M.Tech. (Computer Science) Dissertation Series

Face detection in color images

M.Tech. Dissertation Report

a dissertation submitted in partial fulfillment of the requirements for the M.Tech.(computer Science) degree of the Indian Statistical Institute

By

Vishnuvardhan Reddy.B

M.Tech-Computer Science

Roll No: CS0726

under the supervision of

Prof. Bhabatosh Chanda
Electronics and Communication Science Unit
ISI, Kolkata

INDIAN STATISTICAL INSTITUTE

203, Barrackpore Trunk Road,

Kolkata-700108, West Bengal, India.

Indian Statistical Institute

Certificate of Approval

This is to certify that the thesis entitled "Face detection in color images" by Vishnuvardhan Reddy.B towards partial fulfillment for the degree of M.Tech. in Computer Science at Indian Statistical Institute, Kolkata, embodies the work done under my supervision.

<div align="right">

(Prof. B. Chanda)

ECSU- Indian Statistical Institute

Kolkata.

</div>

Date:

ACKNOWLEDGMENT

Date:

Vishnuvardhan Reddy.B

M.Tech(CS)

Indian Statistical Institute

Kolkata

Abstract

This paper presents a technique for automatically detecting human faces in digital color images. The system relies on a two step process which first detects regions which are likely to contain human skin in the color image and then extracts information from these regions which might indicate the location of a face in the image. The skin detection is performed using a skin filter which relies on color information. The face detection is performed on a gray scale image containing only the detected skin areas. A combination of thresholding and mathematical morphology are used to extract object features that would indicate the presence of a face. The face detection process works predictably and fairly reliably, as experimental results show.

Contents

Chapter 1

Introduction

Face detection is concerned with finding whether or not there are any faces in a given image and, if present, return the image location and content of each face. This is the first step of any fully automatic system that analyzes the information contained in faces (e.g., identity, gender, expression, age and pose).

The face detection problem is challenging as it needs to account for all possible appearance variation caused by change in illumination, facial features, occlusions, etc. In addition, it has to detect faces that appear at different scale, pose, with in-plane rotations. In spite of all these difficulties, tremendous progress has been made in the last decade and many systems have shown impressive real-time performance. The recent advances of these algorithms have also made significant contributions in detecting other objects such as humans/pedestrians, and cars.

Over the past ten years face detection has been thoroughly studied in computer vision research for mainly two reasons. First, face detection has a number of interesting applications. It can be part of a face recognition system, a surveillance system and security control systems, content-based image retrieval, video conferencing and intelligent human-computer interfaces. Second, faces form a class of visually similar objects which simplifies the generally difficult task of object detection. Now a days face detection is used in biometrics, some recent digital cameras use face detection for auto focus. Also detection of faces in a digital image has gained much importance in the last decade, with applications in fields such as law enforcement, security and image database management.

As face detection is the first step of any face processing system, it finds numerous applications in face recognition, face tracking, facial expression recognition, facial feature extraction, gender classification, clustering, attentive user interfaces, digital cosmetics, biometric systems, to name a few. In addition, most of the face detection algorithms can be extended to recognize other objects such as cars, humans, pedestrians, and signs, etc.

1.1 Issues in Face Detection

Face detection is not straight forward because it has lots of variations of image appearance, such as

1. Pose variation (front, non-front).

2. Occlusion.

3. Image orientation.

4. Illuminating condition.

5. Facial expression.

1.2 Brief review of related work

Many novel methods have been proposed to resolve each variation listed above. For example, the template-matching methods are used for face localization and detection by computing the correlation of an input image to a standard face pattern. The feature invariant approaches are used for feature detection of eyes, mouth, ears, nose, etc. The appearance-based methods are used for face detection with eigen face, neural network, and information theoretical approach. Nevertheless, implementing the methods altogether is still a great challenge. These approaches utilize techniques such as machine learning, (deformable) template matching, Hough transform, motion extraction, and color analysis. A recent statistical approach extends the detection of frontal faces to profile views by training two separate classifiers. Model-based approaches are widely used in tracking faces and often assume that the initial locations of faces are known. Skin color provides an important cue for face detection. However, the color-based approaches face difficulties in robust detection of skin colors in the presence of complex background and variations in lighting conditions.

1.3 Objective of the work

Human face detection has become a major field of interest in current research because there is no deterministic algorithm to find face(s) in a given image. Further the algorithms that exist are very much specific to the kind of images they would take as input and detect faces. The problem is to detect faces in the given colored photograph. Here we have restricted ourselves to detect faces from images taken under controlled environment. Such situations are common for using ATM, having access to secured area and like, where we have frontal pose of face image against fixed, usually uniform, background. However, the algorithm is designed to be handle more general cases to ensure robustness.

The goal of this project is to take a color digital image and detect the location of the face in the image. This project represents a face detection technique mainly based on the color model, growcut technique and template matching methods. The proposed approach consists of three parts: a human skin segmentation to identify probable regions corresponding to human faces; growcut technique to get rid the existing non-skin pixels, if any from the skin segments; and template matching to locate the faces in the probable region defined earlier.

1.4   Organization of the report

The following chapters describe the various techniques used to detect the faces. Chapter 2 describes the skin segmentation, chapter 3 describes the morphological processing after skin segmentation, chapter 4 describes the growcut technique,which is used to eliminate most of non-skin pixels from skin segments, chapter 5 describes template matching, and in sixth chapter we have presented the experimental results.

Chapter 2

Skin Segmentation

The first step in this face detection algorithm is skin segmentation,which is used to reject non-skin pixels of the image as much as possible. Since the major part of the images consist of non-skin color pixels. Here we have segmented the image based on skin color. We have converted the given image from RGB space to YCbCr space. That is because RGB components are subject to the lighting conditions thus the face detection may fail if the lighting condition changes. In YCbCr space Y' is the luma component and Cb and Cr are the blue-difference and red-difference chroma components.

2.1 Color Transformation

The main advantage of converting the image to the YCbCr domain is that influence of luminosity can be removed during our image processing. Thus the effect of the varying illumination due to improper lighting arrangement in the environment can be minimized.

In the RGB domain, each component of the picture (red, green and blue) has a different brightness. However, in the YCbCr domain all information about the brightness is given by the Y component, since the Cb (blue) and Cr (red) components are independent from the luminosity. The following formula is used to convert the RGB image to YCbCr image.

$$Y = 0.299 * R + 0.587 * G + 0.114 * B \qquad (2.1)$$

$$Cb = 0.169 * R - 0.331 * G + 0.5 * B + 128 \qquad (2.2)$$

$$Cr = 0.5 * R - 0.419 * G - 0.082 * B + 128 \qquad (2.3)$$

2.2   Development of skin color model

In order to segment human skin regions from non-skin regions based on color, we need a reliable skin color model that is adaptable to people of different skin colors and to different lighting conditions. In the section, we will describe a model of skin color in the chromatic color space for segmenting skin.

Beginning with a color image, the first stage is to transform it to a skin-likelihood image. This involves transforming every pixel from RGB representation to chroma representation and determining the likelihood value based on the equation given in the previous section.

Many studies have come to the conclusion that pixels belonging to skin region have similar Cb and Cr values. It is also conjectured that the perceived difference in skin color cannot be differentiated from the chrominance information of that region. The fairness or darkness of the skin is characterized by the difference in the brightness of the color, which is determined by Y. Therefore solely using Cb and Cr will generalize the filter for people of all skin colors.

We have implemented a skin color model with color statistics gathered from YCbCr color space. Studies have found that pixels belonging to skin region exhibit similar Cb and Cr values. Furthermore, it has been shown that skin color model based on the Cb and Cr values can provide good coverage of different human races. The thresholds be chosen as [Cb1, Cb2] and [Cr1, Cr2], a pixel is classified to have skin tone if the values [Cb, Cr] fall within the thresholds. The skin color distribution gives the face portion in the color image. This algorithm is also having the constraint that the image should be having only face as the skin region.

2.3   Selection of decision threshold

First crop out the smooth skin regions from each training image(RGB image) and then convert them to YCbCr color space. If a random variable X follows normal distribution with mean Mu and standard deviation Sigma then the 2-sigma limits are defined as below.

$$Mu - 2 * Sigma \ and \ Mu + 2 * Sigma$$

It is clear that Cb and Cr follows normal distribution with some mean and standard deviation, we use 2-sigma limits as the threshold for decision.

However, it is important to note that the detected regions may not necessarily correspond to skin. It is only reliable to conclude that the detected region have the same color as that of the skin. The important point here is that this process can reliably point out regions that do not have the color of the skin and such regions would not need to be considered anymore in the face finding process. The algorithm is as follows.

- **Algorithm:**

Step1: Collect some skin segments from the training images (RGB images).

Step2: Convert all these skin segments form RGB space to YCbCr space by using above mentioned formula.

Step3: Get Cb values of all segments into one array and Cr values of all segments into another array.

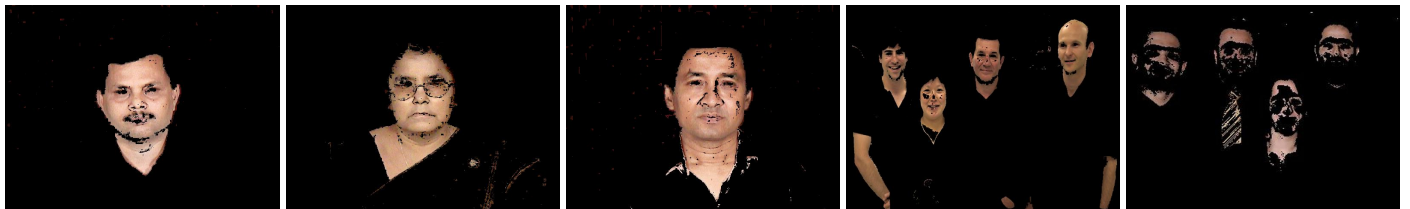**Step4:** Find mean and standard deviation of the above two arrays.

Step5: From the above mean and standard deviation calculate 2-sigma limits for Cb and Cr components, which are required thresholds.

2.4   Skin segmented results

Figure 2.1: Given RGB images.

Figure 2.2: Skin segmented images for the above RGB images.

Chapter 3

Morphological Processing

As can be seen from above, most of the skin pixels have been correctly classified. Background that is close to skin color will have a chance of being wrongly classified too, but further processing by correlation should be able to iron out the false hits.

Here we are further processing the above obtained skin segmented images with some binary morphological operations.The further processing steps namely holes filling, erosion, and dilation are described below.

3.1 Binary Morphological Operations

After extracting the pixels based on their color values, we performed the binary morphological operation "fill holes" on the binary images corresponding to the skin segmented images for removing holes in the skin segmented images.

After filling holes we are forming a binary image corresponding to the holes filled skin segmented image consisting at most two largest connected components. This can be achieved by putting a threshold on the density of the connected component in the holes filled skin segmented images.For the images contain multiple faces, just we have considered all the connected components satisfying the threshold.

Here we have chosen the threshold as average value of the density of the all the connected components in the holes filled skin segmented image. Now we performed the binary morphological operation "erosion" on the above obtained binary image to erode it. Here the structuring element (SE1) of size 6 we have used to erode the obtained binary image is given below.

$$SE1 = \begin{array}{ccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

For the images having multiple faces,we have considered the structuring element of size 4.

Similarly we are dilating the above obtained holes filled images by using the binary morphological operation "dilate " with the large structuring element (SE2) of size 15.

The algorithm is described here.

- **Algorithm:**

Step1: Fill the holes in the binary images corresponding to the skin segmented images obtained from the previous chapter.

Step2: Form the binary images consisting at most two largest connected components satisfying the threshold corresponding to the images obtained in step1.
For the images having multiple faces we have considered all the connected components satisfying the threshold.

Step3: Erode the images obtained in step2 with the structuring element SE1.

Step4: Dilate the images obtained in step1 with the structuring element SE2.

3.2 Binary Morphological Operation Results

Figure 3.1: Skin segmented images obtained in chapter 2.

Figure 3.2: Holes filled images for Figure 3.1.

Figure 3.3: Eroded results for Figure 3.1.

Figure 3.4: Dilated images for Figure 3.1.

Chapter 4

Growcut algorithm

Growcut is an interactive segmentation algorithm. It uses Cellular Automaton as an image model. Automata evolution models segmentation process. Each cell of the automata has some label (in case of binary segmentation - 'object', 'background' and 'empty'). During automata evolution some cells capture their neighbors, replacing their labels.

Given a small number of user-labeled pixels, the rest of the image is segmented automatically by a Cellular Automaton. The process is iterative, as the automaton labels the image, user can observe the segmentation evolution and guide the algorithm with human input where the segmentation is difficult to compute. Here user specifies certain image pixels (we will call them seed pixels) that belong to objects, that should be segmented from each other. The task is to assign labels to all other image pixels automatically, preferably achieving the segmentation result the user is expecting to get.

4.1   Skin segmentation using growcut method

Here we are using the growcut algorithm for finding the skin segments in the given RGB image. In chapter 2 we have found the skin segmented images for the given RGB images. There the skin segmented images may have some of non-skin pixels also. For get riding those non-skin pixels we have used the growcut algorithm.

For this algorithm we are giving the given RGB image as one of the input image, the corresponding eroded binary image obtained in chapter 3 as an another input image, and the corresponding dilated binary image obtained in chapter 3 as the third input image. We are labeling all the pixels in the RGB input image as follows.

- Give object label for all the pixels corresponding to gray level 1 in eroded binary image(i.e: second input image).

- Give background label for all the pixels corresponding to gray level 0 in dilated binary image(i.e: third input image).

- Give empty label for rest of all the pixels.

That is all the pixels in eroded image having gray level 1 and all the pixels in dilated image having gray level 0 will act as the user specified seed pixels for the growcut algorithm. Because the eroded input image corresponding to the image having single face, may have more than one connected component, in such cases after applying growcut algorithm we may left with more than one object labeled connected component.So after applying the growcut algorithm we are considering the largest object labeled connected component as the probable skin region in the given RGB input image and we are showing that with a rectangular bounding box.

In case of images having multiple faces, just we have considered all the object labeled connected components obtained by using growcut algorithm as the probable skin regions and we have shown them by using rectangular boxes.

The growcut algorithm is as follows.

- **Algorithm:**

Step1: Read the given RGB image, eroded binary image obtained from chapter 3, and dilated binary image obtained from chapter 3 as the input images.

Step2: Give the object label for all the pixels in RGB image corresponding to the gray level 1 in eroded image, background label for all the pixels corresponding to the gray level 0 in dilated image, and empty label for all the rest of the pixels.

Step3: Give strength 1 for all pixels having either object label or background label and strength 0 for all the empty labeled pixels.

Step4: For every pixel p in the image do
  copy previous state: labels_ new = labels
  strength_ new = strength
  For all neighbors q of p do
    if(attack_ force*strength(q)> strength(p))
      labels_ new(p) = labels(q)
      strength_ new(p) = strength(q)

Step5: Repeat step 4 until no more changes in the label matrix.

4.2 Growcut algorithm results

Figure 4.1: Given RGB images.

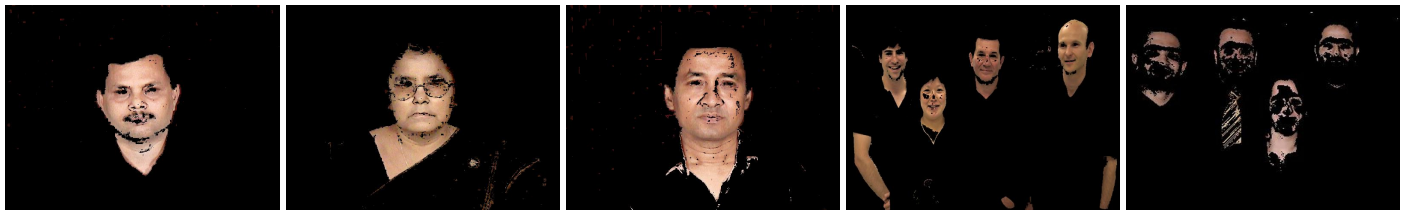Figure 4.2: Skin segmented images obtained in chapter 2.

Figure 4.3: Skin segmented images obtained by using growcut technique.

Figure 4.4: Probable skin area shown by using rectangular box.

Chapter 5

Template Matching

One of the most important characteristics of this method is that it uses a human face template to take the final decision of determining if a skin region represents a face.This section shows how to do the matching between the part of the image corresponding to the skin region and the template face. Once individual probable face images are found, template matching is used as the final detection scheme for faces. The idea of template matching is to perform covariance with a part of the image and a template that is representative of the image.

5.1 Generating face template

The first step is to crop out the faces from each training image. After the images are acquired, re-size all the cropped faces according to the corresponding scale factor which is obtained as follows.

- Find the nose tip and eye ray for all the cropped faces.

- Find the distance between the eye ray to nose tip in each cropped face. Call this distance as d_i for the ith cropped face respectively.

- Find the average distance between the eye ray to nose tip as $D = \Sigma_{i=1}^{n} d_i$.

- Now the scale factor for the ith image is given as $\alpha_i = \dfrac{D}{d_i}$.

Now pad up with zero rows and zero columns to all the re-sized cropped faces such that all the nose tips co-inside and have same size, and find the average face for the above obtained faces by using the following formula.

$$Average\_Face = \tfrac{1}{n}\Sigma_{i=1}^{n}F_i$$

where F_i is the ith face obtained above and n is the number of faces.

Extract a rectangular box region from this obtained average face such that two eyes and mouth retain in that rectangular box and consider this as the template face for the given training image set.

In our case the template is chosen by averaging 12 full frontal view faces of males and females having no facial hair. Cropped faces after scaling are shown below.

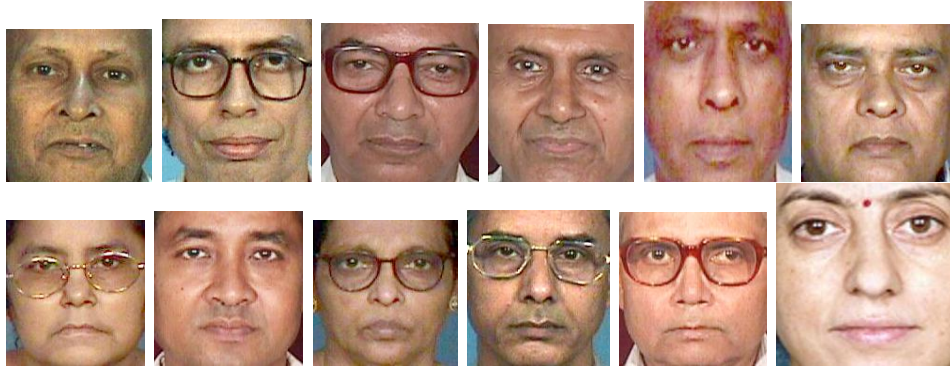Figure 5.1: Cropped faces from the training data set.

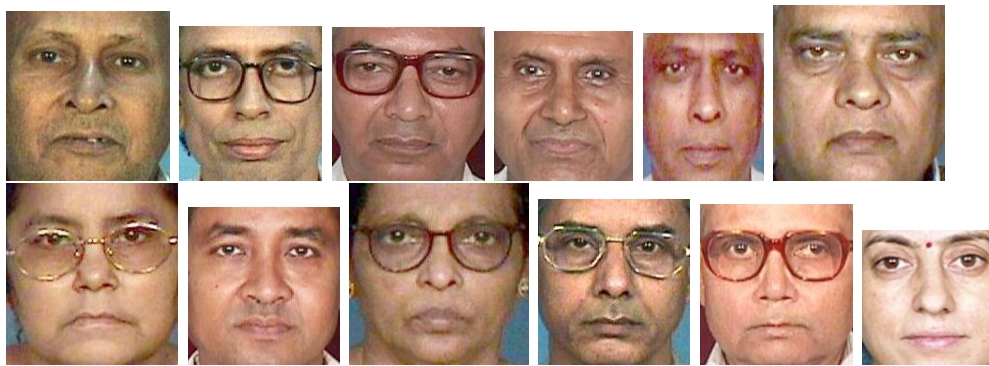Figure 5.2: Re-sized images by using scale factor.

Figure 5.3: Images padded with zero rows and/or zero columns such that nose tips co-inside and have same size.

Algorithm for calculating template face is given below.

- **Algorithm:**

Step1: From training images crop out 12 frontal view faces of males and females wearing no glasses and having no facial hair.

Step2: Find the eye ray and nose tip for each cropped face.

Step3: Find the distance between eye ray and nose tip for all the faces, and re-size the cropped faces according to their scale factors.

$$D = \Sigma_{i=1}^n d_i.$$
$$\alpha_i = \frac{D}{d_i}$$

where α_i and d_i are the scale factor and the distance between eye ray and nose tip for the ith cropped face respectively.

Step4: Pad up with zero rows and zero columns to the faces obtained in step 3 such that all the nose tips co-inside and have the same size.

Step5: Find the average of all the faces obtained in step 4 by using the formula given below.

$$Average\_Face = \tfrac{1}{n}\Sigma_{i=1}^n F_i$$

where F_i is the ith face in Figure 5.3 and n is the number of faces.

Figure 5.4: The obtained template image for the above faces.

5.2 Matching the face template

Here we compute the cross-correlation value between the part of the image corresponding to the skin region and the template face obtained in section 5.1. Obtain the part having maximum correlation with the average template.Correlation of two images is as follows.

$$Correlation = \frac{\Sigma_{i=1}^m \Sigma_{j=1}^n (A_{i,j} - \overline{A}).(B_{i,j} - \overline{B})}{\sqrt{(\Sigma_{i=1}^m \Sigma_{j=1}^n (A_{i,j} - \overline{A})^2).(\Sigma_{i=1}^m \Sigma_{j=1}^n (B_{i,j} - \overline{B})^2)}}$$

where $\overline{A}$ and $\overline{B}$ are the mean of target and template images respectively.

For the images having multiple faces we have re-sized the obtained template to the minimum value among the row value and column value of the probable face region.After applying

template matching we are putting a threshold on the number of pixel corresponding to high correlated part of the probable face region for get riding the non-face boxes.

This part of the image represents the detected face. The coordinates of this part in the given image is determined and a rectangle is drawn in the original color image.
The algorithm is as follows.

- **Algorithm:**

Step1: The template face has to be positioned in the same coordinates as the face candidate regions and calculate the correlation.

Step2: Consider the region in face candidate with high correlation as the detected face.

Step3: Find coordinates and draw a rectangular bounding box in original images.

5.3 Template Matching results

Figure 5.5: Given RGB images.

Figure 5.6: Probable skin area shown by using rectangular box.

Figure 5.7: Detected face after Template Matching.

Chapter 6

Experimental Results

The performance of our detection system is verified on ISI Employee Image Database. This database, there are 1000 different images of each of fixed back ground and different illumination conditions. The size of each image is different. Each image contains a frontal face with out any rotation.The algorithm is also tested for the images having multiple faces with complex back ground and having frontal and non-frontal faces.

6.1   Given RGB images
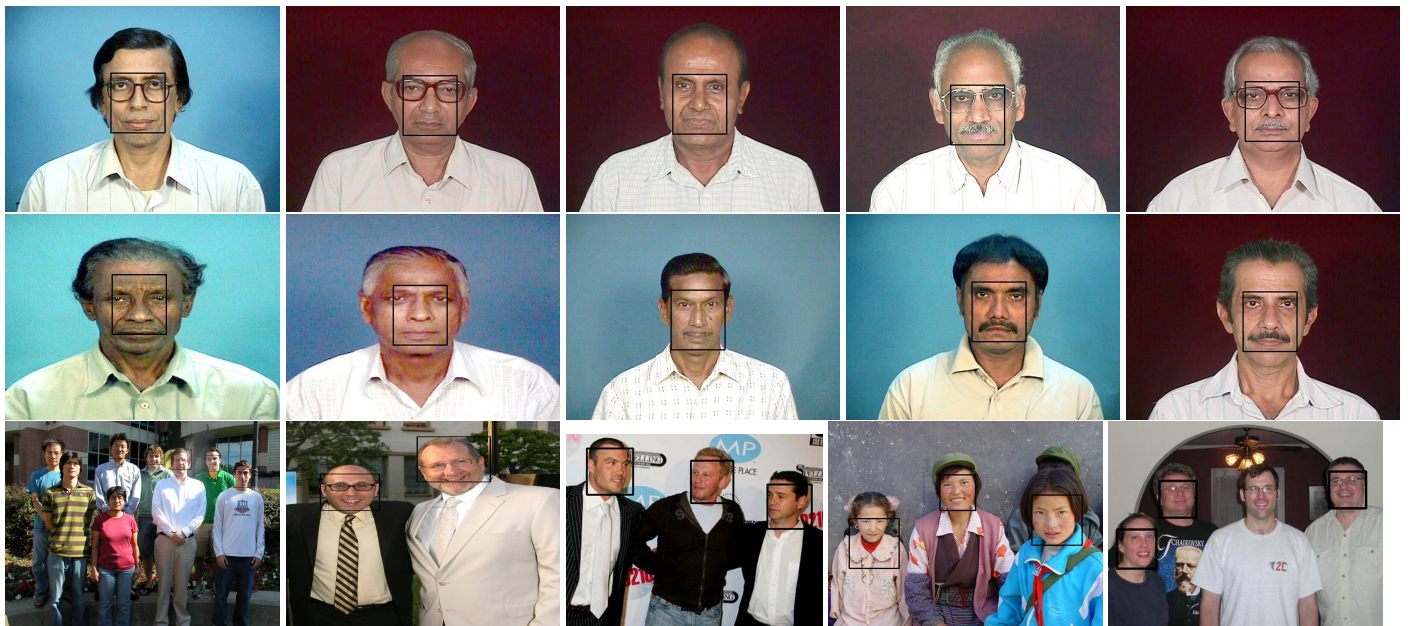
6.2 Results of Skin Segmentation

6.3 Results of Growcut algorithm

6.4   Results for probable face region

6.5   Results of Template Matching

Chapter 7

Conclusion and Future work

In this report we have presented an approach, which can efficiently detect the faces from a color image. Growcut technique will refine the skin pixels from the results of the skin segmentation. There after we are searching the faces in the probable face regions by using template matching technique. Here we are doing the exhaustive search.Restricted search of the template in the face candidate for the best match of the face is an efficient search compare to the exhaustive search. Even the search can not find the highest correlated region of the face candidate image with the template, it will find the region nearest to the highest correlated region.

The proposed method can be further improved, if we use efficient techniques for refinement of skin pixels. We can improve the performance of template matching by using restricted search technique for searching the faces in the face candidates.

References

[1] I. Craw, D. Tock, and A. Bennett, Finding face features, Proc.of 2nd European Conf. Computer Vision. pp. 92-96, 1992.

[2] A. Lanitis, C. J. Taylor, and T. F. Cootes, An automatic face identification system using flexible appearance models, Image and Vision Computing, vol.13, no.5, pp.393-401, 1995.

[3] T. K. Leung, M. C. Burl, and P. Perona, Finding faces in cluttered scenes using random labeled graph matching, Proc. 5th IEEE intl Conf. Computer Vision, pp. 637-644, 1995.

[4] B. Moghaddam and A. Pentland, Probabilistic visual learning for object recognition, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no.7. pp. 696-710, July, 1997.

[5] M. Turk and A. Pentland, Eigenfaces for recognition, J. of Cognitive Neuroscience, vol.3, no. 1, pp. 71-86, 1991.

[6] M. Kirby and L. Sirovich, Application of the Karhunen-Loeve procedure for the characterization of human faces, IEEE Trans. Pattern Analysis and Machine Intelligence, vol.12, no.1, pp. 103-108, Jan. 1990.

[7] I. T. Jolliffe, Principal component analysis, New York: Springer-Verlag, 1986.

[8] T, Agui, Y. Kokubo, H. Nagashi, and T. Nagao, Extraction of face recognition from monochromatic photographs using neural networks, Proc. 2nd Intl Conf. Automation, Robotics, and Computer Vision, vol.1, pp. 18.81-18.8.5, 1992.

[9] O. Bernier, M. Collobert, R. Feraud, V. Lemaried, J. E. Viallet, and D. Collobert, MULTRAK:A system for automatic multiperson localization and tracking in real-time, Proc, IEEE. Intl Conf. Image Processing, pp. 136-140, 1998.

[10] A. J. Colmenarez and T. S. Huang, Face detection with information-based maximum discrimination, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 782-787, 1997.

[11] M. S. Lew, Information theoretic view-based and modular face detection, Proc. 2nd Intl Conf.Automatic Face and Gesture Recognition, pp. 198-203, 1996.

[I2] D. Maio and D. Maltoni, Real-time face location on grayscale static images, Pattern Recognition, vo1.33, no. 9, pp. 1525-1539, Sept. 2000.

[I3] H. Schneiderman and T. Kanade, A statistical method for 3D object detection applied to faces and cars, IEEE CVPR, June 2000.