

Online Multiple Face Detection and Tracking from Video

Bahram Lavi Sefidgari

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
July 2014
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Computer Engineering.

Prof. Dr. Işık Aybay
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Asst. Prof. Dr. Adnan Acan
Supervisor

Examining Committee

1. Asst. Prof. Dr. Adnan Acan

2. Asst. Prof. Dr. Önsen Toygar

3. Asst. Prof. Dr. Ahmet Ünveren

ABSTRACT

Online face detection and tracking systems have received increasing interest in the last decade. Face detection and tracking are subfield of biometric information processing and object tracking, respectively. Recent advances in theory and practical implementations made the online detection and tracking systems work in real time. Face detection and tracking system designed and implemented in this thesis exploits a combination of techniques in two topics; face detection and tracking. Face detection is performed on live achieved images from video. Processes exploited in the system are color balance, skin segmentation, and facial image extraction on face candidates. Then a face classification method that uses a Haar classifier is employed in the system. Finally, the result from detection part is engaged with a Kalman filter for tracking candidate faces in reasonable speed of change.

The system is tested in practice and has shown to have acceptable performance for tracking faces within the proposed limits. The developed system also gave satisfactory results for multiple faces in live achieved images within each video frame.

Keywords: Face detection, object tracking, facial feature extraction, Haar classifier, Kalman filter.

ÖZ

Son yıllarda, çevrimiçi yüz tespit ve takip sistemleri artan bir ilgi çekmişlerdir. Yüz tespit ve takibi biyometrik bilgi işleme ve nesne takibi konularının alt alanlarıdır. Teorik ve pratik uygulamalardaki gelişmeler çevrimiçi tespit ve takip sistemlerinin gerçek zamanlı olarak çalışmasını olanaklı kılmıştır. Bu çalışmada tasarımı ve uygulaması yapılan yüz tespit ve takip sistemi her iki alanda varolan yöntemlerin bileşiminden yararlanılarak oluşturulmuştur. Yüz tespiti canlı elde edilen görüntüler üzerinde yapılmıştır. Bu süreçte kullanılan yöntemler şunlardır: renk dengesi, deri bölünmesi, yüz özelliklerinin çıkarılması ve yüzsel aday bölgelerin elde edilmesi. Bundan sonra Haar tabanlı bir sınıflandırma yöntemi kullanılarak yüzler tespit edilir ve bir dörtgen içine alınırlar. Son olarak, yüz tespiti sürecinden elde edilen yüz görüntüleri takip işlemi için bir Kalman süzgeci ile işlenerek kabul edilebilir bir değişim hızında takip edilirler.

Tasarımlanan sistemin pratik olarak test edilip, yüz tespiti ve takibi için kabul edilir bir başarımla sağladığı görülmüştür. Geliştirilen sistemle canlı görüntüler üzerinden çoklu yüzlerin tespit ve takibi için de yeterli başarımla ulaşılmıştır.

Anahtar Kelimeler: Yüz tespiti, nesne takibi, yüzsel özellik çıkarımı, Haar sınıflayıcı, Kalman süzgeci.

To My Mother, My Father, and My Elder Sister

ACKNOWLEDGMENT

I would like to express my special appreciate and thanks to my advisor Assistant Professor Dr. ADNAN ACAN, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a research scientist. Your advice on both research as well as on my career have been priceless. I would like to thank my committee members for serving as my committee members even at hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

A special thanks to my family. Word cannot express how grateful I am to my mother, my father, and my elder sister for all of the sacrifices that you have made on my behalf. Your prayer for me was what sustained me thus far.

I would also like to thank all of my friends who supported me and incented me to strive towards my goal.

TABLE OF CONTENTS

ABSTRACT.....	iii
ÖZ.....	iv
DEDICATION.....	v
ACKNOWLEDGMENT.....	vi
LIST OF FIGURES.....	x
LIST OF PROCEDURES.....	xii
LIST OF PROGRAMS.....	xiii
1 INTRODUCTION.....	1
2 LITERATURE REVIEW.....	4
2.1 FACE DETECTION METHODS.....	4
2.2 FACE TRACKING METHODS.....	7
3 FACE DETECTION AND TRACKING ALGORITHMS IMPLEMENTED IN THE THESIS WORK.....	9
3.1 BASIC CONCEPTS.....	9
3.2 FACE DETECTION.....	10
3.3 FACE TRACKING.....	19
3.4 CHAPTER SUMMARY AND DISCUSSION.....	23
4 IMPLEMENTATION OF FACE DETECTION AND TRACKING SYSTEM....	24
4.1 FACE DETECTION.....	26
4.2 FACE TRACKING.....	28
5 EXPERIMENTAL RESULTS.....	33
5.1 SYSTEM HARDWARE.....	33
5.2 SYSTEM SOFTWARE.....	34

5.3 FACE DETECTION	35
5.4 FACE TRACKING.....	36
6 DISCUSSION AND CONCLUSION.....	39
6.1 DISCUSSION.....	39
6.2 CONCLUSION	40
7 FUTURE WORKS.....	42
REFERENCES.....	43

LIST OF TABLES

Table 1: Experimental Result of Face Detection and Tracking	37
Table 2: Blocking Time of Tracking the Candidate Faces	38

LIST OF FIGURES

Figure 1: The Implemented Face Detection and Tracking System.....	10
Figure 2: Block Diagram of the Face Detection Procedure	10
Figure 3: An Image Without (Left) and With (Right) Color Balance Correction	12
Figure 4: Color Balanced Image (Left) and Skin Segmentation Result (Right)	12
Figure 5: Features of Haar-like method [27].....	13
Figure 6: Computation of $A[x,y]$ Values [28]	14
Figure 7: Computation of the Area Table [37].....	14
Figure 8: The Five Types of Haar-Like Feature are Applied in an Image Based on a $k \times k$ Sub-Window.....	16
Figure 9: Cascade of stages [2].	17
Figure 10: Whole Procedure of Weak Classifier by Joining Five Types of Haar Features	18
Figure 11: The Two Features of Eyes are Shown in Top Row and then Detected Feature Show in Bottom Row. The First Feature Compares the Intensity Levels of Two Eyes to the Intensity Level Across the Bridge of the Nose. The Second Feature Calculates the Difference of Intensity between regions of two eyes across the cheeks.	18
Figure 12: The Feature of Lips is Shown in Top Row and then Detected Feature Shows in Bottom Row.	18
Figure 13: The Whole Structure of Kalman Filter [30]	19
Figure 14: Flowchart of the Main Stages of the System.....	25
Figure 15: Blocking and Tracking of Candidate Faces.....	31
Figure 16: PIRANHA - Five Megapixel.....	34

Figure 17: Candidate Faces by Mean Value of Detection Time.....	36
Figure 18: The Result of Implemented Method for Detection and Tracking for more than 30 People.....	37

LIST OF PROCEDURES

Procedure 1: Color Balance Correction Procedure [35]	12
Procedure 2: Procedure of Skin Segmentation [32].....	13
Procedure 3: Procedure of Kalman Filter [30].....	22

LIST OF PROGRAMS

Program 1: Capturing Frame from Video Sequence.....	26
Program 2 : Implementation of Color Balance	26
Program 3: Implementation of Skin Segmentation	27
Program 4: Implementation of Face Detection Part.....	27
Program 5: Configuration of Kalman Filter in MATLAB.....	29
Program 6: Implementation of Prediction and Correction of Kalman Filter	30
Program 7: Implementation of User Interface for Tracking of Detected Faces.....	32

Chapter 1

INTRODUCTION

This thesis work focuses on the study of face detection and tracking from video streams. Face detection and tracking systems are sub-fields of image processing and they are receiving increasing interest by researchers and industry.

In the last decade, a growing interest has been shown in the use of image processing tools for security. In this respect, both government and private companies require online video to enhance the security level. Universities and private industries allocate huge amount of research funds for the development of security platforms. In this manner, face detection and tracking can be attractive in several uses, in particular for surveillance, high security entrance, care systems, and automatic teller machines (ATM).

Several methods have already been developed for detecting and tracking faces. Each of the developed algorithms has advantages and drawbacks. For example, some algorithms are computationally simple but feature selection is carried out from huge dataset. Hence, training can be too much time consuming. Some algorithms are very sensitive to noise, while the others are very sensitive facial image variations and image conditions. These points are explained in detail for each of the algorithms described below.

Face tracking system is a complex image processing problem in real time applications due to brightness and various conditions of images on the live video sequences. Face detection and tracking are usually applied techniques in combination for image processing. Face detection is applied to find location of the faces in a frame of video, while the tracking algorithm is employed to track extracted faces with known structural properties, which are used regularly in most of the computer vision applications. Face detection algorithms detect faces and extract facial images based on the location of eyes, nose, and lips whereas the tracking algorithms use estimation/correction methods. This makes the combined algorithm more complicated than individual detection or tracking algorithm.

The face detection and tracking methods used in this thesis work are fundamentally based on Haar classification method [27] and Kalman filter algorithm [7], respectively. The implemented detection and tracking methods have unique characteristics as high speed detection and tracking, and robustness in noisy environments.

In this thesis, the MATLAB Simulink platform is used for creating a system with an online digital camera. Consequently, all the implemented algorithms are integrated with this system to process video frames sequentially one after another. All data processing, detection and tracking algorithms are handled within the simulation framework.

The contents of this thesis work are specified as follows:

- Face detection and tracking system should detect, extract faces, and track frontal faces from live video sequences.

- System should work under indoor and outdoor lighting condition.
- System should detect and track more than 30 people.
- System should extract faces if they use sunglasses and hat.

This thesis is structured as follows:

Chapter 2 gives a literature review of the current state of the art in face detection and tracking. Recent research work is cited to show what has already been done in these fields. Chapter 3 provides the derivation of the face detection and tracking model. The algorithm is explained from the basic concept of the Haar-like formalism. Particular attention is given to the tracking system from the basic concept of Kalman filter. Chapter 4 focuses on the algorithms used to implement the face detection and tracking system. Details to reach an easier implementation and lower algorithm complexity are also described. Chapter 5 shows the development process of the face detection and tracking system. A few photos are attached to provide a visual description. In addition, the experimental results of the described algorithms are reported. Chapter 6 summarizes the goals of this thesis, evaluates the performance, discusses the results of the experiments and proposes solutions to improve and develop this face detection and tracking system. Chapter 7 presents some discussion for future works on the thesis topic.

Chapter 2

LITERATURE REVIEW

In the last few years, the state-of-the-art algorithms put significant contributions to theory and practice of face detection and tracking. Due to the need for online surveillance, particular attention has been focused on high speed real-time detection and tracking methods [8-12]. State-of-the-art research on face detection and tracking is summarized in the following sections.

Among huge literature content on these subjects, we focused on real-time methods that require fast computational procedures to be applied online. Even though there are a lot of different algorithms for face detection, most of the publications have focused on face detection of individual images. It can be stated that 80% of the articles propose methods for detecting faces on still images from photos. Several state-of-the-art techniques for face detection and object tracking are presented in Section 2.1, and 2.2, respectively.

2.1 Face Detection Methods

Nowadays, face detection is an important research subject in the field of biometric information processing, which has an important applied value in the field high security environments. Face detection determines the presence and location of a face in an image, by distinguishing the face from all other regions present in the scene [27].

A widely used face detection method is the Histogram Oriented Gradients (HOG) developed by Corvee et al. [13] and Dalal et al. [14]. This technique detects frontal and profile pose of human faces quite successfully. A drawback of this algorithm is false detections in noisy environments.

AdaBoost is another well-known and widely used algorithm for face detection [15]. The main idea behind AdaBoost algorithm is to obtain a powerful classifier using large number of weak classifiers. This method can be used for the detection of different types of objects such as motorbikes, bicycles, people, and cars. The two main drawbacks of AdaBoost are the requirement of large databases for object and non-object samples and the detection method is not satisfactorily powerful for the similar objects.

Edge detection algorithms are also used for face detection [16]. These algorithms try to combine performance of edge detection algorithms and human visual system to detect and recognize facial objects in real images.

On the other hand, the Baseline algorithm [17] uses a feature dataset that is another form of decision tree classifiers as a numerical model. In principle, the Baseline algorithm works as follows: the algorithm uses the optimum values of a greedy algorithm based on distances. It computes the distances of centroids of the detected facial objects by comparing two frames of video sequences. Computational speed to detect and track images in video streams is not as good as its competitors.

The modified census transformation by Kublbeck et al. [18], presents a local structural feature which is used to compute census transform. Selected candidates are

classified by a linear classifier. The proposed method is satisfactory in detection rate. This method produces two kind of errors, misses a true face position and detects faces at a non-face positions.

Cascia et al. [19], proposed the idea which consist of using a feature mapped model in the texture mapped model. The head is modeled as a texture mapped cylinder. The proposed method is robust in different lighting conditions, but the outcome is not as good as in profile faces.

The edge orientation features by Froba et al. [20] uses edge orientation information to detect face candidates and each candidate is trained by Spars Network of Winnovs (SNoW). All the results are satisfactory and lead to a very fast face detector.

Neural networks are the powerful detection tools that can be used for online applications [21]. The main advantage of this technique is a detection rate between 67% and 85% for faces from images of varying size, background, quality, and an acceptable number of false detections. The algorithm can detect between faces and non-faces by using a limited training data set, but training time for online video tracking may be infeasibly large.

Other face detection algorithms based on eyes and nose detection were proposed by Zhu et al. [22] and Verma et al. [23]. In [22], a geometric function is used to detect pupils. The candidate pupils are trained by a Support Vector Machine (SVM) classifier. But the detection method is not satisfactory for closed eyes. In [23], the camera used for eyes and nose detection can be mounted to any place with different environmental conditions. The main advantage of this work is that the detection is

based on two detectors which are frontal and profile view detectors. Also, the method can detect multiple faces as well as changing scales and poses.

2.2 Face Tracking Methods

A face tracking is a research subject in the field of object tracking and computer vision. Detected faces in one frame of video are tracked and kept the faces in the next frames of video sequence, by blocking the detected face from all prior frames [30].

A stochastic approach for face tracking based on a Bayesian formulation was proposed by Sidenblath et al. [34]. This method provides a motion tracking of human figures in 3D by using monocular sequences of images. Based on the assumption that brightness is constant between frames, the method is claimed to track humans under different viewpoints and within complex backgrounds.

Jang and Choi [36] presented a structural model of Kalman filter to predict the information of motion under difficult situations such as occlusions. The structural Kalman filter consists of two types of component filters: cell Kalman filter and relation Kalman filter. The methods divide an object into a number of sub-regions and employ the motion information of sub-regions to predict motion of objects. The cell Kalman filter uses prediction of information motion associated with sub-regions, and the relation Kalman filter predicts the comparative relationship between two adjacent sub-regions. However, the method is very complicated and it is not easy to partition an object in different real-world application.

The face detection and tracking method proposed in this thesis work is a combination of the ideas presented in [27] and [30]. In [27], a new distinctive feature, called joint Haar-like feature, is presented for face detection. The authors also introduced

feature co-occurrence to capture structural similarities and help to build effective classifiers. AdaBoost is used to train the classifiers and experimental evaluations exhibited that the resulting system achieved higher classification performance than Viola and Jones' detector. In [30], an adaptive Kalman filter is used for object tracking. The authors used the dominant color of the moving object in HSI color space to detect the moving object in the consecutive video frames. The proposed method is tested under complex real-life situations where occlusions, different lightning conditions, directional and speed changes may occur. The authors claimed that their object tracking system exhibited robust performance under all these realistic scenarios.

Chapter 3

FACE DETECTION AND TRACKING ALGORITHMS IMPLEMENTED IN THE THESIS WORK

In this chapter, algorithms used within the implemented system of face detection and tracking are described. Details of the implemented system are illustrated with algorithmic steps and examples exhibiting the corresponding results.

The first section (3.1 Basic Concepts) shows a description of the face detection problem and describes intuitively which features are extracted and how information flow is carried out in this process. The second section (3.2 Face Detection) presents the mathematical background of algorithms used for face detection. In this work, the Haar classifier used to detect faces is also described. In the third section (3.3 Face Tracking), additional information was added for tracking the detected faces. Kalman filter have been chosen for tracking and predicting. The last section (3.4 Chapter Summary and Discussion) provides an overview of the chapter.

3.1 Basic Concepts

Research papers on face detection and tracking are studied and state-of-the-art algorithms were summarized in the previous chapter. Literature survey has revealed that different methods and their combinations can be applied to develop a new online video face detection and tracking system. Among the many possible approaches, the combinations of Haar classification method for face detection and Kalman filter method for tracking have been chosen. The main reason behind this selection is their

effectiveness, applicability and reliability of these methods. A functional description of the implemented face detection and tracking system is presented in Figure 1.

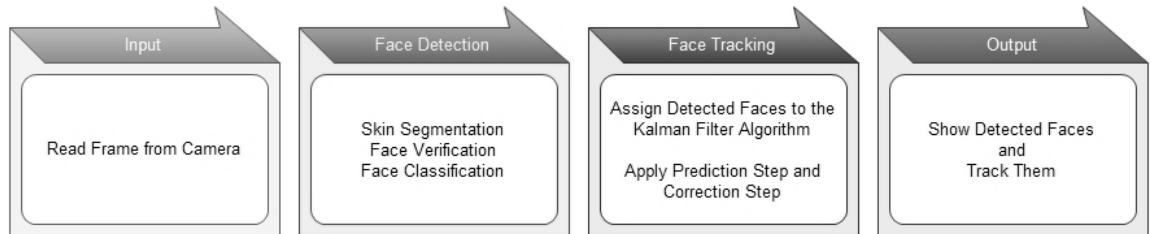


Figure 1: The Implemented Face Detection and Tracking System

3.2 Face Detection

Face detection procedure extracts facial images and locates facial images over the current video frame. A block diagram description of the face detection system used in this thesis work is shown in Figure 2.

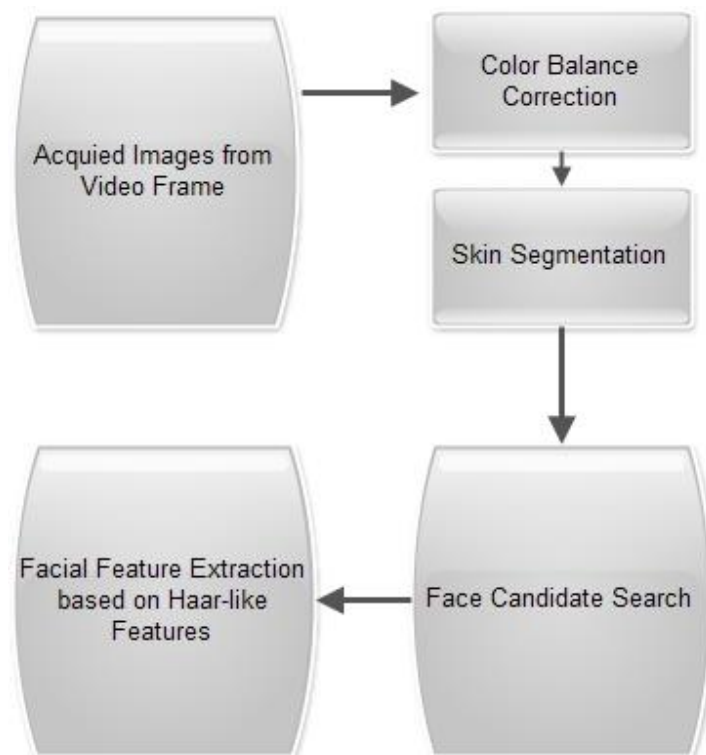


Figure 2: Block Diagram of the Face Detection Procedure

After reading a video frame, the first step of the face detection procedure is color balance correction. Color balance is needed for the global adjustment of color intensities. The main goal of the adjustment is to render specific colors correctly. This is why color balance should be applied before face segmentation [17]. For the purpose of color balance correction, the mean values (mv) of Red (R_{mv}), Green (G_{mv}), and Blue (B_{mv}) colors from $N \times M$ image are calculated together with the mean value of gray color:

$$\begin{aligned}
 R_{mv} &= \sum_{i=1}^N \sum_{j=1}^M R(i, j) / (N \times M) \\
 G_{mv} &= \sum_{i=1}^N \sum_{j=1}^M G(i, j) / (N \times M) \\
 B_{mv} &= \sum_{i=1}^N \sum_{j=1}^M B(i, j) / (N \times M) \\
 Gray_{mv} &= (R_{mv} + G_{mv} + B_{mv}) / 3
 \end{aligned} \tag{3.1}$$

Then, individual color coefficients for Red, Green, and Blue colors are obtained by dividing $Gray_{mv}$ to their mean values:

$$K_R = Gray_{mv} / R_{mv}, K_G = Gray_{mv} / G_{mv}, K_B = Gray_{mv} / B_{mv} \tag{3.2}$$

As the final step of color balance correction, a new image is generated with color components:

$$\begin{aligned}
 New(R) &= K_R * OI(R) \\
 New(G) &= K_G * OI(G) \\
 New(B) &= K_B * OI(B)
 \end{aligned} \tag{3.3}$$

where OI is original image acquired from the video sequence. An example on the implementation of color balance correction is given in Figure 3. Also, Procedure 1 shows the algorithmic procedure of color balance correction for a picture obtained from video sequence.



Figure 3: An Image Without (Left) and With (Right) Color Balance Correction

1. Given image from frame of video sequence.
2. Find the average of RGB values. (Equation 3.1)
3. Find the average of gray value. (Equation 3.2)
4. Compute color coefficients. (Equation 3.3)
5. Generate the color balanced image.

Procedure 1: Color Balance Correction Procedure [35]

In skin segmentation, RGB, HSV color spaces are examined for skin like color [23-25], [35]. RGB is sensitive to light changes but HSV is not because of its separated intensity and color changes. However, in our implementation the best results are obtained using the RGB color space. The result of skin segmentation will be a gray-scale image. Skin regions are observed brighter than the non-skin regions. The final image is applied for pointed locations of faces and results are sent to the next step. Figure 4 shows the result of skin segmentation on acquired image. The procedural steps of skin segmentation are shown in Procedure 6.



Figure 4: Color Balanced Image (Left) and Skin Segmentation Result (Right)

1. Given color balanced image x
2. For $i=1 \dots$ matrix size of red color
 - a. For $j=1 \dots$ matrix size of green color
 - a.1. Initializing R,G, and B variables from $x[i,j,Ints]$ (where Ints is intensity value of Red=1, Green=2, and Blue=3)
 - a.2. If R greater than T_R , G greater than T_G , and B greater than T_B then
 - a.2.1. If subtraction of maximum value of matrix form of R,G, and B and minimum value of matrix form of R,G, and B greater than 15 then
 - a.2.1.1. Pixel $[i,j]$ in final image is assigned to 1

Procedure 2: Procedure of Skin Segmentation [32]

In Procedure 2, T_R , T_G , and T_B are threshold intensity values for Red, Green, and Blue, respectively.

The result of skin segmentation is used by a Haar-like feature extraction method [26]. This method extracts five types of Haar-like features from the skin segmented images. These features use the alteration in contrast values between adjacent rectangular groups of pixel values [27]. The five types of Haar-like feature templates used in this thesis are shown in Figure 5. These five types of feature templates are used for the purpose of extracting lips, eyes, and nose.

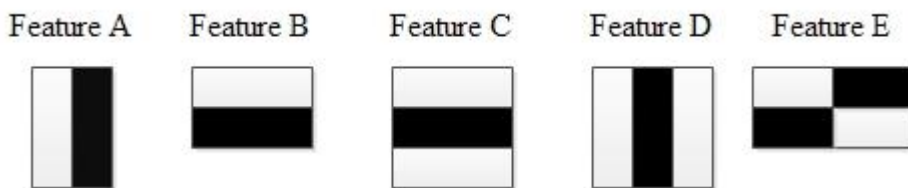


Figure 5: Features of Haar-like method [27]

The simple rectangular Haar-like features of an image are calculated using an intermediate representation of the image, known as the integral image [28]. The integral image is an array containing the sums of the pixels' intensity values located directly to the left and above the pixel at location (x, y) . The integral image ($AI[x,y]$)

of an image $A[x,y]$ is computed using equation 3.4, that is graphically illustrated in Figure 6.

$$AI[x, y] = \sum_{x' \leq x, y' \leq y} A(x', y') \quad (3.4)$$

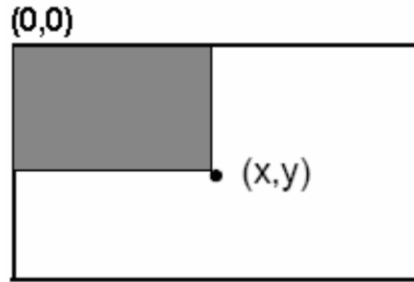


Figure 6: Computation of $AI[x,y]$ Values [28]

The integral image, also known as summed area table can be computed easily in a single pass over the image using the fact that:

$$AI(x, y) = A(x, y) + AI(x-1, y) + AI(x, y-1) - AI(x-1, y-1) \quad (3.5)$$

Once the integral image has been computed, area of any rectangular region (P, Q, R, S) over the image can be computed in constant time as follows: Let $P(x_0, y_1)$, $Q(x_1, y_1)$, $R(x_1, y_0)$, and $S(x_0, y_0)$ (see Figure 7), then

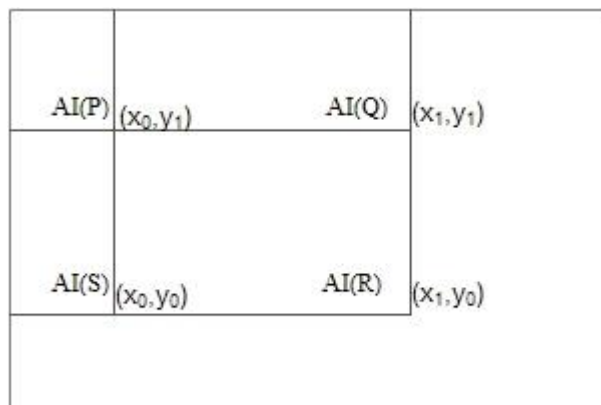


Figure 7: Computation of the Area Table [37]

$$\sum_{\substack{x_0 < x < x_1 \\ y_0 < y < y_1}} A(x, y) = AI(R) + AI(P) - AI(Q) - AI(S) \quad (3.6)$$

After calculating the summed area table (Figure 7), the next step is the extraction of Haar-like features. As seen on Figure 7, Haar features are combination of two or four rectangles. A Haar feature classifier uses the summed area table to calculate the value of each Haar-like feature. A stage comparator sums all the Haar feature results in each stage and compares it with a stage threshold. The threshold is also a constant obtained from the weak classifier. A data set which is used for training stage is presented by Viola and Jones which is containing 5000 faces and 10000 non-face sub-windows [2]. The data set is used 28 stages in total. The window size used to extract Haar-like features in size of $k \times k$ while the image size is $M \times N$.

Each Haar feature is associated with the corresponding image sub-window over the integral image. Each sub-window is applied over the integral image horizontally and vertically when end of a row is reached. In this respect, the number of Haar-like features extracted can be up to five types of Haar-like features that are applied in order (Figure 8).

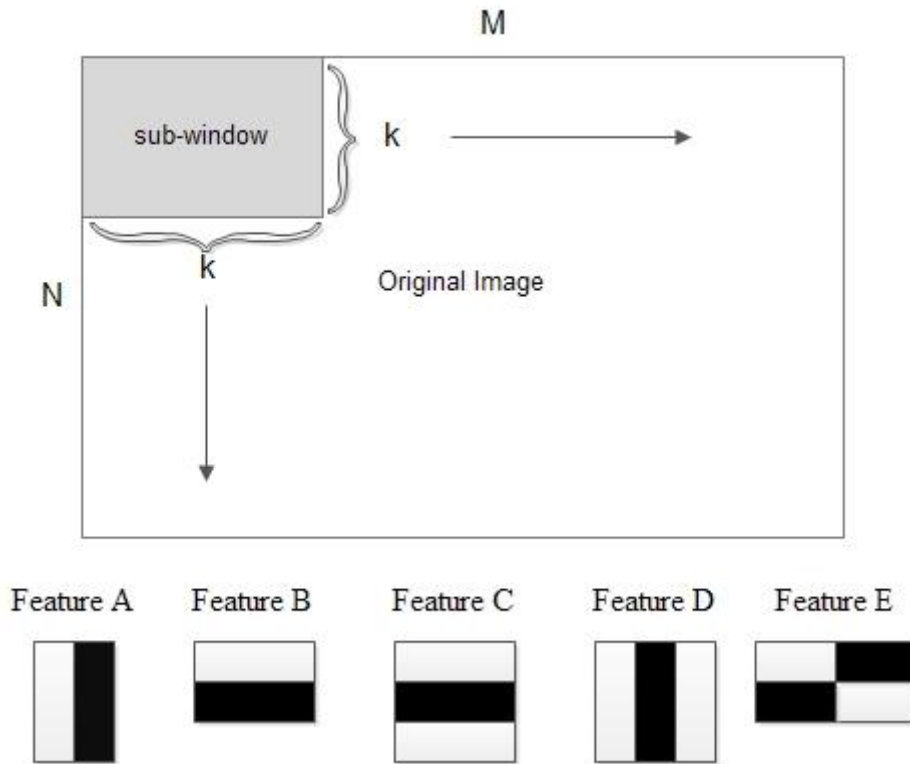


Figure 8: The Five Types of Haar-Like Feature are Applied in an Image Based on a $k \times k$ Sub-Window

where k is the size of sub-window, $M \times N$ is the size of the image. In this thesis work, we used a fixed value of k , as $k=24$, a sub-window size, and 640×480 as original image size.

Viola and Jones face detection method eliminates non-face quickly using a cascade of stages. Each face candidate must be passed through all of stages. This process is shown in Figure 9.

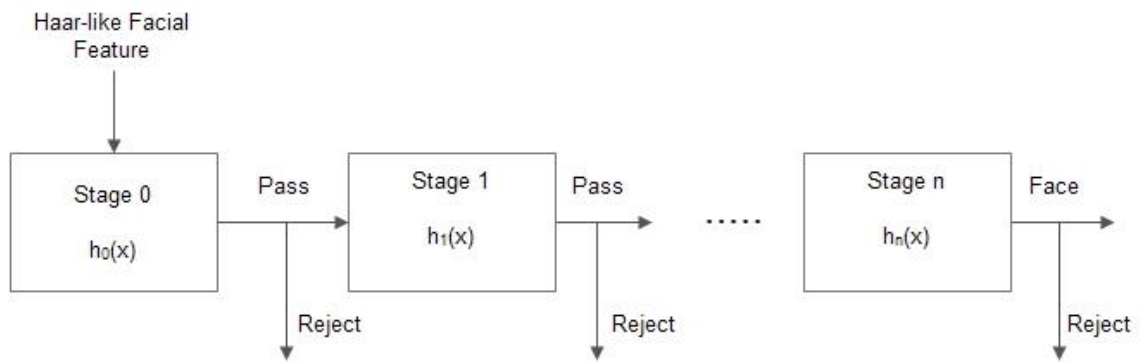


Figure 9: Cascade of stages [2].

In Figure 9, $h_r(x)$ is a weak classifier in each stage, individually. A Haar classifier consists of two or three rectangles and their weight values, feature threshold value, and left and right values. Each rectangle presents four points by coordinates (x,y) . In this manner, an image window buffer is essential to store pixel values moving from the image line. The integral image value can be computed using these points from the integral image window. Facial features, which are used in this thesis work, are two eyes, lips, and nose. If one of the features may be found in the candidate, then it will be considered as face. Weighted summation of Haar-like features is compared with the feature threshold, if the result is larger than the stage threshold it goes to the next stage, and also, if the candidate could pass all stages, it is known as face. Figure 10 shows the whole procedure of weak classifier. Figures 11 and 12 show features of a sample face for two eyes, and features of a sample face for lips, respectively.

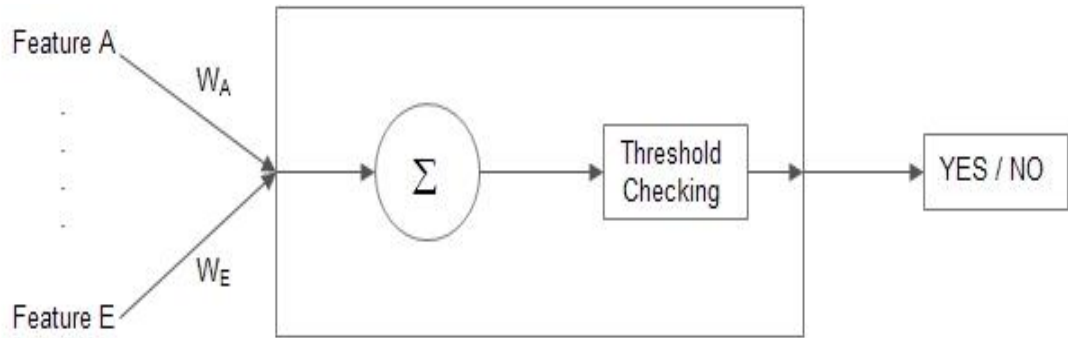


Figure 10: Whole Procedure of Weak Classifier by Joining Five Types of Haar Features

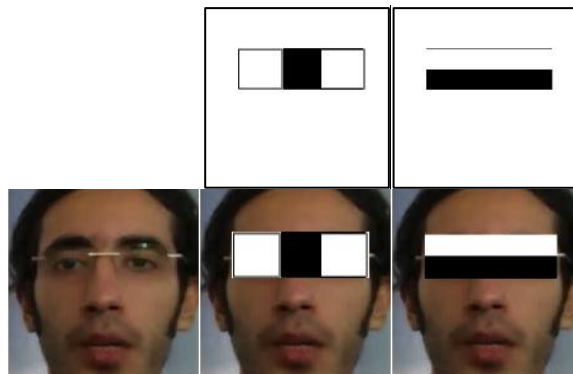


Figure 11: The Two Features of Eyes are Shown in Top Row and then Detected Feature Show in Bottom Row. The First Feature Compares the Intensity Levels of Two Eyes to the Intensity Level Across the Bridge of the Nose. The Second Feature Calculates the Difference of Intensity between regions of two eyes across the cheeks.

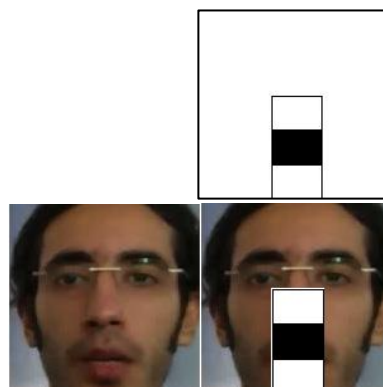


Figure 12: The Feature of Lips is Shown in Top Row and then Detected Feature Shows in Bottom Row.

In this work, the weak classifier method classifies candidate faces based on the value of simple feature. In this regard, feature can act to encode knowledge that is difficult to learn in training data. Thus, the simple feature is much faster in face detection in video sequence. A weak classifier algorithm can be having good performance to select the single feature to best classify faces and non-faces. The weak classifier is determined sub-images between faces and non-faces. In this case, the goal is to eliminate a substantial amount of sub-images that does not contain the face, and selected faces are extracted from each frame of video sequence.

3.3 Face Tracking

The faces which are detected in the face detection system should be used to track the faces. Face tracking system is composed of preprocessing facial image, system state model, prediction, measurement, and correction. The structure of tracking algorithm is given in Figure 13. Descriptions of Kalman filter elements are given in the following sections.

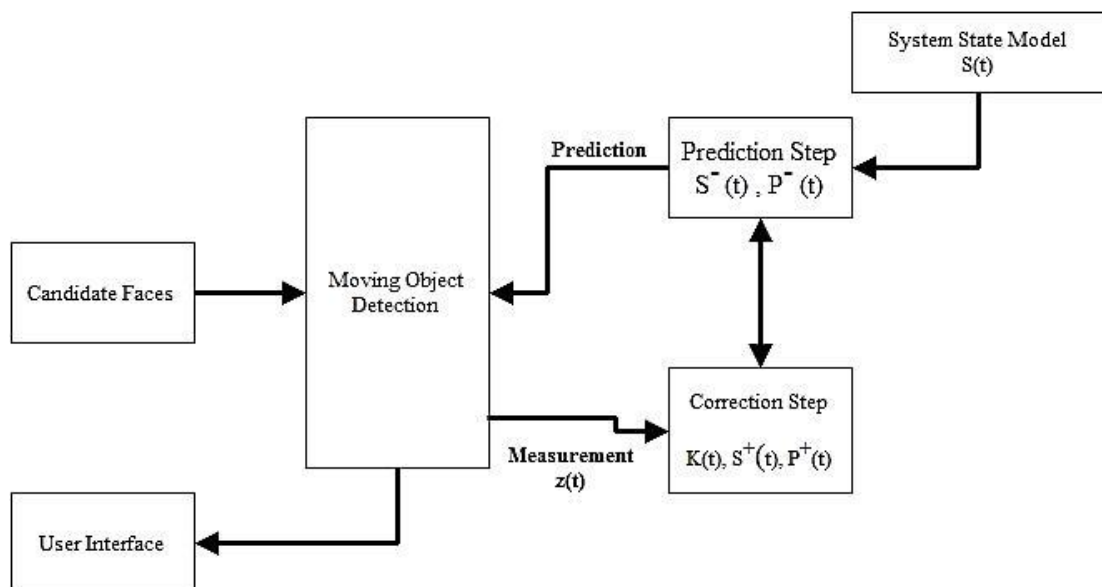


Figure 13: The Whole Structure of Kalman Filter [30]

The Kalman filter has two basic structures which are described by system rate model and measurement model [30]. One can employ the adaptive Kalman filtering to track the objects in video streams. The system state model in Kalman filtering is constructed by the motion model and it is used in prediction step. The system state model and measurement model are defined as:

$$s(t) = O(t-1)s(t-1) + w(t) \quad (3.7)$$

$$z(t) = H(t)s(t) + v(t) \quad (3.8)$$

where $O(t-1)$ and $H(t)$ are the state change matrix and measurement matrix, respectively. The $w(t)$ and $v(t)$ are white Gaussian noise with zero mean. Also, the calculation of covariance matrices of $w(t)$ and $v(t)$ is needed which is defined in Equation 3.9 as:

$$E\{w(k)w^T(l)\} = Q\delta_{kl} \quad , \quad E\{v(k)v^T(l)\} = R\delta_{kl} \quad (3.9)$$

where δ_{kl} is used for the Kronecker delta function. Furthermore, the state vector $s(t)$ of the current time t is predicted from previous estimate and the new measurement $z(t)$. Also, the parameters of measurement model in Equation 3.10 are defined as:

$$s(t) = \begin{bmatrix} d(t) \\ d(t-1) \end{bmatrix}, H(t) = [1 \ 0], \text{ and } z(t) = [d(t)] \quad (3.10)$$

The prediction step of Kalman filter is responsible for projecting forward the current state, obtaining a previous predict of the state $s^-(t)$. The task of correction step is for the feedback. It incorporates an actual measurement into a previous prediction to obtain an improved a posterior predict $s^+(t)$. The $s^+(t)$ is defined as:

$$s^+(t) = s^-(t) + K(t)(z(t) - H(t)s^-(t)) \quad (3.11)$$

where $K(t)$ is the weighting and described as:

$$K(t) = P(t)^-H(t)^T(H(t)P(t)^-H(t)^T + R(t))^{-1} = \frac{P(t)^-H(t)^T}{H(t)P(t)^-H(t)^T + R(t)} \quad (3.12)$$

where $P(t)^-$ is the previously predict error of covariance, it is defined as:

$$P(t)^- = E[e^-(t)e^-(t)^T] \quad (3.13)$$

where $e^-(t) = s(t) - s^-(t)$ is the priori predicting error.

The prediction step and correction step are calculated recursively as follows:

Prediction step:

$$s^-(t) = O(t-1)s^+(t-1) \quad (3.14)$$

$$p^-(t) = O(t-1)P^+(t-1)O(t-1)^T + Q(t-1) \quad (3.15)$$

Correction step:

$$K(t) = P^-(t)H(t)^T (H(t)P^-(t)H(t)^T + R(t))^{-1} \quad (3.16)$$

$$s^+(t) = s^-(t)K(t)^T (z(t) - H(t)s^-(t)) \quad (3.17)$$

$$P^+(t) = (I - K(t)H(t))P^-(t) \quad (3.18)$$

The prediction and correction steps are repeated. At Equation 3.16, the measurement error $R(t)$ and Kalman gain $K(t)$ are in inverse ratio. When $R(t)$ is small and $k(t)$ is bigger, the measurement is more trustable compared to the predicted result. On the other hand, when prior predict error $P^-(t)$ is very close to zero, the effect of gain $K(t)$ become insignificant. Furthermore, the system state model is employed in the prediction step. Since the duration of each frame is very short, the distance of moving face should be replaced as system parameter by location of detected face.

The moving face distance is defined as:

$$d(t) = d(t-1) + (d(t-1) - d(t-2)) \quad (3.19)$$

where $d(t-1)$ and $d(t-2)$ are moving face distances in frame $t-1$ and $t-2$, respectively.

Also, the center of the matched face area is found. It is calculated by bonding box values which are illustrated as the position of detected faces. The center is defined as:

$$(cx, cy) = \left(\frac{x + width}{2}, \frac{y + height}{2} \right) \quad (3.20)$$

For the next frame, searching range is employed by extending pixels of detected area.

This is extended with corner position at left upper ($search_left$, $search_up$) and right

lower corner ($search_right$, $search_down$). They are defined as:

$$\begin{aligned}
 search_left &= (pre_left - (d(t-1) - d(t-2))) \\
 search_up &= (pre_up - (d(t-1) - d(t-2))) \\
 search_right &= (pre_right + (d(t-1) - d(t-2))) \\
 search_down &= (pre_down + (d(t-1) - d(t-2)))
 \end{aligned} \tag{3.21}$$

where prefix “pre” presents the previous frame. In this regard, searching range considers the face’s moving speed and is located at the predicted position of Kalman filter. All in all, Figure 16 presents the algorithmic procedure of Kalman filter which used in this thesis work.

1. Given state model $x_i = (x(t), y(t), v_x(t), v_y(t))^T$
2. Given measurement model $z_i = H(t) S(t) + V(t)$
3. Initializing Error, Gaussian Noise, weighting, and estimate:
 $X_0 = E[x_0]$
 $P_0 = E[x_0 - \bar{x}_0]$
 $K_0 = P_0 H_0^T (H_0 P_0 H_0^T + R_0)^{-1}$
 $S_0^- = K_0 (z_0 - H_0)$
4. Prediction step:
 $P_i^- = x_{i-1} P_{i-1}$
 $K_i = P_i^- H_i^T (H_i P_i^- H_i^T + R_i)^{-1}$
5. Correction step:
 $S_i^+ = S_i^- - K_i (z_i - H_i S_i^-)$
 $P_i^+ = K_i P_i^- H_i$
 $H_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$
6. Calculate distance of candidate face
 $d(t) = d(t-1) + (d(t-1) - d(t-2))$
7. Update system state mode by equation 3.7
8. Repeat step 4- step 7

Procedure 3: Procedure of Kalman Filter [30]

The frame difference is used to segment moving faces. Then, the dominant color is extracted from segmented moving faces. In the tracking method of Kalman filter, the system state is applied to prediction step. Furthermore, the measurement of system is

provided by moving face detection using Haar-like feature method. The measurement errors will make Kalman filter system to trust prediction more strongly.

3.4 Chapter Summary and Discussion

Face detection and tracking system have four main steps, which are input, detection, tracking and output. Input performs image achievement, which converts captured image from video sequence to digital image data. Detection part is composed of color balance correction, skin segmentation, facial feature extraction, and face image selection. Color balance correction is an important step to eradicate color changes of achieved image due to brightness conditions change. Skin color like state segmentation to reduce search time for possibility face state since only segmented states are measured as state may contain face. Performance of skin segmentation is improved with color balance correction and facial feature extraction is improved with Haar-like filter [27,30].

Face images are extracted based on facial features position, and image preprocessing operation is performed to eliminate brightness changes and prepared to be input to classifier. Classifier is a key point of tracking part. Kalman filter is good at object tracking problem. The proposed tracking algorithm will perform fast and truthful for tracking faces.

Chapter 4

IMPLEMENTATION OF FACE DETECTION AND TRACKING SYSTEM

The system on face detection and tracking was studied and state-of-the-art was appraised and summarized in the previous chapters. Literature survey has revealed that different methods and combination of these methods can be applied in advance of a new online video face detection and tracking system. Among the many possible approaches, the combinations of Haar-like features used by a weak classifier for face detection and Kalman filter method for face tracking have been chosen. The main reason of this choice is their effectiveness, applicability and reliability.

To make automatic face detection and tracking system for an online video, we require extracting and tracking faces in an image from a video sequence. For making such type of system, we have included three distinct phases. First, faces are detected and classified in each frame of a video sequence by using weak classifier. Candidate faces are posted to Kalman filter module for prediction and tracking. Then, the candidate faces are updated frame to frame. Finally, the result is shown in rectangular boxes as output.

A flowchart depicting the main stage of the system is shown in Figure 14 and more explanation will be given below.

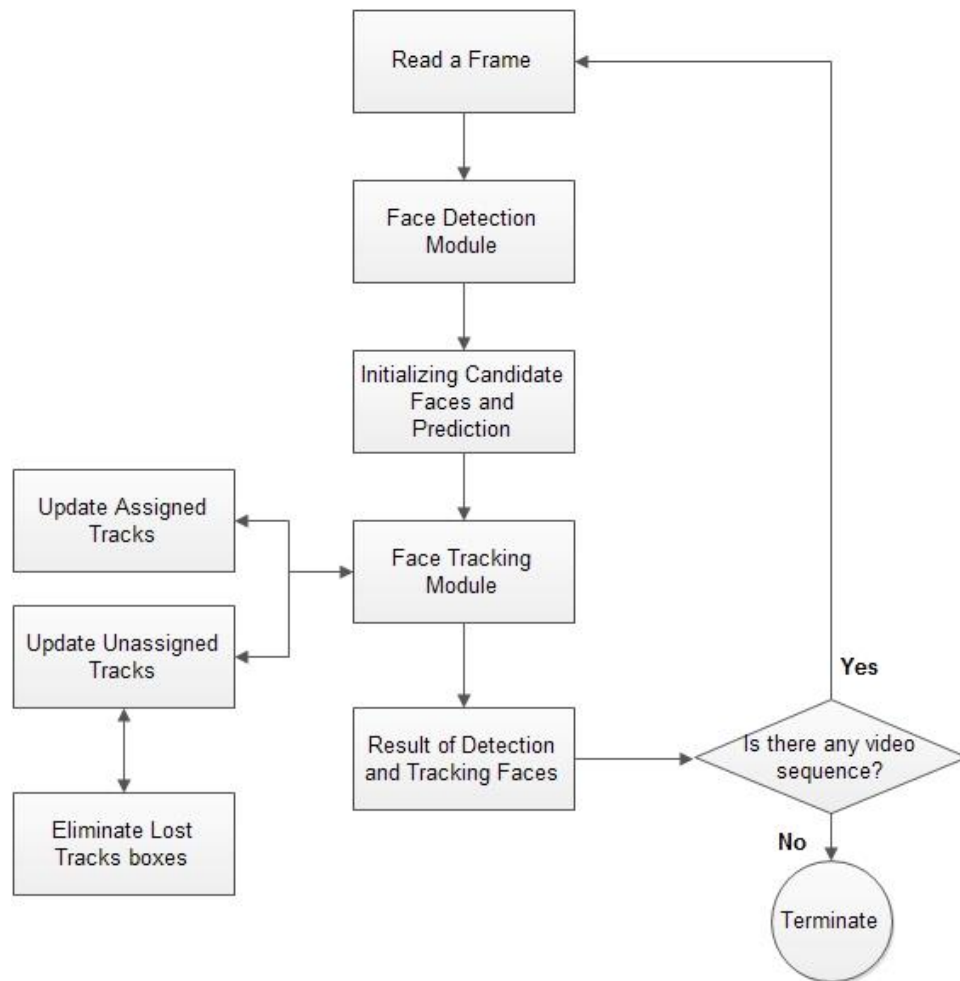


Figure 14: Flowchart of the Main Stages of the System

As the main part of this work, we discuss reading a frame from video sequence. “Read frame” is the action needed to ensure that the system commence its issue with a correct clarification of the current frame of video sequence. “Acquiring image” task is performed in this part. An image is acquired from each frame of video and it is converted to digital data for performing image processing subtractions. Program 1 shows the implementation of capturing frame from video sequence in MATLAB.

```
reader = imaq.VideoDevice('winvideo', 1, 'YUY2_640x480','ROI', [1 1 640 480],...
'ReturnedColorSpace', 'rgb');

obj.videoPlayer = vision.VideoPlayer('Position', [20, 400, 700, 400]);

frame = obj.reader.step();
```

Program 1: Capturing Frame from Video Sequence

A video sequence is defined in 640×480 pixels and color space is defined as RGB.

The capturing frame is sent to face detection part.

4.1 Face Detection

In order to detect the faces, extraction is performed in the RGB color space. In this regard, RGB color space is more suitable for color extraction. To solve face detection problem, as explained in the previous chapter, we use the Haar-like features by engaging the weak classifier method. The Haar-like features consist of eyes, nose, and lips. These features are able to construct a more powerful classification.

Furthermore, color balance and skin segmentation are applied for commencing of the face detection process. Lighting conditions are always changing, or different lighting is occurred in indoor/ outdoor environments. Program 2 and 3 show the implementation of color balance and skin segmentation, respectively.

```
imageData= getsnapshot(frame);
figure, imshow(imageData);
avg_rgb = mean(mean(imageData));

factors = max(mean(avg_rgb), 128)./avg_rgb;

imageData(:,:,1) = uint8(imageData(:,:,1)*factors(1));
imageData(:,:,2) = uint8(imageData(:,:,2)*factors(2));
imageData(:,:,3) = uint8(imageData(:,:,3)*factors(3));
```

Program 2 : Implementation of Color Balance

```

if(size(imageData, 3) > 1)
    for i = 1:size(imageData,1)
        for j = 1:size(imageData,2)
            R = imageData (i,j,1);
            G = imageData (i,j,2);
            B = imageData (i,j,3);
            if(R > 80 && G > 50 && B > 30)
                v = [R,G,B];
                if((max(v) - min(v))> 15)
                    if(abs(R-G) > 15 && R > G && R > B)
                        imageData(i,j) = 1;
                    end
                end
            end
        end
    end
end
end
end
end
end
end

```

Program 3: Implementation of Skin Segmentation

The computation cost of weak classifiers is very low and they are reliable in noisy environments. Regarding to the method by Viola and Jones, each weak classifier includes multiple features. The combinations of features are applied in each round of the process. For improving classification of features, features are added one by one into the process of the method. After training, the candidate faces are appeared in a data set. The result is illustrated as position of each candidate. Thus, the candidate faces are ready to show and bounding box is added. Fortunately, MATLAB provides some essential Toolbox for detecting faces by using Haar-like features and joining weak classifier. Program 4 shows the implementation of face detection part.

```

ViCascade = vision.CascadeObjectDetector;
ViCascade.MinSize=[20 20];

bboxes = ViCascade.step(frame); %imageData

```

Program 4: Implementation of Face Detection Part

The “*CascadeObjectDetector*” provides the Viola and Jones face detection algorithms. The detector can be detecting human faces based on special features which are applied in the detector such as eyes, noses, and mouth. Also, by using this detector each feature can be detected individually. The detector consists of some models which are frontal face, upper body, eye pair, single eye, profile face, mouth, and nose. Also, for each of these models training sets are different. It can be chosen as defined in the detector by selecting the model of training, automatically.

The candidate faces is stored in the *bboxes* data set. The data set will be used in face tracking system.

4.2 Face Tracking

After detecting faces, with the locations stored in the bounding box data set, Kalman filter is applied for each of the candidate faces. In this regard, the location of each candidate faces is initialized as input in the Kalman filter chain. Also, estimation error is initialized for the tracking moving faces in the noisy environments. The estimation error definition was shown in equation 3.4. As the next stage, the system state model is built by motion model and it is applied in prediction step. It is used to predict parameters of Kalman filter automatically. In this respect, if the measurement error and occlusion ratio are in the same rate, the occlusion rate is employed for predicting parameters in the correction step. According to the equation 3.16, the occlusion rate must be less than threshold value to employ in the measurement error $R(t)$ and the prediction error $Q(t-1)$ of equation 3.15. The two errors of measurement and prediction make the Kalman filter tracking system more reliable.

In this regard, MATLAB provides a Toolbox for implementing Kalman filter. Program 5 shows the configuration and implementation of Kalman filter. The motion

model of this algorithm in Toolbox, which is explained in chapter 3, provides two different types which are “ConstantVelocity” and “ConstantAcceleration”. These models are employed when either velocity is an important item for tracking objects or acceleration. Although the velocity of moving faces is an important factor in this thesis work, the motion model based on velocity is chosen. Then, the data set of “*bbox*” is initialized as start position of tracking. Next, the estimate error is initialized. It is specified as a two or three element vector. Estimate error is playing an important role to estimate the location of the tracked object. Then measurement noise is initialized. It is used by the algorithm for the tolerance of deviation from velocity of object. Also, it may be helpful for removing noises from the detected faces. This value is used as constant and affects the long term performance of the algorithm. Finally, motion noise is initialized which is specified as two or three element vector. It can also be useful for decreasing noise of detected faces. Increasing the value of motion noise changes the state of algorithm to appropriately track the detected faces.

```
kalmanFilter = configureKalmanFilter('ConstantAcceleration', bbox, [50, 50,50],[250,100,100], 500);
```

Program 5: Configuration of Kalman Filter in MATLAB

After assigning the tracking part to the detected faces, the prediction and correction methods are employed. Program 6 shows the implementation of these methods.

```

bboxes = bboxes(unassignedDetections, :);

for i = 1:size(bboxes, 1)

    bbox = bboxes(i, :);
        newTrack = struct(...
            'id', nextId, ...
            'bbox', bbox, ...
            'kalmanFilter', kalmanFilter);
        tracks(end + 1) = newTrack;
        nextId = nextId + 1;

end

for i = 1:numberAssignedTracks
    trackIdx = assignmentsDetected(i, 1);
    detectionIdx = assignmentsDetected(i, 2);
    bbox = bboxes(detectionIdx, :);

    if ~isempty(bbox)
        bbox=correct(tracks(trackIdx).kalmanFilter, bbox);
    else
        bbox= predict(tracks(trackIdx).kalmanFilter);
    end

    tracks(trackIdx).bbox = bbox;
end

```

Program 6: Implementation of Prediction and Correction of Kalman Filter

As the result of this implementation, Figure 15 shows that the moving persons can successfully block and track frame to frame.



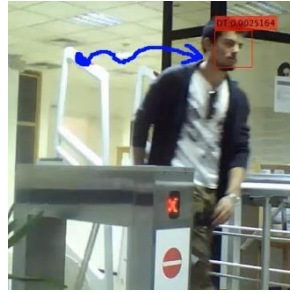
(a) The 166th frame



(b) The 170th frame



(c) The 213th frame



(d) The 219th frame



(e) The 199th frame



(f) The 323th frame

Figure 15: Blocking and Tracking of Candidate Faces

Online tracking faces are shown with respect to output of face detection and tracking system by covering boxes for each candidate face. Program 7 shows the implementation of inserting bounding box in detected and tracked faces. Also, detection of a new face candidate and tracking of candidate faces are updated for each frame of video sequence, because some candidates are probably eliminated or inserted.

```

if ~isempty(tracks)

    bboxes = cat(1, tracks.bbox);
    labels = strcat(labels, isPredicted);

    detectTimes=[reliableTracks(:).DTime];
    timeLables=cellstr(num2str(detectTimes));
    timeLables=strcat('DT:',timeLables);

    frame = insertObjectAnnotation(frame, 'rectangle',bboxes,...
        timeLables,'Color','red','TextColor','black');
    AvgDTime=mean(detectTimes);
    AvgDTime=cellstr(num2str(AvgDTime));
    AvgDTime=strcat('Mean Value along Detection Times: ',AvgDTime);

    frame= insertText(frame,[30 475],AvgDTime,'AnchorPoint','LeftBottom',...
        'BoxOpacity',0.6,'BoxColor','Green');

End

videoPlayer.step(frame);

```

Program 7: Implementation of User Interface for Tracking of Detected Faces

Chapter 5

EXPERIMENTAL RESULTS

A complete hardware and software system is designed and implemented in Department of Computer Engineering at Eastern Mediterranean University. The developed system has been tested for many live-recorded frames from camera and results are satisfactory for such an innovative work in the department. Improvements are required for better performance. System description and possible improvements are discussed in this chapter.

5.1 System Hardware

System has two main hardware parts. They are computer and camera. Computer is the main part of the system which processes the acquired image, image analysis, detection and tracking the faces. The computer used in the test is typical PC with the following configuration:

- Intel(R) Core (TM) 2 Duo CPU
- 3 Gb RAM
- ATI Radeon X1650 Series External Graphic Card

PIRANHA 5 Mb pixels is used in Face detection and tracking (Figure 16). Camera has high quality with Pan and zoom operations. Camera resolution is 640x480 pixels, and it has 10x digital zoom, USB 2.0 communication.



Figure 16: PIRANHA - Five Megapixel

5.2 System Software

Algorithm of the system is applied in MATLAB R2013a software. MATLAB is a production of MathWorks Co. can be performed many algorithms such as data analysis, numeric computation, signal processing, image processing, mathematical computation, etc. MATLAB supplies facility environment for implementing scientific works by reposing toolboxes for any goals which is generation of algorithms are more powerfully. Image Acquisition Toolbox, and Image Processing Toolbox are used while generation of the algorithm of Face detection and tracking system.

Image Acquisition Toolbox provides image achieving from camera system that MATLAB supports. This Toolbox will bridge between output incoming data and MATLAB environment.

Image Processing Toolbox enables many source algorithms, graphical tools, analysis, etc. Filters, transforms, enhancements, etc. are ready to apply functions which simplify to code generation. This toolbox is used in face detection and tracking system.

5.3 Face Detection

First of all, implementation of system is performed on face detection in acquiring image from each frame, and skin like region segmentation is a first implementation of face detection. In that case, many methods have been tried to select which segmentation algorithm works best on acquired images. Based on RGB [22], HSV [25], and HSV&YCbCr [31] are tested on acquired images from frames and the best result are taken from RGB color space.

After segmentation is completed, candidate search is achieved as described in the previous chapter. Candidate faces are followed by facial feature extraction and face verification of candidate.

Facial feature extraction is one of the important parts of face detection section. Detection of exact position of facial features, such as eye and lip positions, for face detection is difficult. Also, it is difficult to determine position of slanted and rotated faces. For the next step, Haar-like classifier is ready to use for classifying face candidates and it can detect and classify faces at a rate of less than half a second. It is important to note that this classifier is more useful for tracking faces. In the online domain tracking, each detected face is updated and is used for tracking. In the next section, we will discuss the details of tracking system. Some experiments are performed and results are given in Figure 17.

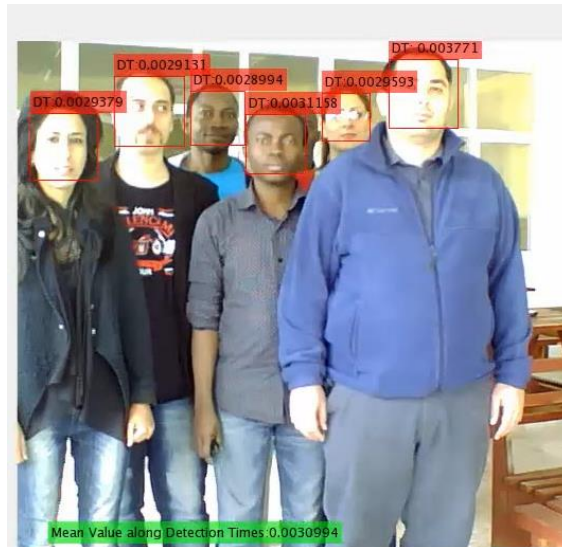


Figure 17: Candidate Faces by Mean Value of Detection Time

5.4 Face Tracking

With the addition of face detection approach, as described in the previous chapter, faces are found and each face is cropped. The candidate faces are prepared for tracking. Sample images of 30 people are given in Figure 18.

We used the Kalman filter method for moving faces in this thesis work. The video frame images consist of RGB image while resolution of image is 640×480 pixels. Some kinds of experimental videos are used to evaluate the robust ability of the proposed method, including the moving face vanishing totally or partially due to obstruction by other ones, the faces which are moving fast, changing the position and direction of the moving faces, and changing the rate of moving faces suddenly.



Figure 18: The Result of Implemented Method for Detection and Tracking for more than 30 People

In the following experimental results, we illustrated blocking in tracking faces in Figure 15. In Figure 15(a-b) and (c-d) we test the case blocking of the tracking. The duration of blocking is within 6 frames. Also, in Figure 15(e-f), the duration of blocking of tracking is from 199th frame to 323th frame. Blocking the face for a long time is very difficult. But our experimental results show that the implemented method can detect and track the moving face successfully when it suddenly disappears or appears as a new one.

Furthermore, Table 1 shows the experimental result with the efficiency on the detection rate, tracking rate, and missed faces. From these results, our proposed method can strongly detect and track faces with high speed of computational cost in different regions. The average detection time of each face is less than 0.1 second.

Table 1: Experimental Result of Face Detection and Tracking

Number of Faces	Correct detection	Missed detection	Correct tracking	Missed tracking
15	93.33%	1	80%	3
25	96	1	92%	2

36	88%	4	97%	1
----	-----	---	-----	---

Finally, Table 2 lists the processing time of a live video sequences using the proposed tracking method. Therefore, the results can be applied to real time systems.

Table 2: Blocking Time of Tracking the Candidate Faces

Video Sequence	Frame number	Average tracking time for each face (second)
Figure 14	79	28
Figure 17	187	53
Figure 18	528	198

Chapter 6

DISCUSSION AND CONCLUSION

6.1 Discussion

This thesis study focuses on implementing a face detection and tracking system. System is composed of acquired images form online video stream, face detection part, and tracking part.

Image acquisition is required by the system to capture frame from video camera device, and prepared it to be handled in the algorithm. Video camera device is PIRANHA while has 5 Mb pixels. Therefore, live captured images have different brilliance, background, color balance different, and position and size of human face. Acquired image is performed by MATLAB software with Image Acquisition Toolbox.

Haar classifier method is performed for face detection part by Haar-Like facial features with combination of skin color balance. Methods are combined due to decreasing computational time and increasing accuracy of detection part. Before skin color segmentation, color balance correction is executed to overcome color problem while achieving image from frames. Also, the method was useful for black race of humans in bright environments.

On the other hand, RGB skin color segmentation used in the algorithm works well at any environment tested (Indoor/Outdoor condition) in our experimental studies. Segmentation is performed on recorded images after color balance. Face candidates are selected from segments and facial feature extraction is executed to verify face candidate and extract face images. Haar-like method is performed to show facial components clearly.

After extraction of faces, face detection part is complete and face images is ready to be classified. Classification is performed by Haar classifier learning algorithm. The algorithm is good for training and classification problem. Algorithm is developed on MATLAB software and it is capable of detecting multiple faces in recorded images from each frame of video.

6.2 Conclusion

Face detection and tracking system is part of image processing applications and its importance as a research area is increasing recently. Implementations of system are video surveillance and similar security activities.

The main goal of the thesis is to implement a face detection and tracking system in Department of Computer Engineering. The goal is reached by face detection and tracking methods. Facial feature face detection methods are used to find position and extract faces in achieved images form frames of video. Implanted methods are color balance and facial features. Also, Kalman filter is used for face tracking.

A Weak classifier is employed to classify and solve face detection problem since face tracking is required. Classification results are satisfactory. Also, classification is satisfactory when extracted faces are small oriented, smiled, closed eye, wore

sunglass, and wore hat. Our proposed algorithm is capable of detecting multiple faces, and leads to acceptable performance in terms of obtained results.

Chapter 7

FUTURE WORKS

Face detection and tracking system is designed, implemented and tested. Test results show that the system has acceptable performance. On the other hand, system has some future works for improvements and implementation.

The first future work can be applied on face detection part to improve detection of half-facial features. Half-faces are challenging issues on face image processing research.

Then, face recognition part can be added. This future goal is to achieve online face recognition in each frame of video sequence. New technology of HMAX object recognition can be helpful for this aim. In the online video stream, speed of detection and tracking is the most important. Because in security activities, getting the useful and best result in less time is vital.

REFERENCES

- [1] Zheng, Wenlong, and Suchendra M. Bhandarkar. "Face detection and tracking using a Boosted Adaptive Particle Filter." *Journal of Visual Communication and Image Representation* 20.1 (2009): 9-27.
- [2] Viola, Paul, & Michael J. Jones. "Robust real-time face detection." *International journal of computer vision* 57.2 (2004): 137-154.
- [3] Rowley, Henry A., Shumeet Baluja, and Takeo Kanade. "Neural network-based face detection." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20.1 (1998): 23-38.
- [4] Lawrence, Steve, C. Lee Giles, Ah Chung Tsoi, & Andrew D. Back. "Face recognition: A convolutional neural-network approach." *Neural Networks, IEEE Transactions on* 8, no. 1 (1997): 98-113.
- [5] Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., & Zhang, J. (2009). Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2), 319-336.
- [6] Bradski, Gary R. "Real time face and object tracking as a component of a perceptual user interface." *Applications of Computer Vision, 1998. WACV'98. Proceedings., Fourth IEEE Workshop on*. IEEE, 1998.

- [7] Lee, Lae-Kyoung, Su-Yong An, & Se-Young Oh. "Efficient Face Detection and Tracking with extended camshift and haar-like features." *Mechatronics and Automation (ICMA), 2011 International Conference on*. IEEE, 2011.
- [8] Osuna, Edgar, Robert Freund, & Federico Girosi. "Support vector machines: Training and applications." (1997).
- [9] Manohar, V., Soundararajan, P., Raju, H., Goldgof, D., Kasturi, R., & Garofolo, J. (2006). Performance evaluation of object detection and tracking in video. In *Computer Vision—ACCV 2006* (pp. 151-161). Springer Berlin Heidelberg.
- [10] Schneiderman, Henry, & Takeo Kanade. "Probabilistic modeling of local appearance and spatial relationships for object recognition." *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*. IEEE, 1998.
- [11] Viola, Paul, & Michael Jones. "Rapid object detection using a boosted cascade of simple features." *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE, 2001.
- [12] Rowley, Henry A., Shumeet Baluja, & Takeo Kanade. "Neural network-based face detection." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20.1 (1998): 23-38.
- [13] Corvee, Etienne, & François Bremond. "Combining face detection and people tracking in video sequences." (2009): 43-43.

- [14] Dalal, Navneet, & Bill Triggs. "Histograms of oriented gradients for human detection." *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE, 2005.
- [15] Laptev, Ivan. "Improvements of Object Detection Using Boosted Histograms." *BMVC*. Vol. 6. 2006.
- [16] Heath, M. D., Sarkar, S., Sanocki, T., & Bowyer, K. W. (1997). A robust visual method for assessing the relative performance of edge-detection algorithms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(12), 1338-1359.
- [17] Manohar, V., Soundararajan, P., Korzhova, V., Boonstra, M., Goldgof, D., & Kasturi, R. (2007, October). A baseline algorithm for face detection and tracking in video. In *Optics/Photonics in Security and Defence* (pp. 674109-674109). International Society for Optics and Photonics.
- [18] Küblbeck, Christian, & Andreas Ernst. "Face detection and tracking in video sequences using the modifiedcensus transformation." *Image and Vision Computing* 24.6 (2006): 564-572.
- [19] La Cascia, Marco, Stan Sclaroff, & Vassilis Athitsos. "Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22.4 (2000): 322-336.

- [20] Froba, Bernhard, & Christian Kublbeck. "Robust face detection at video frame rate based on edge orientation features." *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*. IEEE, 2002.
- [21] Curran, K., X. Li, & N. McCaughley. "Neural network face detection." *The Imaging Science Journal* 53.2 (2005): 105-115.
- [22] Zhu, Z., Ji, Q., Fujimura, K., & Lee, K. (2002). Combining Kalman filtering and mean shift for real time eye tracking under active IR illumination. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (Vol. 4, pp. 318-321). IEEE.
- [23] Verma, Ragini Choudhury, Cordelia Schmid, & Krystian Mikolajczyk. "Face detection and tracking in a video by propagating detection probabilities." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25.10 (2003): 1215-1228.
- [24] Qiang-rong, Jiang, & Li Hua-lan. "Robust human face detection in complicated color images." *Information Management and Engineering (ICIME), 2010 The 2nd IEEE International Conference on*. IEEE, 2010.
- [25] Sawangsri, Teerayoot, Vorapoj Patanavijit, & Somchai Jitapunkul. "Face segmentation based on Hue-Cr components and morphological technique." *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*. IEEE, 2005.

- [26] Xiao-Ning Zhang, Z., Jiang, J., Liang, Z. H., & Chun-Liang Liu, L. (2010). Skin color enhancement based on favorite skin color in HSV color space. *Consumer Electronics, IEEE Transactions on*, 56(3), 1789-1793.
- [27] Mita, Takeshi, Toshimitsu Kaneko, & Osamu Hori. "Joint haar-like features for face detection." *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 2. IEEE, 2005.
- [28] Wilson, Phillip Ian, & John Fernandez. "Facial feature detection using Haar classifiers." *Journal of Computing Sciences in Colleges* 21.4 (2006): 127-133.
- [29] Viola, Paul, & Michael Jones. "Rapid object detection using a boosted cascade of simple features." *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE, 2001.
- [30] Weng, Shiuh-Ku, Chung-Ming Kuo, & Shu-Kang Tu. "Video object tracking using adaptive Kalman filter." *Journal of Visual Communication and Image Representation* 17.6 (2006): 1190-1208.
- [31] Moeslund, Thomas B., & Erik Granum. "A survey of computer vision-based human motion capture." *Computer Vision and Image Understanding* 81.3 (2001): 231-268.
- [32] Sawangsri, Teerayoot, Vorapoj Patanavijit, & Somchai Jitapunkul. "Face segmentation based on Hue-Cr components and morphological technique." *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*. IEEE, 2005.

[33] Siebel, Nils T., & Steve Maybank. "Fusion of multiple tracking algorithms for robust people tracking." *Computer Vision—ECCV 2002*. Springer Berlin Heidelberg, 2002. 373-387.

[34] Sidenbladh, Hedvig, Michael J. Black, & David J. Fleet. "Stochastic tracking of 3D human figures using 2D image motion." *Computer Vision—ECCV 2000*. Springer Berlin Heidelberg, 2000. 702-718.

[35] Zhu, H., Zhou, S., Wang, J., & Yin, Z. (2007, August). An algorithm of pornographic image detection. In *Image and Graphics, 2007. ICIG 2007. Fourth International Conference on* (pp. 801-804). IEEE.

[36] Jang, Dae-Sik, Seok-Woo Jang, & Hyung-Il Choi. "2D human body tracking with structural Kalman filter." *Pattern Recognition* 35.10 (2002): 2041-2049.

[37] Cho, J., Mirzaei, S., Oberg, J., & Kastner, R. (2009, February). Fpga-based face detection system using haar classifiers. In *Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays* (pp. 103-112). ACM.

[38] <https://www.youtube.com/watch?v=KDRXDTjdaBU>