Marmara University Faculty of Engineering

Computer Engineering Department

CSE4097 - Engineering Project I

"DRIVER DROWSINESS DETECTION"

Project Specification Document

Students:	Ayşenur	YILMAZ	150114002
	Mahmut	AKTAŞ	150115010
	Mustafa Abdullah	HAKKOZ	150117509

Supervisor: Prof. Dr. Ciğdem Eroğlu Erdem

- 24.04.2020 -

Contents

1. Problem Statement

According to the KGM (Karayolları Genel Müdürlüğü), 3704 people have died in road accidents last year in Turkey and 92.65% of these accidents were caused by driver defects [1]. In another study performing for commercial vehicle drivers in Edirne district of Turkey, 22 out of 138 drivers (15.9%) have experienced or survived a risk of at least one traffic accident due to sleepiness [2]. On this basis, the development of a robust and practical drowsiness detection system is a crucial step. This project aims to make a system that detects the driver's drowsiness status based on their facial behaviors.

2. Problem Description and Motivation

Driver fatigue is a persistent danger for drivers and road safety which is the dominant reason for road accidents. According to statistics, highway road crashes hold 11.09% of the total number of accidents. There are lots of other reasons that can also form the origin of an accident besides driver fatigue, such as illegal parking, overloading, illegal overtaking, excessive speed, unsafe lane changes, etc. The driver drowsiness can be held responsible for a major contributor to the road crashes, which has a 2%-23% estimated share on the pie. Since it is difficult to estimate the exact number of fatigue-related accidents, these shares are still conservative estimations [3].

Light and dark cycle influences sleep and wakefulness. Therefore humans mostly awake during daylight and asleep during darkness. People like night workers, aircrews, travelers etc. may sleep out of this cycle and can experience sleep loss and drowsiness [8].

In the light of this information, most fatigue-related accidents occur during normal sleeping hours, and the more serious the crash, the more possibility that the driver was drowsy. Fatigue almost covers one-third of the causes of single-vehicle accidents in rural areas.

It is a common belief that fatigue is only a problem for long-distance drivers, however it has the same relevancy for short-distance drivers. Driving generally is not the main cause of the fatigue. Drivers usually are already tired when they get into the car because of working long hours, lack of sleep, sleep apnea or shift work.

There are several reasons of drowsy driving

- A lack of quality sleep
- Driving when you would normally be sleeping (overnight)
- Sleep disorders such as sleep apnea, a sleeping condition that causes tiredness throughout the day.

All people should know that, people can't fight with sleep [4].

Fatigue has a massive effect on drivers' capability of driving and jeopardizes their safe travel. According to a research, being awake for 17 hours has the same effect as BAC (blood alcohol concentration) of 0.05 in case of driving ability [9, 10]. Going without sleep for 24 hours has the same effect as BAC of 0.1 which is double of the legal limit [10, 11].

Driving while tired or fatigued can result in:

- Slower reaction times
- Lack of concentration – errors in calculating speed and distance are common
- Reduced vigilance and poor judgment
- Nodding off – even for a few seconds can result in dire consequences

There are obvious signs that suggest a driver is drowsy, such as:

- Frequently yawning.
- Inability to keep eyes open.
- Swaying the head forward (i.e. head nods).
- Face complexion changes due to blood flow.

There are various ways of avoiding driver drowsiness:

- Getting a good night's sleep before heading off on a long trip
- Don't travel for more than eight to ten hours a day
- Taking regular breaks – at least every two hours
- Sharing the driving wherever possible
- Don't drink alcohol before a trip. Even a small amount can significantly contribute to driver fatigue
- Don't travel at times when you'd usually be sleeping
- Taking a 15-minute power nap if drowsy feeling starts [12]

While these ways can avoid the helping driver drowsiness we need to make sure that drivers are awake on the road. In this project, we will design a system that checks a driver's facial behaviour, mainly eyes and mouth, in real-time to be able to detect drowsiness status of the driver. In order to achieve this goal, firstly we will detect the driver's face and extract the facial features. After extracting these features, we will train a model with drowsy non-drowsy test data. By using this model the application will decide if the driver is drowsy or not.

3. Aims of the Project

- **Detecting Driver Drowsiness with High Accuracy**

The main aim of this project is to detect driver drowsiness. The system will be able to detect driver drowsiness with high accuracy with only using the facial signs and a low-cost camera. This project would be able to attain at least a 65% accuracy rate which is higher by 3.6% as compared to the state-of-art results on the **UTA-RLDD** dataset [13].

- **Early Detection**

The other essential aim for this project is predicting driver drowsiness before the driver falls asleep. Drivers always have to be warned before they fall asleep while driving. This project will be able to extract some facial features that are evaluated from a certain set of landmarks on the human face. Then these features will be interpreted to be able to say if the driver is going to fall asleep. So reading signals as a time-series and predicting at least a couple of seconds forward to prevent possible accidents, is one of the main goals of the project.

- **Real-Time Performance**

Another important and crucial aim for this project is working in real-time. The system will always be working until it is closed manually. During it's working time, this project aims to work fast enough to calculate the necessary features and interpret them in real-time. The maximum delay for this system must be 2 seconds for calculating the drowsiness.

- **Adaptivity to the Subject**

The final aim of this project is to build a system that will be able to detect the doziness of all people from different ethnicities and personal characteristics. Since this project will be detecting drowsiness based on the facial features of the human face, it will be able to adjust itself and give accurate results with different skin color, eye shape, etc. The project will be subjective for each person by using their facial features' standards. In order to have a consistent system, accuracy levels should not differ much for different racial characteristics.

4. Related Work

Since the early 2000s, the automobile industry has spent a huge amount of time and resources with the collaboration of researchers from computer vision and machine learning domains to build a proper **DDD** (Driver Drowsiness Detection) system [14, 15, 16, 17, 18, 19]. In this section we will hierarchically summarize the literature's position on the DDD systems on different topics.

4.1 Drowsiness Detection Methods

There are various ways to determine the drowsiness level:

- **Physiological methods:** In these methods, systems are detecting drowsiness with collecting data through various electrophysiological sensors like Electrocardiogram (ECG) to listen electrical activity of human heart [20], Electroencephalogram (EEG) to record the activity of human brain [21] and Electrooculogram (EOG) to observe human eye [22]. They are reliable and precise methods but placing sensors on the driver's body and collecting data intrusively reduces the driver's comfort.
- **Vehicle-based methods:** These methods include observing steering wheel movements [23] and lane deviation [14]. They are non-invasive methods but highly correlated with driver's skill so they are subjective methods.
- **Behavioral methods:** This category observes behavioral movements of drivers with cameras and predicts his drowsiness level. Since it is non-intrusive and produces high accuracies due to the novel developments in machine learning and computer vision domains, their popularity increases among researchers and commercial solutions. Some of the behavioral signals that may give a cue on driver's fatigue are: Head

position [24, 8], yawning [25], blinks [13], or other facial actions like state of eyebrow, lip or jaw [26].

4.2 Data Collecting Methods

There are also several options when collecting data depending on which type of sensors or cameras are used in drowsiness systems. It could be near-infrared cameras (NIR) [27, 28] or visible light cameras with either CCD (Charge-Coupled Device) or CMOS (Complementary Metal Oxide Semiconductor) sensors [13]. NIR cameras are useful with night-driving scenarios since they are very effective in detecting face and eyes but at the same time, they are not cheap or easily accessible. So, most of the open datasets are collected with visible light cameras.

4.3 Properties of Open Datasets

Other than collecting device differences as mentioned above, there also variety on the content of datasets in the manner of subjects and environment. **NTHU-DDD** [29] is one of the widely used datasets in the literature and offers IR videos along with normal videos of subjects. But one drawback is subjects are pretended to be drowsy in a lab environment.

Another common dataset is **DROZY** [30] and unlike the previous one, it observes realistic behaviors by using EEG-EOG signals and NIR images along with normal cameras. Due to its complicated data collecting process, all data is in a lab environment with the same camera and angle, also consists of a low number of subjects (14).

On the other hand, newly introduced dataset **UTA-RLDD** [31] is the largest, realistic dataset to the 2019 and despite its environment being at home, it offers useful features like different cameras, subjects with several ethnicities, usage of accessories, to the researchers who work with machine learning-deep learning models on video datasets. We are also planning to work on this dataset with our drowsiness detection system, also completing missing parts with **NTHU-DDD** when necessary.

4.4 Feature Detection Methods

Generally, in computer vision systems object detection is based on methods extracting features from pixel data with different techniques. The same concept also goes in drowsiness detection systems, first it's necessary to detect driver's face, then detect facial members to produce features by interpreting them in algebraic and algorithmic processes. Some of the widely used feature extraction methods are,

- **Light Intensity:** In greyscale face images, the eye region is generally darker then nose and cheeks, due to height difference. It's possible to represent these regions with white and black rectangles which is called **Haar-like features** [32]. The most famous algorithm that uses Haar-like features is the **Viola-Jones algorithm** [32, 25]. It basically determines the most significant Haar features using **Cascading**

AdaBoost (Adaptive Boosting) [33] and works with these features when detecting faces (see **Figure-1**).

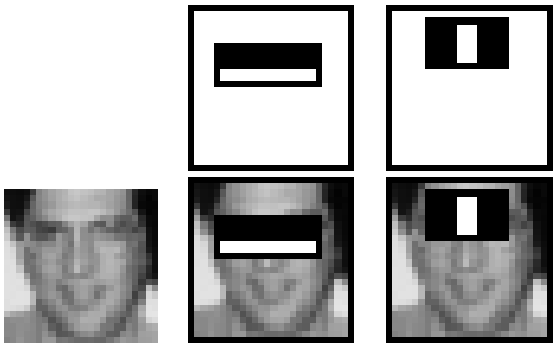


Figure-1: *An original figure from the 2001 Viola-Jones paper [32]. Detecting eyes and nose with two significant Haar-like features.*

- **Skin Color:** Another method for face detection is using a range for color tones of human skin [21, 25] in color spaces like RGB or YCbCr.
- **Texture:** It's also possible to detect faces by using regional color intensities. For example, eye regions have more fine-grained texture than cheeks. To capture this kind of micro-textural patterns, a simple but very efficient method, LBP (Local Binary Patterns) [35] can be used [36]. In this method, all pixels are replaced with a binary code which represents the magnitude of the cells in the 3x3 neighborhood by using the center as a threshold (see **Figure-2**). So, this way facial micro-patterns are found and used in the detection of faces, with the help of LBP histograms.

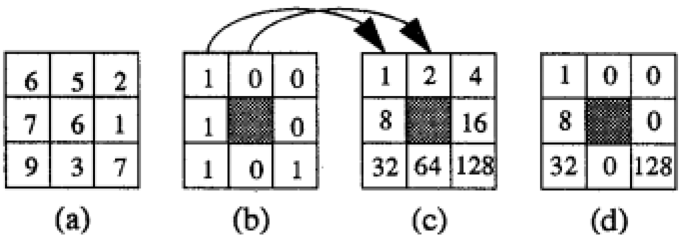


Figure-2: *An original figure from the 1994 LBP paper [35]. It uses the center as a threshold, marks the neighborhood and converts the whole square to binary code.*

- **Eigenfaces**: If a face image is distorted, it will not completely random, there will be some patterns in changes. These changes may represent facial members or distances between them. So, they are called **eigenfaces** in computer vision and the most famous example of it is **2D Gabor Function** which is used in enhancing the topological structure of the human face, i.e. nose as peaks, eyes as a valley. As it can be seen in **Figure-3**, when a face image convolved by a suitable **Gabor Filter,** which is basically a sinusoidal plane wave, facial characteristics may become more emphasized. Hence, these distorted images can be used in face detection [37].

Figure-3: *An original figure from the 2014 book "DDD Systems and Solutions" [8] showing eigenfaces.*

- **Infrared Sensitivity**: One unique feature of the eye's pupil is reflecting IR light while the rest of the face members is absorbing it. Some methods exploiting this feature are using IR sensors and detect pupils with Circular Hough Transform [34].
- **Horizontal and Vertical summation:** Summing values of rows and columns of greyscale images to determine local minima and maxima, then using this information to describe objects is another possible way to detect faces [38].
- **Feature descriptors:** There some algorithms in computer vision like **SIFT** (Scalar-Invariant Feature Transform) [39], **SURF** (Speeded-Up Robust Features) [39], **BRIEF** (Binary Robust Independent Elementary Features) [40], **ORB** (Oriented FAST and Rotated BRIEF) [40] to describe local features in images which are invariant to scaling, orientation, distortion and illumination. They are similar to each other in a way of using **DoG** (Difference of Gaussian) to find key points then describe their importance with different heuristics. Since they are complex and computationally expensive algorithms, it's hard to use them in real-time scenarios. Still, there are some works to use them in facial recognition and Drowsiness Detection domains due to their high accuracies [34, 41, 42]. There's also another prominent feature descriptor **HOG** (Histogram of Oriented Gradient) [43] using the distribution of directions of gradients as features (**see Figure-4**). It's one of the fast and robust

approaches in computer vision domain [25] so that a famous library **Dlib** uses an implementation of it to detect faces with a linear classifier [44].

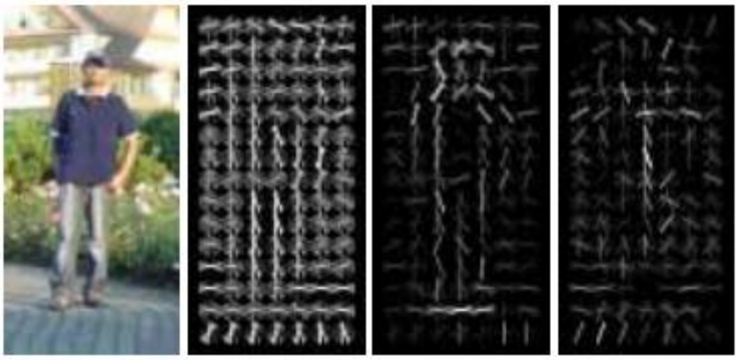


Figure-4: *An original figure from the 2005 HOG paper [43] showing orientations of gradients.*

4.5 Feature Extraction Methods

After the detection of face and facial members, it's necessary to produce some meaningful numerical values to predict the drowsiness of the subject. Some methods that can be used for it are listed below.

- **Eye Closure Analysis:** There are some metrics that can be used to determine eye state, which are **PERCLOS** (Percentage of Eye Closure) [45, 46, 59] and **EAR** (Eye Aspect Ratio) [47]. Since they are also used in our project, they will be explained in their own sections **7.2** and **7.3** in detail.
- **Yawning Analysis:** Yawning is one of the first indicators that comes to mind when detecting drowsiness, so some metrics which are tracking mouth openness are used in literature. One example of them is **MAR** (Mouth Aspect Ratio) [25, 48, 49] and it will be explained in the **section**-**7.3** in detail.
- **Facial Expression Analysis:** It's also possible to use some other facial features like wrinkles by detecting them with **Laplacian Filter** [26].
- **Head Position Analysis:** Another algorithm, **POSIT** is used in drowsiness detection [50] in the literature along with some other **Head Pose Estimation** techniques [24, 25].

4.6 Classifying Methods

After extracting features from raw data and constructing training datasets, there are also many choices of classifiers to predict the drowsiness level. Some of them are listed below:

- **Basic Thresholding:** It's rare to see using thresholds for drowsiness detection like [42] but there are some other usages of thresholds in blink detection [47] or nodding detection [8].
- **Conventional Machine Learning Tools**: While it's possible to use simpler approaches like **Logistic Regression**, **Decision Tree**, **k-NN** (K-Nearest Neighbors), **NB** (Naïve-Bayes) and get satisfactory results [26, 35, 48, 49], most researchers

likely prefer more complex tools i.e. **SVM** (Support Vector Machine) [22, 25, 35, 8, 41, 46, 47, 48], **Random Forest** [25, 35, 49], **HMM** (Hidden Markov Model) [3, 26, 27, 37, 50], **AdaBoost** (Adaptive Boosting) [32, 37, 8] and even **XGBoost** (Extreme Gradient Boosting) [35, 49]. Amongst them, the most popular ones are **SVM** and **HMM**.

SVMs are based on the concept of using hyperplanes with maximum margins as decision boundaries between different classes (see **Figure-5**). Other than using linear decision boundaries, it's also possible to use non-linear ones with the help of mapping functions that are called **Kernels. SVMs** are powerful and easy to understand tools and used in all kinds of research topics since the day they were first introduced by Vapnik et al. in 1992 [51].

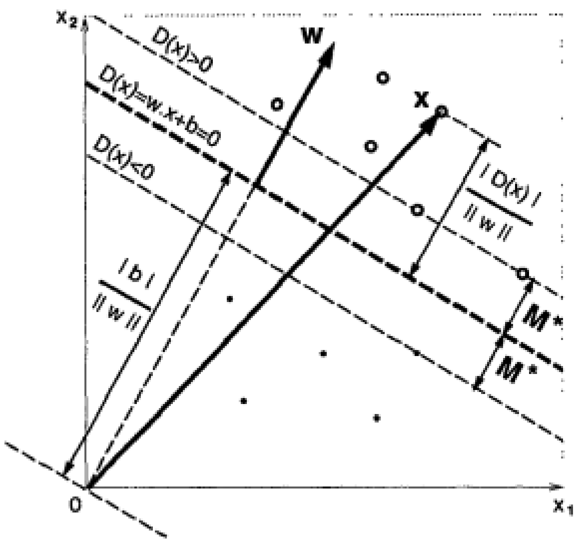


Figure-5: *An original figure from the 1992 SVM paper [51] showing decision boundary with maximum margins between two classes.*

On the other side **HMMs** are probabilistic models, which makes predictions from observable states to hidden states. They are firstly introduced in the 60's by Leonard Baum and colleagues [52] and still remains popular due to its availability of using on sequential data. In the drowsiness detection case (temporal data), **HMM** predicts the drowsiness level of future frame of the video (hidden state) from previous frames (observable states) with the help of extracted features like links, yawning, nodding, etc. (see **Figure-6**).

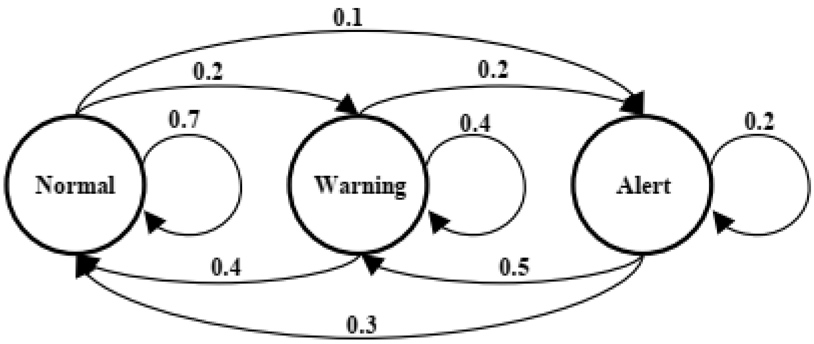


Figure-6: *An original figure from the 2016 paper [50] showing probabilistic transitions between three possible states: Normal, Warning, Alert*

- **Deep Learning Tools: ANNs** (Artificial Neural Networks) are mathematical models that consist of connected artificial neurons, inspired by biological neural network with the connectionist approach. They are used for modeling complex relationships and patterns between input and output data by using learnable weights of connections. **ANNs** with increasing complexity and multiple intermediate layers are called **DNNs** (Deep Neural Networks). Even though the first prototypes were seen at the 50's (the invention of perceptron by Rosenblatt in 1958) [53], **DNNs** had to wait until 2010s to gain their current popularity, due to their data-hungry, computationally expensive nature.

  In the drowsiness detection domain, although some papers use vanilla neural networks [20, 42], in most of cases researches prefer a subtype of **DNNs** which are called **CNN** (Convolutional Neural Network) as classifier [46, 49] almost as common as **SVM** and **HMM**. In addition to normal layers of **DNNs**, **CNNs** also consist of some convolution layers which are convolve input image with filters to assign importance to some aspects/objects of the image in order to reduce computational cost (see **Figure-7**). Since their first usage in computer-vision [54] they become widely popular and produced some of the highest state-of-art results [55].

  Another kind of **DNN** is called **RNN** (Recurrent Neural Network) which uses the concept of directed graphs to emphasize relations between input data, so that way it opens the possibility of using sequential data just like **HMMs**. One particular type of RNN which is called **LSTM** (Long-Short Term Memory), introduces long term memory function to already existing short-term memory behavior of **RNN** architecture [56]. **LSTM** models are well suited dynamic classifiers to sequential data (especially to time series) thus they also have promising future in drowsiness detection systems like **HMMs**, although its usage recently started with just a few examples [13, 49].

  And lastly, one more method worth to mention is **Transfer Learning**. This is basically transferring the knowledge gained from one task to another. **Transfer**

Learning concept is mostly useful when the first task has large and enough data, yet second has limited one. In drowsiness detection, it's possible to use a pre-trained model which is highly successful on one large dataset (i.e. **VGG16** model [57] on ImageNet data), on another dataset of different but related task (detecting drowsiness on a video) with some fine-tuning techniques [49, 58].

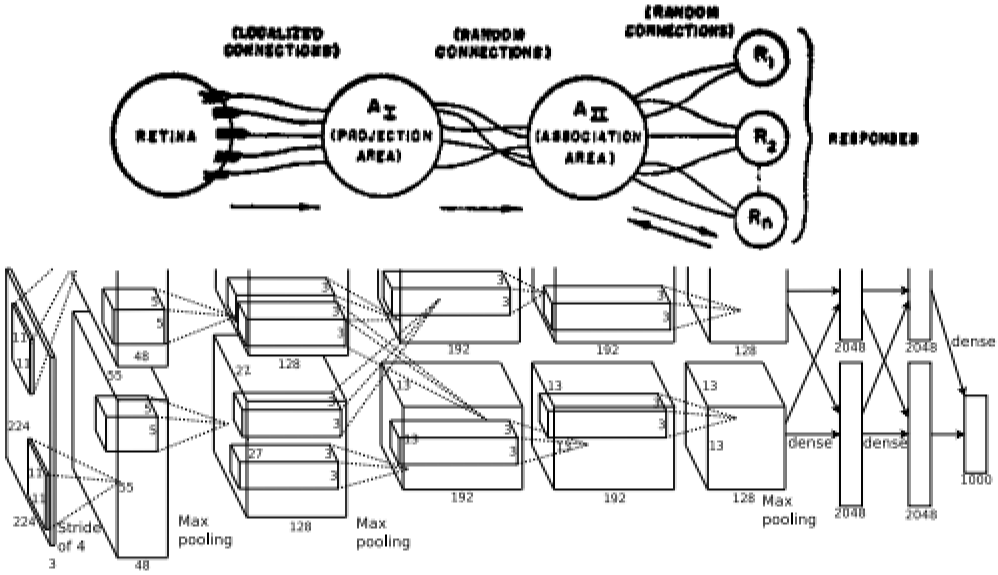


Figure-7: *First is the original figure from the 1958 Perceptron paper [53] and second is ImageNet2010 contest winner AlexNet, a CNN implementation [55].*

4.7 State-of-art Results

The primary challenge in DDD literature is each of researches using different datasets [5] and the absence of a standard, large and realistic datasets that can be used as benchmarks [13]. There is an example of the effort in 2017, about comparing different works on different datasets by using meta-analysis approach (see **Figure-8**) which indicates CNN as a most successful classifier against **SVM** and **HMM** [5]. Yet, it's still not enough to predicate state-of-art results because of the reasons explained in the **section-4.3.** To repeat them briefly, some of the databases are not open to public access and open ones are mostly consist of actor subjects or non-realistic environments and these are the main reasons shadowing accountability of works in the literature.

But recently there is one novel work by Ghoddoosian et al. [13], which aims to fill the gap of benchmark datasets and introduces **UTA-RLDD** dataset (also depicted in **section-4.3**) with a baseline method on it. They propose an LSTM-HM model and declare their results on their own dataset as **62.5%** where human judgement is **57.8%**. Hence, these can be used for future researchers to compare their works (see **Figure-9 and 10**).

We are also planning to work on this dataset so results above is selected as state-of-art because it's a novel work and there's no official research using the database yet rather than a blog post [49]. Additionally, surpassing it determined as one of the aims of the project since the level of goal is not so much high unlike the other datasets in the literature.

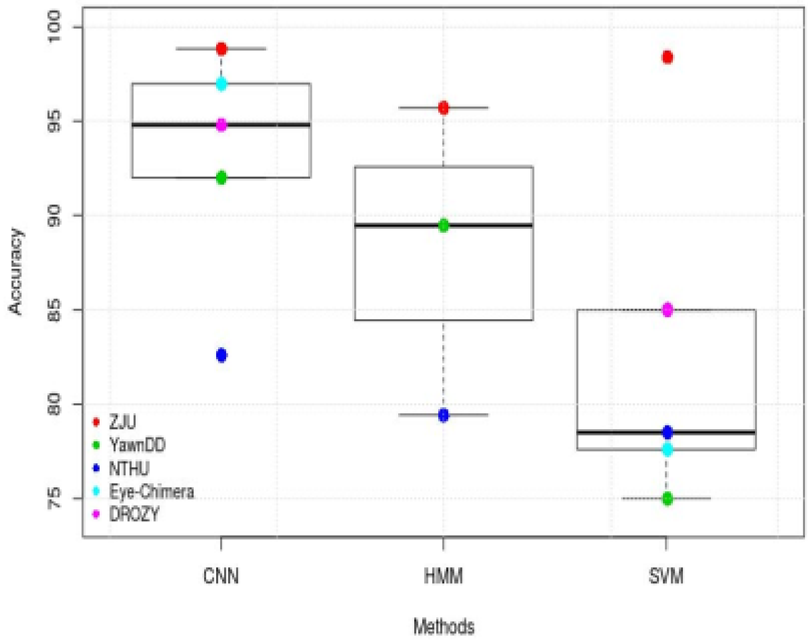


Figure-8: *An original figure from the 2017 survey paper [5] showing boxplots for comparing ML methods on the most common datasets by the date.*

Model	**Evaluation Metric**			
	BSRE	VRE	BSA	VA
HM-LSTM network	**1.90**	**1.14**	**54%**	**65.2%**
LSTM network	3.42	2.68	52.8%	61.4%
Fully connected layers	2.85	2.17	52%	57%
Human judgment	—	2.01	—	57.8%

Figure-9: *An original figure from the 2019 LSTM paper [13] showing results of Video Accuracy (VA) and some other proposed evaluation metrics.*

(a)

Prediction \ Labels	Alert	Low Vigilant	Drowsy
Alert	0.81	0.12	0.05
Low Vigilant	0.18	0.32	0.13
Drowsy	0.01	0.56	0.82

(b)

Prediction \ Labels	Alert	Low Vigilant	Drowsy
Alert	0.63	0.25	0.09
Low Vigilant	0.33	0.45	0.26
Drowsy	0.04	0.30	0.65

Figure-10: *An original figure from the 2019 LSTM paper [13] showing confusion matrix of LSTM model (a) and human judgement line (b).*

4.8 Selected Works

There are five research papers having significant importance in the project so, they are explained briefly in this section.

4.7.1 Driver Drowsiness Detection through HMM based Dynamic Modeling (Tadesse et al., 2014) [37]

This paper investigates the driver drowsiness detection using facial expression recognition for single frame based analysis and **HMM** based dynamic modeling. Firstly they collected data with the Logitech G27 Racing Wheel and webcam in front of the computer while using driving software simulator Simuride. They made two users drive indifferent scenarios in both drowsy and non-drowsy conditions and collected training and testing images for the analysis and evaluation of their approach. Then to detect driver drowsiness firstly they applied **Viola-Jones** and Camshift algorithms in order to detect the driver’s face. After face detection they extracted the facial features by using **Gabor Wavelets**. Then they used **Adaboost** to select extracted features. These parts are exactly the same for both classification system. Then they implemented both single frame based analysis and **HMM based dynamic model**.

For single frame based analysis firstly they combined the weak classifiers working on each selected feature to get a strong classifier by using **Adaboost Cascaded Classifer**. After that, since this topic is a two-class problem(drowsy, non-drowsy) they fed the selected features to the **SVM** by using **kernel method** which proved to have raised in performance over the linear combination of the **Adaboost** weak classifier.

Finally they implemented **HMM** based dynamic modeling. By using **HMMs**, they captured the temporal information of the facial expressions of the driver which leads to more accurate classification results as compared to single frame based drowsiness detection.

In conclusion, a dynamic approach gives more performance than the single frame based analysis. Therefore this paper states that facial expressions are better recognized through the analysis of sequence of frames.

4.7.2 A Practical Driver Fatigue Detection Algorithm Based on Eye State (Liu et al., 2010) [59]

In this article authors aim to detect driver drowsiness by calculating **PERCLOS**. Firstly they collected the tested videos in the natural driving conditions with the active infrared camera fixed on the car dashboard, and the videos were taken under day and night with different drivers. Then in order to achieve detecting driver drowsiness firstly they detect face with **Viola-Jones** algorithm and they detect the eyes from detected face then they adopt mean shift algorithm in case the eye detection failure. After detecting the eyes, they adjusted the contrast of the eyes in order to remove the influence of bright spots or specular reflection caused by glasses or hard light. Next they used their filter to detect the eye corners. Finally they striped the area between right and left eye corners into five and applied a simple filter to each area. Maximum and minimum responses that are given to this filter are up and down eyelids respectively.

After extracting the eyelids they calculated the distance between eyelids. By looking at the distance of eyelids, eye closeness can be detected. Then they counted the number of consecutive frames in which the eye is closed to decide the condition of the driver, to classify him as awake or dozing. Eye closure lasts longer with the increasing level of driver dozing.

In conclusion the algorithm developed in this article is capable of detecting eye closure in high-speed after locating the eye. Although it is irrelevant to the subject's gender and day and night, glasses are a problem for this algorithm.

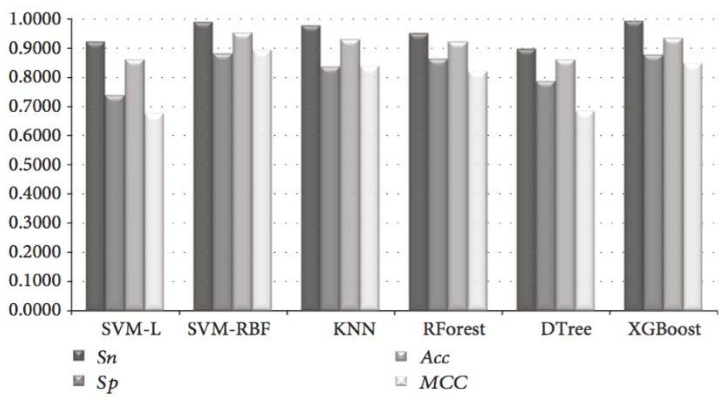
4.7.3 Accurate Fatigue Detection Based on Multiple Facial Morphological Features (Li et al., 2019) [36]

In this study, the researchers are focused on multiple physical features to get better accuracy in detecting drowsiness level. Beside that, many other studies were extracted only on a single feature. In this study LBP(Local Binary Pattern) is mainly used while extracting eye and mouth regions. **REDE** is the name of the proposed algorithm for this study. **REDE** is outperform previous four fatigue detection studies.

Since there is no publicly available database for fatigue detection they have created their own database captured from volunteer participants. Database is consisting of seven male and seven females. Basically this database is built on to reflect the real life situation because participants had recorded their videos according to specification. As explained in the paper, such method is used when collecting database: "A participant took a normal diet and a full rest for the first day and had one video recording at 8:00 AM of the next day in a rest state. After this time, the participant had no rest for 18 hours, and another video recording was taken at 3:00 AM of the third day in a rest state."

When prepocesing data, extracting the regions of two eyes and mouth from each image is done by **Dlib**. Then **LBP** values are extracted for both eyes and mouth and after that eigenvector of the **LBP** features.

Sensitivity, specificity, accuracy and Matthew's correlation coefficient (**MCC**) is widely used to demonstrate how well the binary classification model performs. Here we can see the comparison of different techniques in **Figure-11**.



<u>Figure-11:</u> *An original figure from the 2019 LBP paper [36] showing four evaluation metrics with different classifiers.*

4.7.4 A Realistic Dataset and Baseline Temporal Model for Early Drowsiness Detection (Ghoddoosian et al., 2019) [13]

Main motivation of the paper is collecting largest, public, realistic dataset to 2019: **UTA-RLDD.** Additionally, introducing a baseline method to detect early, subtle cues which uses Hierarchical Multiscale Recurrent Neural Networks, specifically **HM-LSTM**, resulting in a higher accuracy (65%) than human observers (under 60%).

The Proposed Baseline Method uses **Dlib's face detector** (HoG version) and detecting blinks by an improved algorithm of [47] by specializing for consecutive quick blinks. Input of the blink detection module is frames of last minutes of real time video and output of the blink detection module is a **sequence of blink events**: {$blink_1$, …, $blink_k$} where each $blink_i$ is a **4D vector** [duration, amplitude, eye opening velocity, frequency]. After preprocessing and normalization, model continues with a 4-D feature transformation layer with following **HM-LSTM** layer and finally four Fully Connected Layers.

Consequently, the paper represents publicly available, real-life dataset RLDD which is the largest to date (2019) along with end-to-end baseline method using the temporal relationships between blinks to detect early signs before an accident. Overall, the paper hopes that the proposed public dataset will also encourage other researchers to work on drowsiness detection and produce additional and improved results, that can be duplicated and compared to each other.

5. Scope of the Project

The main aim of this project is to build a system that detects the drowsiness of the driver and gives warning to the driver in real-time using image processing and machine learning techniques in order to minimize the traffic accidents due to fatigue. The project consists of three phases. In the initial phase, the development of the blink detection method and generating blink detection features are included. In the second phase, development for frame-based feature extraction with normalization is residing. In the final phase, determining the appropriate classifier method which includes classical machine learning techniques like **SVM**, **k-NN**, HMM and deep learning brand new techniques such as **CNN**, **LSTM**.

The feature extraction phase is proceeded with using two different techniques; Mustafa will be implementing the Blink-based Model and Ayşenur and Mahmut will be implementing feature extraction by frame-based.

In this project, **UTA Real Life Drowsiness Dataset** [31] and **NTHU-DDD** [29] is used for both training and test data. Also, blinks methods are developed mainly based on **Eyeblink8** and **TalkingFace** datasets from **blinkmatters.com** [60].

Since this work is research-based project, developing any mobile/web application or commercial software is not taken into consideration.

5.1 Constraints

In this project, there are some constraints depending on both the dataset and the technical methods. The following cases are out of the scope of this project:

If there is

- Insufficient illumination of the face according to time of the day,
- If the driver is wearing sunglasses or a hat and may have facial hair,
- There might be obstacles in front of the driver's eyes e.g. the driver's hand.

5.2 Assumptions

There are some assumptions that must be taken into consideration in order to make our system to work with a desired result, such as:

- It is assumed that all datasets that are used in this project are correctly annotated.
- The images are collected with sufficient illumination of the face.
- It is assumed that the camera is directly placed in front of the driver and nearly has one arm length distance.
- It is assumed that the driver's head is up and his face fits in the camera.
- It is assumed that annotations provided by datasets for testing are ground-truth annotations .

6. Success Factors and Benefits

6.1 Success Factors

There are four aims of the project denoted in **section-3.** First is detecting driver drowsiness in a higher accuracy. It's planned to use **accuracy metric** to check if results are reaching the baseline level on **UTA-RLDD dataset** (see **section-4.6 State-of-art Results**). Details of evaluation metrics are explained in the **section-7.5 Prediction and Evaluation Phase.**

Second aim is early detection of potential asleep of driver by using sequential methods like HMM and LSTM. Success metric for this aim is determined as how many seconds before the danger level can be detected and desirable result is 3 seconds for now. The goal will be revised after starting of experiments.

And third aim of the project is working in real-time performance. Despite its complicated workflow, the proposed system must not be delayed much than 2 seconds for a typical 10 minutes video of the **UTA-RLDD dataset**, since predictions of a potential accidents will be in a three seconds limit as explained above.

The last aim is adaptivity to the subject's personal and racial characteristics such as eye shape, skin color, blink behaviors etc. Success factor for it is consistency on accuracy levels across subjects in the videos. It's achievable with normalization phase (**see section-7.2 and 7.3**) and testing accuracy metrics on selected false-positive samples with different characteristics from the dataset (**see section 7.5**).

6.2 Benefits

With the success of this project, we might consider four benefits. First benefit of this project is driving safety. This project can help people drive safely without causing an accident due to drowsiness. Second benefit of this project is preventing economic loss. Since this project aims to reduce road crashes, the cost of these road crashes reduces also. Third benefit of this project is to lower the environmental impact of the accidents caused by road crashes. And final benefit would be offering improvements to commercial products by the help of increasing accuracy and provided features.

7. Methodology and Technical Approach

Before presenting the methodology of the pipeline, specifications of datasets are used in the project are listed below:

- **UTA-RLDD:** UTA Real Life Drowsiness Dataset [31] is used for both training and test data. It is created in the University of Texas at Arlington by a research group to detect multi-stage drowsiness. This dataset is obtained with the participation of 60 healthy people and three different videos of each participator is taken: alertness, low vigilance and drowsiness that is total of 180 videos. It is composed of around 30 hours RGB videos. There were 51 men and 9 women from different ethnicities like 10 Caucasian, 5 non-white Hispanic, 30 Indo Aryan and Dravidian, 8 Middle Eastern, and 7 East Asian and age ranges between 20 and 59 and there are 21 out of 180 participants were wearing glasses and 72 out of 180 participants had facial hair.

- **NTHU-DDD:** This dataset [29] is consisting 36 subjects of different ethnicities. It is included many variations of driving scenarios such as normal driving, yawning, show blink rate and falling asleep. The total time of videos almost 10 hours. There are 5 different scenarios; bare face, glasses, night bare face, night glasses and sunglasses. Each record is approximately 1 minute long. The participants are simulated the driving in lab environment. The evaluation and testing datasets contain 90 driving videos (from the other 18 subjects) with drowsy and non-drowsy status mixed under different scenarios. Main property of the dataset is usage of active IR illumination to acquire IR videos.

The whole pipeline of the project comprises four different phases as they can be seen in **Figure-2**. **Preprocessing Phase** to reading and processing videos, **Feature Extraction Phase** to construct ad-hoc features (there will be 2 different paths which are explained in **section-**

7.2 and 7.3), **Classification Phase** to train our input data and finally **Evaluation Phase** to make predictions on test data and to calculate accuracy metrics.

7.1 Preprocessing Phase

Implementation of the project starts with **Preprocessing Phase** which includes steps;

1. **Reading video frames**, either can be done from a dataset or a camera in a real-time manner. For this step, **opencv-python** [61], python version of OpenCV library, is used.

2. **Detecting faces** with **Dlib's get_frontal_face_detector** [62] method. It uses a pre-trained "Histogram of Oriented Gradients + Linear SVM" pipeline for face detection. There's also another CNN-based method in Dlib but it isn't suitable for real-time purposes.

3. **Predicting facial landmarks** with **Dlib's shape_predictor** [63] method which is an implementation of the paper Kazemi and Sullivan (2014) [64]. This method also uses a pre-trained model of ensemble of regression trees and predicts 68 facial landmarks which can be seen in **Figure-12.** All features will be used in later phases, are extracted from these positional landmarks.

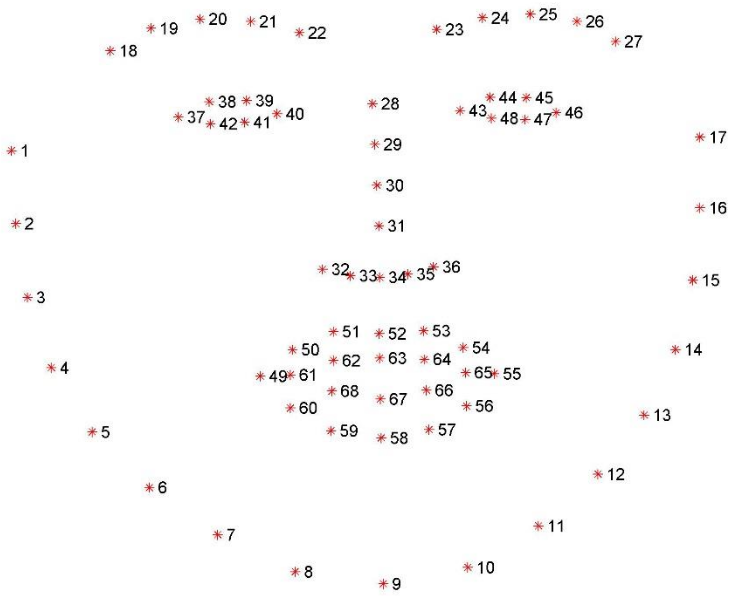


Figure-12: *68 facial landmark coordinates of Dlib's shape_predictor method.*

After the preprocessing phase, implementation continues with the **Feature Extraction** phase. Since there are two models to experience on, as shown in **Figure-13**, there will be different features for both.

- **Blink-based Model**: Detects blinks, extracts blink-related features and classifies video sequences based on these features.

- **Frame-based Model:** Without detecting any facial action (blink, yawning etc.), all features are computed for every frame and all of this information will be used for classification.

  So, in that context, both models will be explained separately.

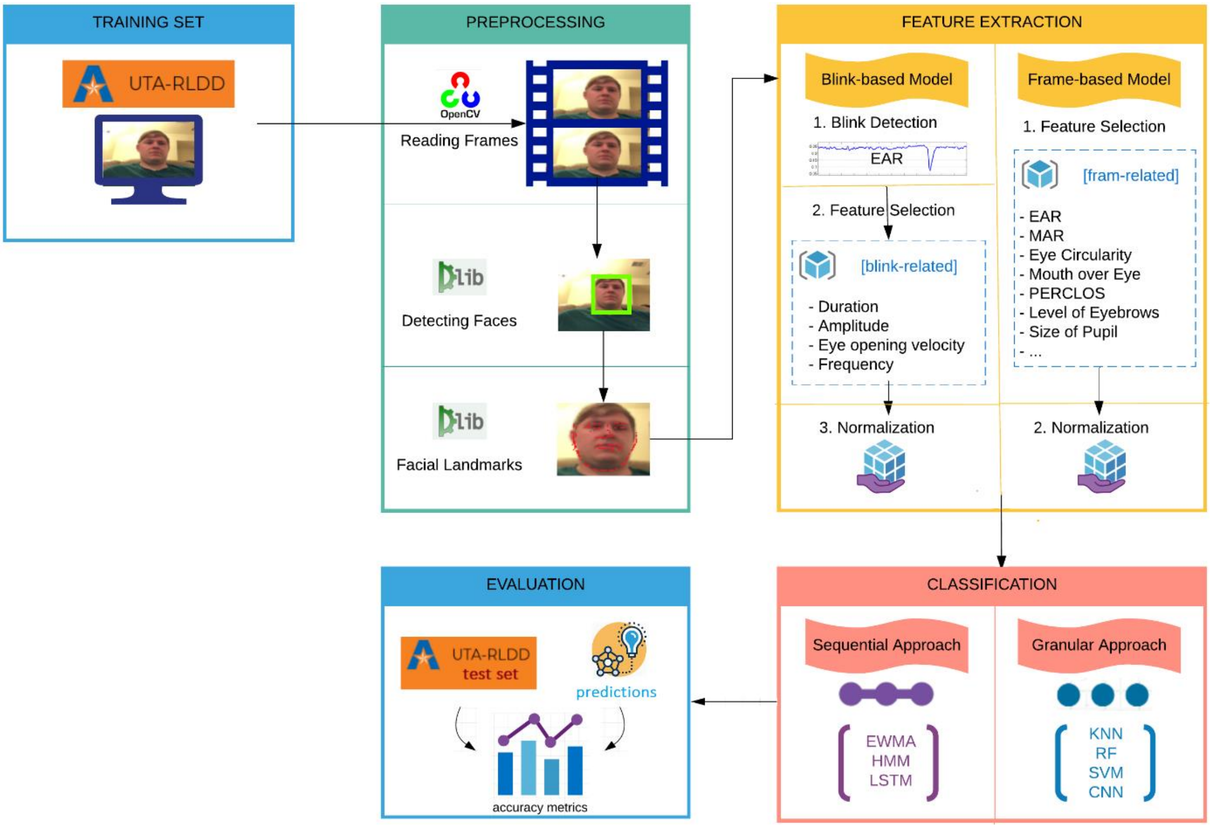


Figure-13: *Conceptual framework of the project*

7.2 Feature Extraction Phase of Blink-based Model

For further detail, since **Blink-based Model** is just an implementation of the work by Ghoddoosian et al (2019) [13], it starts with blink detection, then continues calculating blink-related features.

1. **Blink detection step** is also an implementation of another paper Soukupová and Chech (2016) [47] and investigates of using a simple mathematical formula for real-time purposes which is called "Eye Aspect Ratio (EAR)" (7.1) and can be extracted from eye landmark coordinates in **Figure-12**.

$$EAR(i) = \frac{||p_{38} - p_{42}|| + ||p_{39} - p_{41}||}{2||p_{37} - p_{40}||}, \qquad (7.1)$$

In the formula here, p_{37}, …, p_{42} are 2D landmark locations of the left eye depicted in **Figure-12 and Figure-14** and i is the frame index. $\|p_a - p_b\|$ represents the Euclidian distance between two landmark positions.

After calculation of EAR for both eyes, average of them is calculated for a window of n = 13 frames (n is determined empirically) and these sequences are used for detecting blinks with SVM classifier.

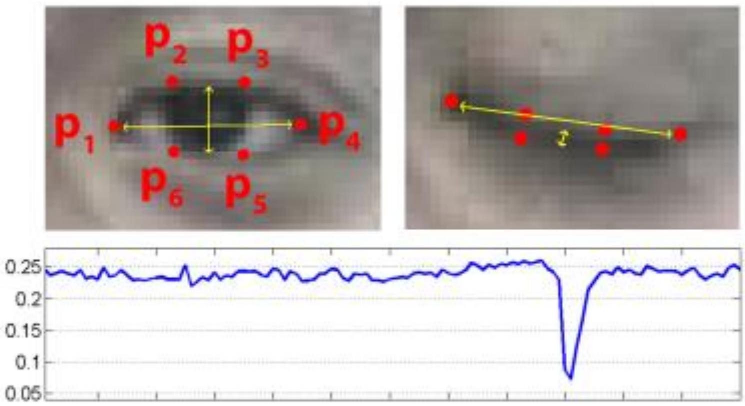


***Figure-14**: An original figure from the 2016 EAR paper []. Eye landmarks are used in the calculation of EAR with open/closed eye scenarios.*

2. **Selection of blink features:** Output of blink detection step is a sequence of blink events $\{blink_1, \ldots blink_k\}$, each $blink_i$is a 4D-vector consisting relevant EAR(i) value along with $start_i$, $bottom_i$ and end_i values which are timestamps (frame count starting from the beginning of the video) of start, bottom and endpoints of a blink (see **Figure-15**). As depicted in [13] 4 blink-related features are calculated from these 4D vectors which are;

- **Duration** measures the duration of a blink, which can be formulated as

$$Duration_i = end_i - start_i + 1, \qquad (7.2)$$

In the formula (7.2), i represents index, end_i represents timestamp (=frame count starting from the beginning of the video) of ending frame and $start_i$ represents the timestamp of starting frame of a blink.

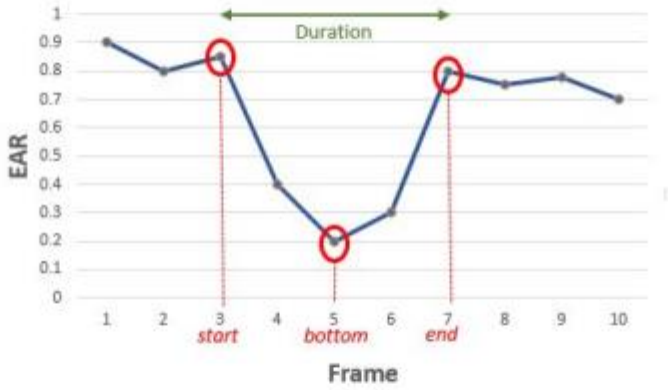


Figure-15: *An original figure from the RLDD paper [13], shows starting, ending and bottom frames of a detected blink.*

- **Amplitude:** Average amplitude of EAR values of starting and ending frames.

$$Amplitude_i = \frac{EAR(start_i) - 2EAR(bottom_i) + EAR(end_i)}{2}, \qquad (7.3)$$

In the formula (7.3), i represents index, end_i represents timestamp (=frame count starting from the beginning of the video) of the ending frame, $start_i$ represents timestamp of the starting frame and $bottom_i$ represents timestamp of the bottom frame of a blink. Also $EAR(i)$ represents EAR value of ith frame which is calculated with the formula (7.1).

- **Eye Opening Velocity** measures how fast a subject opens his eyes. This can be a good feature to indicate the drowsiness level.

$$Eye\ Opening\ Velocity_i = \frac{EAR(end_i) - EAR(bottom_i)}{end_i - bottom_i}, \qquad (7.4)$$

In the formula above (7.4), i represents index, end_i represents the timestamp (=frame count starting from the beginning of the video) of the ending frame and $bottom_i$ represents the timestamp of the bottom frame of a blink. Also $EAR(i)$ represents EAR value of ith frame which is calculated with the formula (7.1).

- **Frequency:** Indicates the frequency of previously occurring blinks up to a blink.

$$Frequency_i = 100 \times \frac{number\ of\ blinks\ up\ to\ blink_i}{number\ of\ frames\ up\ to\ end_i}, \qquad (7.5)$$

In the formula above (7.5), i represents index, $blink_i$ represents ith blink, end_i represents timestamp (= frame count starting from the beginning of the video) of the $blink_i$.

After calculation of 4D vectors of each $blink_i$ in the sequence $\{blink_1$, ... $blink_k\}$, normalization phase starts.

3. **Normalization step:** Each feature calculated above needs to be normalized across subjects in videos because blinking behaviors change person to person. There could be a subject with slant eyes whose EAR values show different characteristics comparing to other subjects. So subject-wise normalization is essential to get more accurate results.

   On the other hand, normalization across features also had to be done. We are planning to use the first third of blinks detected in an alert video of a specific subject to normalize features. So, this process can be formulated as,

$$normalized\ feature_{n,m} = \frac{feature_{n,m} - \mu_{n,m}}{\sigma_{n,m}}, \qquad (7.6)$$

   Where, $\mu_{n,m}$ and $\sigma_{n,m}$ are mean and standard deviation of the feature n of the subject m.

7.3 Feature Extraction Phase of Frame-based Model

Unlike the Blink-based Model, the Frame-based Model doesn't include blink detection step. All of the features that presented below, are calculated for every frame in the video and will be hand into classifier. So, this way model doesn't discard any data that can be valuable in the classification phase.

1. **Feature selection step:** This model uses 3 features defined in an online paper by Grant Zhong [49].
   - **Eye aspect ratio (EAR):** Just like the Blink-based Model, the eye aspect ratio of all frames is calculated with the formula (7.1). The average of both eyes will be selected as a feature.

   - **Mouth aspect ratio (MAR):** This formula resembles to EAR (7.1), in the context of using 68 facial landmarks (see **Figure-12**). It uses inner landmarks of the mouth (61, ..., 68) and calculates a ratio just like EAR. Therefore, it can be useful for detecting yawning behavior [25].

$$MAR(i) = \frac{\|p_{63} - p_{67}\|}{\|p_{61} - p_{65}\|}, \qquad (7.7)$$

In the formula here (7.7), p_{61}, …, p_{67} are 2D landmark locations of the inner mouth shape depicted in **Figure-16** and i is the frame index. $\|p_a - p_b\|$ represents the Euclidian distance between two landmark positions.

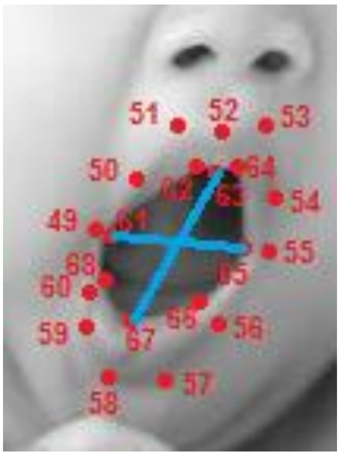


Figure-16: *Ratio between [p63, p67] and [p61, p65] to measure Mouth Aspect Ratio (MAR).*

- **Eye Circularity (EC):** It's a measure like EAR but it puts greater emphasis on the pupil area. [49]

$$EC(i) = \frac{4 \times \pi \times Pupil\ Area}{(Eye\ Perimeter)^2}, \qquad (7.8)$$

$$Pupil\ Area = \left(\frac{\|p_{38} - p_{41}\|}{2}\right)^2 \times \pi, \qquad (7.9)$$

$$Eye\ Perimeter = \|p_{37} - p_{38}\| + \|p_{38} - p_{39}\| + \|p_{39} - p_{40}\| + \|p_{40} - p_{41}\| + \|p_{41} - p_{42}\| + \|p_{42} - p_{37}\|, \quad (7.10)$$

In the formulas above (7.8, 7.9, 7.10) p_{37}, …, p_{42} are 2D landmark locations of the left eye shape depicted in **Figure-17** and i is the frame index. $\|p_a - p_b\|$ represents the Euclidian distance between two landmark positions. The average of both eyes will be selected as a feature.

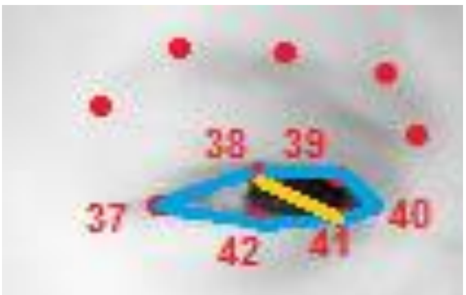


Figure-17: *Blue lines represent Eye Perimeter and the yellow line represents Pupil Diameter which is used in the calculation of Eye Circularity (EC).*

- **Mouth over Eye (MOE):** Basically EAR (7.1) over MAR (7.7). [49] It's an additional feature which can be interpreted as true drowsiness, since some facial actions like smiling and talking may produce some fake yawning MOE values.

$$EC(i) = \frac{MAR(i)}{EAR(i)}, \qquad (7.11)$$

In addition to these 4 features, we are planning to use 3 more features that can be also produced from 68 facial landmarks:

- **PERCLOS**: Indicates the frequency of closed eyes up until that moment. [45] We are planning to use it with a small-time window like t = 30 seconds.

$$PERCLOS = \frac{count\ of\ frames\ when\ the\ eyes\ are\ closed}{total\ count\ of\ frames\ up\ until\ that\ moment} \times 100\%, \quad (7.12)$$

- **Level of Eyebrows (LEB):** Level of eyebrows can be a good measure to drowsiness so another formula that calculates the average distance between first two of inner points of eyebrows and inner corner of an eye. Other points of eyes are ignored due to their moving nature. Other than two points of eyebrows are ignored also, since they are more stationary.

$$LEB(i) = \frac{\|p_{21} - p_{40}\| + \|p_{22} - p_{40}\|}{2}, \qquad (7.13)$$

In the formula above (7.13) p_{21} and p_{22} are the most inner points of the left eyebrow, also p_{40} the most inner point of the left eye and represents 2D locations as depicted in **Figure-18.** i is frame index and $\|p_a - p_b\|$ represents euclidian distance between two landmark positions. The average of both eyes will be selected as a feature.

Figure-18: *Blue lines represent distances [p21, p40] and [p22, p40]. Average of them is calculated to measure the Level of Eyebrows (LEB).*

- **Size of Pupil (SOP):** Size of pupil can be a good measure of alertness. It's not a direct relation but fluctuations of size are related to the fatigue of a subject [65]. So defined formula below measures the ratio of pupil diameter and eye width.

$$SOP(i) = \frac{\|p_{38} - p_{41}\|}{\|p_{37} - p_{40}\|}, \qquad (7.14)$$

In the formula above (7.14), $p_{37}, \ldots, p_{40}$ are 2D landmark locations of the left eye depicted in **Figure-19** and i is the frame index. $\|p_a - p_b\|$ represents the Euclidian distance between two landmark positions. The average of both eyes will be selected as a feature.

Figure-19: *Blue line represent Eye Width [p37, p40] and orange line represents Pupil Diameter [p38, p41]. The ratio of them is called Size of Pupil (SOP).*

2. **Normalization step:** Just like the normalization step in the Blink-based Model, all of the features will be normalized across subjects and features itself by using the formula-7.6. But unlike Blink-based Model, Frame-based Model can use just a few frames of an alert subject instead of a huge number of blinks. So, this method is more suitable in real-time scenarios.

7.4 Classification Phase

After preprocessing and feature extraction, we are planning to try some classification techniques starting from conventional ones (Logistic Regression, Naïve-Bayes, K-Nearest Neighbor, Decision tree, Random Forest, SVM) to novel approaches (XGBoost, CNN, LSTM) with increasing complexity.

One of the important aims of the project is introducing an early warning mechanism to the system so sequential/temporal relations play a key role to achieve it. When conventional machine learning tools investigate each frame, they don't take sequential relation between frames into consideration, unlike some deep learning approaches like LSTM and HMM. Therefore, using EWMA (exponentially weighted moving average) on a small-time window like t = 3 seconds is one of the solutions which can be experimented on conventional machine learning tools.

$$EWMA(i) = w \times x(i) + (1 - w) \times EWMA(i-1), \qquad (7.15)$$

For the formula above (7.15), w is weight, i is frame index and $x(i)$ is a feature we want to soften like EAR, MAR, etc. when $x(i)$ represents the current value of a feature, $EWMA(i-1)$ represents past values. Thus $EWMA(i)$ calculates the recursive mean of a feature. $1/w$ approximately gives a number of past values to be considered and in that manner w will be chosen empirically.

7.5 Prediction and Evaluation Phase

By using the models trained in the previous phase and getting prediction results on the test set, it's possible to compare them with truth labels in two different manners:

- Predictions on the drowsiness level of the subject in a frame or
- Predictions on the drowsiness level of the subject in the whole video

While NTHU-DDD provides truth labels for both of them, UTA-RLDD provides truth labels for only the second approach. As explained in **section-7.4**, time-series tools like LSTM and HMM can be used on classifying whole videos. In addition to them, basic Machine Learning tools also can be used in the same way after softening the values of frames with EWMA (**Formula-7.15**).

Therefore, we are planning to use sequential tools (LSTM, HMM, EWMA) when video labels available (on UTA-RLDD and NTHU-DDD) and static/granular tools (other ML techniques) when frame labels available (only NTHU-DDD).

Officially recommended evaluation method for UTA-RLDD is "*Use one fold of the UTA-RLDD dataset as your test set and the remaining four folds for training. After repeating this process for each fold, the results would be averaged across the five folds.*" [31]. With this way, **RMSE** (Root Mean Square Error) will be used as evaluation metric.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}, \qquad (7.16)$$

Where $\hat{y}_i$ represents predicted results, y_i represents ground truth labels and n is size of a test fold.

There's also an evaluation script provided for NTHU-DDD which uses accuracy as an evaluation metric [29]. We will stick to official evaluation methods to compare the results of papers as baselines to ours.

8. Professional Considerations

8.1 Methodological Considerations and Engineering Standards

It's planned to use **Github** [65] as a version control system. Other than final codes, experimental notebooks will be also shared openly on **Kaggle Kernel** [66] platform. Division of responsibilities and milestones will be addressed with the **GANTT** chart (see **Figure-20**) and will be updated in every project document. We are also planning to use **Trello** [67] for complex tasks.

It has been decided to work on **Python** programming language due to its wide variety of libraries on computer vision and machine learning domains. As computer vision libraries, python forks of **OpenCV** [68], **Dlib** [69] and **OpenFace** [70] will be used. To implement machine learning and deep learning models, **Scikit-Learn** [71] and **Keras** [72] will be used.

8.2 Societal/Ethical considerations

8.2.1 Economical

Every year economic and societal impact of road crashes costs United States citizens $871 billion. $380 million of this money is directly spent on medical costs [6]. In light of these facts we can say that road crashes have a huge impact on the economy. This project aims to reduce car accidents caused by drowsy drivers. By avoiding this cause of road accidents, this project can reduce the accident's impact on the economy.

8.2.2 Health and Safety

The main purpose of the project is to avoid road crashes caused by drowsy drivers. Therefore, this project can save many lives by warning the sleepy drivers before an accident occurs. Since a road accident isn't always an individual incident, this project may also be preventing the damage to the other drivers or pedestrians on the road at the time of the accident. Lastly accidents can have mental effects on the victims' loved ones, so this mental damage would be prevented with this project.

8.2.3 Environmental

Road crashes not only affect health and economy, but they also affect the environment. Car accidents often cause gas and liquid leaks emitting harmful chemicals into the environment that can poison grass and neighboring plants and harm wildlife. The other damage environment takes after the accidents is landfills. When a car is totaled, most insurance companies determine it's more economical to replace the vehicle than repair it. Though many car parts can be recycled, most of the vehicles ends up in a landfill where it will take thousands of years for all the pieces to decompose. Many car parts are also left on the side of the road where they can harm animals or plants. The effects on soil, water, and air pollution influence the entire ecosystem [7].

8.3 Legal Considerations

Python's open-source libraries were used for the project. Likewise, we used open datasets such as NTHU-DDD and UTA-RLDD. Some of the participants (24 out of 60) in the UTA-RLDD don't allow their faces to be published in any future paper. A Complete list of them can be seen here [31]. Also, there is a license agreement for NTHU-DDD [29] to be signed for the usage of databases for research purposes.

In the case of the project proposed, although the system does not keep any personal data of subjects, users still need to allow the system to capture themselves while the system is on.

9. Management Plan

9.1 Task Planning and Milestones

Task Phase 1: Literature Survey

- Previous researches and publications are going to be scanned. Each team member is going to focus on a different topic and make a mini-presentation about it.
- The similarities and differences between previous projects are going to be investigated and the future path of the project is going to be determined.

Task Phase 2: Establishing Environments

- Necessary tools and technologies will be established to work with properly.

Task Phase 3: Implementing Blink Detection for Blink-based Model

- Building and evaluating an adaptive blink detector on available blink datasets in the literature.

Task Phase 4 and <u>Milestone 1</u>: Project Specification Document

- The Project Specification Document is going to be prepared.

Task Phase 5: Feature Extraction & Normalization for Frame-based Model

- The extraction of features is going to be available and after that be normalized to get better accuracy.
- In the feature extraction two different methods will be used. Blink based and frame-based detections are going to be held.

Task Phase 6 and <u>Milestone 2</u>: Analysis and Design Document

- The Analysis and Design Document is going to be prepared.

Task Phase 7: Comparing and Deciding which features will be used for classification

- Eliminating unnecessary features and generating fresh features.
- Searching for if there is a possibility to implement features of blink-based model frame-by-frame and if it does, then is it possible to make the system as a hybrid model.

Task Phase 8: Implementing Classification

- Both using classical Machine Learning techniques(SVM, k-NN, HMM) and deep learning techniques(CNN, LSTM)
- Searching for possibility of ensemble two models (blink-based and frame-based) in classification phase.

Task Phase 9: Testing the Project

- Testing the final project in order to see whether it gives the warning correctly and determining the evaluation metrics for comparing state-of-art results.

Task Phase 10: Real-Time Demonstration

- Constructing the real-time demonstration.

Task Phase 11 and Milestone 3: Project Report & Poster

- The project report and poster are going to be prepared.

9.2 Risk Management

There are some predictable risks related to mostly datasets and packages used in development of the project. Some solutions and alternatives which can be used to overcome the possible problems are shown in the **Table-1.**

Potential Risks	Solutions and Alternatives
One of the possible risks is the dataset we use in this project, namely RLDD may be insufficient in terms of diversity.	As a solution to this potential problem is to use other databases that are ready to use in our Google Drive.
Another risk that we would not want to face is about normalization. We will be doing normalization like this, take the first frames of the dataset and use these in the normalization, however when we try to do it in real-time, then how can we deal with this issue in terms of duration.	We will use first 1-2 seconds of the video for calibration.
We are going to use Dlib and it has 68 facial landmarks. It may not be sufficient if we need to generate more features.	In this case, we will use another library resources such as OpenFace to extract features like head position, gaze direction, etc.
The other risk that we may encounter is about face detection. For instance, Dlib face detector may not be able to detect the eyes under some physical conditions like in the lack of light.	At this point, for face detection we may want to use OpenFace.

Table-1: *Potential risks in development and their solutions*

9.3 Timeline

Time planning of tasks can be seen in **Figure-20** as GANTT chart. Dark green rows are for completed tasks, light green ones are for tasks under process and red ones are for incoming tasks. Stars represent milestones which are deadlines of project documents and presentations.

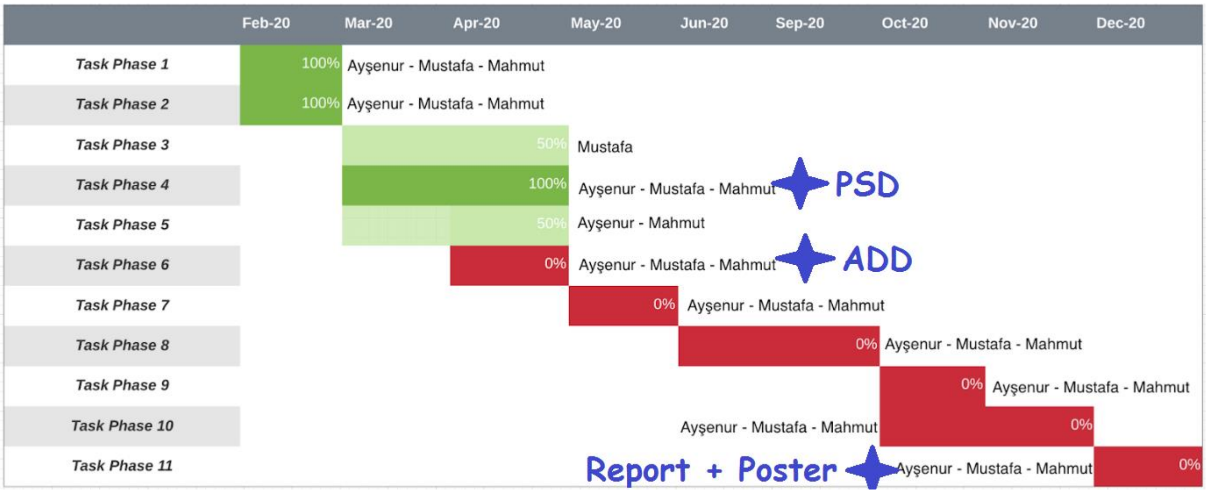


***__Figure-20:__** GANTT Chart for work management, timeline and milestones*

10. References

[1] Karayolları Genel Müdürlüğü, Trafik Kazaları Özeti. [Online]. Available: https://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Trafik/TrafikKazalariOzeti.aspx (Date of Access: 20 / 04 /2020)

[2] İ. Umut, O. Aki, E. Uçar, L. Öztürk, “Detection of Driver Sleepiness and Warning the Driver in Real-Time Using Image Processing and Machine Learning Techniques”, Advances in Science and Technology Research Journal, vol.11, no.2, pp.95-102, 2017.

[3] R. Fu, H. Wang, W. Zhao, “Dynamic Driver Fatigue Detection Using Hidden Markov Model in Real Driving Condition”, Expert Systems with Applications, vol.63, pp.397-411, 2016.

[4] Transport Accident Commission, Avoiding Driver Fatigue. [Online]. Available: http://www.tac.vic.gov.au/road-safety/safe-driving/tips-and-tools/fighting-fatigue (Date of Access 20 / 04 /2020)

[5] M. Ngxande, J-R. Tapamo, M. Burke, Driver Drowsiness Detection Using Behavioral Measures and Machine Learning Techniques: A Review of State-Of-Art Techniques, in: Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), 2017.

[6] Association for Safe International Road Travel, Road Safety Facts. [Online]. Available: https://www.asirt.org/safe-travel/road-safety-facts (Date of Access 20 / 04 /2020)

[7] Blue and Green Tomorrow, How Traffic Accidents Harm the Environment. [Online]. Available: https://blueandgreentomorrow.com/environment/traffic-accidents-harm-environment (Date of Access 20 / 04 /2020)

[8] A. Čolić, O. Marques, B. Furht, Driver Drowsiness Detection Systems and Solutions. Cham Heidelberg New York Dordrecht London: Springer, 2014.

[9] A. M. Williamson, A. M. Feyer, “Moderate Sleep Deprivation Produces Impairments in Cognitive and Motor Performance Equivalent to Legally Prescribed Levels of Alcohol Intoxication”, Occupational and Environmental Medicine, vol.57 no.10, pp.649-655, 2000.

[10] D. Dawson, K. Reid, “Fatigue, Alcohol and Performance Impairment”, Nature, vol.388 no.6639, pp.235, 1997.

[11] N. Lamond, D. Dawson, “Quantifying the Performance Impairment Associated with Fatigue”, Journal of Sleep Research, vol.8, no.4, pp.255-262, 1999.

[12] Centers for Disease Control and Prevention, Drowsy Driving: Asleep at the Wheel. [Online]. Available: https://www.cdc.gov/features/dsdrowsydriving/index.html (Date of Access 20 / 04 /2020)

[13] R. Ghoddoosian, M. Galib and V. Athitsos, A Realistic Dataset and Baseline Temporal Model for Early Drowsiness Detection, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[14] Zer Customs, Volvo Driver Alert Control and Lane Departure Warning. [Online]. Available: http://www.zercustoms.com/news/Volvo-Driver-Alert-Control-and-Lane-Departure-Warning.html (Date of Access 20 / 04 /2020)

[15] Motor 1, Toyota Redesigns Crown Introduces Hybrid Model. [Online]. Available: https://www.motor1.com/news/2106/toyota-redesigns-crown-introduces-hybrid-model/ (Date of Access 20 / 04 /2020)

[16] Daimler, Attention Assist Drowsiness Detection System Warns Drivers to Prevent Them Falling Asleep. [Online]. Available: https://media.daimler.com/marsMediaSite/en/instance/ko/ATTENTION-ASSIST-Drowsiness-detection-system-warns-drivers-to-prevent-them-falling-asleep-momentarily.xhtml?oid=9361586 (Date of Access 20 / 04 /2020)

[17] Volvo Cars, Driver Alert Control (DAC). [Online]. Available: https://www.volvocars.com/en-th/support/manuals/v60/2017-early/driver-support/driver-alert-system/driver-alert-control-dac (Date of Access 20 / 04 /2020)

[18] BMW, The Main Driver Assistance Systems. [Online]. Available: https://www.bmw.com/en/innovation/the-main-driver-assistance-systems.html (Date of Access 20 / 04 /2020)

[19] NISSAN USA, Drowsy Driver Attention Alert Car Feature. [Online]. Available: https://www.nissanusa.com/experience-nissan/news-and-events/drowsy-driver-attention-alert-car-feature.html (Date of Access 20 / 04 /2020)

[20] W. Han, Y. Yang, G. Bin Huang, O. Sourina, F. Klanner, and C. Denk, “Driver Drowsiness Detection Based on Novel Eye Openness Recognition Method and Unsupervised Feature Learning,” Proc. - 2015 IEEE Int. Conf. Syst. Man, Cybern. SMC 2015, no. September, pp. 1470–1475, 2016.

[21] M. Patel, S. K. L. Lal, D. Kavanagh, and P. Rossiter, “Applying neural network analysis on heart rate variability data to assess driver fatigue”, Expert Syst. Appl., 38(6):7235–7242, June 2011

[22] S. Hu and G. Zheng, “Driver drowsiness detection with eyelid related parameters by Support Vector Machine”, Expert Syst. Appl., 36(4):7651–7658, May 2009.

[23] D. McDonald, C. Schwarz, J. D. Lee, and T. L. Brown, “RealTime Detection of Drowsiness Related Lane Departures Using Steering Wheel Angle,” Proc. Hum. Factors Ergon. Soc. Annu. Meet., vol. 56, no. 1, pp. 2201–2205, 2012.

[24] A. Mittal, K. Kumar, S. Dhamija, and M. Kaur, “Head movementbased driver drowsiness detection: A review of state-of-art techniques,” Proc. 2nd IEEE Int. Conf. Eng. Technol. ICETECH 2016, pp. 903–908, 2016.

[25] A. Ramos, J. Erandio, E. Enteria, N. Carmen, L. Enriquez and D. Mangilaya, “Driver Drowsiness Detection Based on Eye Movement and Yawning Using Facial Landmark Analysis”,

International journal of simulation: systems, science & technology, 10.5013/IJSSST.a.20.S2.37., 2019.

[26] T. Nakamura, A. Maejima, and S. Morishima, "Detection of driver's drowsy facial expression," Proc. - 2nd IAPR Asian Conf. Pattern Recognition, ACPR 2013, pp. 749–753, 2013.

[27] Ching-Hua Weng, Ying-Hsiu Lai and Shang-Hong Lai, "Driver Drowsiness Detection via a Hierarchical Temporal Deep Belief Network", In Asian Conference on Computer Vision Workshop on Driver Drowsiness Detection from Video, Taipei, Taiwan, Nov. 2016

[28] Q. Massoz, T. Langohr, C. François and J. G. Verly, "The ULg Multimodality Drowsiness Database (called DROZY) and Examples of Use", Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV 2016), Lake Placid, NY, March 7-10, 2016.

[29] Computer Vision Lab, National Tsuing Hua University, Driver Drowsiness Detection Dataset. [Online]. Available: http://cv.cs.nthu.edu.tw/php/callforpaper/datasets/DDD/ (Date of Access 20 / 04 /2020)

[30] DROZY, The ULg Multimodality Drowsiness Database. [Online]. Available: http://www.drozy.ulg.ac.be/ (Date of Access 20 / 04 /2020)

[31] UTA-RLDD, UTA Real-Life Drowsiness Dataset. [Online]. Available: https://sites.google.com/view/utarldd/home (Date of Access 20 / 04 /2020)

[32] P. Viola and M. Jones, "Robust real-time object detection", *International Journal of Computer Vision*, 2001

[33] Y. Freund and Robert E. Schapire. "A Short Introduction to Boosting." (1999).

[34] A. Lenskiy and J.-S. Lee, "Driver's eye blinking detection using novel color and texture segmentation algorithms", *International Journal of Control, Automation and Systems*, 10(2):317–327, 2012.

[35] T. Ojala, M. Pietikainen and D. Harwood, "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions", *Proceedings of 12th International Conference on Pattern Recognition*, Jerusalem, Israel, 1994, pp. 582-585 vol.1.

[36] Li, Kangning et al. "Accurate Fatigue Detection Based on Multiple Facial Morphological Features." *J. Sensors* 2019: 7934516:1-7934516:10, 2019.

[37] E. Tadesse, W. Sheng and M. Liu, "Driver drowsiness detection through HMM based dynamic modeling," *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, 2014, pp. 4003-4008.

[38] F. Wang, M. Zhou, and B. Zhu, "A novel feature based rapid eye state detection method", In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pages 1236–1240, 2009.

[39] Tech-Quantum, Various Techniques to Detect and Describe Features in an Image Part-1. [Online]. Available: https://www.tech-quantum.com/various-techniques-to-detect-and-describe-features-in-an-image-part-1/ (Date of Access 20 / 04 /2020)

[40] Tech-Quantum, Various Techniques to Detect and Describe Features in an Image Part-2. [Online]. Available: https://www.tech-quantum.com/various-techniques-to-detect-and-describe-features-in-an-image-part-2/ (Date of Access 20 / 04 /2020)

[41] Naz, Saima et al. "Driver Fatigue Detection using Mean Intensity, SVM, and SIFT." *IJIMAI* 5: 86-93, 2019.

[42] R. Prem Kumar, M. Sangeeth, K.S. Vaidhyanathan, A. Pandian. "TRAFFIC SIGN AND DROWSINESS DETECTION USING OPEN-CV", International Research Journal of Engineering and Technology (IRJET) vol. 06 issue 03, 2019.

[43] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 2005, pp. 886-893 vol. 1.

[44] DLib, Face Detector. [Online]. Available: http://dlib.net/face_detector.py.html (Date of Access 20 / 04 /2020)

[45] 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.

[46] A. Dasgupta, A. George, S. L. Happy and A. Routray, "A Vision-Based System for Monitoring the Loss of Attention in Automotive Drivers", in: *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1825-1838, Dec. 2013.

[47] T. Soukupová and Jan Cech. "Real-Time Eye Blink Detection using Facial Landmarks", 2016.

[48] Q. Cheng, W. Wang, X. Jiang, S. Hou and Y. Qin, "Assessment of Driver Mental Fatigue Using Facial Landmarks", in *IEEE Access*, vol. 7, pp. 150423-150434, 2019.

[49] Zhong, G., Ying, R., Wang, H., Siddiqui, A., & Choudhary, G., Drowsiness Detection with Machine Learning. [Online]. Available: https://towardsdatascience.com/drowsiness-detection-with-machine-learning-765a16ca208a (Date of Access 20 / 04 /2020)

[50] I. H. Choi, C. H. Jeong, and Y. G. Kim, "Tracking a driver's face against extreme head poses and inference of drowsiness using a hidden Markov model", Appl. Sci., vol. 6, no. 5, 2016.

[51] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers", Proc. fifth Annu. Work. Comput. Learn. theory, pp. 144–152, 1992.

[52] L. E. Baum and T. Petrie, "Statistical Inference for Probabilistic Functions of Finite State Markov Chains", Statistics (Ber)., pp. 50–60, 1966.

[53] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain", *Psychological Review, 65*(6), 386–408, 1958.

[54] Y. LeCun and Y. Bengio, “The Handbook of Brain Theory and Neural Networks,” no. 1, M. A. Arbib, Ed. MIT Press, 1998, pp. 1–5.

[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Adv. Neural Inf. Process. Syst., pp. 1–9, 2012.

[56] S. Hochreiter and J. Schmidhuber, “Long Short-term Memory. Neural computation”, 9. 1735-80. 10.1162/neco.1997.9.8.1735., 1997.

[57] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition. ArXiv, abs/1409.1556., 2015.

[58] M. Hashemi, A. Mirrashid, and A.B. Shirazi, “Deep learning based Driver Distraction and Drowsiness Detection. ArXiv, abs/2001.05137, 2020.

[59] A. Liu, Z. Li, L. Wang and Y. Zhao, "A practical driver fatigue detection algorithm based on eye state," *2010 Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia)*, Shanghai, 2010, pp. 235-238.

[60] Blink Matters, Research. [Online]. Available: https://www.blinkingmatters.com/research (Date of Access 20 / 04 /2020)

[61] PyPi, opencv-python. [Online]. Available: https://pypi.org/project/opencv-python/ (Date of Access 20 / 04 /2020)

[62] Dlib, Classes. [Online]. Available: http://dlib.net/python/index.html#dlib.get_frontal_face_detector (Date of Access 20 / 04 /2020)

[63] Dlib, Classes. [Online]. Available: http://dlib.net/python/index.html#dlib.shape_predictor (Date of Access 20 / 04 /2020)

[64] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees”, *2014 IEEE Conference on Computer Vision and Pattern Recognition* 1867-1874, 2014.

[65] B. Wilhelm, A. Widmann, W, Durst, C. Heine and G. Otto, “Objective and quantitative analysis of daytime sleepiness in physicians after night duties”, International journal of psychophysiology: official journal of the International Organization of Psychophysiology. 72. 307-13. 10.1016/j.ijpsycho.2009.01.008., 2009.

[65] Github, Home Page. [Online]. Available: http://github.com (Date of Access 20 / 04 /2020)

[66] Kaggle, Notebooks. [Online]. Available: https://www.kaggle.com/hakkoz/notebooks (Date of Access 20 / 04 /2020)

[67] Trello, Home Page. [Online]. Available: https://trello.com/ (Date of Access 20 / 04 /2020)

[68] OpenCV, Home Page. [Online]. Available: https://opencv.org/ (Date of Access 20 / 04 /2020)

[69] Dlib, Home Page. [Online]. Available: http://dlib.net/ (Date of Access 20 / 04 /2020)

[70] Open Face, OpenFace 2.2.0: a facial behavior analysis toolkit. [Online]. Available: https://github.com/TadasBaltrusaitis/OpenFace (Date of Access 20 / 04 /2020)

[71] Scikit Learn, Home Page. [Online]. Available: https://scikit-learn.org/stable/ (Date of Access 20 / 04 /2020)

[72] Keras, Home Page. [Online]. Available: https://keras.io/ (Date of Access 20 / 04 /2020)