his comfort during driving is not compromised. This innovative approach enhances safety and promotes a more interactive driving experience. By leveraging advanced technology to monitor driver alertness, the system aims to reduce the risk of accidents caused by fatigue.

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

2. Related works

This section provides a concise summary of the literature on different researcher methods for detecting drowsy.

Albadawi et al. [14] developed a model to detect drowsy drivers using visual features. The proposed model detects faces using facial landmarks. A face mesh detector is used to locate the region of interest, such as mouth, eyes, and head features. Extracted features were fed into 3 different classifiers, namely, random forest, sequential neural network, and linear support vector machine. The experiment is performed on the NTHU-DDD dataset.

Majeed et al. [15] proposed a novel deep convolutional neural network model to detect drowsy drivers. The author used the YawDD video dataset, detected faces using Dlib's facial points, and labeled them as drowsy and non-drowsy videos using mouth aspect ratio. Then the author preprocessed the images and fed them into CNN. The experiment is performed with and without data augmentation. Both were given slight differences; however, yawning is one of the conditions to detect the drowsy state of the driver.

Essahraui et al. [16]. Developed a system to detect drowsiness of a driver in real-time using facial analysis and machine learning techniques. The experiment is performed on 3 public datasets, namely, NTHU-DDD, YawDD, and UTA-RLDD. The proposed system extracts facial regions and classifies them as drowsy and non-drowsy using KNN, SVM, DT, and Random Forest. KNN performs better among them. The system uses deep learning-based object detection models such as CNN, YOLOv5, YOLOv8, and Faster-RCNN. Both ML and DL methods performed better compared to the state-of-the-art datasets.

Jabbar et al. [17] developed a system to detect drowsy driving behavior of a driver using convolutional neural network techniques for Android applications. The proposed system detects faces using dlib's face detection, and then EAR & MAR are used to detect eye status and mouth status. Deep learning convolutional neural networks are used to detect the drowsy state of a driver.

Bhakheet et al. [18] proposed a framework for instantaneous driver drowsiness detection using improved HOG features and a naive Bayes classifier. The framework consists of face detection, eye status detection, and drowsiness classification. First, face detection is performed using the Viola-Jones algorithm, then the eye region is extracted using improved HOG features, and drowsiness detection is performed based on the Naive Bayes classifier.

Sharanabasappa et al. [19] proposed an ensemble learning model for drowsy driver detection and prevention using behavioral analysis. The proposed model performs preprocessing using contrast-limited adaptive histogram equalization. Object detection and tracking performed using KLT. Face region is performed using eigen values, the eye region is detected by Gabor filters, and Otsu thresholding is performed to detect the mouth region. All the features are selected by Relief, infinite, correlation, and term variance. These are fed into an ensemble learning model consisting of a decision tree, KNN, ANN, and SVM, which produces drowsy and non-drowsy frames.

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Luis et al. [20] developed a method to detect driver drowsiness using deep learning techniques on sequential images. The proposed method has 2 solutions. The first uses face detection using dlib for face detection and cropping; preprocessing is performed for unwanted information removal, and a sequence of 600 images is fed into efficientnetB0, which decides whether or not the person is drowsy. The second solution uses dlib's face detection, calculates EAR, and detects yawning by convolutional neural network. This information is sent to an analysis model, which decides whether or not the person is drowsy using the Mamdani fuzzy inference system.

Jahan et al. [21] developed a 4D model to detect real-time driver drowsiness using deep learning. The proposed model experimented on the MRL eyes dataset. The proposed model uses a custom CNN model called 4D, uses the MRL Eyes dataset, and detects the drowsy state of the driver based on open eyes and closed eyes.

Ahmad et al. [22] proposed a transfer learning approach for drowsy driver detection based on eye movement behavior. The method extracts spatial features from the eye image data, then generates transfer features using the Light Gradient Boosting Machine, and the drowsy state is detected by the k-neighbors classifier.

Yang et al. [23] developed an advanced driver-assistance system to detect the drowsy state of the driver based on a deep learning method. The proposed method is experimented on using the YawDD dataset. The method extracts complex features from the images from ShuffleNet; the drowsy state of a driver is identified by the extreme learning machine.

3. Proposed Methodology

One of the most significant causes of traffic accidents is drowsy driving. Current methods for detecting drowsy drivers require processing every single frame of a video, leading to a high demand for computational resources in terms of both space and time. To address these challenges, researchers are exploring more efficient algorithms that can identify symptoms of drowsiness as they arise. These algorithms aim to function without needing extensive data processing. Ultimately, enhancing road safety may be achievable by employing machine learning techniques, which could improve detection accuracy while simultaneously reducing resource consumption.

To address these challenges, we propose a keyframe-based method for detecting drowsy drivers, which analyzes fewer frames to assess the driver's drowsy state compared to current methods. This approach not only reduces the computational burden but also enables real-time monitoring, making it more feasible for implementation in vehicles. By focusing on keyframes, we can achieve efficient detection without compromising accuracy, potentially improving overall road safety.

Our approach is to convert a video into frames and employs a two-stage filtering strategy to select keyframes. The first frame is designated as the keyframe. In the first stage, we calculate the color histogram difference between the keyframe and the subsequent frame using Bhattacharya distance. A frame is selected as a keyframe in this stage if the distance

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

exceeds a predefined threshold. In the second stage, we analyze motion vectors to further refine our selection of keyframes, ensuring that significant changes in scene dynamics are captured. This dual-filtering mechanism enhances the robustness of our detection algorithm, leading to more reliable outcomes in diverse driving conditions. In the second stage, we assess the structural similarity index between the keyframe and the chosen frame. A frame is selected as a keyframe if the structural similarity index value falls below a set threshold. This process continues until the video is fully processed.

We extracted Histogram of Oriented Gradient (HOG) and Local Binary Pattern (LBP) features and trained various classifiers, including Support Vector Machine (SVM) with a linear kernel, SVM with a Radial Basis Function (RBF) kernel, Random Forest (RF), and K-Nearest Neighbor (KNN). The model utilizing HOG and LBP features with the SVM (RBF) classifier effectively detects the drowsy state of the driver. This model outperformed the other classifiers, achieving an average accuracy of 82.5%. The experiments were conducted using the National Tsing Hua University Drowsy Driver Detection (NTHUDDD) dataset.

3.1 Dataset

The proposed method is experimented on the NTHUDDD (National Tsing Hua University Drowsy Driver Detection) dataset [24]. The dataset includes 18 individuals from various ethnic backgrounds, each represented in five different scenarios. The scenarios include bare eyes, spectacles, bare eyes at night, spectacles at night, and sunglasses during the day. The dataset provides a diverse range of conditions that are essential for developing effective systems to detect drowsy drivers. By analyzing these scenarios, researchers can better understand the impact of different visual factors on driver alertness and improve the accuracy of detection algorithms. These insights can lead to the creation of more effective monitoring systems that can alert drivers before drowsiness becomes a serious issue. Ultimately, advancements in this area could significantly enhance road safety and reduce the number of accidents caused by driver fatigue.

3.2 Algorithms & Experimentation

The proposed method converts video into frames first; while converting, the first frame is considered as keyframe and it is converted into HSV format. Then next frame is considered and converted into HSV format, then we calculate histogram difference using Bhattacharya distance formula [25]. The distance is greater than the predefined threshold then that frame is considered. Then we perform Structural Similarity Index between Key frame and the considered frame. next frame is chosen as keyframe if the SSI [26] is less than threshold (0.99). otherwise, next frame is chosen for the keyframe if it greater than Bhattacharya distance and SSI is less than threshold value. Next keyframe is selected if it passes the 2 conditions otherwise the process is repeated for all the frames. In the second stage keyframe based drowsiness of a driver is detected and the results obtained from the experiment are accurate compared with the original drowsy frames.

3.2.1 Novel method for key frame extraction

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Initially video is converted into frames. By default, we considered initial frame as keyframe (eq.1), then we resized image to 256 X 256 size and converted into HSV format. We choose next frame, resized to 256 X 256 size and converted into HSV format as shown in figure 1. We find difference between color histogram of both the keyframe and the next frame using Bhattacharya distance formula. We select the next frame if the distance between the keyframe and next frame crossed predefined threshold this is the first stage of selecting keyframe otherwise next frame is selected for first stage filtering. Then we calculate structural similarity index between selected frame and the keyframe. The next frame is selected as keyframe (eq.2) if the SSI is less than the predefined threshold, the second stage of filtering. Otherwise, next frame is chosen and processed for 2 stages filtering to select keyframe. This process is repeated until end of the video.

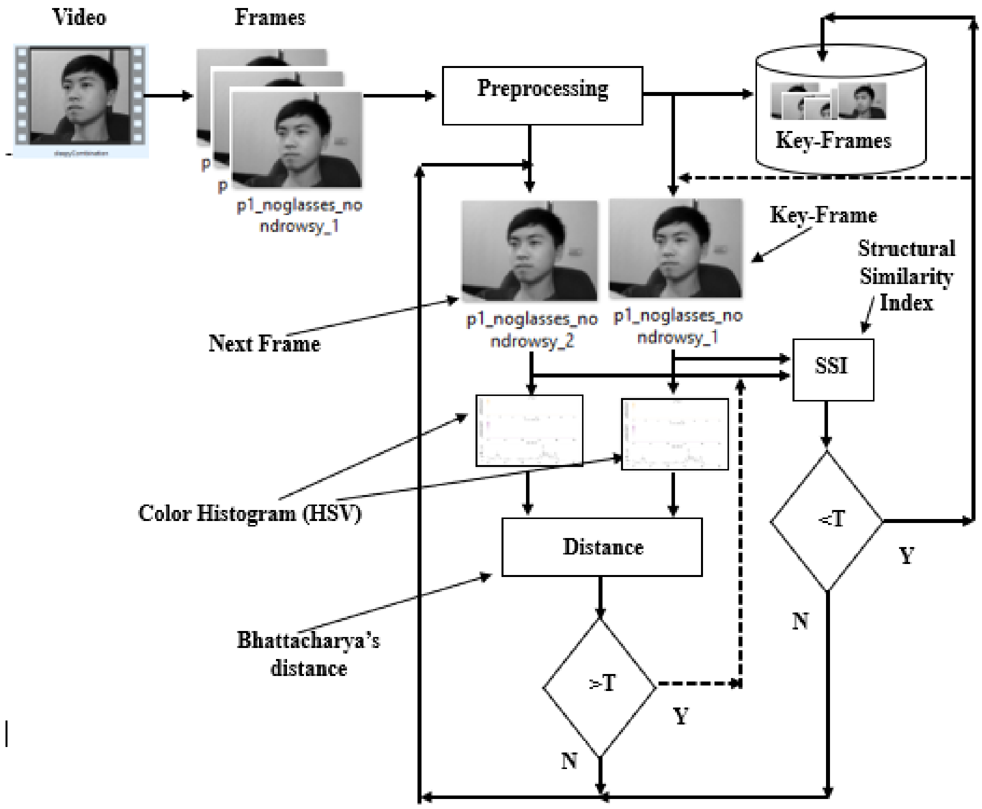

Fig 1: Keyframe Selection Architecture

$$\boldsymbol{kf}_{(1)} \leftarrow \boldsymbol{f}_{(1)},\quad \boldsymbol{i} = \mathbf{1} \text{ ....} \quad (1)$$

$$\boldsymbol{kf}_{(i)} \leftarrow \boldsymbol{f}_{(i+1)}, \boldsymbol{if}\left(\left(\boldsymbol{Bt\_d}\left(\boldsymbol{C\_h}\left(\boldsymbol{kf}_{(i)}\right), \boldsymbol{C\_h}\left(\boldsymbol{f}_{(i+1)}\right)\right) > \mathbf{0.02}\right) \& \left(\boldsymbol{SSI}\left(\boldsymbol{kf}_{(i)}, \boldsymbol{f}_{(i+1)}\right) < \mathbf{0.99}\right)\right), \boldsymbol{i} > \mathbf{1} \text{ ....} \quad (2)$$

$$where,\ f \leftarrow frame,$$

$$kf \leftarrow keyframe,$$

$$Bt\_d \leftarrow Bhattacharya\ distance,$$

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$C\_h \leftarrow Color\ Histogram,$$

$$SSI \leftarrow Structural\ Similarity\ Index$$

3.2.2 Algorithm: Key-Frame Selection

Input: Video

Output: Key-Frames, [Kf_1, Kf_2, Kf_3, Kf_4,….. Kf_n]

Method:

1. Initialize Key-Frame

   Key-Frames = [ ]

2. Convert Video to frames

   V $\rightarrow$ f_1, f_2, f_3, f_4,….. f_n.

3. Assign Key-Frame and select the next Keyframe

   for i = 1 to (n-1)

   {

   if (i == 1)

   {

   $Kf_{(i)} \leftarrow f_{(i)}$

   Key-Frames.append($Kf_{(i)}$)

   }

   Compute Color Histogram for Key-Frame and next Frame

   Kf_hist $\leftarrow$ Color_Hist(Kf_i)

   $f\_hist_{(i+1)}$ $\leftarrow$ Color_Hist($f_{(i+1)}$)

   Compute Bhattacharya distance between them

   d $\leftarrow$ Bhattacharya_distance(Kf_hist, $f\_hist_{(i+1)}$)

   if (d > 0.02)

   {

   Compute Structural Similarity Index between Key-Frame and next Frame

   Score $\leftarrow$ SSI ($Kf_{(i)}$, $f_{(i+1)}$)

   If (Score < 0.99)

   {

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$Kf_{(i)} \leftarrow f_{(i+1)}$

Key-Frames.append($Kf_{(i)}$)

}

}

}

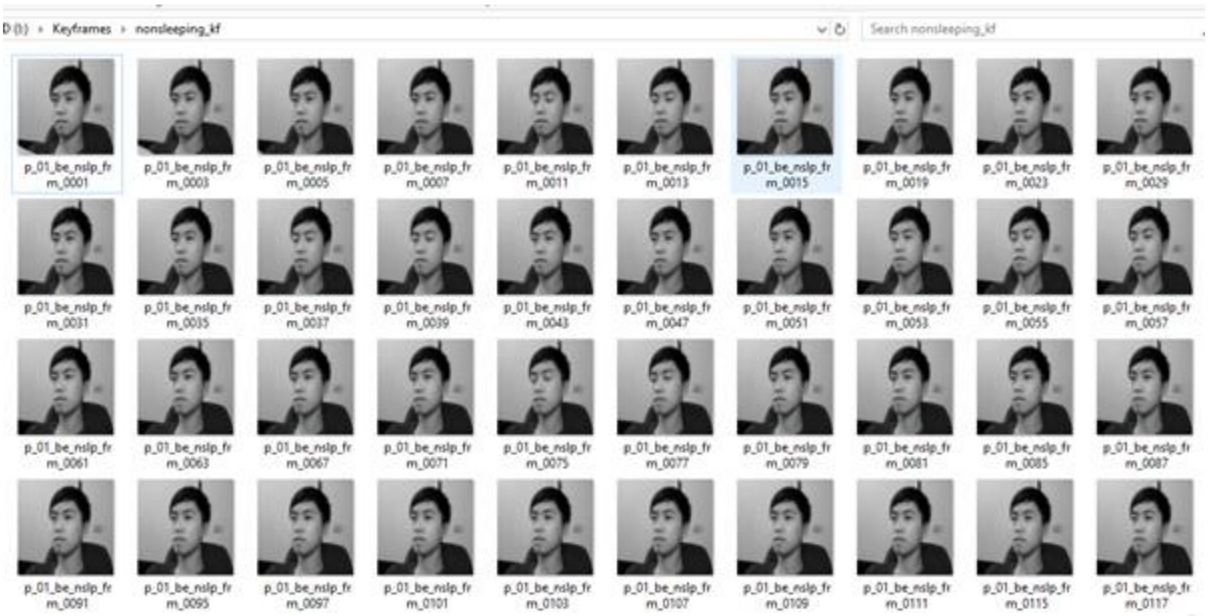

Fig 2: Frames from a Video

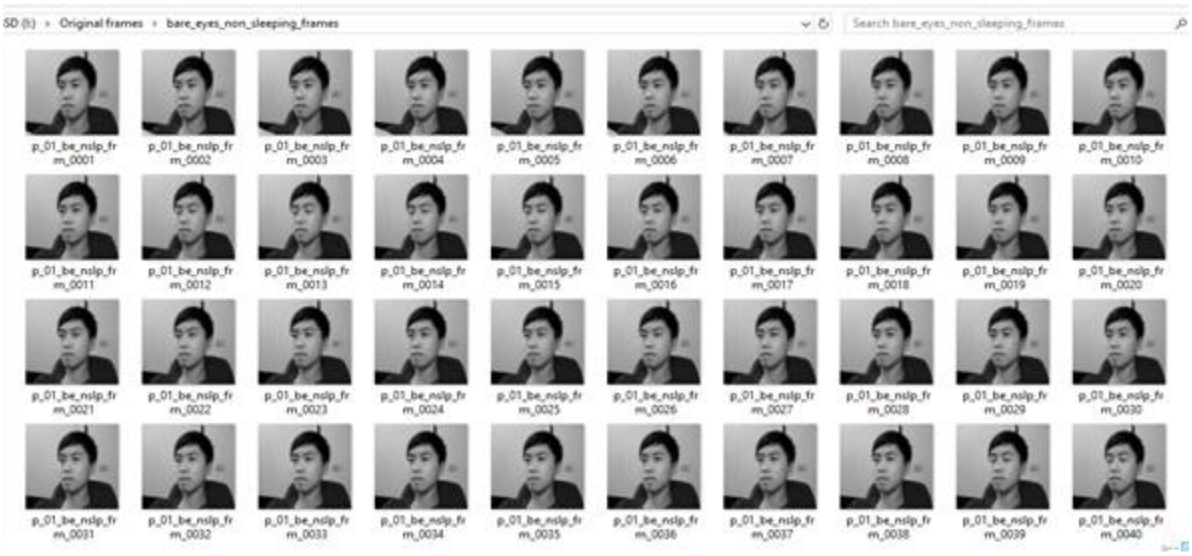

Fig 3: Key Frames from a Video

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

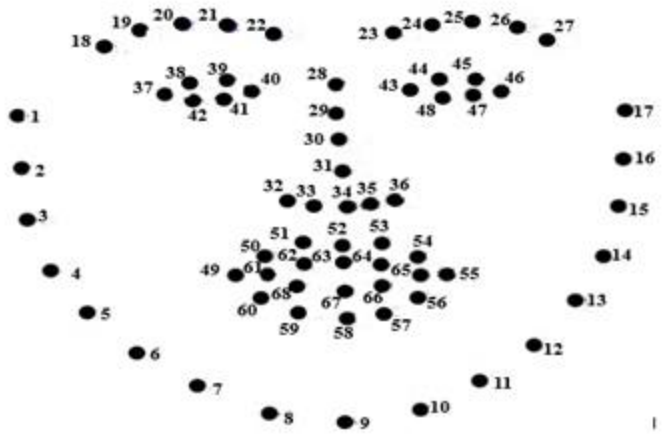

Fig 4: Dlib's face detection using 68 facial points

Face Detection: Dlib's face detection algorithm localizes facial region and detects key facial structures [27] called 68 facial points (fig 4). It detects facial parts such as nose (28 to 36), mouth (49 to 68), eyes (37 to 46), eye brows (18 to 27) and jaws (1 to 17).

Eyes Aspect Ratio: It uses 6 points to detect status of eyes [28]. Sum of 2 vertical distances divided by the product of 2 and the horizontal difference (eq.3). Average of 2 eyes is considered as EAR. The differences identified by Euclidian distance.

Mouth Aspect Ratio: It uses 8 points to detect status of mouth [28] i.e. sum of 3 vertical distances divided by the product of 2 and the horizontal difference (eq.4). The differences identified by Euclidian distance.

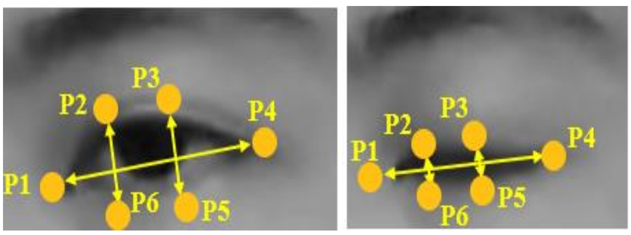

Fig 5: (a) open eye, (b) closed eye with facial points

$EAR = \frac{\|P2-P6\|+\|P2-P6\|}{2(P1-P4)}$……………..(3)

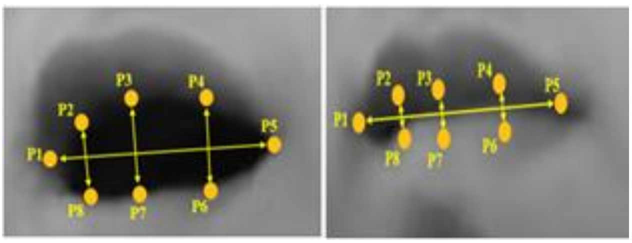

Fig 6: (a) open mouth, (b) closed mouth with facial points

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$MAR = \frac{\|P2-P8\|+\|P3-P7\|+\|P4-P6\|}{2(P1-P5)}$……..(4)

Table 1: Training and testing results of a single Video

Sl No	Feature	Classifier	Training		Testing	
			Accuracy	Time (sec.)	Accuracy	Time (sec.)
1	LBP	SVM Linear)	91.32	0.23	91.32	0.23
2	LBP	SVM (RBF)	93.9	0.14	93.9	0.14
3	LBP	Random Forest	92.15	1.09	92.15	1.09
4	LBP	KNN (k = 5)	93.43	0.35	93.43	0.35
5	HOG	SVM Linear)	95.43	27.58	95.43	27.58
6	HOG	SVM (RBF)	95.66	49.86	95.66	49.86
7	HOG	Random Forest	94.61	30.65	94.61	30.65
8	HOG	KNN (k = 5)	94.96	2.65	94.96	2.65
9	HOG+LBP	SVM Linear)	95.43	21.31	95.43	21.31
10	**HOG+LBP**	**SVM (RBF)**	**95.66**	**47.67**	**95.66**	**47.67**
11	HOG+LBP	Random Forest	94.61	26.25	94.61	26.25
12	HOG+LBP	KNN (k = 5)	94.96	2.58	94.96	2.58

We have performed experiments on a several videos, however we have presented information about a single video. We have converted video into frames, divided all frames to Drowsy and non-drowsy frames based on EAR and MAR and extracted HOG + LBP features and classified as drowsy and non-drowsy using SVM(RBF). Later, we have taken all the frames and extracted keyframes based on color histogram and structural similarity index, then we divided all keyframes as drowsy and non-drowsy using EAR and MAR. then we extracted LBP[29], HOG[30], LBP & HOG features [31]. We used classifier SVM [32], Random Forest [33], and KNN [34].

In the training phase, we had imbalanced dataset (Table 2) because drowsy frames are less compared with non-drowsy frames. We balanced (Table 3) by choosing the minimum samples (Drowsy) and choose equal number of other samples (non-drowsy) and are used for training process. Similarly, equal number of keyframes are choosen for testing.

Table 2: Original frames and Key-frames (imbalance dataset)

Sl	Video	Drowsy	Non-	Drowsy	Non-

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

No		frame	Drowsy frame	Keyframe	Drowsy Keyframe
1	Bare eyes day time sleepy	4720	3872	580	4710
2	With Spectacles day time	5392	42734	730	5426
3	Bare eyes night time	5804	16342	548	1756
4	With Spectacles night time	1229	19165	184	2064
5	With Sun glasses day time	5083	37430	412	4139

Table 3: Original frames and Key-frames (balanced dataset)

Sl No	Sleeping Videos	Drowsy frame	Non-Drowsy frame	Drowsy Keyframe	Non-Drowsy Keyframe
1	Bare eyes day time	4702	4702	580	580
2	With Spectacles day time	5392	5392	730	730
3	Bare eyes night time	5804	5804	548	548
4	With Spectacles night time	1229	1229	184	184
5	With Sun glasses day time	5083	5083	412	412

The training set contains all the drowsy and non-drowsy frames except keyframes. The test set contains all the drowsy and non-drowsy keyframes. The aforementioned table shows experimental results.

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

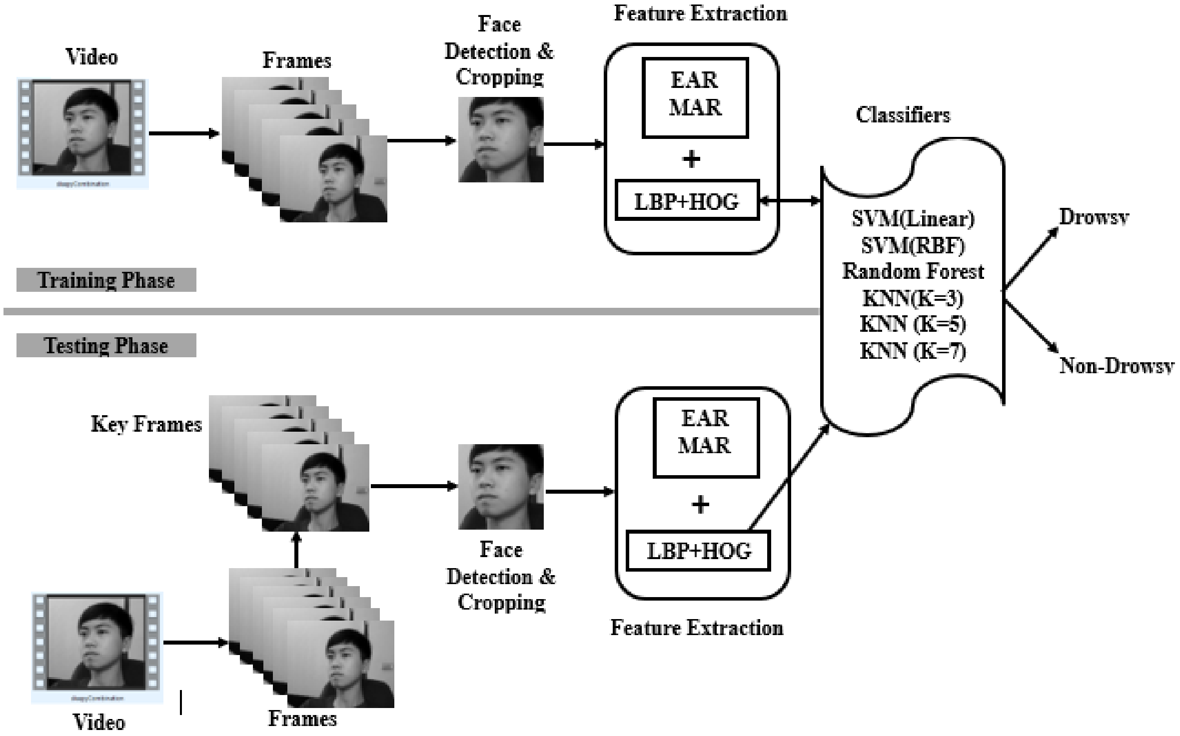

Fig 7: Drowsy Driver Detection based on key-frame architecture.

Training phase: Drowsy and Non-Drowsy frames

$$f(f) = \begin{cases} D, \ if \ (HOG + LBP) \ and \ (eyes_c > 10 \ cons\_f \ \ or \ mouth_o > 8 \ cons\_f) \\ ND, \ otherwise \end{cases} \quad \ldots\ldots\ldots\ldots\ldots.. (5)$$

Testing phase: Drowsy and Non-Drowsy key-frames

$$f(kf) = \begin{cases} Dkf, \ if (HOG + LBP) \ and \ ((eyes_c > 6 \ cons\_kf) \ \ or \ (mouth_o > 3 \ cons\_kf)) \\ NDkf, \ otherwise \end{cases} \quad \ldots\ldots (6)$$

$Where, f \leftarrow frame$,

$D \leftarrow Drowsy,$

$ND \leftarrow NonDrowsy,$

$HOG \leftarrow Histogram\ of\ Oriented\ Gradients,$

$LBP \leftarrow Local\ Binary\ Pattern,$

$eyes_c \leftarrow Closed\ Eyes,$

$con\_f \leftarrow Consecutive\ frames,$

$mouth_o \leftarrow Open\ Mouth,$

$kf \leftarrow keyframe,$

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$Dkf \leftarrow Drowsy\ keyframe,$$

$$NDkf \leftarrow NonDrowsy\ keyframe,$$

$$cons\_kf \leftarrow Consecutive\ keyframe$$

3.3 Drowsy State

In a frame, closed eyes state is detected by EAR less than 0.3 and drowsy state (fig 5) is identified if the state of the eyes is closed for 10 consecutive frames. Similarly, open mouth state is identified by MAR is greater than 0.75 and drowsy state is detected if the mouth of a driver is opened for 8 consecutive frames. Along with that, we extracted face features with LBP and Hog. Trained with SVM(Linear), SVM(RBF), Random Forest, KNN(K=3), KNN(K=5) and KNN(K=7). This is performed in training phase. Similarly in the testing phase, we extracted keyframes from the video, drowsy state is detected using closed eyes status for 6 consecutive keyframes or mouth open state for 3 consecutive keyframes. Along with that, we extracted face features with LBP and Hog. Trained with SVM(Linear), SVM(RBF), Random Forest, KNN(K=3), KNN(K=5) and KNN(K=7). Then trained classifiers tested on the keyframes. SVM(RBF) with LBP and HOG features performs better among the other classifiers.

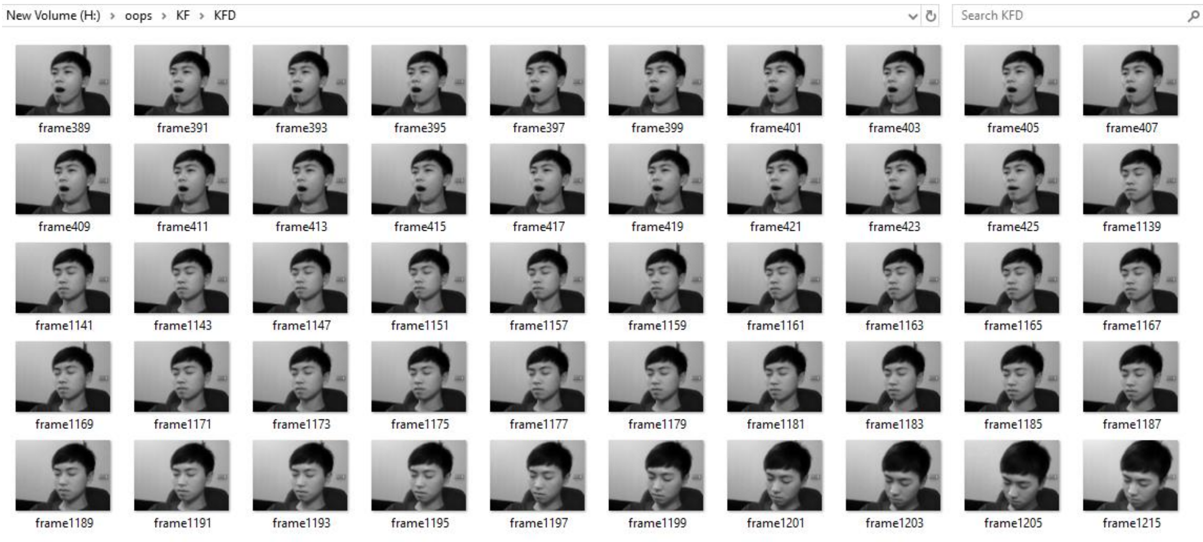

Fig 8: Drowsy Key-Frame

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

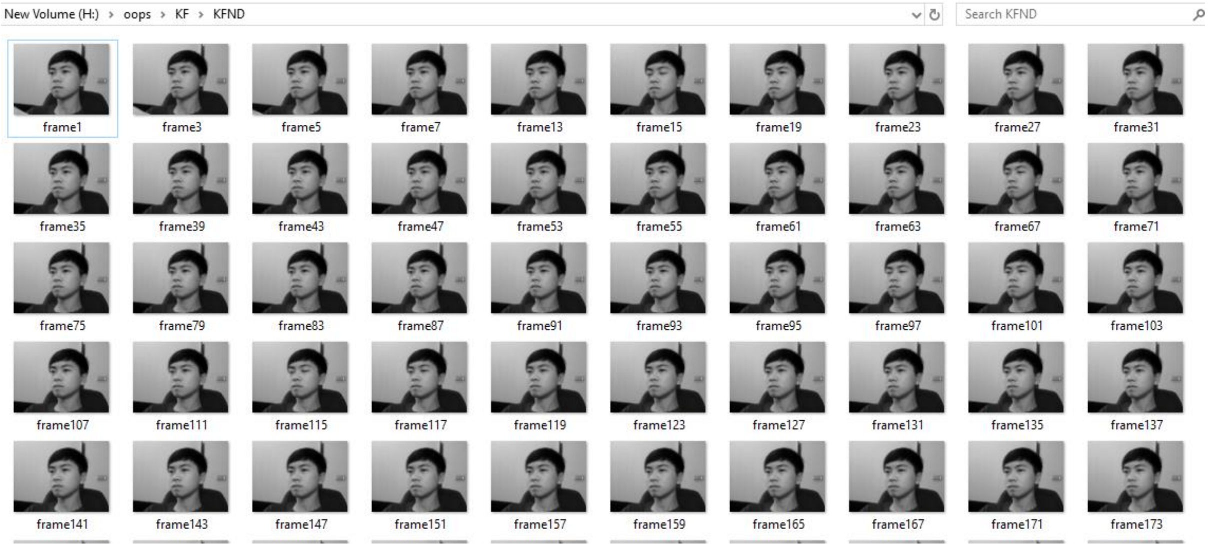

Fig 9: Non-Drowsy Key-Frame

Following graphs show the comparative results of SVM(linear), SVM(RBF), Random Forest, KNN(K=3), KNN(K=5) and KNN(K=7) experimented on LBP + Hog features in all the scenarios i.e bare eyes in day-time(fig 7), bare eyes in night-time(fig 8), eyes with spectacles in day-time(fig 9), eyes with spectacles in night-time(fig 10), eyes with sunglasses in day-time(fig 11). SVM(RBF) classifier performs better than among others(i.e. the gap between training accuracy and testing accuracy is low compared with other models).

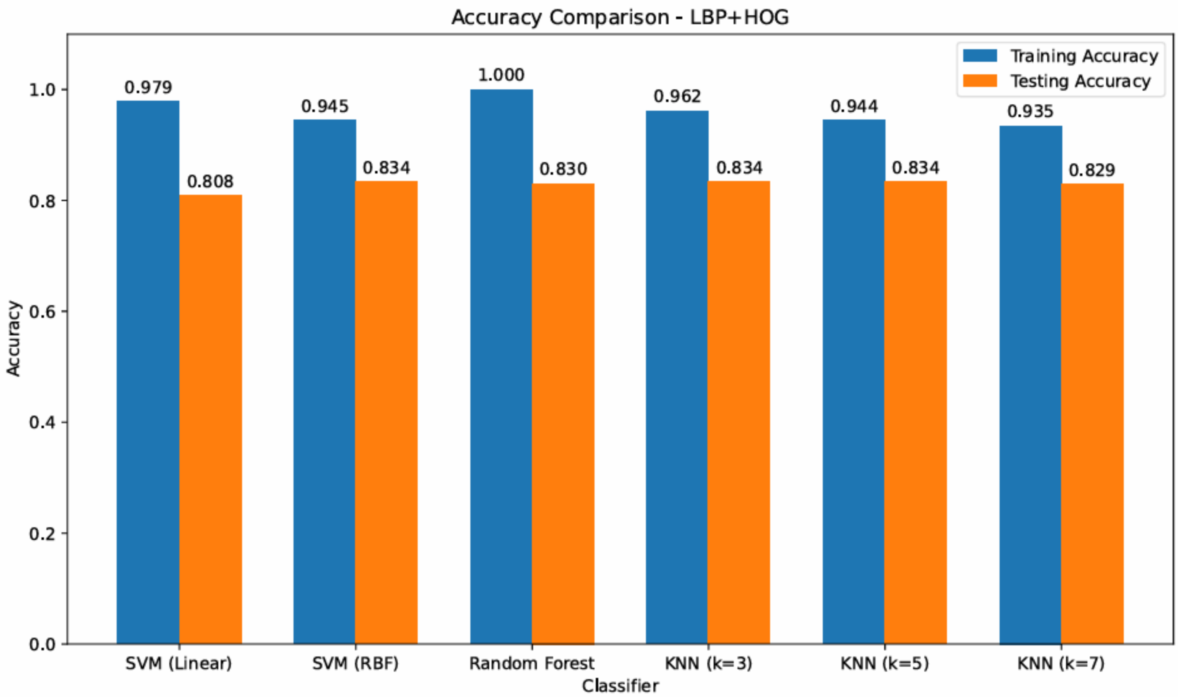

Fig 10: classifiers accuracy comparison for bare eyes day time

Received: August 03,2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

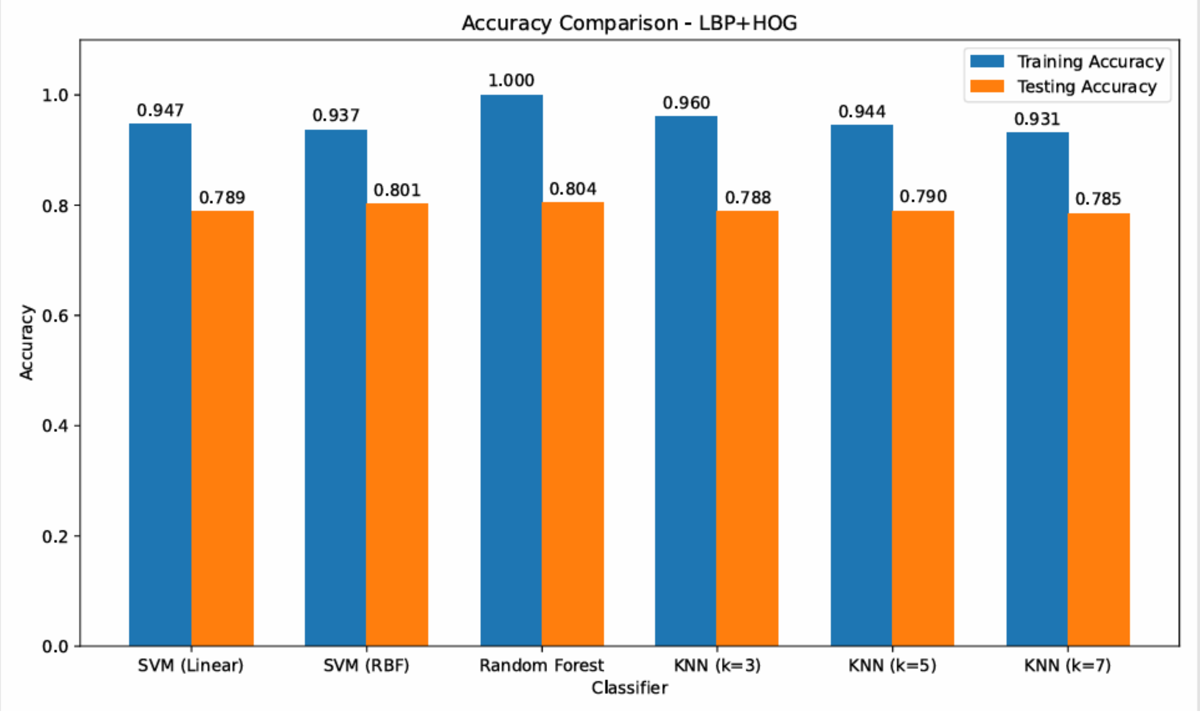

Fig 11: classifiers accuracy comparison for bare eyes night time

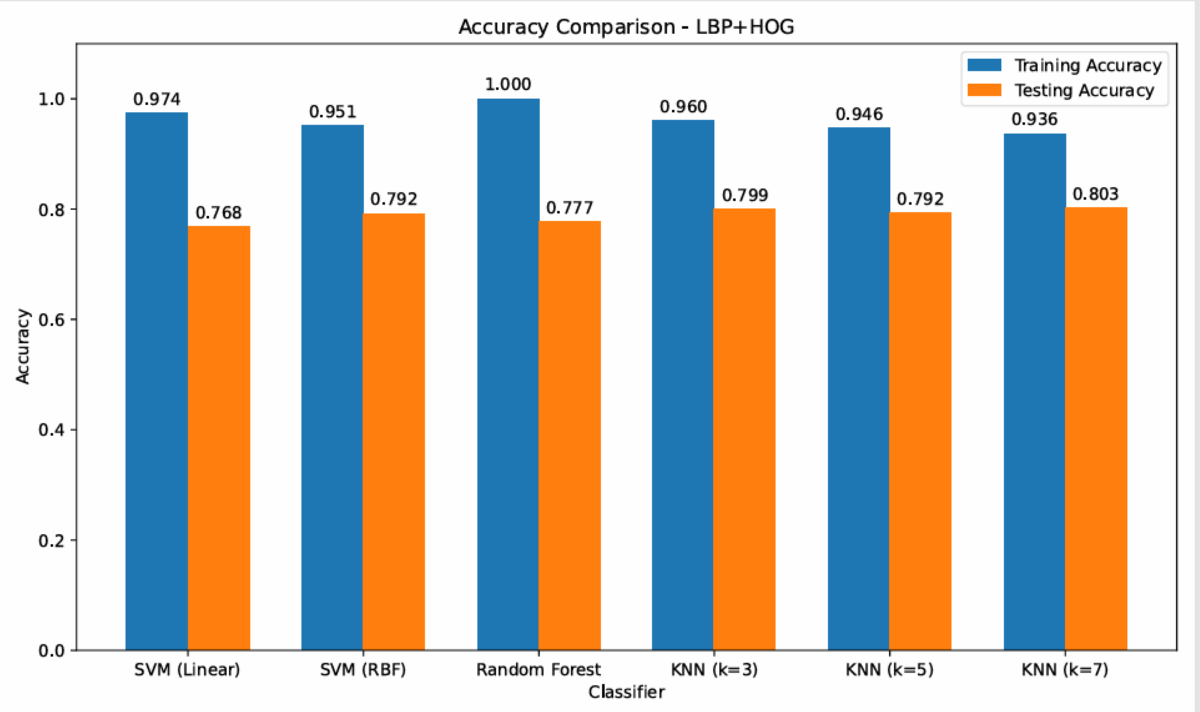

Fig 12: classifiers accuracy comparison for spectacles day time

Received: August 03,2025

International Journal of Applied Mathematics

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

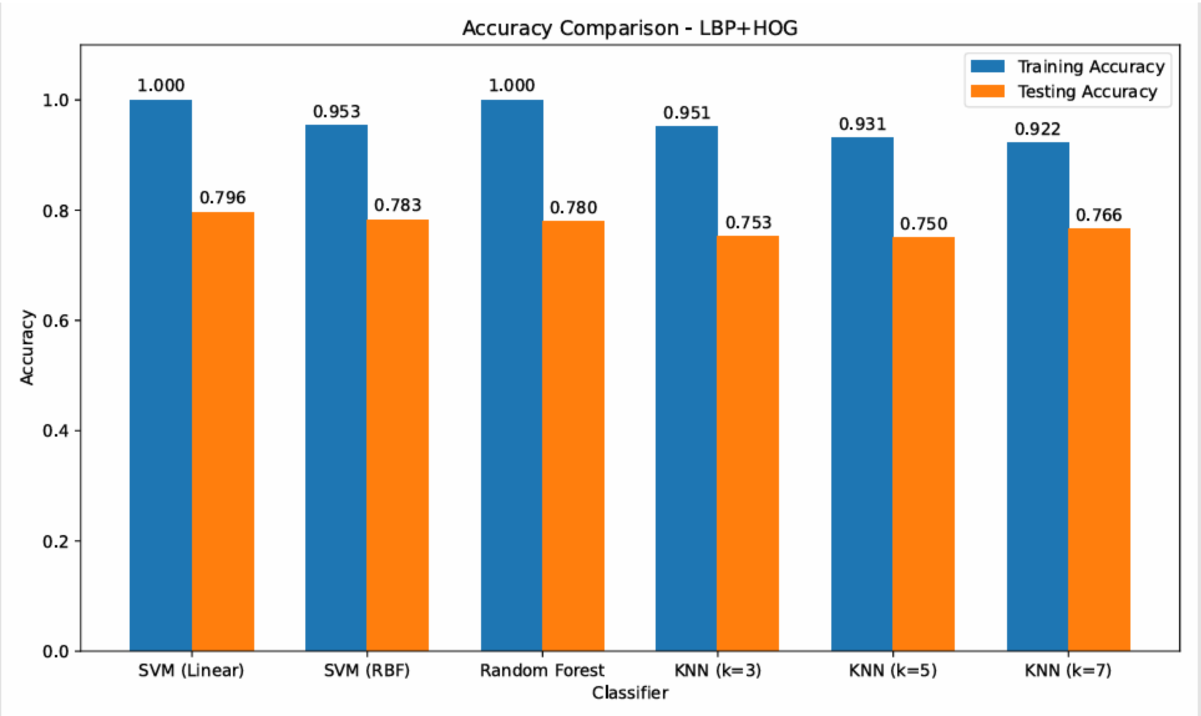

Fig 13: classifiers accuracy comparison for spectacles night time

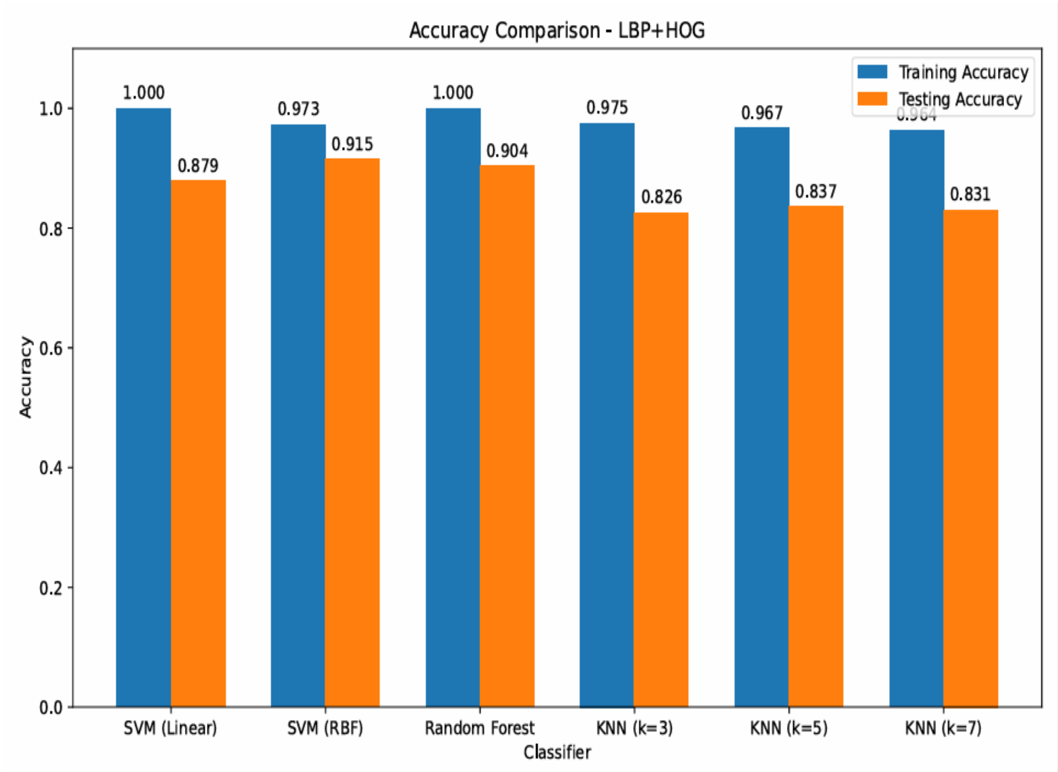

Fig 14: classifiers accuracy comparison for sunglasses

Validation of keyframe: We experimented on several videos and we have presented the details of one video for the validation of keyframe whether the keyframe belongs to Drowsy

Received: August 03,2025

Volume 38 No. 5s 2025,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

or not. Out of 2712 frames we got 867 Key frames. We have tried different combination of consecutive keyframes having closed eyes state or opened mouth state, out of them 712 keyframes are correctly classified as Drowsy and 155 is miss classified as Drowsy. The following table shows correct and miss classification of Drowsy Keyframes with Drowsy frames. The following table (table 2) shows correct and miss classification of Drowsy Keyframes.

Table 4: combination of consecutive keyframe for drowsy state detection

Sl. No	Closed Eye	Open Mouth	Correct Match	Miss Match	Matching Drowsy keyframe with Drowsy frame (%)
1	5	2	692	175	79.7
2	5	3	695	172	80.1
3	5	4	698	169	80.4
4	6	2	705	162	81.3
5	**6**	**3**	**712**	**155**	**82.12**
6	6	4	709	158	81.31
7	7	2	703	164	81.08
8	7	3	700	167	80.7
9	7	4	693	174	79.8

4. Conclusion

In this paper we proposed keyframe-based drowsy driver detection. The algorithm was tested on the NTHUDDD dataset. We used 2 stages of filtering to select keyframes from the video. Obtained keyframes were used as test samples, and the remaining frames except keyframes were used as training samples. We extracted HOG and LBP features from cropped faces of all the frames, then the model was trained with 4 classifiers, namely SVM with linear kernel, SVM with RBF kernel, Random Forest, and KNN. The performance of the model having SVM with RBF kernel showed less difference between training and testing accuracy in all the scenarios compared with other classifiers. However, other classifiers are given better accuracy and are overfitted. The proposed method detects drowsy drivers with a smaller number of frames, which reduces both computational complexity and space complexity. The method extracts keyframes and has few misclassifications of drowsy keyframes; hence, we try to minimize the misclassification by utilizing more features in the future.

References:

[1] Klose, Felix et al. "Sampling based scene-space video processing." *ACM Transactions on Graphics (TOG)* 34 (2015): 1 - 11.

Received: August 03,2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[2] Ji, Zhong et al. "Surveillance video summarization based on moving object detection and trajectory extraction." *2010 2nd International Conference on Signal Processing Systems* 2 (2010): V2-250-V2-253.

[3] Omidyeganeh, Mona et al. "Video Keyframe Analysis Using a Segment-Based Statistical Metric in a Visually Sensitive Parametric Space." *IEEE Transactions on Image Processing* 20 (2011): 2730-2737.

[4] Rashmi, B. S. and H. S. Nagendraswamy. "Shot-Based Keyframe Extraction Using Bitwise-XOR Dissimilarity Approach." *International Conference on Recent Trends in Image Processing and Pattern Recognition* (2016).

[5] Fujiwara, Koichi et al. "Heart Rate Variability-Based Driver Drowsiness Detection and Its Validation With EEG." *IEEE Transactions on Biomedical Engineering* 66 (2019): 1769-1778.

[6] Yaacob, Sazali Bin et al. "Drowsiness detection using EEG and ECG signals." *2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)* (2020): 1-5.

[7] Abidi, Afef et al. "Automatic Detection of Drowsiness in EEG Records Based on Machine Learning Approaches." *Neural Processing Letters* 54 (2022): 5225 - 5249.

[8] Lawoyin, Samuel et al. "Accelerometer-based steering-wheel movement monitoring for drowsy-driving detection." *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 229 (2015): 163 - 173.

[9] Wierwille, Walter W. et al. "RESEARCH ON VEHICLE-BASED DRIVER STATUS/PERFORMANCE MONITORING; DEVELOPMENT, VALIDATION, AND REFINEMENT OF ALGORITHMS FOR DETECTION OF DRIVER DROWSINESS. FINAL REPORT." (1994).

[10] Krajewski, Jarek et al. "Detecting Sleepy Drivers by Pattern Recognition based Analysis of Steering Wheel Behaviour." (2010).

[11] Sharath Kumar, G. N. et al. "Detection of Driver's Drowsiness in a Video based on Deep Features." *2024 5th International Conference on Innovative Trends in Information Technology (ICITIIT)* (2024): 1-9.

[12] Kumar, Ashish and Rusha Patra. "Driver drowsiness monitoring system using visual behaviour and machine learning." *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (2018): 339-344.

[13] Ou, Wei-Liang et al. "Intelligent video-based drowsy driver detection system under various illuminations and embedded software implementation." *2015 IEEE International Conference on Consumer Electronics - Taiwan* (2015): 192-193.

[14] Albadawi, Yaman et al. "Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features." *Journal of Imaging* 9 (2023): n. pag.

[15] Majeed, Fiaz et al. "Detection of Drowsiness among Drivers Using Novel Deep Convolutional Neural Network Model." *Sensors (Basel, Switzerland)* 23 (2023): n. pag.

[16] Essahraui, Siham et al. "Real-Time Driver Drowsiness Detection Using Facial Analysis and Machine Learning Techniques." *Sensors (Basel, Switzerland)* 25 (2025): n. pag.

[17] Jabbar, Rateb et al. "Applied Internet of Things IoT: Car monitoring system for Modeling of Road Safety and Traffic System in the State of Qatar." (2018).

[18] Bakheet, Samy and Ayoub Al-Hamadi. "A Framework for Instantaneous Driver Drowsiness Detection Based on Improved HOG Features and Naïve Bayesian Classification." *Brain Sciences* 11 (2021).

Received: August 03,2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[19] Sharanabasappa and Suvarna S. Nandyal. "An ensemble learning model for driver drowsiness detection and accident prevention using the behavioral features analysis." *Int. J. Intell. Comput. Cybern.* 15 (2021): 224-244.

[20] Javier, Luis et al. "Driver Drowsiness Detection by Applying Deep Learning Techniques to Sequences of Images." *Applied Sciences* (2022): n. pag.

[21] Jahan, Israt et al. "4D: A Real-Time Driver Drowsiness Detector Using Deep Learning." *Electronics* (2023): n. pag.

[22] Madni, Hamza Ahmad et al. "Novel Transfer Learning Approach for Driver Drowsiness Detection Using Eye Movement Behavior." *IEEE Access* 12 (2024): 64765-64778.

[23] Yang, Eunmok and Okyeon Yi. "Enhancing Road Safety: Deep Learning-Based Intelligent Driver Drowsiness Detection for Advanced Driver-Assistance Systems." *Electronics* (2024): n. pag.

[24] Weng, Ching-Hua et al. "Driver Drowsiness Detection via a Hierarchical Temporal Deep Belief Network." *ACCV Workshops* (2016).

[25] Kailath, Thomas. "The Divergence and Bhattacharyya Distance Measures in Signal Selection." *IEEE Transactions on Communications* 15 (1967): 52-60.

[26] Brunet, Dominique et al. "On the Mathematical Properties of the Structural Similarity Index." *IEEE Transactions on Image Processing* 21 (2012): 1488-1499.

[27] Kazemi, Vahid and Josephine Sullivan. "One millisecond face alignment with an ensemble of regression trees." *2014 IEEE Conference on Computer Vision and Pattern Recognition* (2014): 1867-1874.

[28] Wang, Y., & Chen, M. (2014). A Real-Time Method for Driver Drowsiness Detection. *IEEE Transactions on Intelligent Transportation Systems, 15(1), 128-138.*

[29] Ojala, Timo et al. "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns." *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (2002): 971-987.

[30] Dalal, Navneet and Bill Triggs. "Histograms of oriented gradients for human detection." *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* 1 (2005): 886-893 vol. 1.

[31] Ghorbani, Mohsen et al. "HOG and LBP: Towards a robust face recognition system." *2015 Tenth International Conference on Digital Information Management (ICDIM)* (2015): 138-141.

[32] Zeng, Chengbin et al. "Fast human detection using mi-sVM and a cascade of HOG-LBP features." *2010 IEEE International Conference on Image Processing* (2010): 3845-3848.

[33] Adouani, Amal et al. "Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences." *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD)* (2019): 266-271.

[34] Tan, Hengliang et al. "Face recognition based on the fusion of global and local HOG features of face images." *IET Comput. Vis.* 8 (2014): 224-234.

Received: August 03,2025