DROWSY DRIVER DETECTION USING IMAGE PROCESSING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ARDA GİRİT

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2014

Approval of the thesis

DROWSY DRIVER DETECTION USING IMAGE PROCESSING

submitted by **ARDA GİRİT** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Gönül Turhan Sayan _____
Head of Department, **Electrical and Electronics Engineering**

Assoc. Prof. İlkay Ulusoy _____
Supervisor, **Electrical and Electronics Eng. Dept., METU**

Examining Committee Members:

Prof. Dr. Gözde Bozdağı Akar _____
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. İlkay Ulusoy _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Uğur Halıcı _____
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. Dr. Cüneyt Bazlamaçcı _____
Electrical and Electronics Engineering Dept., METU

Erdem Akagündüz, Dr. _____
MGEO, ASELSAN

Date: 07.02.2014

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Arda Girit

Signature :

ABSTRACT

DROWSY DRIVER DETECTION USING IMAGE PROCESSING

Girit, Arda
M.Sc., Department of Electrical and Electronics Engineering
Supervisor: Assoc. Prof. İlkay Ulusoy

February 2014, 100 pages

This thesis is focused on drowsy driver detection and the objective of this thesis is to recognize driver's state with high performance. Drowsy driving is one of the main reasons of traffic accidents in which many people die or get injured. Drowsy driver detection methods are divided into two main groups: methods focusing on driver's performance and methods focusing on driver's state. Furthermore, methods focusing on driver's state are divided into two groups: methods using physiological signals and methods using computer vision. In this thesis, driver data are video segments captured by a camera and the method proposed belongs to the group that uses computer vision to detect driver's state. There are two main states of a driver, those are alert and drowsy states. Video segments captured are analyzed by making use of image processing techniques. Eye regions are detected and those eye regions are input to right and left eye region classifiers, which are implemented using artificial neural networks. The neural networks classify the right and left eye as open, semi-closed or closed eye. The eye states along the video segment are fused and the driver's state is predicted as alert or drowsy. The proposed method is tested on 30-second- long video segments. The accuracy of the driver's state recognition method is 99.1%

and the accuracy of our eye state recognition method is 94%. Those results are comparable with the results in literature.

Keywords: Drowsy driver detection, traffic accidents, image processing, artificial neural networks, eye closure rate.

ÖZ

GÖRÜNTÜ İŞLEME İLE UYKULU SÜRÜCÜ TESPİTİ

Girit, Arda
Yüksek Lisans, Elektrik Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Doc. Dr. İlkay ULUSOY

Şubat 2014, 100 sayfa

Bu tez uykulu sürücü tespiti üzerine odaklanmıştır ve sürücünün durumunu yüksek bir performansla tespit etmeyi amaçlamaktadır. Bir çok insanın yaşamını yitirdiği veya sakatlandığı trafik kazalarının ana sebeplerinden biri de uykulu araç kullanımıdır. Uykulu sürücü tespit yöntemleri sürücünün performansına veya sürücünün durumuna odaklananlar olmak üzere iki ana gruba ayrılır. Sürücünün durumuna odaklanan yöntemler de, fizyolojik sinyalleri kullananlar veya bilgisayarla görmeyi kullananlar olmak üzere ikiye ayrılır. Bu tezde sürücü verisi kamera ile kaydedilen video dilimleridir ve önerilen yöntem, sürücünün durumunu tespit etmek için bilgisayarla görmeyi kullanan yöntemler grubuna aittir. Sürücünün uyanık ve uykulu olmak üzere iki durumu vardır. Kaydedilen video dilimleri analiz edilir, görüntü işleme teknikleriyle göz bölgeleri bulunur ve bulunan göz bölgeleri yapay sinir ağları kullanılarak oluşturulan sağ ve sol göz sınıflandırıcılarına girdi olarak verilir. Sinir ağları sağ ve sol gözleri açık, yarı açık ve kapalı olmak üzere sınıflandırır. Video dilimi boyunca bütün göz durumları birleştirilir ve sürücünün durumu uyanık veya uykulu olmak üzere tahmin edilir. Önerilen yöntem 30 saniyelik video dilimleri üzerinde test

edilmiştir. Sürücü durumu tespit yönteminin başarı oranı %99.1'dir ve göz durumu tespit yönteminin başarı oranı %94'tür. Bu sonuçlar literatürdeki sonuçlarla benzerlik göstermektedir.

Anahtar Kelimeler: Uykulu sürücü durumu tanıma, trafik kazaları, görüntü işleme, yapay sinir ağları, göz kapalılık oranı.

to the angel in heaven,

my beloved mother

ACKNOWLEDGEMENTS

As the author I would like to wish to express my deepest gratitude to my supervisor Assoc. Prof. İlkay Ulusoy for her guidance, advice, criticism, encouragements, insight throughout the research and for trusting me.

I would like to thank to Assoc. Prof. Ece Güran Schmidt for showing me the way when I felt lost.

I would like to thank to love of my life, my wife for being full of love to me.

I would like to thank to one of my best friends, Fatih Lokumcu for being devoted.

I would like to thank to the members of Computer Vision and Pattern Analysis Laboratory at Sabanci University and Drive Safe Project for providing me UYKUCU database.

Finally, I would like to thank to TÜBİTAK for their financial support.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Definition And Objective

Traffic accidents is a threatening phenomenon for human beings. According to Turkish Statistical Institute, about 1.3 million traffic accidents occured throughout 2012 [1]. Many of these accidents result from drivers' being drowsy or asleep. In Table 1.1, total accidents, accidents involving death and personal injury, number of persons killed and injured are listed year by year between 2003 and 2012.

Table 1.1: Total accidents, number of accidents involving death and personal injury, number of persons killed and injured between 2003 and 2012 in Turkey [1]

Years	Total Accidents	Accidents involving death and personal injury	Number of persons killed	Number of persons injured
2003	455 637	67031	3946	118214
2004	537 352	77008	4427	136437
2005	620 789	87273	4505	154086
2006	728 755	96128	4633	169080
2007	825 561	106994	5007	189057
2008	950 120	104212	4236	184468
2009	1 053 346	111121	4323	201380
2010	1 106 201	116804	4045	211496
2011	1 228 928	131845	3835	238074
2012	1 296 634	153552	3750	268079

According to Turkish Statistical Institute, the number of traffic accidents increases gradually every year. This is not just a problem of our country, it is rather a worldwide problem. According to Association for Safe International Road Travel, about 1.3 million persons die in road crashes each year, which means 3287 deaths a day [2]. The situation is similar in the United States. Statistics show that each year 37000 persons die in road crashes and 2.35 million are injured or disabled. Road crashes cost the United States $230.6 billion per year. One of the main reasons of traffic accidents is drowsy driving. According to the estimation made by US National Highway Traffic Safety Administration, each year approximately 100000 traffic accidents arise from driver drowsiness and fatigue [3,4]. According to statistics gathered by federal government, in the U.S. at least 1500 persons die and 40000 persons are injured in drowsy driver crashes each year. These numbers are most likely an underestimation. Unless someone witnesses or survives the car accident and can testify to the driver's condition, it is difficult to detect the driver being drowsy [4]. 37% of surveyed American adults said they have dozed off while driving at least once and 27% said that they dozed off while driving in the past year. In the USA, a series of studies by the National Transportation Safety Board (NTSB) have pointed out the significance of sleepiness as being the reason of accidents involving heavy vehicles [5]. According to NTSB, 52% of 107 single-vehicle accidents involving heavy trucks were drowsiness-related.

Fatigue affects a driver’s ability to drive. With increasing fatigue, the driver’s reaction time increases and driver’s ability to avoid accidents decreases. As seen on Figure 1.1, there is a strong positive correlation between hours spent driving and the number of fatigue-related accidents [6].

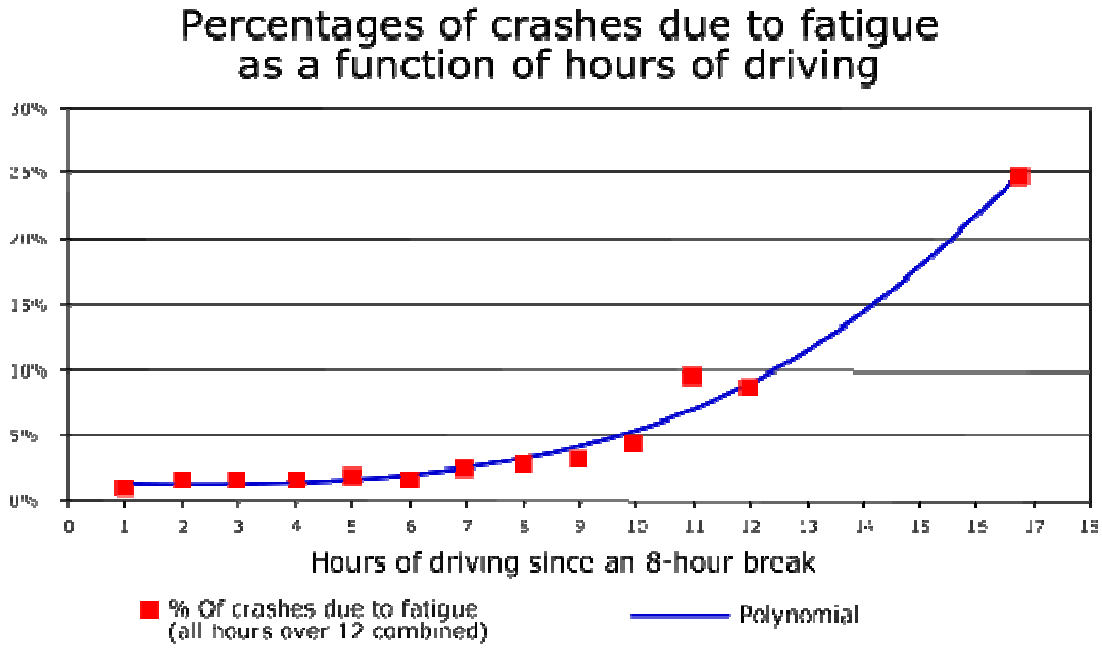

Figure 1.1: The figure displays the strong positive correlation between hours of driving and fatigue-related crashes [6].

Adelaide Centre for Sleep Research conducted a study which shows that drivers being awake for 24 hours are seven times more likely to have an accident and have a driving performance which is equal to a person who has a blood alcohol content of 0.1g/100ml [6].

The facts and statistics stated make it obvious that drowsy driving is a significant problem for human beings. The objective of this thesis is to recognize drivers' state with high performance. An effective and user-friendly drowsiness detection system will save people's lives and make our world a better place to live.

1.2 Literature Review

Drowsiness detection methods are seperated into two main categories: methods focusing on driver's performance and methods focusing on driver's state.

1.2.1 Methods Focusing on Driver's Performance

In order to detect drowsiness, studies on driver's performance use lane tracking, distance between driver's vehicle and the vehicle in front of it; place sensors on

components of the vehicle such as steering wheel, gas pedal and analyze the data taken by these sensors. Pilutti and Ulsoy used vehicle lateral position as the input and steering wheel position as the output and they obtained a model which can be useful to detect drowsiness [10] [11]. Some of the previous studies make use of driver steering wheel movements and steering grips as an indicator to detect drowsiness. Some car companies such as Nissan [12] and Renault [13] used this technology. Since these systems are too dependent on the characteristics of the road, they can only function well on motorways which make them work in limited situations [14]. These systems are affected too much by the road quality and lighting. Another disadvantage of these systems is that they cannot detect drowsiness that has not affected vehicle's situation yet. When a driver is drowsy and the vehicle is in the appropriate lines, these systems cannot detect drowsiness.

1.2.2 Methods Focusing on Driver's State

Methods focusing on driver's state are separated into two main groups: methods using physiological signals and methods using computer vision.

1.2.2.1 Methods Using Physiological Signals

The methods use physiological signals such as Electroencephalography(EEG), heart rate variability (HRV), pulse rate and breathing. The spectral analysis of heart rate variability shows that HRV has three frequency bands: high frequency band (0.15-0.4 Hz), low frequency band (0.04-0.15 Hz) and very low frequency band (0.0033-0.04Hz) [15] [16]. Researchers have found out that LF/HF ratio decreases and HF power increases when a person goes from alert state to drowsy state [17].

Power spectrum of EEG brain waves is used as an indicator to detect drowsiness; as drowsiness level increases, EEG power of the alpha and theta bands increases and beta band decreases[18][19]. EEG-based drowsiness detection methods are not easily implementable because they require the driver to wear an EEG cap during driving the vehicle. Devices being distractive is the main disadvantage of this group of methods.

1.2.2.2 Methods Using Computer Vision

This group of methods are not offensive and does not make any disturbance to the driver, that's why these methods are more pereferable. These methods are seperated into two groups: the ones using infrared illumination and the ones using day illumination. The former find the location of the eyes and detect eye states by making use of retinal reflections of infrared waves [20]. Matsuo and Khiat, who work for NISSAN, divided driver's condition into 4 categories: normal (alert), slightly sleepy, intensely sleepy and drowsy [25]. They used eye closure rate (ECR), head sway and subsidiary behaviours to detect the driver's condition. IR-based systems work well at night but do not work well during daylight illumination since sunlight reflections make it impossible to detect retinal reflections of infrared waves.

Methods using daylight illumination generally find the location of the face and eyes by making use of computer vision techniques. Driver State Sensor (DSS), developed by SeeingMachines, is a commercial product in this group [21]. DSS uses face tracking and gets information about eyelid opening and percentage of eye closure in order to detect drowsiness. Viola Jones used the Haar-like features for face and eye detection and this method is used in many studies [9]. Wu et al. uses adaboost classifier for face detection and uses intensity image to detect the pupils of eyes. Then, they take radial-symmetry transform and use support vector machine (SVM) to find the location of eyes [22]. After finding the location of eyes, using local binary patterns (LBP) and SVM classifier, they determine the eye state. Flowchart of their system is seen on Figure 2.1.

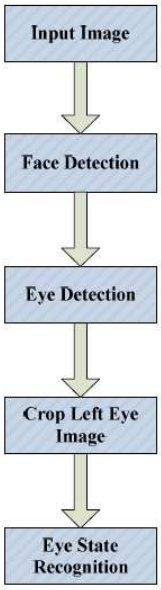

Figure 1.2: Flowchart of eye state recognition of [22]

Support vector machine is used by Flores et al. in order to identify eye state of a driver [23]. They have used their drowsiness detection system as a new module for Advanced Driver Assistance System(ADAS). In [24], they assumed that eye region is detected successfully and then they use the eye region image to detect whether the eye is fully open, partially open or fully closed. They use singular value decomposition (SVD) and the term called eigen-eyes and decide which group the eye image belongs to. In addition to eye closure, yawning data is used as an indicator of drowsiness in [26]. They merged mouth and eye state data to detect drowsiness. However, Vural et al. states that yawning, which is assumed to be predictive of drowsiness, is a negative predictor of the 60-second window prior to crash [27]. Drivers yawn less in the moments before falling asleep. They use facial movements in order to detect drowsiness. Facial movements are extracted by the help of a toolbox called Computer Expression Recognition Toolbox (CERT), which is a fully-automatic tool for facial expression recognition [28]. They use facial action coding system (FACS) and find out which action units have postive or negative correlations with drowsiness and then by making use of this action unit data, they decide whether the driver is drowsy or alert [29] [30].

1.3 Approach To The Solution

The method we propose belongs to the group which focuses on driver's state by making use of computer vision. Eye closure rate, in other words, percentage of eye closure (PERCLOS) is a reliable measure to detect drowsiness [48]. This thesis makes use of PERCLOS to decide whether the driver in a video segment is drowsy or alert. For every frame in the video segment, eye state estimation is performed. There are 3 states of an eye: open eye, semi-closed eye and closed eye. The estimations for each frame in a video segment are combined and the driver's state is estimated.

The video segment is extracted to its frames. After extraction of the frames of the video segment, the frames are input to the part called eye region extractor. Eye region extractor firstly finds the candidates for right and left eye regions and face by making use of extended version of Viola-Jones algorithm, which is available in Computer Vision System Toolbox of MATLAB [7] [8] [9]. Among the candidates of face, the wrong candidates are eliminated by some decision rules and the face region is detected. The detected face region is used to select the valid right and left eye region among the candidates found by extended version of Viola-Jones. After the detection of both eye regions, the eye images are converted to grayscale, resized to [12 18] and histogram equalized. After this process, every right and left eye image is input to neural networks separately which are trained with the subject's eye region images. For each frame, the outputs of right and left eye neural networks are both digitized and merged in order to estimate the eye state of the subject. After eye state estimation for all of the frames of a video segment is completed, the mean of the estimated eye states is calculated by assigning "0" to open eyes, "0.5" to semi-closed eyes and "1" to closed eyes. The mean value obtained is called "eye closure point per frame" and an eye closure point per frame more than a threshold value means a drowsy driver, whereas an eye closure point per frame less than a threshold value means an alert driver.

Combining the estimations for right and left eyes increases the accuracies for both eye state and drowsiness detection. Since combination of the estimations for right and left eyes is not a common method used in the literature, increasing the accuracy with this method is a contribution of our proposed algorithm. Most of the studies assign eyes only two states: open and closed. As another contribution, this study reveals the fact that semi-closed state has an important role in detecting drowsiness and defining three states instead of two states increases the accuracy of the drowsiness detection method proposed.

1.4 Outline

This thesis totally consists of 5 chapters. The remainder of the thesis is organized as follows:

Chapter 2 gives background information on some image processing methods and neural networks. Some background methods of which implementation is available on MATLAB are analyzed, as well.

Chapter 3 describes the method we propose to detect drowsiness and all of the steps of the solution are analyzed in this chapter.

Chapter 4 presents the test data, ground truth data, experiments and results of the experiments. The comments are made about the results and the results are compared.

Chapter 5 includes the conclusion and some potential topics for future studies

CHAPTER 2

BACKGROUND METHODS

In this chapter, background information on some concepts used in our method is provided.

2.1 Artificial Neural Networks

In computer vision, artificial neural networks are models which are capable of machine learning and pattern recognition. They are inspired from central nervous systems and they are systems of interconnected neurons. A neuron consists of input weights, a summer and an output (activation) function. An example of a neuron model can be seen in Figure 2.1.

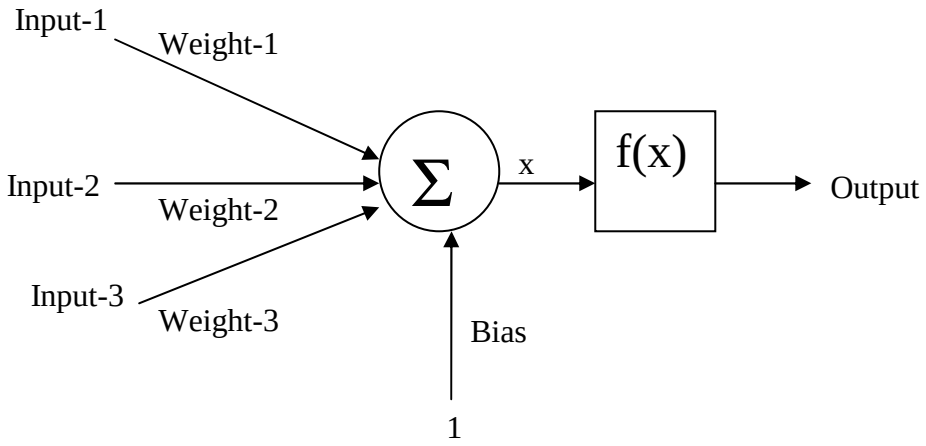

Figure 2.1: A simple neuron model

Output = f(Input-1*Weight-1 + Input-2*Weight-2 + Input-3*Weight-3 + Bias)

A sample neural network with one hidden layer, an input layer and an output layer is shown in Figure 2.2. This neural network has 3 neurons in the input layer, one hidden layer with 4 neurons and 2 neurons in the output layer.

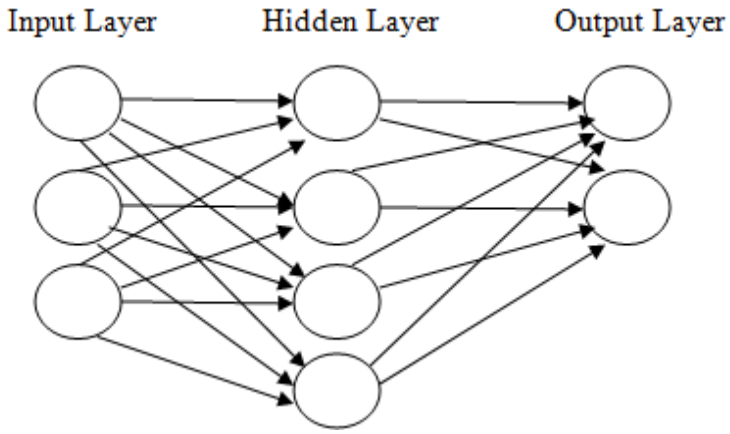

Figure 2.2: A neural network with a hidden layer

Feedforward neural networks are networks in which information moves only in one direction from input layers to output layers through hidden layers. There is no backward connection in this kind of neural networks.

Backpropagation (backward propagation of errors) is a form of supervised training. In order to use backpropagation training method, the sample inputs and corresponding outputs must be given. Making use of the corresponding outputs, the backpropagation training algorithm takes a calculated error and adjusts the weights of layers backwards from the output layer to the input layer.

There are many functions used as activation functions of neurons. Two examples of these are linear transfer function (purelin) and hyperbolic tangent function (tansig). Characteristic of linear transfer function and the symbol of a neuron having this type of activation function are shown in Figure 2.3.

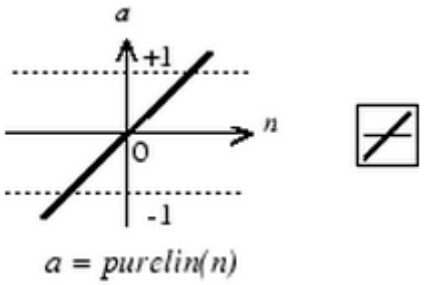

Figure 2.3: Linear transfer function

Characteristic of hyperbolic tangent sigmoid transfer function and the symbol of a neuron having this type of activation function are shown in Figure 2.4. This is a nonlinear function and nonlinear functions ought to be used for real life problems. Its formulation is:

tansig(n) = 2/(1+exp(-2*n))-1.

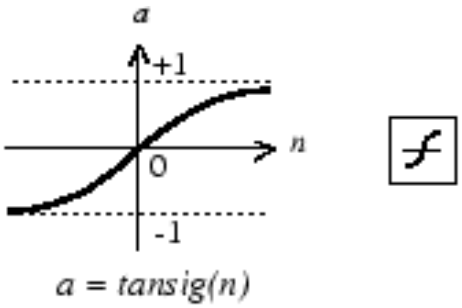

Figure 2.4: Hyperbolic tangent sigmoid transfer function

The structure of the neural networks we use are shown in Figure 2.5. They have 216 neurons in the input layer, 1 neuron with purelin as the activation function in the output layer and a hidden layer with n neurons each having hyperbolic tangent sigmoid transfer function as the activation function.

Inputs: $I_1, I_2, I_3, \ldots, I_{216}$

Weights between input neuron-i and and hidden neuron-j:

$w_{1,1}, w_{1,2}, \ldots, w_{1,n}, w_{2,1}, w_{2,2}, \ldots, w_{2,n}, \ldots\ldots\ldots\ldots, w_{216,1}, w_{216,2}, \ldots, w_{216,n}$

Weight between hidden neuron-i and output neuron: $w_1, w_2, w_3, \ldots, w_n$

Bias values for hidden neurons: $b_1, b_2, b_3, \ldots, b_n$

Bias value for output neuron = 0

$$\text{Output} = w_1 \times (\text{tansig}(b_1 + I_1 \times w_{1,1} + I_2 \times w_{2,1} + \ldots + I_{216} \times w_{216,1})) +$$
$$w_2 \times (\text{tansig}(b_2 + I_1 \times w_{1,2} + I_2 \times w_{2,2} + \ldots + I_{216} \times w_{216,2})) +$$
$$\vdots$$
$$w_n \times (\text{tansig}(b_n + I_1 \times w_{1,n} + I_2 \times w_{2,n} + \ldots + I_{216} \times w_{216,n}))$$

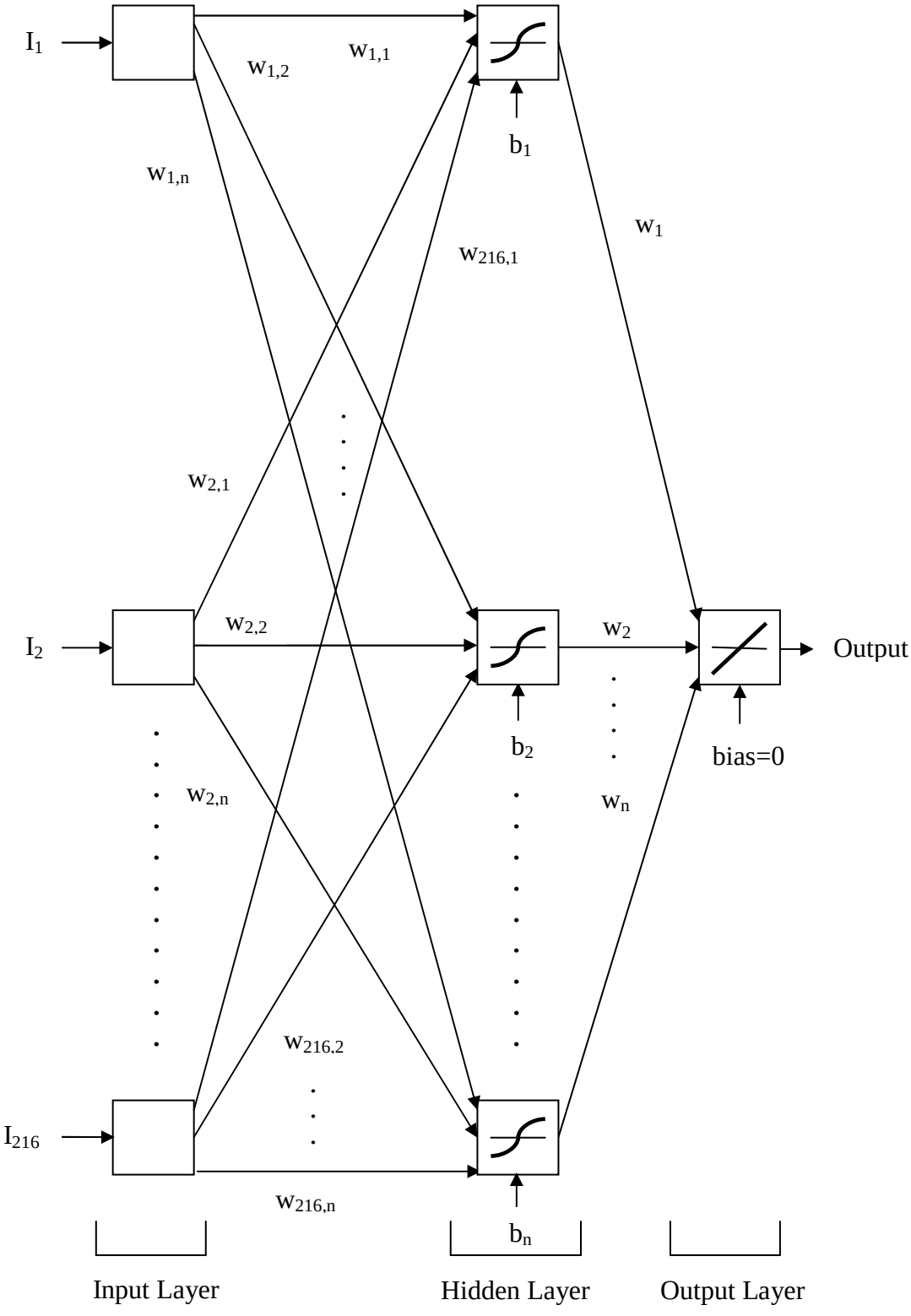

Figure 2.5: The structure of our neural networks

In the method we propose, we use neural networks in order to decide whether an eye is open, semi-closed or closed. We first extract the frames of the video segments to be used for training and for each frame right and left eye regions are detected. We grayscale, resize and histogram equalize the eye region images cropped from the frames, then we input the eye region images and ground truth data to neural networks in order to train them.

For the test stage, we first extract the frames of the video segment to be tested and for each frame, right and left eye regions are detected and cropped from the original frame. We grayscale, resize and histogram equalize the eye region images cropped and we input these eye region images to neural networks we have trained. Neural networks specified for right and left eye estimates the eye state separately for right and left eye. The eye state estimation of right and left eye neural networks are then merged and final estimation for the eye state is reached.

2.2 Gray-Level Images, Histogram Equalization and Resizing an Image

In computer vision, a gray-level digital image is an image of which every pixel carries intensity information. Gray-level images are also known as black-and-white images. Weakest intensity has the color black and strongest intensity has the color white. A sample conversion from a colorful image to a gray-level image is seen on Figure 2.6.

Figure 2.6: Sample rgb2gray conversion

Histogram equalization is a method of contrast adjustment by using image's histogram. Histogram equalization usually increases the global contrast of the input image. In Figure 2.7, conversion to gray-level and histogram equalization are applied consecutively to an eye region image.

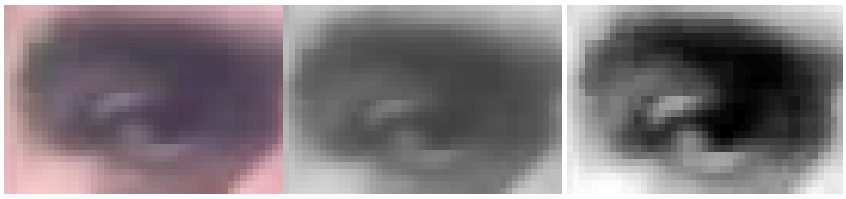

Figure 2.7: Eye region image converted to gray-level and then histogram equalized

Resizing an image is the operation of fitting the image to a desired size and the value that pixels will take are computed by different algorithms. The most primitive one of these algorithms is nearest-neighbor interpolation in which the output pixel value is taken only from the value of the pixel that falls within. Another algorithm is bilinear interpolation in which the output pixel value is computed as the weighted average of pixels in the nearest 2-by-2 neighborhood. The interpolation method we use in our methodology is bi-cubic interpolation in which the output pixel value is computed as the weighted average of pixels in the nearest 4-by-4 neighborhood. Some examples of resizing of a gray-scaled frame extracted from video data are shown in Figure 2.8.

Figure 2.8-a: Original gray-level image with size [480 640]

Figure 2.8-b: Resized image with size [240 320]

Figure 2.8-c: Resized image with size [120 160]

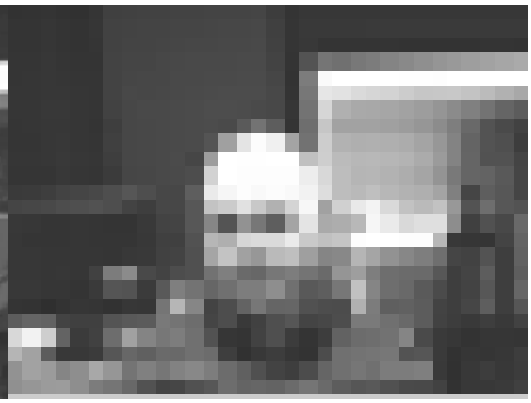

Figure 2.8-d: Resized image with size [24 32]

Figure 2.8: Examples of resized images

In Figure 2.9, there is an example of eye region image, grayscaled version of it and resized version of it. The eye region detected has dimensions [33 49] and it is resized to [12 18].

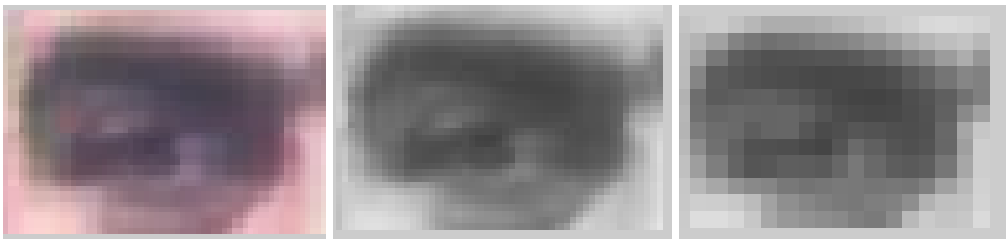

Figure 2.9-a: Eye region image with size [33 49]

Figure 3.9-b: Gray-scaled eye region image with size [33 49]

Figure 3.9-c: Resized image with size [12 18]

Figure 2.9: An example of resized eye region image

2.3 Viola-Jones's Object Detector

The method proposed by Viola and Jones is composed of weak classifiers cascaded and their output is a strong classifier which detects the target object [9]. For each stage in the cascade, a weak classifier is trained to reject a certain fraction of the non-target object patterns and not rejecting any part of the target object.

The classifiers use Haar features in order to encode facial features. Features used in [9] are seen on Figure 2.10.

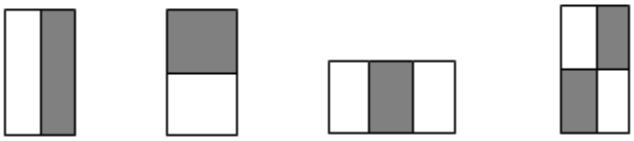

Figure 2.10: Haar features used in [9]

For each feature, the value is the difference between the sum of the pixels in black regions and the sum of the pixels in white regions.

Rectangle features are computed very rapidly using the integral image which is an intermediate representation. The integral image at location pixels (x, y) contains the sum of the upper left pixels of the original image, inclusively. As seen on Figure

2.11, the value of the integral image at location 1 is the sum of the pixels in rectangle A, the value at location 2 is A+B, location 3 is A+C and at location 4 is A+B+C+D. Then, the sum within D is computed as 4 + 1 – 2 – 3 that means the sum of the pixels within rectangle D is computed with four array references. Integral image provides the advantage of fast feature evaluation.

Adaboost is used to select a small set of features and train the classifiers. The learning algorithm for weak classifiers is designed to select the single rectangle feature best separating the positive and negative examples. For each feature, the weak classifier determines the optimal threshold.

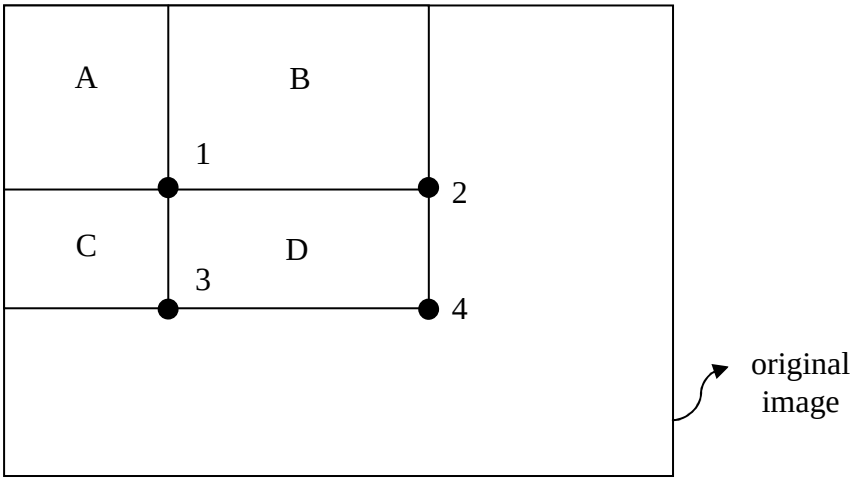

Figure 2.11: Integral image

Weak classifiers rejecting non-face regions and accepting face regions, are trained in such a way that their false negative rate approaches 0. Then, they are combined in a form of a degenerate decision tree which is called “cascade”. Figure 2.12 shows how cascaded weak classifiers with hit rate=h and false alarm rate=f form a strong classifier with hit rate=h^N and false alarm rate=f^N.

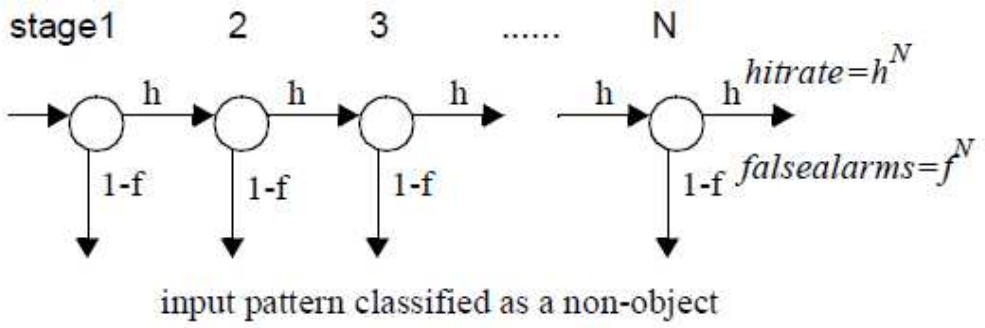

Figure 2.12: Cascade of classifiers with N stages. The classifier at each stage is trained to achieve a hit rate of h and a false alarm rate of f [7].

2.4 CART-based Face Detection

CART-based face detection method is composed of cascaded weak classifiers and their output is a strong classifier which detects face region. It is composed of weak classifiers based on the classification and regression tree analysis (CART) [34]. Boosting, a powerful learning concept, is used as the basic classifier. Many of the weak classifiers which are simple and inexpensive are combined resulting in a strong classifier. Gentle Adaboost whose algorithm is shown in Figure 2.13, is used during training of weak classifiers [34].

Gentle AdaBoost

1. Given N examples $(x_1,y_1),\ldots,(x_N,y_N)$ with $x \in \Re^k, y_i \in \{-1,1\}$
2. Start with weights $w_i = 1/N$, $i = 1, \ldots, N$.
3. Repeat for $m = 1, \ldots, M$
   (a) Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i
   (c) Set $w_i \leftarrow w_i \cdot \exp(-y_i \cdot f_m(x_i))$, $i = 1, \ldots, N$, and renormalize weights so that $\sum_i w_i = 1$.
4. Output the classifier $sign\left[\sum_{m=1}^{M} f_m(x)\right]$

Figure 2.13: Gentle Adaboost training algorithm [7]

The classifiers use Haar features in order to encode facial features and CART-based classifiers provide modelling higher-order dependencies between facial features. The algorithm is an extended version of Viola-Jones object detection [9],

the haar-like features are extended and Gentle Adaboost is used for learning. Extended features are shown in Figure 2.14. In [9, 46, 47], only features (1a), (1b), (2a), (2c) and (4a) have been used. Extending features enhanced the power of the learning system which results in improved object detection performance.

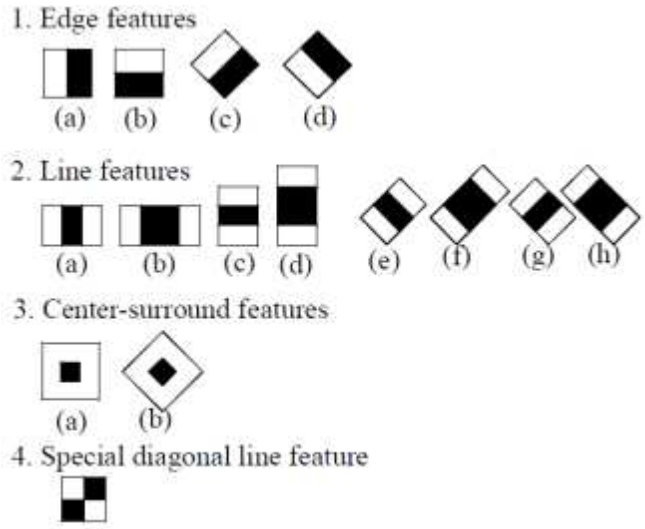

Figure 2.14: Features used in CART-based face detection method [7]

The implementation of this method is available in Computer Vision System Toolbox of MATLAB [36].

2.5 Local Binary Patterns and LBP-based Face Detection

Local binary patterns (LBP) is a feature used for classification in computer vision and it is a powerful feature for texture classification. The basic LBP operator is shown in Figure 2.15. As seen in Figure 2.15, the operator assigns a label to every pixel of an image by thresholding the 3x3-neighborhood of each pixel with the center pixel value and assigning a binary number to the result. Since {12, 13, 21} are less than the value of center pixel which is 54, they take the value 0. Since {54, 57, 86, 99, 85} are greater than or equal to 54, they take the value 1.

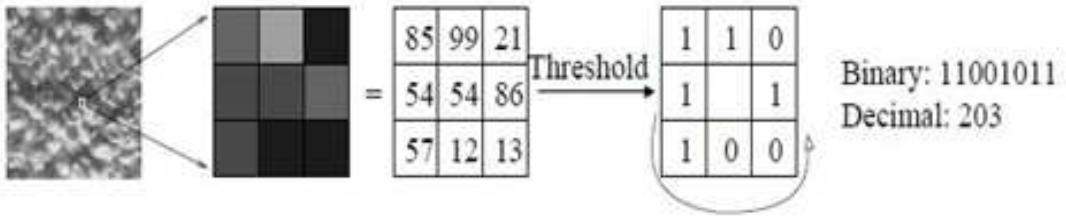

Figure 2.15: The basic LBP operator (3,1)

Local binary patterns for (8,1), (16,2) and (8,2) neighbourhoods are shown in Figure 2.16.

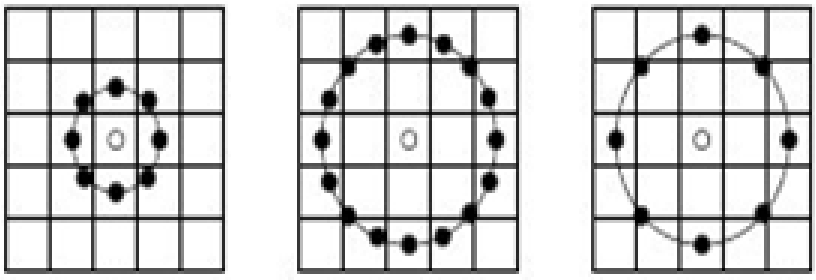

Figure 2.16: Local binary patterns for (8,1), (16,2) and (8,2) neighbourhoods respectively

LBP-based face detection method is composed of weak classifiers based on a decision stump [35]. The weak classifiers in this method use local binary patterns (LBP) to encode facial features unlike the CART-based method uses Haar features. LBP features provide robustness against illumination variations.

The implementation of this method is available in Computer Vision System Toolbox of MATLAB [36].

2.6 ENCARA2: A Real-time Eye Detection Method

Castrillon et al. proposed a system called ENCARA2 which includes real-time face and eye detection [8]. The eyes are detected pairly, the eye pair detection process is shown in Figure 2.17 and it is as follows:

The face blob boundaries are detected by making use of the skin colour model. The elements which are not part of the face are removed heuristically and an ellipse is fitted to the blob in order to rotate it to a vertical position [8] [37]. After the blobs are found, different alternatives are used to detect the eyes. Since eye pixels are darker than their surrounding pixels, dark areas are searched [38]. Viola-Jones based eye detector is used to search eyes with a minimum size [12 16], since the eye position is roughly estimated it provides fast performance. If eyes could not be found yet, Viola-Jones based eye pair detector with minimum size [5 22] is used. Then, detected eye positions are used to normalize the face region to a standard size. After normalization of face region, an area of size [11 11] around both eyes in the normalized face image is projected to a Principal Component Analysis (PCA) space and reconstructed. According to the reconstruction error, incorrect eye detections are identified and eliminated [39]. The PCA space consists of eigeneyes as an orthogonal basis.

The implementation of this method is available in Computer Vision System Toolbox of MATLAB [36].

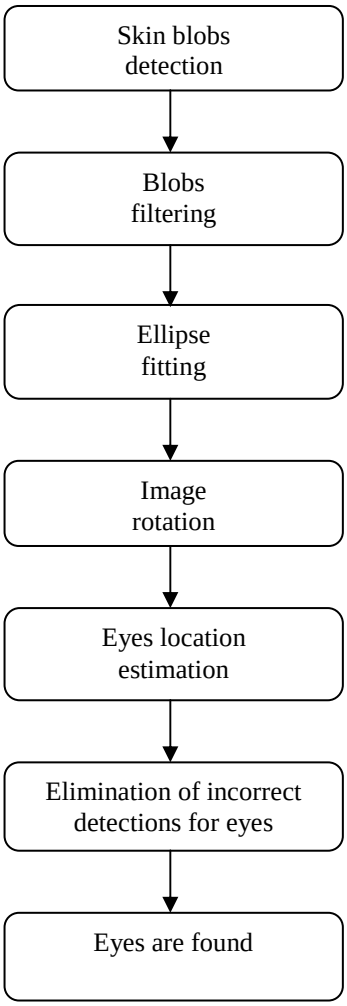

Figure 2.17: Eye detection process [8]

CHAPTER 3

PROPOSED METHOD

3.1 Summary of the System

This thesis makes use of eye closure rate to decide whether the driver in a video data is drowsy or alert. Procedure for testing a video is shown in Figure 3.1. A 30-second-long video segment is given as input to the system and the system outputs the decision of our method as drowsy or alert. Since the video database we use is 30 fps, for every 30-second video, 900 frames are extracted. After the frame extraction process is completed, all of the frames are input to the module called "Right and Left Eye Region Extractor". In this module, for each frame, right and left eye regions are found, then right and left eye region images are cropped from the original frames. This module outputs image of selected right and left eye regions and corresponding detection states. Image of selected right and left eye regions are converted to gray-level, resized to [12 18] and histogram equalized by the module Eye Region Image Modifier for Neural Networks. Now, the eye region images are ready to be input to neural networks. Those histogram equalized gray-level eye region images are input to right and left eye region neural networks which are trained with the same subject's eye region images obtained from different video segments. For each frame, the outputs of neural networks for right and left eye are input to the module called "drowsiness evaluator". In drowsiness evaluator module, the outputs of the right eye and left eye neural networks are digitized and then combined in order to estimate whether the eye is in open (0), semi-closed (0.5) or closed (1) state. After the estimation process is completed, the mean of the eye

states found is taken and this value is called “average eye state point”. For drowsy cases, average eye state point exceeds a threshold value and for alert videos it does not exceed that threshold value. That’s how the system decides whether the driver is drowsy or alert. The testing procedure and all of the modules are explained in detail in this chapter.

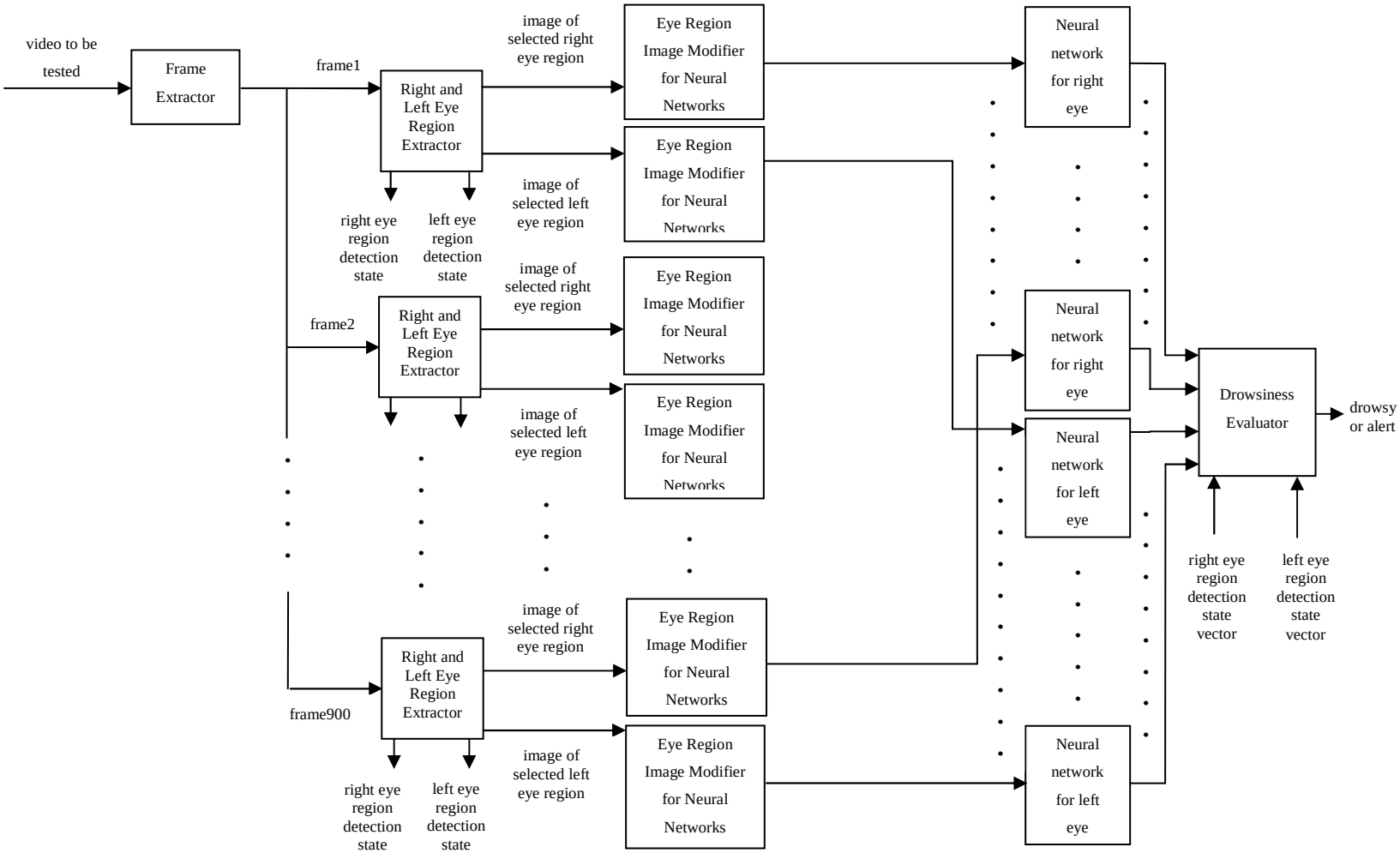

Figure 3.1: General procedure for testing

3.2 Extracting the Frames of a Video Segment

The module called “Frame Extractor” extracts the frames of a 30-second-long video segment. Since the video is 30 fps, Frame Extractor gives 900 frames with size [480 640] as its output. Frame Extractor’s input and output are clearly shown in Figure 3.2.

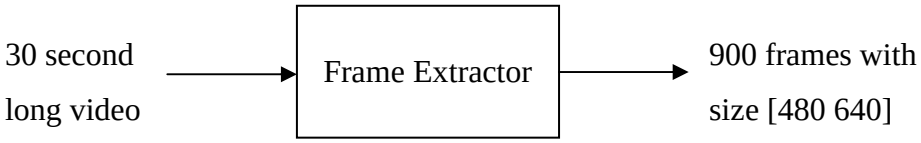

Figure 3.2: Frame Extractor module

3.3 Extracting Right and Left Eye Regions in a Frame

This module is called “Right and Left Eye Region Extractor” and its block diagram is shown in Figure 3.3. As seen on Figure 3.3, it is composed of modules called “Face Detector”, “Right and Left Eye Region Candidates Finder” and “Right and Left Eye Region Selector”. These modules are explained in detail in sections 3.3.1, 3.3.2 and 3.3.3 respectively.

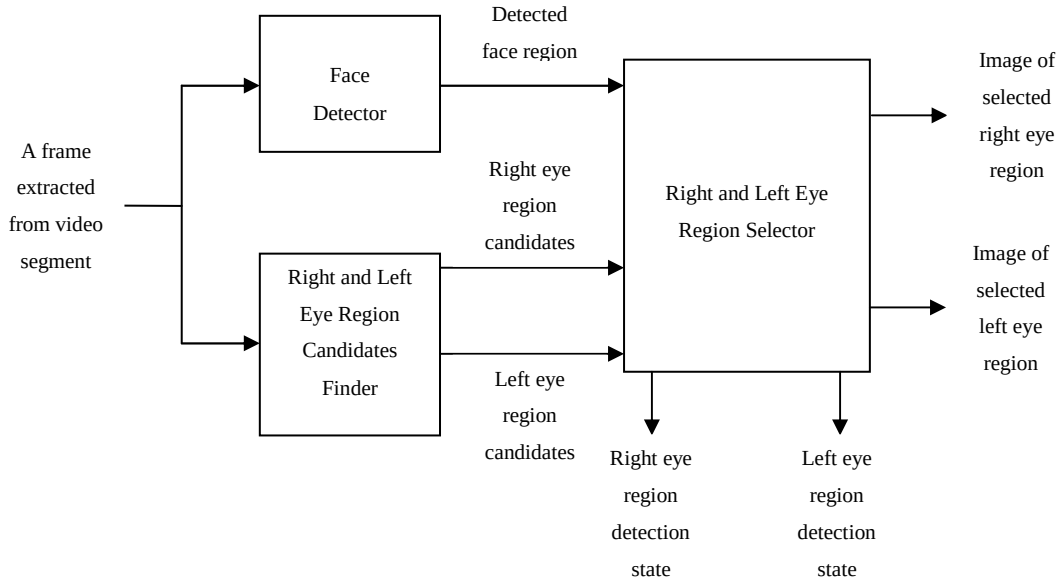

Figure 3.3: Block diagram of “Right and Left Eye Region Extractor” module

3.3.1 Face Detector

We use face region in order to make elimination among right and left eye candidates. That is, we use face borders and area to choose the valid detection of right and left eye among candidates found. The input of the face detector module is a frame extracted from video segment and the output is the detected face region. The simple block diagram of our Face Detector module is shown in Figure 3.4.

Figure 3.4: Block diagram of Face Detector module

In order to detect face region of the subjects, we tried the methods which are explained in sections 2.4 and 2.5. Both of the methods we have tried are composed of cascaded weak classifiers and their output is a strong classifier which detects face region.

We have tested 2 methods on 1800 frames and the test results are shown in Table 3.1

Table 3.1: Accuracy of the face detection methods

Method	Number of frames in which face region found detected successfully	Number of frames in which face region could not be detected successfully	Success Rate
Method-1(CART)	1794	6	99.7%
Method-2(LBP)	1788	12	99.3%

Since the first method's success rate is more than that of the second, we use the first one for detecting the face of the subjects. The implementations of these methods are available in Computer Vision System Toolbox of MATLAB [36].

This CART-based face detection method is not enough to successfully detect the face region because it does not always find only one face and in those cases, incorrect face detections need to be eliminated. Face Detector module includes face detection method mentioned above and an algorithm to avoid incorrect detections. Incorrect detections are eliminated according to the flowchart in Figure 3.5.

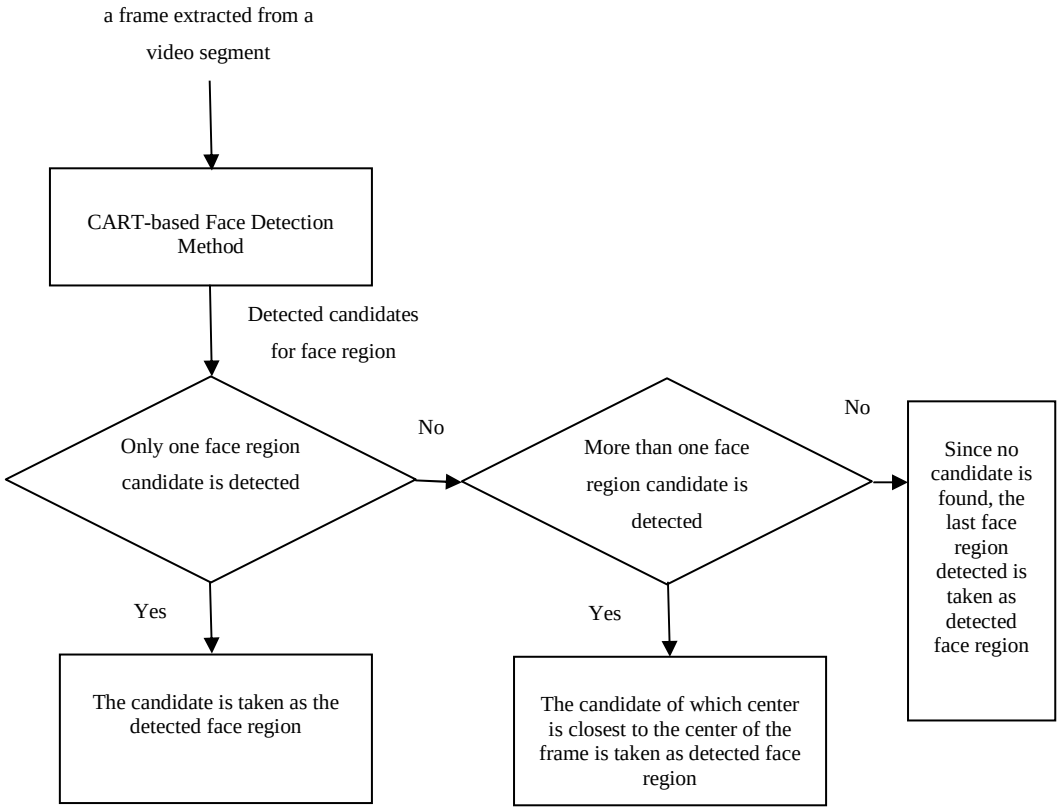

Figure 3.5: Flowchart of Face Detector

As seen on Figure 3.5 when just one face region candidate is detected by the face detection method we use, the candidate is taken as the detected face region. When more than one face region candidate exists, the candidate whose center is closest to

the center of the frame is taken as detected face region. When no candidate is detected, the last face region detected in the previous frame is taken as the detected face.

Some face detection examples of 4 subjects are shown in Figure 3.6.

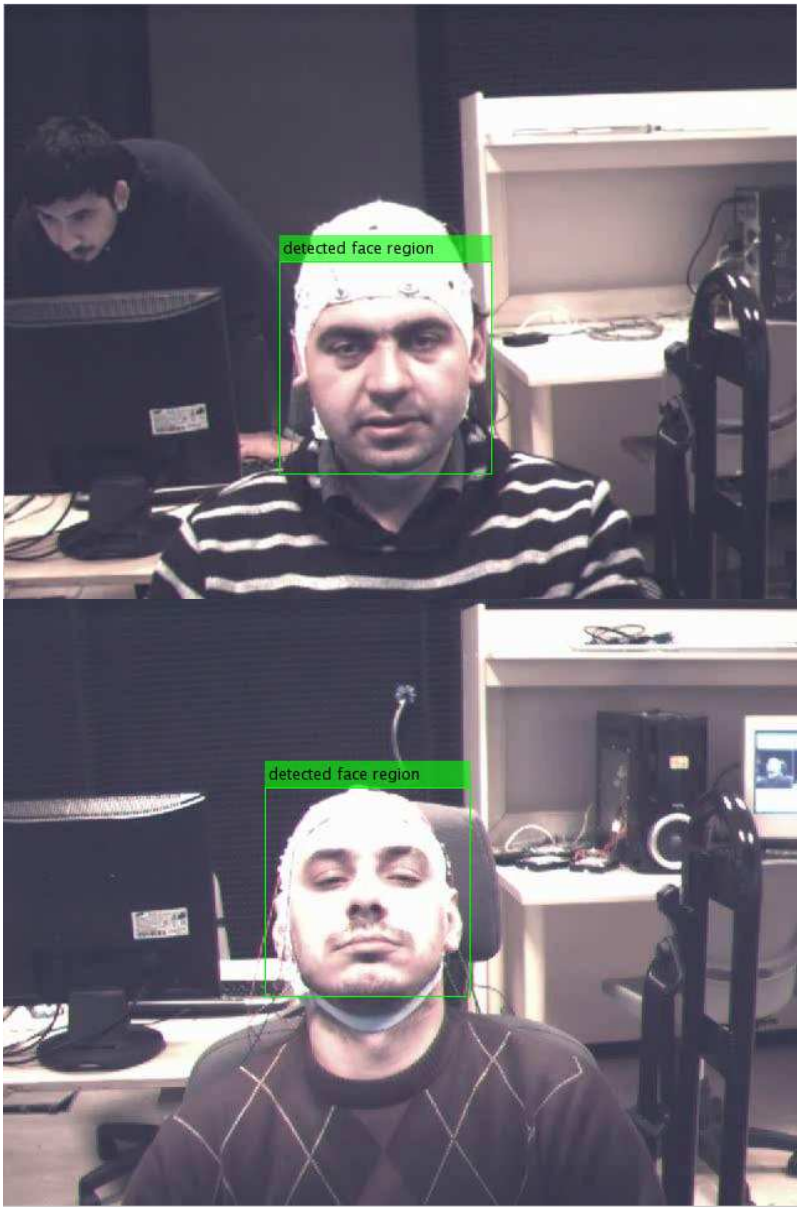

Figure 3.6: Sample face detection for subjects A,B,C and D respectively

Figure 3.6 (cont'd): Sample face detection for subjects A,B,C and D respectively

An example of elimination between face region candidates of subject A is shown in Figure 3.7. Among face region candidates, the candidate whose center is closest to the center of the frame is selected.

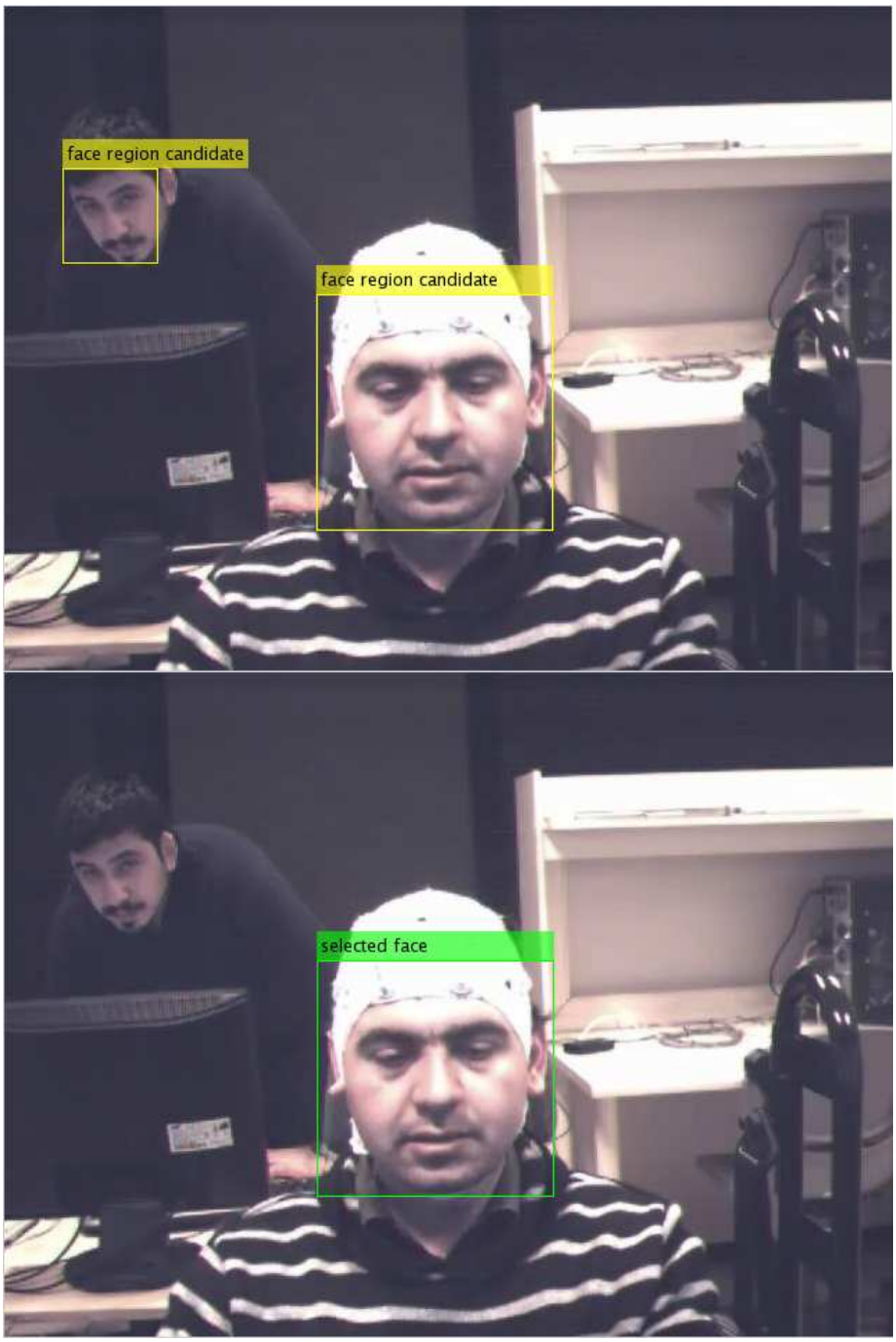

Figure 3.7: An example of elimination between face region candidates

Since eye state is used as an indicator of drowsiness, our objective is to find eye state after successfully finding eye region. Face region detected is used to choose

valid eye region among many eye region candidates, it only forms a reference in successfully finding right and left eye regions.

3.3.2 Right and Left Eye Region Candidates Finder

The method explained in section 2.6 is used to find the candidates for eye regions [8]. The block diagram of "Right and Left Eye Region Candidates Finder" module is shown in Figure 3.8.

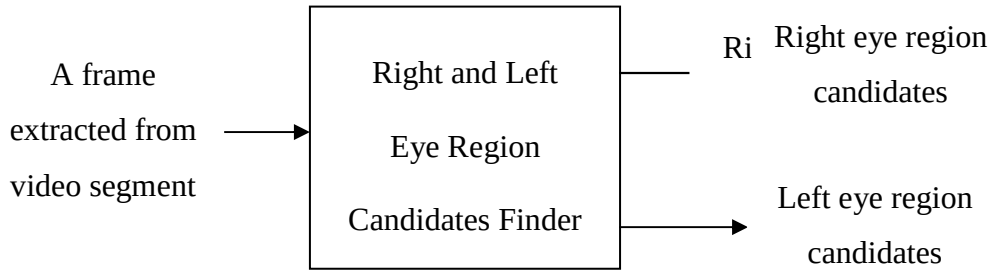

Figure 3.8: Right and Left Eye Region Candidates Finder module

Some examples of right eye region candidates found are shown in Figure 3.9.

Figure 3.9: Some examples of right eye region candidates found for subjects A, B,C and D respectively

Figure 3.9 (cont'd): Some examples of right eye region candidates found for subjects A, B,C and D respectively.

In Figure 3.9, 3 right eye region candidates are found for subject A, 2 right eye region candidates are found for subject B, 3 right eye region candidates are found for subject C and 1 right eye region candidate is found for subject D.

Some examples of left eye region candidates found are shown in Figure 3.10

Figure 3.10: Some examples of left eye region candidates found for subjects A, B,C and D respectively

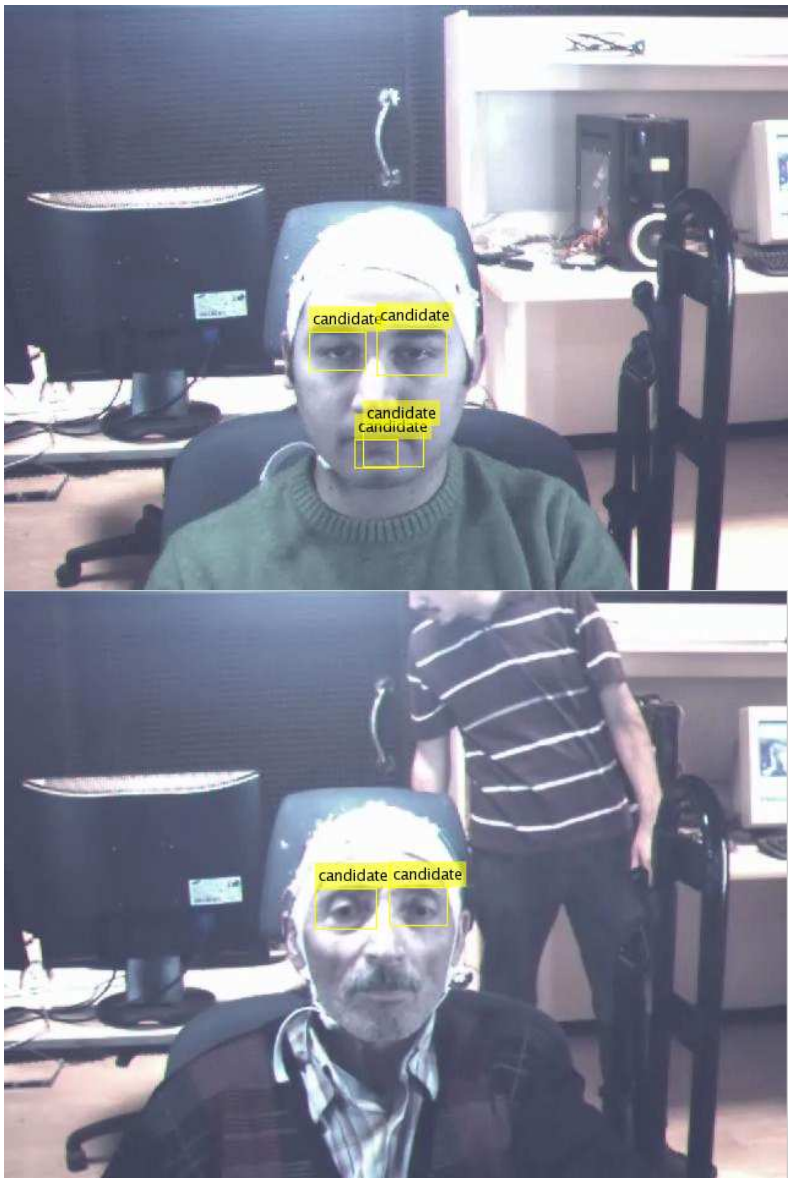

Figure 3.10 (cont'd): Some examples of left eye region candidates found for subjects A, B,C and D respectively

In Figure 3.10, 3 left eye region candidates are found for subject A, 2 left eye region candidates are found for subject B, 4 left eye region candidates are found for subject C and 2 left eye region candidates are found for subject D.

3.3.3 Right and Left Eye Region Selector

"Right and Left Eye Region Selector" is a module whose inputs are detected face, right and left eye region candidates which are the outputs of "Right and Left Eye Region Candidates Finder" module. The outputs of "Right and Left Eye Region Selector" module are image of selected right eye region, image of selected left eye region, right eye region detection state and left eye region detection state. Image of selected right eye region is the selected right eye region image cropped from the original frame and image of selected left eye region is the selected left eye region image cropped from the original frame.

Right eye region detection state is set to "1" if a right eye region is detected successfully and set to "0" if not. Left eye region detection state is set to "1" if a left eye region is detected successfully and set to "0" if not.
The block diagram of "Right and Left Eye Region Selector" module is shown in Figure 3.11.

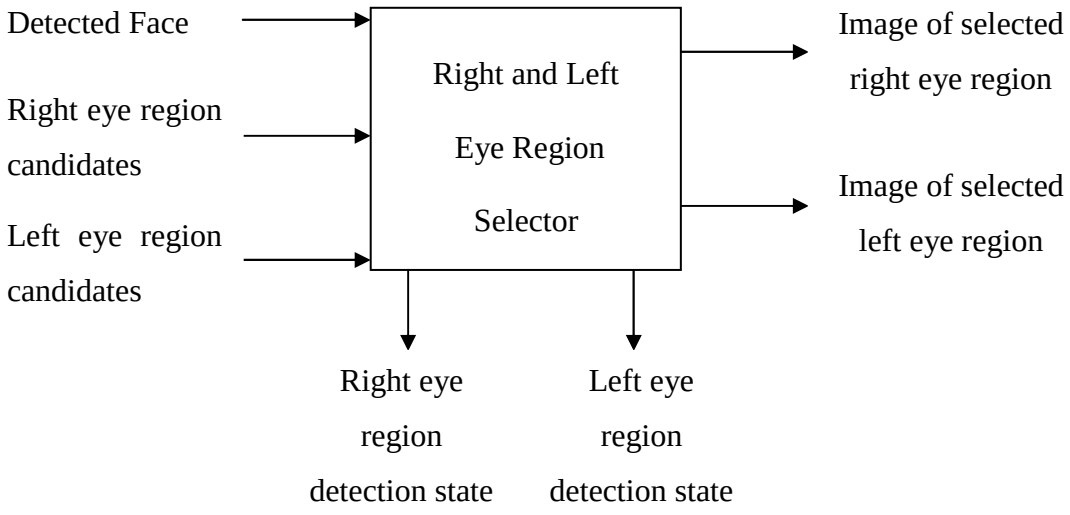

Figure 3.11: Right and Left Eye Region Selector module

As mentioned in section 3.3.1, the detected face region is used in order to select the valid right and left eye regions among candidates. Detected face region's width is w and height is h and the regions tagged as R and L are shown in Figure 3.12. R region is the appropriate region for the right eye and L is the appropriate region for the left eye. The right eye region candidates which do not belong to the region R will be eliminated and the left eye region candidates which do not belong to the region L will be eliminated.

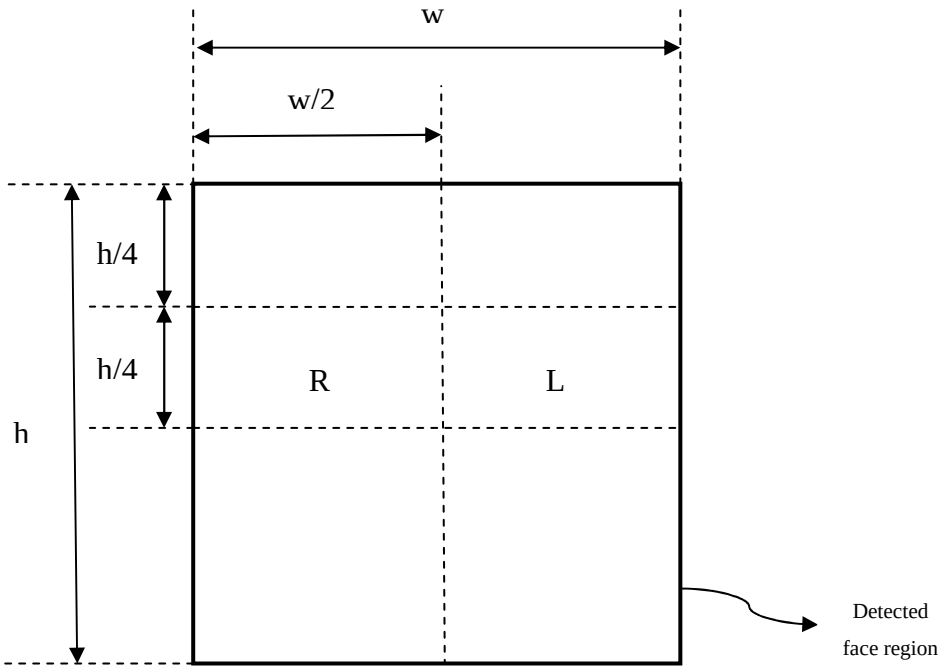

Figure 3.12: R and L regions of the detected face

The area of the detected face region is w×h.

The area of the detected eye region image is limited in order to avoid incorrect detections. After making empiric evaluations, maximum possible eye region area is found as:

Maximum possible eye region area = (w×h) ÷ 8

Flowchart for selecting valid right eye region among right eye candidates is shown in Figure 3.13.

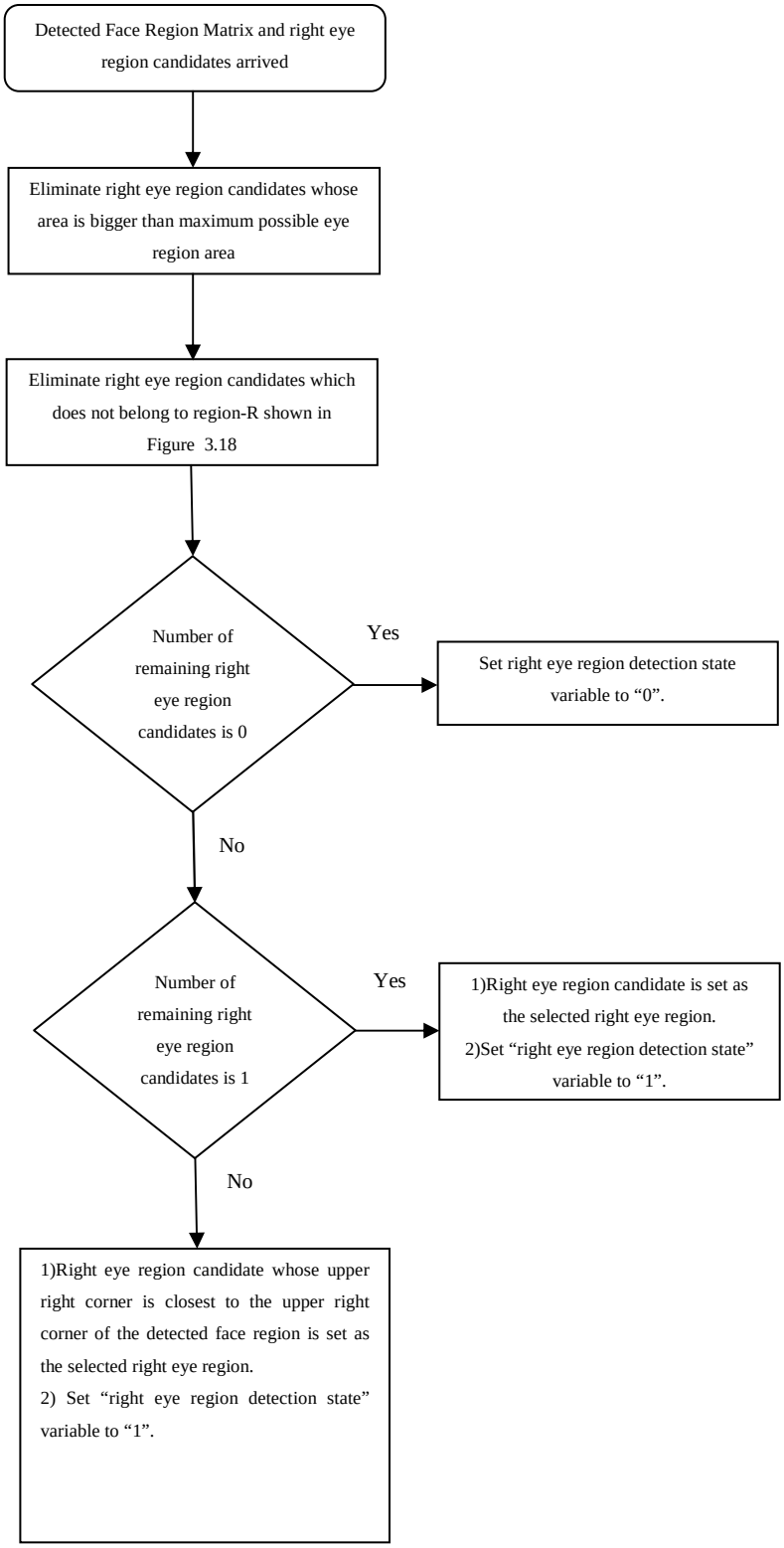

Figure 3.13: Selecting valid right eye region among candidates

As it is shown in Figure 3.13, the right eye region candidates whose area is bigger than the maximum possible eye region area((w×h)÷ 8) or the right eye region candidates whose center is not in region R are eliminated. If all of the right eye region candidates are eliminated, this means no valid right eye region could be found and the right eye detection state variable is set to “0”. If only one right eye region candidate remained after the elimination process, that right eye region candidate is set as the selected right eye region and the right eye region detection state variable is set to “1”. If two or more right eye region candidates remained after the elimination process, right eye region candidate whose upper right corner is closest to the upper right corner of the detected face region is set as the selected right eye region and the right eye region detection state variable is set to “1”.

An example of selecting valid right eye region among candidates for subject C is shown in Figure 3.14.

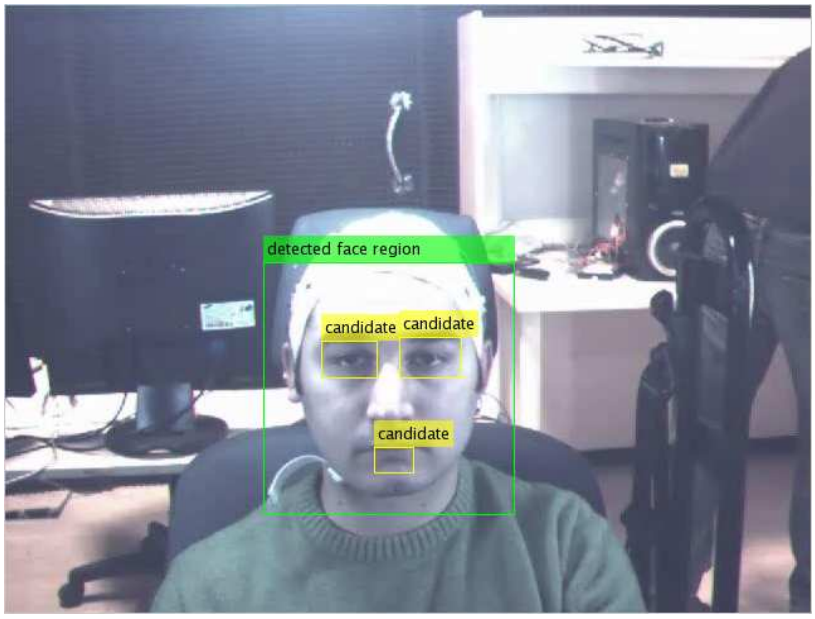

Figure 3.14: Detected face region and right eye region candidates

The coordinates of the upper right corner of the detected face region is (206,204), the width and height of the detected face is 200.
The coordinates of the critical points are shown in Figure 3.15.

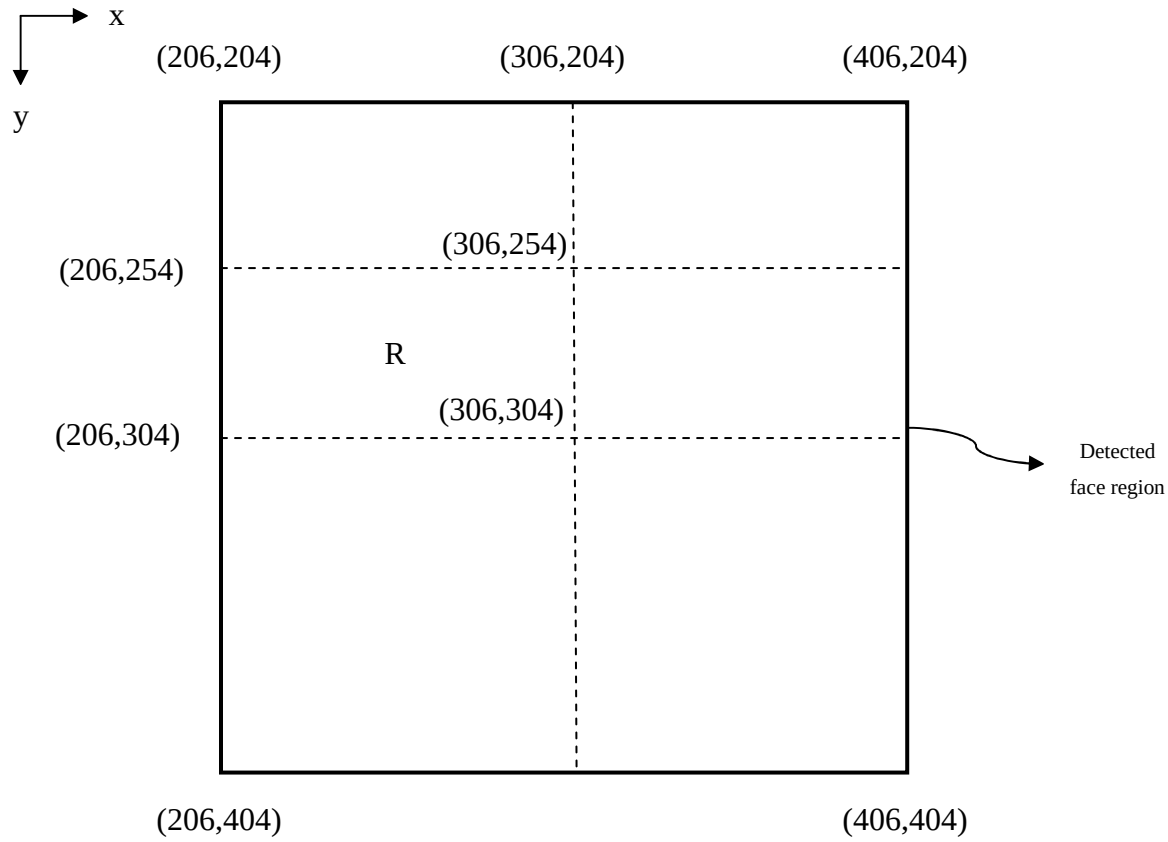

Figure 3.15: Coordinates of the critical points

R = {(x,y) | 206<x<306,254<y<304}

maximum possible eye region area = (w×h) ÷ 8 = 5000

There are 3 eye candidates:

Eye candidate 1: center = (274,281) & area = 1350

Eye candidate 2: center = (310,360) & area = 672

Eye candidate 3: center = (339,279) & area = 1650

All of right eye candidates' area is smaller than maximum possible eye region area.

None of the candidates is eliminated at this stage.

Only candidate 1 is in region R, so it is selected as the valid right eye region, it is shown in Figure 3.16.

Figure 3.16: Selected right eye region for the example in Figure 3.14

Some examples for selecting right eye region among candidates are shown in Figure 3.17, Figure 3.18 for subjects B and D, respectively.

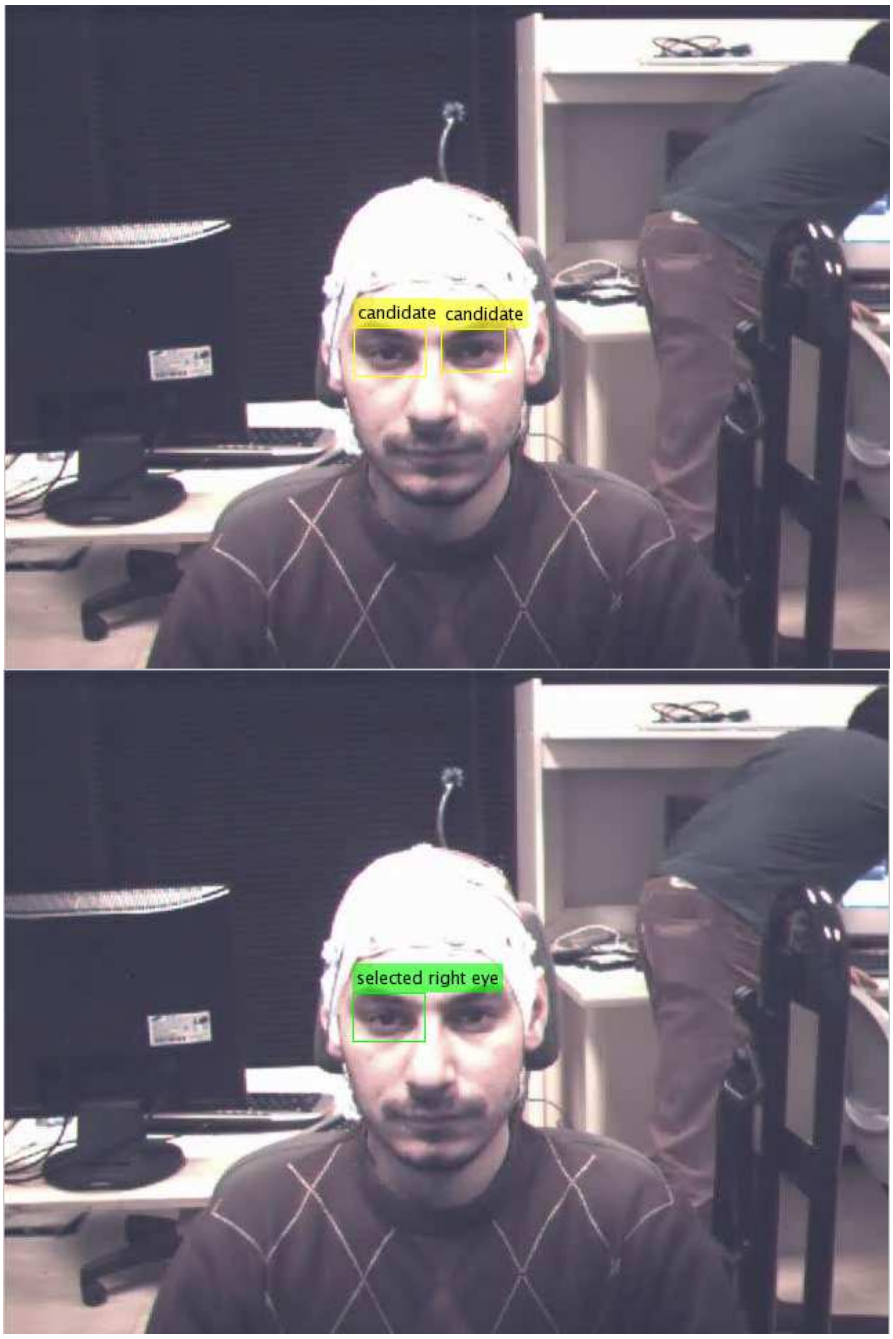

Figure 3.17: An example of valid right eye region selection among candidates for subject B

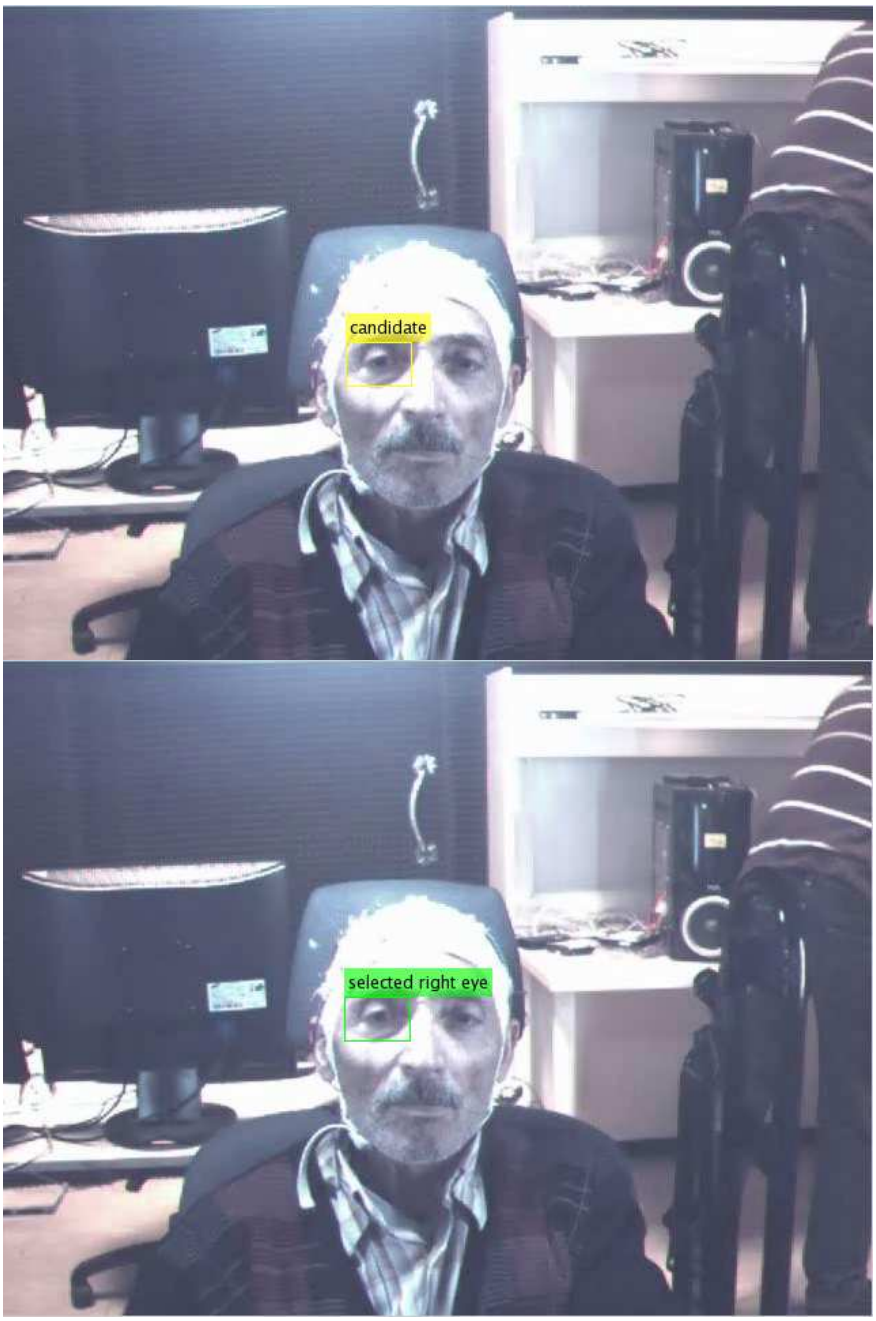

Figure 3.18: An example of valid right eye region selection among candidates for subject D

Selecting procedure for valid left eye region among candidates is the same as the procedure for right eye. Region L is used instead of region R and upper left corners are used instead of upper right corners.

An example of selecting valid left eye region among candidates for subject C is shown below:

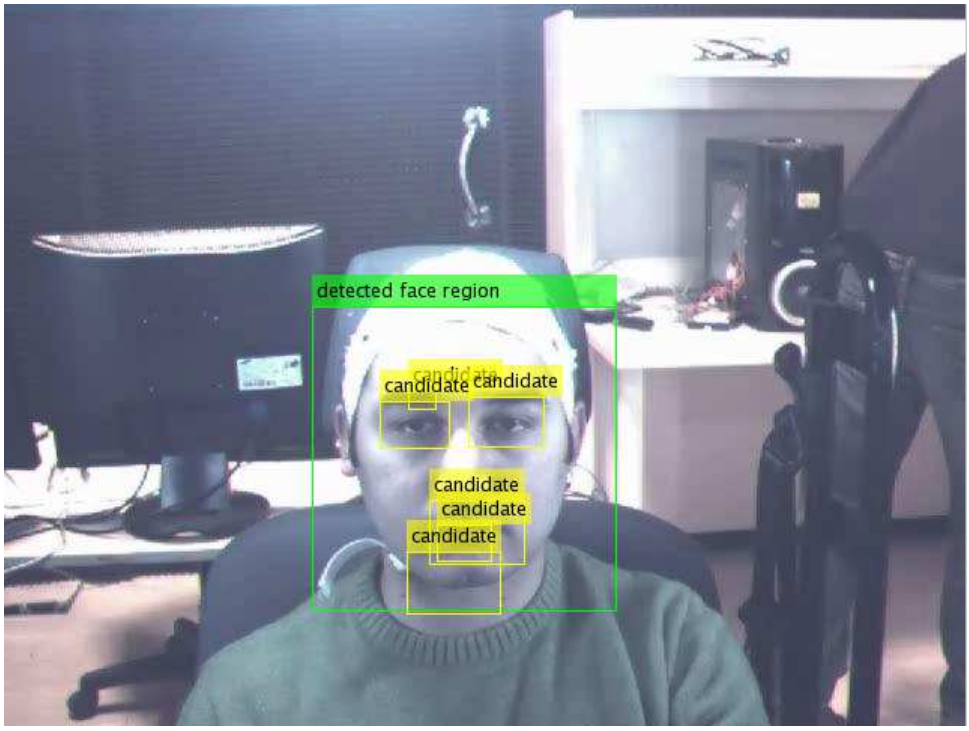

Figure 3.19: Detected face region and left eye region candidates

The coordinates of the upper right corner of the detected face region is (206,202), the width and height of the detected face is 203 both.

The coordinates of the critical points are shown in Figure 3.20.

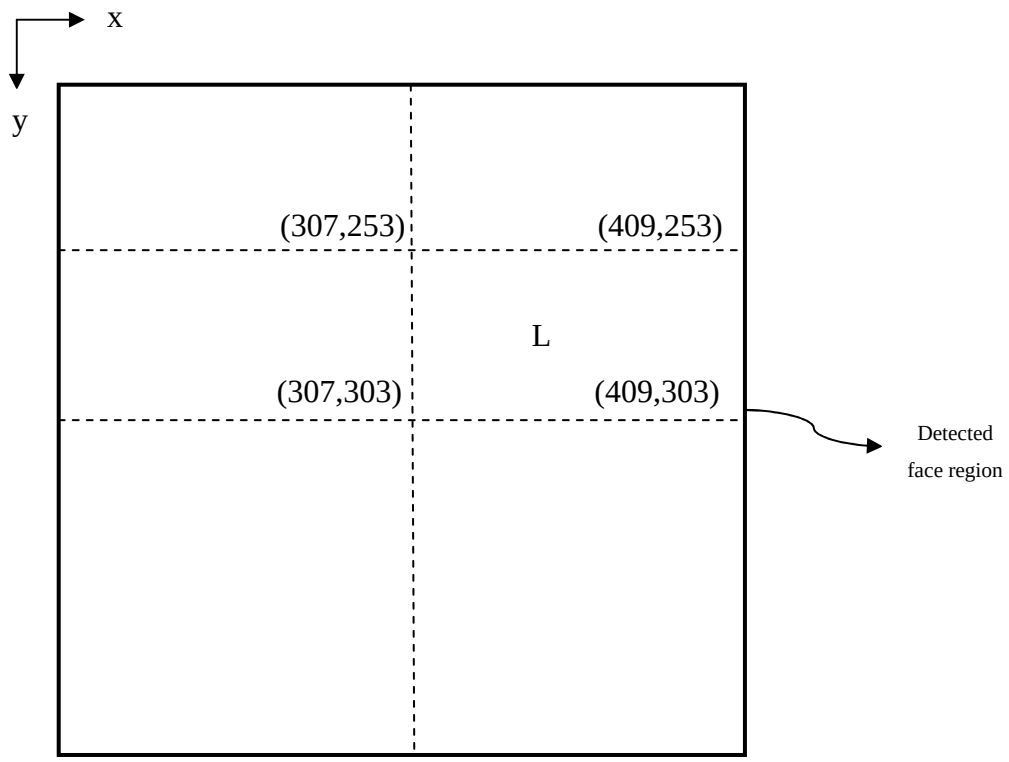

Figure 3.20: Coordinates of the critical points

L = {(x,y) | 307<x<409,253<y<303}

maximum possible eye region area = (w×h) ÷ 8 = 5151

There are 6 eye candidates:

Eye candidate 1: center = (279,264) & area = 247

Eye candidate 2: center = (274,281) & area = 1504

Eye candidate 3: center = (335,279) & area = 1734

Eye candidate 4: center = (307,359) & area = 925

Eye candidate 5: center = (300,386) & area = 2646

Eye candidate 6: center = (316,352) & area = 2752

All of right eye candidates' areas are smaller than maximum possible eye region area. None of the candidates is eliminated at this stage.

Only candidate 3 is in region L, so it is selected as the valid left eye region, which is shown in Figure 3.21.

Figure 3.21: Selected left eye region for the example in Figure 3.19

Some examples for selecting left eye region among candidates are shown in Figure 3.22, Figure 3.23 for subjects A and B, respectively.

Figure 3.22: An example of valid left eye region selection among candidates for subject A

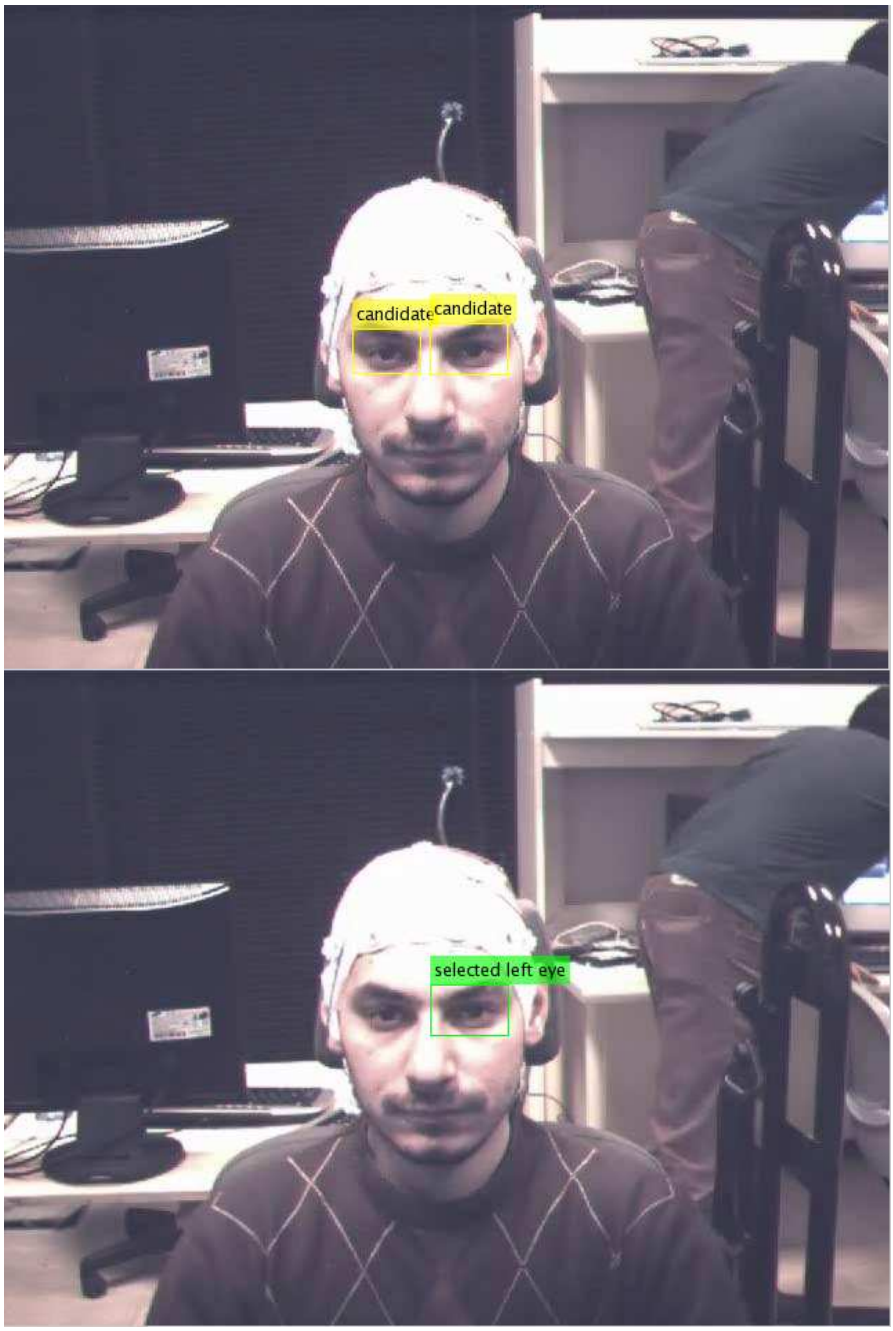

Figure 3.23: An example of valid left eye region selection among candidates for subject B

3.4 Modifying Eye Region Images for Neural Network

After the detection of right and left eye regions are completed and the selected right and left eye images are cropped from frames, we need to modify those eye region images. The part of the methodology performing this modification is called “Eye Region Image Modifier for Neural Networks” module.

The frames extracted from video segments we are working on, are in RGB format. Therefore, every pixel in these extracted frames have three channels. Memory and time required to train neural networks increase with increasing input size, so we need to minimize the input size by not decreasing the performance. Neural networks are models of biological neuron systems. When eye region image in RGB format and a gray-level eye region image are compared, RGB format image has negligible advantage in displaying the eye state. This situation can easily be observed in Figure 3.24. This is verified by tests on neural networks. In addition, the size of RGB images are three times the size of gray-level images, that’s why we work with gray-level images instead of images in RGB format.

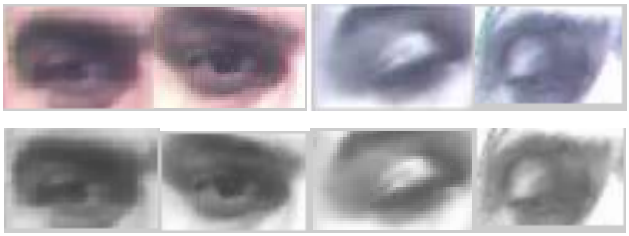

Figure 3.24: Selected eye region images(first row) and corresponding gray-level images(second row)

The eye regions detected have different sizes. That’s why the eye region images which are cropped from corresponding frames differ in size. However, the input of the neural networks must have a standard size. Generally, human eye regions have rectangular shape with 2÷3 as height-per-width ratio. As we mentioned above, we need to minimize the size of the input of the neural network, which is eye region images in our case. After minimization, eye region images must still include

enough information about the state of the eyes. Empirical analysis conducted on eye region images leads us to resize the eye region images to [12 18]. All of the eye region images to be input to neural networks are resized to [12 18].

Histogram equalization is useful to minimize the effects of different illumination between video segments. In order to see if this is true or not, the neural networks are trained by gray-level(not histogram equalized) and histogram equalized gray-level eye region images. The neural networks trained with histogram equalized gray-level eye region images give 2% better performance than the neural networks trained with gray-level(not histogram equalized) eye region images. This leads us to use histogram equalized gray-level eye region images. In addition, illumination conditions in the video segments used for training and testing for this decision are close to each other, if some video segments with more different illumination conditions are used, the performance gain of histogram equalization increases.
Block diagram of “Eye Region Image Modifier for Neural Networks” module is shown in Figure 3.25.

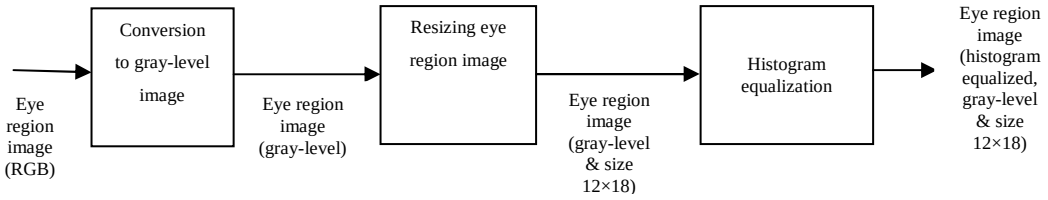

Figure 3.25: Eye Region Image Modifier for Neural Networks

Some examples of phases of “Eye Region Image Modifier for Neural Networks” for some right eye regions are shown in Figure 3.26. The eye images in the rows belong to subject A, B, C and D, respectively, and the columns are original eye region images, gray-level eye region images, gray-level eye region images with size 12×18 and histogram equalized gray-level eye region images with size 12×18, respectively. The size of the original eye regions detected are 33×49, 36×53, 32×48 and 38×58 for subjects A, B, C and D, respectively.

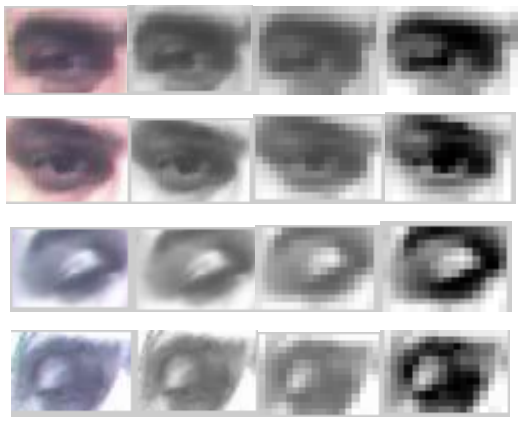

Figure 3.26: Some examples of phases of Eye Region Image Modifier for Neural Networks

3.5 Arranging the Ground Truth Data and the Eye Region Images Modified

In order to train neural networks, intensity values in the histogram equalized gray-level eye region images, each with size 12×18, and ground truth data are arranged. At the end of this arrangement process, they are ready to be used for training of neural network.

The intensity values of the eye region images are read and transferred to matrices with size 12×18. Then, these matrices are reshaped to 216×1, all of the matrices obtained after reshaping are concatenated and eye region images matrix, whose size is [216 900×N], is obtained.

Since there are 900 frames for each video segment, the size of the ground truth data is 900×1 for each video. The ground truth data for each video is transposed and matrices with size 1×900 are obtained after this process. These matrices with size 1×900 are concatenated and ground truth matrix which includes ground truth data for N videos is obtained. Its size is [1 900×N].

3.6 Training of Neural Networks

As it is shown in Figure 3.27, for each of our 4 subjects, two neural networks are trained; one for right eye and one for left eye.

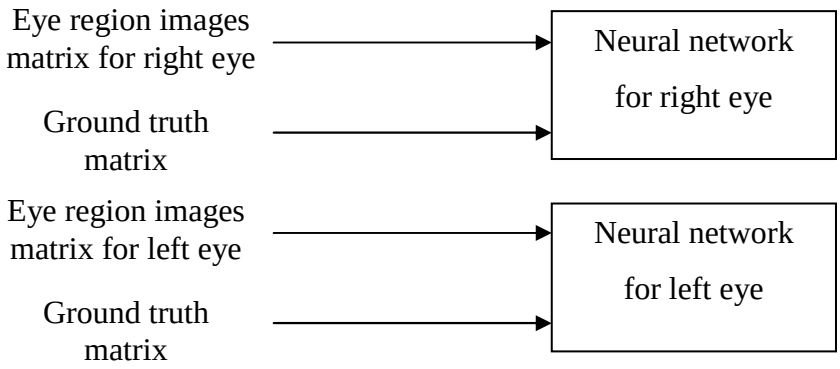

Figure 3.27: Training of neural networks for right and left eye

The time and memory required for training of neural networks increases with increasing neuron number. Therefore, we minimize the number of neurons which means minimizing both the number of layers and the number of neurons in each layer. The number of neurons in the input layer of our neural networks must be equal to the number of intensity values in eye region images. That's why, there are 216 neurons in the input layer of neural networks used. Since the output of neural networks for each eye region image is an estimation of eye state, there is 1 neuron in the output layer. Empirical analysis shows us that 1 hidden layer is enough for our case because performance of the neural networks we use shows negligible increment with increasing hidden layer number.

The neural networks we use are feedforward networks which suit the objective, deciding the state of an eye from eye region image. We tried to train neural networks by the backpropagation methods Levenberg-Marquardt, resilient and scaled conjugate gradient [32][33][40][41]. Levenberg-Marquardt backpropagation method has outperformed others by about 3%, this leads us to use this method in

training neural networks. Levenberg-Marquardt is very fast compared to the other training methods, however, as a disadvantage, its memory requirement is more than the other methods. Since our problem can be categorized as a nonlinear problem, we need to use nonlinear activation functions. This leads us to use hyperbolic tangent sigmoid function as the activation function of the neurons in the hidden layer. We use the linear transfer function as the activation function of the neurons in the output layer. Empirical analysis shows us that our networks need about 10-14 hidden neurons. Increasing the number of hidden neurons beyond this provides negligible increment in the performance of the neural networks.

We use the frames of 6 video segments in training the neural networks for subject A, 4 video segments for subject B, 5 video segments for subject C and 4 video segments for subject D. Since, every video has 900 frames, 5400, 3600, 4500 and 3600 frames are used for training for neural networks of subject A, B, C and D respectively.

The neural network of subject C is shown in Figure 3.28. It has 216 inputs, 12 hidden neurons (n=12) with hyperbolic tangent sigmoid function and one output neuron with linear transfer function as the activation function.

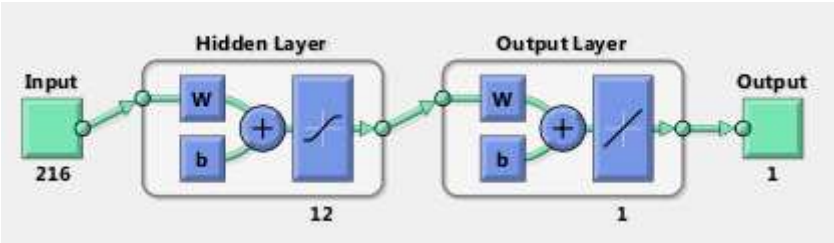

Figure 3.28: Neural network of subject C

During the training of neural networks, we randomly choose 60% of the input eye region images and use that portion for training. Randomly chosen 20% is used for validation process during training, remaining 20% is not used for training. In order to give an idea of the performance of the trained neural network, a micro test is done with this remaining 20%.

Regression plots obtained during the training of right eye neural network of subject C is shown in Figure 3.29. The results of the micro test done is seen on the third plot on the Figure 3.29. For the right eye region images whose ground truth is “0, which points out an open eye, all of the neural network outputs are less than 0.3, except for 2 samples. For the right eye region images whose ground truth is “0.5”, which means semi-closed eye, many of the estimations of the neural network are between 0.3 and 0.8. For the right eye region images whose ground truth is “1”, which means closed eye, many of the outputs are larger than 0.7.

After analyzing many regression plots and neural network estimations, it has been observed that generally the estimations for open eyes are less than 0.3. The estimations for semi-closed eyes are between 0.3 and 0.7 and the estimations for closed eyes are larger than 0.7.

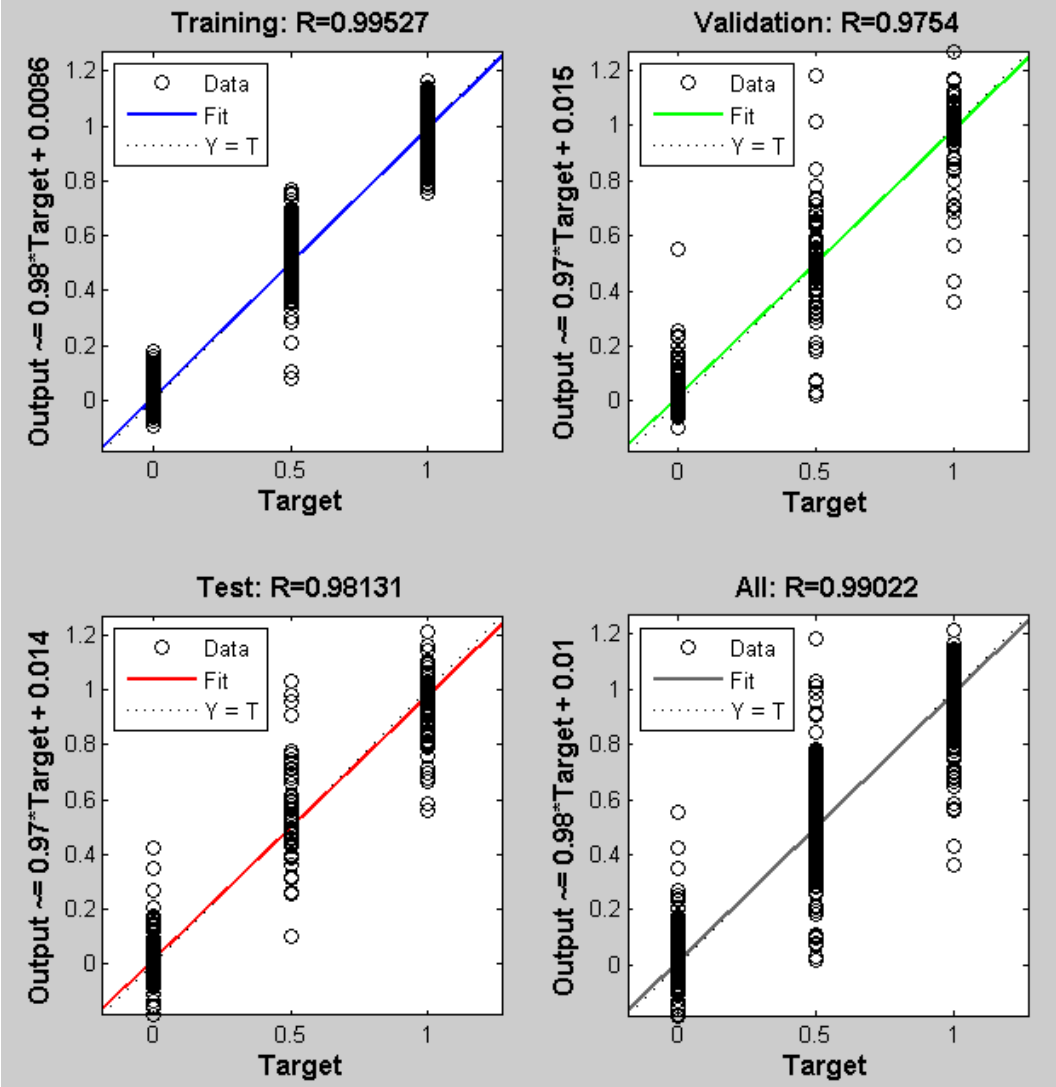

Figure 3.29: Regression plots obtained during the training of right eye neural network of subject C

Procedure for training a video is shown in Figure 3.30. First, the video is input to the Frame Extractor module followed by the analysis of the output 900 frames one by one and then each frame is labelled according to eye states as open, semi-closed or closed. Numerically, open is labelled as 0; semi-closed as 0.5 and closed as 1. These values form the ground truth for eye state for each frame. After the ground truth is ready for 900 frames, all of the frames are input to Right Eye Region Extractor and Left Eye Region Extractor modules and as the output of these

modules the histogram equalized gray-level images of right and left eyes with size [12 18] are obtained. The intensity values of each pixel in the obtained images with size [12 18] are transferred into a matrix with size [216 1]. For a 30-second video segment, we have 900 left eye images, 900 right eye images and ground truth matrix with size [1 900], which contains corresponding ground truth values. Reshaped eye image matrices with size [216 1] are concatenated and a matrix with size [216 900] is generated for a 30- second video segment. Every column of this matrix is actually the reshaped version of an eye image, which is the output of Right Eye Region Extractor or Left Eye Region Extractor modules. As we have mentioned, 6 video segments are used to train the neural networks of subject A. Each video segment has a right eye image matrix with size [216 900] and a left eye image matrix with the same size. In order to train the neural networks with 6 video segments, all of these right eye image matrices are concatenated and a matrix called eye region images matrix with size [216 5400] is obtained. The same process is carried out for ground truth values of the eye images and a ground truth matrix with size [1 5400] is obtained. Every column of eye region images matrix actually consists of the intensity values of right eye images cropped from original frames and every entry of ground truth matrix is the ground truth for that frame i.e. whether the subject's eyes are open (0), semi-closed (0.5) or closed (1) in the original frame. A feedforward backpropagation neural network is created. Eye region images matrix and ground truth matrix are input to the created neural network and the network is trained by Levenberg-Marquardt backpropagation method in which weights and bias values are updated according to Levenberg-Marquardt optimization [32] [33].

The same procedure is applied for left eye images and left eye image neural network is trained.

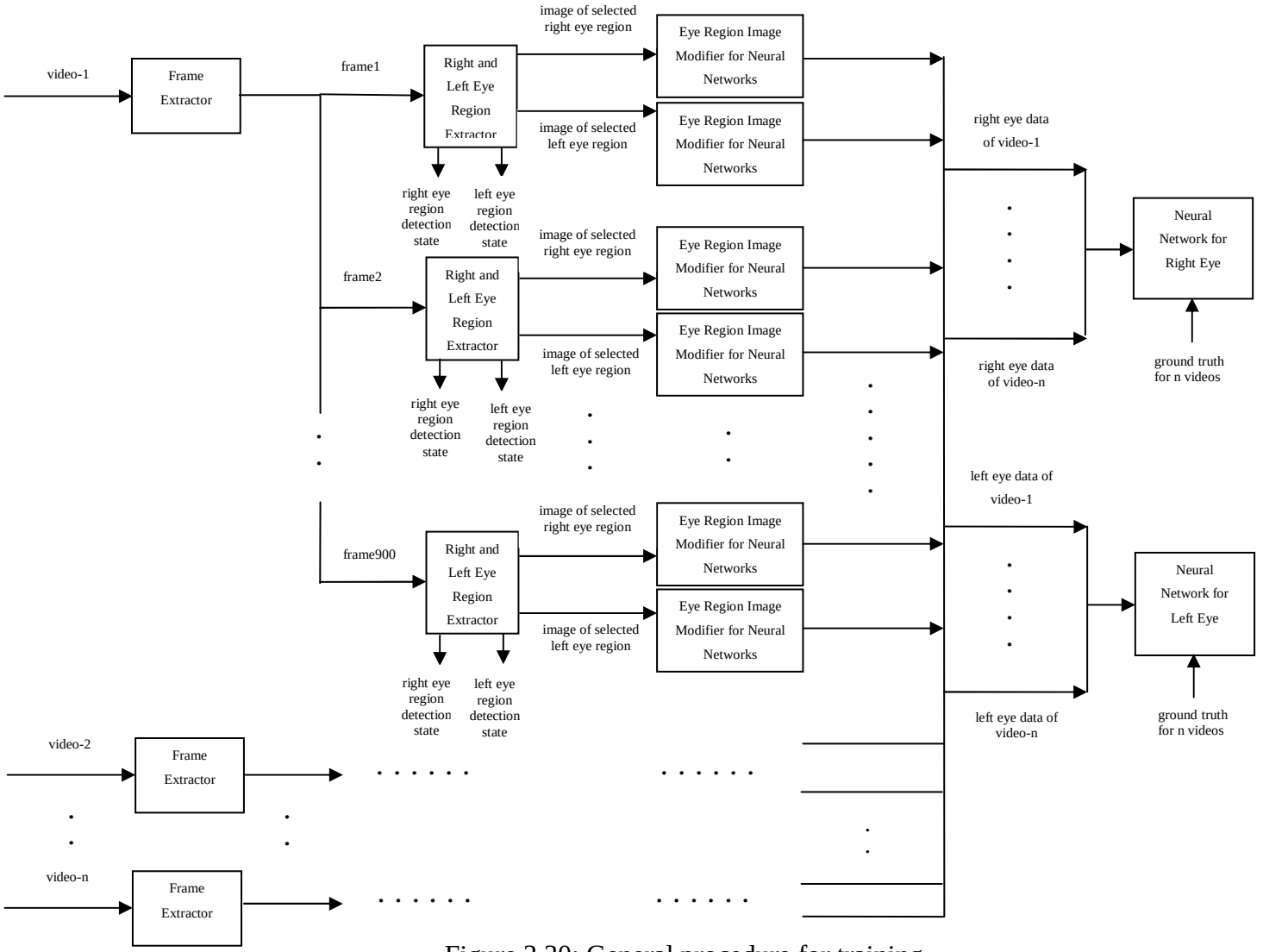

Figure 3.30: General procedure for training

3.7 Drowsiness Evaluator

As shown in Figure 3.1, outputs of neural network for right eye and neural network for left eye are input to a module called “Drowsiness Evaluator” block diagram of which is shown in Figure 3.31.

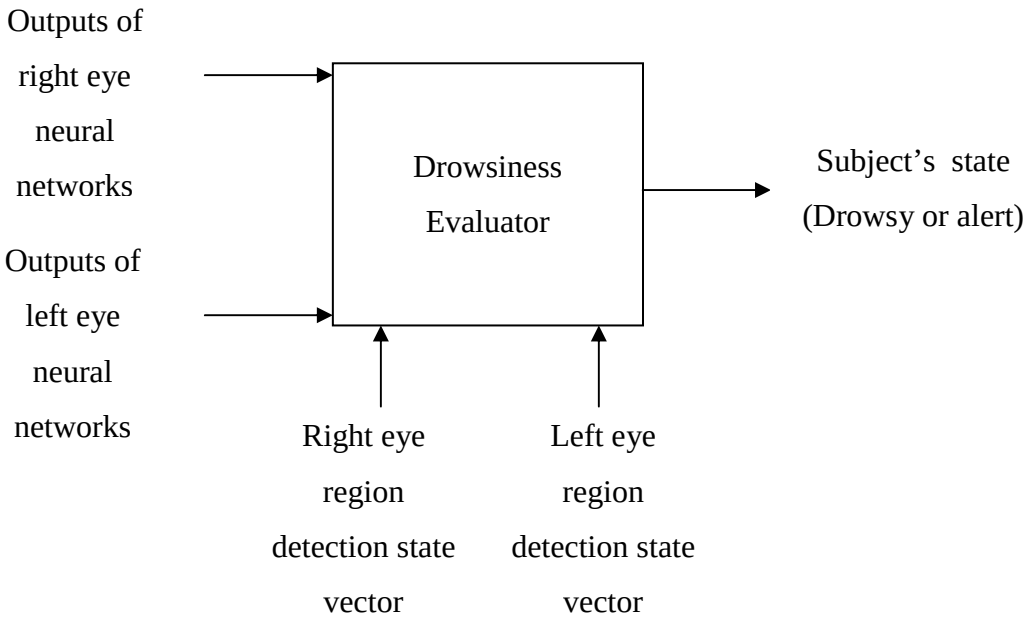

Figure 3.31: Block diagram of Drowsiness Evaluator module for a video segment

Outputs of right eye neural networks include the estimations of the right eye neural networks for every frame of the video segment to be tested and its size is 1×900. Outputs of left eye neural networks include the estimations of the left eye neural networks for each frame of the video segment to be tested and its size is 1×900.

Right eye region detection state vector includes the right eye region detection states for each frame of the video segment to be tested and its size is 1×900. The elements of this vector are taken from the outputs of Right and Left Eye Region Extractor module. Elements in this vector takes the values “1” if right eye region is successfully detected in the corresponding frame and takes the values “0” if right eye region could not be detected in the corresponding frame.

Left eye region detection state vector includes the left eye region detection states for each frame of the video segment to be tested and its size is 1×900. The elements of this vector are taken from the outputs of Right and Left Eye Region Extractor module. Elements in this vector takes the values “1” if left eye region is successfully detected in the corresponding frame and takes the values “0” if left eye region could not be detected in the corresponding frame.

As we have stated in the previous section, analysis on many regression plots and neural network estimations leads us to determine 0.3 and 0.7 as the threshold values for the digitization process. A neural network estimation less than 0.3 means that the neural network predicts an open eye; estimation between 0.3 and 0.7 means predicting a semi-closed eye and an estimation larger than 0.7 means predicting a closed eye.

After digitization process is completed, eye state estimation is done according to the flowchart shown in Figure 3.32. We have 900 frames in the video segment to be tested and at the end of the eye state estimation process given in this flowchart, valid eye state estimation could not be performed for some of the frames. In other words, some frames are eliminated and we could not get any valid information about the state of the eye in those frames. However, valid eye state estimation could be performed for many of the frames and that’s enough to detect whether the subject is drowsy or alert.

After eye state estimation process is completed for all of the frames of the video segment, each frame is tagged as open (0), semi-closed (0.5), closed (1) and “no valid estimation”. The mean of the eye states for which valid estimation could be performed is taken and this value is named as “average eye state point”. For

drowsy cases, average eye state point exceeds a threshold value whereas for alert videos it does not exceed that threshold value. Our observations leads us to set this threshold value as “0.18”. Video segments whose average eye state point exceeds

0.18 are detected as drowsy and video segments whose average eye state point does not exceed 0.18 are detected as alert.

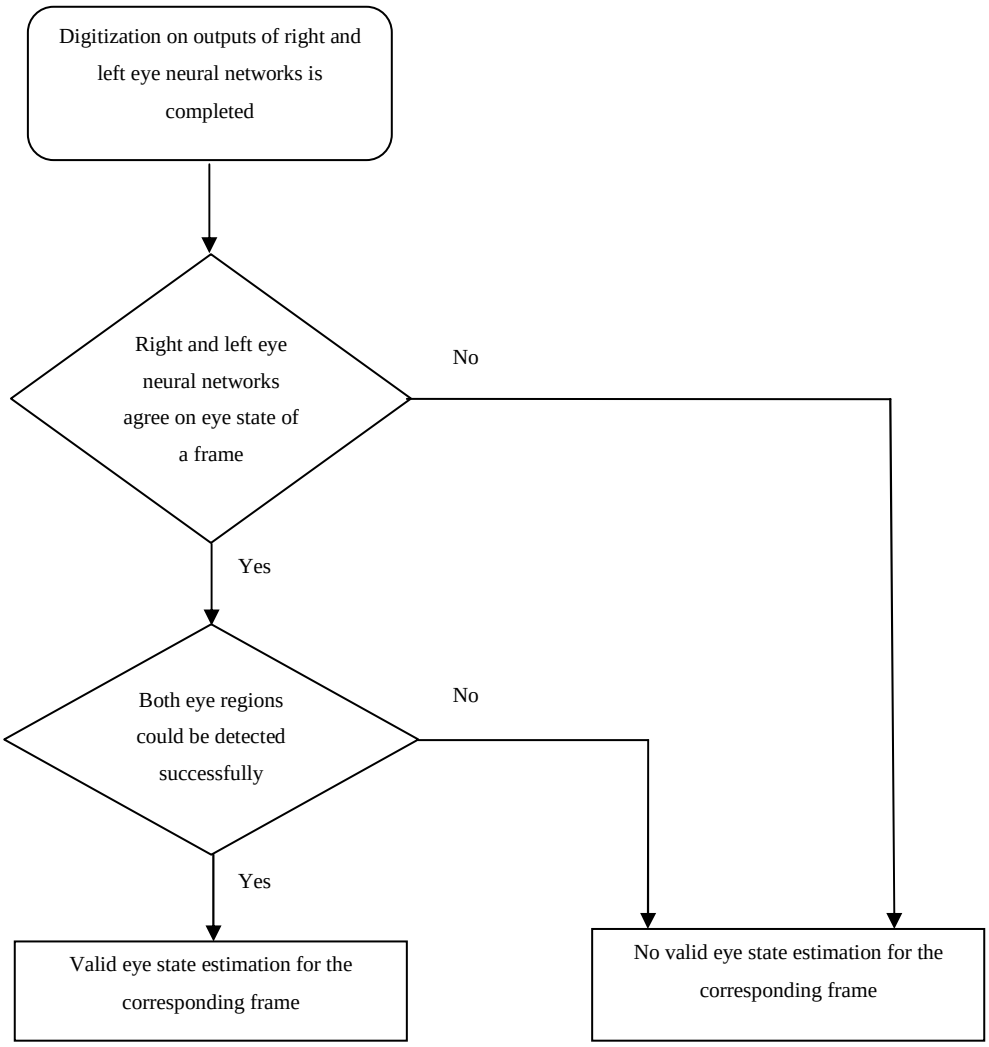

Figure 3.32: Flowchart for eye state estimation procedure for a single frame

CHAPTER 4

EXPERIMENTS AND RESULTS

4.1 Video Data Used in Experiments and Forming the Ground Truth for Drowsiness

The database used to test the method we propose, is called UYKUCU database and all of the video segments and frames are taken from UYKUCU database [48]. In this database, subjects have driven a virtual car simulator which displays the driver's view of a car through a computer terminal. An open source multiplatform video game[1] and a steering wheel[2] constitutes the interface with the simulator. The video game was maintained such that at random times, a wind effect was applied that dragged the car to the right or left which forces the subject to correct the positon of the vehicle. This manipulation type had been found in the past to increase fatigue [42]. Driving speed is constant. Each of the four subjects performed the driving task over three hours beginning at midnight. The subjects fell asleep many times. Video of the subjects' faces was recorded using a digital video camera which is 480×640 and 30 fps. The video segments are tagged as drowsy or alert. Alert video segments are taken from the first ten minutes of the driving task. Drowsy video segments are tagged by analyzing the condition of the driver. Every video segment is 30 seconds long. For each subject, the number of video segments tagged as ground truth is shown in Table 4.1.

[1] Torcs
[2] ThrustMaster steering wheel

Table 4.1: The number of video segments tagged as ground truth for each subject

Subject	Drowsy	Alert
A	9	13
B	25	16
C	28	14
D	14	13

4.2 Forming the Ground Truth for Eye States

We formed ground truth for the eye states for all of the 4 subjects. There are three eye states: open, semi-closed and closed. Corresponding state value is assigned to “0” for eyes which are in open state, “0.5” for eyes which are in semi-closed state and “1” for eyes which are in closed state. Some examples for subject A are shown in Figure 4.1.

Figure 4.1-a: Examples of eyes in open state for subject A

Figure 4.1-b: Examples of eyes in semi-closed state for subject A

Figure 4.1-c: Examples of eyes in closed state for subject A

Figure 4.1: Examples of eyes in open, semi-closed and closed state for subject A

Some examples for subject B are shown in Figure 4.2.

Figure 4.2-a: Examples of eyes in open state for subject B

Figure 4.2-b: Examples of eyes in semi-closed state for subject B

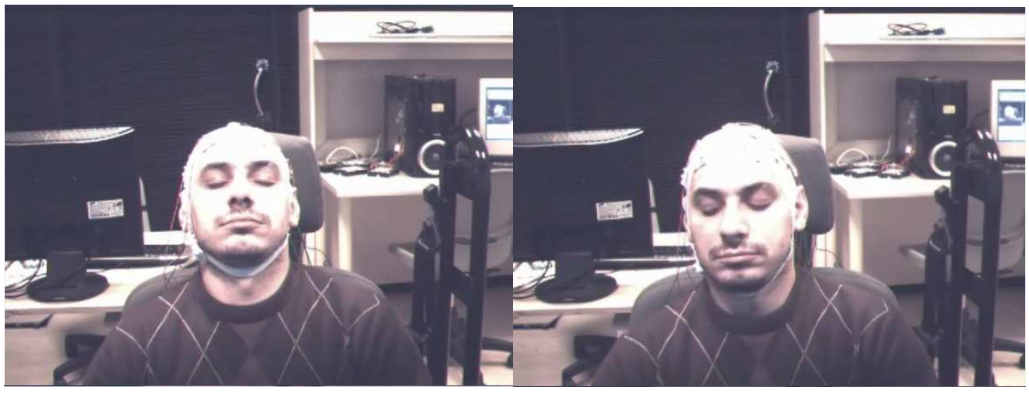

Figure 4.2-c: Examples of eyes in closed state for subject B

Figure 4.2: Examples of eyes in open, semi-closed and closed state for subject B

Some examples for subject C are shown in Figure 4.3.

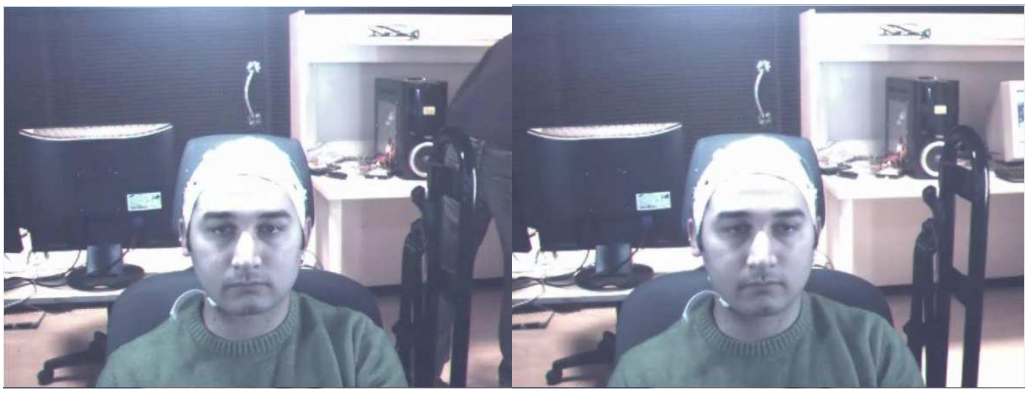

Figure 4.3-a: Examples of eyes in open state for subject C

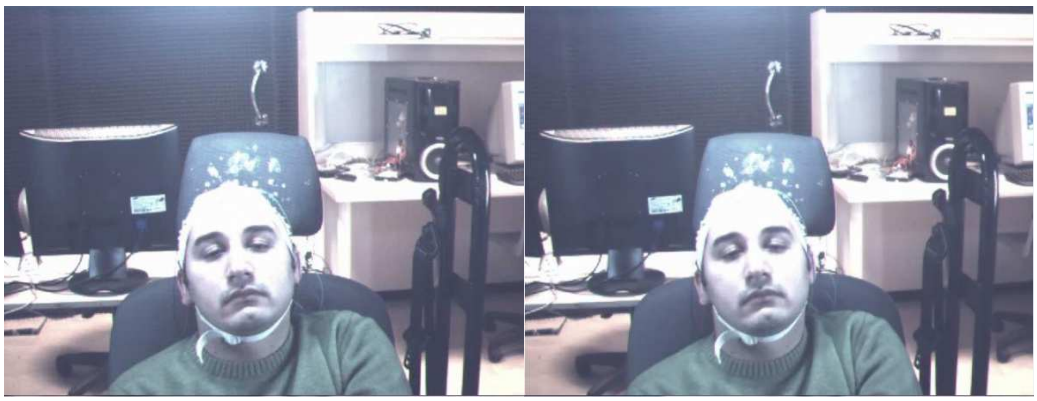

Figure 4.3-b: Examples of eyes in semi-closed state for subject C

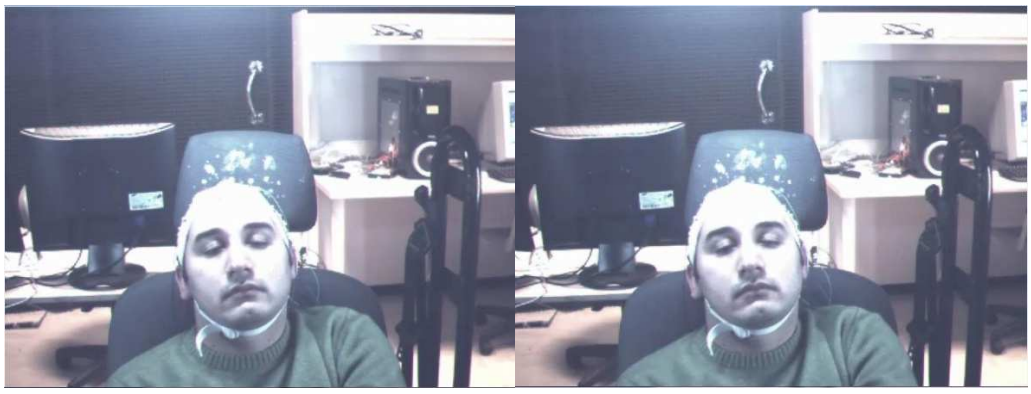

Figure 4.3-c: Examples of eyes in closed state for subject C

Figure 4.3: Examples of eyes in open, semi-closed and closed state for subject C

Some examples for subject D are shown in Figure 4.4.

Figure 4.4-a: Examples of eyes in open state for subject D

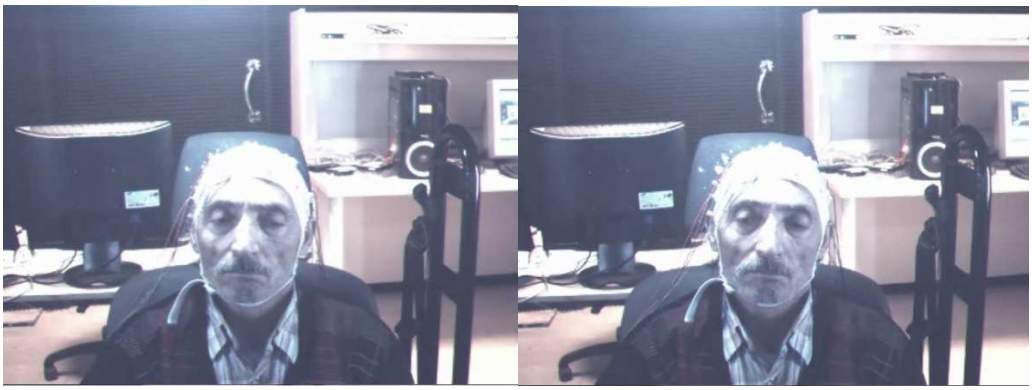

Figure 4.4-b: Examples of eyes in semi-closed state for subject D

Figure 4.4-c: Examples of eyes in closed state for subject D

Figure 4.4: Examples of eyes in open, semi-closed and closed state for subject D

Since the main objective of this thesis is detecting drowsiness, we just need to form eye state ground truth for the video segments we use for training. We do not need to form eye state ground truth for every video segment. However, we desired to analyze the success rate of our eye state estimation, as well. That's why, in addition to the frames we used for training neural networks, we formed eye state ground truth for 2700, 3600, 4500 and 3600 frames for subjects A, B, C and D, respectively.

4.3 Results of the Experiments

As we mentioned in section 3.6, we use the frames of 6 video segments in training the neural networks for subject A, 4 video segments for subject B, 5 video segments for subject C and 4 video segments for subject D. Since, every video has 900 frames, 5400, 3600, 4500 and 3600 frames are used for training for neural networks of subject A, B, C and D, respectively. The right and left eye region neural networks of each subjects are trained according to the method we explained in detail in section 3.6. For the method we propose, it takes 3.3 seconds to classify a 30-second-long video segment as alert or drowsy.

4.3.1 Results of the Method for Within Subject Recognition

In within subject recognition, the system is trained with the video segments of the subject whose video segments are going to be tested. Our eye state detection method is tested on 2700, 3600, 4500 and 3600 frames for subjects A, B, C and D, respectively.

The results of eye state tests performed on subject A are listed in Table 4.2 and 4.3. The eye state estimations and corresponding ground truth values of subject A are listed in Table 4.2. For subject A, the number of frames for which estimation could and could not be performed is listed in Table 4.3.

Table 4.2: Eye state estimations of subject A in within subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1756	41	0
Semi-closed eye	119	72	79
Closed eye	15	63	104

Table 4.3: Eye state estimation ratio of subject A in within subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1890	176	183
The number of frames for which estimation could not be performed	310	129	12
Total number of frames	2200	305	195

The results of eye state tests performed on subject B are listed in Table 4.4 and 4.5. The eye state estimations and corresponding ground truth values of subject B are listed in Table 4.4. For subject B, the number of frames for which estimation could and could not be performed is listed in Table 4.5.

Table 4.4: Eye state estimations of subject B in within subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1715	3	0
Semi-closed eye	0	0	24
Closed eye	0	1	1339

Table 4.5: Eye state estimation ratio of subject B in within subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1715	4	1363
The number of frames for which estimation could not be performed	76	2	440
Total number of frames	1791	6	1803

The results of eye state tests performed on subject C are listed in Table 4.6 and 4.7 The eye state estimations and corresponding ground truth values of subject C are listed in Table 4.6. For subject C, the number of frames for which estimation could and could not be performed is listed in Table 4.7.

Table 4.6: Eye state estimations of subject C in within subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1634	15	0
Semi-closed eye	1	4	6
Closed eye	0	1	1723

Table 4.7: Eye state estimation ratio of subject C in within subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1635	20	1729
The number of frames for which estimation could not be performed	91	47	978
Total number of frames	1726	67	2707

The results of eye state tests performed on subject D are listed in Table 4.8 and 4.9. The eye state estimations and corresponding ground truth values of subject D are listed in Table 4.8. For subject D, the number of frames for which estimation could and could not be performed is listed in Table 4.9.

Table 4.8: Eye state estimations of subject D in within subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1785	49	12
Semi-closed eye	71	39	110
Closed eye	17	73	736

Table 4.9: Eye state estimation ratio of subject D in within subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1873	161	858
The number of frames for which estimation could not be performed	122	98	488
Total number of frames	1995	259	1346

The results of eye state tests performed on all subjects are listed in Table 4.10 and 4.11. The eye state estimations and corresponding ground truth values are listed in Table 4.10. The number of frames for which estimation could and could not be performed is listed in Table 4.11.

Table 4.10: Eye state estimations of all subjects in within subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	6890	108	12
Semi-closed eye	191	115	219
Closed eye	32	138	3902

The accuracy of our method's eye state estimations is 96.7% for open eyes, 94.4% for closed eyes and 31.9% for semi-closed eyes. Since semi-closed is a transient state between open and closed states, there are low number of frames in which eyes are in semi-closed state. That means low number of semi-closed states for training neural networks and neural networks cannot learn well with low number of examples. That's the main reason for low accuracy in semi-closed eyes. Another reason is that it is difficult to classify an eye as semi-closed since a semi-closed eye is both near to an open eye and a closed eye. While forming the ground truth for eye states, we had difficulty in classifiying semi-closed eyes. High accuracy in estimations of open and closed eyes means our method is useful and accuracy on semi-closed eyes can be raised by increasing the number of semi-closed samples used for training.

The accuracy of our method's eye state estimation is 94%. The results are convincing when compared to the other studies in the literature. In [22], support vector machine (SVM) is used to classify the eyes as open and closed. The accuracy of the method proposed in [22] is 90.4% and our eye state detection method is more accurate than that method. In [43], eyes are classified as open and closed according to geometrical computations performed and 94% accuracy is obtained. In [23], Flores et al. uses SVM and classify eyes as open or closed and 95.1% accuracy is obtained. Unlike most studies in the literature, we categorized

eyes into 3 states, which makes our approach a more realistic one and our task a more challenging one.

Table 4.11: Eye state estimation ratio of all subjects in within subject

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	7113	361	4133
The number of frames for which estimation could not be performed	599	276	1918
Total number of frames	7712	637	6051

Eye estimation is performed for 80.6% of the frames, which means about every 20 frames out of 100 frames are discarded in deciding whether the subject is drowsy or alert. Since video segments are 30 fps, this will not be an obstacle in predicting drowsiness. For any one second period, we have 24 frames, in average, to be used in drowsiness detection.

For subject A, 10 alert and 6 drowsy videos are used for testing. Our method estimates all of the videos correctly except for 1 alert video segment. For subject B, 14 alert and 23 drowsy videos are used for testing. Our method estimates all of the videos correctly. For subject C, 12 alert and 25 drowsy videos are used for testing. Our method estimates all of the videos correctly. For subject D, 11 alert and 12

drowsy videos are used for testing. Our method estimates all of the videos correctly.

Totally, 47 alert videos and 66 drowsy videos are used for testing as seen on Table 4.12. The method makes accurate estimations for all of the video segments except for 1 alert video segment.

Table 4.12: Results of drowsiness detection for all of the subjects in within subject recognition

Estimation	Ground Truth	
	Alert	Drowsy
Alert	46	-
Drowsy	1	66

Drowsiness detection accuracy of our method is 99.1%. This is a high and convincing result. The study in [44] is tested on the same database with our method [48]. In this study, facial action coding system is used and facial actions are used as indicators of drowsiness. They encode facial actions by making use of a robust tool called computer expression recognition toolbox which they have trained with many subjects and they obtain drowsiness detection accuracy 99% [28]. The method we propose has the same accuracy with this method.

4.3.2 Results of the Method for Across Subject Recognition

In across subject recognition, the system is trained with all of the video segments of the subjects except for the subject whose video segments are going to be tested. Our eye state detection method is tested on 2700, 3600, 4500 and 3600 frames for subjects A, B, C and D, respectively.

The results of eye state tests performed on subject A are listed in Table 4.13 and 4.14. The eye state estimations and corresponding ground truth values of subject A are listed in Table 4.13. For subject A, the number of frames for which estimation could and could not be performed is listed in Table 4.14.

Table 4.13: Eye state estimations of subject A in accross subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1348	11	0
Semi-closed eye	222	105	10
Closed eye	221	119	173

Table 4.14: Eye state estimation ratio of subject A in accross subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1791	235	183
The number of frames for which estimation could not be performed	409	70	12
Total number of frames	2200	305	195

The results of eye state tests performed on subject B are listed in Table 4.15 and 4.16. The eye state estimations and corresponding ground truth values of subject B are listed in Table 4.15. For subject B, the number of frames for which estimation could and could not be performed is listed in Table 4.16.

Table 4.15: Eye state estimations of subject B in accross subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	1591	0	20
Semi-closed eye	2	2	796
Closed eye	0	0	72

Table 4.16: Eye state estimation ratio of subject B in accross subject

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1593	2	888
The number of frames for which estimation could not be performed	198	4	915
Total number of frames	1791	6	1803

The results of eye state tests performed on subject C are listed in Table 4.17 and 4.18 The eye state estimations and corresponding ground truth values of subject C are listed in Table 4.17. For subject C, the number of frames for which estimation could and could not be performed is listed in Table 4.18.

Table 4.17: Eye state estimations of subject C in accross subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	945	10	5
Semi-closed eye	28	6	131
Closed eye	0	0	1594

Table 4.18: Eye state estimation ratio of subject C in accross subject

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	973	17	1730
The number of frames for which estimation could not be performed	753	51	977
Total number of frames	1726	67	2707

The results of eye state tests performed on subject D are listed in Table 4.19 and 4.20. The eye state estimations and corresponding ground truth values of subject D are listed in Table 4.19. For subject D, the number of frames for which estimation could and could not be performed is listed in Table 4.20.

Table 4.19: Eye state estimations of subject D in accross subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	2	5	11
Semi-closed eye	983	79	452
Closed eye	135	77	393

Table 4.20: Eye state estimation ratio of subject D in accross subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	1873	161	856
The number of frames for which estimation could not be performed	122	98	490
Total number of frames	1995	259	1346

The results of eye state tests performed on all subjects are listed in Table 4.21 and 4.22. The eye state estimations and corresponding ground truth values are listed in Table 4.21. The number of frames for which estimation could and could not be performed is listed in Table 4.22.

Table 4.21: Eye state estimations of all subjects in accross subject recognition

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	3886	26	36
Semi-closed eye	1235	192	1389
Closed eye	356	196	2232

The accuracy of our method's eye state estimation is 66.1% in across subject recognition. The reason for low accuracy in across subject recognition is eye shapes being different from person to person and 3 subjects' video segments are not enough in order to train neural networks for a different eye shape. As the eye shape of subject D is too different from the eye shapes of the other subjects, the accuracy for subject D is too low. When tests performed on subject D are not taken into account, eye state estimation accuracy is 78.7%.

Table 4.22: Eye state estimation ratio of all subjects in accross subject recognition

	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
The number of frames for which estimation could be performed	5477	414	3657
The number of frames for which estimation could not be performed	2235	223	2394
Total number of frames	7712	637	6051

Eye estimation is performed for 66.3% of the frames, which means about every 34 frames out of 100 frames are discarded in deciding whether the subject is drowsy or alert. Since video segments are 30 fps, this will not be an obstacle in predicting drowsiness. For any one second period, we have 20 frames, in average, to be used in drowsiness detection.

For subject A, 10 alert and 6 drowsy videos are used for testing. Our method estimates all of the videos correctly except for 2 alert video segments. For subject B, 14 alert and 23 drowsy videos are used for testing, and our method estimates all of the videos correctly. For subject C, 12 alert and 25 drowsy videos are used for testing. Our method estimates all of the videos correctly except for 1 drowsy video segment. For subject D, 11 alert and 12 drowsy videos are used for testing. Our method estimates all of the drowsy videos correctly, however, estimates all of the alert video segments as drowsy.

Totally, 47 alert videos and 66 drowsy videos are used for testing as seen on Table 4.23.

Table 4.23: Results of drowsiness detection for all of the subjects in accross subject recognition

Estimation	Ground Truth	
	Alert	Drowsy
Alert	34	1
Drowsy	13	65

Drowsiness detection accuracy of our method is 87.6% in across subject recognition. As we mentioned, the eye shape of subject D is too different from the eye shapes of the other subjects. When tests performed on the subject D are not taken into account, drowsiness detection accuracy of our method is 96.7%.

In order to achieve a high and convincing result in across subject recognition, neural networks need to be trained with many subjects with various eye shapes. The robustness and accuracy of the method will increase with increasing number of subjects used in training. As we mentioned in section 4.3.1, the study in [44] is tested on the same database we use and in this study, facial action coding system is used and facial actions are used as indicators of drowsiness. In order to encode facial actions, they use a robust tool called computer expression recognition toolbox which they have trained with many subjects. That’s the reason for them obtaining higher results compared to our results in drowsiness detection for across subject recognition.

4.3.3 Gain of Combining the Estimations for Both Eyes

In this section, we investigate the gain of combining the estimations for both eyes in within subject recognition. The results of eye state tests when only right eyes of the subjects are considered, are listed in Table 4.24.

Table 4.24: Eye state estimations when only right eyes are considered

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	7239	136	7
Semi-closed eye	305	163	133
Closed eye	68	137	3431

The accuracy of eye state estimation when only right eyes of the subjects are considered is 93.2%. However, as mentioned in section 4.3.1, this accuracy is 94% when both eyes are considered and the estimations are combined.

Totally, 47 alert videos and 66 drowsy videos are used for testing as seen on Table 4.25.

Table 4.25: Results of drowsiness detection when only right eyes are

Estimation	Ground Truth	
	Alert	Drowsy
Alert	44	0
Drowsy	3	66

When only right eyes are considered drowsiness detection accuracy is 97.3%. However, as mentioned in section 4.3.1, this accuracy is 99.1% when both eyes are considered and the estimations are combined.

The results of eye state tests when only left eyes of the subjects are considered, are listed in Table 4.26.

Table 4.26: Eye state estimations when only left eyes are considered

Estimation of the method we propose	Ground Truth		
	Open eye	Semi-closed eye	Closed eye
Open eye	7013	158	29
Semi-closed eye	345	190	277
Closed eye	29	154	2903

The accuracy of eye state estimation when only left eyes of the subjects are considered is 91.1%. However, as mentioned in section 4.3.1, this accuracy is 94% when both eyes are considered and the estimations are combined.

Totally, 47 alert videos and 66 drowsy videos are used for testing as seen on Table 4.27.

Table 4.27: Results of drowsiness detection when only left eyes are considered

Estimation	Ground Truth	
	Alert	Drowsy
Alert	42	0
Drowsy	5	66

When only left eyes are considered drowsiness detection accuracy is 95.6%. However, as mentioned in section 4.3.1, this accuracy is 99.1% when both eyes are considered and the estimations are combined.

Drowsiness detection accuracies and eye state detection accuracies are shown in Table 4.28. Combining the estimations for right and left eyes increases drowsiness detection accuracy by about 3% and eye state detection accuracy by about 2%. Since combination of the estimations for right and left eyes is not a common method used in the literature, increasing the accuracy with this method is a contribution of our proposed algorithm.

Table 4.28: Gain of combining the estimations for right and left eyes

Eyes Considered in Estimation Process	Drowsiness Detection Accuracy(%)	Eye State Detection Accuracy(%)
Right Eyes	97.3	93.2
Left Eyes	95.6	91.1
Both Eyes	99.1	94

4.3.4 Advantage of Using Three Eye States

We have assigned three states to eyes, however, semi-closed eyes are counted as open eyes in most studies in the literature. In this section, we investigate the advantage of using three eye states instead of using two.

The results of eye state tests when semi-closed state is cancelled, are listed in Table 4.29 and 4.30. The eye state estimations and corresponding ground truth values are listed in Table 4.29. The number of frames for which estimation could and could not be performed is listed in Table 4.30.

Table 4.29: Eye state estimations in two eye state case

Estimation of the method we propose	Ground Truth	
	Open eye	Closed eye
Open eye	7734	66
Closed eye	31	3527

The accuracy of our method's eye state estimation is 99.1%. This result is higher than the result obtained for three eye states case, which is 94%.

Table 4.30: Eye state estimation ratio in two eye state case

	Ground Truth	
	Open eye	Closed eye
The number of frames for which estimation could be performed	7765	3593
The number of frames for which estimation could not be performed	584	2458
Total number of frames	8349	6051

Eye estimation is performed for 78.9% of the frames.

Totally, 47 alert videos and 66 drowsy videos are used for testing as seen on Table 4.31.

Table 4.31: Results of drowsiness detection in two eye state case

Estimation	Ground Truth	
	Alert	Drowsy
Alert	47	7
Drowsy	47	59

Drowsiness detection accuracy is 93.8% for two eye state case, which means that the result for drowsiness detection is more accurate when semi-closed state is counted as a an eye state. Since the eye state detection accuracy is higher in two eye state case, this is a surprising result. When we analyze the video segments in Table 4.31, which are detected incorrectly by the method we propose, we observe that these video segments mostly consist of the frames with semi-closed as the eye state. That's the reason for decreasing accuracy in drowsiness detection. These results show that unlike most studies in the literature, semi-closed state is needed to be taken into account in order to achieve a high accuracy in drowsiness detection. Emphasizing the importance of the semi-closed state as the third eye state is another contribution of our proposed method.

CHAPTER 5

CONCLUSION AND FUTURE WORK

Eye closure rate is used as the indicator of drowsiness in this thesis. We extract the video data to its frames and the frames are input to the part eye region extractor. Eye regions found by eye region extractor are grayscaled, resized to [12 18] and histogram equalized. After this process, every right and left eye image is input to neural networks separately which are trained with the subject's eye region images. The outputs of right and left eye neural networks are both digitized and merged in order to estimate the eye state of the subject. After eye state estimation process is completed for all of the frames of the video segment, each frame is tagged as open (0), semi-closed (0.5), closed (1) and "no valid estimation". We take the mean of the eye states for which valid estimation could be performed, we call this value "average eye state point". Video segments whose average eye state point exceeds the threshold value are detected as drowsy and video segments whose average eye state point does not exceed the threshold value are detected as alert.

We obtain 99.1% accuracy in drowsiness detection, which is a convincing result. There is a trade-off between neural network's input size and the memory and time required by neural network. Gray-scaling and resizing the eye region images to 12×18 gives us the chance to use less neurons in neural network. Histogram equalization increases the performance of eye state detection since it decreases negative effects arising from illumination variations. Merging the output of right and left neural networks, in other words, eliminating frames for which right and left

neural networks do not agree, increases the eye state detection accuracy of our method.

As we mentioned in section 4.3.3, combining the estimations for right and left eyes increases the accuracies for both eye state and drowsiness detection. Since combination of the estimations for right and left eyes is not a common method used in the literature, increasing the accuracy with this method is a contribution of our proposed algorithm.

As we mentioned in section 4.4.4, most of the studies assign eyes only two states: open and closed. As a contribution, this study reveals the fact that semi-closed state has an important role in detecting drowsiness and defining three states instead of two states increases the accuracy of the drowsiness detection method proposed.

We are discarding about 20% of the frames in eye state estimation process. That is, in 20% of the frames, right and left neural networks do not agree. We do not use that 20% in drowsiness prediction. Since video segments used are 30 fps, this does not prevent us from accurately detecting drowsiness. However, when fps rate of a video to be tested decreases, this might be a problem. That's why, we are planning to increase this rate as a future study.

Forming the ground truth for eye states was a challenging task. During this process, we managed difficulties in distinguishing semi-closed versus open eyes, and semi-closed versus closed eyes. For each frame, increasing the number of persons forming the ground truth will increase the accuracy of the ground truth. This method can be used to form a much reliable eye state database as a future study.

When we analyze the drowsy videos, we realized that drowsiness has stages and the situation is the same in alert videos, as well. As a future study, both drowsy and alert states can be divided into 2 categories resulting in totally 4 categories for the subject's condition.

Since eye shapes differ from person to person, using 4 subjects is not enough to train neural networks for across subject recognition; that is the reason for low accuracy in across subject recognition tests. As a future study, the number of subjects can be increased and the accuracy in accross subject recognition can be increased.

The objective of this thesis is to accurately detect drowsiness and the method we proposed achieves this objective. In the future, this thesis will be a part of a safety system being used in vehicles and help us save many lives.

REFERENCES

[1] Traffic Accident Statistics Road, 2012. Available from: <http://www.tuik.gov.tr/Kitap.do?metod=KitapDetay&KT_ID=15&KITAP_ID=70>. [2 December 2013].

[2] Association for safe international road travel. Available from: <http://www.asirt.org/KnowBeforeYouGo/RoadSafetyFacts/RoadCrashStatistics/tabid/213/Default.aspx>. [20 November 2013].

[3] Wang J.S. Knipling, R.R. Revised estimates of the US drowsy driver crash problem based on general estimates system case reviews. In Proceedings of the 39th Annual Association for the Advancement of Automative Medicine, pages 451–466, Chicago, IL, 1995.

[4] Statistics related to drowsy driver crashes. Available from: <http://www.americanindian.net/sleepstats.html>. [20 November 2013].

[5] Driver fatigue is an important cause of road crashes. Available from: <http://www.smartmotorist.com/traffic-and-safety-guideline/driverfatigue-is-an-important-cause-of-road-crashes.html>. [20 November 2013].

[6] Regulatory impact and small business analysis for hours of service options. Federal Motor Carrier Safety Administration. Retrieved on 2008-02-22.

[7] Lienhart R., Kuranov A., and V. Pisarevsky, “Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection”. In Proceedings of the 25th DAGM Symposium on Pattern Recognition. Magdeburg, Germany, 2003.

[8] Castrillón Marco, Déniz Oscar, Guerra Cayetano, and Hernández Mario, “ENCARA2: Real-time Detection of Multiple Faces at Different Resolutions in Video Streams”. In Journal of Visual Communication and Image Representation, 2007 (18) 2: pp. 130-140.

[9] Paul Viola and Michael J. Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features”. IEEE CVPR, 2001.

[10] T. Pilutti, and A.G. Ulsoy, “On-line Identification of Driver State for Lane-keeping Tasks,” in Proc. American Control Conference, Seattle,Washington, vol. 1, pp. 678-681, 1995.

[11] T. Pilutti, and A.G. Ulsoy, “Identification of Driver State for Lane-keeping Tasks”, in IEEE Trans. Systems, Man, and Cybernetics, Part A, vol. 29, pp. 486-502, 1999.

[12] Iizuka H. Yanagishima-T. Kataoka Y. Seno T. Yabuta, K., “The Development of Drowsiness Warning Devices”. In Proceedings 10th International Technical Conference on Experimental Safety Vehicles, Washington, USA., 1985.

[13] Planque S. Lavergne, C. Cara H. de Lepine, P. Tarriere, C. Artaud P., “An On-board System for Detecting Lapses of Alertness in Car Driving”. In 14th E.S.V. conference, session 2 - intelligent vehicle highway system and human factors Vol 1, Munich, Germany, 1994.

[14] C. Lavergne, P. De Lepine, P. Artaud, S. Planque, A. Domont, C. Tarriere, C. Arsonneau, X. Yu, A. Nauwink, C. Laurgeau, J.M. Alloua, R.Y. Bourdet, J.M. Noyer, S. Ribouchon, and C. Confer. “Results of the Feasibility Study of a System for Warning of Drowsiness at the Steering Wheel Based on Analysis of Driver Eyelid Movements.” In Proceedings of the Fifteenth International Technical Conference on the Enhanced Safety of Vehicles, Melbourne, Australia, 1996.

[15] Heart rate variability: standards of measurement, physiological interpretation and clinical use. task force of the european society of cardiology and the north american society of pacing and electrophysiology. Circulation, 93(5):1043–1065, March 1996.

[16] Xun Yu., “Real-time Nonintrusive Detection of Driver Drowsiness.” Technical Report CTS 09-15, Intelligent Transportation Systems Institute, 2009.

[17] S. Elsenbruch, M. J. Harnish, and W. C. Orr., “Heart Rate Variability During Waking and Sleep in Healthy Males and Females.” Sleep, 22:1067–1071, Dec 1999.

[18] Chin-Teng Lin, Ruei-Cheng Wu, Tzyy-Ping Jung, Sheng-Fu Liang, and Teng-Yi Huang, “Estimating Driving Performance Based on EEG Spectrum Analysis”. EURASIP J. Appl. Signal Process., 2005:3165–3174, 2005.

[19] T. P. Jung, S. Makeig, M. Stensmo, and T. J. Sejnowski, ”Estimating Alertness From the EEG Power Spectrum”. In IEEE Transactions on Biomedical Engineering, Vol. 44, pp. 60-69, Jan. 1997.

[20] I. Garcia, S. Bronte, L. M. Bergasa, J. Almazan, J. Yebes, “Vision-based Drowsiness Detector for Real Driving Conditions”. In Intelligent Vehicles Symposium, pp. 618-623, June 2012.

[21] Driver State Sensor developed by seeingmachines Inc. Available from: http://www.seeingmachines.com/product/DSS. [20 November 2013].

[22] Yu-Shan Wu, Quen-Zong Wu, Ting-Wei Lee, Heng-Sung Liu, “An Eye State Recognition Method for Drowsiness Detection”. In Vehicular Technology Conference, pp. 1-5, 2010.

[23] Marco Javier Flores, Jose Maria Armingol, Arturo de la Escalera, “Real-Time Warning System for Driver Drowsiness Detection Using Visual Information”, Dec. 2009.

[24] Tapan Pradhan, Ashutosh Nandan Bagaria, Aurobinda Routray, “Measurement of PERCLOS using Eigen-Eyes”, 4th International Conference on Intelligent Human Computer Interaction, pp. 1-4, Dec. 2012.

[25] Haruo Matsuo, Abdelaziz Khiat, Mobility Services Laboratory, Nissan Research Center, “Prediction of Drowsy Driving by Monitoring Driver’s Behavior”, in 21st International Conference on Pattern Recognition(ICPR 2012), pp. 3390-3393, Nov. 2012.

[26] M.Omidyeganeh, A.Javadtalab, S.Shirmohammadi, “Intelligent Driver Drowsiness Detection Through Fusion of Yawning and Eye Closure”, in IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems, pp. 1-6, 2011.

[27] Esra Vural, Mujdat Cetin, Aytul Ercil, Gwen Littlewort, Marian Bartlett, Javier Movellan, “Drowsy Driver Detection Through Facial Movement Analysis”, ICCV 2007.

[28] Gwen Littlewort, Jacob Whitehill, Tingfan Wu, Ian Fasel, Mark Frank, Javier Movellan, Marian Barlett, “The Computer Expression Recognition Toolbox (CERT)”, Machine Perception Laboratory, IEEE International Conference on Automatic Face & Gesture Recognition and Workshops, pp. 298-305, 2011.

[29] P. Ekman and W. Friesen, “Facial Action Coding System: A Technique for the Measurement of Facial Movement”, Consulting Psychologists Press, Palo Alto, CA, 1978.

[30] Esra Vural, Marian Bartlett, Gwen Littlewort, Mujdat Cetin, Aytul Ercil, Javier Movellan, “Discrimination of Moderate and Acute Drowsiness Based on Spontaneous Facial Expressions”, in International Conference on Pattern Recognition (ICPR), pp. 3874-3877, 2010.

[31] Vogl, T.P., J.K. Mangis, A.K. Rigler, W.T. Zink, and D.L. Alkon, "Accelerating the Convergence of the Backpropagation Method", Biological Cybernetics, Vol. 59, 1988, pp. 257–263.

[32] K. Levenberg, “A Method for the Solution of Certain Problems in Least Squares”, Quart. Appl. Math., 1944, Vol. 2, pp. 164–168.

[33] D. Marquardt, “An Algorithm for Least-squares Estimation of Nonlinear Parameters”, SIAM J. Appl. Math., 1963, Vol. 11, pp. 431–441.

[34] Y. Freund and R. E. Schapire, “Experiments with a New Boosting Algorithm”. In Machine Learning: Proceedings of the Thirteenth International Conference, Morgan Kauman, San Francisco, pp. 148-156, 1996.

[35] Ojala Timo, Pietikäinen Matti, and Mäenpää Topi, “Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns”. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002. Volume 24, Issue 7, pp. 971-987.

[36] Computer Vision System Toolbox, MATLAB.

[37] K. Sobottka, I. Pitas, “A Novel Method for Automatic Face Segmentation, Face Feature Extraction and Tracking”, in Signal Processing: Image Communication 12 (3), pp. 263-281, 1998.

[38] S. Feyrer, A. Zell, Detection, “Tracking and Pursuit of Humans with Autonomous Mobile Robot”, in Proceedings of the International Conference on Intelligent Robots and Systems, Kyongju, Korea, 1999, pp. 864–869.

[39] E. Hjelmas, I. Farup, “Experimental Comparison of Face/non-face Classifiers”, in Proceedings of the Third International Conference on Audio- and Video-Based Person Authentication. Lecture Notes in Computer Science 2091, pp. 65–70, 2001.

[40] Martin Riedmiller und Heinrich Braun, “Rprop - A Fast Adaptive Learning Algorithm”. In Proceedings of the International Symposium on Computer and Information Science VII, 1992.

[41] Straeter, T. A. "On the Extension of the Davidon-Broyden Class of Rank One, Quasi-Newton Minimization Methods to an Infinite Dimensional Hilbert Space with Applications to Optimal Control Problems". NASA Technical Reports Server. NASA. Retrieved 10 October 2011.

[42] Karl F. Van Orden, Tzyy-Ping Jung, and Scott Makeig, “Combined Eye Activity Measures Accurately Estimate Changes in Sustained Visual Task Performance”. Biological Psychology 52(3), pp. 221-240, 2000.

[43] Lei Yunqi, Yuan Meiling, Song Xiaobing, Liu Xiuixia, Ouyang Jiangfan, “Recognition of Eye States in Real Time Video”, in: International Conference on Computer Engineering and Technology, Singapore, pp. 554-559, 2009.

[44] Esra Vural, Mujdat Cetin, Aytul Ercil, Gwen Littlewort, Marian Bartlett, Javier Movellan, “Machine Learning Systems for Detecting Driver Drowsiness”, in: In-Vehicle Corpus and Signal Processing for Driver Behaviour, pp. 97-110, 2009.

[45] A. Mohan, C. Papageorgiou, T. Poggio, “Example-based Object Detection in Images by Components”. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 4, pp. 349-361, April 2001.

[46] C. Papageorgiou, M. Oren, and T. Poggio, “A General Framework for Object Detection”. In International Conference on Computer Vision, pp. 555-562, 1998.

[47] D.F. Dinges and R. Grace, “PERCLOS: A Valid Psychophysiological Measure of Alertness as Assessed by Psychomotor Vigilance”, U.S. Department of Transportation, Federal Highway Administration, Report No. FHWA-MCRT-98-0006, 1998.

[48] UYKUCU Database, Drive Safe Project, Sabanci University Computer Vision and Pattern Analysis Laboratory (VPALAB), 2009.