ANKARA YILDIRIM BEYAZIT UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

IRIS DETECTION USING HYBRID MODELS

M.Sc. Thesis by

Nour N.M. NASSAR

Department of Electrical and Electronics Engineering

June, 2017

ANKARA

IRIS DETECTION USING HYBRID MODELS

A Thesis Submitted to
The Graduate School of Natural and Applied Sciences of
Ankara Yıldırım Beyazıt University
In Partial Fulfillment of the Requirements for the Degree of Master of Science in
Electronics Engineering, Department of Electrical and Electronics Engineering

by

Nour N.M. NASSAR

June, 2017
ANKARA

M.Sc. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **“IRIS DETECTION USING HYBRID MODELS”** completed by **NOUR N.M. NASSAR** under the supervision of **ASSIST. PROF. DR. ÖMER KARAL** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Ömer KARAL

Supervisor

Assist. Prof. Dr. Enver ÇAVUŞ

Jury Member

Assist. Prof. Dr. Mahmut SİNECEN

Jury Member

Prof. Dr. Fatih V. ÇELEBİ

Director

Graduate School of Natural and Applied Sciences.

ETHICAL DECLARATION

I hereby declare that, in this thesis which has been prepared in accordance with the Thesis Writing Manual of Graduate School of Natural and Applied Sciences,

- All data, information and documents are obtained in the framework of academic and ethical rules,

- All information, documents and assessments are presented in accordance with scientific ethics and morals,

- All the materials that have been utilized are fully cited and referenced,

- No change has been made on the utilized materials,

- All the works presented are original,

And in any contrary case of above statements, I accept to renounce all my legal rights.

2017, 21 June **Nour N. M. NASSAR**

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Assist. Prof. Dr. Ömer KARAL, for the unlimited guidance and the continuous support throughout my master study and research, who always welcomed my questions and advised me. Without his supervision; this work would not have been possible.

Besides my advisor I would like to express my thanks to my thesis committee members Assist. Prof. Dr. Enver ÇAVUŞ and Assist. Prof. Dr. Mahmut SİNECEN for their support, guidance and helpful feedback.

I have no words to express my thankfulness and gratitude to my husband AHMED NASSAR for the infinite patience, support and encouragement throughout my work and study. Without his assist I would not have been complete this work.
Special thanks and dedication to MOHAMMED, NESMA, ASMAA' and MAJD my beloved children.

My sincere thanks also goes to my family in Palestine, my father, mother brothers and sisters for their unlimited support, and encouragement.

Moreover, I would like to recognize YTB for offering the success scholarship in my second year of study, and the Turkish people for their friendly accommodation.

2017, 21 June **NOUR N. M. NASSAR**

IRIS DETECTION USING HYBRID MODELS

ABSTRACT

Face and iris detection are great of interests to researchers in the field of video processing and computer vision. In recent years, great advances in robotics and artificial intelligence have further increased the importance of face recognition and iris detection.

In this study, iris of the human eye is detected through hybrid models. These models first detect the face, then the eyes, and finally the iris.

Two classifiers, the Artificial Neural Network (ANN) and the Decision Tree (DT), are trained to distinguish face images from other. They are also used to extract face images using the sliding window technique. After extracting the face from the given image, the same steps are applied to detect the eyes from the face. It has been observed that the ANN performs better than the DT with respect to detection rate of faces and eyes.

The eyes that are finally detected are used to find circles that indicate the iris. Two methods are discussed to detect the iris: Hough Transform (HT) and the mean of gradients (MoG). In comparison with HT, MoG yields higher detection rate.

Keywords: Object detection, iris detection, face detection, Artificial Neural Network classifier, Decision tree classifier, Hough transform method, means of gradients method.

HİBRİT MODELLERİNİ KULLANARAK IRIS ALGILAMA

ÖZ

Yüz ve iris tanıma, bilgisayarla video ve görüntü işleme alanında çalışan araştırmacılar tarafından büyük ilgi çekmektedir. Robotik ve yapay zekâ alanlarında son yıllarda ortaya çıkan büyük ilerlemeler, yüz ve iris tanımanın önemini daha da artırmıştır.

Bu çalışmada, insan gözünün irisi hibrit modellerle tespit edilmiştir. Önce yüz, daha sonra gözler ve en sonunda iris algılanır.

Yapay Sinir Ağı (YSA) ve Karar Ağacı (KA) olmak üzere iki sınıflandırıcı, sadece insan yüzü'nden oluşan görüntüleri diğerlerinden ayırmak için eğitilmiştir. Ayrıca bu iki sınıflandırıcı, kayan pencere tekniğinden faydalanarak verilen görüntüden yüz içeren bölgelerin ayrılmasında da kullanılır. Elde edilen yüz verilerinden gözlerin algılanması için de aynı adımlar uygulanır. Yüzleri ve gözleri tanıma oranına göre YSA'nın, KA'dan daha iyi performans gösterdiği gözlenmiştir.

Son olarak, irisi belirten daireleri bulmak için algılanan gözler kullanılır. İrisi saptamak için Hough Dönüşümü (HD) ve eğimlerin ortalaması (EO) yöntemleri ele alınmıştır. EO, HD ile karşılaştırıldığında, daha yüksek algılama hızı sağladığı görülmüştür.

Anahtar kelimeler: Nesne tanıma, iris tanıma, yüz tanıma, Yapay Sinir ağı sınıflandırıcısı, karar ağacı sınıflandırıcısı, Hough dönüşümü yöntemi, gradyan elemanları yöntemi.

CONTENTS

M.Sc. THESIS EXAMINATION RESULT FORM ii
ETHICAL DECLARATION iii
ACKNOWLEDGEMENTS iv
ABSTRACT v
ÖZ vi
NOMENCLATURE x
LIST OF TABLES xi
LIST OF FIGURES xii

CHAPTER 1-INTRODUCTION 1
1.1 Iris Detection Importance and Definition 1
1.2 Study Methodology 4
1.3 Study Objectives and Motivations 4
1.4 Thesis Outlines 4

CHAPTER 2-BACKGROUND AND LITERATURE REVIEW 6
2.1 Object Detection 6
2.1.1 Geometry Based Methods 6
2.1.2 Appearance Based Methods 7
2.1.3 Feature Based Methods 8
2.2 Object Representation 8
2.3 Object Tracking Models 9
2.3.1 Stochastic Models 10
2.3.2 Deterministic Models 10
2.3.3 Hybrid Models 12

CHAPTER 3-OVERVIEW OF DETECTION METHODOLOGY 14
3.1 Detection Architecture 14
3.1.1 Learning Phase 14
3.1.2 Detecting Phase 14
3.1.2.1 Face Detection 16
3.1.2.2 Eye Detection 17

3.1.2.3 Iris Detection 17
3.2 Histogram of Oriented Gradients (HOG) 17
3.2.1 Algorithm Implementation 17
3.2.1.1 Gradient Computation 18
3.2.1.2 Orientation Binning 19
3.2.1.3 Block Normalization 19
3.2.2 HOG Descriptor 20
3.2.3 Features Vector Computation: 20
3.3 Classification 23
3.3.1 Artificial Neural Network Classifier 25
3.3.2 Decision Tree Classifier (DT) 27
3.4 Iris Detection 28
3.4.1 Hough Transform 28
3.4.2 Means of Gradients 29

CHAPTER 4-EXPERIMENTAL RESULTS AND EVALUATION 32
4.1 Datasets 32
4.1.1 The Extended Yale Face Database B 33
4.1.2 CBCL Face Database #1 33
4.1.3 MMU Iris Database 34
4.1.4 BioID Face Database 35
4.1.5 Talking Face video 36
4.2 Evaluation Measures 37
4.2.1 Metrics Evaluation Measures 37
4.2.2 Estimated Eye Centres Evaluation 38
4.3 Experimental Results of Face and Eyes Detection 39
4.3.1 Training Phase 39
4.3.1.1 Face Detector Training 39
4.3.1.2. Eye Detector Training 40
4.3.2 Testing Phase 40
4.3.2.1 Face Detector Testing 42
4.3.2.2 Eye Detector Testing 43
4.4 Experimental Results of Iris Detection 46
4.4.1 Evaluation by Calculating Detection Rate 46

4.4.2 Detection Evaluation with Respect to The Ground Truth Table 48

CHAPTER 5-CONCLUSION AND FUTURE WORK 55
REFERENCES 57
CURRICULUM VITAE 61

NOMENCLATURE

Acronyms

ANN	Artificial Neural Network
BCI	Brain Computer Interface
CHT	Circle Hough Transform
DT	Decision Tree
HOG	Histogram of Oriented Gradients
HT	Hough Transform
MoG	Means of Gradients
SAM	Spatial optical Appearance Model
SVM	Support Vector Machines

LIST OF TABLES

Table 3.1 Magnitude of 64 pixels in the cell. 21

Table 3.2 Direction of 64 pixels in the cell. 21

Table 3.3 Nine bins of 8x8 pixels in the cell. 22

Table 4.1 Datasets. 32

Table 4.2 Evaluation measures. 37

Table 4.3 Training results of ANN and DT classifiers for face detection. 39

Table 4.4 Training results of ANN and DT classifiers for eye detection. 40

Table 4.5 Face Detection rate for the first 150 images of the Bio ID Face Database. 42

Table 4.6 Face Detection rate for 150 images of the Talking Face database. 42

Table 4.7 Eye detection rate for 140 images from BioID Face Database. 43

Table 4.8 Eye detection rate for 150 images from the Talking Face video database. 43

Table 4.9 Eye detection rate for 150 images from the Talking Face video database after excluding images with false detected eyes (Evaluation by eyes). 47

Table 4.10 Examples of correct and estimated eye centers. 48

Table 4.11 Examples of calculations of normalized error. 49

Table 4.12 Eye detection rate for 150 images from the Talking Face video database by means of gradients method. 49

Table 4.13 Eye detection rate for 150 images from the Talking Face video database by Hough transform method ($e < 0.25$). 51

LIST OF FIGURES

Figure 1.1 The human eye [1]. 2

Figure 1.2 Remote iris tracking system [4]. 3

Figure 1.3 Head mounted system [5]. 3

Figure 2.1 (a) cropped image from PETS 2006 dataset [17], object represented by (b) a point, (c) an ellipse, (d) a rectangle, (e) contour and (f) silhouette. 9

Figure 3.1 Learning phase for face and eye detection... 15

Figure 3.2 Detecting phase: Iris detection architecture 16

Figure 3.3 Frame number 3000 from the talking face dataset. 20

Figure 3.4 Binning representation of 64 pixels in the cell 22

Figure 3.5 Binning of a block vector consists of 2x2 cells. 23

Figure 3.6 Classifier training steps. 24

Figure 3.7 Testing process. 24

Figure 3.8 Neuron architecture. 25

Figure 3.9 MLP Artificial Neural network architecture 26

Figure 3.10 Backpropagation artificial Neural network architecture. 26

Figure 3.11 Decision Tree for face and eye detection. 27

Figure 3.12 Transformation of many points from I to H space [38]. 29

Figure 3.13 Blue arrows Points in the direction of greatest increase of the function. 30

Figure 3.14 Representing gradient and displacement vectors in a circle of center c. 30

Figure 4.1 Examples of the cropped version of The Extended Yale Face Database B [44] 33

Figure 4.2 Examples of non-faces images from CBCL Face Database #1[34] 34

Figure 4.3 Examples of iris images from the MMU iris database[46]...................... 35

Figure 4.4 Examples of faces from the BioID Face Database. 36

Figure 4.5 Examples from the talking face dataset. 36

Figure 4.6 Test image (to the left) and resized image (to the right). 40

Figure 4.7 (a)Image after applying Gaussian filter, (b) After converting to gray scale. 41

Figure 4.8 The yellow box represents a sliding window at the beginning of the image that does not contain a face. The red box contains a face, while the blue one does not contain a face and it is at the end of the image. 41

Figure 4.9 (a) Bounding Boxes after classification. (b) Chosen bounding box. (c) Applying Bounding Boxes positions on the colored image. 42

Figure 4.10 Detection is implemented using DT classifier, in (a) face and both eyes were detected. (b) none of the eyes have been detected. (c) right eye has been detected correctly while left eye is not detected. 44

Figure 4.11 Detection is implemented using ANN classifier, in (a) face and both eyes were detected. (b) none of the eyes have been detected. (c) left eye has been detected correctly while right eye is not detected. 44

Figure 4.12 Examples of correct face and eyes detection using DT classifier.......... 44

Figure 4.13 (a) Correct left eye detection and false right eye detection. (b) Correct right eye detection and false left eye detection. 45

Figure 4.14 Examples of correct face detection with false eyes detection. 45

Figure 4.15 Examples of false face and eyes detection... 45

Figure 4.16 Examples of correct face and eyes detection using ANN classifier. 45

Figure 4.17 (a) Correct left eye detection and false right eye detection. (b) Correct right eye detection and false left eye detection. 46

Figure 4.18 Examples of correct face detection with false eyes detection. 46

Figure 4.19 Examples of false face and eyes detection... 46

Figure 4.20 In (a) both irises where correctly detected, while in (b) only left iris was correctly detected. 47

Figure 4.21 In (a) only right iris was correctly detected. In (b) both irises where correctly detected. 48

Figure 4.22 Normalized error distribution for left eye after using means of gradients method (e < 0.1). 50

Figure 4.23 Normalized error distribution for right eye after using means of gradients method (e < 0.1). 50

Figure 4.24 Normalized error distribution for both eyes after using means of gradients method (e < 0.1). 51

Figure 4.25 Normalized error distribution for left eye after using Hough transform method (e < 0.25). 52

Figure 4.26 Normalized error distribution for right eye after using Hough transform method (e < 0.25). 52

Figure 4.27 Normalized error distribution for right eye after using Hough transform method (e < 0.25). 52

CHAPTER 1

INTRODUCTION

Nowadays, many researches focus on object detection and tracking as a significant field. The importance of this field comes from the unlimited applications related to human life improvements such as robotics and human computer interaction.

Human eyes can detect and track objects of interest effortlessly, even with changing in scale, illumination and occlusion. On the other hand, this task is more difficult for machines, so many approaches and methods have been developed and examined to achieve this goal.

1.1 Iris Detection Importance and Definition

Among the fields of object detection, detection the iris of the human eye is considered as an important issue, this belongs to the wide applications follows the detection and tracking of the iris.

Iris tracking is an expression which refers to determining the position of the iris at a given time, or the sequence of its movements, and it is differs from eye tracking which refers to determine the gaze direction of a person.

The iris is an item in the sight system, which is a great gift for human. The first component of this system, the human eye, is shown in Figure 1.1.

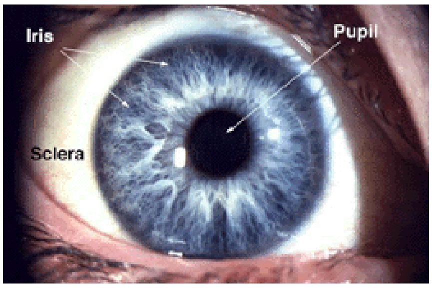


Figure 1.1 The human eye [1].

The black center of the eye is the pupil, at which light enters the eye. The iris is a colored muscle that controls the size of the pupil so that less or more light enters the eye.

Movement of iris represents a good research topic due to the variety of applications correspond to it. Iris tracking enables the analysis of eye movements and gaze estimation; in the medical field it gives a representation of human dealing with visual information which is used for diagnosis purposes.

Besides that, this kind of tracking gains more attention in the field of human computer interaction such as gaming and controlling.

Persons with disabilities or paralysis can achieve behavior that increases quality of life, such as environmental control, self-motivation, and computer access, by tracking the iris. Today, this field continues to develop Brain Computer Interface (BCI) systems. In this system, the messages and commands that the individual sends to the outside world do not pass through the peripheral nerves and muscles that are the normal output pathways of the brain [2]. Therefore, BCI technology can take advantage of iris follow-up to get these voluntary signals.

Recently many software and hardware applications have been developed for iris detection and tracking. Different companies and research groups have developed a variety of equipment for such tracking, Tobii and MyGaze are examples of these applications [3].

Meanwhile, these applications are efficient and suitable for many people; they need special equipment and focuses on commercial usage. Some of these applications are head mounted and others are remote. Examples can be shown in Figures 1.2, and 1.3.

Figure 1.2 Remote iris tracking system [4].

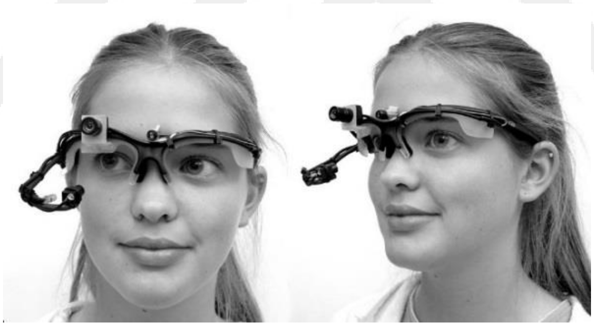


Figure 1.3 Head mounted system [5].

These techniques are used to take a snapshot of the face and eyes, then the iris is detected using different algorithms, moreover many of these techniques use a source of Infrared rays besides a web camera to make iris extraction easier.

In this work, the images will be taken from a pre-recorded video, and then iris detection will be applied.

1.2 Study Methodology

Referring to the small area of the iris, this makes it's tracking a challenging problem. This problem can be overcome by reducing the area of searching and processing, which can be enclosed in the face and especially in eyes region.

In this thesis, a sliding window is scanning the image to extract patches, then these patches are classified as faces or not faces. Afterwards, another sliding window searches the extracted face image to separate eyes from non-eyes.

When eyes are detected, iris detection algorithms are applied in the same region.

1.3 Study Objectives and Motivations

This study aims to achieve three objectives: Firstly, to choose the more powerful classifier for face detection. Secondly, select an appropriate eye detector. Finally, to compare between different iris detection techniques.

The main factor which motivated us to choose iris detection as a research field, is that the applications of iris detection require an efficient and fast detection technique.

Besides that, most of face and eye detection methods use the support vector machine (SVM) method as a classifier. However, SVM method suffers from large time consumptions and need large memory due to the large number of optimization variables especially, in case of a large data set. Therefore, we aimed to find another computationally efficient classifier that can achieve the detection in a short time.

1.4 Thesis Outlines

This thesis is organized as follows: Following the introduction, Chapter 2 discusses object tracking from general viewpoint, it gives the idea of object detection and main approaches used for this purpose, then displays object representation in the image, following that a literature review is submitted.

Chapter 3 introduces an overview of the detection methodology, starting with the detection architecture which contains learning and detecting phases of face and eye. Then the features extraction method is introduced. After that classification methods are explained. Finally, iris detection methods are displayed.

Chapter 4 begins with displaying the datasets that were used in this work, then it explains the evaluation methods for detection. Following that the experimental results are introduced and discussed.

The thesis is concluded in Chapter 5 which summarizes the whole work, and also offers some ideas about future work.

CHAPTER 2

BACKGROUND AND LITERATURE RIVIEW

Object tracking is one of the most important fields in computer vision, and it has many applications in our daily life such as traffic flow monitoring, pedestrian counting, surveillance, and human-computer interaction [6].

2.1 Object Detection

Object detection determines the object of interest in the image. Furthermore, it is also used in videos and image sequences to track objects as a first step.

Although object detection methods developed many decades ago, they also need some improvements to obtain efficient and accurate detection. In the following subsections, the most widely used methods for object detection will be introduced.

2.1.1 Geometry Based Methods

Geometry-based methods are called shape or model methods as well. In these methods, a geometric description of a 3D object is projected as 2D image, and then this projection is compared to object of interest. In the recognition process, the information of edges or boundaries of object are used, and some geometric primitives such as lines, circles, etc. are extracted [7].

Edges and geometric shapes are not very sensitive to illumination changes, so using these methods provides some advantages.

According to the geometry of the eye, it can be modeled by the contours of the iris and the pupil, besides the eyelids shape [8].

The iris is often approximated by a circle in image processing; meanwhile, it is rarely appearing as a true circle in images due to different poses [9].

Kothari et al. considered the iris as an ellipsoidal shape, that is darker than its background, which is the sclera, so when computing the gradients of iris, they will point outwards its center [10]. This idea is very similar to the means of gradients method that have been used in this work.

Although the Geometry-based methods have been extensively used in early stages of object recognition, nowadays they are used rarely since their limited efficiency. For complex objects, they are not very efficient and they need extra time to create the models, due to the manual process [11].

2.1.2 Appearance Based Methods

Appearance-based methods contain two phases [11] In the first phase, a model is built from a set of reference images. This set can contain various appearances of the object under different conditions such as orientations, and illumination. In the second phase, the same size sub- images are extracted as training images from the input image.

This can be implemented either by segmentation method using some features like: texture, color and motion, or by applying image windows over whole image. Then, the extracted part of the input image is compared to the reference images. Appearance – based approaches are used mostly in recognition of faces.

B. Moghaddam et al. used the eigenspace decomposition to estimate density, and they used the Gaussian distribution besides a mixture-of-Gaussians density model. They detected faces and facial features such as eyes and mouths [12]. Pentland et al. recognized faces using a large face database that contains at least two images for every person. Many people in this database have images that varies in facial expressions, headwear and facial hair [13].

2.1.3 Feature Based Methods

Feature–based methods are used to find interest points employing a set of feature detectors, which build a representative vector for images [14].

A searching can be applied to match between object features and image features. The corners, linear edges and surface patches are considered as samples of features. As an example, authors used the contours of a mug to extract the features, also they reduced the contour to a set of simple geometric features including lines and arcs [15].

2.2 Object Representation

An object in the image is considered as a single unit, which is formed from many pixels, which construct its shape. The shape of the object can be detected using one of the methods mentioned in section 2.1.

After detecting the object, it can be illustrated in the image in many ways in order to be recognized in successive frames [16].

The object of interest can be represented in the image as one of the following shapes:

1.Point: The object is represented by a point, which is it's centroid. (Figure 2.1 b)

2. Geometric shape: The Object is represented by a geometric shape such as rectangle or ellipse. (Figure 2.1 c, d)

3. Contour: The contour is defined as the outer lines that determines the shape of the object, and these lines can give a good representation of it. (Figure 2.1 e)

4. Object silhouette: The Object silhouette is the region inside the contour and it is used for object representation. (Figure 2.1 f)

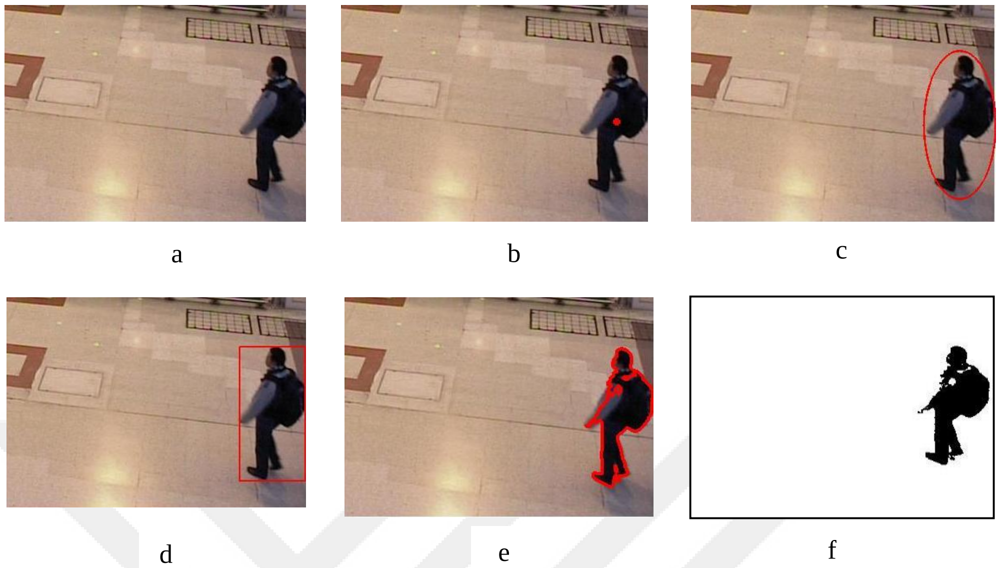


Figure 2.1 (a) cropped image from PETS 2006 dataset [17], object represented by (b) a point, (c) an ellipse, (d) a rectangle, (e) contour and (f) silhouette.

2.3 Object Tracking Models

Object tracking can be defined as the process of tracking the object, or it's features as they move in a sequence. This process requires to estimate a path of movement of the object of interest.

According to A. Yilmaz et al. in order to track an object the trajectory of that object in the image plane need to be estimated as it moves around a scene [16].

Object tracking approaches can be categorized in many different models, such as: template based model, shape matching model and statistical model [16].

In this study the statistical model which depends on probability theory, has been considered, because it can overcome some tracking problems such as noise and random perturbations [16].

The statistical model can be classified into three categories: stochastic, deterministic, and hybrid stochastic-deterministic models [6].

2.3.1 Stochastic Models

In these models, the algorithm learns the joint probability distribution and focuses on which assumption can generate the desired signal with aiming to encode the target appearance.

When limited data is provided to the algorithm the generative approach can generalize the observation in a high rate and can generate values to any variable in the model [18].

Many examples can be listed under the generative models such as Gaussian mixture model, hidden Markov model, and Naive Bayes.

In literature McKenna et al. used adaptive Gaussian mixture models to track color objects in real time situations, they used adaptive color model to track face even when there is change in illumination [19].

Yu and Wu have implemented a Differential Tracking based on Spatial-Appearance Model (SAM) to track non-rigid objects, with ability to estimate different types of motions as translation, rotation, scaling, and affine [20].

Chen et al. introduced a hand gesture recognition system in order to recognize continuous gesture using a stationary background.

The system consists of three steps: firstly, a real time hand tracking and extraction algorithm is applied, then Fourier descriptor are used to extract features. Finally, hidden Markov model (HMM) is used to recognize the gesture [21].

2.3.2 Deterministic Models

Sometimes they are called conditional models and used to find the dependence of a variable *y* on *x* and this is done by using the conditional probability distribution.

These models are used in classification tasks because they can provide a model for the target variable depending on the observed variables, and they can work will in supervised learning with enough data but cannot do the same in unsupervised learning [18].

As examples of the deterministic models: Artificial Neural Networks (ANN), adaptive boosting, Decision Tree (DT) and Support Vector Machines (SVM) can be considered. These methods separate one class from the other by computing a hyper-surface, and require manually labeled data for each object class [16].

Many researchers have used the classifiers for object tracking, where the tracking problem is converted to a classification problem at which the foreground is separated from the background using the classifier.

Bouzenada et al. have used the ANN to track objects online, firstly they trained the ANN offline over a set of input and output examples, then they used the relation between intensity variations and position variations for the online tracking [22]. Rowley et al. [23] have developed an upright frontal face detection system using an ANN that examines small windows of an image and decides whether each window contains a face or not.

They used positive face examples for training while the non-face training examples, which are the negative examples have been generated using a bootstrap algorithm.

Adaptive boosting was used by Viola et al. [24] to detect pedestrians at very small scales (as small as 20 × 15 pixels). They used AdaBoost to train the detector using motion and appearance information that was the first to combine both sources of information in a one detector. In addition, they used the boosting with decision tree that was learned to select the appropriate detector to run on faces in order to detect in-plane rotated faces and profile faces [25].

Marée, R. et al. also used the decision tree classifier with random subwindows for classifying biomedical images. A database that contains 10000 gray level x-ray images was classified to 57 categories. Before classification, subwindows were randomly extracted from the trained images, then tree boosting and axtra tree algorithm were applied to build classafication model [26].

Huang et al. used decision trees for human face detection. The applied algorithm consists of three steps: Firstly, they find a possible position of the bounding box of the face, then improving the output image by cropping, and finally, if a given image contains a face they normalize it. The decision trees are trained to recognize faces from non-faces in images by using some features such as entropy, mean and standard deviation. After training, the decision trees are used in the cropping stage to detect faces [27].

As a deterministic model Support Vector Machines (SVM) used to track objects, Avidan [28] has integrated the support vector machine into an optical flow tracker and was called support vector tracker, which was used to track vehicles of different colors and size, the classifier was trained over about 10,000 images of vehicles and non-vehicles.

Fu et al. [29] used one class SVM to avoid limited negative samples used to describe the background, also they used an online updating scheme to provide accurate tracking, this method gives good results even when occlusion by another face or 360-degree rotation occurs.

2.3.3 Hybrid Models

The generative and the discriminative models have their own advantages and disadvantages, for example when the number of data points is increased, the discriminative models works better than the generative. On the other hand, a generative model approaches its asymptotic error faster [30].

Many studies used a combination of generative-discriminative models in order to achieve the best detection and tracking. In [30] authors used the two models to detect non-rigid objects (animals) and extract features, they observed that with small number of data the generative model performs better than the discriminative model, but for large data sets the two models are similar.

CHAPTER 3

OVERVIEW OF DETECTION METHODOLOGY

3.1 Detection Architecture

Iris tracking is a challenging problem, due to changing in light conditions, scale differences, and occlusion [31]. Usually more than one model can be used to detect face and eyes to implement an active iris tracker. In recent years, Histogram of Oriented Gradient (HOG) method has attracted considerable interest [32] due to its significant properties such as robustness, effectiveness and adaptation for various classification issues. In the HOG method, face and eyes are detected using Artificial Neural Network (ANN) and Decision Tree (DT) classifiers. Then the iris is detected using Hough transform and means of gradients methods. Two phases are observed in the tracking process: learning phase and detection phase.

3.1.1 Learning Phase

In the learning phase, a training dataset is constructed using images with and without faces; then HOG features of the dataset are extracted to perform the training matrix. A classifier is then trained using this dataset as shown in Figure 3.1-a. The same process is used to train another classifier to separate images with eyes and non-eyes in the face as shown in Figure 3.1-b.

3.1.2 Detecting Phase

In the detection phase, the trained classifier is used to detect faces firstly, and then to detect eyes. After that, iris is detected. The process is repeated for all images in the video or sequences of images to implement iris tracking as shown in Figure 3.2.

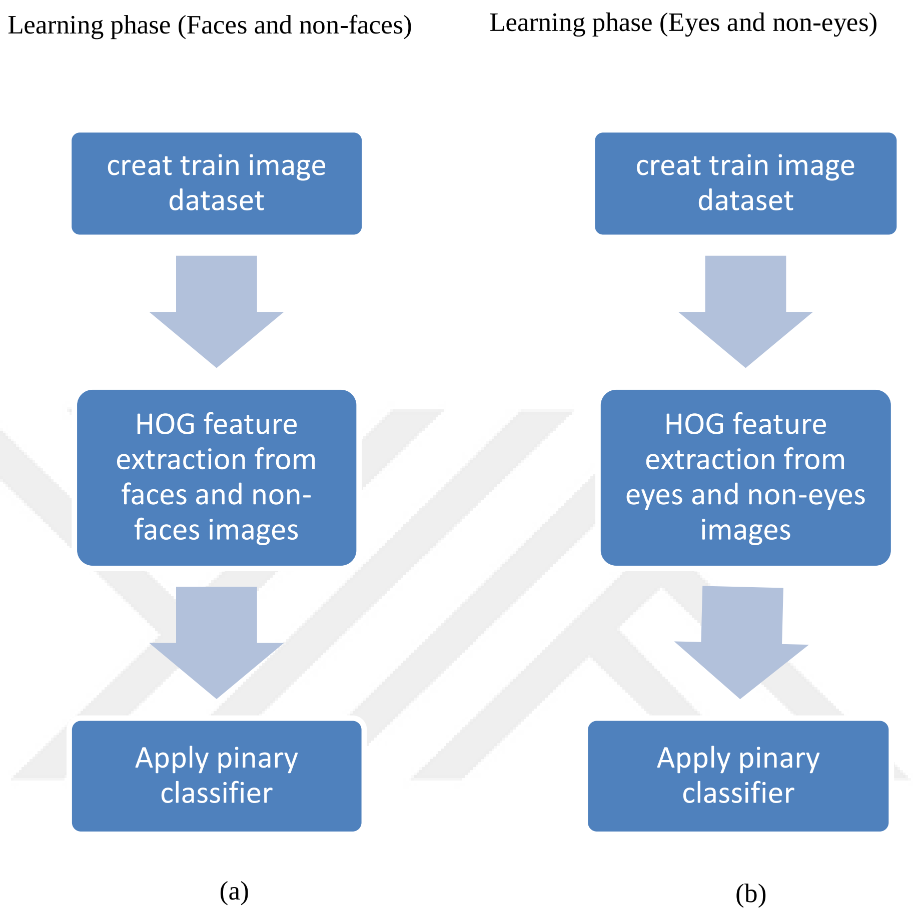


Figure 3.1 Learning phase for face and eye detection

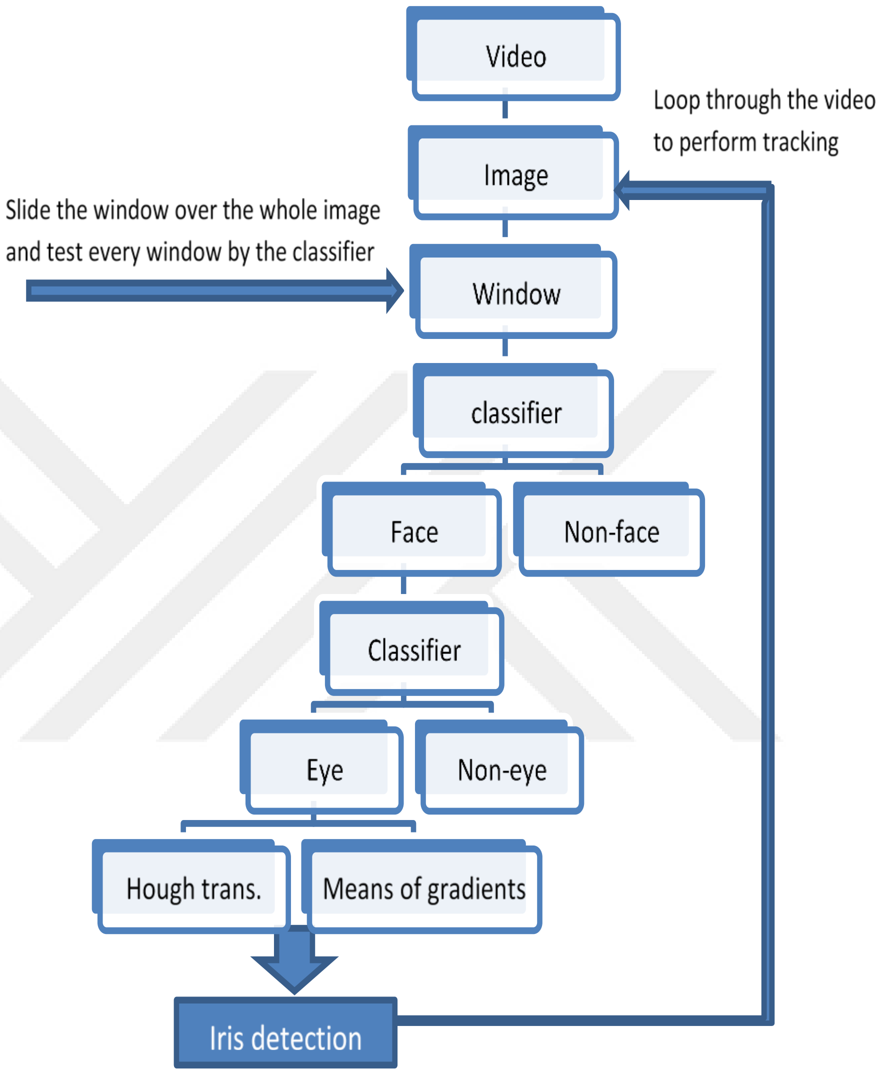


Figure 3.2 Detecting phase: Iris detection architecture

Each step of the iris detection process is detailed as follows:

3.1.2.1 Face Detection

Face detection in images and videos has become a hot research topic in the last few decades. It is an entrance to many applications such as eye detection, facial expressions recognition, face tracking and computer-human interaction.

In chapter 2, different methods of face detection has been introduced. In the face detection, the HOG features of a set of cropped faces and non-faces images have been computed. After that, a classifier is trained using the extracted features. In order to recognize the face in a given image, a sliding window is applied to the image and the HOG feature vectors are computed for every window. Then the trained classifier is used to determine whether the window contains a face or not.

3.1.2.2 Eye Detection

Eyes are considered as one of the most important features in the face. The process of eye detection usually follows face detection. After detecting the face, a slide window is applied inside the detected region and a pre-trained classifier is applied to determine the window that contains eye. The HOG features are also used to compute the features vectors.

3.1.2.3 Iris Detection

After face and eye detection, the detected regions of eyes are used to detect iris of each eye separately. Two methods are applied to perform the detection: Hough transform and means of gradients.

3.2 Histogram of Oriented Gradients (HOG)

The histogram of oriented gradients is a feature descriptor used for object detection. This descriptor was firstly described by Robert K. McConnell in 1986, but without using HOG term. Furthermore the HOG descriptor became well known in 2005 when Navneet Dalal and Bill Triggs used it to detect pedestrians in static images, and then they developed their work to include human detection in videos [32].

3.2.1 Algorithm Implementation

The HOG feature extraction consists of three main steps: Gradient computation, Orientation binning and Block normalization.

3.2.1.1 Gradient Computation

Image gradient is a directional change in the color or the intensity in an image, and it may be used to extract information from images. The gradient can be computed by using the following formula:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{3.1}$$

where $g_x = \frac{\partial f}{\partial x}$ is the gradient in the *x* direction, and $g_y = \frac{\partial f}{\partial y}$ is the gradient in the *y* direction. The gradient in the x direction $\frac{\partial f}{\partial x}$ is implemented by convolving a mask with the given image as in 3.2. It was found that a good result can be achieved by using a 1-dimentional centered mask such as $[-1\ 0\ 1]$ in the x direction[19].

$$g_x = \frac{\partial f}{\partial x} = [-1 \quad 0 \quad 1] * I \tag{3.2}$$

Where the symbol * is the convolution operator, and *I* is the given Image.

The gradient in the y direction $\frac{\partial f}{\partial y}$ can be computed using 3.3:

$$g_y = \frac{\partial f}{\partial y} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} * I \tag{3.3}$$

The magnitude of the gradient (g) can be computed as in 3.4:

$$g = \sqrt{g_x^2 + g_y^2} \tag{3.4}$$

The direction of the gradient (Θ) is given by 3.5:

$$\theta = arctan\left(\frac{g_y}{g_x}\right) \tag{3.5}$$

Where g_x and g_y can be calculated from equations 3.2 and 3.3 respectively.

3.2.1.2 Orientation Binning

In HOG feature descriptor method, the image is divided into smaller blocks, and then each block is divided to small cells that contains pixels. An example will be introduced in section 3.3.2.

In order to find the cell histogram, every pixel in the cell implement a weighted vote for an orientation-based histogram channel using the gradient value of each pixel.

The cells can be rectangular or circular in shape, in this study the rectangular cells [32] will be considered due to their simplicity. The histogram channels range from 0 to 180 degrees when it is unsigned, or range from 0 to 360 degrees for signed gradients. In this work, the unsigned gradients have been considered because they show better performance than the signed ones [32].

3.2.1.3 Block Normalization

Gradients differ in strength due to local differences in illumination and contrast between object and background. In order to remove these effects a vector normalization is applied to the vectors of the cells within the block. If *v* is the resulting vector after voting in a given block, then the normalization can be obtained by dividing vector *v* by its length which is given in the following equation:

$$\|v\| = \sqrt{v_1^2 + v_2^2 + v_3^2 + \ldots + v_m^2} \tag{3.6}$$

where m is the size of the vector.

3.2.2 HOG Descriptor

For example, to extract the feature vector by HOG descriptor, firstly the image is divided into overlapping blocks. Every block contains the same number of cells, and every cell contains the same number of pixels. Figure 3.3 illustrates blocks and cells in a given image.

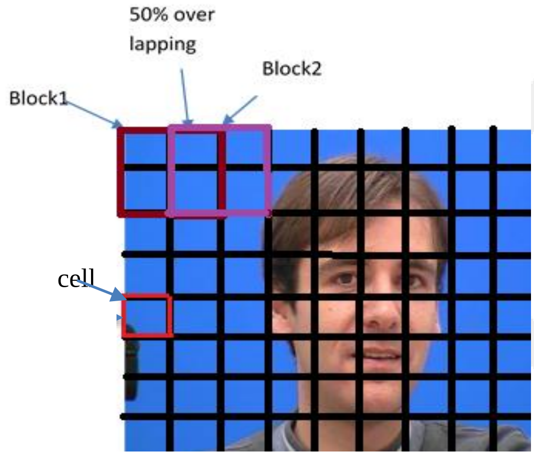


Figure 3.3 Frame number 3000 from the talking face dataset.

For example, if an image is consisting of 64x 64 pixels, then it can be divided to 8 cells, every cell contains 8 pixels. The block is consisting of 4 cells, so one block will represent 16x16 pixels. Moreover, if 50 % overlapping is considered, then every row will contain 7 blocks as well as the column.

3.2.3 Features Vector Computation:

Image in Figure 3.3 is used as example of vector computation.

1- Calculate the magnitude and direction of every pixel in the cell. For a cell that contains 8x8 pixels, magnitude and direction are shown in Tables 3.1 and 3.2 respectively.
2- Apply orientation binning, so that every cell will contain 9 values as shown in Table 3 and Figure 3.4.

3- Calculate the block vector, the block consists of 2x2 cells, and every cell have 9 values. That gives: 2x2x9=36 block vector as shown in Figure 3.5.
4- Normalize the block vector using Equation 3.6.
5- The whole image consists of 7x7 blocks, with 36 value for every block, the result will be: 7x7x36=1764

The length of HOG feature vector of the provided image is 1764x1.

Table 3.1 Magnitude of 64 pixels in the cell.

157.6864	113	112.0402	111	113.5341	116.3486	119.0042	174
113.0398	5	8.246211	14.03567	15.81139	10	13.45362	120.3536
116.4302	12.08305	14.31782	14.76482	18.68154	22.2036	21.09502	129.3136
124.1008	5	1	7.211103	16.12452	15.23155	16.27882	137.3645
120.2664	8.062258	6.708204	22.02272	20.51828	14.21267	16.55295	146.113
119.8165	38.62642	48.02083	43.46263	34.52535	24.04163	17.08801	152.1184
166.7843	61.18823	46.09772	32.75668	32.06244	25.96151	18.02776	161.3103
230.6014	156	161.0279	166.0482	168.003	164.3046	161.2514	227.1673

Table 3.2 Direction of 64 pixels in the cell.

45.25693	90	91.53434	90	84.44005	85.56353	89.51853	133.6028
1.52077	90	104.0362	85.91438	71.56505	53.1301	41.98721	171.3986
4.92711	65.55605	65.22486	61.69924	74.47589	82.23483	58.57043	176.0091
2.309063	53.1301	90	33.69007	29.74488	66.80141	100.6197	175.8252
176.1859	97.12502	26.56505	50.52754	43.02507	39.28941	64.98311	172.9236
14.50017	68.74949	58.62699	66.97451	79.99202	73.07249	69.44395	177.7395
20.71688	78.69007	77.47119	77.66091	93.57633	105.6422	123.6901	176.4458
139.7491	90	88.9325	91.38035	89.65896	86.51068	86.80046	42.68088

Table 3.3 Nine bins of 8x8 pixels in the cell.

3	5	5	5	5	5	5	7
1	5	6	5	4	3	3	9
1	4	4	4	4	5	3	9
1	3	5	2	2	4	6	9
9	5	2	3	3	2	4	9
1	4	3	4	4	4	4	9
2	4	4	4	5	6	7	9
7	5	5	5	5	5	5	3

Table 3 is represented in Figure 3.4.

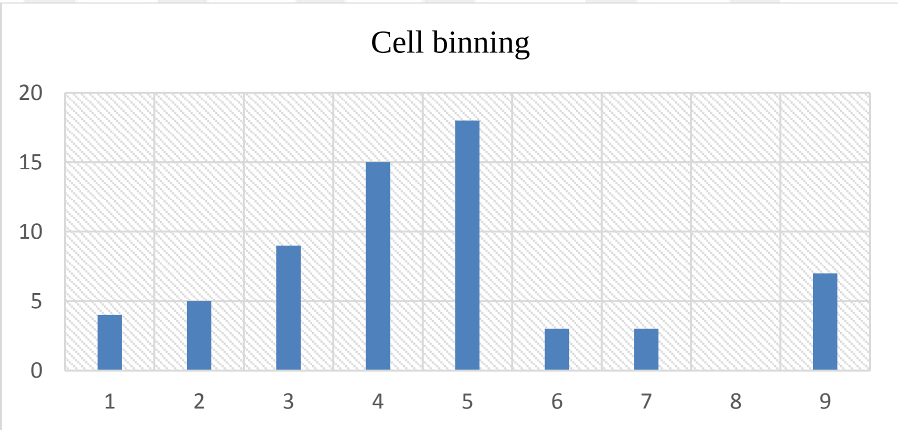


Figure 3.4 Binning representation of 64 pixels in the cell

Where 1 represents the pixels with orientation from 0 to 10 degrees, 2 represents the pixels with orientation from 11 to 20 degrees and so on.

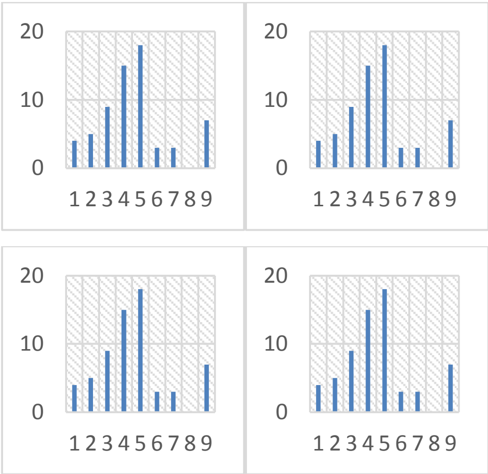


Figure 3.5 Binning of a block vector consists of 2x2 cells.

3.3 Classification

Classification is the term that refers to assigning an object to a category, due to the features of the object. There are several methods of classification such as artificial neural network, support vector machines, nearest neighbors, decision trees, etc.

Generally, in classification problems, a training set of objects are given. The features of these objects are extracted using the proposed algorithm.

Then these features are fed to the classifier with or without their labels. If the training dataset has class labels, then the process is known as supervised classification, otherwise the classification is unsupervised. Training process is shown in Figure 3.6.

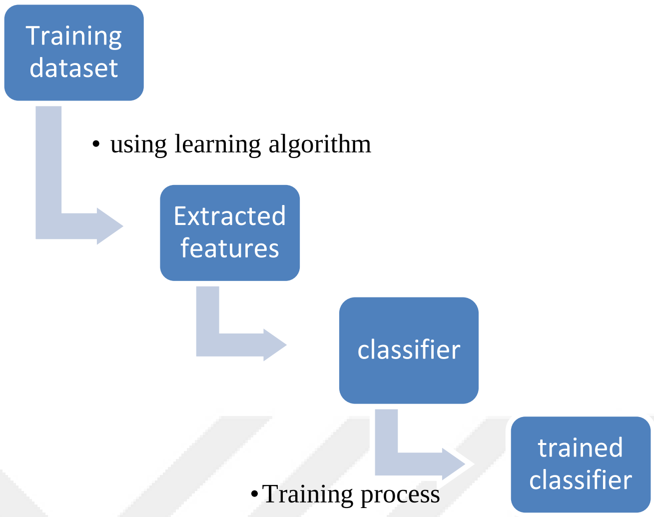


Figure 3.6 Classifier training steps.

After training is performed, the trained classifier can be used to classify new dataset, which is known as testing process. In this process the same learning algorithm is used to extract features from new objects, then these features are fed to the trained classifier to decide which category they belong to. Testing process is shown in Figure 3.7.

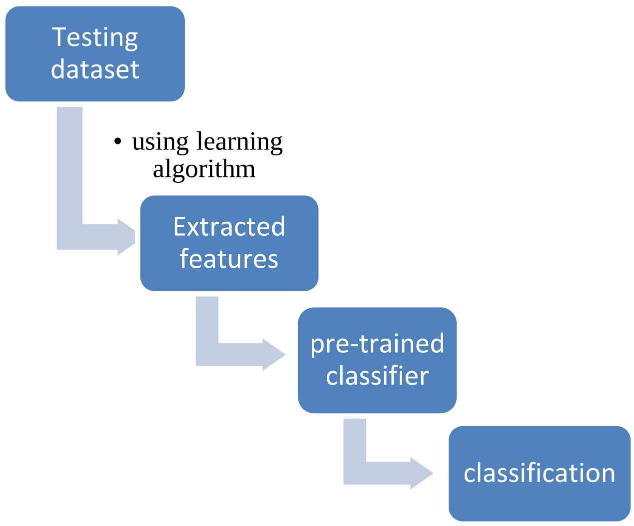


Figure 3.7 Testing process.

3.3.1 Artificial Neural Network Classifier

Artificial Neural Network is an interconnected group of neurons, which is used for information processing. These neurons or nodes are similar to biological neurons in function. The neuron receives information as inputs, processes them according to mathematical model, and then produces an output [33].

The principle function that is used for processing the input is the sigmoid. In addition, several functions can be used, such as step function and piecewise linear functions [33].

The neuron consists of three stages which are input stage, activation function or transfer function and finally, the neuron output S.
The input stage consists of the inputs X and their weights W. The summation of the inputs and their corresponding weights S_j is given in Equation 3.7.

$$S_j = \sum w_{ij} x_i \tag{3.7}$$

The stages are shown in Figure 3.8

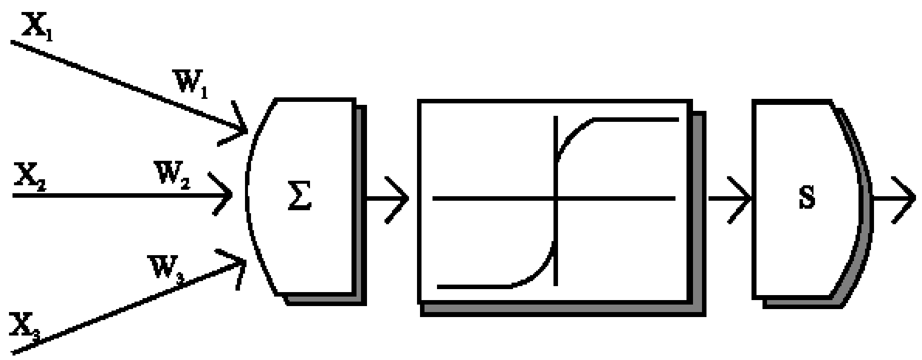


Figure 3.8 Neuron architecture.

There are two main categories of ANN:

1.Feed forward neural networks, where information goes just in one direction, from input to output direction. An example of this category is the Multi-layer perceptron (MLP), which is shown in Figure 3.9.

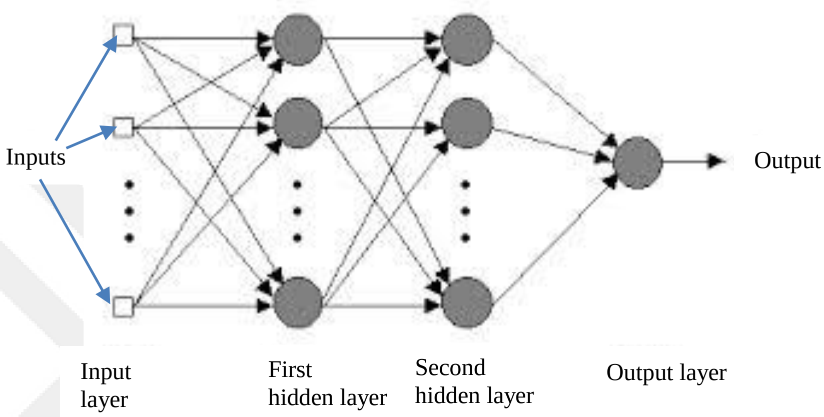


Figure 3.9 MLP Artificial Neural network architecture

2.Feed backward artificial neural networks, and it is also called backpropagation artificial neural networks, where a feedback is given from output layer to input layer, or from a hidden layer to input layer in order to improve weights [34]. Example of the architecture of backpropagation artificial neural network is shown in Figure 3.10.

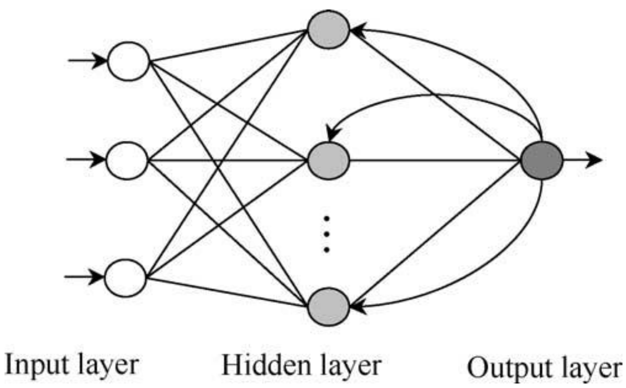


Figure 3.10 Backpropagation artificial Neural network architecture.

In this thesis a feed forward neural network has been used. It consists of input layer, one hidden layer with 15 neurons and one output layer. The activation function is the tan-sigmoid transfer function (tansig).

3.3.2 Decision Tree Classifier (DT)

Decision tree (DT) is one of the well-known machine learning techniques, it consists of three basic elements: decision node, branch and leaf.

The decision node specifies the test features, while the branch is corresponding to one of the possible outcomes, and the leaf refers to the class of the object [35]. Figure 3.11 shows a decision tree for face and eye detection.

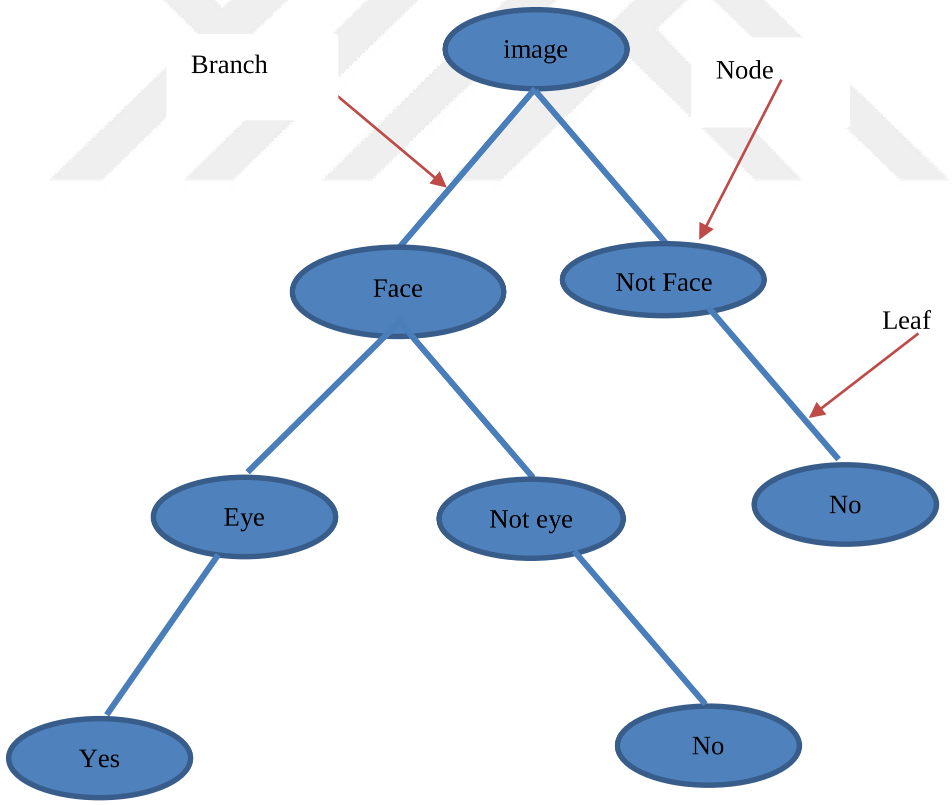


Figure 3.11 Decision Tree for face and eye detection.

In this work default parameters have been used as implemented in the matlab software.

3.4 Iris Detection

Iris detection is an essential step before tracking. After the detection of face and eyes, iris detection became easier. That is because the area that contains iris became smaller. Two methods will be used for iris detection: Hough transform and means of gradients.

3.4.1 Hough Transform

The Hough Transform is an algorithm that was presented by Paul Hough in 1962 in order to detect lines in digital images.

In 1972 Richard Duda and Peter Hart have expanded the original Hough Transform to the generalized Hough Transform, which is now used to detect more shapes such as circles [36].

The main principle of Hough Transform, that any curve in an image can be presented as a set of points. These points can be linked to their coordinate information using a set of parameters. So that curve can be represented by a mathematical equation that clarify the relationship between the points and the coordinates [37].

The main principle of Hough Transform that was mentioned in previous paragraph, can be applied to detect circular shapes in an image. This transformation is known as Circle Hough Transform (CHT). using the equation of circle which is given in 3.8:

$$(x - a)^2 + (y - b)^2 = r^2 \tag{3.8}$$

where the point (a, b) is the center of the circle, and r is the radius

(3.8) can be written as:

$$x = a + r.cos\theta \text{ and } y = b + r.sin\theta \tag{3.9}$$

where $x, y, r \in R$, θ is the angle between x-axis and the vector of the point (x,y).

A point (x_i, y_i) will transferred as a center of a new circle of the same radius in the Hough space (H). If there is a set of points that would be transferred to H space, then these points will produce many circles. These circles will intersect in one point that will be the center of a circle represents these points. Figure 3.12 illustrates these circles.

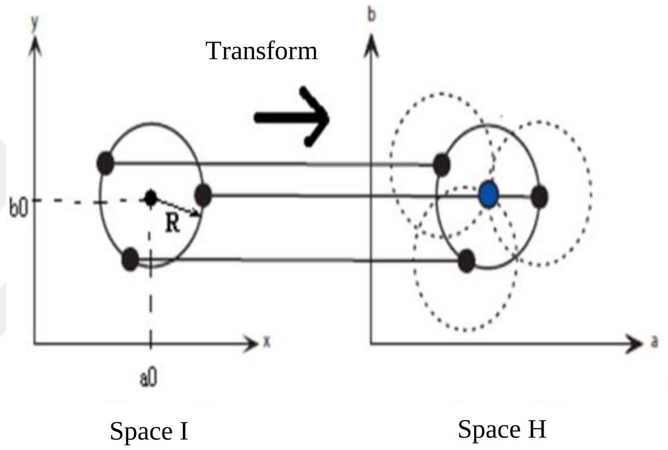


Figure 3.12 Transformation of many points from I to H space [38].

In this work, before applying the Hough Transform to the image, a thresholding have been implemented to the area of detected eyes. Then the circle Hough transform have been used to search for circles, and the founded circles have been considered as the iris of the eyes

3.4.2 Means of Gradients

The gradient is a vector-valued function. The direction of that function is toward the direction of the greatest rate of increase of the function, while the slope of the graph in that direction represents it's magnitude [39]. Figure 3.13 shows the direction of gradients of a function.

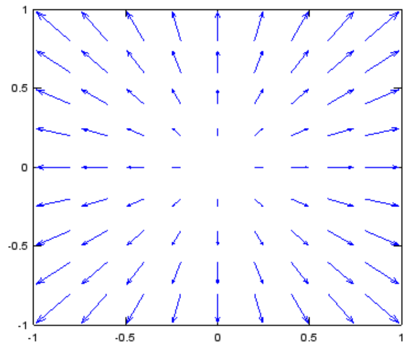


Figure 3.13 Blue arrows Points in the direction of greatest increase of the function.

When the gradient is calculated for every point, then the resultant vectors can be called the vector field of the plane, which was shown in Figure 3.13.

The iris detection can be obtained by calculating the gradient vector field in the image, then computing the dot products between the normalized displacement vectors d_i and the gradient vectors g_i. As can be notice from Figure 3.14. Where the normalized displacement vector d_i should have the same orientation as the gradient g_i [40].

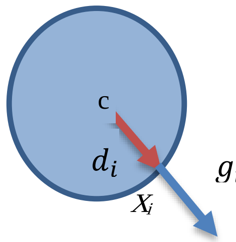


Figure 3.14 Representing gradient and displacement vectors in a circle of center c.

The gradient points in the direction of increase of a function, so in the image of eye, since that iris is darker than sclera, the gradient's direction is outwards[41].
To find the optimal centre c*of a circular object in an image with pixel positions x_i , where $i \in \{1,\dots N\}$, the following formula (3.10) is used.

$$c^* = argmax\left(\frac{1}{N}\sum_{i=1}^{N}\left(d_i^T g_i\right)^2\right) \tag{3.10}$$

where d_i , g_i ϵ R^n are the arguments of the maxima, which refers to the inputs, or arguments, at which the function's outputs are maximized.

The displacement vector d_i can be calculated by the following formula (3.11).

$$d_i = \frac{x_i - c}{\| x_i - c \|_2} \forall i : \| g_i \| = 1 \tag{3.11}$$

Where, c ϵ R.

When applying Equation (3.10) to the image contains iris, then the center can be obtained as c* and then the iris is detected.

CHAPTER 4

EXPERIMENTAL RESULTS AND EVALUATION

This study focuses on face and eye detection for the purpose of iris detection. Two classifiers are considered for face and eye detection, which are neural network and decision tree. Then Hough transform and means of gradients are used for detecting the iris.

4.1 Datasets

Face and eye detection process requires datasets for training and testing. Information about these datasets can be illustrated in Table 4.1.

Table 4.1 Datasets.

Dataset's Name	Purpose of use	Number of images	Description
Cropped version of Extended Yale Face Database B	Training Face detector (positive samples)	16128	Images of 28 subjects under 9 poses, in 64 illumination conditions.
CBCL Face Database #1	Training Face detector (negative samples)	4,548	Images that does not contain faces
(MMU) iris database	Training eye detector (positive samples)	1446	Images of left and right iris, some of images with glasses and different orientations.
BioID Face Database	Testing face and eye detectors	1521	Images of 23 subjects with different facial views
Talking face video	Testing face and eye detectors and iris tracking	5000	Frames from a video of conversation between two subjects

More information and examples from the datasets are in the following subsections.

4.1.1 The Extended Yale Face Database B

The Extended Yale Face Database B [42] consists of 16128 images of 28 subjects under 9 poses, these images have been taken in different 64 illumination conditions.

These images are grey scale images, which stored in Portable Grey Map (PGM) format, with size of 640 x 480 pixels. In this work the cropped version of The Extended Yale Face Database B was used [43], where the size of each image is 168x192.

Figure 4.1 Examples of the cropped version of The Extended Yale Face Database B [44]

4.1.2 CBCL Face Database #1

The CBCL Face Database #1 was generated by the Center of Biological and Computation Learning (CBCL) in Massachusetts Institute of Technology [44]. In this work, the training set that includes 4,548 of non-faces images was used. The dimensions of the CBCL Face Database #1 images are 19x19 pixels and were generated from images that does not contain faces [45].

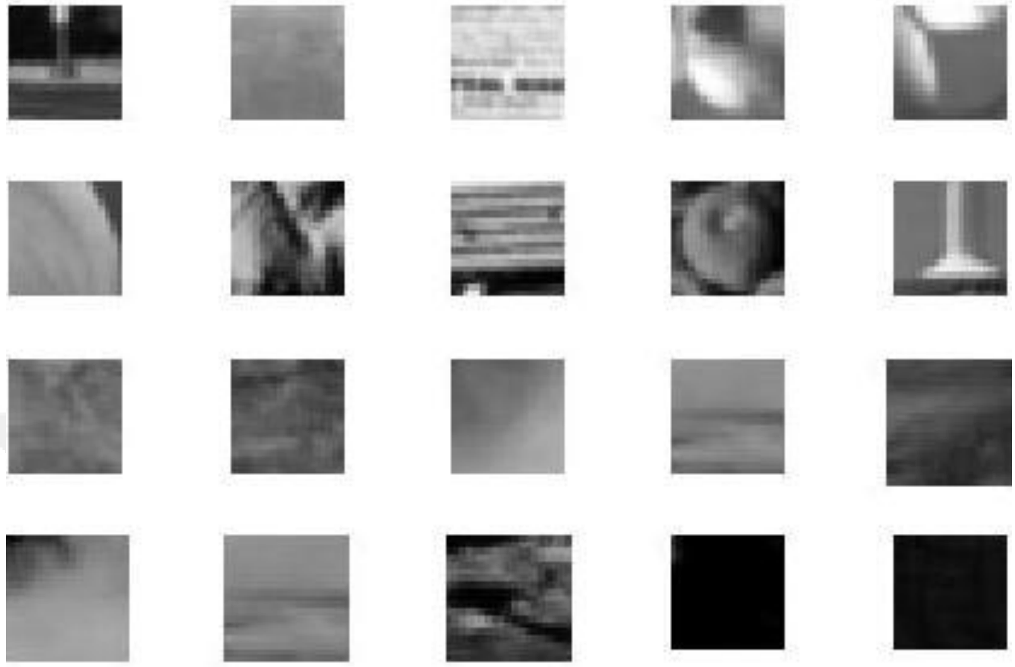


Figure 4.2 Examples of non-faces images from CBCL Face Database #1[34]

4.1.3 MMU Iris Database

Multimedia University (MMU) iris database consists of two groups: MMU1 and MMU2. The MMU1 consists of 450 images. These images represent the irises of 45 persons at which each person has 5 images of the right iris and 5 images of the left one. The MMU2 includes 996 images of irises [46].

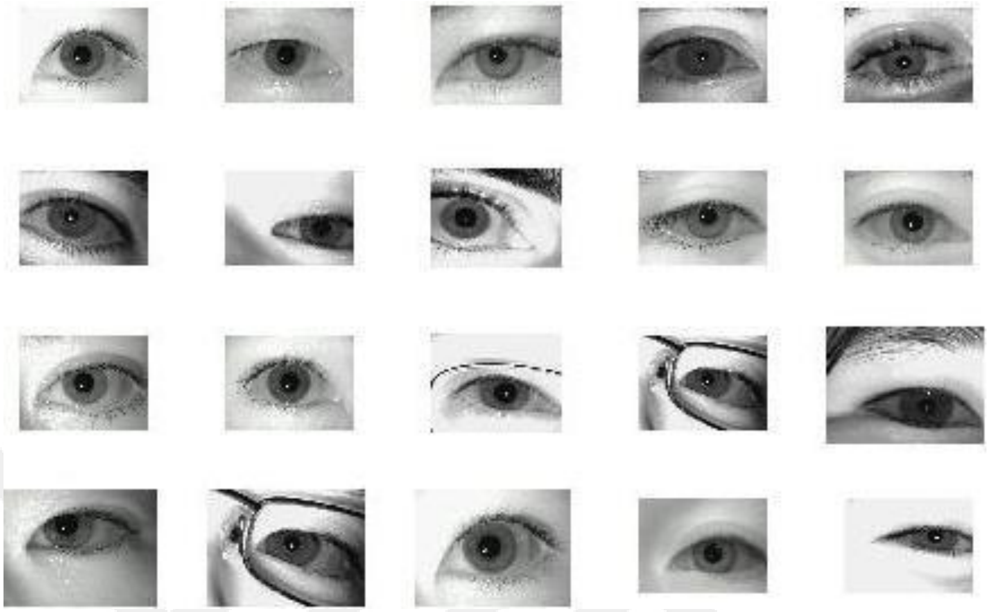


Figure 4.3 Examples of iris images from the MMU iris database[46].

4.1.4 BioID Face Database

The BioID Face Database is constructed from 1521 gray level images with reseliotion of 384x286 pixels[47]. These images represent different facial views of each one of the 23 subjects who are partisipated in this dataset. In some images the participants wore glasses, while hair around face affects some other images. Deffirent illumination conditions and scales are represented in this dataset, so it is one of the high challinging datasets.

Figure 4.4 Examples of faces from the BioID Face Database.

4.1.5 Talking Face Video

This dataset consists of 5000 frames that were extracted from a video. The video was recorded for a subject talking in a conversation with another person, at which different facial expressions can be noticed [48].

Figure 4.5 Examples from the talking face dataset.

4.2 Evaluation Measures

Evaluation measures are used to show the efficiency of the used methods on test data. In this work, two categories of evaluation measures have been used; metrics evaluation and relative error measure.

4.2.1 Metrics Evaluation Measures

In face and eye detection, it is important to choose an efficient classifier in order to get the best results. The profeciency of the classifier is determined by using the commonly used metrics in the literature[49] as shown in Table 4.2.

Table 4.2 Evaluation measures.

Evaluation measures	Used equations	Equation number
Detection Rate	Detection Rate = $\frac{TP}{TP+FN}$	4.1
Positive Prediction	Positive Prediction = $\frac{TP}{TP+FP}$	4.2
Negative Prediction	Negative Prediction = $\frac{TN}{FN+TN}$	4.3

where True Negative, (TN): Number of frames where the object is absent and detected as absent.True Positive, (TP): Number of frames where object is present, and the detector can detect it as found.

False Negative, (FN): Number of frames where the object is present but the classifier can not recognize it, or recognize another unwanted object.

False Positive, (FP): Number of frames where object is absent, while the classifier detects another object as the required object.

Equations from 4.1 to 4.3 are used to choose the classifier with the higher detection rate, which are used in face and eye detection, then the winner classifier is used to detect face and eye before iris tracking.

4.2.2 Estimated Eye Centres Evaluation

The evaluation of estimated eye center depends on the ground truth of the dataset. In this work the method of normalized error that was introduced by Jesorsky et al. has been used [50].

The normalized error method depends on calculating the Euclidean distance between the correct eyes centers and the estimated eyes centers, then choose the max of the two values and divide it by the Euclidean distance between the correct eye centers. The Equations (4.4) and (4.5) compute these distances.

$$e_r = \sqrt{(x_c - x_e)_r^2 + (y_c - y_e)_r^2} \tag{4.4}$$

$$e_l = \sqrt{(x_c - x_e)_l^2 + (y_c - y_e)_l^2} \tag{4.5}$$

where e_r and e_l are the Euclidean distances between the correct and estimated right and left eyes centers respectively.

The distance *d* between the correct left and right eyes canters is given by the following equation.

$$d = \sqrt{(x_{c_l} - x_{c_r})^2 + (y_{c_l} - y_{c_r})^2} \tag{4.6}$$

The error is normalized to remove scaling effect using Equation (4.7).

$$e = \frac{1}{d} * max\,(e_r, e_l) \tag{4.7}$$

where e is the relative error .

In order to perform detection of the iris, the relative error e must be less than 0.1 which it's diameter.

4.3 Experimental Results of Face and Eyes Detection

The experiments have been implemented in the MATLAB environment using Core i3 (370M) PC with 4 GB DDR3 memory.

Face and eyes detection process consists of two phases: Training phase and testing phase. The two phases are explained in the following subsections.

4.3.1 Training Phase

4.3.1.1 Face Detector Training

The cropped version of the Extended Yale Face Database B represents the positive sampels of faces and the CBCL Face Database #1 denotes the negative samples. These databases are used to train an artificial neural network and a decision tree classifiers.

The training results of both ANN and DT classifiers are shown in Table 4.3.

Table 4.3 Training results of ANN and DT classifiers for face detection.

Evaluation measures	ANN	DT
Training Time (Average)	67.991 s	100.25 s
Detection Rate	0.9849	0.9955
Positive Prediction	0.9996	0.9951
Negative Prediction	0.9919	0.9976

4.3.1.2. Eye Detector Training

In order to detect eyes, an ANN classifier and a DT classifier are trained using the images of eyes from the MMU iris database as positive samples, and the non-faces images from the CBCL Face Database #1 as negative samples. The results of training are shown in Table 4.4.

Table 4.4 Training results of ANN and DT classifiers for eye detection.

Evaluation measures	ANN	DT
Training Time (Average)	11.46 s	6.336 s
Detection Rate	0.9585	0.9903
Positive Prediction	0.9631	0.9862
Negative Prediction	0.9602	0.9906

4.3.2 Testing Phase

In the testing phase, the trained classifiers that have been introduced in subsection 4.3.1 are also used to search the images for faces. However, to get good results, images are processed befor testing.

The processing operation is implemented as follows:

(i) The image are resized to 160x240 pixels in order to reduce calculations as in Figure 4.6.

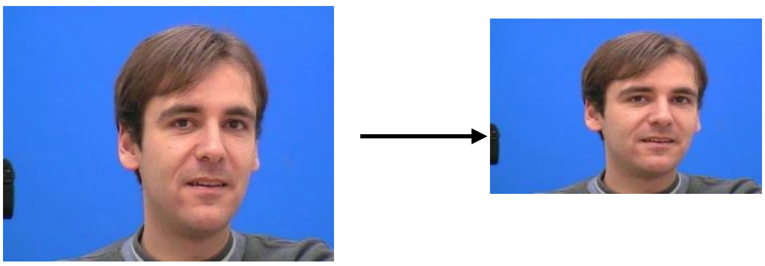
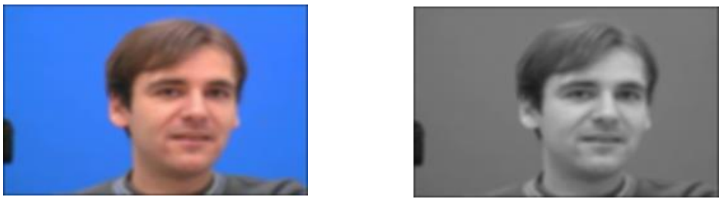


Figure 4.6 Test image (to the left) and resized image (to the right).

(ii) A Gaussian filter is applied to the image to eliminate lightning change effects, then the image is converted to gray scale. As in Figure 4.7.



a b

Figure 4.7 (a)Image after applying Gaussian filter, (b) After converting to gray scale.

(iii) After image processing, a sliding window is used to slide over the image. The Histogram of oriented gradients (HOG) is computed for every window from the start until the end of the image. Sliding window process is illustrated in Figure 4.8.

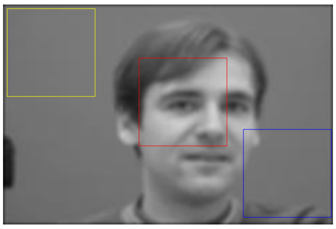
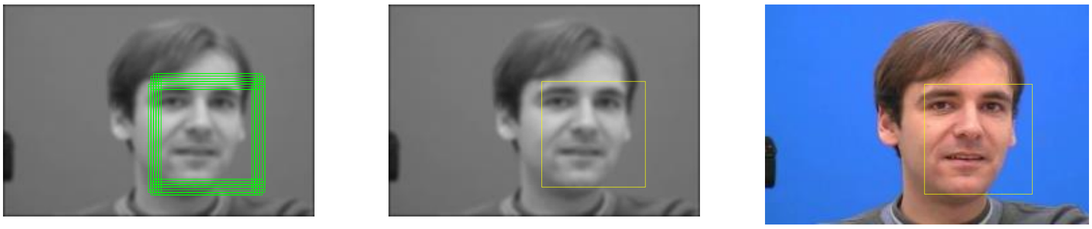


Figure 4.8 The yellow box represents a sliding window at the beginning of the image that does not contain a face. The red box contains a face, while the blue one does not contain a face and it is at the end of the image.

(iv) Then every window is classified as positive or negative using the pretrained ANN and DT classifiers. Moreover, many windows will be classified as positive. So there is a need to choose the best bounding box that contains the face. This can be done by calculating the mean of these boxes. As in Figure 4.9.



a b c

Figure 4.9 (a) Bounding Boxes after classification. (b) Chosen bounding box. (c) Applying Bounding Boxes positions on the colored image.

4.3.2.1 Face Detector Testing

The tests are performed on 150 images from the BioID Face Database, and the Talking Face video database. Results are presented in tables 4.5 and 4.6

Table 4.5 Face Detection rate for the first 150 images of the Bio ID Face Database.

Evaluation measures	ANN	DT
Detection Rate	0.933	0.873
False positives	0.067	0.127

Discussion: As seen in Table 4.5 the detection rate is 0.933 at which 10 images are classified incorrectly, these images will be excluded when applying eyes detection.

Table 4.6 Face Detection rate for 150 images of the Talking Face database.

Evaluation measures	ANN	DT
Detection Rate	1	1
False positives	0	0

Discussion: Table 4.6 shows that face have been detected in all images.

4.3.2.2 Eye Detector Testing

The test is performed on 140 images from the BioID Face Database, at which face was detected correctly. The left eye refers to the subject's left eye which appears on right in the image, and the right eye appears at left. Results are shown in Table 4.7

Table 4.7 Eye detection rate for 140 images from BioID Face Database.

Evaluation measures	ANN			DT		
	L_eye	R_eye	Both_eyes	L_eye	R_eye	Both_eyes
Detection Rate	0.65	0.63	0.55	0.47	0.44	0.41

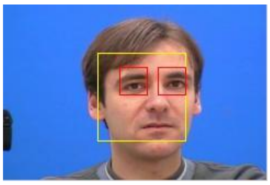
Discussion: The detection rate is higher when using ANN classifier.

Table 4.8 Eye detection rate for 150 images from the Talking Face video database.

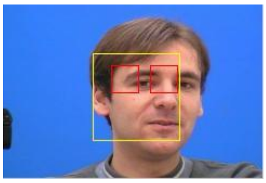
Evaluation measures	ANN			DT		
	L_eye	R_eye	Both_eyes	L_eye	R_eye	Both_eyes
Detection Rate	0.76	0.77	0.71	0.49	0.86	0.49

Discussion: The detection rate of right eye is higher when using Decision Tree classifier, while for both eye detection and left eye detection, the Artificial Neural Network performs higher detection rate.

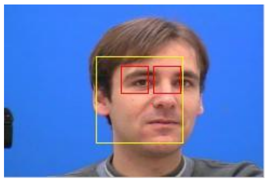
Examples of face and eye detection for images from the Talking Face video database can be seen in Figures 4.10 and 4.11.



a

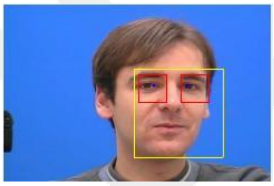


b

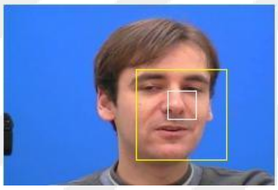


c

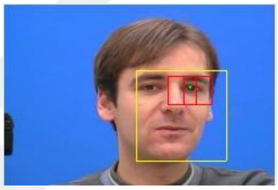
Figure 4.10 Detection is implemented using DT classifier, in (a) face and both eyes were detected. (b) none of the eyes have been detected. (c) right eye has been detected correctly while left eye is not detected.



a



b



c

Figure 4.11 Detection is implemented using ANN classifier, in (a) face and both eyes were detected. (b) none of the eyes have been detected. (c) left eye has been detected correctly while right eye is not detected.

Examples of face and eye detection from the BioID Face Database using dessicion tree classifier can be seen in Figures 4.12,4.13,4,14 and 4.15 .

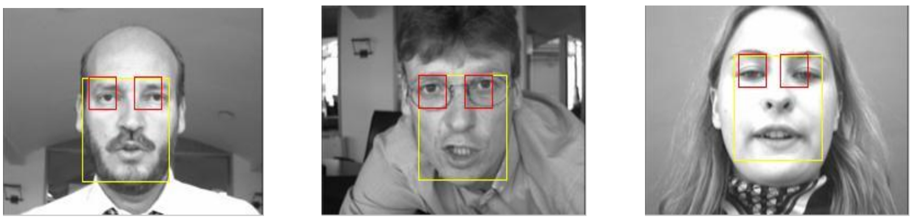


Figure 4.12 Examples of correct face and eyes detection using DT classifier.

a b

Figure 4.13 (a) Correct left eye detection and false right eye detection. (b) Correct right eye detection and false left eye detection.

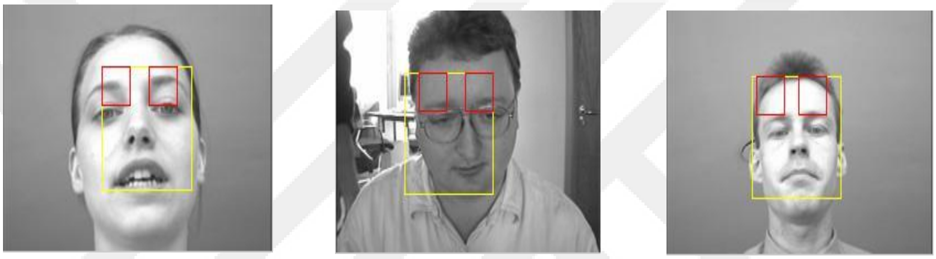


Figure 4.14 Examples of correct face detection with false eyes detection.

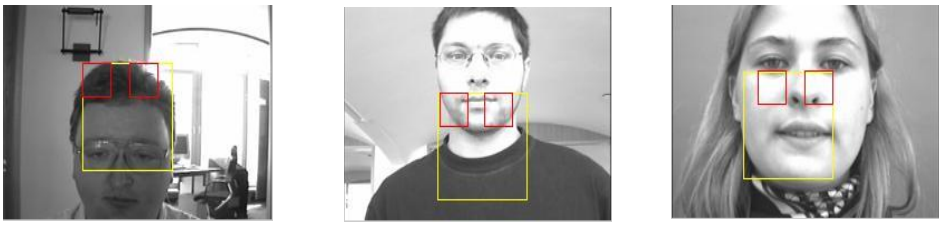


Figure 4.15 Examples of false face and eyes detection.

Examples of face and eye detection from the BioID Face Database using ANN classifier can be seen in Figures 4.16,4.17,4,18 and 4.19 .

Figure 4.16 Examples of correct face and eyes detection using ANN classifier.

a b

Figure 4.17 (a) Correct left eye detection and false right eye detection. (b) Correct right eye detection and false left eye detection.

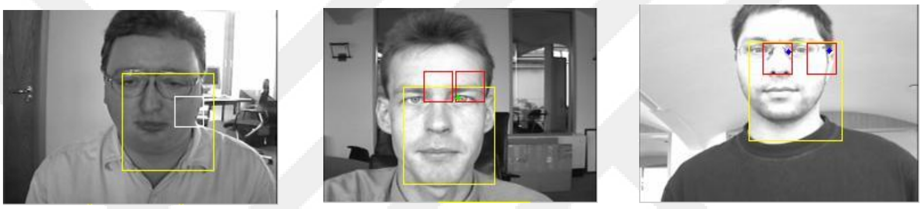


Figure 4.18 Examples of correct face detection with false eyes detection.

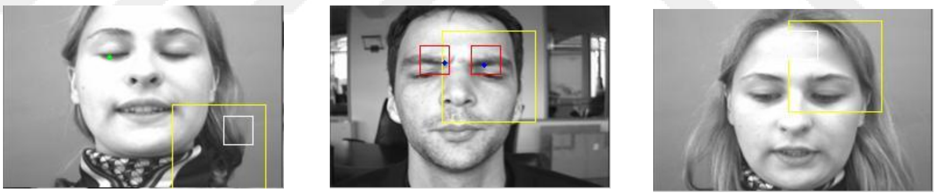


Figure 4.19 Examples of false face and eyes detection.

4.4 Experimental Results of Iris Detection

The evaluation is done by two methods: calculating the detection rate and evaluation with respect to the ground truth table.

4.4.1 Evaluation by Calculating Detection Rate

Iris tracking is performed after face and eye detection. Using artificial neural network classifier, detection rate of face and eyes was higher, so the ANN classifier have been used in face and eye detection before iris detection.

Testing is implemented on the Talking face dataset . Methods that have been used are: Hough transform and means of gradients which have been explained in chapter 3.

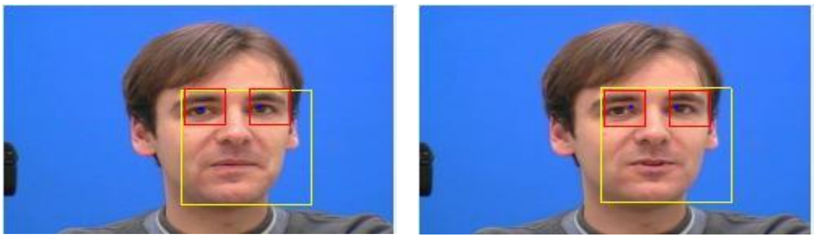
Results of iris tracking using Hough transform and means of gradients can be shown in Table 4.9.

Table 4.9 Eye detection rate for 150 images from the Talking Face video database after excluding images with false detected eyes (Evaluation by eyes).

Evaluation measures	Hough transform			Means of gradients		
	L_eye	R_eye	Both_eyes	L_eye	R_eye	Both_eyes
Detection Rate	0.94	0.3	0.3	0.873	0.93	0.827

Discussion: From table 4.9 it can be noticed that Hough transform detects left eye with 0.94 detection rate, while the detection rate of right eye is 0.3. On the other hand detection rate of right eye by means of gradients is 0.93.

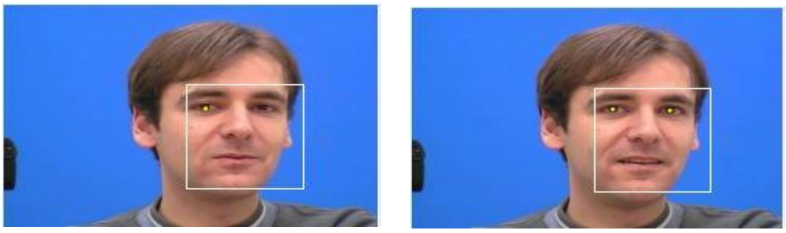
Examples of iris tracking using Hough transform can be seen in Figure 4.20.



a b

Figure 4.20 In (a) both irises where correctly detected, while in (b) only left iris was correctly detected.

Examples of iris tracking using means of gradients method can be seen in Figure 4.21.



a b

Figure 4.21 In (a) only right iris was correctly detected. In (b) both irises where correctly detected.

4.4.2 Detection Evaluation with Respect to The Ground Truth Table

The ground truth of the talking face video dataset contains many facial points, the irises points have been selected for the 150 images that have been tracked.

The normalized error have been evaluated as mentioned in 4.2.2. Example of calculations of the error from the means of gradients method can be shown in Tables 4.10 and 4.11.

Table 4.10 Examples of correct and estimated eye centers.

Right eye		Left eye		Right eye		Left eye	
x_c	y_c	x_c	y_c	x_e	y_e	x_e	y_e
392.96	270.59	503.98	275.35	391.94	266.19	502.94	266.19
392.64	271.46	503.64	276.05	394.23	267.25	505.24	267.25
392.11	271.99	503.36	276.56	391.63	270.41	502.63	270.41
397.18	271.69	503.2	276.2	394.17	266.8	505.17	266.8
391.66	271.26	502.9	275.65	392.45	265.75	503.45	265.75

Table 4.11 Examples of calculations of normalized error.

e_r	e_l	Max(e_r, e_l)	*d*	*e*
4.51	9.23	9.22	111.13	0.083
4.50	8.94	8.94	111.10	0.081
1.65	6.196	6.196	111.35	0.06
5.74	9.60	9.60	106.12	0.09
5.57	9.92	9.92	111.33	0.089

Results of iris tracking using the normalized error evaluation method when using means of gradients method can be shown in Table 4.12.

Table 4.12 Eye detection rate for 150 images from the Talking Face video database by means of gradients method.

Evaluation measures	Including false detected eyes			Excluding false detected eyes		
	L_eye	R_eye	Both_eyes	L_eye	R_eye	Both_eyes
Detection Rate	0.77	0.93	0.77	1	1	1

Discussion: As shown in Table 4.12, when excluding images with false detected eyes, the detection rate is 1. While when using all images including images with false eye detection and using rough eye position in the face, the results varies from0.77 to 0.93.

The distribution of normalized error (e < 0.1) is illustrated in Figures 4.22, 4.23 and 4.24.

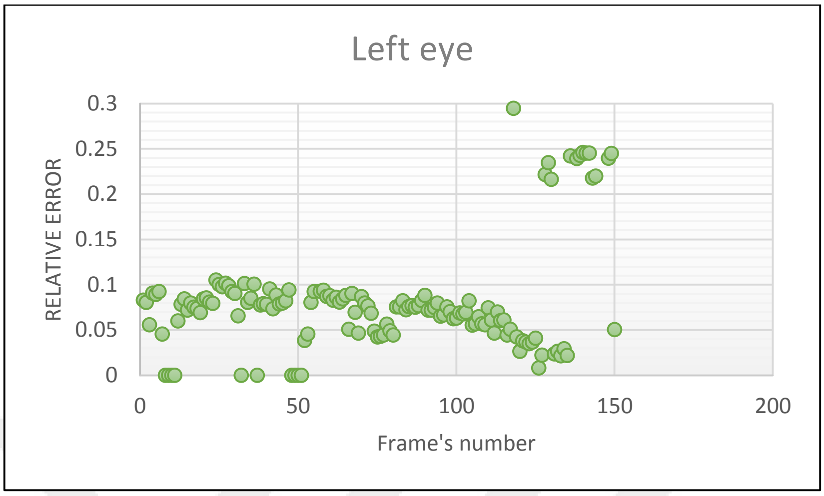


Figure 4.22 Normalized error distribution for left eye after using means of gradients method (e < 0.1).

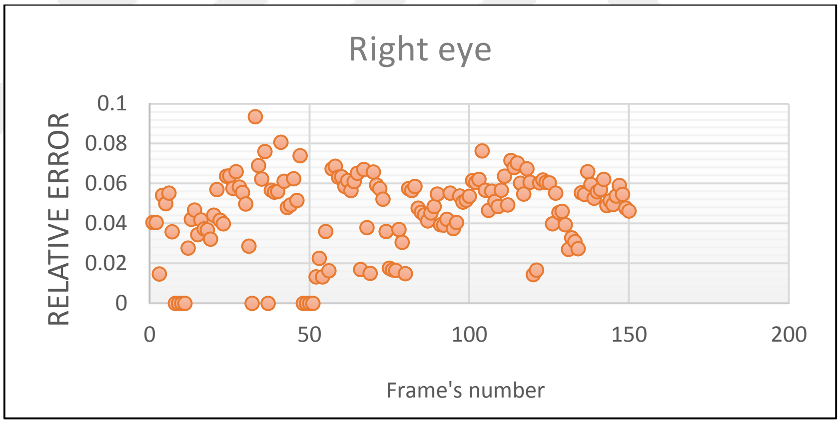


Figure 4.23 Normalized error distribution for right eye after using means of gradients method (e < 0.1).

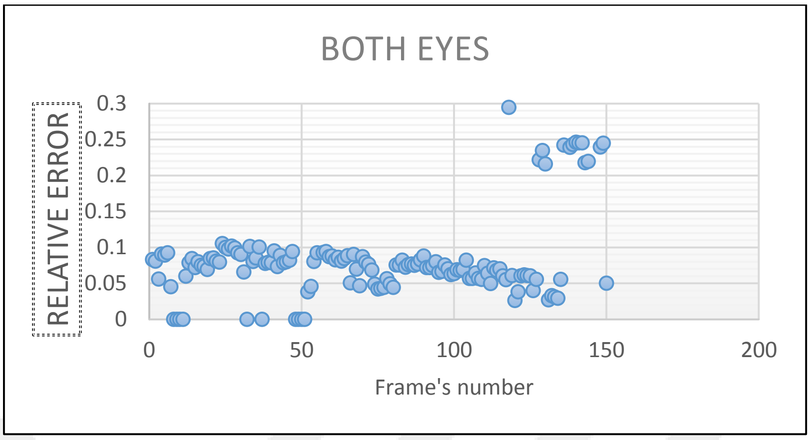


Figure 4.24 Normalized error distribution for both eyes after using means of gradients method (e < 0.1).

Results of iris tracking using the normalized error evaluation method when applying Hough transform can be shown in Table 4.13.

Table 4.13 Eye detection rate for 150 images from the Talking Face video database by Hough transform method (e < 0.25).

Evaluation measures	Including false detected eyes			Excluding false detected eyes		
	L_eye	R_eye	Both_eyes	L_eye	R_eye	Both_eyes
Detection Rate	0.47	0.35	0.33	0.61	0.45	0.43

Discussion: As shown in table 4.13, when excluding images with wrong eye, the detection rate is higher. While when using all images including images with false eye detection and using rough eye position in the face, the results varies from0.43 to 0.61.

The distribution of normalized error (e < 0.25) is illustrated in Figures 4.25, 4.26 and 4.27.

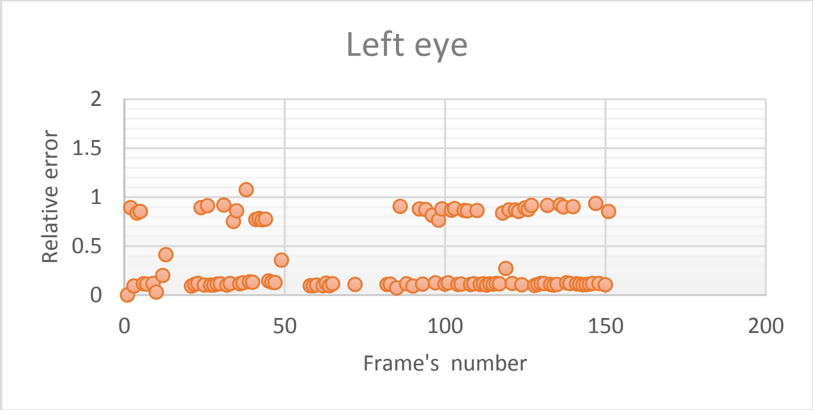


Figure 4.25 Normalized error distribution for left eye after using Hough transform method (e < 0.25).

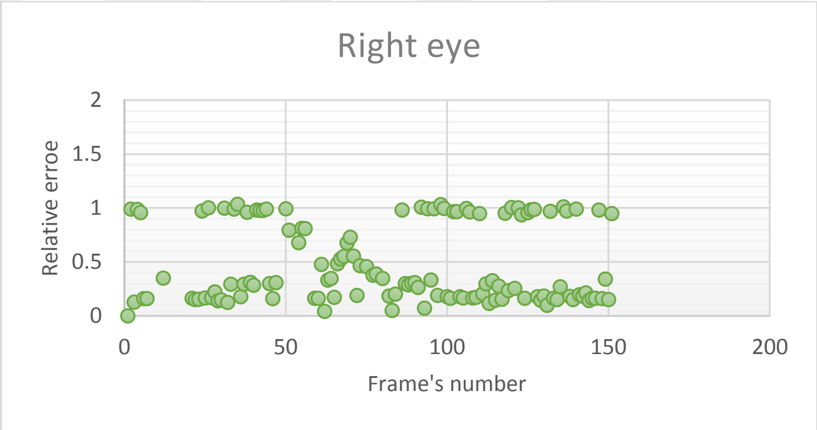


Figure 4.26 Normalized error distribution for right eye after using Hough transform method (e < 0.25).

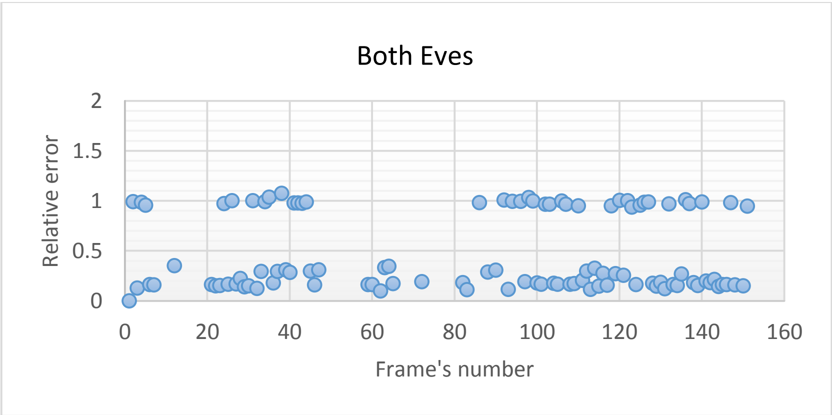


Figure 4.27 Normalized error distribution for right eye after using Hough transform method (e < 0.25).

Summary

In this chapter a comparison has been implemented between ANN and DT classifiers, the results vary with superiority to ANN.

The ANN classifier achieved 0.933 accuracy result when used to detect faces from BioID database, while the Decision tree classifier detected 0.873 of the faces. When the talking face dataset is used for testing, the accuracy of both of the two classifiers is 1.

In the phase of eye detection, employing the BioID dataset in testing presents that ANN shows higher detection rate which was 0.65, 0.63 and 0.55 for left, right and both eye respectively.

On the other hand, the detection rate of decision tree of the same dataset was 0.47, 0.44 and 0.41 for left, right and both eye respectively.

When the test is performed on the talking face dataset, ANN remained the best with 0.76, 0.77 and 0.71 respectively for left, right and both eyes. The DT achieved 0.49, 0.86 and 0.49 detection rates respectively.

After applying face and eye detection, it can be noticed that ANN achieves better results, so it is used for face and eye detection before the tracking process.

The tracking of iris was performed using two methods, Hough transform and the mean of gradients. The experiments have been carried out on the talking face dataset and evaluation has been done using two methods; evaluation by eyes, which is done by counting true detections, and evaluation by calculating the normalized error.

When using the evaluation by eyes means of gradient method shows better results, at which the detection rate is 0.87, 0.93 and 0.827 for left, right and both eyes. Moreover, the Hough transform achieved 0.94 detection rate of left eye, while the right eye was detected with a rate of 0.3, and the same rate corresponds to both eyes.

The normalized error evaluation method gives 0.77 detection rate of left eye using means of gradients method, 0.93 for right eye and 0.77 for both eyes. This rate was computed including undetected eyes. However, when images with undetected eyes were excluded the detection rate increases to 1 for left, right and both eyes.

The Hough transform method gives 0.45 of overall iris detection when the normalized error is 0.25.

CHAPTER 5

CONCLUSION AND FUTURE WORK

The rapid developments in the computer world have increased their usage areas even more. In particular, these developments and changes in computer technology have led to the emergence and even development of digital image processing technology in recent years. With the development of image processing techniques, their use has increased in every area. One of these areas is to detect the iris from the given image.

In this study, iris of the human eye is detected through hybrid models. These models first detect the face, then the eyes, and finally the iris. First, for this purpose, a database was created from the Extended Yale Face B dataset representing the positive samples and the CBCL dataset representing the negative samples. In order to detect faces in images the Histogram of Oriented Gradients (HOG) descriptor was used to extract features from these datasets. The extracted features are then used to train an Artificial Neural Network (ANN) and Decision Tree (DT) classifiers. It has been observed that the ANN performs 93% detection rate when used to detect faces from BioID database, while DT performs 87%. On the other hand, when the talking face dataset is used for testing, the accuracy of both of the two classifiers is 1.

Secondly, to detect eyes from the image, a new database was created from MMU iris data set denoting positive samples and the CBCL dataset denoting the negative samples. The same steps are used to extract eyes separately from face region. ANN classifier also outperforms the DT classifier, at which 71% detection rate is achieved for detecting both eyes using ANN classifier, while DT classifier detects 49% of both eyes from the talking face dataset.

After face and eyes were detected, the extracted region was used to find circles that indicate the iris.

Two methods are discussed to detect the iris: Hough Transform (HT) and the mean of gradients (MoG). In comparison with HT, MoG yields higher detection rate which is 83% for detecting both irises, while HT yields 30% detection rate.

The future challenges of this work includes using another feature descriptor in order to speed up the detection process, besides applying dimensionality reduction techniques for more efficient detection and to upgrade the method for online detection.

Tracking algorithms can be studied to widen the range of detection toward tracking, to perform more efficient and less time consuming trackers.

The study can be extended to involve online tracking, besides gaze tracking to specify where a person is looking.

REFERENCES

[1] Helga, K. Webvision The Organization of the Retina and Visual System, Retrived June 4, 2017 from http://webvision.med.utah.edu/book/part-i-foundations/gross-anatomy-of-the-ey/. January 25, 2012.

[2] Wolpaw R. J., Birbaumer N., McFarland, D.J., Pfurtscheller, G., & Vaughan, T.M. Brain-computer interfaces for communication and control, Clinical Neurophysiology, 113., 767-791. 2002.

[3] Lupu, R. G., & Ungureanu, F. A survey of eye tracking methods and applications. Bul Inst Polit Iasi, 71-86. 2013.

[4] Penzo, M. Introduction to eyetracking: Seeing through your users' eyes. Retrieved June 4, 2017 From http://uxmatters. com/MT/archives/000040. php. 2005.

[5] San Agustin, J., Skovsgaard, H., Mollenbach, E., Barret, M., Tall, M., Hansen, D. W., & Hansen, J. P. Evaluation of a low-cost open-source gaze tracker. In Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications . 77-80. ACM. March, 2010.

[6] Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., & Hengel, A. V. D. A survey of appearance models in visual object tracking. ACM transactions on Intelligent Systems and Technology (TIST), 4(4), 58. 2013.

[7] Yang, M. Object Recognition University of California at Merced, Retrived June 2016, from https://pdfs.semanticscholar.org/

[8] Daugman, J. New methods in iris recognition. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 37(5), 1167-1175. 2007.

[9] Ryan, W. J., Woodard, D. L., Duchowski, A. T., & Birchfield, S. T. Adapting starburst for elliptical iris segmentation. In Biometrics: Theory, Applications and Systems. 2nd IEEE International Conference. 1-7. IEEE. 2008.

[10] Kothari, R., & Mitchell, J. L. Detection of eye locations in unconstrained visual images. In Image Processing, 1996. Proceedings., International Conference Vol. 3. 519-522. IEEE. September 1996.

[11] Matas, J., & Obdržálek, Š. Object recognition methods based on transformation covariant features. In Signal Processing Conference, 12th European IEEE . 1721-1728. September 2004.

[12] Moghaddam, B., Jebara, T., & Pentland, A. Bayesian face recognition. Pattern Recognition, 33(11), 1771-1782. 2000.

[13] Essa, I. A., & Pentland, A. P. Coding, analysis, interpretation, and recognition of facial expressions. IEEE transactions on pattern analysis and machine intelligence, 19(7), 757-763. 1997.

[14] Viola, P. Feature-based recognition of objects. In Proceedings of the AAAI Fall Symposium Series: Machine Learning in Computer Vision: What, Why, and How. 60-64. 1993.

[15] Howarth, J. W., Bakker, H. H., & Flemmer, R. C. Feature-based object recognition. In Autonomous Robots and Agents, ICARA 2009. 4th International Conference 375-379. IEEE February,2009.

[16] Yilmaz, A., Javed, O., & Shah, M. Object tracking: A survey. Acm computing surveys (CSUR), 38(4), 13. 2006.

[17] PETS 2006 Benchmark Data, Retrieved November 2016, from http://www.cvg.reading.ac.uk/PETS2006/data.html. Edited 18 June 2006.

[18] Dinh, T. B., & Medioni, G. Co-training framework of generative and discriminative trackers with partial occlusion handling. In Applications of Computer Vision (WACV), IEEE Workshop 642-649. January 2011.

[19] McKenna, S. J., Raja, Y., & Gong, S. Tracking colour objects using adaptive mixture models. Image and vision computing, 17(3), 225-231. 1999.

[20] Yu, T., & Wu, Y. Differential tracking based on spatial-appearance model (SAM). In Computer Vision and Pattern Recognition, IEEE Computer Society Conference on (1). 720-727. June 2006.

[21] Chen, F. S., Fu, C. M., & Huang, C. L. Hand gesture recognition using a real-time tracking method and hidden Markov models. Image and vision computing, 21(8), 745-758. 2003.

[22] Bouzenada, M., Batouche, M. C., & Telli, Z. Neural network for object tracking. Information Technology Journal, 6(4), 526-533. 2007.

[23] Rowley, H. A., Baluja, S., & Kanade, T. Neural network-based face detection. IEEE Transactions on pattern analysis and machine intelligence, 20(1), 23-38. 1998.

[24] Viola, P., Jones, M. J., & Snow, D. Detecting pedestrians using patterns of motion and appearance. IEEE 734. October 2003.

[25] Jones, M., & Viola, P. Fast multi-view face detection. Mitsubishi Electric Research Lab TR-20003-96, 3, 14. 2003.

[26] Marée, R., Geurts, P., Piater, J., & Wehenkel, L. Biomedical image classification with random subwindows and decision trees. In International Workshop on Computer Vision for Biomedical Image Applications. Springer Berlin Heidelberg. 220-229 October 2005.

[27] Huang, J., Gutta, S., & Wechsler, H. Detection of human faces using decision trees. In Automatic Face and Gesture Recognition, Proceedings of the Second International Conference IEEE, 248-252. October 1996.

[28] Avidan, S. Support vector tracking. IEEE transactions on pattern analysis and machine intelligence, 26(8), 1064-1072. 2004.

[29] Fu, K., Gong, C., Qiao, Y., Yang, J., & Guy, I. One-class SVM assisted accurate tracking. In Distributed Smart Cameras (ICDSC), Sixth International Conference 1-6. IEEE. October 2012.

[30] Ulusoy, I., & Bishop, C. M. Generative versus discriminative methods for object recognition. In Computer Vision and Pattern Recognition, CVPR 2005. IEEE Computer Society Conference (2). 258-265. June 2005.

[31] Hansen, D. W., & Ji, Q. In the eye of the beholder: A survey of models for eyes and gaze. IEEE transactions on pattern analysis and machine intelligence, 32(3), 478-500. 2010.

[32] Dalal, N., & Triggs, B. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference. 1. 886-893. June 2005.

[33] Bouzenada, M., M. C. Batouche, and Z. Telli. "Neural network for object tracking." Information Technology Journal 6.4: 526-533. 2007

[34] Chiang, Y. M., Chang, L. C., & Chang, F. J. Comparison of static-feedforward and dynamic-feedback neural networks for rainfall–runoff modeling. Journal of hydrology, 290(3), 297-311. 2004.

[35] Amor, N. B., Benferhat, S., & Elouedi, Z. Naive bayes vs decision trees in intrusion detection systems. In Proceedings of the 2004 ACM symposium on Applied computing 420-424. March, 2004.

[36] Hough transform Retrived May 2017, from https://en.wikipedia.org/wiki/Hough_transform.

[37] Cherabit, N., Chelali, F. Z., & Djeradi, A. Circular Hough transform for iris localization. Science and Technology, 2(5), 114-121. 2012.

[38] Rhody, H. Lecture 10: Hough circle transform. Chester F. Carlson Center for Imaging Science, Rochester Institute of Technology. 2005.

[39] Gredient, Retrived May, 2017 https://en.wikipedia.org/wiki/Gradient

[40] Timm, F., & Barth, E. Accurate Eye Centre Localisation by Means of Gradients. VISAPP, 11, 125-130. 2011.

[41] Kothari, R., & Mitchell, J. L. Detection of eye locations in unconstrained visual images. In Image Processing, IEEE International Conference (3). 519-522. September 1996.

[42] Georghiades, A. S., Belhumeur, P. N., & Kriegman, D. J. From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE transactions on pattern analysis and machine intelligence, 23(6), 643-660. 2001.

[43] Lee, K. C., Ho, J., & Kriegman, D. J. Acquiring linear subspaces for face recognition under variable lighting. IEEE Transactions on pattern analysis and machine intelligence, 27(5), 684-698. 2005.

[44] Center For Biological and Computation Learning at MIT, CBCL Face Database #1, Retrived March 2017 from http://cbcl.mit.edu/cbcl/software-datasets/FaceData2.

[45] B. Heisele, T. Poggio and M. Pontil. Face Detection in Still Gray Images.
CBCL Paper , Massachusetts Institute of Technology, Cambridge, MA , June 2000.

[46] MMU Iris Database, Retrived March, 2017 from http://www.cs.princeton.edu /~andyz/irisrecognition

[47] BioID Face Database – FaceDB (n.d.). Retrived April, 2017, from https://www.bioid.com /About/BioID-Face-Database.

[48] Talking Face Video. Retrived April, 2017, from http://www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html.

[49] Bashir, F., & Porikli, F. Performance evaluation of object detection and tracking systems. 9th IEEE International Workshop on PETS . 7-14. June 2006.

CURRICULUM VITAE

PERSONAL INFORMATION

Name, Surname	:	Nour N.M. NASSAR
Date of Birth	:	1985, 19 June
Phone	:	543 862 04 98
E-mails	:	no28or.2013@gmail.com

EDUCATION

High School	:	Ramiz Fakhira Secondary School, Gaza / Palestine
BSc.	:	Department of Physics, Science Faculty Islamic University of Gaza, Gaza / Palestine
MSc.	:	Department of Electrical and Electronics Engineering, Ankara Yıldırım Beyazıt Unıversıty

WORK EXPERIENCES

[2009 -2011] Teacher of physics at Basheer El Raies high school, ministry of education, Palestine.

[October 2008- February 2009] Lecturer assistant at Al-Aqsa university, Gaza, Palestine.

[2007- 2008] Lecturer assistant at The Islamic university of Gaza.

LANGUAGES:

Arabic - Mother Tongue.

English - High level.

Turkish - elementary level.

TOPICS OF INTEREST

-Machine Learning
- Image processing
-Video processing