Human Face Detection Using Skin Color Information

By

S.PRASATH

Reg. No. 1020106013

of

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution affiliated to Anna University of Technology, Coimbatore)

COIMBATORE - 641049

A MINI PROJECT REPORT

Submitted to the

FACULTY OF ELECTRONICS AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF ENGINEERING

IN

APPLIED ELECTRONICS

MAY 2011

TABLE OF CONTENT

ABSTRACT

Skin and human face detection are one of the most interesting research areas in image processing and computer vision applications. A great effort has been devoted on this area; many algorithms and systems have been developed recently.

This project implements a new method to detect a human skin and faces from colored images. It is based on the detection of all pixels in colored images which are probably a human skin via a reference skin colors matrix. The image then goes through some modifications to enhance the face detection. The circularity feature was used to distinguish human faces from other objects with similar skin color. The proposed system was tested using MatLab using different real images and the simulation results show effectiveness of the proposed method.

ACKNOWLEDGEMENT

I express my profound gratitude to our chairman **Padmabhusan Arutselvar Dr.N.Mahalingam B.Sc., F.I.E.** for giving this opportunity to pursue this course.

At this pleasing moment of having successfully completed the project work, I wish to acknowledge my sincere gratitude and heartfelt thanks to our beloved Principal **Dr.S.Ramachandran, Ph.D.,** for having given me the adequate support and opportunity for completing this project work successfully.

I express my sincere thanks to **Dr.Ms.Rajeswari Mariappan, Ph.D.,** Head of the Department of Electronics and Communication Engineering, who rendering us all the time by helps throughout this project

In particular, I wish to thank and everlasting gratitude to the project coordinator **Ms.R.Hemalatha, M.E.,** Asst. professor(SRG), Department of Electronics and Communication Engineering for her expert counseling and guidance to make this project to a great deal of success. With her careful supervision and ensured me in the attaining perfection of work.

I extend my heartfelt thanks to my internal guide **Ms.A.Amsaveni, M.E.,(Ph.D),** Asst.Professor(SRG) for her ideas and suggestion, which have been very helpful for the completion of this project work.

Last, but not the least, I would like to express my gratitude to my family members, friends and to all my staff members of Electronics and Communication Engineering department for their encouragement and support throughout the course of this project.

BONAFIDE CERTIFICATE

Certified that this project report entitled "**HUMAN FACE DETECTION USING SKIN COLOUR INFORMATION**" is the bonafide work of **S.PRASATH** [Reg. no. 1020106013] who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

18/5/11

Project Guide

Mrs.A.Amsaveni, M.E., (Ph.D)

for Head of the Department

Dr. Mrs. Rajeswari Mariappan, PhD

The candidate with university Register no.1020106013 is examined by us in the project viva-voce examination held on 23-05-2011

Internal Examiner

External Examiner

LIST OF FIGURES

1.3 SOFTWARE USED

- MATLAB

1.4 ORGANIZATION OF THE REPORT

- **Chapter 2** discusses about the skin colour pixels.
- **Chapter 3** discusses about the skin detection.
- **Chapter 4** discusses about the edge detection.
- **Chapter 5** discusses about the face detection.
- **Chapter 6** shows the result.
- **Chapter 7** shows the conclusion of the project.

CHAPTER 1

INTRODUCTION

The process of human face identification in images is a complex process for the computer, but with some information and features provided to the computer, it will be able to identify a human face. One of the most important information is the color, the color of a human face. However, the color is not enough to detect and localize human faces in images since other parts of the body often have the similar color. Therefore, the computer must use other information beside the color to detect the face and ignore other objects of the human body. This includes using the geometry of the face, because it differs from the other parts of the body such as the hand.

1.1. PROJECT GOAL

The ultimate goal of this project is detect the skin areas in an input image based on RGB color space by comparing it with a matrix contains a collection of skin color possible values which represents the skin color of most people. The circularity of the human faces was used as a feature to detect, localize and distinguish human faces form other objects.

1.2 OVERVIEW

Skin color detection can applied on different color spaces. The most common color spaces are RGB, HSV, and TSL spaces. One can use a combination of different color spaces to get more accuracy in skin and face detection. In RGB, YCrCb and HSV (or HSI) color spaces were used together to detect the skin areas in color images, an YCrCb color space can be used in parallel with space to minimize the effect of brightness in RGB space. It can be done also by using K-mean clustering method. Facial features are important to detect faces in colored images. Such features are Eyes and mouth which used to detect and localizes faces in the image.

CHAPTER 2

SKIN COLOUR PIXELS

The skin colour and many other features should be provided to the computer in order to detect human faces in input images. If the computer already has reference of the human skin colours, then it will determine all pixels that represent the human face along with the rest of the non-covered body parts such as hands. Therefore, the main challenge is to train the computer about all possible skin colours. Then the computer will select only the pixels of an input image that represent the colour of the body, and then use the facial characteristics such as geometry and any other properties to detect and localize the faces from all other parts.

The main idea is to create a database containing all the pixels that represent the colours of the human skin. The problem is that the colour of the human body varies from person to person, and there are different colours for different races. Therefore, the pixel compilation process that represents the human skin must compile pixels from different images and different skin colours.

The algorithm will compare all the pixels in the database, reference, image with each pixel in the possessing image, if the pixel in the processing image is close enough to one of the pixels in the database image, then that pixels will be labelled as 1, which is a skin colour, if not the pixels will be labelled as 0 i.e. non-skin pixel.

CHAPTER 3

SKIN COLOUR DETECTION

The first and most important feature on face detection and localizing systems is the skin colour. The proposed system uses a matrix that contains a collection of RGB skin values as a reference; these values were used to detect all similar pixels in input RGB image. Each pixel in RGB image can be represented as a vector with three values; Red, Green and Blue. These values are added together in order to produce range of colours. When building a system, that uses skin colour as a feature for face detection, the researcher usually faces three main problems. First, what colour space to choose, second, how exactly the skin colour distribution should be modelled, and the third is the way of processing of color segmentation results for face detection

3.1 RGB Colour space- skin modelling

RGB is a colour space originated from CRT (or similar) display applications, when it was convenient to describe colour as a combination of three coloured rays (red, green and blue). It is one of the most widely used colour spaces for processing and storing of digital image data. However, high correlation between channels, significant perceptual non-uniform, mixing of chrominance and luminance data make RGB not a very favourable choice for colour analysis and colour based recognition algorithms.

Once the difference is less than a threshold (the threshold here is the maximum allowed difference), then this pixel is set to be a skin. The results of this process is a binary image contains 1's and 0's; while the 1's represent all pixels that have similar skin colour, and the 0's represents non-skin pixels.

The produced binary image goes into a set of modifications. The first modification is removing the small areas; these small areas are mostly located far away from the actual human skin in the image. On the other hand, it may be located inside the actual skin area, so the actual skin areas will contain holes, this can be fixed by filling the holes in the binary image.

3.4 Gray Levels

Gray levels represent the interval number of quantization in gray scale image processing. At present, the most commonly used storage method is 8-bit storage. There are 256 gray levels in an 8 bit gray scale image, and the intensity of each pixel can have from 0 to 255, with 0 being black and 255 being white. Another commonly used storage method is 1-bit storage. There are two gray levels, with 0 being black and 1 being white a binary image, which, is frequently used in medical images, is being referred to as binary image. As binary images are easy to operate, other storage format images are often converted into binary images when they are used for enhancement or edge detection

3.2 Normalized RGB

Normalized RGB is a representation that is easily obtained from the RGB values by a simple normalization procedure:

$$r = \frac{R}{R+G+B} \quad g = \frac{G}{R+G+B} \quad b = \frac{B}{R+G+B}$$

As the sum of the three normalized components is known ($r+g+b = 1$), the third component does not hold any significant information and can be omitted, reducing the space dimensionality. The remaining components are often called "pure colours", for the dependence of r and g on the brightness of the source RGB colour is diminished by the normalization. A remarkable property of this representation is that for surfaces, while ignoring ambient light, normalized RGB is invariant (under certain assumptions) to changes of surface orientation relatively to the light source.

3.3 Binary Image

Since the reference matrix used contains all RGB vectors that might be a skin colour, then the system computes the difference (norm) between each vector in the reference matrix and pixels in the input RGB image within a loop, the equation used to calculate the difference is:

$$norm = \sqrt{(R_i - R_r)^2 + (G_i - G_r)^2 + (B_i - B_r)^2}$$

Where:

Ri: The Red component in the input image.

Rr: The Red component in the reference matrix.

Gi: The Green component in the input image.

Gr: The Green component in the reference matrix.

Bi: The Blue component in the input image.

Br: The Blue component in the reference matrix.

CHAPTER 4

EDGE DETECTION

Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. If the background pixels have values similar to the skin colour, then the background and the faces will be as one object. This problem can be solved by finding the edges. The input image was transformed into gray level format and then the edges of the image were found.

The edge-finding can be done by using two methods

- Sobel method
- Canny method

4.1 Sobel Edge Detection Method

Sobel which is a popular edge detection method is considered in this work. In the edge function, the Sobel method uses the derivative approximation to find edges. Therefore, it returns edges at those points where the gradient of the considered image is maximum. The horizontal and vertical gradient matrices whose dimensions are 3×3 for the Sobel method has been generally used in the edge detection operations. In this work, a function is developed to find edges using the matrices whose dimensions are 5×5 in matlab.

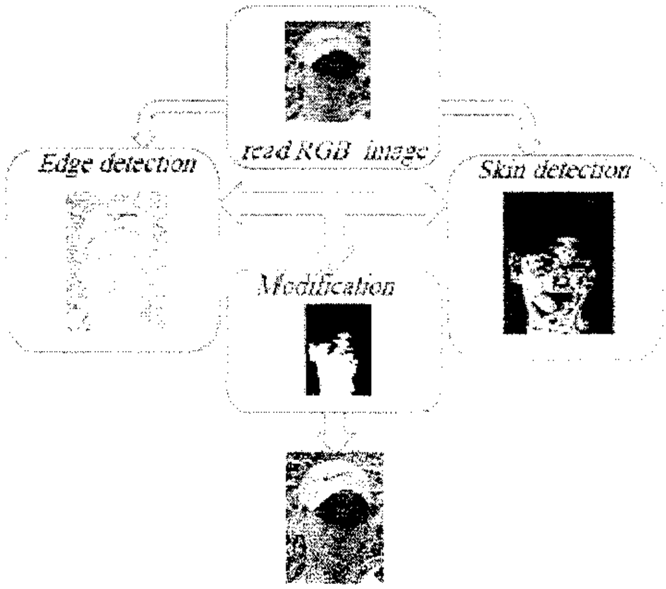

Fig 3.1 Block diagram of the system

Another operation is to create a morphological structuring element with the specified neighbourhood. The specified neighbourhood represents the 1`s in the skin binary image. The skin binary image would have a flat linear structure, the dots and non-uniformity in the image will disappear. This operation makes the calculation of the features more accurate.

The following weighting functions for x and y components were obtained by using the above vector.

1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

The resultant vector G' (similar to the determination of Sobel 3×3 method) for 5×5 is given as follows:

$$
\begin{aligned}
G' = [&20(n-l)+10(i-r-g+t+o-k)+5(e-v-a+z)+4(d-w-b+y) \\
&+8(j-p-f+u), 20(h-s)+10(i-r+g-t)+5\cdot(e-v+a-z) \\
&+4(j-p+f-u)+8(d-w+b-y)]
\end{aligned}
$$

The horizontal and vertical masks are obtained by using the coefficients in this equation such

as

-5	-4	0	4	5
-8	-10	0	10	8
-10	-20	0	20	10
-8	-10	0	10	8
-5	-4	0	4	5

5	8	10	8	5
4	10	20	10	4
0	0	0	0	0
-4	-10	-20	-10	-4
-5	-8	-10	-8	-5

.1.2 Edge Detection Function

Each direction of Sobel masks is applied to an image, then two new images are reated. One image shows the vertical response and the other shows the horizontal response. wo images combined into a single image. The purpose is to determine the existence and ocation of edges in a picture. The new image on which edge pixels are located obtained the alue which is the squared of the above summation

4.1.1 Sobel Operator

Standard Sobel operators, for a 3×3 neighbourhood, each simple central gradient estimate is vector sum of a pair of orthogonal vectors. Each orthogonal vector is a directional derivative estimate multiplied by a unit vector specifying the derivative's direction. The vector sum of these simple gradient estimates amounts to a vector sum of the 8 directional derivative vectors. Thus for a point on Cartesian grid and its eight neighbours having density values as shown

a	b	c
d	e	f
g	h	i

The directional derivative estimate vector G was defined such as density difference / distance to neighbour. This vector is determined such that the direction of G will be given by the unit vector to the approximate neighbour. Note that, the neighbours group into antipodal pairs: (a,i), (b,h), (c,g), (f,d).

$$G = \frac{(c-g)}{R} \cdot \frac{[1,1]}{R} + \frac{(a-i)}{R} \cdot \frac{[-1,1]}{R} + (b-h)\cdot[0,1] + (f-d)\cdot[1,0]$$

where, $R = 2$. This vector is obtained as

$$G = [(c-g-a+i)/2 + f - d,\ (c-g+a-i)/2 + b - h]$$

Here, this vector is multiplied by 2 because of replacing the divide by 2. The resultant formula is given as follows

$$G' = 2.G = [(c-g-a+i) + 2.(f-d),\ (c-g+a-i) + 2.(b-h)]$$

The value of threshold in this above process is used to detect edge pixels. An algorithm is developed to find edges using the new matrices and then, a matlab function, which is called as Sobel5×5.m, is implemented in matlab. This matlab function requires a gray scale intensity image, two-dimensional array. The result which is returned by this function is the final image in which the edge pixels are denoted by white colour.

4.2 Canny Edge Detection

Canny edge detection algorithm is also known as the optimal edge detector. Canny's intentions were to enhance the many edge detectors in the image.

- The first criterion should have low error rate and filter out unwanted information while the useful information preserve.
- The second criterion is to keep the lower variation as possible between the original image and the processed image.
- Third criterion removes multiple responses to an edge.

Canny method calculates the gradient by using the derivative of Gaussian filter and then finding the local maxima of the gradient.

4.2.1 Smoothing

It is inevitable that all images taken from a camera will contain some amount of noise. To prevent that noise is mistaken for edges, noise must be reduced. Therefore the image is first smoothed by applying a Gaussian filter.

4.2.2 Finding Gradients

The Canny algorithm basically finds edges where the grayscale intensity of the image changes the most. These areas are found by determining gradients of the image. Gradients at each pixel in the smoothed image are determined by applying what is known as the Sobel-operator. First step is to approximate the gradient in the x- and y-direction respectively by applying the kernels

The gradient magnitudes (also known as the edge strengths) can then be determined as an Euclidean distance measure by applying the law of Pythagoras. The computed edge strengths are compared to the smoothed image

$$|G| = \sqrt{G_x^2 + G_y^2}$$

$$|G| = |G_x| + |G_y|$$

where:

G_x and G_y are the gradients in the x- and y-directions respectively.

The direction of the edges must be determined and stored as shown in Equation

$$\theta = \arctan\left(\frac{|G_y|}{|G_x|}\right)$$

4.2.4 Double Thresholding

The edge-pixels remaining after the non-maximum suppression step are (still) marked with their strength pixel-by-pixel. Many of these will probably be true edges in the image, but some may be caused by noise or colour variations for instance due to rough surfaces. The simplest way to discern between these would be to use a threshold, so that only edges stronger that a certain value would be preserved. The Canny edge detection algorithm uses double thresholding. Edge pixels stronger than the high threshold are marked as strong; edge pixels weaker than the low threshold are suppressed and edge pixels between the two thresholds are marked as weak.

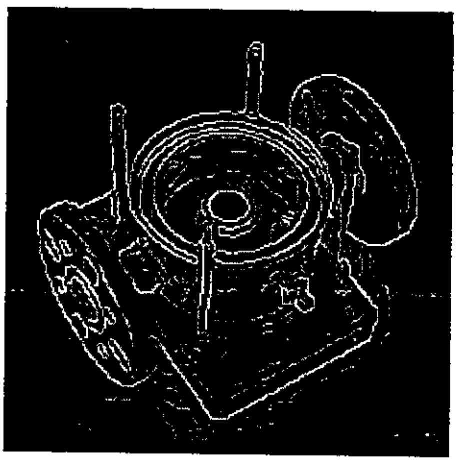

Fig 4.2 Double tresholding

4.2.3 Non-Maximum Suppression

The purpose of this step is to convert the "blurred" edges in the image of the gradient magnitudes to "sharp" edges. Basically this is done by preserving all local maxima in the gradient image, and deleting everything else. The algorithm is for each pixel in the gradient image:

1. Round the gradient direction to nearest 45°, corresponding to the use of an 8-connected neighbourhood.

2. Compare the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient direction. i.e. if the gradient direction is north ($\theta = 90°$), compare with the pixels to the north and south.

3. If the edge strength of the current pixel is largest, then preserve the value of the edge strength. If not, suppress (i.e. remove) the value.

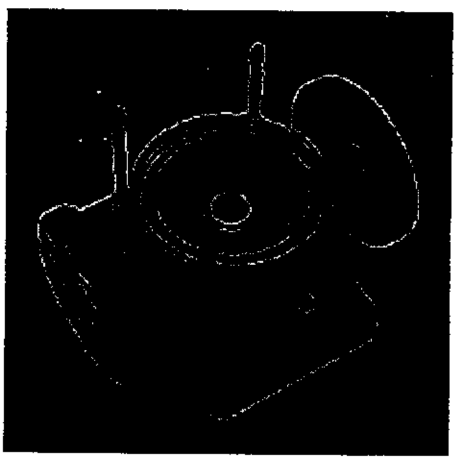

Fig 4.2 Edge after non –maximal suppression

CHAPTER 5

FACE DETECTION

Human face localization and detection are often the first step in applications such as ʼideo surveillance, human computer interface, face recognition and image database nanagement. Locating and tracking human faces is a prerequisite for face recognition and/or acial expressions analysis, although it is often assumed that a normalized face image is ːvailable.

.1 Connectivity Analysis

Using the skin detected image, one knows whether a pixel is a skin pixel or not, but annot say anything about whether a pixel belongs to a face or not. One cannot say anything bout it at the pixel level. It is necessary to go to a higher level and to categorize the skin ixels into different groups so that they will represent something meaningful as a group, for xample a face, a hand etc. It is necessary to form meaningful groups of pixels, it makes ense to group pixels that are connected to each other geometrically. The skin pixels in the nage are grouped based on a 8-connected neighbourhood i.e. if a skin pixel has got another kin pixel in any of its 8 neighbouring places, then both the pixels belong to the same region.

or finding height,

- ➢ The y-coordinate of the centroid is subtracted from the y-coordinates of all pixels in the region.
- ➢ Find the average of all the positive y-coordinates and negative y-coordinates separately.
- ➢ Add the absolute values of both the averages and multiply by 2. This gives the average height of the region.

Average width can be found similarly by using x coordinates. Since the height to idth ratio of human faces falls within a small range on the real axis, using this parameter long with percentage of skin in a region, the algorithm should be able to throw away most of on face skin regions. So if the height to width ratio falls within the range of well known olden ratio $(= (1 + \sqrt{5})/2) \pm$ tolerance and if the percentage of skin is higher than a ireshold called percentage threshold, then that region is considered a face region. The gorithm works with faces of all sizes and does not assume anything about the scale at which face appears. Instead it gives the size of the face detected which will be useful when the ices detected are sent for further processing by a face recognition system.

.2 Using Edge Information

There are many ways to perform edge detection. However, the most may be grouped to two categories, gradient and Laplacian. The gradient method detects the edges by oking for the maximum and minimum in the first derivative of the image. The Laplacian ethod searches for zero crossings in the second derivative of the image to find edges. Sobel, rewitt and Roberts operators come under gradient method while Marrs-Hildreth is a iplacian method. Among these, the Sobel operator is fast, detects edges at finest scales and is smoothing along the edge direction, which avoids noisy edges. So we used Sobel operator get the edge image.

The skin colour algorithm can be improved by using edge information and classifying e skin pixels. A skin pixel in addition to exceeding the skin threshold for Histogram should so have a gradient less than a certain threshold called edge threshold.

The perimeter is calculated by the equation:

$$perimeter = \#\{k \mid (r_{k+1}, c_{k+1}) \in N_4(r_k, c_k)\}$$
$$+\sqrt{2}\,\#\{k \mid (r_{k+1}, c_{k+1}) \in N_8(r_k, c_k) - N_4(r_k, c_k)\}$$

Where k+1 is the modulo K; N4 is the 4-connected neighbours and N8 is the 8-connected neighbours

The circularity of each object can be calculated as follow:

$$circularity = \frac{|Perimeter|^2}{Area}$$

The circularity describes how circular the shape of an object is. If this value is small, then the object will look like a circle. This feature was chosen since the human face is likely to be circular. A threshold value for the circularity was set to work on different images such that the system will locate the faces and bound them in boxes.

APPLICATIONS:

- Face detection is used in biometrics, often as a part of (or together with) a facial recognition system.
- It is also used in video surveillance, human computer interface and image database management.
- Face detection is also useful for selecting regions of interest in photo slideshows that use a pan-and-scale ken-burn effect.

The various thresholds used in the algorithm are shown in the following table. These thresholds are arrived at after some experimentation.

Type of threshold	Value
SkinThreshold	0.1
EdgeThreshold	125
PercentageThreshold	55
Tolerance	0.65

5.3 CIRCULARITY CALCULATION

Once a connected component analysis is applied on the modified skin binary image, we can get the number of objects and the binary labelled matrix. Any features for all objects can be calculated. Our interests are the area and the perimeter for all objects.

The area of each object is defined as follow:

$$Area = \sum_{(r,c)\in R} 1$$

Where r and c are the row and column of image R, respectively. It is the summation of all the 1's in each connected object.

CHAPTER 6

RESULTING IMAGES

Fig 6.1 Input image

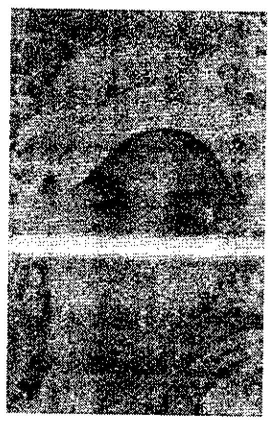

Fig 6.2 Colour Conversion

Fig 6.5 Edge detected

Fig 6.6 Modified skin image

Fig 6.7 Detected face

BIBLIOGRAPHY

[1] K. Sandeep and A.N. Rajagopalan "Human Face Detection in Cluttered Colour Images Using Skin Colour and Edge Information". Computational Geometry: Theory and Applications, vol. 17, pp. 135- 163, 2000.

[2] F. Tomaz, T. Candeias and H. Shahbazkia. "Improved Automatic Skin Detection in Colour Images" .Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney.

[3] Sanjay Kr. Singh, D. S. Chauhan and Mayank Vatsa. "A Robust Skin Colour Based Face Detection Algorithm". Tamkang Journal of Science and Engineering, Vol. 6, No. 4, pp. 227-234 (2003).

[4] Aouatif Amine, Sanaa Ghouzali and Mohammed Rziza. "Face Detection in Still Colour Images Using Skin Colour Information". In Int'l Conference on Pattern Recognition, 2002.

[5] S. Ravichandran and B. Ananthi. "Colour Skin Segmentation Using KMeans Cluster". International Journal of Computational and Applied Mathematics ISSN 1819-4966 Volume 4 Number 2 (2009), pp. 153– 157.

[6] J Yao-Jiunn Chen, Yen-Chun Lin. "Simple Face-detection Algorithm Based on Minimum Facial Features". The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON). Nov. 5-8, 2007, Taipei, Taiwan.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

The detected face is bounded by a green box in the final output image. The above algorithm was implemented in MATLAB by giving a test image which was given as an input to the algorithm. An algorithm of face detection system based on skin colour and circularity was implemented and done successfully. The proposed algorithm can detect faces for different races.

FUTURE SCOPE

As an extension to the detection, recognition of faces was also done using the crop images from the detected output, using PCA (Principal Component Analysis). The efficiency of the face detection was found to be near about 90% while that of face recognition using the detected crop images was found to be between 80-90%. The run time error for face detection was 35 to 40 second while for face recognition it was between 60 to 90 seconds. Hence it can be concluded that the present algorithm demonstrate the super performance with respect to speed, zero repeats, low false positive rate and high accuracy.