FACE RECOGNITION BASED ATTENDANCE SYSTEM USING PYTHON GUI

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in

INFORMATION TECHNOLOGY

By

SANJUDHA M (Reg No 38120074)
MOKITHA B (Reg No 38120051)

DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY 2022

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with “A” grade by NAAC I 12B Status by UGC I Approved by AICTE
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TEHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide workof**SANJUDHA M(Reg no:38120074)** and **MOKITHA B (Reg no:38120051)** carried out the projectentitled **“FACE RECOGNITION BASED ATTENDANCE SYSTEM USING PYTHON GUI”**under our supervision fromOCTOBER 2021 to APRIL 2022

Internal Guide

Dr.JEBERSON RETNARAJ,M.E., Ph.D.,

Head of the Department

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on ___________________

InternalExaminer **ExternalExaminer**

DECLARATION

We, **SANJUDHA M**and**MOKITHA B**hereby declare that theProject Reportentitleddonebymeunder theguidanceof **Dr. R. JEBERSON RETNA RAJ M.E., Ph.D.,** atSathyabamaInstituteofScienceandTechnology(DeemedtobeUniversity),JeppiaarNagar,RajivGandhiSalai,Chennai - 600119 is submitted in partial fulfillment of the requirements for theawardofBachelorofTechnologydegreeinInformationTechnology.

.

DATE:
PLACE:CHENNAI SIGNATUREOFTHECANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge my sincere thanks to Board of management of**SATHYABAMA** for their kind encouragement in doing this project and for completing itsuccessfully. We aregrateful tothem.

We convey my thanks to **Dr. T. SASIKALA M.E., Ph.D., Dean, School ofComputing**and**Dr.R.SUBHASHINIM.E.,Ph.D.,HeadoftheDepartment,Department of Information Technology** for providing me the necessary support and detailsat therighttimeduringthe progressivereviews.

I would like to express our sincere and deep sense of gratitude to our Project Guide**Dr. R. JEBERSON RATNA RAJ M.E., Ph.D.,** forhervaluableguidance,suggestionsandconstantencouragementpavedwayforthe successfulcompletionofour projectwork.

I wish to express my thanks to all Teaching and Non-teaching staff membersoftheDepartmentof**INFORMATIONTECHNOLOGY**whowerehelpfulinmanyways for the completion of theproject.

ABSTRACT

FACE RECOGNITION BASED ATTENDANCE SYSTEM USING PYTHON GUI

Face recognition systems are part of facial image processing applications and theirsignificanceasaresearchareaareincreasingrecently.Theyusebiometricinformationo fthehumansandareapplicableeasily insteadoffingerprint,iris,signature etc., because these types of biometrics are not much suitable for non-collaborative people. Face recognition systems are usually applied and preferred forpeople andsecurity camerasinmetropolitanlife.Thesesystemscanbe usedforcrimeprevention,videosurveillance,personverification,andsimilarsecurityactiviti es. We describe a face recognition-based automated attendance system utilizing a Python GUI in this work. This technique has a lot of applications in everyday life, notably in school and college. Scaling of the image size is conducted at the first phase, or pre-processing stage, to avoid or minimize information loss. HAAR CASCADE and XGBOOST are the algorithms involved. Overall, we created a Python programme that accepts an image from a database, performs all necessary conversions for picture identification, then confirms the image in video or real time using a user-friendly interface by accessing the camera. The name and time of the successful match is then recorded.Face detection and recognition are two of the most demanding computer vision applications and outcomes. This area has traditionally been a prominent focus of image analysis research because to its function as the primary identification strategy for human faces. It's both exciting, as well as biometrics, pattern recognition Also, teaching a machine to accomplish this is challenging. Face recognition is another one of the toughest problems in computer vision. Recognizing and detecting faces and computer vision, are all hot issues in the medical and research industries.There are several software programmes or technologies that have improved to the point where even blurry images may be reconstructed and analysed to understand more about a person's personality. Facial recognition technology is a framework or programme that analyses an image or video footage to recognise people's faces and authenticate their identification. The face is a one-of-a-kind reflection of a person's personality. Face recognition is a biometric approach that involves matching a real-time image with previously stored photographs of the same individual in a database to identify a person

Keywords:**HAAR CASCADE, XGBOOST,** Face Recognition, Face Detection, Attendance automation.

INDEX

LISTOFFIGURES

LISTOFABBREVATION

ACRONYM	ABBREVIATION
RGB	Red-Green-Blue
YCbCr	Luminance-blue Difference Chroma-red Difference Chroma
HSV	Hue-Saturation-Value
YUV	Luminance-Blue Luminance Difference-Red Luminance Difference
FFNN	Feed Forward Neural Network
SOM	Self Organizing Map
NN	Neural Network
PCA	Principal Component Analysis
SVM	Support Vector Machines
BP	Back Propagation
DCT	Discrete Cosine Transform
RBNN	Radial Basis Neural Network
LDA	Linear Discriminant Analysis
ICA	Independent Component Analysis
HMM	Hidden Markov Model
PTZ	Pan/Tilt/Zoom
LoG	Laplacian of Gaussian

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Facerecognitionsystemisacompleximage-processingprobleminrealworldapplications with complex effects of illumination, occlusion, and imaging conditionontheliveimages.Itisacombinationoffacedetectionandrecognitiontechniquesin image analyzes. Detection application is used to find position of the faces in agiven image. Recognition algorithm is used to classify given images with knownstructured properties, which are used commonly in most of the computer visionapplications.Theseimageshavesomeknownpropertieslike;sameresolution,including same facial feature components, and similar eye alignment. These imageswill be refered as “standard image” in the further sections. Recognition applicationsuses standard images, and detection algorithms detect the faces and extract faceimages which include eyes, eyebrows, nose, and mouth. That makes the algorithmmore complicated than single detection or recognition algorithm. The first step forface recognition system is to acquire an image from a camera. Second step is facedetection from theacquired image.As a third step, face recognition thattakes theface images from output of detection part. Final step is person identity as a result ofrecognition part. An illustration of the steps for the face recognition system is giveninFigure1.

Acquiringimagestocomputerfromcameraandcomputationalmedium(environment)via framegrabberisthefirststepinfacerecognitionsystemapplications. The input image, in the form of digital data, is sent to face detectionalgorithm part of a software for extracting each face in the image. Many methods areavailable for detecting faces in the images in the literature [1 - 29]. The availablemethods could be classified into two main groups as; knowledge-based [1 - 15] andappearance-based [16 - 29] methods. Briefly, knowledge-based methods are derivedfrom human knowledge for features that makes a face. Appearance-based methodsare derived from training and/or learning methods to find faces. The details about themethodswill besummarized in thenext chapter.

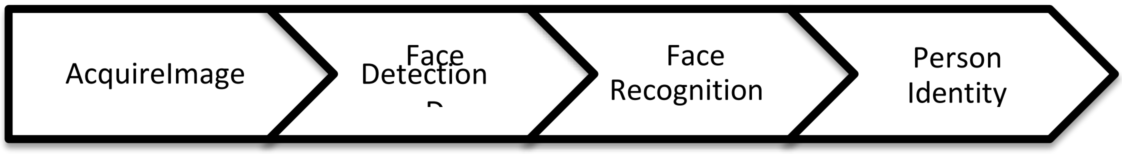


Figure1.1:Stepsof FaceRecognitionSystemApplications

After faces are detected, the faces should be recognized to identify the persons in thefaceimages.Intheliterature,mostofthemethodsusedimagesfromanavailableface library, which is made of standard images [30 - 47]. After faces are detected,standardimagesshouldbecreatedwithsomemethods.Whilethestandardimage sarecreated,thefacescouldbesenttorecognitionalgorithm.Intheliterature,methodscanbe dividedintotwogroupsas2Dand3Dbasedmethods.In2Dmethods,2Dimagesareusedasin putandsomelearning/trainingmethodsareusedtoclassifytheidentificationofpeople[30-43].In3Dmethods,thethree-dimensional data of face are used as an input for recognition. Different approachesare used for recognition, i.e. using corresponding point measure, average half face,and 3D geometric measure [44 - 47]. Details about the methods will be explained inthenextsection.

Methodsforfacedetectionandrecognitionsystemscanbeaffectedbypose,presence or absence of structural components, facial expression, occlusion, imageorientation,imagingconditions,andtimedelay(forrecognition).Availableapplication s developed by researchers can usually handle one or two effects only,thereforetheyhavelimitedcapabilitieswithfocusonsomewell-structured application. A robust face recognition system is difficult to develop which worksunderall conditions with awidescopeofeffect.

1.2. STATEMENT OF THE PROBLEM

Main problem of thesis is to design and implement a face recognition system in theRobot Vision Laboratory of the Department of Mechatronics Engineering at theAtılım University. The system should detect faces in live acquired images inside thelaboratory,anddetectedfacesshouldberecognized.Thissystemwillbeintegratedto projects of guide robot, guard robot, office robot, etc. Later on, this thesis will bepartofhumanoid robotproject.

1.3. SCOPE AND OUTLINE OF THESIS

Scope of the thesis is stated as follows:

- Facerecognitionsystemwilldetect,extractandrecognizefrontalfacesfromacquiredli veimages in laboratoryenvironment.
- Systemshouldworkunderchanginglightingconditionsinthelaboratory.
- Systemshouldrecognize50peopleatleast.
- Systemshouldnot extractfaces iftheyusesunglasses.
- Systemwillnotdetectprofileandnon-frontalfaceimages.
- Outlineofthesis is stated as follows:
- Chapter2introducesfacedetection,facerecognition,andfacerecognitionsystemap plications that exist inliterature.
- Chapter3describestheoryoffacerecognitionsystemthatisbasedonproblemstatem ent ofthesis.
- Chapter4summarizestheexperimentsperformedandtheirresultsusingtheface recognition system.
- Chapter5discussesandconcludesthesis.
- Chapter6givessomefutureworksonthethesistopic.

CHAPTER 2

LITERATURE SURVEY

Although face recognition systems are known for decades, there are many activeresearchwork on thetopic. Thesubjectcan bedividedinto threeparts;

1. Detection
2. Recognition
3. Detection&Recognition

Face detection is the first step of face recognition system. Output of the detection canbe location of face region as a whole, and location of face region with facial features(i.e.eyes,mouth,eyebrow,noseetc.).Detectionmethodsintheliteraturearediffi culttoclassifystrictly,becausemostofthealgorithmsarecombinationofmethodsfordetec tingfacestoincreasetheaccuracy.Mainly,detectioncanbeclassified into two groups as Knowledge-Based Methods and Image-Based Methods.Themethods fordetection aregiveninFigure2.

Knowledge-Based methods use information about Facial Features, Skin Color orTemplate Matching. Facial Features are used to find eyes, mouth, nose or other facialfeatures to detect the human faces. Skin color is different from other colors andunique, and its characteristics do not change with respect to changes in pose andocclusion. Skin color is modeled in each color spaces like RGB (Red-Green-Blue),YCbCr (Luminance-Blue Difference Chroma-Red Difference Chroma), HSV (Hue-Saturation-Value), YUV (Luminance-Blue Luminance Difference-Red LuminanceDifference), and in statistical models. Face has a unique pattern to differentiate fromotherobjects andhencea templatecanbegeneratedto scanand detectfaces.

Facialfeaturesareimportantinformationforhumanfacesandstandardimagescanbe generated using these information. In literature, many detection algorithms basedonfacial features areavailable[1 -6]. Zhi-fang et al. [1] detect faces and facial features by extraction of skin like regionwithYCbCrcolorspaceandedgesaredetectedintheskinlikeregion.Then,eyesar e found with Principal Component Analysis (PCA) on the edged region. Finally,Mouth is found based on geometrical information. Another approach extracts skinlikeregionwithNormalizedRGBcolorspaceandfaceisverifiedbytemplatematching.

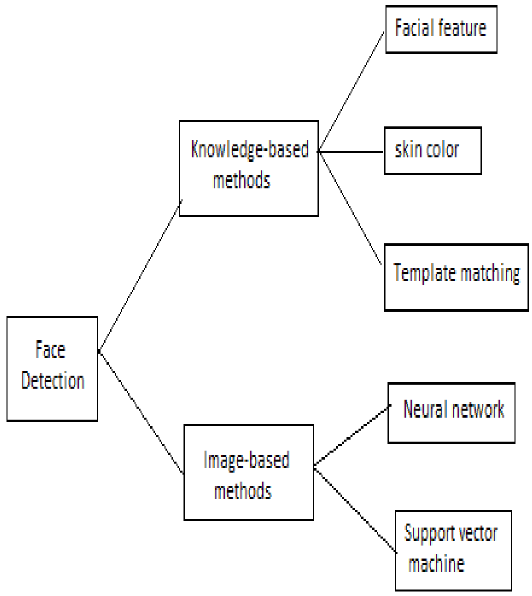


Figure 1.2:MethodsforFaceDetection

Tofindeyes,eyebrowsandmouth,colorsnakesareappliedtoverifiedface image [2]. Ruan and Yin [3] segment skin regions in YCbCr color space andfaces are verified with Linear Support Vector Machine (SVM). For final verificationof face, eyes and mouth are found with the information of Cb and Cr difference. Foreye region Cb value is greater than Cr value and for mouth region Cr value is greaterthan Cb value. Another application segments skin like regions with statistical model.

StatisticalmodelismadefromskincolorvaluesinCbandCrchannelinYCbCr color space. Then, face candidates are chosen with respect to rectangular ratio ofsegmented region. Finally, the candidates are verified with eye & mouth map [4].Also, RGB color space can be used to segment skin like region and skin color likeregionisextractedtobe facecandidate.Candidate isverifiedby findingfacialfeatures. Eyes and mouth are found based on isosceles triangle property. Two eyesand one mouth create an isosceles triangle and also distance between two eyes anddistance from mid-point of eyes to mouth are equal. After eyes and mouth is found,FeedForward Neural Network (FFNN) is used for final verification of face candidate[5]. Bebar et al. [6] segment with YCbCr color space and eyes & mouth are found onthecombinationofsegmentedimageandedgedimage.Forfinalverification,horizontal and vertical profiles of the images are used to verify the position of theeyes and mouth. All the methods are using firstly skin segmentation to eliminatenonfaceobjects in theimages to savecomputationaltime.

Skin color is one of the most significant features of human face. Skin color can bemodeled with parameterized or non parameterized methods. Skin color region can beidentified in terms of threshold region, elliptical modeling, statistical modeling (i.e.Gaussian Modeling), or Neural Network. Skin color is described in all color spaceslike RGB, YCbCr, and HSV. RGB is sensitive to light changes but YCbCr and HSVare not sensitive to light changes. The reason is that these two color spaces haveseparate intensity and color channel. In literature many algorithms based on skincolor available [7 - 13]. Kherchaoui and Houacine [7] modeled skin color usingGaussian Distribution Model with Cb and Cr channel in YCbCr color space. Thenskinlikeregionischosenasafacecandidatewithrespecttotheboundingboxratioof the region and candidates are verified with template matching. Another methodpreprocesses the given image to remove background part as a first step. It is done byapplying edge detection on the Y component of YCbCr color space. Then, the closedregion is filled to take it as foreground part. After that, skin segmentation is done onYCrCb color space with conditions. The segmented parts are taken as candidate andverificationisdonebycalculatingtheentropyofthecandidateimageandusethresholding to verify face candidate [8]. Qiang-rong and Hua-lan [9] applied whitebalancecorrectionbeforedetectingfaces.Thecolorvalueisimportantforsegmentation,sowhileacquiringtheimagecolorsmayreflectfalsecolor.Toovercomethis,whitebalancecorrectionshouldbedoneasafirststep.

Then Skin, color like regions are segmented using elliptical model in YCbCr. After skin regionsarefound,theycombinedwithedgedimagestograyscaleimage.Finally,thecombined regions are verified as face by checking bounding box ratio and area insidethebounding box.Anotherapplicationsegmentsskinlikeregionwiththresholdvalue in Cb, Cr, Normalized r and Normalized g. Then candidate for face is chosenwith respect to bounding box ratio, ratio of area inside and area bounding box, andminimum area of the region. After candidates are found, then AdaBoosting method isapplied to find face candidates. The verification is done with combining both resultsfrom skin like region and AdaBoosting [10]. Also, skin color can be modeled inelliptical region in Cb and Cr channel in YCbCr color space. Skin like region issegmentedifthecolorvalueisinsideellipticregionandcandidateregionsareverified using template matching [11]. Peer et al. [12] detect faces using only skinsegmentationinYCbCrcolorspaceandresearchersgeneratetheskincolorconditions in RGB color space as well. Another approach for skin color modeling isdone by Self Organizing Map (SOM) Neural Network (NN). After skin segmentationis applied, each segment is taken as candidate and verified if they can fit into ellipticregion ornot [13].

Another significant information about the human face detection is pattern of humanface.Templatematchingcanbeappliedoverwindowscanningtechniqueorsegmented region. Scanning technique is applied with small size window like 20x20or 30x30 pixel window. This approach scans all over the original image, and thendecreases the image size with some iteration suitable to re-scanning. Decreasing thesize is important to locate the large or medium size faces. However, this requiresexcessivecomputationaltimetolocatefaces.Templatematchinginsegmentedregionreqiuresmuchlesscomputationaltimethanscanning,becauseitonlyconsidersthe matchingofsegmentedpart.Inliteraturemanyapplicationsareavailable by using template matching, i.e., [14], [15]. Chen et al. [14], use half-facetemplate, instead of full-face template. This method decreases computational time.Andthishalf-facecanbeadoptedtofaceorientations.Anotherapproachusesabstract templates, that are notimage like but composed of some parameters (i.e.size, shape, color, and position). Skin like regions are segmented with respect toYCbCr color space. Then, eye and eye pair abstract templates are applied to thesegmentedregion.Firsttemplateslocatetheregionofeyesandsecondtemplates locate the each eye. Second template also determines the orientation of eyes. ThenTexturetemplateis applied toverifythefacecandidateregion [15].

Image-Based methods use training/learning methods to make comparison betweenface and non-face images. For these methods, large number of images of face andnon-face should be trained to increase the accuracy of the system. AdaBoost [16],EigenFace [17 - 19], Neural Networks [20 - 25] and Support Vector Machines [26 -29] are kind of methods that are used commonly in face detection algorithms. Faceand non-face images are described in terms of wavelet feature in AdaBoost method.Principal Component Analysis (PCA) is used to generate the feature vector of faceand non-face image in EigenFace method. Also, PCA is used to compress the giveninformationvector.Kernelfunctioniscreatedtodescribefaceandnon-faceimagesin Support Vector Machines (SVM). Face and non-face images are also classified byartificialneuron structureinNeural Networks (NN).

AdaBoost is an algorithm that constructs strong classifier from weak classifiers. FacecandidatesarefoundbyapplyingAdaBoostalgorithm.Then,verificationisdonewith Cascade Classifier. This algorithm can handle of faces; left, left+45,front, andright+45,right pose.

Some performance statistics of algorithms are given in Appendix section.Detection/Recognitionratesshowtheperformanceofcorrectdetectionoffacesi nthe given image or recognition of given face image. Miss rate shows percentage ofmissed face in the given image. False rate gives the percentage of wrong detectedfaceorwrong classified face. Methodsforfacerecognitionsystemareinvestigatedandpossiblesolutionsarestudied extensively. Selected face detection method is skin segmentation and facecandidateisverifiedwitheyesandmouthfinding.Then,extractedfacesareclassified with FFNN. The detail about the face recognition system will be explainedinnext chapter.

CHAPTER3

DESIGN OF A FACE RECOGNITION SYSTEM

Research papers on Face recognition systems are studied and state of the currenttechnology are reviewed and summarized in the previous chapter, results of whichwillguideustodesignafacerecognitionsystemforfuturehumanoidand/orguide/guardrobot.Athroughoutsurveyhasrevealedthatvariousmethodsandcombinationofthesemethodscanbeappliedindevelopmentofanewfacerecognition system. Among the many possible approaches, we have decided to use acombination of knowledge-based methods for face detection part and neural networkapproach for face recognition part. The main reason in this selection is their smoothapplicability and reliability issues. Our face recognition system approach is given inFigure4.

3.1. INPUT PART:

Input part is prerequisite for face recognition system. Image acquisition operation isperformedinthispart.Livecapturedimagesareconvertedtodigitaldataforperforming image-processing computations. These captured images are sent to facedetectionalgorithm.

3.2. FACE DETECTION PART:

Facedetectionperformslocatingandextractingfaceimageoperationsforfacerecognitionsystem.Face detection partalgorithmis giveninFigure5.

Our experiments reveal that skin segmentation, as a first step for face detection,reducescomputationaltimeforsearchingwholeimage.Whilesegmentationisapplied, only segmented region is searched weather the segment includes any face ornot.

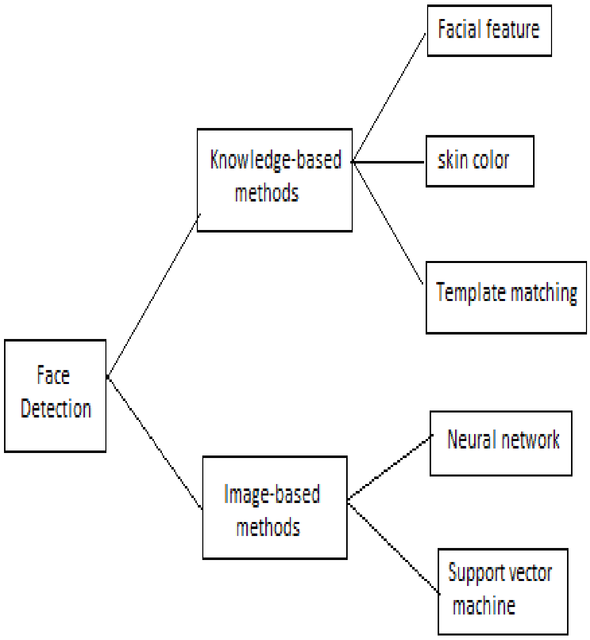


Figure3.1: AlgorithmofFaceDetectionPart

For this reason, skin segmentation is applied as a first step of detection part. RGBcolor space is used to describe skin like color [12], and also other color spaces areexaminedforskinlikecolors,i.e.HSV&YCbCr[54],HSV[55],andRGB&YCbCr&HSV[56].However,bestresultsgiveRGBcolorspaceskinsegmentation. The results of skin segmentation on different color spaces are given inthenext chapter. Skincolorlikepixel conditionsaregiven below [12]:

r>95	|r-g|>15
g>40	r>g
b>20	r>b

max(r,g,b)-min(r,g,b)>15

“r”, “g”, and “b” parameters are red, green and blue channel values of pixel. If theseseven conditions are satisfied, then pixel is said to be skin color and binary image iscreatedfrom satisfied pixels.

Whitebalanceofimagesdiffersduetochangeinlightingconditionsoftheenvironmentwhileacquiringimage.Thissituationcreatesnon-skinobjectsthatbelong to skin objects. Therefore, white balance of the acquired image should becorrected before segmenting it. The implemented white balance algorithm is givenbelow[57]:

- Calculateaveragevalueofredchannel(R_{av}),greenchannel(G_{av}),andbluechannel(B_{av})
- CalculateaveragegrayGray$_{av}$=(R_{av}+G_{av}+B_{av})/3
- Then,K_R=Gray$_{av}$/R_{av},K_G=Gray$_{av}$/G_{av},andK_B=Gray$_{av}$/B_{av}
- Generate new image (NewI) from original image (OrjI) byNew(R)=K_R*Orj(R),New(G)=K_G*Orj(G),andNew(B)=K_B*Orj(B)

White balance algorithm, as a brief, makes image hotter if image is cold, and makescolder if image is hot. If image appears as blue, then image is called as cold. If imageappears as red or orange, then image is called as hot. Lighting conditions in thecaptureareaarealwayschanging,duetochangeinsunlightdirection,indoorlighting, and other light reflections. Generally, taken pictures are hotter than theyshould be. Figure 6 shows hotter image that is taken in capture area and skin colorsegmentationto hotterimage, and whitebalancecorrectedimage.

If the image is not balanced, then some part of the wall will be taken as skin color asin Figure 6. Under some lighting conditions, acquired image can be colder. Then, thecolderimagewill bebalanced to hotterimage.

On the contrary, this process will generate unwanted skin color like regions. To getrid of this problem and create final skin image, logical “and operation” is applied onboth segmented originalimage and white balance corrected. This operation willeliminate change of color value due to change of lighting condition. Also, bad resultsof segmentation on uncorrected image and good results on corrected image are givenin Figure 7. In uncorrected image, distinguishing of face part is hard and face partseemsto bepart ofbackground.

a.)OriginalImage (OI)b.)Skin Segmentation on OI

n OI

C)WhiteBalanceCorrectiononOI(WBI) d.) Skin Segmentation on WBIFigure4

FIGURE 3.2: Exampleoftaken/whitebalancecorrectedimageandskincoloursegmentation

After“andoperation”isappliedonsegmentedimages,somemorphologicaloperations are applied on final skin image to search face candidate. Noisy like smallregions, that are less than 100 pixel square area, are eliminated. Then, morphologicalclosing operation is applied to merge gaps with 3-by-3 square structure. Applyingdilation operation and then applying erosion operation are considered as closingoperation. After these two morphological operations, face candidate regions can bedetermined. To select candidate, each 1’s are labeled.

On each label two conditionsare concerned to be face candidate. First condition is ratio of bounding box, whichcoversthelabel.Theratioofboundingbox,widthoverheight,shouldliebetween0.3 and 1.5. The limits determined experimentally. Lower limit is taken to be as lowas possible, to get facial part that include neck or some part of chest. Other conditionistocoversomegapsinsidetheregion.Thispropertywilldistinguishfacefromother body part, i.e. hand. Segmentation on hand will have no gap which makedifferentfrom face.

Figure 3.3:Results of Segmentation on Uncorrected (Left) and Corrected Image(Right)

Based on these conditions, face candidates are extracted frominput image withmodifiedboundingboxfromoriginalboundingbox.Asitismentioned,withlowering the bounding box limit, chest or neck could be included in the candidate,thoughchestorneckshouldbediscarded.Theheightofboundingboxmodified as1.28 times bigger than width of bounding box because width of face candidate doesnotchange if candidateincludeschest/neckornot.Thismodificationvalue havebeen determined experimentally. After this modification, new bounding box coversonly face. These face candidates will be sent to facial feature extraction part tovalidatethecandidates.

Face candidates are found after white balance correction, skin like color detection,morphologicaloperation,andfacecandidatesearching.Forfinalverificationof candidateandfaceimageextraction,facialfeatureextractionisapplied.Facialfeature is one of the most significant features of face. Facial features are eyebrows,eyes, mouth, nose, nose tip, cheek, etc. If some of the features can be found in thecandidate,thenthecandidatewillbeconsideredasface.Twoeyesandmouthgenerate isosceles triangle, and distance between eye to eye and mid point of eyesdistance to mouth is equal [5]. On the other hand, candidate facial feature should beextracted from face candidate image, because it is difficult to determine the features.Some filtering operations are applied to extract feature candidates and steps are listed below:

- LaplacianofGaussianFilteronRedchannelofcandidate
- Contrastcorrectionto improvevisibilityoffilterresult
- Averagefilteringtoeliminatesmallnoises
- Convertingto binaryimage
- Removingsmall noisesfrom binaryimages

- Instead of Laplacian of Gaussian (LoG) Filter, binary thresholding is applied inprevious application. Binary thresholding is sensitive to lighting. If shadow appearson some part of the face, some facial feature components can be eliminated. In sometrials, left eye has eliminated due to shadowing on left part of face. Also, beard onface can eliminate mouth while thresholding. Due to these problems, Sobel edgedetection method is tried to eliminate thresholding problem. Edge detection canreveal facial components better than thresholding and not sensitive to light change orshadows. On the other hand, edge detection gives higher response than LoG, and eyepart is not clear as LoG result. Due to this reasons, LoG filter is used to extract facialfeatures.
- Result of some filtering operations on face candidate and face candidate is given inFigure8.

a) Face Candidate Imageb.) Face Image After Filtering

Figure3.4: Result offiltering operationsonfacecandidate

Figure 8 shows that, facial features can be selected easily. Eyes, mouth line can beselected and with some operations, it may be feasible for computers as well. Afterobtaining filtered image, labeling operation is applied to determine which labels arepossible to be facial features. Then, filtered image is divided into three regions whichisillustrated in Figure9.

In Figure 9, R denotes right region, L denotes left region, and D denotes down regionofface.

Criteriacheckingareappliedoneachlabeltodetermineleftand righteyes.

Criteriaarelistedbelow:

1. Width denotes width of face candidate image and height denotes height offace candidateimage
2. ypositionofleft/righteyeshouldbelessthan0.5*height
3. xpositionofrighteyeshouldbeinregion of0.125*widthto0.405*width
4. xpositionoflefteyeshouldbeinregion of0.585*widthto0.875*width
5. Areashouldbegreaterthan100 pixelsquare
6. BoundingBoxratiooflabel shouldbein theregion of1.2to 4

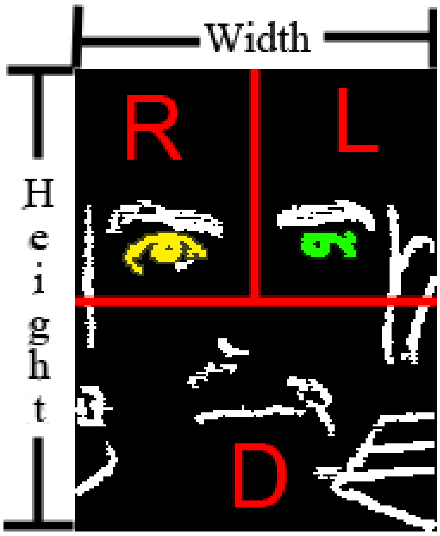


Figure3.5:RegionsofFilteredImage

Ifalabelisin regionRand satisfiescriteria2,3,5 and 6,then itissaid to becandidate to right eye. Yellow label is a right eye candidate (Figure 9). If a label is inregion L and satisfies 2,4,5 and 6, then it is said to be candidate to left eye. Greenlabel is a left eye candidate (Figure 9). A right eye candidate is said to be right eye ifits distance to center point of image is minimum in all right eye candidates. Also, aleft eye candidate is said to be left eye if its

distance to center point of image isminimum in all left eye candidates. Left and right eye are mostly found correctly butsometimes bottom eyelid is found falsely. If left and right eyes are detected, thenmouthfindingapplication can beapplied.

Each label inside down region chooses as mouth candidate and candidate propertyvectoriscalculated.Euclidiandistanceofrighteyetomouthcandidate(right-distance) and Euclidian distance of left eye to mouth candidate (left-distance) arecalculated. Also, Euclidian distance between two eyes (eye-distance) and Euclidiandistancebetweenmidpointofeyestomouthcandidate(center-distance)arecalculated.Then, propertyvectoris created byusingthedistances. Propertyvector:

- Labelnumberofthemouth candidate
- Absolutedifferencebetweenleft-distanceandright-distance(error1)
- Absolutedifferencebetweeneye-distanceandcenter-distance(error2)
- Summationoferror1and error2(error-sum)

If error1 and error2 are smaller than 0.25*eye-distance, then candidate is possibly amouth. Minimum error-sum inside possible mouths is considered as mouth. Requiredfacial features are found which are right eye, left eye and mouth. Face image can beextractedwhichcoverstwoeyesandmouth.Facecoveringiscreatedwitharectanglein which cornerpositions are;

- Rightupcorner:0.3*eye-distanceupandleftfromrighteyelabelcentroid,
- Leftupcorner:0.3*eye-distanceupandrightfromlefteyelabelcentroid,
- Right down corner: 0.3*eye-distance from left from right eye label
- centroidanddown from mouth label centroid,
- Left downcorner: 0.3*eye-distancefromright from lefteye labelcentroidanddown from mouth label centroid,

After face cover corner points are calculated, face image can be extracted. Facialfeatureextraction,coveringandfaceimage extraction are givenin Figure10.

Figure3.6: FacialFeatureExtractions(Left)andFaceImage (Right)fortheauthor

Up to here, face detection partis completed, and face images are found in theacquired images. This algorithm is implemented using MATLAB and tested for morethanhundredimages.Thisalgorithmdetectsnotonlyonefacebutalsomorethanone face.Small amountof orientedface are acceptable. Results are satisfactory forallpurpose.

3.3 FACERECOGNITIONPART

Modified face image which is obtained in the Face recognition system, should to beclassified to identify the person in the database.This is face recognition part of aFace Recognition System. Face recognition part is composed of preprocessing faceimage, vectorizing image matrix, database generation, and then classification. Theclassification is achieved by using FeedForward Neural Network (FFNN) [39]. Facerecognitionpart algorithm is given.

Beforeclassifyingthefaceimage,itshouldbepreprocessed.Preprocessingoperations are histogram equalizing of grayscale face image, resizing to 30-by-30pixels, and finally vectorizing the matrix image. Histogram equalizing is used forcontrast adjustment. After histogram equalization is applied, input face image issimilar to faces in database. Input face image has a resolution about 110-by-130pixels which is large for computation of classifier. So, dimension reduction is madewithresizing imagesto30-by-30pixelsimagetoreducecomputationaltimeinclassification. After resizing, image matrix should be converted to vector becauseclassifier does not work with two-dimensional input. Input vector size will be 900-by-1vectorto classifier.

Neural Network is used to classify given images. Neural Network is a mathematicalmodelthatisinspiredfrombiologicalneuralnetworksystem.Neuralnetworkconsists of neurons, weights, inputs and output. A simple neuron model is given inFigure 12. Inside neuron, summation (∑) and activation function (f) operations areapplied. ‘x’ denotesinput of neuron, ‘w’ denotes weight of input, ‘I’ denotes outputof summation operation, and ‘y’ denotes output of neuron or output of activationfunction. Equations of I and y is given in Eq.1 and Eq.2. Network structure may bemultilayered(Figure13).

$$I= \sum(x_1*w_1+x_2*w_2+\ldots+x_n*w_n) \quad (3.1)$$

$$y= f(I) \quad (3.2)$$

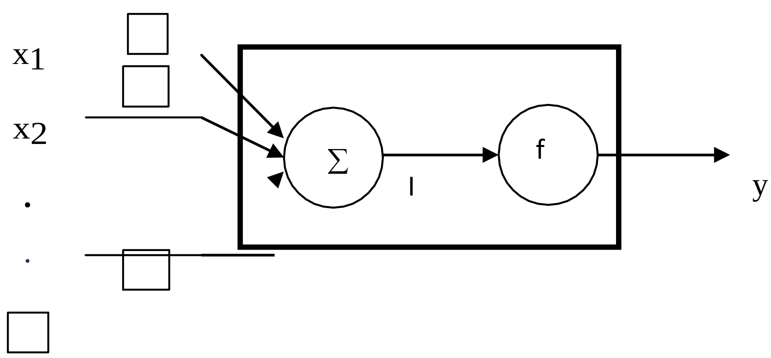


Figure3.7: NeuronModel

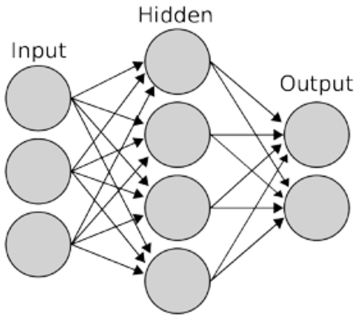


Figure3.8: MultilayerNetworkstructure

Output value range shows a difference with respect to selected activation function.Common activation functions are threshold, linear and sigmoid functions, and showninFigure14.

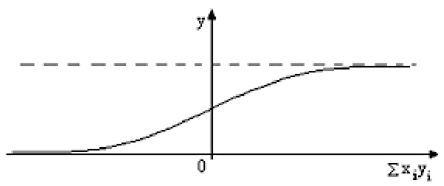


Figure3.9: (Activation Functions – Sigmoid)

Also,many differenttypeofnetworkstructuresexistinliterature.Inclassifier,FeedForward Neural Network (FFNN) is used. FFNN is the simplest structure in theneural network. Figure 13 is a kind of FFNN. Information flows through input tooutput and does not perform any loop or cycle operations. Two-layer with sigmoidtransfer function FeedForward Neural Network is used for classification operation.This type of network structure is generally used for pattern recognition applications.Systemnetwork properties are:inputlayer has 900 inputs, hidden layer has 41neurons and output layer has 26 neurons. Output layer has 26 neuron since thenumberofpeoplein databaseis 26.

After structure is generated, then network should be trained to classify the givenimages with respect to face database. Therefore, face database is created before anytests. A database is created for 26 people with 4 samples for each person. This results104 training sample. Due to that, 900-by-104 size matrix will be training matrix.Training matrix vector element is arranged with four groups due to the number ofsamples for each person. Though, first 26 vector element belongs to first samples of26 people, and it continues. Training matrix's columns are made from preprocessingimage and then vectorizing to face image which generate database. Then, targetmatrix is generated to inform network vectors belonging to persons. Target vector iscreated with putting '1' with respect to order number of the name in database andother elements '0' for target vector. Due to vector arrangement of training matrix,target matrix is combination of 4 horizontally concatenated identity matrix with sizeof26.

Aftertrainingmatrixandtargetmatrixiscreated,thentrainingofNNcanbeperformed.

Training means that configuring the values of the weight to make bestrelationship between training matrix and target matrix. When weights are configuredwell, classification of the given new face image will be classified correctly. Backpropagation is used to train the network. Back propagation has two phases, that arepropagation and weight update. In propagation phase, first flowing input throughoutput to see result. Then, going back output to input with calculating errors for eachweights. After that, weights are updated with respect to their error values. Trainingperformance andgoalerrorsaresetto1e-17 toclassifygivenimage correctly.

When training is completed, network can be used to classify new faces that are fedfromfacedetectionpart.Thepersonnameselectionisbasedonoutputofthenetwork.

The row, which has maximum value in output vector, is matched with orderofthepeoplein databaseand nameis printed.

3.4.OUTPUT PART

This part is final step of face recognition system. Person name is determined withrespect to output of face recognition. Output vector of neural network is used toidentify person name. The row number which has maximum value is match withsamerownumberin thedatabasenameorder.

3.5.CHAPTER SUMMARY AND DESCRIPTION

Face recognition system has four main steps, which are input, detection, recognition. and output Input performs image acquisition part, which converts live capturedimage to digital image data. Detection part composed of white balance correction toacquired image, skin like region segmentation, facial feature extraction, and faceimage extraction. White balance correction is an important step to eliminate colorchages of acquired imade due to illumination conditions change. Skin like regionsegmentationperformancecanbeimprovedwithintegratingwhitebalancecorrection before segmenting. Skin color like region segmentation decreases searchtime for possible face region since only segmented regions are considered as regionmay contain face. Facial feature extraction is important to extract face image whichwill be standard face image. LoG filter gives best results to extract facial featureswith respect to black and white convertion. Facial features are found with property oftwoeyes andamouth creates isosceles triangle.

Face image is extracted based on facial features positions, and image preprocessoperation is performed to eliminated illuminate changes and prepared to be input toclassifier. Classifier is a key point of recognition part. FFNN is good at patternrecognition problem. Decreasing gradient value of performance, increase accuracy ofclassification.Outputofnetworkdeterminespersonidentity.Eachpersonhasidentity number andthe row with a maximum value in the output vector matcheswithidentitynumber.Thismatchingestablishconnectionbetweenoutputofclassificationand personnames.

This face recognition system algorithm will perform fast and accurate person nameidentification. Performance of skin segmentation is improved with white balancecorrection and facial feature extraction performance is improved with LoG filter withrespect to Lin’s implementation [5]. Accuracy of classification is improved withdecreasinggradient valueofperformancevalue.

CHAPTER-4

EXPERIMENTS AND RESULTS

A complete hardware and software system is designed and implemented in the RobotVision Laboratory of the Department of Mechatronics Engineering at the AtılımUniversity. The ultimate goal of the larger project (umbrella project) is to develop ahumanoid robot with a narrower application like Guide robot, Guard robot, Officerobot, etc. The developed system has been tested for many live acquired images andresults are satisfactory for such a pioneering work in the department. Improvementsare required for better performance. System description and possible improvementsarediscussed in this chapter.

4.1.SYSTEM HARDWARE:

Systemhasthreemainhardwareparts.Theyarecomputer,framegrabber,andcamera. Computer is brain of system, which processes acquired image, analyzesimageanddeterminesthepersonsname.Thecomputerusedinthetestisatypic alPCwith thefollowingspecifications:

- IntelCore2Duo3.0GHz
- 3.0GbRAM
- OnBoardGraphicCard

Sony EVI-D100P camera is used in Face recognition system (Figure 15). Camera hashigh quality CCD sensor with remote Pan/Tilt/Zoom (PTZ) operations. Camera has10x optic and 4x digital zoom, so totally 40x zoom capability. Optically, it has 3.1mm wide angle to 31 mm tele angle focal length and 1.8 to 2.9 minimum aperturecapacities. Resolution of CCD sensor is 752x582 pixel. Pan/Tilt capacity is ±100° forpan and ±25° for tilt operations. Camera has RS232 serial communication. Camerasettings and PTZ operation can be performed via serial communication. Camerasettings are shutter time, aperture value, white balance selection, etc. Camera videooutputsignalsareS-VideoandCompositesignals.Compositevideosignalisused.

Figure-4.1: SonyEVI-D100p

Image acquisition from camera is performed by frame grabber. Imagenation PXC200A frame grabber from CyberOptics is used (Figure 16). This grabber is connectedtocomputerviaPCIconnection.Upto4compositeand1s-videosignalstypecamera can be connected. YCbCr, RGB and Monochrome color channel can beselected, and in our system RGB color channel is used. Resolution of grabber is up to768x576 pixel.

Figure-4.2: PXC200AFrameGrabber

4.2.SYSTEM SOFTWARE:

Algorithm of system is implemented on MATLAB R2011a software. MATLAB is aproductionofMathWorksCo.andcanperformalgorithmdevelopment,datavisualization,dataanalysis,andnumericcomputationwithtraditionalprogramming language i.e. C. Signal processing, image processing, controller design, mathematicalcomputation, etc. may be implemented easily with MATLAB that includes manytoolboxeswhichsimplifiesgenerationofalgorithmmorepowerfully.ImageAcquisition Toolbox, Image Processing Toolbox, and Neural Network Toolbox areusedwhilegeneratingalgorithm ofFacerecognition system.

Image Acquisition Toolbox enables image acquiring from frame grabber or otherimaging system that MATLAB supports. This toolbox supports acquiring resolutionof frame grabber, triggering specification, color space, number of acquired image attriggering, region of interest while acquiring, etc. This toolbox will bridge betweenframe grabberand MATLABenvironment.

ImageProcessingToolboxprovidesmanyreferencealgorithms,graphicaltools,analysis, etc. Reference algorithm provides fast development of algorithms.

Filters,transforms, enhancements, etc. are ready to use functions which simplify to codegeneration. This toolbox is used in face detection and some part of face recognitionsections.

NeuralNetworkToolboxprovidesdesigning,implementing,visualizing,andsimulating neuralnetworks.Patternrecognition,clustering,datafittingtoolsaresupported.Supervisedlearningnetworks,i.e.FeedForward,RadialBasis,TimeDelay,etc.,andunsupervised learningnetworks,i.e.SelfOrganizingMap,Competitive Layer, are also supported. For classifying of face images are performedbyPatternRecognitionToolwithtwolayer,sigmoid,FeedForwardNeuralNetwork.

4.3FACE DETECTION:

First implementation of system is performed on detection of faces in acquired image.Therefore,facedetectionhasstartedwithskinlikeregionsegmentation.Manymethods have been tried to select which segmentation algorithm works best on ourimage acquisition area. Based on RGB [12], HSV&YCbCr [54], HSV [55], andRGB&YCbCr&HSV[56]colorchannelsskinlikesegmentationaretestedonacquiredimagesandbestresultsaretakenfromRGBcolorspace.RGB&YCbCr&HSV are not performed well, based on our acquired images. Resultsofperformed skin likesegmentation aregiveninFigure17-19.

Figure4.3:OriginalImage (Left) &RGBSkinSegmentation(Right)

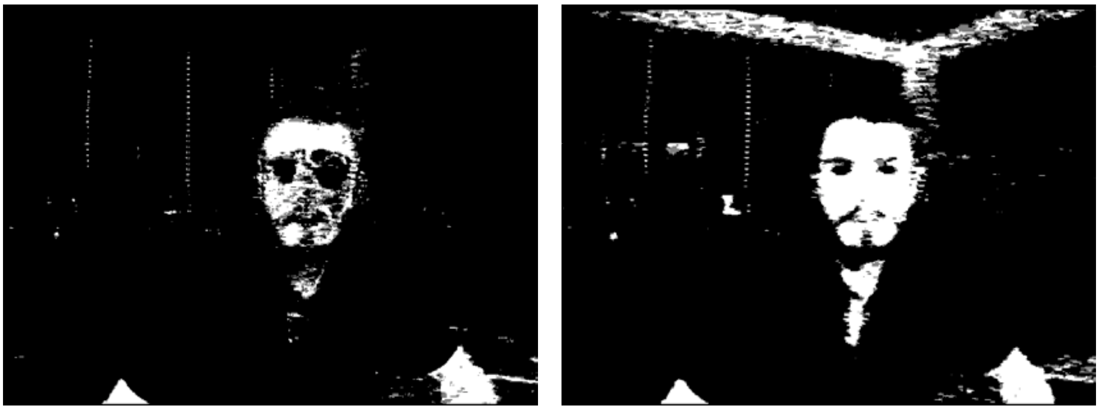


Figure 4.4: Skin Segmentation on Original Image with HS (Left) and CbCr Channel(Right)

Figure4.5:SkinSegmentationonOriginalimagewithHCbCrCombinations

Besides RGB gives the best result, colors of wall inside laboratory can be skin likecolor due to white balance value of camera. Unwanted skin like color regions canaffectdetectionanddistortfaceshape.Thiscolorproblemcanbeeliminatedbywhite balance correction of acquired image. The implementation of white balancecorrection is given Figure 20. Wardrobe color is white in real (Figure 20). On theother hand, color in acquired image (left image) is cream and also wall color lookslikeskincoloranditaffectssegmentationresults.Figure21showstheresultsof

segmentation on acquired image and white balance corrected image. Results showsthatwhitebalance correction should beapplied afterimageis acquired.

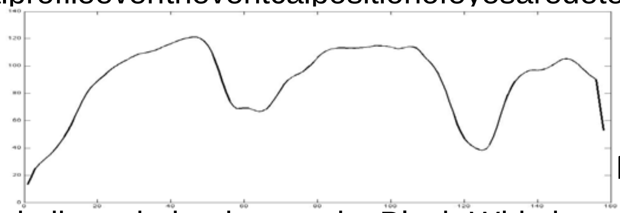
Figure4.6: AnImageWithout(Left)and With (Right)WhiteBalanceCorrection

Figure4.7: SkinSegmentationResultsonAcquired(Left) &Corrected Image (Right)

To guarantee color correction both segmentation are performed on acquired andcorrectedimage,thenlogical“andoperation”isperformed.Thereasonisthatcorrection operation make hotter image if image is cold and make colder if image ishot. When image is hotter, color of objects in the laboratory becomes similar to skincolor, i.e. wall, floor, wardrobe. After segmentation is performed, morphologicaloperations and candidate search is achieved as described in the previous chapter.Selection of face candidate followed by facial feature extraction and face verificationofcandidate.

Facial feature extraction is one of the most important parts of face detection sectionbecause this part makes bridging between detection and recognition. First trial onextractionismadewithprofileextractionoffacecandidate.Verticalprofileofcandidate is performed by taking mean of each row of image matrix. Then, localminimum shows possible positions of eyebrow, eye, nose tip, mouth, and chin. Aftereyepositionisdeterminedinverticalprofile,thenhorizontalprofileisextractedto determine eyes positions.

Vertical and horizontal profiles of four test face images aregiven in Figure 22 - 25. Determination of exact position of eye position and mouthposition is difficult to determine in vertical profile. Also, it is difficult to determinepositionofeyesinhorizontalprofileeventheveritcalpositionofeyesaredetermi nedin vertical profile.



Due to difficulty in determination files infacecandidate,faceprofileextractionisdiscardedandconvertingBlack-Whiteimage to find facial feature is performed. Some experiments are performed andresultsaregiven inFigure26 and 27.

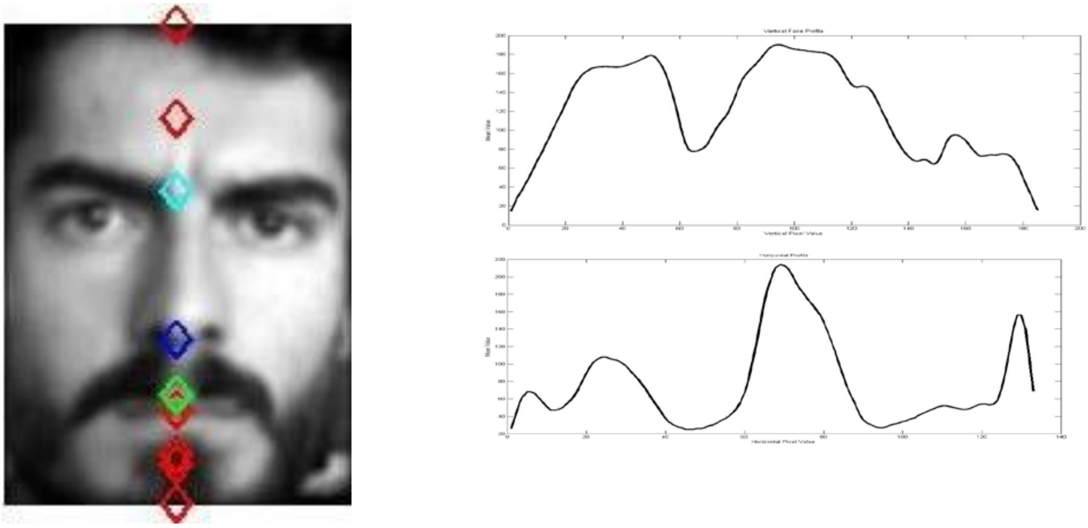


Figure 4.8: Test Image 1 (Left) & Vertical (Right-Top) - Horizontal (Right-Bottom)

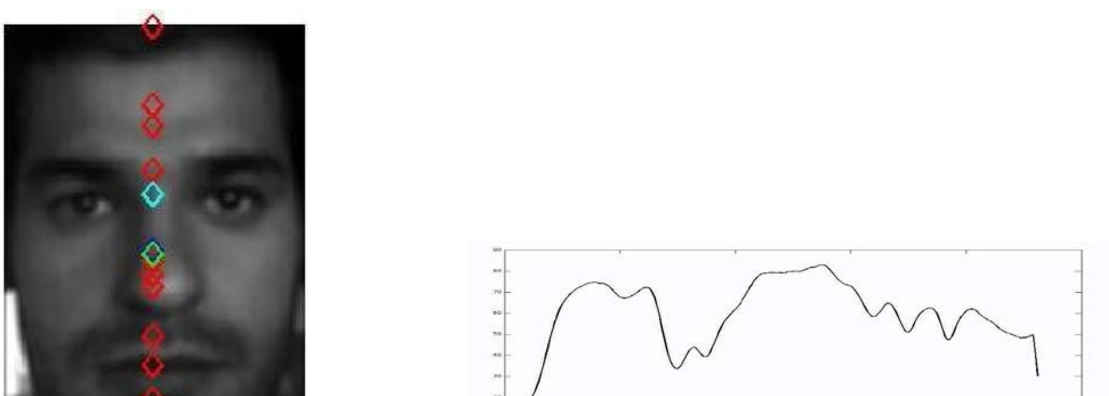


Figure 4.9: Test Image 2 (Left) & Vertical (Right-Top) - Horizontal (Right-Bottom)Profiles

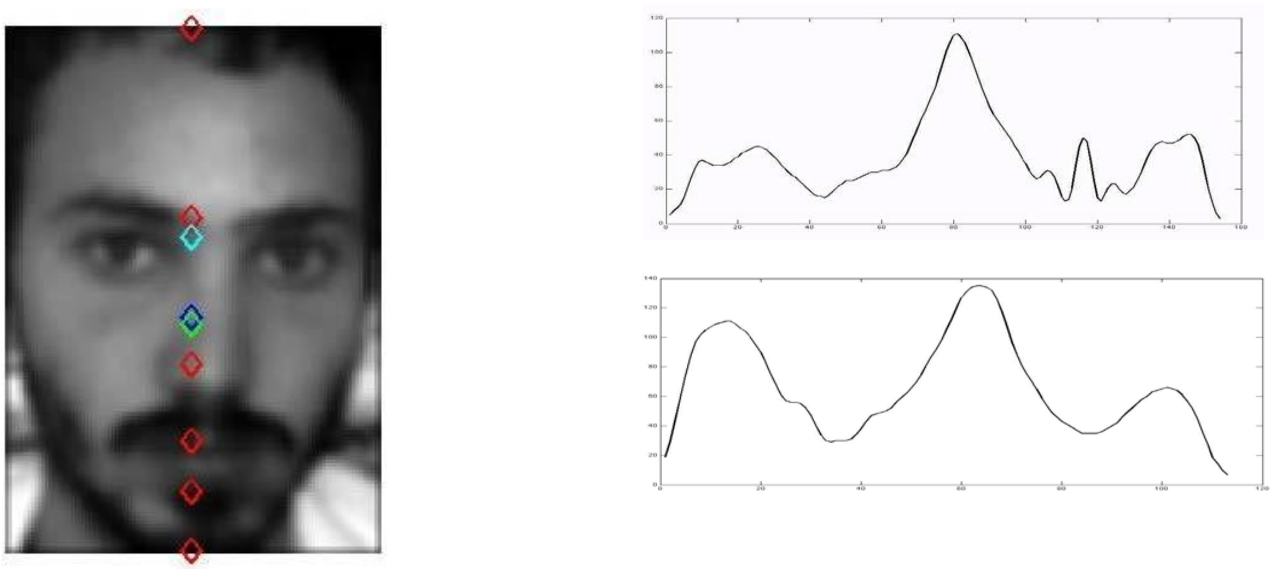


Figure 4.10: Test Image 3 (Left) & Vertical (Right-Top) - Horizontal (Right-Bottom)Profiles

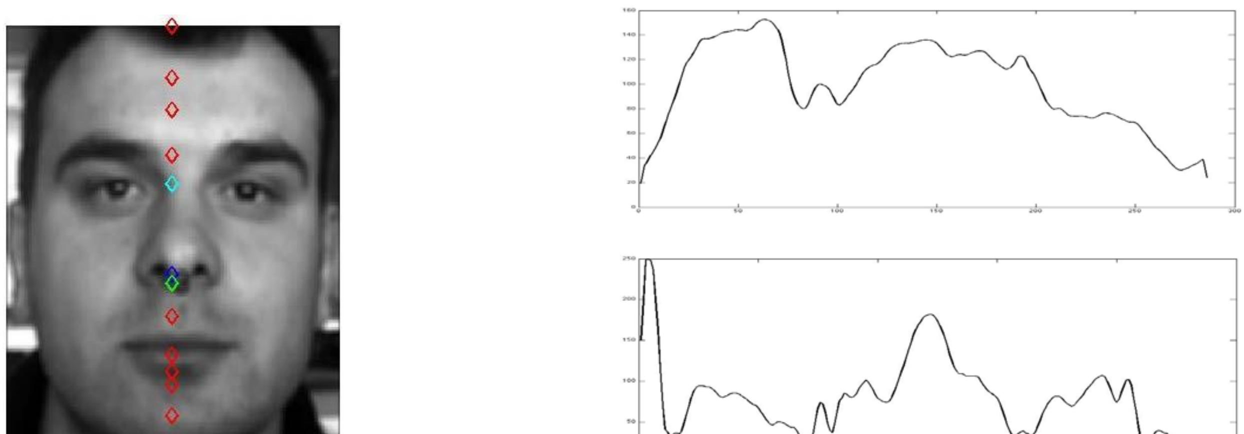


Figure 4.11: Test Image 4 (Left) & Vertical (Right-Top) - Vertical (Right-Bottom)Profile

Figure 4.12:TestImage5(Left)&Black-WhiteConversiononthe5(Right)

Figure4.13: TestImage6(Left)&Black-WhiteConversiononthe6(Right)

RighteyeisisolatedbutlefteyeiscombinedwitheyebrowinFigure26.Also,mouth is nearly erased. On the other hand, right eye and mouth are combined withbackgroundinFigure27.Thatmakesdifficulttofindeyeandmouth.Thecombination problem is due to lighting conditions while acquiring the image. Since,Black-White conversion is sensitive to light condition/changes, this approaches cannot be applicable easily. So, approaches that are not much sensitive to light shouldbeprefered.

Edge detection methods can be applicable on this problem because they are nearlyinsensitive to light change. Sobel edge detector is used to extract features. Figure 28showsresults ofedgedetection on test image5and 6.

Figure4.14: EdgeDetectionResultsonTestImage 5(Left)and6(Right)

Results show that, edge detection is not sensitive to light condition as Black-Whiteconversion. On both images, eyes and mouth can be selected with human eyes butmouth can be difficult to extract on the images and eye parts also vary on shapes.Also,edgedetection hashigh responses.

In order to use edge detection, Laplacian of Gaussian (LoG) filter can be used. LoGfilter has low responses than edge detection. Itmakes usefull enhancements on facialfeatures.Figure29shows resultsofLoGfilterontest image5and 6.

Figure 4.15 LaplacianofGaussianFilterResultsonTestImage5(Left)and6(Right)

Results of LoG filter are better than previous three trials. Mouth is more significantthanothers and eyes canbeselected moreaccurately.

4.4FACE RECOGNITION:

With addition of isosceles triangle approach, as described in the previous chapter,eyesandmoutharefoundandcroppedfaceimage.Then,databaseoffaceimag escan be generated. Database is generated from 26 people with 4 samples image foreach person. Database is created from face detection part. Sample images of 26peoplearegiven in Figure30.

Figure 4.16:26ParticipantsFaceImagesFromDatabase.

Name of the Participant are: Natasha, Thor, Bruce, Tony, Steve, Clint, Lizzy, Nick, Clint, Tom, Peter, Harry, Nick, Ari, Jessy, Eric, Damon, Klaus, Stefan, Enzo, Mathew, Loki, Tessa, Mike, Dustin, William, Loki, Hari, Henry, Robert, Christopher, Scarlet. While generating database, four different sample images are stored for each persons.Thereasonisthatacquisitionoffaceimagemaydiffereachtimetheimagetaken. For example, shaved and no-shaved faces are included for my samples (Left image inFigure). Also, differentcaptured face framesare added (Rightimagein Figure).

Figure4.17DifferentFaceSamples

900x104 size training matrix is generated to train neural network which will be usedtoclassify givennewfaceimage.PatternRecognitionToolinNeuralNetworkToolboxisusedtogen erateandtrainneuralnetwork.Thegeneratednetworkconsists of 2 layers with sigmoid transfer function. 2 layers are hidden layer andoutput layer. Output layer has 26 neurons due to number of persons in the facedatabase. Hidden layer number is an approach that is applied in [22]. The approachproposes that to guess initial neuron number, use Eq. 4.1, then train with this neuronnumber and record training time. After that, increase neuron number until trainingtime remain constant. When starting point of remaining constant, this will be numberofhidden neurons in thenetwork.

$n = \log_2 N$

Nisnumberofinputlayer,andnisthenumberofneuronsinhiddenlayer.Theinitial guess is 9.81 for 900 inputs. So, start initially as 10. Figure 32 shows graph ofnumber of neurons vs. training time. The graphshows that at 41 neurons trainingtime is 4 s and after this neuron number training time remain constant in 5 seconds.Therefore, 41 neurons are used in our system. Databasing and training of networkcodeis given in App 2.Also,performanceofclassificationisaffectedbytrainingparameterwhichisgradient value. Gradient value is related with error of target value and output value.Tests show that aminimum gradient valueresults in more accurate classification.The low gradient value causes false selection in the database. The comparison ofgradient values for errors 1e-6 and 1e-17 for the same input image is given TestImage2 (Table1).

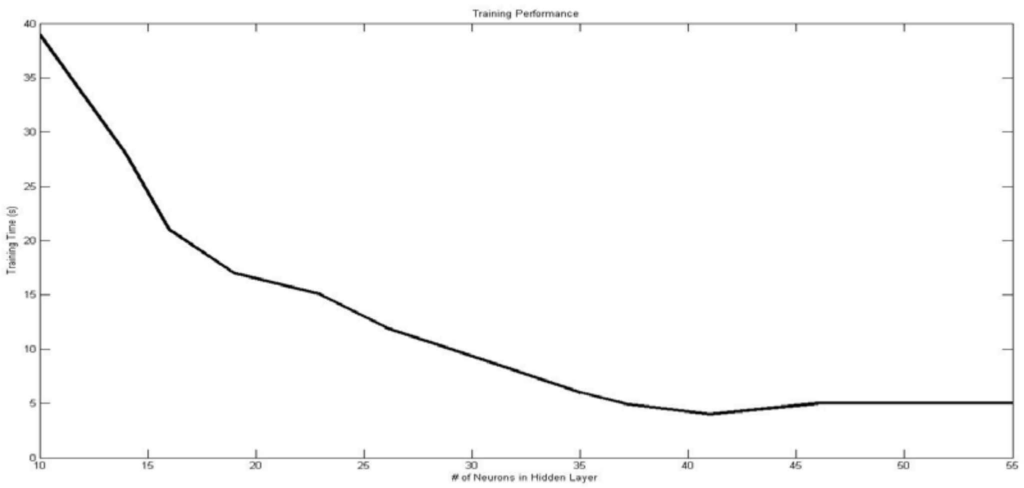


Figure4.18:NumberofNeurons vs.TrainingTime

Table1showsthatminimumgradientvalueshouldbeconsideredinsystemtogetmore accurateresults.

Table4.1GradientValueEffectsonClassification

1e-6	**1e-17**
0.0076	0.0000
0.0041	0.0000
98.0832	99.99 25
0.6032	0.0049
0.0040	0.0000

0.0989	0.0000
0.0000	0.0000
0.0766	0.0000
0.0000	0.0000
0.0799	0.0000
0.6118	0.0000
0.0015	0.0000
0.0001	0.0000
0.0968	0.0000
0.0530	0.0000
0.0021	0.0000
0.0009	0.0000
0.2094	0.0000
0.1310	0.0000
12.4808	0.1210
0.2932	0.0000
0.0164	0.0000
0.7166	0.0000
0.0006	0.0000
0.0033	0.0000
6.9192	0.0004

4.5FACE RECOGNITION SYSTEM:

Finally,facedetectionandrecognitionpartsaremergedtoimplementfacerecognition system. System can also handle more than one faces in the acquiredimage. Code is generated on MATLAB environment and given in App. 3 Results areshownin Figure33 -43.

Fiveskinlikeregionsareextractedandlabeled.Label4&5istakenasfacecandidate. Facial feature extraction operation is performed and eyes and mouth arefound, then faces are validated. Validated faces are classified, output results are; firstface belongs to Ayça and second face belongs to Cahit. Output result of system givescorrect results. Experiment and results are shown that algorithm can find multiplefaces in the acquired image and classify correctly. This results important, since somemethodscan onlydetectonefaceinagiven image.

Fourskinlikeregionsarefoundinacquiredimage2.Thirdlabelisconsideredasface candidate. After LoG filter is applied, eyelashes appear clearly. Then, algorithmconsiders eyelashes are eyes. Extracted face image classified correctly. Above resultsshow thatalgorithmcan recognizewheneyesareclosedwith 99.0498networkoutput.

Thistime,ifpersonstandsfarfromcamera,cansystemdetectandrecognizecorrectly or not is tested. Five skin like regions are labeled. Then, label three is takenas face candidate. Taking height of face candidate as 1.28 times bigger than widtheliminates neck of person. Since, only face part is considered as face candidate. LoGfilter performs well to extract facial feature regions. Due to low resolution, both leftand right, eye and eyebrow are merged but centroids of merged region do not affectthe results. Low resolution of extracted face image is classified correctly. Aboveresultshowthatfaceimage can berecognizedcorrectlyevenresolution isnotlarge.

Face image is identified if maximum network output value is greater than 90 percent.Classification result for right image (Figure 40) is maximum 52.7422 because thisfaceisnotinthefacedatabase.Therefore,algorithmsaysthat'Personisnotrecognized'.

Since thisface isnotinthe face database,networkresultismaximum1.3872.Therefore,answerofalgorithmis 'Person isnot recognized'.

Manyexperimentsareperformedonliveacquiredimages.Facedetectionandrecognition parts performed well. Skin segmentation both decrease computationaltime and search area for face. Experiments show that connection is established wellbetweendetectionandrecognitionparts.Thenetworkcancorrectlyclassifywhen eye/eyes are closed, eyebrows are moved and face is smiled or showed teeth. Also,number of people in database can be increased and most probably will correctlyclassifyfaces.

4.6LIMITATIONS OF THE SYSTEM:

Somelimitationsofthedesignedsystemaredeterminedafterthe experiments:

Skincolorlikeclothes:Skin color like clothes are segmented at skin segmentationstagebutitwillaffectresultsoffacecandidate.Mostofexperimentswithskincolorlikeclothesshowthat,faceandclothsegmentsaremerged.Thesoftwaredoesnotrecognizethemastwoseparatebodyregions.Thus,facialfeatureextractionoperation can be resulted, as candidate is not a face or wrong face image extraction.

Presenceofobjectonface:Glassesonthefacemayaffectfacialfeatureextractionresults.Ifglasseshavereflection,LoGfiltercannotperformwelloneyeregion.Also,sunglasseswillcovereyeregionandeyescannotbedetectedbasedontheproposedalgorithm.

Contrast of face candidate image:Contrast value of image affects results of filter.Less contrast image has fewer edges on the image. Thus, facial component could notbe visible after filtered. Therefore, face image can not be extracted if candidatecontainsface.

Systemworkingrange:Systemcandetectandrecognizethepersonifpersonstandingrangeisbetween50cmto180cm.Thisrangeisacquiredwithpropertyof3.1 mm focal length and 768x576 pixels resolution. Thus, working range can bechangedbycameraproperty.

Skin color range:RGB skin color segmentation work well in the range of light tonetodark tone.

Headpose:Frontalheadposescanbedetectedandextractedcorrectly.Smallamountof roll andyawrotation canbeacceptableforsystem.

CHAPTER 5
CONCLUSION AND DISCUSSION

5.1.DISCUSSION

This thesis study focuses on designing and implementing of a face recognition system.System is composed of image acquisition, face detection part, face recognition part,andidentification ofperson name.

Image acquisition is prerequisite for system. Frame grabber is used to capture framefrom video camera device, and digitalize it to be processed in the algorithm. VideocameradeviceisSONYEVI-100Pwhichhascapabilityofpan,tiltandzoom.Therefore,livecapturedimageshavedifferentillumination,background,whitebalance difference, and position and size of human face. Image acquisition process isperformedbyMATLAB software withImageAcquisitionToolbox.

Knowledge-Based face detection method is performed for face detection part withcombination of skin color and facial feature methods. Methods are combined due todecreasing computational time and increasing accuracy of detection part. Before skincolor segmentation white balance correction is performed to overcome color problemwhileacquiringimage.Colorproblemaffectsskincolorsegmentation,becauseirrelevantobjects can haveskin colorvalue.

RGB skin color segmentation is performed in the algorithm. Many methods useYCbCrcolorspace.Also,YCbCrandHSVcolorspacesaretestedbutbestresultsareobtainedwithRGBcolorspace.RGBcolorspaceworkswellatindoorconditions but performance is not tested for outdoor condition. If modelling of skincolor is performed with statistical model, skin color segmentation may be moreaccurate. Segmentationis performed on bothacquired image andwhite balancecorrected in acquired image. Then, logical “and operation” is performed on bothsegmentedimages to reducecolorproblem.

Face candidates are chosen from segments and facial feature extraction operation isperformed to verify face candidate and extract face image. LoG filter is performed toshowfacialcomponentsclearly.BeforeLoGfilterisperformed,black-whiteconversion and edge detection are performed. Black-white conversion is sensitive tolight changes and some components can be eliminated due to shadowing on face. Onthe other hand,edge detection is not sensitive to light changes but shapes are notclear as LoG filter. Facial components can be selected clearly with eyes after appliedLoG filter. Two eyes and mouth are found with property oftwo eyes and a mouthcreate isosceles triangle. Eyes and mouth are found with this property and face imageis extracted based on positions of facial components. On the other hand, componentsarefound byestimation but can befound moreaccurately.

Withextractionoffacialcomponents,facedetectionpartiscompletedandfaceimage is ready to be classified. Before sending to classifier, histogram equalization,resizing and vectorizing operations are performed. Histogram equalization is appliedto eliminate light changes on image and equalizing contrast of image. Finally, faceimageisreadytobeclassified.ClassificationisperformedbytwolayersFeedForward Neural Network. Sigmoid function is used for activation function inneurons. This type of network structure and activation function are good at patternrecognition problem since face recognition is a kind of pattern recognition problem.Inthehiddenlayer,41neuronsareusedandtrainingtimevs.numberofneurongraph is given in Figure 32. Best performance is achieved by 41 neurons. Outputlayerneuron numberis determined bynumberofpeoplein thedatabase.

Output of network gives classification result. The row with a maximum value givesorder number of names in database. Classification result is affected by performancevalue of network. The minimum gradient value gives more accurate result. Gradientvalue for system while training is taken as 1e-17. Performance with lower gradientvalueis given in Table1.

Algorithm is developed on MATLAB environment and it gives capability to detectmultiplefacesinacquiredimage.Personnamingisachievedwhenmaximu mvalueof row is greather than 90%. If it is lower, output is “Person is not recognized”. Thesystemhasacceptableperformancetorecognizefaces withinintendedlimits.

5.2.CONCLUSION

Face recognition systems are part of facial image processing applications and theirsignificance as a research area are increasing recently. Implementations of system arecrimeprevention,videosurveillance,personverification,andsimilarsecuritya ctivities.Thefacerecognitionsystemimplementationwillbepartofhumanoidrobot project at Atılım University.

Main goal of thesis is to design and implement face recognition system in the RobotVisionLaboratory oftheDepartmentofMechatronicsEngineering.Thegoalisreached by face detection and recognition methods. Knowledge-Based face detectionmethods are used to find, locate and extract faces in acquired images. Implementedmethodsareskincolorandfacialfeatures.Neuralnetworkisusedforf acerecognition.

RGB color space is used to specify skin color values, and segmentation decreasessearching time of face images. Facial components on face candidates are appearedwith implementation of LoG filter. LoG filter shows good performance on extractingfacialcomponents underdifferent illuminationconditions.

FFNNisperformedtoclassify tosolvepatternrecognitionproblemsincefacerecognitionisakindofpatternrecog nition.Classificationresultisaccurate.Classificationisalsoflexibleandcorrectwh enextractedfaceimageissmalloriented,closedeye,andsmallsmiled.Proposedal gorithmiscapableof detectmultiplefaces, andperformanceofsystem hasacceptablegood results.

Proposedsystemcanbeaffectedbypose,presenceorabsenceofstructuralcomp onents,facialexpression, imagingcondition, andstrongillumination.

CHAPTER 6

FUTURE WORKS

Face recognition system is designed, implemented and tested. Test results show thatsystemhasacceptableperformance.Ontheotherhand,systemhassomefuturewor ksforimprovementsand implementation onhumanoid robot project.

Future works will be stated in the order of algorithm. First future work can beapplied on camera device to improve imaging conditions. Sony camera that is used inthesis, can communicate with computer. Camera configurations can be changed viacomputer and these changescan improve imaging conditions. Exposure values canbe fixed to capture all frames with same brightness value / similar histogram. Also,fixing white balance value can improve performance of skin segmentation which willlead to eliminate non-skin objects. Maybe, white balance correction section may notbe needed any more. For later implementations, pan, tilt and zoom actuators can becontrolled.Cameraiscontrolled viaremotecontrollerinthetestofthesis.

Then skin color modelling can be improved. In the thesis work, some conditions areused to describe skin color. On the other hand, broad skin color modelling can beachieved by use of statistical modelling. Dark skins, skins under shadow or brightlight can be modelled and more skin region segmentation can be achieved. Skin colorsegmentation is an important step for algorithm of system. If more correct skinregions are segmented, more faces can be detected. Also, instead of RGB,
YCbCrskincolormodellingwithstatisticalmodelcanbeperformed,sinceCbandCrchan nelsvaluesarenot sensitiveto light changes.

On the other hand, some improvements can be applied on facial feature extractionsection in face detection part. Computational volume is the biggest with respect toother sections in the algorithm. Computations of facial feature extraction can bereduced.Otherpointis thattocalculateeyeorientation,whichwillbeusedtoreorient face candidate and extract horizontally oriented face image. This operationwilldecreaseworkinglimitations ofdetection part.

Some improvements can be performed on recognition part. Firstly, number of peoplein the face database can be increased. However, increasing face images may createproblem in classification performance due to less number of sample image for eachpeople. Therefore, sample number could be increased.Later, input neuron numbersof neural network can be decreased with use of feature extraction method to inputfaceimage.Thisfeatureextractionmethodwilldecreasecomputationaltimeofnetwork. Possible extraction methods can be PCA, ICA, DCT or LDA. Also, iffeature extraction will be applied, face image database should be generated withfeatureextracted faceimages.

Later on, this system will be integrated to humanoid robot or narrower applications.Therefore this algorithm should be designed and implemented on embedded systems.DigitalSignalProcessorsorFieldProgrammableGateArrayscanbeusedforembeddedsystems.Withuseofembeddedsystems,realtimefacerecognitionsystemcan beachieved.

CHAPTER 7

REFERENCES

[1] Bernie DiDario, Michael Dobson, and Douglas Ahlers, February 16, 2006. United States Patent Application Publication, Pub. No.: US 2006/0035205 A1. "Attendance Tracking System," United States Patent Application Publication, Pub. No: US 2006/0035205 A1.

[2] O.A. Idowu and O. Shoewu, "Attendance Management System Using Biometrics," May 2012 (Spring). pp. 300-307 in Pacific Journal of Science and Technology, Volume 13, Number 1.

[3] S. Kherchaoui and A. Houacine, "Face Detection Based on A Model of Skin Color With Constraints and Template Matching," in Proc. 2010 International Conference on Machine and Web Intelligence, Algiers, Algeria, pp. 469 - 472.

[4] Shubhobrata Bhattacharya, Shubhobrata Bhattacharya, Shubhobrata Bhattacharya, Shubhobrata Bhattacharya, Shubhobrata "A Face Recognition Based Attendance System for Classroom Environment." "Smart Attendance Monitoring System (SAMS): A Face Recognition Based Attendance System for Classroom Environment." The 18th International Conference on Advanced Learning Technologies (IEEE) will be held in 2018. (ICALT).

[5] Xiang-Yu Li and Zhen-Xian Lin, Xiang-Yu Li and Zhen-Xian Lin, Xiang-Yu Li and Zhen-Xian Lin, X "Face recognition using a quick PCA algorithm and the HOG algorithm." The Euro-China Conference on Intelligent Data Analysis and Applications is a collaboration between Europe and China. Cham. Springer.

[6] June 2012, Cahit Gurel and Abdulkadir Erden. The 15th Conference on Machine Design International and Production, "Design of a Face Recognition System." Pamukkale, Denizli, Turkey, June 19–22, 2012.

[7] Himanshu Tiwari, "Live Attendance System using Face Recognition," International Journal for Research in Applied Science and Engineering Technology (IJRASET), Volume 6 Issue IV, ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887.

[8] Kaneez Laila Bhatti, Laraib Mughal, Faheem Yar Khuhawar, and Sheeraz Ahmed Memon, "Smart Attendance Management System Using Face Recognition," MUET, Jamshoro, Pakistan.

[9] N. Rekha and M. Z. Kurian, "Face identification in real time based on HOG," 2014. The International Journal of Advanced Research in Computer Engineering and Technology is a publication dedicated to cutting-edge research in the field of computer engineering and technology (IJARCET).

[10] Lyon, M.J., Akamatsu, S., Kamachi, M., and Gyoba, J. (1998). The 3rd IEEE International Conference on Automatic Face and Gesture Recognition is published in the proceedings.

Page 205 of 200.

[11] Tan, K.Y., and SEE, A. K. B. [11] Tan, K.Y., and SEE, A. K. B. (2005). Facial Recognition Technology: A Comparison and Implementation of Various Methods

CGST International Journal on Graphics, Vision, and Image Processing, Volume 5, Issue 9, Pages 11-19.

[12] Pang, Y. Zhang, L. LI, M. Liu, Z. Liu, Z. Liu, Z. Liu, Z. Liu, Z. Liu, Z. Liu, Z (2004). A Face Recognition Method Based on Gabor-LDA. In Computer Science Lecture Notes. Germany's Springer-Verlag. This is issue 3331.

[13] A. Amine, S. Ghouzali, and M. Rziza (2006). Face Detection Using Skin Color Information in Still Color Images. The Second International Symposium on Communications, Control, and Signal Processing's Proceedings.

[14] Y.N. Chae, J.N. Chung, and H.S. Yang (2008). Efficient Face Detection Using Color Filtering In IEEE's 19th International Conference on Pattern Recognition Proceedings. 1–4 pages

[15] J. Brand and J. Mason, J. Brand and J. Mason, J. Mason, J. Mason, J. Mason, J. Mason, J (2000). A Comparison of Three Methods for Detecting Human Skin at the Pixel Level. Volume 1, page 1056–1059. In Proceedings of the

International Conference on Pattern Recognition, Vol. 1, page 1056–1059.

CHAPTER 8

SOURCE CODE

```
function varargout = main(varargin)
% MAIN MATLAB code for main.fig
%      MAIN, by itself, creates a new MAIN or raises the existing
%      singleton*.
%
%      H = MAIN returns the handle to a new MAIN or the handle to
%      the existing singleton*.
%
%      MAIN('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in MAIN.M with the given input arguments.
%
%      MAIN('Property','Value',...) creates a new MAIN or raises the
%      existing singleton*.  Starting from the left, property value pairs are
```

```
%       applied to the GUI before main_OpeningFcn gets called.  An
%       unrecognized property name or invalid value makes property application
%       stop.  All inputs are passed to main_OpeningFcn via varargin.
%
%       *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%       instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help main

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @main_OpeningFcn, ...
                   'gui_OutputFcn',  @main_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```

```
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT



% --- Executes just before main is made visible.
function main_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to main (see VARARGIN)

% Choose default command line output for main
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes main wait for user response (see UIRESUME)
```

```
% uiwait(handles.figure1);



% --- Outputs from this function are returned to the command line.
function varargout = main_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% standard size of image is 300 *300
global co
clc
warning off
st = version;
if str2double(st(1)) < 8
    beep
    hx  = msgbox('PLEASE RUN IT ON MATLAB 2013 or
Higher','INFO...!!!','warn','modal');
    pause(3)
    delete(hx)
    close(gcf)
    return
```

```
end
co = get(hObject,'color');
addpath(pwd,'database','codes')
if size(ls('database'),2) == 2
%     delete('features.mat');
%     delete('info.mat');
end
% Get default command line output from handles structure
varargout{1} = handles.output;




function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of edit1 as a double



% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
```

```
% hObject    handle to edit1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    empty - handles not created until after all CreateFcns called


% Hint: edit controls usually have a white background on Windows.

%       See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

    set(hObject,'BackgroundColor','white');

end



% --- Executes on button press in pushbutton1.

function pushbutton1_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

p  = get(handles.edit1,'UserData');

if strcmp(p,'123') == 1

    delete(hObject);

    delete(handles.pushbutton2)

    delete(handles.edit1);

    delete(handles.text2);
```

```
    delete(handles.text3);

    delete(handles.text1);

    delete(handles.text4);

    msgbox('WHY DONT U READ HELP BEFORE
STARTING','HELP....!!!','help','modal')

    set(handles.AD_NW_IMAGE,'enable','on')

    set(handles.DE_LETE,'enable','on')

    set(handles.TRAIN_ING,'enable','on')

    set(handles.STA_RT,'enable','on')

    set(handles.RESET_ALL,'enable','on')

    set(handles.EXI_T,'enable','on')

    set(handles.HE_LP,'enable','on')

    set(handles.DATA_BASE,'enable','on')

    set(handles.text5,'visible','on')

else

    msgbox('INVALID PASSWORD FRIEND... XX','WARNING....!!!','warn','modal')

end


% --- Executes on button press in pushbutton2.

function pushbutton2_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton2 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)
```

```
close gcf


% --------------------------------------------------------------------
function AD_NW_IMAGE_Callback(hObject, eventdata, handles)
% hObject    handle to AD_NW_IMAGE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)



% --------------------------------------------------------------------
function DE_LETE_Callback(hObject, eventdata, handles)
% hObject    handle to DE_LETE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)



% --------------------------------------------------------------------
function TRAIN_ING_Callback(hObject, eventdata, handles)
% hObject    handle to TRAIN_ING (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
% --------------------------------------------------------------------
function STA_RT_Callback(hObject, eventdata, handles)
% hObject    handle to STA_RT (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --------------------------------------------------------------------
function DATA_BASE_Callback(hObject, eventdata, handles)
% hObject    handle to DATA_BASE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --------------------------------------------------------------------
function RESET_ALL_Callback(hObject, eventdata, handles)
% hObject    handle to RESET_ALL (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --------------------------------------------------------------------
function EXI_T_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to EXI_T (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)




% --------------------------------------------------------------------

function HE_LP_Callback(hObject, eventdata, handles)

% hObject    handle to HE_LP (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)




% --------------------------------------------------------------------

function READ_ME_Callback(hObject, eventdata, handles)

% hObject    handle to READ_ME (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

winopen('help.pdf')


% --------------------------------------------------------------------

function PRE_CAP_Callback(hObject, eventdata, handles)

% hObject    handle to PRE_CAP (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB
```

```
% handles    structure with handles and user data (see GUIDATA)

if exist('features.mat','file') == 0

    msgbox('FIRST TRAIN YOUR DATABASE','INFO...!!!','MODAL')

    return

end

ff = dir('database');

if length(ff) == 2

    h = waitbar(0,'Plz wait Matlab is scanning ur database...','name','SCANNING IS
IN PROGRESS');

    for k = 1:100

        waitbar(k/100)

        pause(0.03)

    end

    close(h)

    msgbox({'NO IMAGE FOUND IN DATABASE';'FIRST LOAD YOUR
DATABASE';'USE ''ADD NEW IMAGE''
MENU'},'WARNING....!!!','WARN','MODAL')

    return

end

fd = vision.CascadeObjectDetector();

[f,p] = uigetfile('*.jpg','PLEASE SELECT AN FACIAL IMAGE');

if f == 0

    return

end
```

```
p1 = fullfile(p,f);

im = imread(p1);

bbox = step(fd, im);

vo = insertObjectAnnotation(im,'rectangle',bbox,'FACE');

r = size(bbox,1);

if isempty(bbox)

    axes(handles.axes1)

    imshow(vo);

    msgbox({'NO FACE IN THIS PIC';'PLEASE SELECT SINGLE FACE
IMAGE'},'WARNING...!!!','warn','modal')

    uiwait

    cla(handles.axes1); reset(handles.axes1);
set(handles.axes1,'box','on','xtick',[],'ytick',[])

    return

elseif r > 1

    axes(handles.axes1)

    imshow(vo);

    msgbox({'TOO MANY FACES IN THIS PIC';'PLEASE SELECT SINGLE FACE
IMAGE'},'WARNING...!!!','warn','modal')

    uiwait

    cla(handles.axes1); reset(handles.axes1);
set(handles.axes1,'box','on','xtick',[],'ytick',[])

    return

end
```

```
axes(handles.axes1)

image(vo);

set(handles.axes1,'xtick',[],'ytick',[],'box','on')

bx = questdlg({'CORRECT IMAGE IS SELECTED';'SELECT OPTION FOR FACE
EXTRACTION'},'SELECT AN OPTION','MANUALLY','AUTO','CC');

if strcmp(bx,'MANUALLY') == 1

    while 1

        fhx = figure(2);

        set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')

        imc = imcrop(im);

        bbox1 = step(fd, imc);

        if size(bbox1,1) ~= 1

            msgbox({'YOU HAVENT CROPED A FACE';'CROP AGAIN'},'BAD
ACTION','warn','modal')

            uiwait

        else

            close gcf

            break

        end

        close gcf

    end

 imc = imresize(imc,[300  300]);

 image(imc)
```

```
 text(20,20,'\bfUr Precaptured image.','fontsize',12,'color','y','fontname','comic sans ms')
 set(handles.axes1,'xtick',[],'ytick',[],'box','on')
end
 if strcmp(bx,'AUTO') == 1
    imc = imcrop(im,[bbox(1)-50  bbox(2)-250   bbox(3)+100    bbox(4)+400]);
    fhx = figure(2);
    set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
    imshow(imc)
    qx = questdlg({'ARE YOU SATISFIED WITH THE RESULTS?';' ';'IF YES THEN PROCEED';' ';'IF NOT BETTER DO MANUAL CROPING'},'SELECT','PROCEED','MANUAL','CC');
    if strcmpi(qx,'proceed') == 1
        close gcf
        imc = imresize(imc,[300  300]);
        axes(handles.axes1)
        image(imc)
        text(20,20,'\bfUr Precaptured image.','fontsize',12,'color','y','fontname','comic sans ms')
        set(handles.axes1,'xtick',[],'ytick',[],'box','on')
    elseif strcmpi(qx,'manual') == 1
        while 1
            fhx = figure(2);
            set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
```

```
            imc = imcrop(im);
            bbox1 = step(fd, imc);
            if size(bbox1,1) ~= 1
                msgbox({'YOU HAVENT CROPED A FACE';'CROP AGAIN'},'BAD
ACTION','warn','modal')
                uiwait
            else
                break
            end
            close gcf
        end
    close gcf
    imc = imresize(imc,[300 300]);
    axes(handles.axes1)
    image(imc)
    text(20,20,'\bfUr Precaptured image.','fontsize',12,'color','y','fontname','comic
sans ms')
    set(handles.axes1,'xtick',[],'ytick',[],'box','on')
    else
    end
 end
 immxx = getimage(handles.axes1);
 zz = findsimilar(immxx);
```

```
 zz = strtrim(zz);
fxz = imread(['database/'  zz]);
q1= ehd(immxx,0.1);
q2 = ehd(fxz,0.1);
q3 = pdist([q1 ; q2]);
disp(q3)
if q3 < 0.5
    axes(handles.axes2)
    image(fxz)
    set(handles.axes1,'xtick',[],'ytick',[],'box','on')
    text(20,20,'\bfUr Database Entered
Image.','fontsize',12,'color','y','fontname','comic sans ms')
    set(handles.axes2,'xtick',[],'ytick',[],'box','on')
    xs = load('info.mat');
    xs1 = xs.z2;
    for k = 1:length(xs1)
        st = xs1{k};
        stx = st{1};
        if strcmp(stx,zz) == 1
            str = st{2};
            break
        end
    end
```

```
fid = fopen('attendence_sheet.txt','a');
fprintf(fid,'%s            %s            %s            %s\r\n\n', 'Name','Date','Time',
'Attendence');
c = clock;
if c(4) > 12
    s = [num2str(c(4)-12) ,':',num2str(c(5)), ':', num2str(round(c(6))) ];
else
    s = [num2str(c(4)) ,':',num2str(c(5)), ':', num2str(round(c(6))) ];
end
fprintf(fid,'%s        %s        %s              %s\r\n\n', str, date,s,'Present');
fclose(fid);
set(handles.text5,'string',['Hello ' str ' ,Your attendence has been Marked.'])
try
    s = serial('com22');
    fopen(s);
    fwrite(s,'A');
    pause(1)
    fclose(s);
    clear s
catch
    msgbox({'PLZ CONNECT CABLE OR';'INVALID COM PORT
SELECTED'},'WARNING','WARN','MODAL')
    uiwait
```

```
        delete(s)
        clear s
    end
else
    msgbox('YOU ARE NOT A VALID PERSON', 'WARNING','WARN','MODAL')
    cla(handles.axes1)
    reset(handles.axes1)
    cla(handles.axes2)
    reset(handles.axes2)
    set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5);
    set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)
end



% --------------------------------------------------------------------
function LIVE_CAM_Callback(hObject, eventdata, handles)
% hObject    handle to LIVE_CAM (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global co
if exist('features.mat','file') == 0
```

```
    msgbox('FIRST TRAIN YOUR DATABASE','INFO...!!!','MODAL')

    return

end

ff = dir('database');

if length(ff) == 2

    h = waitbar(0,'Plz wait Matlab is scanning ur database...','name','SCANNING IS
IN PROGRESS');

    for k = 1:100

        waitbar(k/100)

        pause(0.03)

    end

    close(h)

    msgbox({'NO IMAGE FOUND IN DATABASE';'FIRST LOAD YOUR
DATABASE';'USE ''ADD NEW IMAGE''
MENU'},'WARNING....!!!','WARN','MODAL')

    return

end


if isfield(handles,'vdx')

    vid = handles.vdx;

    stoppreview(vid)

    delete(vid)

    handles = rmfield(handles,'vdx');

    guidata(hObject,handles)
```

```
    cla(handles.axes1)

    reset(handles.axes1)

    set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)

    cla(handles.axes2)

    reset(handles.axes2)

    set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)

end

info = imaqhwinfo('winvideo');

did = info.DeviceIDs;

if isempty(did)

    msgbox({'YOUR SYSTEM DO NOT HAVE A WEBCAM';' ';'CONNECT A
ONE'},'WARNING....!!!!','warn','modal')

    return

end

fd = vision.CascadeObjectDetector();

did = cell2mat(did);

for k = 1:length(did)

    devinfo = imaqhwinfo('winvideo',k);

    na(1,k) = {devinfo.DeviceName};

    sr(1,k) = {devinfo.SupportedFormats};

end

[a,b] = listdlg('promptstring','SELECT A WEB CAM
```

```
DEVICE','liststring',na,'ListSize', [125, 75],'SelectionMode','single');
if b == 0
    return
end
if b ~= 0
    frmt = sr{1,a};
    [a1,b1] = listdlg('promptstring','SELECT RESOLUTION','liststring',frmt,'ListSize',
[150, 100],'SelectionMode','single');
    if b1 == 0
        return
    end
end
frmt = frmt{a1};
l = find(frmt == '_');
res = frmt(l+1 : end);
l = find(res == 'x');
res1 = str2double(res(1: l-1));
res2 = str2double(res(l+1 : end));
axes(handles.axes1)
vid = videoinput('winvideo', a);
vr  =  [res1    res2];
nbands  =  get(vid,'NumberofBands');
h2im  = image(zeros([vr(2)  vr(1)  nbands] , 'uint8'));
```

```
preview(vid,h2im);

handles.vdx = vid;

guidata(hObject,handles)

tx = msgbox('PLZ STAND IN FRONT OF CAMERA STILL','INFO......!!!');

pause(1)

delete(tx)

kx  =  0;

while 1

    im = getframe(handles.axes1);

    im =  im.cdata;

    bbox = step(fd, im);

    vo = insertObjectAnnotation(im,'rectangle',bbox,'FACE');

    axes(handles.axes2)

    imshow(vo)

    if size(bbox,1) > 1

        msgbox({'TOO MANY FACES IN FRAME';' ';'ONLY ONE FACE IS
ACCEPTED'},'WARNING.....!!!','warn','modal')

        uiwait

        stoppreview(vid)

        delete(vid)

        handles = rmfield(handles,'vdx');

        guidata(hObject,handles)

        cla(handles.axes1)
```

```
        reset(handles.axes1)

        set(handles.axes1,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1
1],'color',co,'linewidth',1.5)

        cla(handles.axes2)

        reset(handles.axes2)

        set(handles.axes2,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1
1],'color',co,'linewidth',1.5)

        return

    end

    kx = kx + 1;

    if kx > 10 && ~isempty(bbox)

        break

    end

end

imc = imcrop(im,[bbox(1)+3    bbox(2)-35    bbox(3)-10      bbox(4)+70]);

imx  =   imresize(imc,[300  300]);

axes(handles.axes1)

image(imx)

text(20,20,'\bfUr Current image.','fontsize',12,'color','y','fontname','comic sans ms')

set(handles.axes1,'xtick',[],'ytick',[],'box','on')

 immxx = imx;

 zz = findsimilar(immxx);

 zz = strtrim(zz);
```

```
fxz = imread(['database/'  zz]);

q1= ehd(immxx,0.1);

q2 = ehd(fxz,0.1);

q3 = pdist([q1 ; q2]);

disp(q3)

if q3 < 0.5

    axes(handles.axes2)

    image(fxz)

    set(handles.axes1,'xtick',[],'ytick',[],'box','on')

    text(20,20,'\bfUr Database Entered
Image.','fontsize',12,'color','y','fontname','comic sans ms')

    set(handles.axes2,'xtick',[],'ytick',[],'box','on')

    xs = load('info.mat');

    xs1 = xs.z2;

    for k = 1:length(xs1)

        st = xs1{k};

        stx = st{1};

        if strcmp(stx,zz) == 1

            str = st{2};

            break

        end

    end

    fid = fopen('attendence_sheet.txt','a');
```

```
    fprintf(fid,'%s            %s            %s            %s\r\n\n', 'Name','Date','Time',
'Attendence');

    c = clock;

    if c(4) > 12

        s = [num2str(c(4)-12) ,':',num2str(c(5)), ':', num2str(round(c(6))) ];

    else

        s = [num2str(c(4)) ,':',num2str(c(5)), ':', num2str(round(c(6))) ];

    end

    fprintf(fid,'%s       %s       %s            %s\r\n\n', str, date,s,'Present');

    fclose(fid);

    set(handles.text5,'string',['Hello ' str ' ,Your attendence has been Marked.'])

    try

        s = serial('com22');

        fopen(s);

        fwrite(s,'A');

        pause(1)

        fclose(s);

        clear s

    catch

        msgbox({'PLZ CONNECT CABLE OR';'INVALID COM PORT
SELECTED'},'WARNING','WARN','MODAL')

        uiwait

        delete(s)
```

```
        clear s
    end
else
    msgbox('YOU ARE NOT A VALID PERSON', 'WARNING','WARN','MODAL')
    cla(handles.axes1)
    reset(handles.axes1)
    cla(handles.axes2)
    reset(handles.axes2)
    set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5);
    set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)
end
% --------------------------------------------------------------------
function SINGL_PIC_Callback(hObject, eventdata, handles)
% hObject    handle to SINGL_PIC (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
flist  =  dir('database');
if length(flist) == 2
    msgbox('NOTHING TO DELETE','INFO','modal');
    return
end
```

```
cd('database')

[f,p] = uigetfile('*.jpg','SELECT A PIC TO DELETE IT');

if f == 0

    cd ..

    return

end

p1 = fullfile(p,f);

delete(p1)

flist  =  dir(pwd);

if length(flist) == 2

    cd ..

    return

end

for k =  3:length(flist)

    z = flist(k).name;

    z(strfind(z,'.') : end) = [];

    nlist(k-2) = str2double(z);

end

nlist  = sort(nlist);

h = waitbar(0,'PLZ WAIT, WHILE MATLAB IS
RENAMING','name','PROGRESS...');

for k = 1:length(nlist)

    if k ~= nlist(k)
```

```
        p = nlist(k);
        movefile([num2str(p)  '.jpg']  ,  [num2str(k)  '.jpg'])
        waitbar((k-2)/length(flist),h,sprintf('RENAMED  %s   to  %s',[num2str(p)
'.jpg'],[num2str(k)  '.jpg']))
    end
    pause(.5)
end
close(h)
cd ..



% --------------------------------------------------------------------
function MULTI_PIC_Callback(hObject, eventdata, handles)
% hObject    handle to MULTI_PIC (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
flist  =  dir('database');
if length(flist) == 2
    msgbox('NOTHING TO DELETE','INFO','modal');
    return
end
for k =  3:length(flist)
    na1(k-2,1) = {flist(k).name};
```

```
end
[a,b] = listdlg('promptstring','SELECT FILE/FILES TO
DELETE','liststring',na1,'listsize',[125  100]);
if b == 0
    return
end
cd ('database')
for k = 1:length(a)
    str = na1{k};
    delete(str)
end
cd ..
flist  =  dir('database');
if length(flist) == 2
    msgbox({'NOTHING TO RENAME';'ALL DELETED'},'INFO','modal');
    return
end
cd('database')
flist  =  dir(pwd);
for k =  3:length(flist)
    z = flist(k).name;
    z(strfind(z,'.') : end) = [];
    nlist(k-2) = str2double(z);
```

```
end

nlist  = sort(nlist);

h = waitbar(0,'PLZ WAIT, WHILE MATLAB IS
RENAMING','name','PROGRESS...');

for k = 1:length(nlist)

    if k ~= nlist(k)

        p = nlist(k);

        movefile([num2str(p)  '.jpg']  ,  [num2str(k)  '.jpg'])

        waitbar((k-2)/length(flist),h,sprintf('RENAMED  %s   to  %s',[num2str(p)
'.jpg'],[num2str(k)  '.jpg']))

    end

    pause(.5)

end

close(h)

cd ..




% --------------------------------------------------------------------

function BR_OWSE_Callback(hObject, eventdata, handles)

% hObject    handle to BR_OWSE (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

[f,p] = uigetfile('*.jpg','PLEASE SELECT AN FACIAL IMAGE');
```

```
if f == 0
    return
end
p1 = fullfile(p,f);
im = imread(p1);
fd = vision.CascadeObjectDetector();
bbox = step(fd, im);
vo = insertObjectAnnotation(im,'rectangle',bbox,'FACE');
r = size(bbox,1);
if isempty(bbox)
    fhx = figure(2);
    set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
    imshow(vo);
    msgbox({'WHAT HAVE U CHOOSEN?';'NO FACE FOUND IN THIS
PIC,';'SELECT SINGLE FACE IMAGE.'},'WARNING...!!!','warn','modal')
    uiwait
    delete(fhx)
    return
elseif r > 1
    fhx = figure(2);
    set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
    imshow(vo);
    msgbox({'TOO MANY FACES IN THIS PIC';'PLEASE SELECT SINGLE FACE
```

```
IMAGE'},'WARNING...!!!','warn','modal')
    uiwait
    delete(fhx)
    return
end
bx = questdlg({'CORRECT IMAGE IS SELECTED';'SELECT OPTION FOR FACE
EXTRACTION'},'SELECT AN OPTION','MANUALLY','AUTO','CC');
if strcmp(bx,'MANUALLY') == 1
    while 1
        fhx = figure(2);
        set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
        imc = imcrop(im);
        bbox1 = step(fd, imc);
        if size(bbox1,1) ~= 1
            msgbox({'YOU HAVENT CROPED A FACE';'CROP AGAIN'},'BAD
ACTION','warn','modal')
            uiwait
        else
            break
        end
        close gcf
    end
    close gcf
```

```
imc = imresize(imc,[300  300]);

cd ('database');

l = length(dir(pwd));

n = [int2str(l-1)    '.jpg'];

imwrite(imc,n);

cd ..

while 1

   qq = inputdlg('WHAT IS UR NAME?','FILL');

   if isempty(qq)

      msgbox({'YOU HAVE TO ENTER A NAME';' ';'YOU CANT CLICK
CANCEL'},'INFO','HELP','MODAL')

      uiwait

   else

      break

   end

end

qq = qq{1};

if exist('info.mat','file') == 2

   load ('info.mat')

   r = size(z2,1);

   z2{r+1,1} = {n , qq};

   save('info.mat','z2')

else
```

```
        z2{1,1} = {n,qq};
        save('info.mat','z2')
    end
end
if strcmp(bx,'AUTO') == 1
    imc = imcrop(im,[bbox(1)-50  bbox(2)-250   bbox(3)+100    bbox(4)+400]);
    fhx = figure(2);
    set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
    imshow(imc)
    qx = questdlg({'ARE YOU SATISFIED WITH THE RESULTS?';' ';'IF YES THEN
PROCEED';' ';'IF NOT BETTER DO MANUAL
CROPING'},'SELECT','PROCEED','MANUAL','CC');
    if strcmpi(qx,'proceed') == 1
        imc = imresize(imc,[300  300]);
        cd ('database');
        l = length(dir(pwd));
        n = [int2str(l-1)    '.jpg'];
        imwrite(imc,n);
        cd ..
        while 1
        qq = inputdlg('WHAT IS UR NAME?','FILL');
        if isempty(qq)
            msgbox({'YOU HAVE TO ENTER A NAME';' ';'YOU CANT CLICK
CANCEL'},'INFO','HELP','MODAL')
```

```
        uiwait
    else
        break
    end
    end
    qq = qq{1};
if exist('info.mat','file') == 2
    load ('info.mat')
    r = size(z2,1);
    z2{r+1,1} = {n , qq};
    save('info.mat','z2')
else
    z2{1,1} = {n,qq};
    save('info.mat','z2')
end
close gcf
elseif strcmpi(qx,'manual') == 1
    while 1
        fhx = figure(2);
        set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')
        imc = imcrop(im);
        bbox1 = step(fd, imc);
        if size(bbox1,1) ~= 1
```

```
            msgbox({'YOU HAVENT CROPED A FACE';'CROP AGAIN'},'BAD
ACTION','warn','modal')
            uiwait
         else
            break
         end
         close gcf
      end
   close gcf
   imc = imresize(imc,[300  300]);
   cd ('database');
   l = length(dir(pwd));
   n = [int2str(l-1)    '.jpg'];
   imwrite(imc,n);
   cd ..
   while 1
      qq = inputdlg('WHAT IS UR NAME?','FILL');
      if isempty(qq)
         msgbox({'YOU HAVE TO ENTER A NAME';' ';'YOU CANT CLICK
CANCEL'},'INFO','HELP','MODAL')
         uiwait
      else
         break
```

```
        end
    end
    qq = qq{1};
    if exist('info.mat','file') == 2
        load ('info.mat')
        r = size(z2,1);
        z2{r+1,1} = {n , qq};
        save('info.mat','z2')
    else
        z2{1,1} = {n,qq};
        save('info.mat','z2')
    end
    else
        return
    end
end



% --------------------------------------------------------------------
function FRM_CAM_Callback(hObject, eventdata, handles)
% hObject    handle to FRM_CAM (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
global co
if isfield(handles,'vdx')
    vid = handles.vdx;
    stoppreview(vid)
    delete(vid)
    handles = rmfield(handles,'vdx');
    guidata(hObject,handles)
    cla(handles.axes1)
    reset(handles.axes1)
    set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431 
0.5176  0.7804],'linewidth',1.5)
    cla(handles.axes2)
    reset(handles.axes2)
    set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431 
0.5176  0.7804],'linewidth',1.5)
end
fd = vision.CascadeObjectDetector();
info = imaqhwinfo('winvideo');
did = info.DeviceIDs;
if isempty(did)
    msgbox({'YOUR SYSTEM DO NOT HAVE A WEBCAM';' ';'CONNECT A 
ONE'},'WARNING....!!!!','warn','modal')
    return
end
```

```
did = cell2mat(did);

for k = 1:length(did)

    devinfo = imaqhwinfo('winvideo',k);

    na(1,k) = {devinfo.DeviceName};

    sr(1,k) = {devinfo.SupportedFormats};

end

[a,b] = listdlg('promptstring','SELECT A WEB CAM
DEVICE','liststring',na,'ListSize', [125, 75],'SelectionMode','single');

if b == 0

    return

end

if b ~= 0

    frmt = sr{1,a};

    [a1,b1] = listdlg('promptstring','SELECT RESOLUTION','liststring',frmt,'ListSize',
[150, 100],'SelectionMode','single');

    if b1 == 0

        return

    end

end

frmt = frmt{a1};

l = find(frmt == '_');

res = frmt(l+1 : end);

l = find(res == 'x');
```

```
res1 = str2double(res(1: l-1));

res2 = str2double(res(l+1 : end));

axes(handles.axes1)

vid = videoinput('winvideo', a);

vr  =  [res1    res2];

nbands  =  get(vid,'NumberofBands');

h2im  = image(zeros([vr(2)  vr(1)  nbands] , 'uint8'));

preview(vid,h2im);

handles.vdx = vid;

guidata(hObject,handles)

tx = msgbox('PLZ STAND IN FRONT OF CAMERA STILL','INFO......!!!');

pause(1)

delete(tx)

kx  =  0;

while 1

    im = getframe(handles.axes1);

    im =  im.cdata;

    bbox = step(fd, im);

    vo = insertObjectAnnotation(im,'rectangle',bbox,'FACE');

    axes(handles.axes2)

    imshow(vo)

    if size(bbox,1) > 1

        msgbox({'TOO MANY FACES IN FRAME';' ';'ONLY ONE FACE IS
```

```
ACCEPTED'},'WARNING.....!!!','warn','modal')
        uiwait
        stoppreview(vid)
        delete(vid)
        handles = rmfield(handles,'vdx');
        guidata(hObject,handles)
        cla(handles.axes1)
        reset(handles.axes1)
        set(handles.axes1,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1
1],'color',co,'linewidth',1.5)
        cla(handles.axes2)
        reset(handles.axes2)
        set(handles.axes2,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1
1],'color',co,'linewidth',1.5)
        return
    end
    kx = kx + 1;
    if kx > 10 && ~isempty(bbox)
        break
    end
end
imc = imcrop(im,[bbox(1)+3    bbox(2)-35    bbox(3)-10      bbox(4)+70]);
imx  =   imresize(imc,[300  300]);
```

```
fhx = figure(2);

set(fhx,'menubar','none','numbertitle','off','name','PREVIEW')

imshow(imx)

cd ('database');

l = length(dir(pwd));

n = [int2str(l-1)    '.jpg'];

imwrite(imx,n);

cd ..

   while 1

      qq = inputdlg('WHAT IS UR NAME?','FILL');

      if isempty(qq)

         msgbox({'YOU HAVE TO ENTER A NAME';' ';'YOU CANT CLICK
CANCEL'},'INFO','HELP','MODAL')

         uiwait

      else

         break

      end

   end

   qq = qq{1};

   if exist('info.mat','file') == 2

      load ('info.mat')

      r = size(z2,1);

      z2{r+1,1} = {n , qq};
```

```
        save('info.mat','z2')
    else
        z2{1,1} = {n,qq};
        save('info.mat','z2')
    end
close gcf
stoppreview(vid)
delete(vid)
handles = rmfield(handles,'vdx');
guidata(hObject,handles)
cla(handles.axes1)
reset(handles.axes1)
set(handles.axes1,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1 1],'color',co,'linewidth',1.5)
cla(handles.axes2)
reset(handles.axes2)
set(handles.axes2,'box','on','xtick',[],'ytick',[],'xcolor',[1 1 1],'ycolor',[1 1 1],'color',co,'linewidth',1.5)


% --- Executes on key press with focus on edit1 and none of its controls.
function edit1_KeyPressFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
```

```
% eventdata  structure with the following fields (see UICONTROL)
%       Key: name of the key that was pressed, in lower case
%       Character: character interpretation of the key(s) that was pressed
%       Modifier: name(s) of the modifier key(s) (i.e., control, shift) pressed
% handles    structure with handles and user data (see GUIDATA)
pass = get(handles.edit1,'UserData');
v = double(get(handles.figure1,'CurrentCharacter'));
if v == 8
    pass = pass(1:end-1);
    set(handles.edit1,'string',pass)
elseif any(v == 65:90) || any(v  == 97:122) || any(v == 48:57)
    pass = [pass  char(v)];
elseif v == 13
    p  = get(handles.edit1,'UserData');
    if strcmp(p,'123') == true
        delete(hObject);
        delete(handles.pushbutton2)
        delete(handles.pushbutton1);
        delete(handles.text2);
        delete(handles.text3);
        delete(handles.text1);
        delete(handles.text4);
        msgbox('WHY DONT U READ HELP BEFORE
```

```
STARTING','HELP....!!!','help','modal')
        set(handles.AD_NW_IMAGE,'enable','on')
        set(handles.DE_LETE,'enable','on')
        set(handles.TRAIN_ING,'enable','on')
        set(handles.STA_RT,'enable','on')
        set(handles.RESET_ALL,'enable','on')
        set(handles.EXI_T,'enable','on')
        set(handles.HE_LP,'enable','on')
        set(handles.DATA_BASE,'enable','on')
        set(handles.text5,'visible','on')
        return
    else
        beep
        msgbox('INVALID PASSWORD FRIEND...
XX','WARNING....!!!','warn','modal')
        uiwait;
        set(handles.edit1,'string','')
        return
    end
else
    msgbox({'Invalid Password Character';'Can''t use Special
Character'},'warn','modal')
    uiwait;
```

```
    set(handles.edit1,'string','')
    return
end
set(handles.edit1,'UserData',pass)
set(handles.edit1,'String',char('*'*sign(pass)))



% --------------------------------------------------------------------
function VI_EW_Callback(hObject, eventdata, handles)
% hObject    handle to VI_EW (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

f = dir('database');
if length(f) == 2
    msgbox('YOUR DATA BASE HAS NO IMAGE TO DISPLAY','SORRY','modal')
    return
end
l = length(f)-2;
while 1
    a = factor(l);
    if length(a) >= 4
        break
```

```
    end

    l = l+1;

end

d  = a(1: ceil(length(a)/2));

d = prod(d);

d1 = a(ceil(length(a)/2)+1  : end);

d1 = prod(d1);

zx = sort([d d1]);

figure('menubar','none','numbertitle','off','name','Images of
Database','color',[0.0431  0.5176  0.7804],'position',[300 200 600 500])

for k = 3:length(f)

    im = imread(f(k).name);

    subplot(zx(1),zx(2),k-2)

    imshow(im)

    title(f(k).name,'fontsize',10,'color','w')

end




% --------------------------------------------------------------------

function Start_Training_Callback(hObject, eventdata, handles)

% hObject    handle to Start_Training (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB
```

```
% handles    structure with handles and user data (see GUIDATA)

ff = dir('database');

if length(ff) == 2

    h = waitbar(0,'Plz wait Matlab is scanning ur database...','name','SCANNING IS IN PROGRESS');

    for k = 1:100

        waitbar(k/100)

        pause(0.03)

    end

    close(h)

    msgbox({'NO IMAGE FOUND IN DATABASE';'FIRST LOAD YOUR DATABASE';'USE ''ADD NEW IMAGE'' MENU'},'WARNING....!!!','WARN','MODAL')

    return

end

if exist('features.mat','file') == 2

    bx = questdlg({'TRAINING HAS ALREDY BEEN DONE';' ';'WANT TO TRAIN DATABASE AGAIN?'},'SELECT','YES','NO','CC');

    if strcmpi(bx,'yes') == 1

        builddatabase

        msgbox('TRAINING DONE....PRESS OK TO CONTINUE','OK','modal')

        return

    else

        return
```

```
    end
else
    builddatabase
    msgbox('TRAINING DONE....PRESS OK TO CONTINUE','OK','modal')
    return
end



% ----------------------------------------------------------------
function BYE_Callback(hObject, eventdata, handles)
% hObject    handle to BYE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close gcf



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%end%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%



% ----------------------------------------------------------------
function ATTENDENCE_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to ATTENDENCE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if  exist('attendence_sheet.txt','file') == 2
    winopen('attendence_sheet.txt')
else
    msgbox('NO ATTENDENCE SHEET TO DISPLAY','INFO...!!!','HELP','MODAL')
end


% --------------------------------------------------------------------
function DEL_ATTENDENCE_Callback(hObject, eventdata, handles)
% hObject    handle to DEL_ATTENDENCE (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if  exist('attendence_sheet.txt','file') == 2
   delete('attendence_sheet.txt')
   msgbox('ATTENDENCE DELETED','INFO...!!!','MODAL')
else
    msgbox('NO ATTENDENCE SHEET TO DELETE','INFO...!!!','HELP','MODAL')
end



% --------------------------------------------------------------------
```

```
function Untitled_1_Callback(hObject, eventdata, handles)

% hObject    handle to Untitled_1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

x  = questdlg({'Resetting will Clear the followings: ';'1. Attendence_sheet';'2. Database';'3. features.mat';'4. Info.mat';'Do u want to continue?'},'Please select...!!');

if strcmpi(x,'yes') == 1

    delete('attendence_sheet.txt')

    delete('features.mat')

    delete('info.mat')

    cd ([pwd, '\database'])

    f = dir(pwd);

    for k = 1:length(f)

        delete(f(k).name)

    end

    cd ..

    cla(handles.axes1);

    reset(handles.axes1);

    set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431 0.5176  0.7804],'linewidth',1.5)

    cla(handles.axes2);

    reset(handles.axes2);

    set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
```

```
0.5176  0.7804],'linewidth',1.5)

    set(handles.text5,'string','')

    beep

    msgbox('All Reset','Info','modal')

end



% --------------------------------------------------------------------

function Untitled_2_Callback(hObject, eventdata, handles)

% hObject    handle to Untitled_2 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

cla(handles.axes1);

reset(handles.axes1);

set(handles.axes1,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)

cla(handles.axes2);

reset(handles.axes2);

set(handles.axes2,'box','on','xcolor','w','ycolor','w','xtick',[],'ytick',[],'color',[0.0431
0.5176  0.7804],'linewidth',1.5)

set(handles.text5,'string','')
```

```
% --------------------------------------------------------------------
function Untitled_3_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --------------------------------------------------------------------
function Untitled_4_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% --------------------------------------------------------------------
function Untitled_5_Callback(hObject, eventdata, handles)
% hObject    handle to Untitled_5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```