Ministry of Higher Education and Scientific Research

University of Diyala

College of Engineering

Computer and Software Engineering Department

DESIGN OF EYE DETECTION SYSTEM

A project

Submitted to the Department of Computer and Software Engineering University of Diyala-College of Engineering, in Partial Fulfillment of the Requirements for the Degree of Bachelor in Computer and Software Engineering

By

Reem Ali Sadiq,

Rusul Majeed Ahmeed

& **AbeerYasain**

Supervised by

Assistant lecturer: **Ahmed Salah Hameed**

2016

بسم الله الرحمن الرحيم

اقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ۝ خَلَقَ الإِنسَانَ مِنْ عَلَقٍ ۝ أقْرَأْ وَرَبُّكَ الأَكْرَم ۝

الَّذِي عَلَّمَ بِالْقَلَمِ ۝ عَلَّمَ الإِنسَانَ مَا لَمْ يَعْلَمْ ۝ كَلاَّ إِنَّ الإِنسَانَ لَيَطْغَى ۝

أَن رَّآهُ اسْتَغْنَى ۝ إِنَّ إِلَى رَبِّكَ الرُّجْعَى ۝ أَرَأَيْتَ الَّذِي يَنْهَى ۝

Dedication

إلى من ركع العطاء أمام قدميها

وأعطتنا من دمها وروحها وعمرها حبا وتصميما ودفعا لغدٍ أجمل

إلى الغالية التي لا نرى الأمل إلا من عينيها(الام الحبيبة)

إلى سندنا وقوتنا وملاذنا بعد الله

إلى من آثرونا على أنفسهم

إلى من علمونا علم الحياة

إلى من أظهروا لنا ما هو أجمل من الحياة

إخوتنا

إلى المتربعة على عرش الأيام

الطفلة التي عمرت بيتها من الحب والحجارة

المهرة الأصيلة التي طالما سبقت دنياها وزمانها

بلدتنا

إلى كل من ساعدنا في انجاز هذا العمل... شكرنا الجزيل و امتناننا

الى الذي تفضل بالإشراف على هذا البحث فجزاه الله عنا كل خير فله منا كل التقدير والاحترام

(الاستاذ احمد صلاح حميد)

SUPERVISOR CERTIFICATION

I certify that this project entitled **"DESIGN OF EYE DETECTION SYSTEMVHDL"** was prepared under my supervision at the Computer and Software Engineering Department/College of Engineering by (**Reem Ali Sadiq, Rusul Majeed Ahmeed, and Abeer Yasain**) as a partial fulfillment of the requirements for the degree of B. Sc. in Computer and Software Engineering.

Signature:
Name: Ahmed Salah Hameed
Title: Asst. Lecturer
Date: / / 2016

I certify that I have carefully read and corrected the final draft of this project for errors in punctuation, spelling and usage.

Proof reader's signature:
Name:
Title: Lecturer
Date: / / 2016

In view of the available recommendations, I forward this project for debate by the examining committee.

Signature:
Name: DR. ALI J. ABBOUD
(Head of Department)
Title: Lecturer
Date: / / 2016

CERTIFICATION OF THE EXAMINATION COMMITTEE

We certify that we have read this project entitled **"DESIGN OF EYE DETECTION SYSTEMVHDL"** and as examining committee examined the students (**Reem Ali Sadiq, Rusul Majeed Ahmeed, & Abeer Yasain)** in its contents and that in our opinion it meets the standards of a project for the degree of B.Sc. in Computer and Software Engineering.

Signature: Signature:

Name:Name:

Title:Title:

(Member) (Member)

Date: / / 2016 Date: / / 2016

Signature:

Name:

Title:

(Chairman)

Date: / / 2016

Approved for Computer and Software Engineering Department

Signature:

Name: DR. ALI J. ABBOUD

(Head of Department)

Title: Lecturer

Date: / / 2016

DECLARATION

we hereby declare that our project entitled **“DESIGN OF EYE DETECTION SYSTEMVHDL”** is the result of our own work and includes nothing which is the outcome of work done in collaboration except as declared in the preface and specified in the text, and is not substantially the same as any that we have submitted, or, is concurrently submitted a degree or diploma or other qualification at the University of Diyala or any other university or similar institution except as declared in the preface and specified in the text. We further state that no substantial part of our thesis has already been submitted, or is concurrently submitted for any such degree, diploma, or other qualification at the University of Diyala or any other university or similar institution except as declared in the preface and specified in the text.

Reem Ali Sadiq

Rusul Majeed Ahmeed

&Abeer Yasain

Abstract

Eye detection is required in many applications like eye-gaze tracking, iris detection, video conferencing, auto-stereoscopic displays, face detection and face recognition. This project proposes a novel technique for eye detection using color and morphological image processing. It is observed that eye regions in an image are characterized by low illumination, high density edges and high contrast as compared to other parts of the face. The method proposed is based on assumption that a frontal face image (full frontal) is available. Firstly, the eye region is detected on RGB scale. Further analysis involves morphological processing using boundary region detection and detection of light source reflection by an eye, commonly known as an eye dot. This gives a finite number of eye candidates from which noise is subsequently removed. This technique is found to be highly efficient and accurate for detecting eyes in frontal face images.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

VPF	Variance Projection Function
IPT	Image Processing Toolbox
RGB	Red Green Blue Image
PSF	point spread function
NTSC	National Television System Committee
FRGC	Face Recognition Grand Challenge
RBFs	Ready Built Factories

CHAPTER ONE
INTRODUCTION

1.1 Introduction of eye detection

Detection of eye is a crucial aspect in many useful applications ranging from face recognition and face detection to human computer interface design model based video coding driver behavior analysis compression techniques development and automatic annotation for image data bases etc. a large number of materials proposing various techniques have been published on this subject in satisfactory enough due to the complexity of the problem .given the benefits of this aspect in multitude of areas a solution to this problem has to be found .generally ,the detection of eyes is done in two phase: locating the face to extract eye regions and then extract the eyes from these regions .previous work on this subject came up with different approaches such as neural network skin color based methods and independent components .little research has been done on the direct search for eyes .an eye pair detection algorithm that can be illustrated using a standard web camera in the real world while skipping the initial segmentation step to extract the face region as commonly done in literature the development eye detection algorithm works on the grey-level image the structure of the eye region is used as a robust cue to locate eye pair candidates .eye pairs are extracted by using binary template matching and eye ball detection method. Detection of the human eye is very difficult task because the contrast of the eye is very poor de formable template (Xie et al ,1994;lam and yan,1996) is the popular method in locating the human eye .in this method ,an eye model is first designed and the eye position can be obtained through a recursive process .however this method is feasible only if initial position of the eye model is placed near the actual eye position moreover deformable template suffer from two other limitations .first it is computation expensive .second the weight factors for energy terms are determined manually .improper selection of the weight factors will yield unexpected result.in view of these limitations lam and yan(1996) introduced the concept of eye corners to guide the recursive process, which partially solved the problem.in (lam and yan,1996), is adopted. However, Xie et al detected corners based on the eye image is relatively low, agood edge image is hard to obtain, and in turn the performance of the eye detection algorithm will be degraded. A typical edge image obtained using the well-known sobel edge detector in line with lam et al. we developed the variance projection function (VPF) which is applied to locate the landmarks (corner points) of an eye. It is observed that some eye landmarks have relatively high contrast such as boundary points between eye white and eyeball .the located landmarks are then employed to guide the eye detection process. A simple and effective eye detection method is also proposed.

1.2 Outline of Project

The project is divided as following:

Chapter one provided a general Introduction to eye detection researches,

Chapter two discusses the image processing as overview,

Chapter three shows a theory background of the project,

Chapter four contains the explanation of main algorithm used in this project,

And Chapter five shows the conclusion and future work related to this work.

CHAPTER TWO
IMAGE PROCESSING OVERVIEW

2.1 Introduction of image processing

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too.

- Image processing includes the following three steps:
- Importing the image via image acquisition tools.
- Analyzing and manipulating the image.
- Output in which result can be altered image or report that is based on image analysis.

There are two types of methods used for image processing namely, analogue and digital image processing. Analogue image processing can be used for the hard copies like printouts and photographs. Image analysts use various fundamentals of interpretation while using these visual techniques.

2.2 Type of Image Processing

1-comperssed image processing

2-Binary: Each pixel is just black or white. Since there are only two possible values for each pixel (0, 1), we only need one bit per pixel.

3- Grayscale: Each pixel is a shade of gray, normally from 0 (black) to 255 (white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally, they are a power of 2.True Color, or RGB: Each pixel has a particular color; that coloris described by the amount of red, green and blue in it. If each of these components has a range 0–255, this gives a total of 256 different possible colors. Such an image is a “stack” of three matrices; representing the red, green and blue values for each pixel. This means that for every pixel there correspond3 value.

2.3 Noise

Image processing is also useful for noise reduction and edge enhancement. We will focus on the types of noise in this work.

2.4 Common Types of Noise:

1**-Salt and pepper noise**

Which is contains random occurrences of black and white pixels

2-Impulse noise:

It is a random occurrence of white pixels

3-Gaussian noise: Variations in intensity drawn from a Gaussian normal distribution**.** Usually and conceptually, when it comes to noise removal for a picture with Gaussian noise, what are the advantages and disadvantages between using a Gaussian averaging filter and not filtering the image at all?

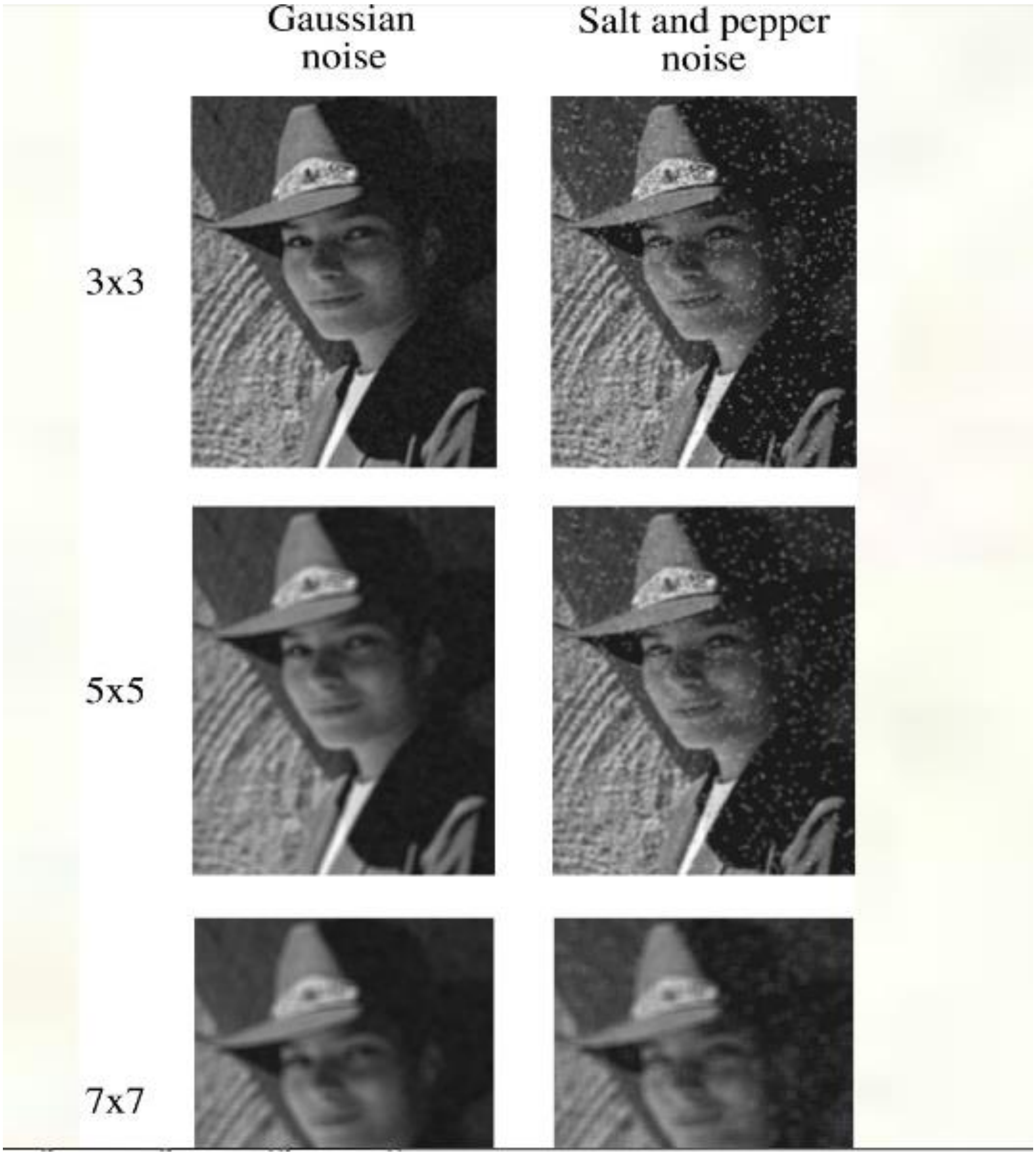


Figure (2.1)

2.5 Types of Filters:

Median filter:

In signal processing, it is often desirable to be able to perform some kind of noise reduction on an image or signal. The median filter is a nonlinear digital filtering technique, often used to remove noise. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise

Mean filter:

The idea of mean filtering is simply to replace each pixel value in an image with the mean (`average') value of its neighbors, including itself. This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the mean. Often a 3×3 square kernel is used, as shown in Figure 1, although larger kernels (*e.g.* 5×5 squares) can be used for more severe smoothing. (Note that a small kernel can be applied more than once in order to produce a similar but not identical effect as a single pass with a large kernel.)

2.6 Applications of Digital Image Processing

Some of the major fields in which digital image processing is widely used are mentioned below:

- Image sharpening and restoration
- Medical field
- Remote sensing
- Transmission and encoding
- Machine/Robot vision
- Color processing
- Pattern recognition
- Video processing
- Microscopic Imaging

There are three major benefits to digital image processing: a consistently high quality of the image, a low cost of processing, and the ability to manipulate all aspects of the process. As long as computer processing speed continues to increase while the cost of storage memory continues to drop, the field is likely to grow.

2.7 Image processing with Matlab

MATLAB is being used as a platform for laboratory exercises and the problems classes in the Image Processing half of the Computer Graphics and Image Processing course unit. This handout describes the MATLAB development environment you will be using, you are expected to have read it and be familiar with it before attempting the Laboratory and Coursework Assignments. MATLAB is a data analysis and visualization tool designed to make matrix manipulation as simple as possible. In addition, it has powerful graphics capabilities and its own programming language. The basic MATLAB distribution can be expanded by adding a range of toolboxes, the one relevant to this course is the image-processing toolbox (IPT).

2.8 MATLAB's Development Environment

MATLAB is started from within the Windows environment by clicking the icon that should be on the desktop. (MATLAB is also available for Linux and MacOS, but these environments are not supported in the laboratories, and these versions have their own licensing arrangements which are not covered by the University's licence.) MATLAB's IDE has five components: the Command Window, the Workspace Browser, the Current Directory Window, the Command History Window and zero or more Figure Windows that are active only to display graphical objects. The Command window is where commands and expressions are typed, and results are presented as appropriate. The workspace is the set of variables that have been created during a session. They are displayed in the Workspace Browser. Additional information about a variable is available there; some variables can also be edited. The current directory window displays the contents of the current working directory and the paths of previous working directories. The working directory may be altered. MATLAB uses a search path to find files. The search path includes the current directory, all of the installed toolboxes plus any other paths that the user has added – via the Set Path dialogue accessed from the File menu. The command history window gives a historical view of the current and previous sessions.

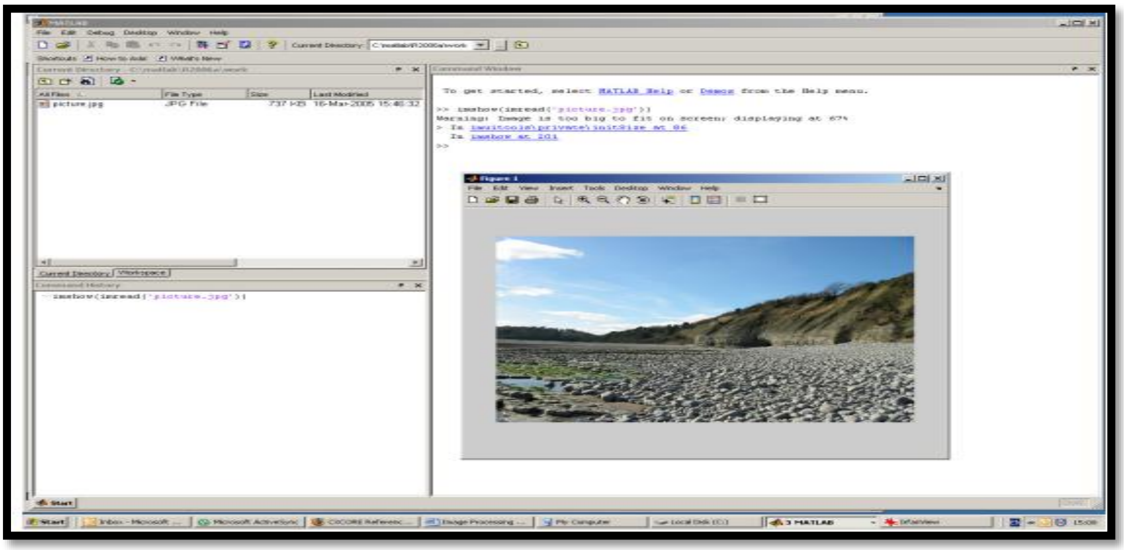


Figure (2.2)

2.9 Histogram

The histogram of an image measures the number of pixels with a given grey or color value. Histograms of color images are not normally used so will not be discussed here. The histogram of an image with L distinct intensity levels in the range [0, G] is defined as the function

$$h(rk) = nk$$

rk is the intensity level in the image, and nk will be the number of pixels with grey value rk . G will be 255 for a uint8 image, 65536 for uint16 and 1.0 for a double image. Since the lower index of MATLAB arrays is one, never zero, r1 will correspond to intensity level 0, etc. For the integer valued images, G = L-1.

We often work with normalized histograms. A normalized histogram is obtained by dividing each element of h(rk) by the total number of pixels in the image (equal to the sum of histogram elements). Such a histogram is called the probability density function (pdf) and reflects the probability of a given intensity level occurring.

$$p(rk) = nk /n$$

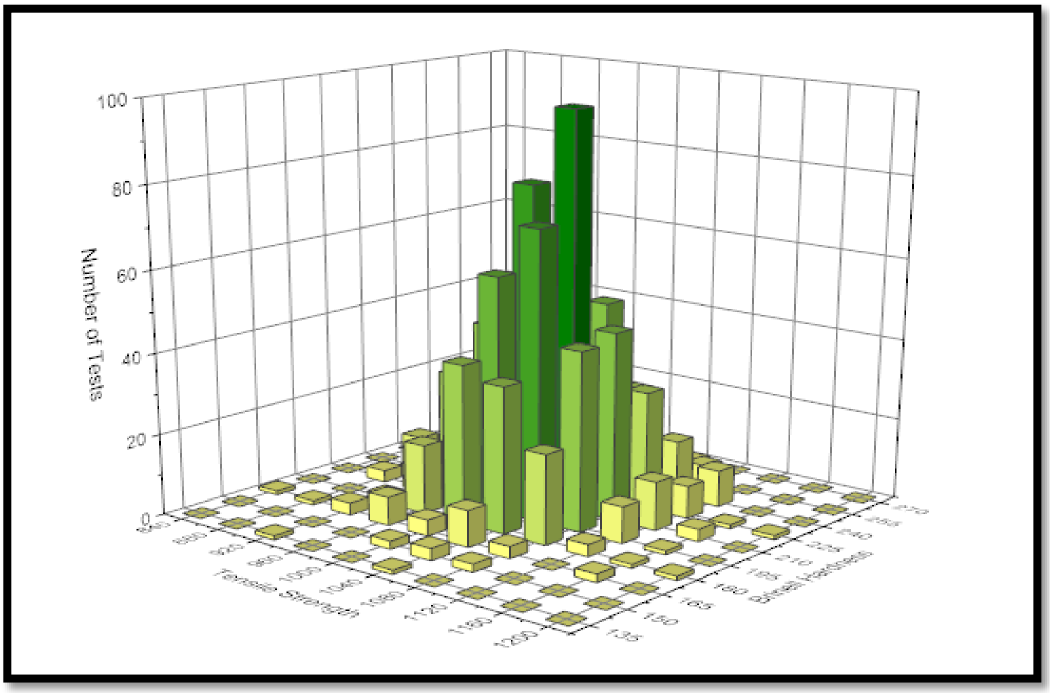


Figure (2.3)

2.10 Thresholding

This is a simple method of differentiating between an object and the background, which works provided they are of different intensities. A threshold value is defined. Intensities greater than this are set to one value, intensities less than to another (1 or max, and 0 or min are often used). Whilst a threshold can be decided manually, it is better to use the image data to compute one.

Gonzalez and Woods suggested this method:

1. Choose a threshold arbitrarily.

2. Threshold the image using it.

3. Compute the mean grey value of the pixels with intensities above and below the threshold, and then compute the average of these two values.

4. Use the new value to re-threshold the image.

5. Repeat steps 3 and 4 until the threshold changes by an insignificant amount.

Method provided by the MATLAB toolbox:

```
>> T = graythresh(A);
```

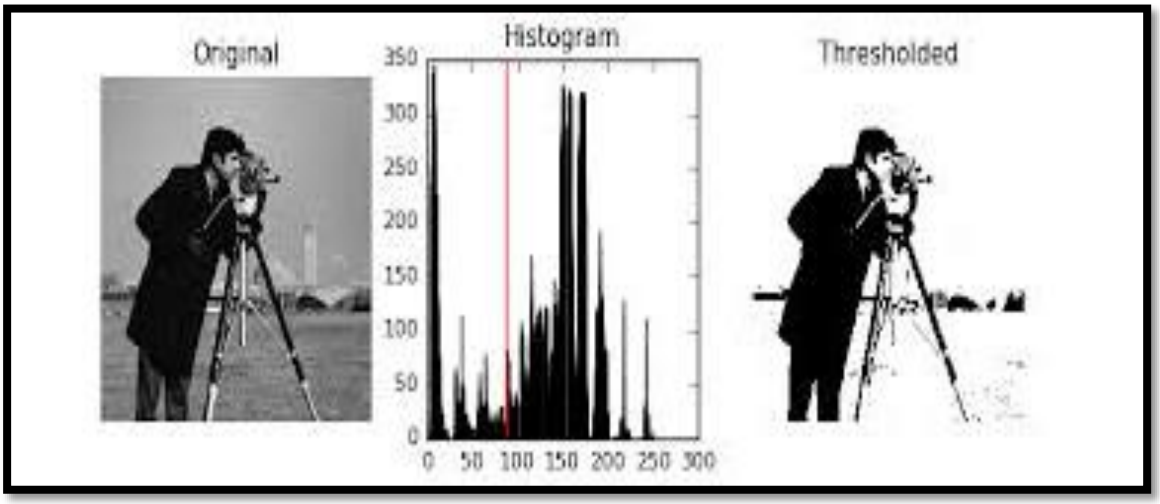


Figure (2.4)

2.11 Image representation

There are five types of images in MATLAB.

1-Grayscale.

A grayscale image *M* pixels tall and *N* pixels wide is represented as a matrix of double data type of size *M*×*N*. Element values (e.g., MyImage(m,n)) denote the pixel grayscale intensities in [0,1] with 0=black and 1=white

2-Truecolor RGB.

A true color red-green-blue (RGB) image is represented as three-dimensional M × N × 3 double matrixes. Each pixel has red, green, blue components along the third dimension with values in [0,1], for example, the color components of pixel (m,n) areMyImage(m,n,1) = red, MyImage(m,n,2) = green, MyImage(m,n,3) = blue.

3-Indexed.

Indexed (paletted) images are represented with an index matrix of size *M*×*N* and a colormap matrix of size *K*×*3*. The colormap holds all colors used in the image and the index matrix represents the pixels by referring to colors in the colormap. For example, if the 22nd color is magenta MyColormap(22,:) = [1,0,1], then MyImage(m,n) = 22 is a magenta-colored pixel.

4- Binary

A binary image is represented by an $M \times N$ logical matrix where pixel values are 1 (true) or 0 (false).

5- Uint8

This type uses less memory and some operations compute faster than with double types. For simplicity, this tutorial does not discuss uint8 further.

2.12 Region Processing

1- High Pass - Edge Detection

An edge is defined as a significant, local change in image intensity. The simplest (ideal) edge is a line (the edge) separating uniform regions of different intensity. In practice, this is never observed as noise superimposes random fluctuations on the image data, and imperfect optics and digitization blur the image. The ideal edge is therefore corrupted and we see a rather more gradual change in intensity from one region to the other. The problem of edge detection is to locate the edge in this data (if it is possible, and if it makes sense). If an edge is a discontinuity in intensity, there must be a region around it where the intensity changes by large amounts over small distances, i.e. the gradient is high. The gradient can be estimated by measuring the differences in intensity over small distances and dividing by the distance

2- Low Pass – Smoothing

The aim of smoothing algorithms is to reduce the small scale, small amplitude fluctuations in the data, usually associated with noise. (Noise is due to a multitude of causes, too many to list, it is observed in one of two ways: salt and pepper noise where isolated individual pixels have very different values to their neighbors', and random noise where all pixels' values may differ from the correct value – the statistics of the differences can be described.)

2.13 Wiener Filter

The Wiener filter is designed to remove image degradations, provided we are able to model the degradation. Degradations such as relative motion of the camera and subject, out-of-focus optics or light scattering can be modelled. We assume that the observed image, f, is generated by adding noise, n, to the ideal image g that is convolved with some function, H, that represents the total imaging system (i.e. includes the degradation).so the observed image can be described as:

$$f = gH + n$$

The process of computing g from f, assuming n and H, is called deconvolution. MATLAB provides a single function to perform it:

>>dst = deconvwnr(src, PSF); PSF is the degradation model that you assume. The function has further arguments that can improve the results – if their values are known accurately.

CHAPTER THREE
THEORY BACKGROUND

3.1 Eye Detection

The eye detection technique used here is based on the fact that whenever an eye is properly illuminated, it has a sharp point of reflection. This point will be referred to as a light dot. NTSC format is most efficient in exploiting this light dot. Hence, the image is first converted into an NTSC image and the chrominance values of its pixels are used for further processing the next step in eye detection involves edge detection. Morphological techniques are used for boundary detection. Dilation, followed by erosion and the calculation of differences between the two produces an image with boundaries. The structuring element used in dilation and erosion has a large matrix (9×9 ones), so that clear and thick boundaries are detected. For the purpose at hand, this technique is found to be more efficient that the Canny edge detector. This is followed by suitable thresholding of the image. The light dot is one of the first few candidates that stand out when the image is thresholded. Selection of a proper thresholding value is important, and a complex process, as this value varies as the image changes. Hence, an adaptive thresholding technique is used. Calculation of this value is achieved by an iteration process. The value is started at 220 on a 255 scale and is decreased by 5 per iteration. The aim of the iteration process is to obtain a minimum number of four and a maximum number of six blobs. It was observed statistically that between these values, thresholding is most probable to output both eyes and with minimum noise. Thresholding is followed by morphological closing, which improves the ease of eye detection and eliminates stray points. Sample outputs of figure () are shown in figure ().Some of the blobs obtained in the image are too small, and some are too large, and both are unlikely to be candidates for an eye. Hence, they are morphologically eliminated. This is followed by eliminating long and slender blobs, either horizontal or vertical ones, as these blobs are certainly not eyes. Again, morphological binary image processing is used. This image and skin image are combined by using an 'And' operation. Only the common candidates survive, as eyes invariably lie in the face region. This whole process is included in a loop and the number of emerging candidates is checked at the end of the loop. If this number is not between 4 and 6, the threshold value is changed, for the next loop.

Figure (3.1)

3.2 Eye Detection Error on Face Recognition

To observe how the recognition performance varies according to eye localization error, the eye positions of the ground truth are artificially perturbed with random noise. Face recognition is then performed using the perturbed eye positions. The impact of eye detection on recognition accuracy is illustrated in Figure (), where face images are aligned based on perturbed eye positions. The eye localization error in Figure () is the pixel error normalized by the distance between two eyes. Given a normalized error, the random noise is uniformly distributed at a circle in 2D space. The data from FRGC 1.0 and PCA baseline algorithm are used for this experiment Figure () clearly shows that eye location errors significantly affect the recognition accuracy. For example, about1% (about 3 pixels for FRGC image or 0.5 pixel if the inter-ocular distance is 50 pixels) eye location error reduces the face recognition accuracy by over 10%. When the error is about 5%, the face recognition accuracy reduces by50%. These numbers, of course, vary, depending on the recognition methods. But still, they show the significant impact of eye position error on face recognition.

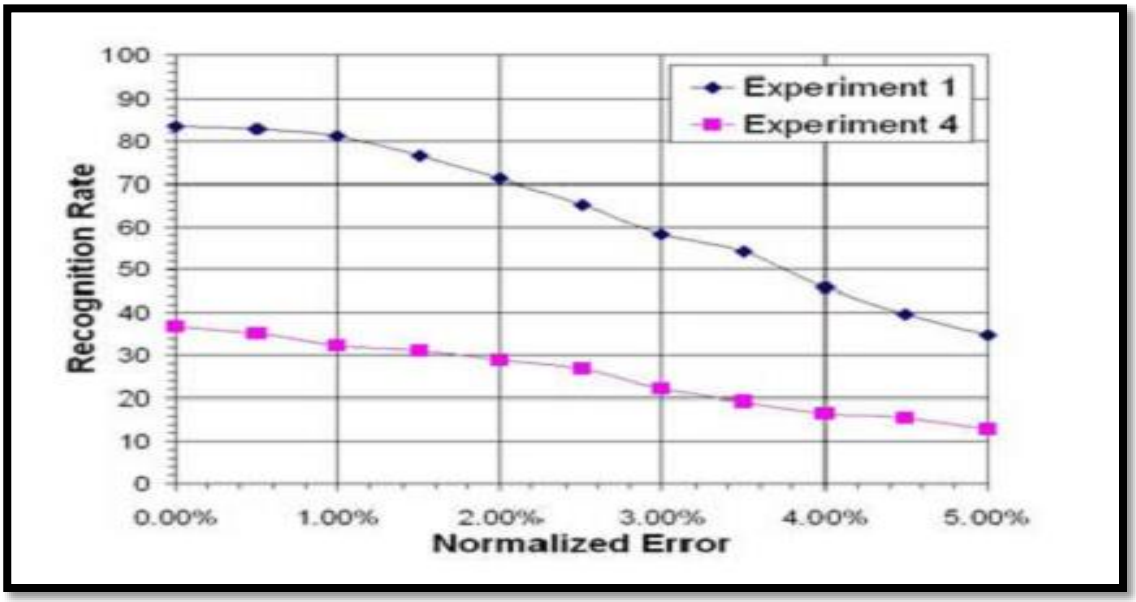


Figure (3.2)

3.3 Brief Review on Automatic Eye Detection

There are two purposes of eye detection. One is to detect the existence of eyes, and another is to accurately locate eye positions. Under most situations, the eye position is measured with the pupil center .Current eye detection methods can be divided into two categories: active and passive eye detection the active detection methods use special types of illumination. Under IR illumination, pupils show physical properties which can be utilized to localize eyes the advantages of active eye detection methods are that they are very accurate and robust. The disadvantages are that they need special lighting sources and have more false detections with an outdoor environment, where the outdoor illumination impacts the IR illumination .Passive methods directly detect eyes from images within visual spectrum and normal illumination. Some early work extracts distinct features from eyes localization. The features include image gradients projection and tem-plates However, in these methods, heuristics and Post processing are usually necessary to remove false detections, and these features are sensitive to image noise. Be-sides the above features, wavelets are shown to be able to localize facial features. Huang and Wechsler propose to select optimal Wavelet packets and classify the eye and non-eye with Radial Basis Functions (RBFs) Gabor wavelets are robust to moderate illumination change ,Eye Detection ValidationTo quantitatively validate the performance of our eye detection method, we performed two experiments. In the first experiment, we compared the detected eye positions with the manually labeled eye positions. The performance of our eye detector is characterized by the eye detection rate and eye localization error. The localization error is measured as the Euclidean distance between the detected eye positions and manual eye positions. In the second experiment, we quantify the performance of our eye detection based on its influence on face recognition accuracy of two standard baseline methods: PCA and PCA+LDA. For both experiments

3.4 Eye Detection Accuracy

We apply our eye detection method to all of the 2D images in the FRGC 1.0 database. The frontal face detection rate is approximately 95.0%. Usually the missing faces are caused by uncontrolled illumination. For eye detection alone, it achieves a detection rate about 99.0% on the detected faces.

As a result, we have an overall 94.5% eye detection rate for FRGC 1.0. Table 1 shows the horizontal and vertical eye localization errors, as well as the total error. Additionally, Table shows both pixel and normalized errors, where the normalized error is the pixel error normalized by the distance between two eyes. The average Euclidean distance between automatic eyes and ground truth is about 6.4 pixels, which accounts for 2.67% normalized error.

3.5 Eye Tracking by using Integrated Eye Tracker

Eye tracking plays an important role in many applications such as gesture understanding, fatigue driving, eye blink detecting, disabled-helping domain, psychology domain, human-machine interaction, face recognition in video, and so on. So eye tracking is the focus problem in the researching domain of human-machine interaction and computer vision in recent years.

The difficulty of eye tracking is that eye is affected by many factors includinglighting, gesture and covering objects. Many methods of eye tracking are introduced in. A variety of computational methods can be used to solve the tracking parameters by amongthese; the Kalman filtering method has the following advantages. First, it is a computationally efficient recursive procedure requiring minimum amount of storage for the past samples. It embodies the information about the system and measurement noise in its model. It can also effectively deal with time varying signals. The results of the previous step are used to predict the current states. Further, the accuracy of the estimation can be assessed by monitoring the error covariance The Kalman tracker, however, may fail if pupils are not bright enough under the conditions mentioned previously. In addition, rapid head movement may also cause the tracker to lose the eyes. This problem is addressed by augmenting the Kalman tracker with the Mean shift tracker. Figure () summarizes our eye tracking scheme

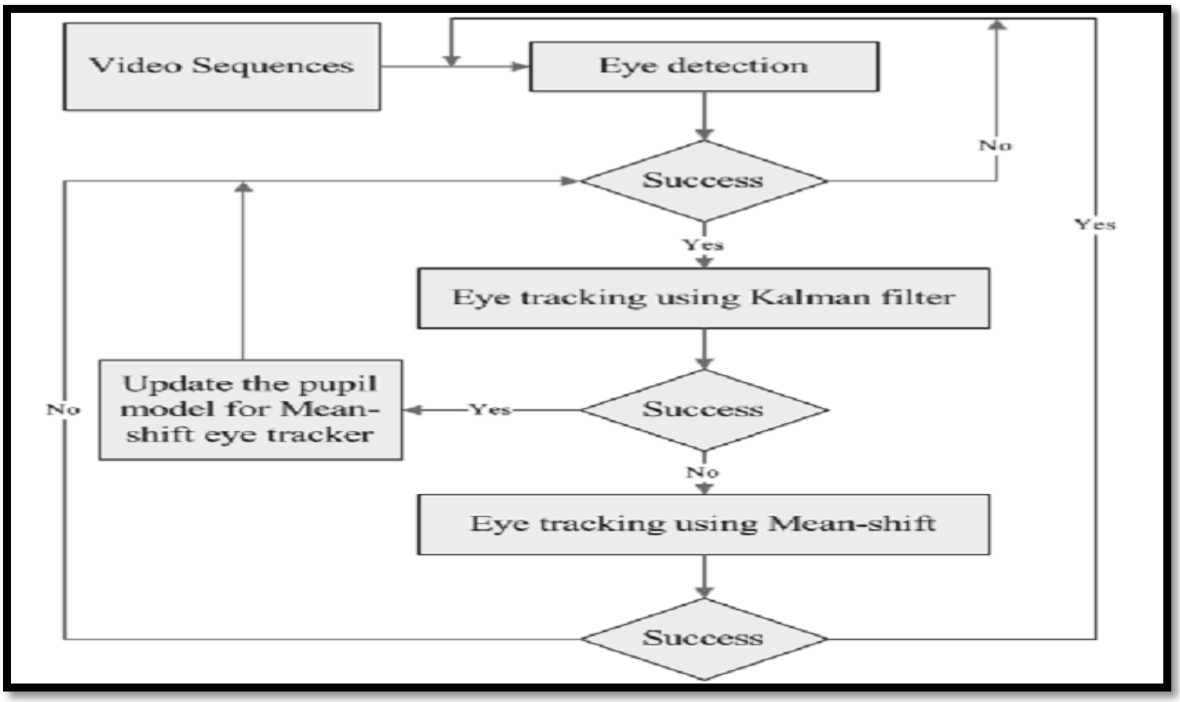


Figure (3.3)

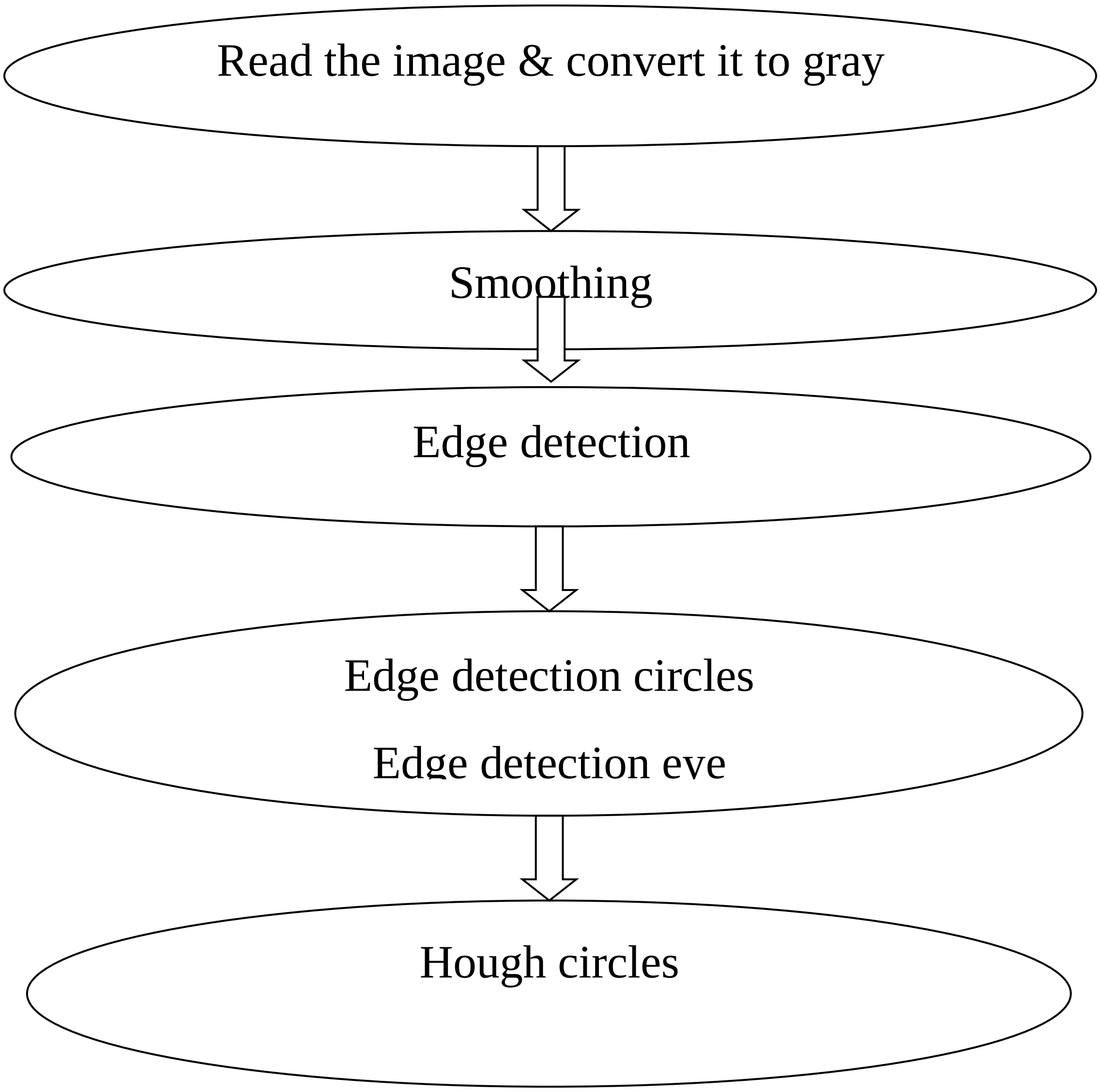
CHAPTER FOUR

ALGORITHM IMPLEMENTATION

4.1 Introduction

The first step we have to accomplish this project is that we used the Matlab effectively is the best software language of the most common to use we have adopted in this project on a group of algorithms.

4.2 The main algorithms



- **The first step**

Read the image and turn it into a gray level

```
w=imread('C:\Users\mntther\Desktop\g.png')
figure,imshow(w)
i=rgb2gray(w);
figure,imshow(i)
```

Figure (4.1)

Figure (4.2)

- **The second step**

Smoothing and soften the image and remove the flaws and highlight the required features and hide the rest of the features of non-grata

```
m=medfilt2(i);
figure,imshow(m)
j1=fspecial('disk',radius);
k1=imfilter(m,j1,'circular');
figure,imshow(k1)
```

Figure (4.3)

- **The third step**

The third step is to identify the edges and the purpose of this step is that we are after our fine-tuning of the image is that we made a member of the eye highlights over the eye that are more members will smoothing as are at least containing the chromatin who black and white as well as the rest of the colors who is green Blue and brown but the rest of the member is meant here is the mouth and nose, it is possible to be affected by the process of smoothing but few mainly because it comes off approach to the color of a person's face Only in this step has been to identify the edges private eye to it is the previous step, you have led features of the mouth and nose and highlight the features of the eye more

```
x=edge(m,'sobel');
imshow(x)
```

Figure (4.4)

- **The fourth step**

The goal of this step is to fill in the gaps in your pupil eye as shown in the picture

```
E=imfill(x,'holes');
figure,imshow(E);
```

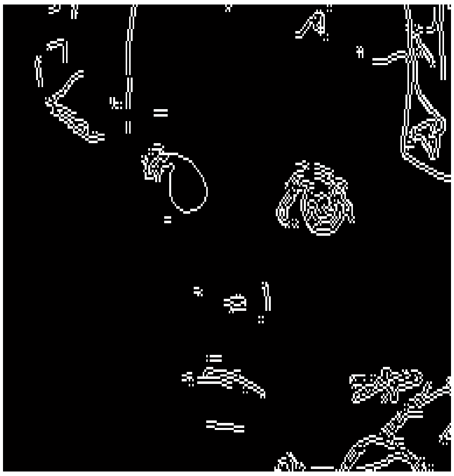


Figure (4.5)

Figure (4.6)

- **The fifth step**

In this step we are right idea of the project in this step was radii specific department account and then characterize this fair to the original image and determine the center of the eye as shown in Fig.

```
h=circle_hough(F,radii,'same');
theta=linspace(0,2*pi,N);
rho=ones(1,N)*radius;
[X,Y]=pol2cart(theta,rho);
figure;
```

Figure (4.7)

CHAPTER FIVE
CONCLUSION AND FUTURE WORKS

5.1 Conclusion

The conclusion of this research is to distinguish between the closed eyes and open the exception of a set of images by the instruction (imfill) where this instruct fill gaps in the eye (pupil) when the eye is open, but in the case of eye closed is disabled this instruct and fails cycle because in the case of closed-eye is determined edges only either deviate circuits and determine the pupil Flaimkn and so was the distinction between the eye closed or open through this instruct.

As in the case of the video application, the video contains a collection of images in if the video contains a collection of images in the case of the open eye can distinguish the number of eye lashes if the person sitting in front of the computer or television and so on

As in the case of the video contains images in the case of eye closed so doing no longer be there movement or eyelashes not are or eye be open but without eyelashes (static) In these cases can distinguish that the person is asleep and that a person is sitting in front of a particular device does not exist

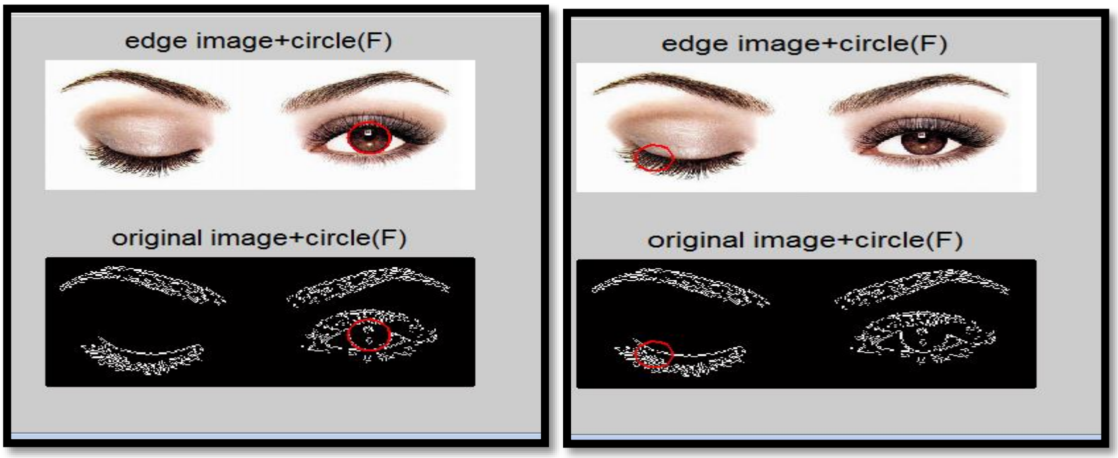


Figure (5.1)

5.2 Future works

Steps for future research that is implemented by hardware through technology (FpgA), a field-programmable gate array (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing – hence "field-programmable". The FPGA configuration is generally specified using a hardware description language (HDL), similar to that used for an application-specific integrated circuit (ASIC)

where it was this research included in several applications of the most important of these applications is that the person sitting in his car and lost his focus when driving is possible through this device give alert him either in medical applications can figure out the sick person has consciousness or not, and in many other applications, for example, the long focus on something specific as possible to give the alert him That this is harm to the eye and this is a step for the future of this research because it is form the threads modern and consider is the application of materials for (image processing)

REFRENCES

[1] Foley, J.D., VanDam, A. (1982) “FUNDAMENTALS OF INTERACTIVE COMPUTER GRAPHICS”

[2] Gonzalez,R.C. (1977), “DIGITAL IMAGE PROCESSING, Reading Massachusetts”

[3]https://www.google.iq/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=image%20processing

[4]https://www.google.iq/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=image%20proce%20ssing%20type

[5]http://studentnet.cs.manchester.ac.uk/ugt/COMP27112/doc/matlab.pdf

[6]http://publications.theseus.fi/bitstream/handle/10024/43274/FinalThesis.pdf

[7]https://icme.hpc.msstate.edu/mediawiki/index.php/Image_Processing_with_MATLAB_1

[8]http://www.getreuer.info/tutorials/matlabimaging

[9]http://www.mil.ufl.edu/publications/fcrar09/eye_detection_tanmay_rajpathak_fcrar_09.pdf

[10]https://cdn.fbsbx.com/hphotos-xft1/v/t59.2708 21/13105086_560088754182409_35446222_n.pdf/3.pdf?oh=7f034404b82c4bcf4ab7ba3e5afa9691&oe=57286E1B&dl=1

[11]https://www.google.iq/search?q=eye_detection_tanmay_rajpathak_fcrar_091+(3)&oq=eye_detection_tanmay_rajpathak_fcrar_091+(3)&aqs=chrome..69i57.1247j0j7&sourceid=chrome&ie=UTF-8

APPENDIX

```
f=imread('C:\Users\ammeer\Desktop\n.png')
i=rgb2gray(f);
figure,imshow(i)
m=medfilt2(i);
figure,imshow(m)
radius=2;
j1=fspecial('disk',radius);
k1=imfilter(m,j1,'circular');
figure,imshow(k1)
x=edge(m,'sobel');
[m,n]=size(x);
E=imfill(x,'holes');
figure,imshow(E);
F=edge(E,'prewitt');
figure,imshow(F);
radii=14:44:49;
h=circle_hough(F,radii,'same');
[~,maxIndex]=max(h(:));
[i,j,k]=ind2sub(size(h),maxIndex);
radius=radii(k);
center.x=j;
center.y=i;
N=20
theta=linspace(0,2*pi,N);
rho=ones(1,N)*radius;
[X,Y]=pol2cart(theta,rho);
figure;
```

```
subplot(2,1,1)
imshow(F);hold on
plot(center.x-X,center.y-Y,'r-','linewidth',2);
title('edge image+circle(F)','fontsize',16)
subplot(2,1,2)
imshow(f);hold on
plot(center.x-X,center.y-Y,'r-','linewidth',2);
title('original image+circle(F)','fontsize',16)
plot(center.x-X,center.y-Y,'r-','linewidth',2);
```

الخلاصة

ان الكشف عن العين يدخل في العديد من التطبيقات مثل تتبع العين المبصرة، والكشف عن قزحية العين، و في المؤتمرات الفيديوية، وكشف الوجه والتعرف على الوجه. في هذا المشروع تقدم تقنية جديدة للكشف عن العين باستخدام معالجة الصور . لوحظ من خلال البحث ان مناطق العين في صورة تتميز بإضاءة منخفضة، وحواف عالية الكثافة والتباين العالي بالمقارنة مع الأجزاء الأخرى من الوجه. وتستند هذه الطريقة المقترحة على افتراض أن صورة الوجه الأمامية متاحة للعمل عليها . أولا، يتم تحديد العين في مستوى الصورة الملونةRGB . و من ثم يتم تطبيق المزيد من التحليل الصوري باستخدام الكشف عن منطقة الحدود والكشف عن انعكاس مصدر الضوء من العين، والمعروف باسم نقطة العين. و من ثم ترحل الصورة الى عدد محدود من مرشحات العين حيث يتم إزالة الضجيج في وقت لاحق . ان هذه التقنية تعتبر ذات كفاءة عالية ودقيقة للكشف عن عيون في صور الوجه الأمامية

وزارة التعليم العالي والبحث العلمي

كلية الهندسة

جامعة ديالى

قسم هندسة الحاسبات والبرامجيات

تصميم نظام الكشف عن العين

مشروع مقدم الى قسم هندسة الحاسبات والبرامجيات في جامعة ديالى/ كلية الهندسة كجزء من متطلبات نيل درجة

البكالوريوس في هندسة الحاسبات والبرجيات

اعداد الطلاب

ريم علي صادق

رسل مجيد احمد

عبير ياسين

بأشراف

المدرس المساعد: احمد صلاح حميد

2016