Fuzzy Face Detection

Major Project submitted in partial fulfillment of the requirements for the award of degree of

Master of Technology

Information Systems

Submitted By:

Kartik Sain

(05/IS/09)

Under the Guidance of:

Prof. O. P. Verma

(HOD, IT Department)

Department of Information Technology

Delhi Technological University

New Delhi, 2011

CERTIFICATE

This is to certify that **Mr.Kartik Sain (05//IS/09)** has carried out the major project titled "**Fuzzy Face Detection**" as a partial requirement for the award of Master of Technology degree in Information Systems by Delhi Technological University.

The major project is a bonafide piece of work carried out and completed under my supervision and guidance during the academic session **2009-2011**.

The matter contained in this report has not been submitted elsewhere for the award of any other degree.

(Project Guide)

Prof. O.P. Verma
Head Of Department
Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I express my gratitude to my major project guide ***Prof. O.P. Verma***, HOD, IT Dept.,DelhiTechnologicalUniversity, for the valuable support and guidance he provided in making this project. It is my pleasure to record my sincere thanks to my respected guide for his constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

Kartik Sain

05/IS/09
M.Tech (Information System)

Department of IT

e-mail: kartiksain@gmail.com

Abstract

Face Detection is an important part of face recognition. The face recognition is done after detecting a face in the still image or video. We proposedan approach for face detection in still color images which is capable of detecting face invariant to pose variations in the color image. The faces are detected in still image after segmentation of skin pixels in the input image. The segmentation of input image is done using Fuzzy C-means clustering technique. The clustering of modified color space Y'Cb'Cr' is applied on Cr' channel. The clusters those qualify for skin pixels are selected and rest are neglected. These cluster are selected if there minimum and maximum value lie between the thresholds. Detecting skin pixels gives the region of faces and other body parts and thus make easy to detect face in the image. These regions which are not faces are eliminated by fuzzy rules in the proposed approach and if some non faces are left over by these rules then a decision rule gives final output as it check the presence of eyes in the current region. The eye regions are detected in each left over region after applying fuzzy rules of elimination. After detecting faces the comparison is done with other standard techniques of face detection.

Table of Contents

Chapter 1

Biometrics

1.1 Introduction

In an increasingly digital world, reliable personal authentication has become an important human computer interface activity. National security, e-commerce, and access to computer networks are some examples where establishing a person's identity is vital. Existing security measures rely on knowledge-based approaches like passwords or token-based approaches such as swipe cards and passports to control access to physical and virtual spaces. Though ubiquitous, such methods are not very secure. Tokens such as badges and access cards may be shared or stolen. Passwords and PIN numbers may be stolen electronically. Furthermore, they cannot differentiate between authorized user and a person having access to the tokens or knowledge.

Biometrics such as fingerprint, face and voice print offers means of reliable personal authentication that can address these problems and is gaining citizen and government acceptance. Biometrics is the science of verifying the identity of an individual through physiological measurements or behavioral traits. Since biometric identifiers are associated permanently with the user they are more reliable than token or knowledge based authentication methods.

Biometrics offers several advantages over traditional security measures. These include

***Non-repudiation**:* With token and password based approaches, the perpetrator can always deny committing the crime pleading that his/her password or ID was stolen or compromised even when confronted with an electronic audit trail. There is no way in which his claim can be verified effectively. This is known as the problem of deniability or of 'repudiation'. However, biometrics is indefinitely associated with a user and hence it cannot be lent or stolen making such repudiation infeasible.

Accuracy and Security: Password based systems are prone to dictionary and brute force attacks. Furthermore, such systems are as vulnerable as their weakest password. On the other hand, biometric authentication requires the physical presence of the user and therefore cannot be circumvented through a dictionary or brute force style attack. Biometrics have also been shown to possess a higher bit strength compared to password based systems and are therefore inherently secure.

Screening*:* In screening applications, we are interested in preventing the users from assuming multiple identities (e.g. a terrorist using multiple passports to enter a foreign country). This requires that we ensure a person has not already enrolled under another assumed identity before adding his new record into the database. Such screening is not possible using traditional authentication mechanisms and biometrics provides the only available solution.

The various biometric modalities[1] can be broadly categorized as

Physical biometrics*:* These involve some form of physical measurement and include modalities such as face, fingerprints, iris-scans, hand geometry etc.

Behavioral biometrics*:* These are usually temporal in nature and involve measuring the *way* in which a user performs certain tasks. This includes modalities such as speech, signature, gait, keystroke dynamics etc.

Chemical biometrics*:* This is still a nascent field and involves measuring chemical cues such as odor and the chemical composition of human perspiration.

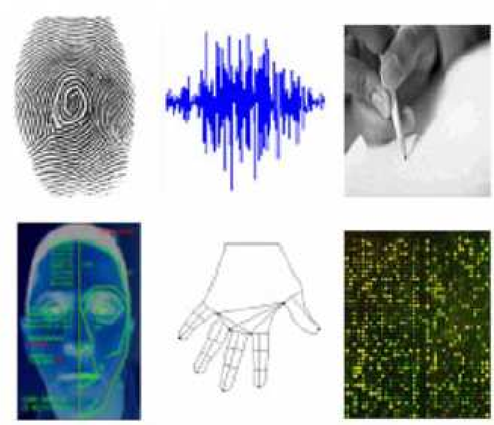


Figure 1.1: Various biometrics modalities: Fingerprints, speech, handwriting, face, hand geometry and chemical biometrics

It is also instructive to compare the relative merits and de-merits of biometric and password/cryptographic key based systems. Table 1.1 provides a summary of them.

Table 1.1: Comparison of biometric and password/key based authentication

Biometric Authentication	**Password/Key based authentication**
Based on physiological measurements or behavioral traits	Based on something that the user 'has' or 'knows'
Authenticates the user	Authenticates the password/key
Is permanently associated with the user	Can be lent, lost or stolen
Biometric templates have high uncertainty	Have zero uncertainty
Utilizes probabilistic matching	Requires exact match for authentication

Depending on the application, biometrics can be used for identification or for verification. In verification, the biometric is used to validate the claim made by the individual. The biometric of the user is compared with the biometric of the claimed individual in the database. The claim is rejected or accepted based on the match. (In essence, the system tries to answer the question, “Am I whom I claim I am?”)[1]. In identification, the system recognizes an individual by comparing his biometrics with every record in the database (In essence, the system tries to answer the question, “Who am I?”).

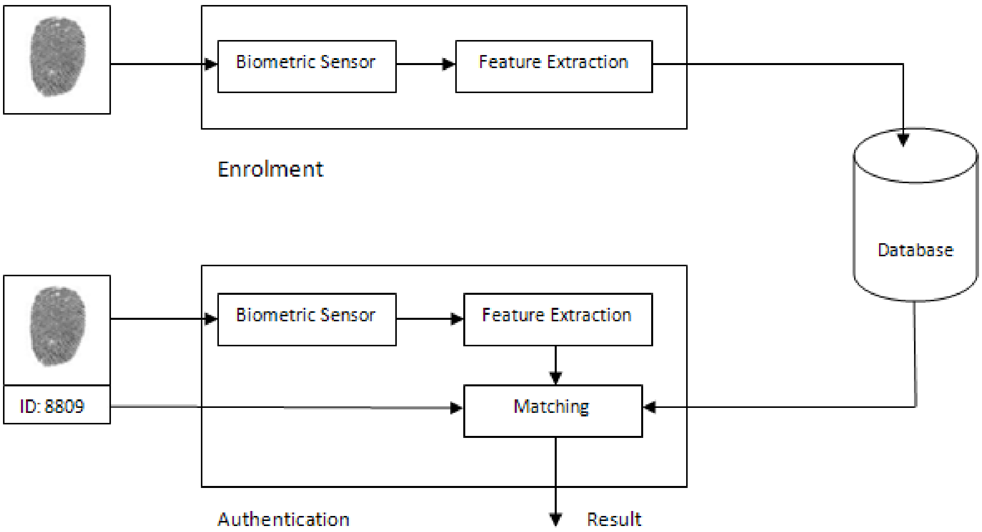


Figure 1.2: General architecture of a biometric system [1]

In general, biometric verification consists of two stages (Figure 1.2) (i) Enrollment and (ii) Authentication. During enrollment, the biometrics of the user is captured and the extracted features (template) are stored in the database. During authentication, the biometrics of the user is captured again and the extracted features are compared with the ones already existing in the database to determine a match. The specific record to fetch from the database is determined using the claimed identity of the user. The database itself may be central or distributed with each user carrying his template on a smart card.

1.2 Biometrics and Pattern Recognition

As recently as a decade ago, biometrics did not exist as a separate field. It has evolved through interaction and confluence of several fields. Fingerprint recognition emerged from the application of pattern recognition to forensics. Speaker verification evolved out of the signal processing community. Face detection and recognition was largely researched by the computer vision community. While biometrics is primarily considered as application of pattern recognition techniques, it has several outstanding differences from conventional classification problems as enumerated below

1. In a conventional pattern classification problem such as Optical Character Recognition (OCR) recognition, the number of patterns to classify is small (A-Z) compared to the number of samples available for each class. However in case of biometric recognition, the number of classes is as large as the set of individuals in the database. Moreover, it is very common that only a single template is registered per user.
2. The primary task in biometric recognition is that of choosing a proper feature representation. Once the features are carefully chosen, the act of performing verification is fairly straightforward and commonly employs simple metrics such as Euclidean distance. Hence the most challenging aspects of biometric identification involves signal and image processing for feature extraction.

3. Since biometric templates represent personally identifiable information of individuals, security and privacy of the data is of particular importance unlike other applications of pattern recognition.
4. Modalities such as fingerprints, where the template is expressed as an unordered point set (minutiae) do not fall under the category of traditional multi-variate/vectorial features commonly used in pattern recognition.

1.3 The Verification Problem

Here we consider the problem of biometric verification in a more formal manner. In a verification problem, the biometric signal from the user is compared against a single enrolled template. This template is chosen based on the claimed identity of the user. Each user *i* is represented by a biometric *Bi*. It is assumed that there is a one-to-one correspondence between the biometric *Bi* and the identity *i* of the individual. The feature extraction phase results in a machine representation (template) *Ti* of the biometric.

During verification, the user claims an identity *j* and provides a biometric signal *Bj*. The feature extractor now derives the corresponding machine representation *Tj* . The recognition consists of computing a similarity score *S* (*Ti*, *Tj*). The claimed identity is assumed to be true if the *S*(*Ti*, *Tj*) > *Th* for some threshold *Th*. The choice of the threshold also determines the trade-off between user convenience and system security.

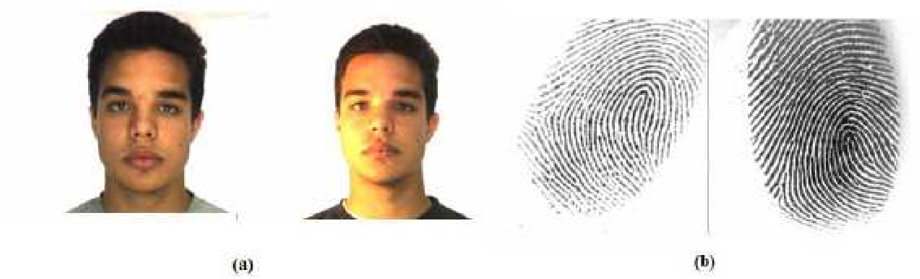


Figure 1.3: An illustration showing the intra user variation presents in biometric signals (a) Face (b) Fingerprints

1.4 Performance Evaluation

Unlike passwords and cryptographic keys, biometric templates have high uncertainty. There is considerable variation between biometric samples of the same user taken at different instances of time as shown Figure 1.3. Therefore the match is always done probabilistically. This is in contrast to exact match required by password and token based approaches. The inexact matching leads to two forms of errors:

• **False Accept** An impostor may sometime be accepted as a genuine user, if the similarity with his template falls within the intra-user variation of the genuine user.

• **False Reject** When the acquired biometric signal is of poor quality, even a genuine user may be rejected during authentication. This form of error is labeled as a ’false reject’.

The system may also have other less frequent forms of errors such as:

• **Failure to enroll (FTE)** It is estimated that nearly 4% of the population have illegible fingerprints. This consists of senior population, laborers who use their hands a lot and injured individuals. Due to the poor ridge structure present in such individuals, such users cannot be enrolled into the database and therefore cannot be subsequently authenticated. Such individuals are termed as ’goats’. A biometric system should have exception handling mechanism in place to deal with such scenarios.

• **Failure to authenticate (FTA)** This error occurs when the system is unable to extract features during verification even though the biometric was legible during enrollment. In case of fingerprints this may be caused due to excessive sweating, recent injury etc. In case of speech, this may be caused to due cold, sore throat etc. It should be noted that this error is distinct from False Reject where the rejection occurs during the matching phase. In FTA, the rejection occurs in the feature extraction stage itself.

1.5 System Errors

A biometric matcher takes two templates T and T' and outputs a score

$$S = S\,(T, T') \qquad \textbf{(1)}$$

which is a measure of similarity between the two templates. The two templates are identical if S(T, T')=1 and are completely different if S(T, T')=0. Therefore the similarity can be related to matching probability in some monotonic fashion. An alternative way to compute the match probability is to compute the matching distance D(T, T'). In this case, identical templates will have D(T, T')=0 and dissimilar templates should ideally have D(T, T')=∞. Usually a matcher outputs the similarity score $S(T, T') \in [0, 1]$.

Given two biometric samples, we construct two hypothesis[1]

• The null hypothesis $H0$: The two samples match.

• The alternate hypothesis $H1$: The two samples don't match.

The matching decides whether $H0$ is true or $H1$ is true. The decision of the matcher is based on some fixed threshold Th:

$$\text{Decide } H0 \text{ if } S\,(T, T') > Th \qquad \textbf{(2)}$$

$$\text{Decide } H1 \text{ if } S\,(T, T') \leq Th \qquad \textbf{(3)}$$

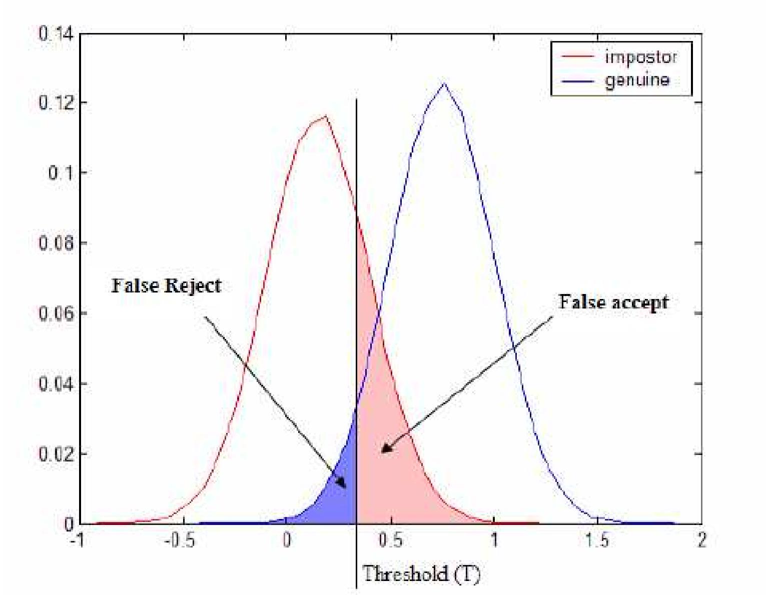


Figure 1.4: Genuine and Impostor Distributions [1]

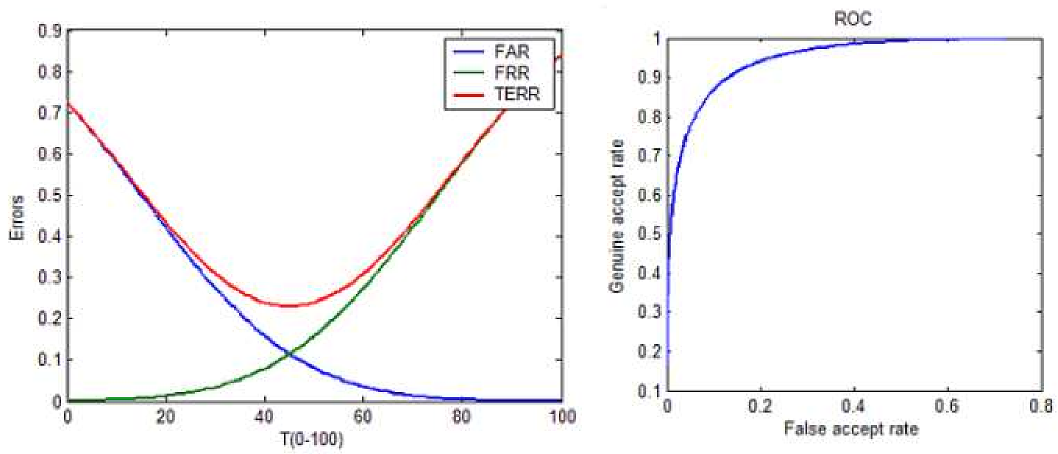
Due to variability in the biometric signal, the scores S(T, T') for the same person is not always unity and the score S(T, T') for different person is not exactly zero. In general the scores from matching genuine pairs are usually 'high' and the results from matching impostor pairs are usually 'low' which is shown in Figure 1.4.

Given that *pg* and *pi* represent the distribution of genuine and impostor scores respectively, the FAR and FRR at threshold T is given by

$$FAR(T) = \int_{T}^{1} pi(x)dx \qquad \textbf{(4)}$$

$$FRR(T) = \int_{0}^{T} pg(x)dx \qquad \textbf{(5)}$$

The corresponding ROC curve is obtained by plotting FAR (x-axis) vs 1-FRR(y axis). Figures 1.5 display a typical curve.



(i) (ii)

Figure 1.5: (i) Typical FAR and FRR vs threshold (ii) Typical ROC curve

Chapter 2
Face Detection

Computer vision, in general, aims to duplicate (or in some cases compensate) human vision, and traditionally, have been used in performing routine, repetitive tasks, such as classification in massive assembly lines. Today, research on computer vision is spreading enormously so that it is almost impossible to itemize all of its subtopics. Despite of this fact, one can list relevant several applications, such as face processing (i.e. face, expression, and gesture recognition), computer human interaction, crowd surveillance, and content-based image retrieval. All of these applications, stated above, require face detection, which can be simply viewed as a preprocessing step, for obtaining the “object”. In other words, many of the techniques that are proposed for these applications assume that the location of the face is pre identified and available for the next step.

Face detection is one of the tasks which human vision can do effortlessly. However, for computer vision, this task is not that easy. A general definition of the problem can be stated as follows: Identify all of the regions that contain a face, in a still image or image sequence, independent of any three dimensional transformation of the face and lighting condition of the scene. There are several methods issued for this problem and they can be broadly classified in two main classes, which are *feature-based*, and *image-based* approaches. Previous research has shown that both feature-based, and image-based approaches perform effectively while detecting upright frontal faces, whereas feature-based approaches show a better performance for the detection scenarios especially in simple scenes.

2.1 Face Detection: The Challenges

Face detection is the problem of determining whether a sub-window of an image contains a face. Looking from the point of view of learning, any variation which increases the complexity of decision boundary between face and non-face classes will also increase the difficulty of the problem. For example, adding tilted faces into the training set increases the variability of the set, and may increase the complexity of the decision boundary. Such complexity may cause the classification to be harder. There are many sources introducing variability when dealing with the face. They can be summarized as follows:

• ***Image plane variations*** is the first simple variation type one may encounter. Image transformations, such as rotation, translation, scaling and mirroring may introduce such kind of variations. Utilization of image pyramids with a sliding detector window is one common way to deal with such transformations for the input image. Variations in the global brightness, contrast level can also be expressed in the same category. Typical examples for such variations can be seen in Figure 2.1.

• ***Pose variations*** can also be listed under image plane variations aspects. However, changes in the orientation of the face itself on the image can have larger impacts on its appearance. Rotation in depth and perspective transformation may also cause distortion. The common way to deal pose variation is to isolate pose types (i.e. frontal, profile, rotated). Some examples for such pose variations are shown in Figure 2.1.

• ***Lighting variations*** may dramatically change face appearance in the image. Such variations are the most difficult type to cope with due to fact that pixel intensities are directly affected in a nonlinear way by changing illumination intensity or direction. For example, when using skin color as a feature for face detection, varying color temperature [2] of the light source may cause skin color filtering to fail. Some examples for lighting variations shown in Figure 2.1.

• ***Background variations*** is another challenging factor for face detection in cluttered scenes. Discriminating windows including a face from non-face is more difficult when no constraints exist on background. Most of the examples shown in Figure 2.1 have complex backgrounds which makes the face detection problem harder.

Figure 2.1: Examples of several variations

2.2 BACKGROUND OF FACE DETECTION

Over the last ten years, there has been a great deal of research concerning important aspects of face detection. Using generalized face shape rules, motion, and color information many segmentation schemes have been presented [3, 4, 5]. The use of probabilistic [6] and neural network methods [7] has made face detection possible in cluttered scenes and variable scales.

Face detection research can be heuristically classified in two main categories: feature-based approaches and image-based approaches.

According to the taxonomy in Figure 2.2, feature-based methods make explicit use of face knowledge and follow the classical detection methodology, in which low level features that are used prior to analysis mostly rely on heuristics or advance templates. The apparent properties of the face, such as skin color and face geometry, are used at different levels of the system. Since features are the main ingredients, these techniques are named as the feature-based approach. These approaches [8] have embodied the majority of interest in face detection research starting as early as the 1970s.

Taking the advantage of the current advances in pattern recognition theory, image-based approaches address face detection as a general pattern recognition problem. Partly due to well known work by [9], these approaches have attracted much attention in recent years, and have demonstrated remarkable results. According to the image-based methods, face detection is a two class (face, non-face) object recognition problem which uses pure image (intensity) representations instead of abstract feature representations.

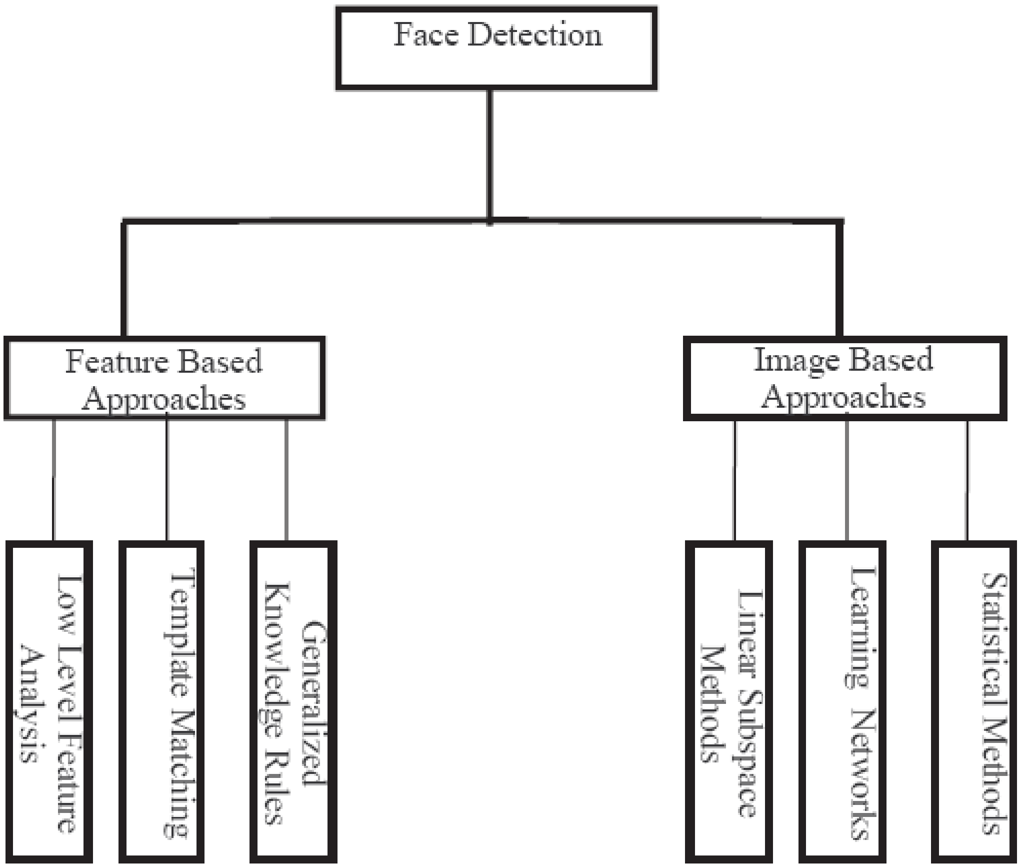


Figure 2.2: Face detection methods divided into main and sub categories [8].

2.2.1 Feature-Based Approaches

Most feature-based approaches share similar consecutive steps. Usually, the first step is to make pixel level eliminations by utilizing low level feature(s) e.g. skin color filtering, edge detection. Due to the low level properties, the result that is generated in the first step is ambiguous. In the second step, visual features which are not eliminated in the first step are organized within a global face knowledge or geometry. Using this feature analysis, feature ambiguities are reduced and the locations of face and facial features are determined. The final step may involve the use of templates or active shape models.

2.2.1.1 Low Level Feature Analysis

1) Edges

As a useful primitive feature in computer vision, edge representation was applied to early face detection system by Sakai et al. [10]. Later, based on this work, a hierarchical frame work was proposed by Craw et al. [11] to trace the human head line. This approach included a line follower which is implemented with a curvature constraint. Some more recent examples of edge-based techniques can be found in [18, 13, 14, 15].

Edge detection is the important step in edge-based techniques. For detecting edges, various types of edge detector operators are used. The Sobel operator is the most common filter among others for detecting edges [13, 16].

Also, a variety of 1st and 2nd derivatives (Laplacian) of Gaussians have also been used in some approaches [10, 17]. While a large scale Laplacian was used to obtain lines [10], and steerable and multi-scale-orientation filters are preferred in [17].

In a general face detector, which uses edge representation, labeling of the edges are needed. Then the labelled edges are tried to be matched against to a face model. Govindaraju [18] accomplishes this goal by labelling edges as the left side, hairline, or right side of a front view face and then tries to match these edges against a face model by using predetermined ratio of an ideal face.

2) Skin Color

Human skin color has been used and proven to be effective feature for face detection, and related applications. Although skin color differs among individuals, several studies have shown that the major difference exists in the intensity rather than the chrominance. Several color spaces have been used to label skin pixels including RGB [19, 20], NRGB (normalized RGB) [21, 22, 23], HSV (or HSI) [24, 25], YCrCb [26], CIE-XYZ [27], CIE-LUV [2]. Although, the effectiveness of the different color spaces is arguable, common point of all above works is the removal of intensity component. Terrillon et al. [28] recently presented a comparative study of several widely used color spaces for face detection. In this the authors compare normalized TSL (tint saturation-luminance), NRGB and CIE-xy chrominance spaces, and CIE-DSH, HSV, YIQ, YES, CIE-L* u* v* , and CIE L* a* b* chrominance spaces by modeling skin color distributions with either a single Gaussian or a Gaussian mixture density model in each space. In their face detection test, the normalized TSL space provides the best results, however, their general conclusion is about the most important criterion for skin color filtering, which is the degree of overlap between skin and nonskin distributions in a given space (and this is highly dependent on the number of skin and nonskin samples, available).

Color segmentation can basically be performed using appropriate skin color thresholds where skin color is modeled through histograms or charts [29, 12, 25]. More complex methods make use of statistical measures that model face variation within a wide user spectrum [30, 21, 22, 31]. For instance, Oliver et al. [22] and Yang et. al. [31] employs a Gaussian distribution to represent a skin color cluster, consisting of thousands of skin color samples, taken from the different human races. The Gaussian distribution is simply characterized by its mean and covariance matrix. Any pixel color of an input image is compared with the skin color model by computing the Mahalanobis distance [32]. This distance measure gives an idea of how close the pixel color resembles the skin color of the model.

Even though color information seems to be an efficient tool for identifying facial areas, the skin color models may fail when the spectrum (correlated color temperature) of light source varies significantly.

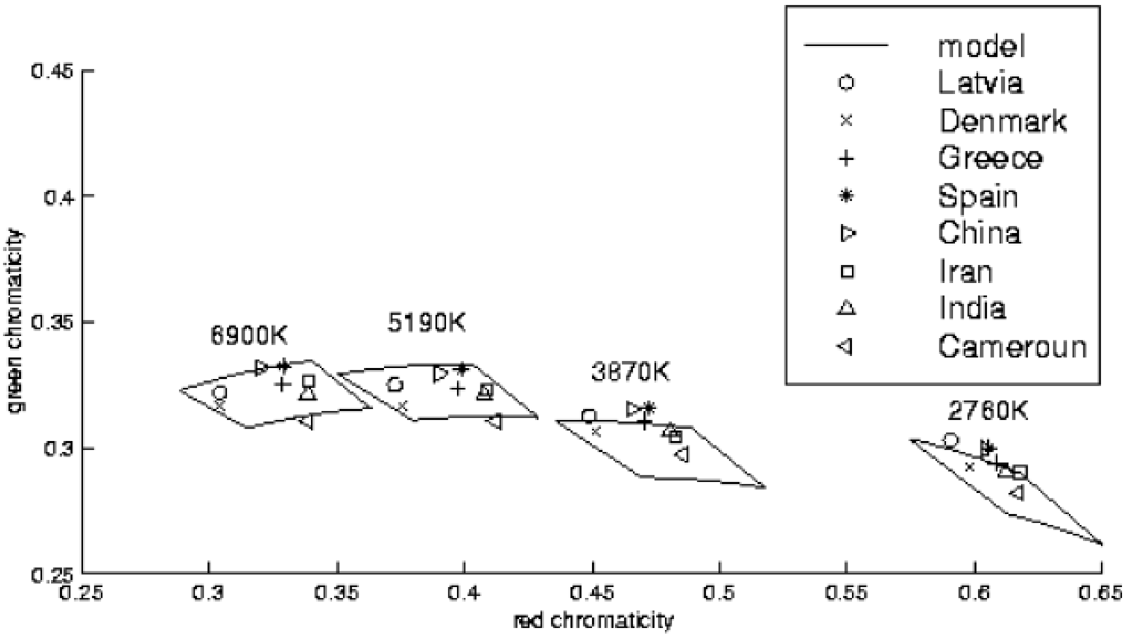


Figure 2.3: Skin color locus in NRGB space w.r.t light source CCT[2]

In addition, characteristics of acquisition device (specifically white balance) will also effect color transformation between the environment and the image. To address this problem, Storring et al. [2] modeled skin color based on the reflectance model of the skin, the camera parameters, and the spectrum of the light source. In particular, these researchers have estimated and verified skin color area in the chromaticity plane for different light sources, while the camera characteristics are given in Figure 2.3. An important conclusion of their work was the dependency of the skin color model on the spectrum of the light source and camera characteristics [2].

We have also studied the skin color information to utilize a skin color filter in the preprocessing step in face detection. However, in general, the skin color filters are constructed by using fixed boundaries (thresholds) for sample pixel distributions in color space. Illumination and camera parameters are omitted.

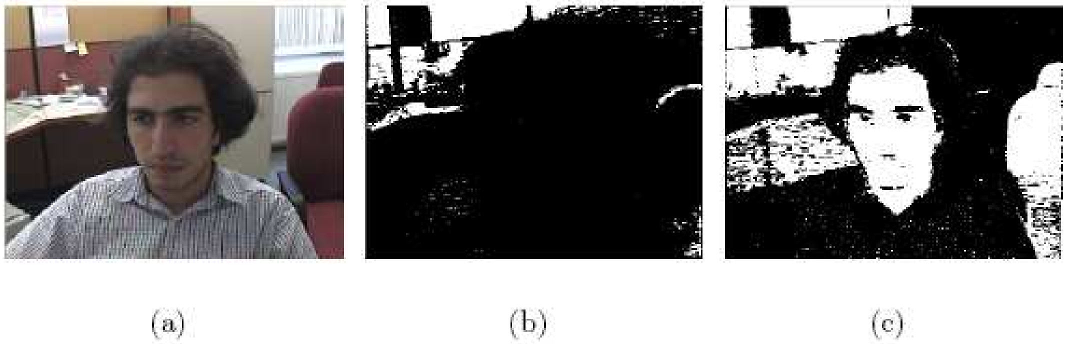


Figure 2.4: Responses of skin color filters for an image taken under fluorescent light. (a) Input color image, (b) HIS skin color filter response with fixed bounding thresholds, (c) NRGB skin color filter response with varying bounding thresholds.

Hence, the exhaustiveness in the variations for sample pixel set may bottleneck performance of the resulting skin color filter. Response of two skin color filters for same color image can be seen in Figure 2.4. Note that the HSI skin color filter with fixed thresholds is unsuccessful in determining skin color pixels. On the other side, NRGB skin color filter that is using adjustable thresholds is successful in determining skin color pixels by adding false alarms. Although, it may be more deeply experimented, we may state that a varying threshold skin color filter which includes self adaptation to image illumination properties (e.g. CCT) may result in more effective skin color filtering results.

3) Motion

Motion information is a convenient way of locating moving objects when a video sequence is provided. It is possible to narrow face searching area utilizing this information. The simplest way to achieve motion information is frame difference analysis. Accumulated frame difference is improved frame difference analysis which is used by many reported face detection research [5, 33]. Besides face region, Luthon et. al. [34], also employ frame difference to locate facial features, such as eyes. Another way of measuring visual motion is through the estimation of moving image contours. Compared to frame difference, results generated from moving contours are always more reliable, especially when the motion is insignificant [35].

2.2.1.2 Template Matching

Given an input image, the correlation values in predetermined standard regions, such as face contour, eyes, nose and mouth are calculated independently. Although, this approach has the simplicity, it has been insufficient for face detection since it cannot handle variations in scale, rotations pose and shape. Multiresolution, multiscale, subtemplates and deformable templates have been proposed to achieve scale and shape invariance template matching [36, 37].

In [36], Miao et al. proposed a hierarchical template matching method for face detection. Initially, the input image is rotated from -20^0 to 20^0 degrees to handle rotation. Then, each rotated image form a mosaic at different scales in which edges are extracted using Laplacian operator. The face template consists of six facial components of two eyebrows, two eyes, nose, and mouth. Face candidates are located by matching templates of face models represented in edges. In the final step, some heuristics are used to determine existence of a face. Experiments show better detection performance for images containing single face, rather than multiple.

Kwon et al. [37] proposed a detection method based on *snakes* and templates. In this approach, an image is first convolved with a blurring filter then with morphological operator to enhance edges. A modified *n-pixel* snake is used to find and eliminate small curve segments. Each candidate is approximated using an ellipse and for each of these candidates, a deformable template method is used to find detailed features. If a sufficient number of facial features are found, and their ratios satisfy the ratio tests based on the template, a face is considered to be detected.

Lanitis et al. [38] established a detection method utilizing both shape and intensity information. In this approach, training images are formed in which contours are manually labeled with sampled points, and vector sample points are used as shape feature vectors to be detected. They use a point distribution model (PDM) together with the principal components analysis (PCA) to characterize the shape vectors over an ensemble of individuals. A face shape PDM can be used to detect face in test images using active shape model search to estimate face location and shape parameters. The shape patch is then deformed to the average shape, and intensity parameters are extracted. Then the shape and intensity parameters are used together for measuring Euclidian distance from the faceness.

2.2.1.3 Generalized Knowledge Rules

In generalized knowledge-based approaches, the algorithms are developed based on heuristics about face appearance. Although, it is simple to create heuristics for describing the face, the major difficulty is in translating these heuristics into classification rules in an efficient way. If these rules are over detailed, they may come up with missed detections; on the other hand, if they are more general they may introduce many false detections. In spite of this, some heuristics can be used at an acceptable rate in frontal faces existed on uncluttered backgrounds.

Yang and Huang [4] used a hierarchical knowledge-based method to detect faces. Their system consists of three level rules going from general to detail. This method does not report a high detection rate, their ideas for mosaicing (multi-resolution), and multiple level rules have been used by more recent methods.

2.2.2 Image-Based Approaches

In contrast to feature-based approaches, image-based approaches utilize example image representations, instead of abstract representations consisting of features. In general, image-based approaches rely on machine learning and statistical analysis. Face detection is two class (face, non-face) classification problems which rely on learned characteristics generally in the form of distributions.

The specific need for a face knowledge is avoided by formulating the problem as a learning paradigm to discriminate a face pattern from a non-face pattern.

Image-based approaches can be better understood by considering statistical supervised pattern recognition. A raw image can be taken as random variable x, and this random variable is characterized by class-conditional density functions $p(x/face)$ and $p(x/non - face)$. If the dimensionality of x was not so high, a Bayesian or maximum likelihood classification would be possible.

Hence, image-based approaches utilize more complex techniques such as subspace representations and learning networks to overcome the high dimensionality of the problem space. Most of the image-based approaches apply a window scanning technique for detecting faces. The window scanning algorithm employs an exhaustive search of the input image for possible face locations at all scales, but there are variations in the implementation of this algorithm for almost all the imagebased systems. Typically, the size of the scanning window, the subsampling rate, the step size, and the number of iterations vary depending on the method proposed and the need for a computationally efficient system.

2.2.2.1 Linear Subspace Methods

In the late 1980s, Sirovich and Kirby [39] developed a technique using PCA to efficiently represent human faces. Given a set of face images, the proposed technique obtains the principal components of the distribution of faces, expressed in terms of eigenvectors (of the covariance matrix of the distribution). Then, each individual face in the set can be approximated by a linear combination of the largest eigenvectors (*eigenfaces*) corresponding to largest eigen values, using appropriate weights. Later, Turk and Pentland [33] improved this technique for face recognition.

Their method takes the advantage of the distinct nature of the weights of eigenfaces for individual face representation. Since, face reconstruction, by using its principal components, is an approximation, a residual error is defined in the algorithm as a preliminary measure of "faceness".This residual error which they termed "distance-from-face-space" (DFFS), gives an indication of face existence through the observation of global minima in the distance map.

Pentland et al. [40] later proposed a facial feature detector, using DFFS generated from eigenfeatures (*eigeneyes, eigennose, eigenmouth*), which are obtained from various facial feature templates in a training set. The feature detector is better while accounting for features under different viewing angles, since features of different discrete views were used during the training. The performance of the eye locations was reported to be 94% with 6% false positive rate in a database of 7562 frontal face images in front of a plain background.

More recently, Moghaddam and Pentland have further developed this technique within a probabilistic framework [41]. Unlike the usual PCA framework, they did not discard the orthogonal complement of face space. This leads to uniform density assumption of the face space. Hence, so they developed a maximum likelihood detector which takes into account both face space and its orthogonal complement to handle arbitrary densities. They report a detection rate of 95% on a set of 7000 face images for detecting the left eye. Compared to the DFFS detector, the results were significantly better. On a task of multiscale head detection of 2000 face images from the FERET [32] database which includes mug shot faces in front of a uniform background, the detection rate was 97%.

PCA is an intuitive and appropriate way of constructing a subspace for representing an object class in many cases. However, for modelling the variety in face images, PCA is not necessarily optimal. Face space might be better represented by dividing it into subclasses. Several methods have been proposed which are mostly based on some mixture of multidimensional Gaussians.

This approach was first applied to face detection by Sung and Poggio [9]. They modelled the empirical distribution of face and non-face patterns using six multi dimensional Gaussian clusters, as it is shown in Figure 2.5. Their system includes two main components, distribution models for face/non-face patterns and a multilayer perceptron (MLP) classifier. Training of MLP was established using 47316 examples. In order to collect non-face patterns, they have introduced a method called as *bootstrap*. By using bootstrap method, they train network using a small set of non-face examples. Then run the face detector and add false detected windows into set of non-face examples. They have reported 81.9% true detection rate in a database of 23 cluttered scene images including 149 faces with 13 false detections.

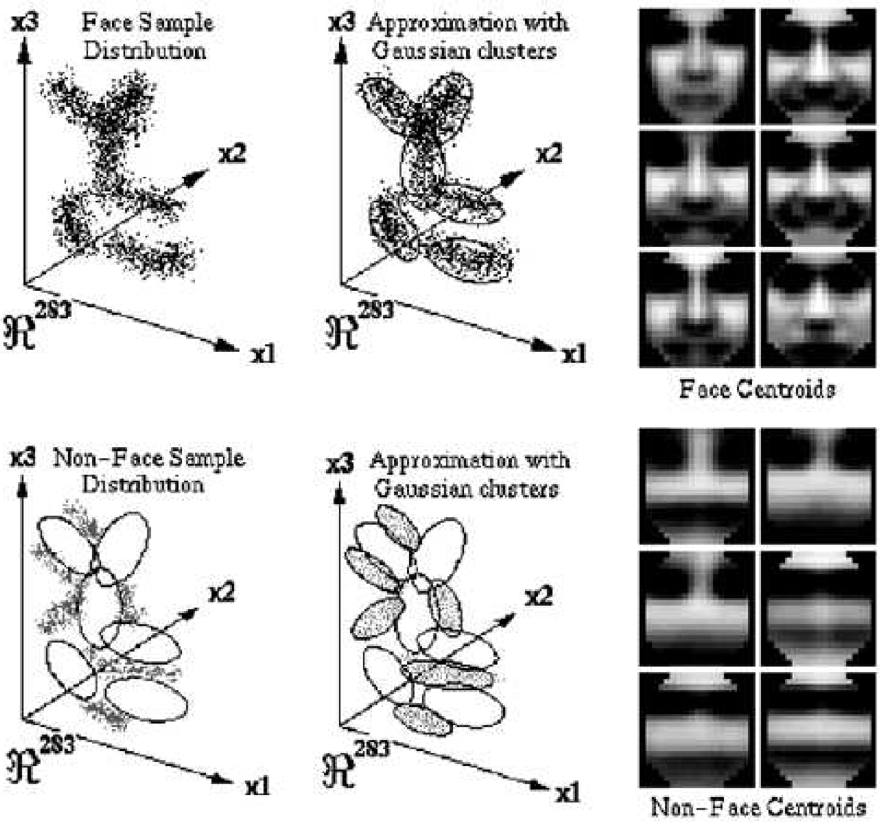


Figure 2.5: Distribution-based canonical face model. Top Row: Empirical distribution of face patterns using six multi-dimensional Gaussian clusters, whose centers are as shown on the right. Bottom Row: Sample of non-face patterns using six multidimensional Gaussian clusters to help localize the boundaries of the face distribution. The final model consists of six Gaussian face clusters and six non-face clusters. Sung and Poggio [9].

2.2.2.2 Learning Networks

Since face detection can be understood as a two class pattern recognition problem, several neural network-based approaches have been introduced for solution. A review of the neural network-based face detection methods can be found in Viennet et al. [42]. Other than basic multilayer perceptron approaches (MLP) [43, 44], the first advanced neural approach which reported significant results on a large, complex dataset was by Rowley et al. [7]. The system incorporates face knowledge in a retinally connected neural network as shown in Figure 2.6. The neural network is designed to look at windows of 20 x 20 pixels. There is one hidden layer with 26 units, where 4 hidden units connected to 10 x 10 pixel subregions, 16 units connected to 5 x 5 subregions, and 6 units connected to 20 x 5 pixels via input units. The input window is pre-processed through lighting correction (a best fit linear function is subtracted) and histogram equalization. This pre-processing method was adopted from Sung and Poggio's system mentioned earlier and it is illustrated in Figure 2.7. A major problem which arises with window scanning techniques, is overlapping detections. Rowley et al. [45] deals with this problem through two heuristics:

Thresholding: the number of detections in a small neighborhood surrounding the current location is counted, in output pyramid which is including both location and scale and if it is turn out to be above a certain threshold, a face is assumed to be present at this location.

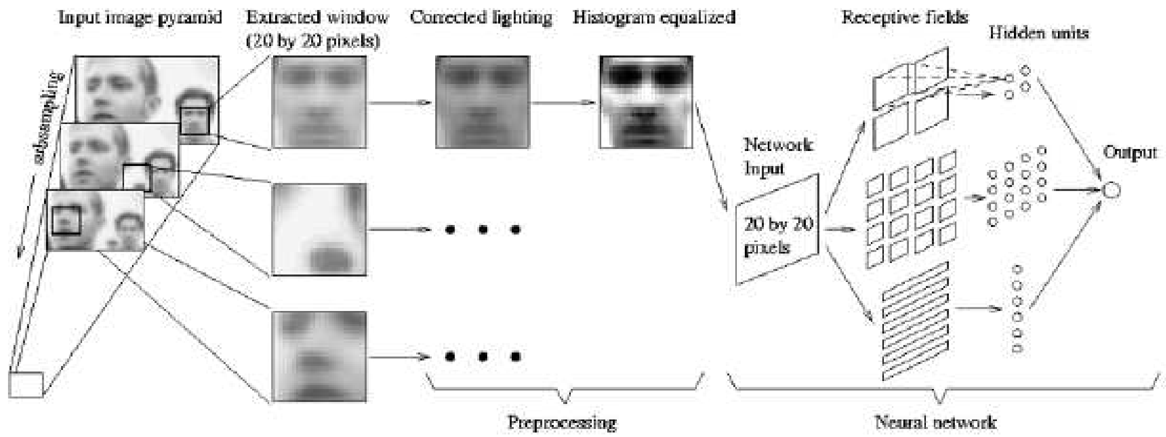


Figure 2.6: The system by Rowley et al [7]

Overlap elimination: when a region is classified as a face by using heuristic thresholding, then overlapping detections are likely to be false positives and removed.

Moreover, they also train multiple neural networks and combine the output with an arbitration strategy (ANDing, ORing, voting, or a separate arbitration neural network) [45].

In Lin et al. [22], a fully automatic face recognition system is proposed based on probabilistic decision-based neural networks (PDBNN).A PDBNN is a classification neural network with a hierarchical modular structure. Instead of converting input image to a raw vector, they preferred to use features based on intensity and edge information.

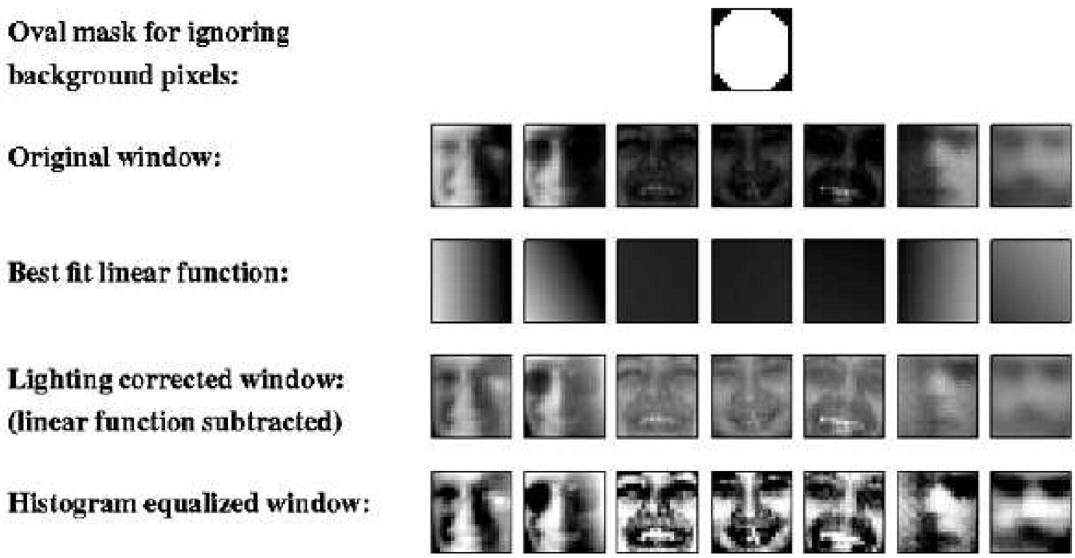


Figure 2.7: The preprocessing method applied by Rowley et al [7]. A linear function is fit to the intensity values in the window and then subtracted from the image. Finally, histogram equalization is applied to improve contrast.

Sparse Network of Winnows (SNoW), which is a new learning architecture in visual domain, is applied to face detection by Roth et al. [47]. The SNoW system, applied to face detection is a learning network consisting of two linear threshold units (LTU) (representing the subnetworks for face and non-face). The two target subnetworks operate on an input space of Boolean features. The best performing system derives features from 20 x 20 subwindows in the following way: for 1 x 1, 2 x 2, 4 x 4, and 10 x 10 subwindows, compute (position x intensity mean x intensity variance). This gives Boolean features in a 135424-dimensional feature space, since the mean and variance have been quantized into a predefined number of classes. The LTUs are separate from each other and are sparsely connected over the feature space. During training, weights linked to face and non-face subnetworks are promoted or demoted utilizing Winnow update algorithm [48] according to mistakes made in classification. Similar to the previously mentioned methods, Roth et al. [47] also use the bootstrap method of Sung and Poggio for generating training samples and preprocess all images with histogram equalization.

2.2.2.3 Statistical Approaches

There are several statistical approaches for face detection. Some of the proposed systems are based on information theory [49], a support vector machine [50] and Bayes [6] decision rule. Colmenarez and Huang [49] proposed a system based on Kullback relative information (Kullback divergence). This divergence is a nonnegative measure between two probability density functions for a random process *Xn*. During training, for each pair of pixels in the training set, a joint-histogram is used to create probability functions for the classes of faces and non-face. Since pixel values are highly dependent on neighboring pixel values, *Xn* is treated as a first order Markov process and the pixel values in the gray-level images are requantized to four levels. The authors use a large set of 11 x 11 images of faces and non-face for training, and the training procedure results in a set of look-up tables with likelihood ratios. In order to further improve performance, pairs of pixels which contribute poorly to the overall divergency are dropped from the look-up tables and not used in the face detection system. This technique is further improved by including error bootstrapping which is described earlier, and later the technique was also incorporated in a real-time face tracking system [49]. Another major approach is, Support Vector Machines (SVM) can be considered as a new paradigm to train polynomial function, or neural network classifiers. While most methods training classifier (e.g. Bayesian, neural networks) based on minimizing of training error *empirical risk*, SVMs exist on another principle called *structural risk minimization*, which aims to minimize an upper bound on the expected generalization error. A SVM classifier is a linear classifier. And, the optimal hyperplane is defined by a weighted combination of a set of training (support) vectors, and is chosen to minimize expected classification error of the previously unseen test patterns. Osuna et al. [51] develop an efficient method to train a SVM for large scale problems, and applied it to face detection. SVMs also applied to the problem in wavelet domain to detect pedestrians and faces [50]. Kumar and Poggio [52] recently incorporated Osuna et al.'s SVM algorithm in a system for real-time tracking and analysis of faces. They apply SVM algorithm on segmented skin regions of the input images to avoid exhaustive scanning.

As another approach, Schneiderman and Kanade [53, 6] describe two face detectors based on Bayes decision rule (presented as a likelihood ratio test as $P(image|object)/P(image|non - object) > P(non - object)/P(object)$). If the likelihood ratio (left side) of above equation is greater than the right side, then it is decided that an object (a face) is present at the current location.

The advantage of this approach is the optimality of the Bayes decision rule [32], if the representations are accurate. In the proposed face detection system [53], the posterior probability function is derived based on a set of modifications and simplifications. The resolution of a face image is first normalized to 64 x 64 pixels, and the face images are decomposed into 16 x 16 subregions while there is no modelling of statistical dependency among the subregions. Afterwards, the subregions are projected onto a 12-dimensional subspace by using local eigenvector coefficients constructed by PCA, and the entire face region is normalized to have zero mean and unit variance. In addition, the authors also used wavelets to obtain image visual attributes instead of eigenvectors in a consecutive work [6]. By the help of this approach, a view-based detector is developed with a frontal view detector and a right profile detector (to detect left profile images, the right profile detector are applied to mirror reversed images). Some examples of outputs which are processed using wavelets can be seen in Figure 2.8.

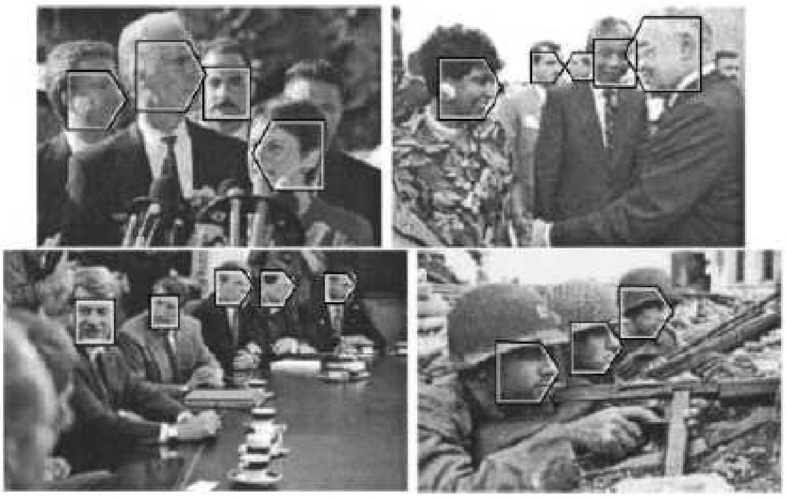


Figure 2.8: Face detection examples from Schneiderman and Kanade[6]

Chapter 3

Color Spaces Used for Skin Modeling

3.1 Color spaces used for skin modeling

Colorimetric, computer graphics and video signal transmission standards have given birth to many color spaces with different properties. A wide variety of them have been applied to the problem of skin color modeling. We will briefly review the most popular color spaces and their properties.

3.1.1 RGB

RGB is a color space originated from CRT (or similar) display applications, when it was convenient to describe color as a combination of three colored rays (red, green and blue). It is one of the most widely used color spaces for processing and storing of digital image data. However, high correlation between channels, significant perceptual non-uniformity [54, 55], mixing of chrominance and luminance data make RGB not a very favorable choice for color analysis and color based recognition algorithms. This color space was used in [56, 57].

3.1.2 Normalized RGB

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

$$\mathbf{r} = \frac{R}{R+G+B}, \mathbf{g} = \frac{G}{R+G+B}, \mathbf{b} = \frac{B}{R+G+B} \quad (6)$$

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 colors”, for the dependence of *r* and *g* on the brightness of the source RGB color is diminished by the normalization. A remarkable property of this representation is that for matte surfaces, while ignoring ambient light, normalized RGB is invariant (under certain assumptions) to changes of surface orientation relatively to the light source [58].

3.1.3 HSI, HSV, HSL - Hue Saturation Intensity (Value, Lightness)

Hue-saturation based color spaces were introduced when there was a need for the user to specify color properties numerically. They describe color with intuitive values, based on the artist's idea of tint, saturation and tone. *Hue* defines the dominant color (such as red, green, purple and yellow) of an area, saturation measures the colorfulness of an area in proportion to its brightness [59].

The "intensity", "lightness" or "value" is related to the color luminance. Several interesting properties of Hue were noted in [58]: it is invariant to highlights at white light sources, and also, for matte surfaces, to ambient light and surface orientation relative to the light source. However, [59], points out several undesirable features of these color spaces, including hue discontinuities and the computation of "brightness" (lightness, value), which conflicts badly with the properties of color vision.

$$\mathbf{H} = \arccos\frac{\frac{1}{2}((\mathbf{R}-\mathbf{G})+(\mathbf{R}-\mathbf{B}))}{\sqrt{((\mathbf{R}-\mathbf{G})^2+(\mathbf{R}-\mathbf{B})(\mathbf{G}-\mathbf{B}))}} \qquad \textbf{(7)}$$

$$\mathbf{S} = 1 - 3\,\frac{\min\{\mathbf{R},\mathbf{G},\mathbf{B}\}}{(\mathbf{R}+\mathbf{G}+\mathbf{B})} \qquad \textbf{(8)}$$

$$\boldsymbol{V} = \frac{1}{3}(\boldsymbol{R}+\boldsymbol{G}+\boldsymbol{B}) \qquad \textbf{(9)}$$

The polar coordinate system of Hue-Saturation spaces, resulting in cyclic nature of the colorspace makes it inconvenient for parametric skin color models that need tight cluster of skin colors for best performance. A different representation of Hue-Saturation using Cartesian coordinates can be used [60]:

$$\mathbf{X} = \mathbf{S}\cos\mathbf{H} \qquad \mathbf{Y} = \mathbf{S}\sin\mathbf{H} \qquad \textbf{(10)}$$

3.1.4 TSL - Tint, Saturation, Lightness

A normalized chrominance-luminance TSL space is a transformation of the normalized RGB into more intuitive values, close to hue and saturation in their meaning.

$$\boldsymbol{s} = \sqrt{\frac{9}{5}(\boldsymbol{r}'^2 + \boldsymbol{g}'^2)} \qquad \textbf{(11)}$$

$$\boldsymbol{T} = \begin{cases} \frac{\arctan\left(\frac{r'}{g'}\right)}{2\pi} + \frac{1}{4} & , g' > 0 \\ \frac{\arctan\left(\frac{r'}{g'}\right)}{2\pi} + \frac{3}{4} & , g' > 0 \\ 0 \ , & g' = 0 \end{cases} \qquad \textbf{(12)}$$

$$\mathbf{L = 0.299R + 0.587G + 0.114B} \qquad \textbf{(13)}$$

where r’=(r-1/3), g’=(g-1/3) and *r*, *g* come from equation (6). [28] have compared nine different colorspaces for skin modelling with a unimodal Gaussian joint pdf (only chrominance components of the colorspaces were used). They argue that normalized TSL space is superior to other colorspaces for this task. [60] has also employed this representation for their approach.

3.1.5 YCbCr

YCbCr is an encoded nonlinear RGB signal, commonly used by European television studios and for image compression work. Color is represented by *luma* (which is luminance, computed from nonlinear RGB [59]), constructed as a weighted sum of the RGB values, and two color difference values *Cr* and *Cb* that are formed by subtracting luma from RGB red and blue components.

$$\begin{aligned} \mathbf{Y} &= \mathbf{0.299R+0.587G+0.114B} \\ \mathbf{Cb} &= \mathbf{B-Y} \\ \mathbf{Cr} &= \mathbf{R-Y} \end{aligned} \qquad \textbf{(14)}$$

The transformation simplicity and explicit separation of luminance and chrominance components makes this colorspace attractive for skin color modeling [62, 28, 63].

Chapter 4
Proposed Approach .

The Face Detection in proposed approach is developed by detecting skin pixels first in a given image. The skin pixels are detected by applying FCM on modified color space channel Cr’. The FCM is used to segment skin like region from an input image. These skin regions are filtered according to rules developed in proposed approach. After applying rules we get the final output as face and non-face and we decide face or non-face on the basis of Decision Rule given in this section.

4.1 MODIFIED Y’Cb’Cr’

In this we modify the YCbCr color space to Y’Cb’Cr’ to detect skin region from an input image. To transform the original color space into modified color space the equation (10) below is used.

Y’=Y;
Cb’=Cb/beta; **(15)**
Cr’=(Cr-alpha)/beta; ***Where alpha & beta are transformation parameters***

The color space is modified on these two parameters alpha and beta. These parameters modify the color space which is used to segment skin color using FCM only on one channel to make it fast. These two parameters reduces the range of searching for segmenting skin pixels, the parameter alpha reduces the range of overall Cr and beta normalize the values in the range [0 1] of both Cr and Cb. Thus gives Cr’ and Cb’ whereas luma component(Y’=Y) kept unchanged to filter the clusters values. After transforming the color space the clustering of data is done on Cr’ to detect skin color.

4.2 Clustering Technique

Clustering is the process of grouping a set of unlabelled multidimensional patterns (objects or data points), such that patterns in the same cluster have the most similar characteristics, and patterns within different clusters have the most dissimilar characteristics. In most cases a cluster is represented by a cluster centre or a ‘centroid’ [64]. Clustering has been applied to a wide range of applications, such as pattern recognition, image segmentation, spatial data analysis, machine learning, data mining, economic science, and internet portals. Classification, another data analysis method, is often confused with clustering. The distinction between the two approaches is that classification is a supervised learning process which is trained on a set of pre-labeled patterns in order to predict into which class new patterns should be placed. In contrast, clustering is unsupervised, has no predefined classes and does not involve training examples [65, 66]. As mentioned above, the aim of clustering is to group the patterns into clusters based on their similarity.

A basic outline of a general clustering process can be described as follows [64, 66]:

1) ***Perform feature selection and/or feature extraction*** from the original dataset. Feature selection is the process used to find the most representative subset of the original features to be used within clustering. Feature extraction uses one or more transformations of the original features to produce new salient features [64]. The purpose of this step is to make the clustering process work more ‘efficiently’ (in some way) as only the most important characteristics need to be considered. The objective is usually to reduce the time required for the clustering process without adversely affecting the quality of the clusters obtained.

2) ***Select a proximity measure to be used***. This is used to evaluate the similarity (or dissimilarity) of two data points. This could be the Euclidean distance, correlation coefficient or other measures.

3) ***Apply the clustering technique to classify the dataset.*** Many different clustering techniques have been developed within the literature. This literature review identifies and describes these approaches.

4) ***Validate the clusters.*** Cluster validation evaluates the clustering scheme obtained from step 3. Cluster validity indices are often used to assess the quality of the clusters.

In general, the different clustering techniques can be divided into two main categories. They are hierarchical and partitional clustering [64]. In each category, many subtypes and variants have been applied to diverse types of clustering problems. In conventional clustering algorithms, each pattern has to be assigned exclusively to one cluster. Where the physical boundaries of clusters are well defined, this approach can work well. However, when using data from real world applications, the boundaries between clusters might be vague. For this reason, fuzzy clustering extends the traditional clustering concept by allowing each pattern to be assigned to every cluster with an associated membership value. Therefore, for unclear cluster boundaries, fuzzy clustering may obtain more reasonable results. In partitional clustering, normal process is to optimize an objective function which somehow reflects the quality of the clusters. In order to find better solutions, some search based approaches have also been combined with these clustering algorithms in order to maximize or minimize the objective function [67]. In the rest of this section, the key literature is identified and described. This includes hierarchical clustering, partitional clustering, fuzzy clustering and hybridisations of the above clustering approaches with search based algorithms.

4.2.1 Fuzzy C-Means (FCM)

Fuzzy clustering has become an interesting and important branch of partitional clustering. It was originally developed in 1969 when Ruspini applied fuzzy set theory to clustering [68]. One of the major differences between fuzzy clustering and hard clustering is that fuzzy clustering allows each pattern to belong to more than one cluster with varying degrees of certainty, based on their distance to the cluster centres. This is called the ‘membership’ or ‘soft membership’ function.

The fuzzy c-means algorithm is one of the most popular fuzzy clustering algorithms. It was first developed by Dunn in 1973 [69] and was subsequently improved by Bezdek in 1981 [70]. In comparison with Dunn’s algorithm, Bezdek’s fuzzy c-means algorithm introduces a *fuzzifier* parameter, $1 \leq m < \infty$. The purpose of the fuzzy c-means algorithm is to minimise the fuzzy objective function as shown in Equation **(16)**.

$$J(U,V) = \sum_{i=1}^{n} \sum_{j=1}^{c} (\mu_{ij})^m ||x_i - v_j||^2 \tag{16}$$

The resulting algorithm can recognise spherical patterns in *multi-dimensional* space . Once again, this can be formulated as followings: $X=\{x_1, x_2, \ldots\ldots x_n\}$ represents a collection of data and $V=\{v_1, v_2, \ldots\ldots v_c\}$ is set of corresponding cluster centres. In addition, μ_{ij} is the membership degree of pattern x_i to the cluster centre v_j and μ_{ij} must satisfy the following conditions:

$$\mu_{ij} - [0,1], \qquad i - 1, \ldots\ldots n, \quad j - 1, \ldots c, \tag{17}$$

$$\sum_{j=1}^{c} \mu_{ij} = 1 \tag{18}$$

Parameter *m* is called the ‘fuzziness index’ (or fuzzifier) and is used to control the fuzziness of the membership of each data point. A larger value of *m* makes the method ‘more fuzzy’ whilst a smaller value makes the method ‘less fuzzy’. There is no theoretical basis for the optimal selection of *m*, but a value of *m = 2.0 is* most commonly used [70]. The Euclidean distance between x_i and v_j is represented by $||x_i - v_j||$. $U = (\mu_{ij})$ is a fuzzy partition matrix, which contains all of the membership degree values from each data to all cluster centres.

The fuzzy c-means clustering algorithm is shown in the following steps:

1. Fix the number of clusters, c , where $2 \leq c < n$, and initialise the fuzzy partition matrix U with a random value such that it satisfies conditions **(17)** and **(18)**.
2. Calculate the fuzzy centres v_j using

$$v_j = \frac{\sum_{i=1}^{n}(\mu_{ij})^m x_i}{\sum_{i=1}^{n}(\mu_{ij})^m}, \forall j = 1, \ldots, c \qquad \textbf{(19)}$$

3. Update the fuzzy partition matrix U with

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}} \qquad \textbf{(20)}$$

Where $d_{ij} = \|x_i - v_j\|$, $i=1\ldots n$ and $j=1\ldots c$

4. Repeat step (2) to (3) until one of the termination criterion is satisfied.

In fuzzy c-means clustering algorithm [71], the fuzzy c-means procedure continues until one of the termination criterion is satisfied. Termination criteria can be that the difference between updated and previous objective function J is less than a predefined minimum threshold. Additionally, the maximum number of iteration cycles can also be a termination criterion.

In this thesis, after the fuzzy c-means clustering process a pattern is set to a specific cluster for which the degree of membership is maximal. This process is known as 'hardening' the results. Studies have shown that hardening the results obtained from fuzzy c-means produces different solutions from the hard clustering results obtained directly from k-means, and that the fuzzy c-means solutions can be better [72,73]. However, as with the k-means algorithm, fuzzy c-means needs the number of clusters to be pre-specified in advance as an input parameter to the algorithm. However, both approaches can still suffer premature convergence to local optima. This is due to the fact that both these algorithms begin with random initialisation of the cluster centres. If the initial cluster centres are not appropriate, the iterative improvement of the centre positions can result in locally optimal solutions being obtained.

4.2.2 Xie-Beni Validity Index

Xie-Beni (XB) defined a new validity index which not only involved the membership values, but also included properties taken from the data itself [74]. The XB index (also named the compactness and separation validity function) is a representative index of relative validity indices [75]. In the following, let us assume that V_{XB} represents the overall XB index value, π is the compactness of data in the same cluster and s is the separation of the clusters. The XB validity index can now be expressed as:

$$V_{XB}=\pi/s \qquad \textbf{(21)}$$

Where $$\pi = \frac{\sum_{i=1}^{n}\sum_{j=1}^{c}(\mu_{ij})^2||x_i - v_j||^2}{n} \qquad \textbf{(22)}$$

and $s = (d_{min})^2$, here d_{min} is the minimum distance between cluster centres, given by $d_{min}=\min_{ij}|| v_i - v_j ||$. From the expressions **(21)** and **(22),** it can be seen that a smaller value of π indicates that the clusters are more compact whilst a larger value of s indicates the clusters are well separated. As a result, a smaller value of V_{XB} means that the clusters have a greater separation from each other and are more compact within each cluster. It should be noted that the XB validity index also has a disadvantage in that, as the number of clusters c gets very large or close to the number of data n, the index value monotonously decreases [75].

4.3 Skin Color Segmentation using FCM

After preprocessing of the image to enhance the contrast, it is subjected to a skin color based segmentation process to separate the faces from the background regions.

Skin is one of the most distinguishing properties of the human face surface. Skin color is quite a concentrated and stable region in images. Using this useful clustering data provided by the chrominance (color) information of the skin, it can be distinguished from the background in order to locate the possible areas which might contain the face.

Skin color is segmented from the input image for the processing of face detection. The skin color segmentation is done applying FCM on the modified color space Y'Cb'Cr'. The values of third component i.e. Cr' of input image is clustered using FCM. The FCM clustered the data in different clusters defined in FCM for example it clustered data in 12 clusters which is validated by cluster validity index for a given input image. Each cluster contains some values of data which corresponds to input data i.e. Cr' which means that in some clusters will have the locations to which skin pixels lies in that range. We plot all the clusters to see in which cluster skin pixels lies, below are the images of each clustered data by FCM.

4.3.1 Filtering of clusters data

In this stage the clusters are filtered again on basis of Y' component of modified Y'Cb'Cr' color space. The pixels in the each cluster which have value greater than THR_Y are remain unchanged and rest are marked white in each cluster so that the noise in segmentation skin pixels can be removed.

Tmin & Tmax is decided for selecting the clusters that belongs to skin region. Tmin is the minimum value of each cluster and Tmax is the maximum value of each cluster. After analyzing the clusters that have maximum skin pixels on any input image we can set that Tmin and Tmax for rest images so that the segmentation takes place automatically for same database. For a different database the values might differ and we should check on a sample image of database to get Tmin & Tmax values for different database. Tmin & Tmax may depend on different databases because acquiring of images takes place from sources like cameras of different resolutions and quality or web.

If the chrominance properties are found to be unsatisfactory, the pixel at that particular spatial position is changed to black, else kept same as before. Thus, the output image contains only the faces and the face-like regions. This image further tested and false regions rejected to finally give the face regions.

4.4 Removal of unwanted regions and noise

4.4.1 Morphological Cleaning

After the FCM based segmentation, it is observed that all the non-skin regions have been rejected from the image. However, the image is still noisy and cluttered. The image is then subjected to a series of morphological operations, which are performed sequentially to ‘clean up’ the image. The end objective is to get a ‘mask image’ that can be put on the input image to yield face regions without any noise and clutter.

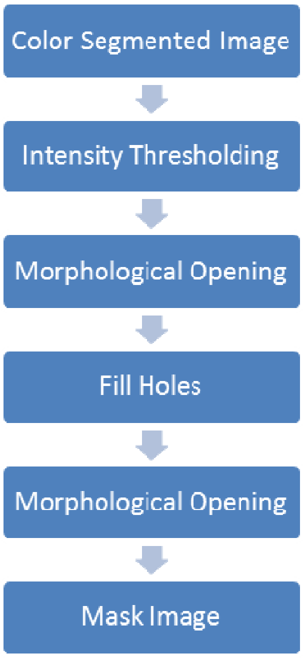


Figure 4.1 Sequence of steps to 'clean' the image

The sequence of steps is further described as follows:

1. The color image is first converted into a gray scale image since morphological operations are known to easily work on intensity and binary images.

2. ‘Intensity’ thresholding so that they can be effectively cleaned up by morphological opening. The threshold is set low enough so that it does damage parts of face but only create holes in it.

3. Morphological opening is then performed to eliminate very small objects from the image while keeping the shape and size of larger objects in the image unbroken. A disk shaped structuring element of radius 1 is used.

0	1	0
1	1	1
0	1	0

Figure 4.2 Disk shaped structuring element of unity radius

4. ‘Hole-filling’ is performed to maintain the face regions as single connected regions to account for a second morphological opening with a much larger structuring element. Or else, the mask image will contain many cavities and holes in the faces.

5. Morphological opening is again performed to remove small to medium sized objects which can be safely neglected as non-face regions. This time a bigger structuring element, disk shaped with radius 4, is used.

4.4.2 Labeling of Connected Regions

Label the each region in a binary segmented image as labeling is done in binary form of an image to mark each region with unique index. Marking of each region with unique index is done to easily traverse each region. Labelling of each region can be done in rgb form also so that we can give a different color to each region which can be seen visually.

4.4.3 Bounding Box Formation

After labeling of each region we form a bounding box around each region in segmented image which is masked by mask image. In this bounding box is formed around each region. Figure shows below the bounding box around each image.

4.5 Rules for first step elimination

The faces are generally contained in rectangular moderate sized bounding boxes. Similarly some non-faces are in bounding box. To eliminate the non-faces the rules defined are used to eliminate non-faces in first step of elimination. These rules neglect all the non-faces which are very dissimilar to faces objects in the image.

Rule 1: If Width is SMALL and Height is SMALL then Non-Face

Rule 2: If Width is SMALL and Height is LARGE then Non-Face

Rule 3: If Width is LARGE and Height is SMALL then Non-Face

Rule 4: If Width is MEDIUM and Height is SMALL then Non-Face

Rule 5: If Width is LARGE and Height is LARGE then multiple Face.

Rule 6: If Width is SMALL and Height is MEDIUM then Face

Rule 7: If Width is MEDIUM and Height is LARGE then Face.

4.6 Eye Map

After first step of elimination of non-faces by fuzzy rules defined above the segmented image still contain some noisy or face like regions which is similar property to faces. To eliminate further we localized eyes on each left over region by using eye map equation (23) et.al [63].
Hsu et al [63] used Eye Map to determine eye region in different light condition. This Eye Map is obtained by combining the two separate Eye Maps (EyeMapL and EyeMapC). These two Eye Maps are built from a segmented facial color image, one from the luminance component (EyeMapL) and the other from the chrominance component (EyeMapC). EyeMapC is constructed using[63]:

$$EyeMapC = \frac{1}{3}\{C_b^2 + (\tilde{C_r})^2 + (\frac{C_b}{C_r})\} \quad \textbf{(23)}$$

Where C_b and C_r are chrominance components of YCbCr color space. $\tilde{C_r}$ is $255 - C_r$. The values of Cb^2, $\tilde{C_r}^2$ and Cb/Cr are normalized to the range [0,1]. The 1/3 scaling factor is applied to ensure that the resultant EyeMapC stays within range of [0,1].The Eye Map based on luminance component (EyeMapL) is obtained by combining the output from dilation and erosion operations. EyeMapL is constructed using :

$$EyeMapL = \frac{Y(x,y) \oplus se(x,y)}{Y(x,y) \ominus se(x,y)+1}, \quad \textbf{(24)}$$

where se represents the structuring element (taken to be a ball structuring element) and $\oplus$ *and* $\ominus$ denote gray-scale dilation and erosion, respectively. The “1” is added to the denominator in the above formula to avoid division by zero. Finally, the EyeMapL multiplies with the EyeMapC to produce the desired Eye Map. This Eye Map determines eye region very well.. Then we apply Otsu’s method [77] to convert intensity image to binary image.

4.7 Eye Region Localization

Eye maps are created on each bounding box left over by first elimination step but in these bounding box may contain some noisy part also isolated in the box. The EyeMap brighten those areas also so detect only eye position is difficult. To mark only eye positions we select the area which has highest number of skin pixels in the current bounding box. To select region highest skin pixel region in current bounding box we calculate the area of each region. After selecting region with highest skin pixel we detect the boundary of that region. The boundary of this region is detected because we have to traverse in this region boundary only. Now we apply a 5x5 window within this boundary but we reduce this boundary size by 5 pixels as we applying 5x5 window which lies outside the region also as it is a sliding window based method. The window is moved on each pixel within new boundary defined and on each pixel the total sum of window is calculated. If the total sum of window is greater than THR_SUM then on the pixel is marked as white pixel otherwise black. By performing this operation we get white pixels on eyes.

4.8 Decision rule

Now all the regions are gone through eye map detection and it marks white pixels which have eyes and presence of eyes say that region which have eye pixels are faces. So final elimination of unwanted region are done with this decision that region which have eye pixels are faces and rest other are non-faces so eliminate all the non-faces and thus we get final faces in the image. It greatly reduces false positives which are coming after applying fuzzy rules of height width.

Decision Rule: If eye is present than face otherwise non-face

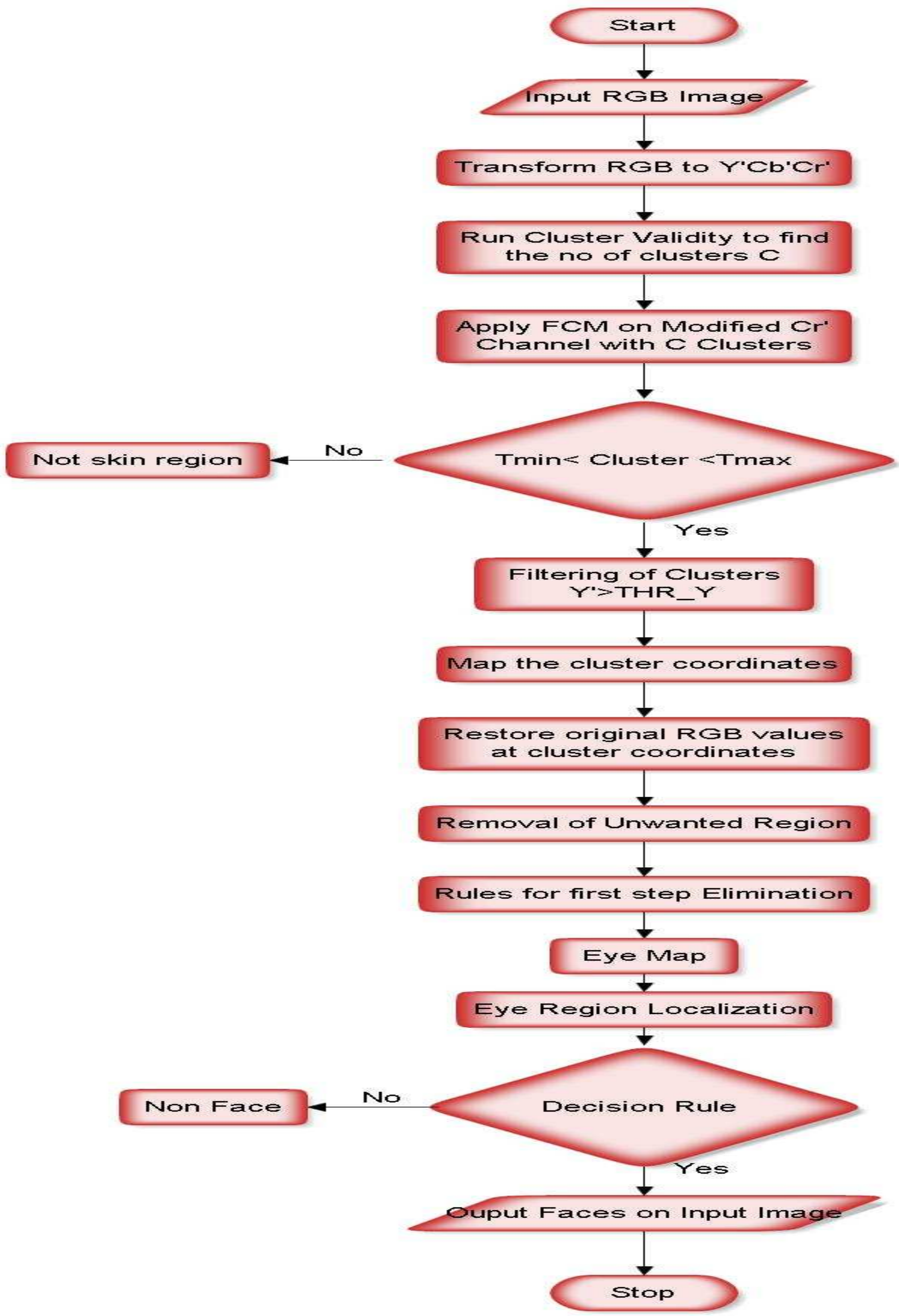


Figure 4.3: FLOWCHART OF PROPOSED ALGORITHM

Chapter 5

Experimental Results

5.1 Results using Proposed approach

To evaluate the performance of our approach numerically, we randomly select 100 images from the LWF face skin database as the test data. **LFW** data contain 13233 Images of 5749 people with 1680 people with two or more images. The result of skin-like region detection using the modified Y'Cb'Cr' color space and FCM works well over those regions under strong lighting but also work well to detect those skin regions under shadow. The values of alpha =128 and beta=256 is taken for modifying color space and then rounding is done on those obtained values. The no of clusters are decided by cluster validity Xie Beni index rule. For same resolution image the no of clusters are kept still once validate by Xei Beni index so that the process of segmenatation should become fast. For our sample input RGB image Figure 5.1 shown below the no of clusters is taken 12 given by validity index rule. Tmin=0.04 and Tmax=0.17 is taken for filtering the clusters qualifying for the skin pixels. These Tmin and Tmax is taken experimentally on a sample image of same resolution. The clusters whose values are falling in the range of Tmin and Tmax are selected for detection of skin like regions otherwise the whole cluster is ruled out.

The input RGB image shown in Figure 5.1 is then modified the color space Y'Cb'Cr' which is shown in Figure 5.2. After modifying color space FCM is applied and Figure 5.3 shows the result of segmented image after FCM. Then again filtering of clusters is done on the basis of Y' component whose values are greater than THR_Y are kept in segmented image are otherwise other pixels set to 255 shown in Figure 5.4. The value of THR_Y =75 is taken for filtering of clusters data. The binarisation is done on finally segmented image shown in Figure 5.5. After segmentation skin color from the input image, the removal of unwanted region is done applying

morphological operations on it. The first step is applying morphological opening on binarised image with disk radius of 1, the output of this is shown in Figure 5.6. Then the morphological filling takes place with morphological opening operation of disk radius 4 shown in Figure 5.7.
Each region is now labeled to RGB form for visual perception of different regions shown by Figure 5.8 with a bounding box on each labeled region. Finally Bounding Box are plotted on input RGB image to visualize our performance on detection shown by Figure 5.9. These bounding box are further eliminated by rules of first step elimination shown in Figure 5.10. Now each bounding box taken from an input image to detect eye map, a sample of this is shown by Figure 5.11 which is taken as input image to detect Eye Map and Figure 5.12 shows the ouput of detected Eye Map. Finally decision rule gives answer that if eye region is detected in the box then it belongs to face otherwise not which is shown in Figure 5.13 the final output of whole process. Below the images are shown for each step used in the process of face detection.

Figure 5.1: Input RGB IMAGE

Figure 5.2: MODIFIED COLOR SPACE

Figure 5.3: Segmented image after FCM

Figure 5.4: Filtered image after FCM

Figure 5.5: Binary Image of Skin Regions

Figure 5.6: Morphological Opening Image of Disk radius 1

Figure 5.7: Filled Image with morphological Opening of Disk Radius 4

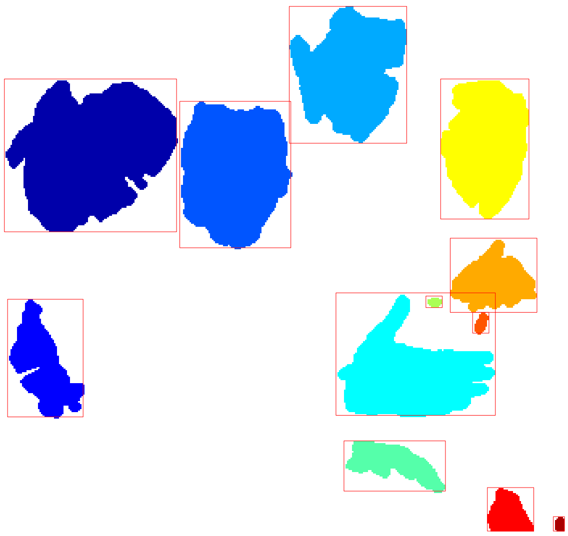


Figure 5.8: Labeled Region in RGB FORM

Figure 5.9 Bounding Box Formation on Input image by using Labeled region

Figure 5.10: Rules of Elimination In first Step is used to eliminate noises

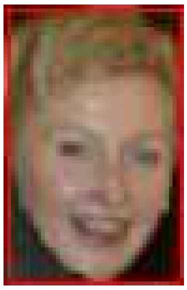


Figure 5.11: Bounding Box Image taken for Eye Map Localization

Figure 5.12: Eye Map Image using Hsu et al method[12]

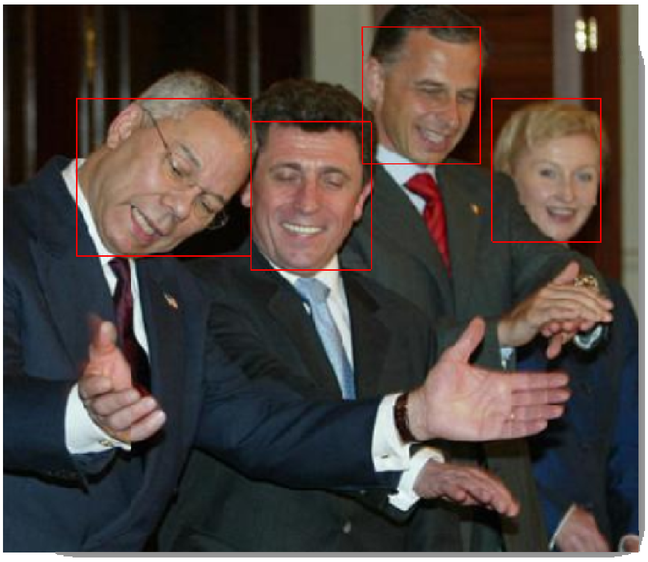
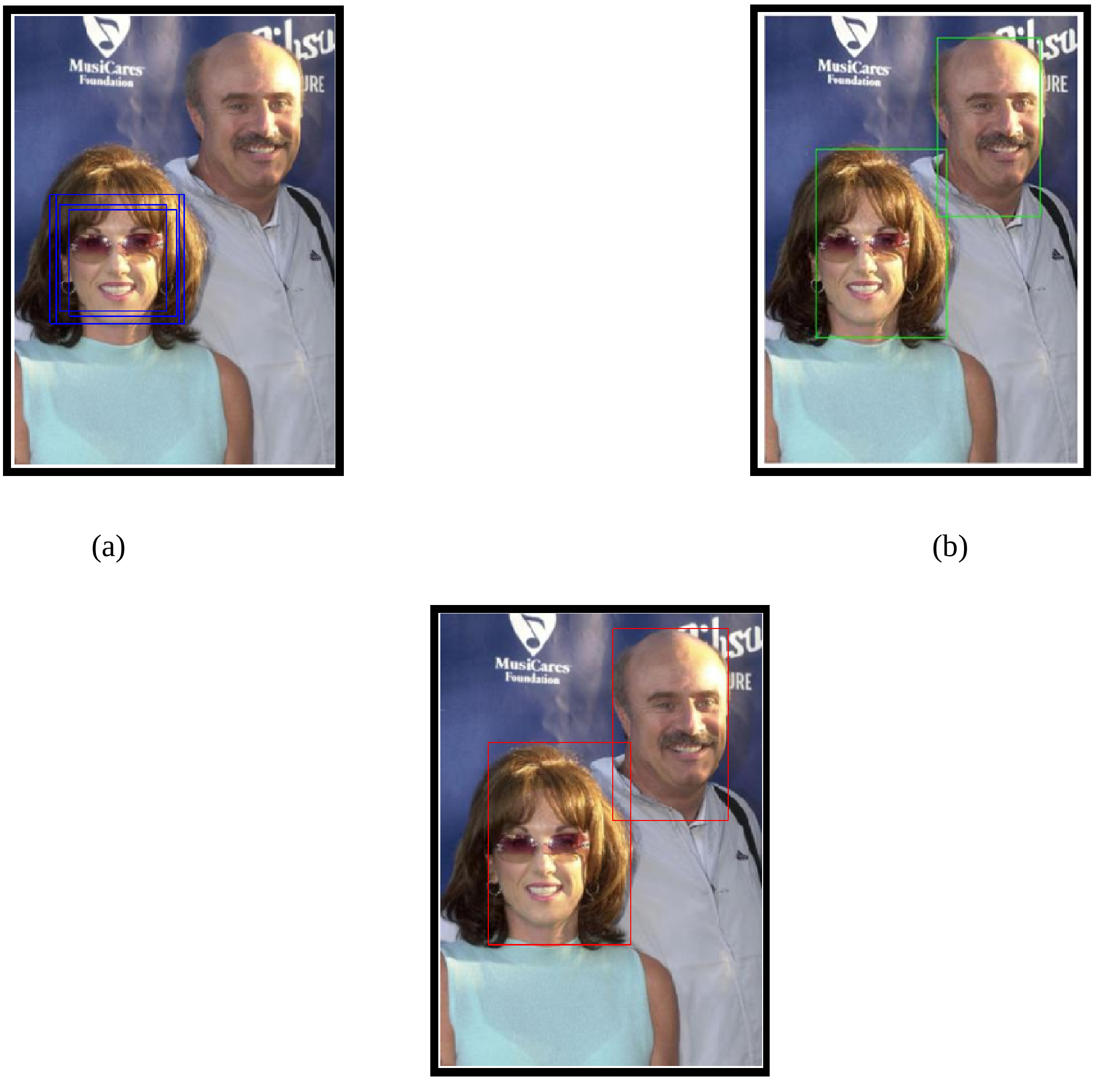


Figure 5.13: Final Output Image After DECISION RULE

5.2 Comparison with Other Standard techniques

The face detection has several methods to detect faces in still images and videos. For our comparison with other techniques we have taken two very well known methods which detect faces very efficiently in still images in videos the first method is Voila Jones[76] and second method is Hsu et al [63] . Our approach is developed to detect faces in still images so we take same images to differentiate with our method and takes output from these methods to compare with our proposed approach. The figures with showing blue boxes around faces are detected by Voila Jones method [76], green boxes on faces by Hsu method [63] and final ly our proposed approach showing result with red boxes.

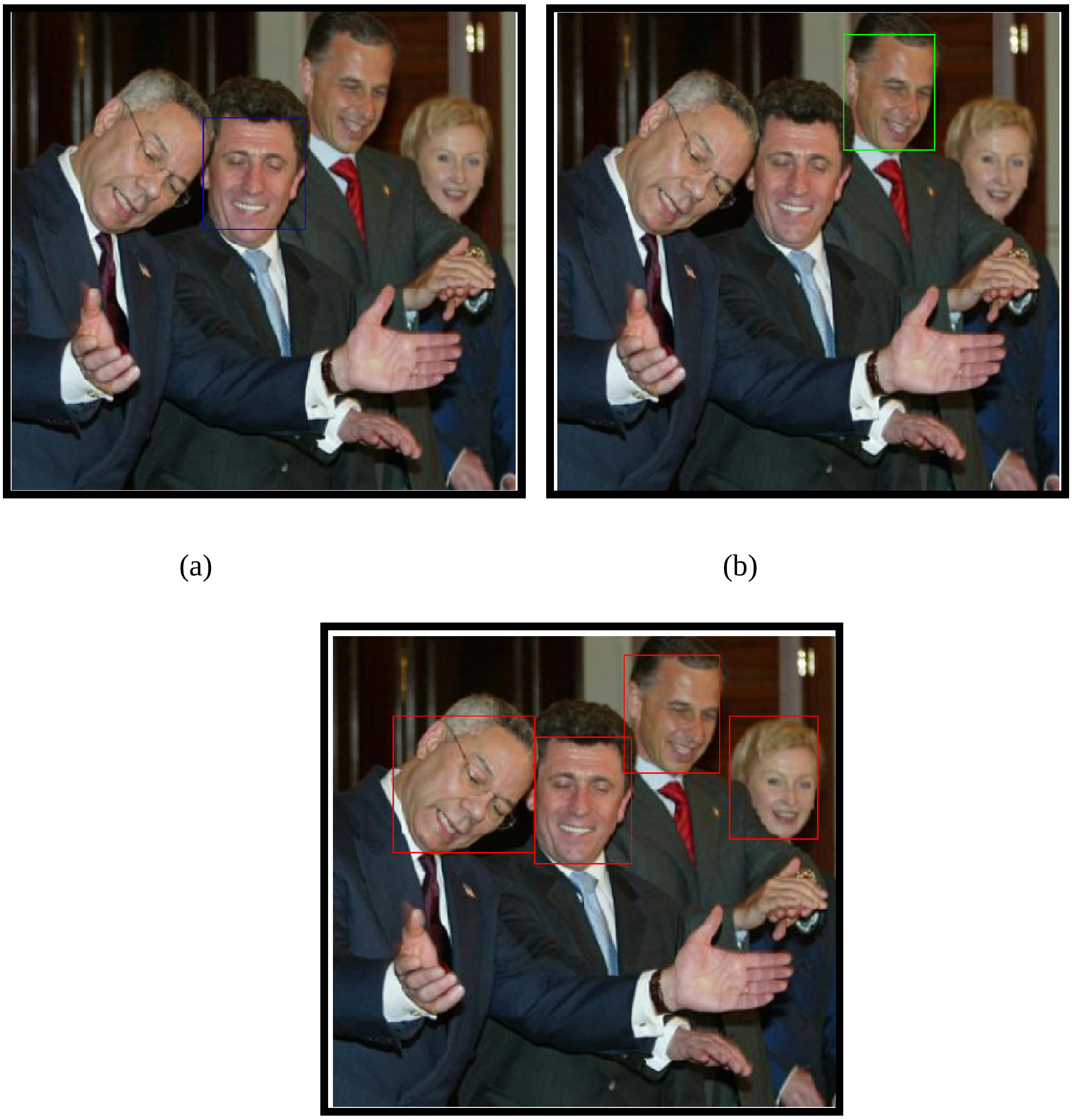
In Figure 5.14 and Figure 5.16 the no of faces detected by Voila Jones method is 1 where as Hsu method and proposed approach detects 2 faces in an image. The Figure 5.15 consists of 4 different persons and both methods detect only 1 face whereas our method detects all the 4 faces in this image. Similarly in Figure 5.17 the faces detected by other methods are 1 which contain no of persons 3 and our method detects all the three faces in that image. The Figure 5.18 consist of 2 persons the Hsu method and our proposed approach detects 2 faces but voila jones method not able to detect single face.



(a) (b)

(c)

Figure 5.14: Face detected by methods (a) Viola Jones[76] (b) Hsu [63] (c) Proposed Approach



(a) (b)

(c)

Figure 5.15: Face detected by methods (a) Viola Jones[76] (b) Hsu [63] (c) Proposed Approach

(a) (b)

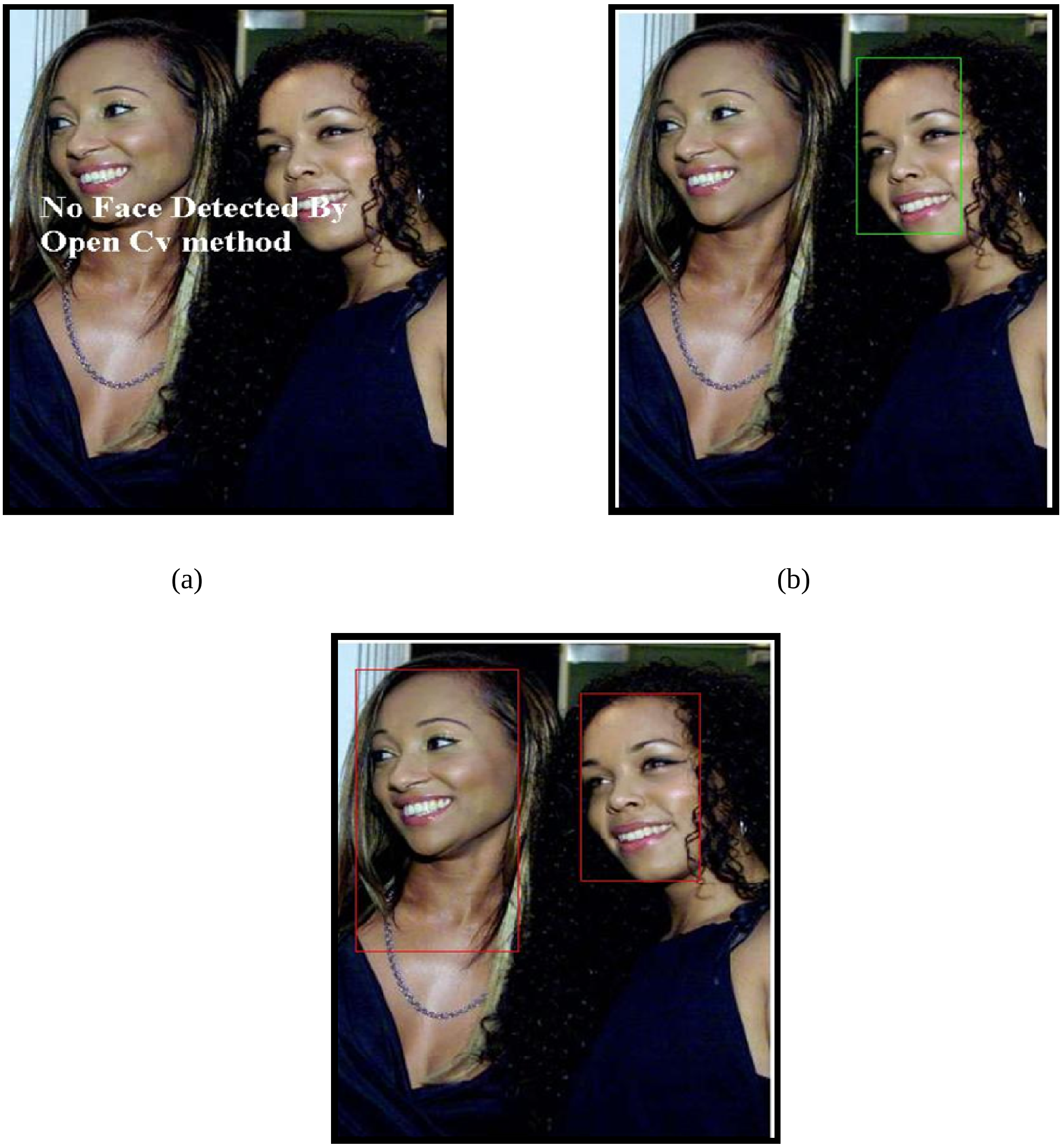
Figure 5.16: Face detected by methods (a) Viola Jones[76] (b) Hsu [63] (c) Proposed Approach

(a)

(b)

(c)

Figure 5.17: Face detected by methods (a) Viola Jones[76] (b) Hsu [63] (c) Proposed Approach



(a) (b)

(c)

Figure 5.18: Face detected by methods (a) Viola Jones[76] (b) Hsu [63] (c) Proposed Approach

Chapter 6

Conclusions

Face Detection is a very important step in recognition of faces. Face recognition is done mainly in two step face detection and then recognition. The recognition can only be possible if face is detected appropriately by face detection method used. The different method exist for face detection but we have evaluated our results with two most commonly used method ie. Voila Jones method and Hsu method for comparing our work to these methods.

Our proposed approach shows that our method works more fine with these existing methods as it is able to detect those faces which are not detected by these two methods. Our algorithm detects well because it is based on skin color model to detect faces which are invariant technique to poses variations in the faces.

Although it detect more skin pixels compared to other methods which makes easy to detect face in input image because the skin pixels are detected using FCM which an optimization technique and it optimizes detection of skin pixels in an input image. The fuzzy rules used efficiently remove noises from the segmented image and left over only face region or face like region. These regions are finally evaluated by decision rule and give more appropriate faces than other methods.

The future scope of the work is to optimize the outputs of segmentation image so that background noises removed without implementing any specific procedure to decrease the computation time and more criteria should be decided to give result as face or non face in conjunction with decision rule.

References

[1] Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S., "Handbook of Fingerprint Recognition", Springer –Verlag, New York, 2003.

[2] M. Störring, H. J. Andersen, and E. Granurn, "Skin color detection under changing lighting conditions", 7th Symposium on Intelligent Robotics Systems, Coimbra, Portugal, 20-23 July 1999.

[3] M.H. Yang and N. Ahuja, "Detecting human faces in color images", Proceedings of IEEE Conference on Image Processing, Chicago, IL ,USA, 4-7 Oct 1998, pp. 127–130.

[4] G. Yang and T. S. Huang, "Human face detection in complex background", Pattern Recognition, 1994, Vol. 27, No 1, pp. 53-63.

[5] M.J.T. Reinders, P.J.L. van Beek, B. Sankur, and J.C.A. van der Lubbe, "Facial feature localization and adaptation of a generic face model for model-based coding", Signal Processing: Image Communication, 1995, pp. 57–74.

[6] H. Schneiderman and T. Kanade, " A statistical model for 3D object detection applied to faces and cars", IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, SC , USA ,13-15 June 2000, pp. 746-751.

[7] H.A.Rowley, S. Baluja and T. Kanade, "Neural network-based face detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, January 1998, Vol. 20, No. 1, pp. 23-38.

[8] E. Helomas and B. K. Low. "Face detection: A survey", Computer Vision and Image Understanding , September 2001, Vol 83, Issue 3, pp. 236–274.

[9] K. Sung and T. Poggio, “Example-based learning for view-based human face detection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, January 1998, Vol. 20, No.1, pp39–51.

[10] T. Sakai, M. Nagao, and T. Kanade, “Computer analysis and classification of photographs of human faces”, Proc. First USA-Japan Computer Conference, NSF, Tokyo, October 1972, pp. 55-62.

[11] I. Craw, H. Ellis, and J. R. Lishman, “Automatic extraction of face-feature”, Pattern Recognition Letter, February 1987, Vol 5, No 2, pp. 183–187.

[12] S. Kawato and J. Ohya, “Real-time detection of nodding and head-shaking by directly detecting and tracking the between-eyes”, Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble , France, 28-30 March 2000, pp. 40-45.

[13] A. Jacquin and A. Eleftheriadis, “Automatic location tracking of faces and facial features in video sequences”, IEEE Int.Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland, June 1995.

[14] J. Wang and T. Tan, “ A new face detection method based on shape information”, Pattern Recognition Letters, June 2000, Vol 21, pp 463–471.

[15] A. L. Yuille, P.W. Hallinan, and D. S. Cohen, “Feature extraction from faces using deformable templates”, International Journal of Computer Vision, August 1992, Vol 8, No 2, pp. 99–111.

[16] J. Choi, S. Kim, and P. Rhee, “Facial components segmentation for extracting facial feature”, Proc. 2.nd Int. Conf. on Audio- and Video-based Biometric Person Authentication, Washington, D.C , USA, 22-23 March , 1999.

[17] R. Herpers, K.H. Lichtenauer, and G. Sommer, “Edge and keypoint detection in facial regions”, IEEE Proc. of 2nd Int. Conf. on Automatic Face and Gesture Recognition, Killington, VT , USA, 14-16 Oct 1996, pp. 212–217.

[18] V. Govindaraju, “Locating Human Faces in Photographs,” Int’l Journal of Computer Vision, August 1996, Vol. 19, no. 2, pp. 129-146.

[19] T. S. Jebara and A. Pentland, “Parameterized structure form motion for 3D adaptive feedback tracking of faces”, IEEE Conf. on Computer Vision and Pattern Recognition, San Juan , Puerto Rico, 17-19 Jun 1997, pp. 144–150.

[20] S. Satoh, Y. Nakamura, and T. Kanade, “Name-it: naming and detecting faces in news videos”, IEEE Multimedia, Jan-Mar 1999, Vol 6, Issue 1, pp. 22–35.

[21] J. L. Crowley and F. Berard, “Multi-modal tracking of faces for video communications”, IEEE Int. Conf. on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 17-19 Jun 1997, pp. 640-645.

[22] N. Oliver, A. Pentland, and F. Berard, “A real-time face and lips tracker with facial expression recognition”, Trans. on Pattern Recognition, 2000, Vol 33, pp. 1369–1382.

[23] S. Kim, N. Kim, S. C. Ahn, and H. Kim, “Object oriented face detection using range and color information”, Proc. of the 3rd Int. Conf. on Automatic Face and Gesture Recognition, Nara, JAPAN, 14-16 Apr 1998, pp. 76–81.

[24] R. Kjedsen and J. Kender, “Finding skin in color images”, Proc. of the 2nd Int. Conf. on Automatic Face and Gesture Recognition, Killington, Vermont, 14-16 Oct 1996, pp. 312–317.

[25] K. Sobottka and I. Pitas, “Extraction of facial regions and features using color and shape information”, Proc. of 13th Int. Conf. on Pattern Recognition, Vienna , Austria, 25-29 Aug 1996, pp. 421-425.

[26] H. Wang and S.-F Chang, “A highly efficient system for automatic face region detection in mpeg video”, IEEE Trans. on Circuits and Systems for Video Technology , Aug 1997, Vol 7, Issue 4, pp. 615–628.

[27] Q. Chen, H. Wu, and M. Yachida, “Face detection by fuzzy Pattern matching”, Proc. of the 5th IEEE Conf. on Computer Vision, Cambridge, MA, 20-23 Jun 1995, pp. 591-596.

[28] J.-C. Terrillon, M. Shirazi, H. Fukamachi, and S. Akamatsu, “Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images”, Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble , France, 28-30 March 2000, pp. 54-61.

[29] J. Cai and A. Goshtasby, “Detecting human faces in color images”, Image and Vision Computing, 8 December 1999, Vol 18, Issue 1, pp. 63-75.

[30] A. Albiol, C. A. Bouman, and E. J. Delp, “Face detection for pseudosemantic labeling in video databases”, Proceedings in International Conference on Image Processing, Kobe , Japan, 24-28 Oct 1999, pp. 607-611.

[31] J. Yang and A. Waibel, “A real-time face tracker”, IEEE Proc. of the 3rd Workshop on Applications of Computer Vision, Sarasota, FL ,USA, 2-4 Dec 1996, pp. 142-147.

[32] R. O. Duda, P. Hart, and D. G. Stork, “Pattern Classification”, Wiley-Interscience, 2 edition, 2001.

[33] M. Turk and A. Pentland, “Eigenfaces for recognition”, Journal of Cognitive Neuroscience, 1991, Vol 3, no 1, pp. 71–86.

[34] F. Luthon and M. Lievin, “Lip motion automatic detection”, Scandinavian Conference on Image Analysis, Lappeenranta, Finland, June 9-12, 1997.

[35] S. McKenna, S. Gong, and H. Liddell, “Real-time tracking for an integrated face recognition system”, Workshop on Parallel Modelling of Neural Operators, Faro, Portugal, Nov. 1995.

[36] J. Miao, B. Yin, K.Wang, L. Shen, and X. Chen, “A hierarchical multiscale and multiangle system for human face detection in a complex background using gravity-center template”, Pattern Recognition, July 1999, Vol 32, Issue 7, pp. 1237–1248.

[37] Y. H. Kwon and N. da Vitoria Lobo, “Face detection using templates”, IEEE Int Conf. on Pattern Recognition, Jerusalem , Israel , 9-13 Oct 1994, pp. 764–767.

[38] A. Lanitis, C. J. Taylor, and T.F Cootes, “An automatic face identification system using flexible appearance models”, Image and Vision Computing, June1995, Vol 13, Issue 5, pp. 393–401.

[39] L. Sirovich and M. Kirby, “Low-dimensional procedure for the characterization of human faces”, Journ. Optical Society America, March 1987, Vol 4, pp. 519–524.

[40] A Pentland, B. Moghaddam, and T. Strarner, “View-based and modular eigenspaces for face recognition”, IEEE Proc. of Int. Conf. on ComputerVision and Pattern Recognition, Seattle, WA , USA, 21-23 Jun 1994, pp. 84–91.

[41] B. Moghaddam and A. Pentland, “Probabilistic visual learning for object representation”, IEEE Trans. Pattern Analysis and Machine Intelligence, July 1997, Vol 19, no 7, pp. 696-710.

[42] E. Viennet and F. Fogelman Souli´e, “Connectionist methods for human face processing, in Face Recognition: From Theory to Application”, Springer-Verlag, Berlin/New York, 1998.

[43] G. Burel and D. Carel, “Detection and localization of faces on digital images”, Pattern Recognition Letters, October 1994, Vol 15, no 9, pp. 963–967.

[44] P. Juell and R. Marsh, “A hierarchical neural network for human face detection”, Pattern Recognition Letters, May 1996, Vol 29, no 5, pp. 781–787.

[45] H. A. Rowley, “Neural network-based face detection”, PhD thesis, Carnegie Mellon University Computer Science Dept, May 1999.

[46] S.-H. Lin, S.-Y. Kung, and L.-J. Lin, “Face recognition/detection by probabilistic decision-based neural network”, IEEE Trans. Neural Networks, Jan 1997, Vol. 8, No. 1, pp. 114-132.

[47] D. Roth, M.-H. Yang, and N. Ahuja, “A snow-based face detector”, Advances in Neural Information Processing Systems, 2000.

[48] N. Littlestone, “Learning quickly when irrelevant attributes abound: A new linear threshold learning using winnow”, Machine Learning, 1988, Vol 2, pp 285–318.

[49] A. J. Colmenarez and T. S. Huang, “Face detection with information-based maximum discrimination”, IEEE Proc. of Int. Conf. on Computer Vision and Pattern Recognition, San Juan , Puerto Rico, 17-19 Jun 1997, pp. 782-787.

[50] C. Papageorgiou, M. Oren, and T. Poggio, “A general framework for object detection”, IEEE Proc. of 6th Int. Conf. on Computer Vision, Bombay , India, 1998, pp. 555–562.

[51] E. Osuna, R. Freund, and F. Girosi “Training support vector machines: An application to face detection”, IEEE Proc. of Int. Conf. on Computer Vision and Pattern Recognition, San Juan , Puerto Rico, 17-19 Jun 1997, pp. 130-136.

[52] V. Kumar and T. Poggio. “Learning-based approach to real time tracking and analysis of faces”, Proc. 4th IEEE Int. Conf. on Automatic Face and Gesture Recognition, Grenoble, France, 28-30 Mar 2000, pp. 96-101.

[53] H. Schneiderman and T. Kanade, “Probabilistic modeling of local appearance and spatial relationships for object recognition”, IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA , USA, 23-25 Jun 1998, pp. 45-91.

[54] YANG, M., AND AHUJA, N., “Gaussian mixture model for human skin color and its application in image and video databases”, In Proc. of the SPIE 99: Conf. on Storage and Retrieval for Image and Video Databases, 17 December 1998, Vol. 3656, pp. 458–466.

[55] YANG, J., LU, W., AND WAIBEL, A., “Skin-color modeling and adaptation”, In Proceedings of ACCV 1998, Hong Kong, China, 8-10 January 1998, Vol 2, pp. 687–694.

[56] BRAND, J., AND MASON, J., “A comparative assessment of three approaches to pixel level human skin-detection”, In Proc. of the International Conference on Pattern Recognition, Barcelona , Spain , 3-7 Sep 2000, Vol. 1, pp. 1056–1059.

[57] JONES, M. J., AND REHG, J. M., “Statistical color models with application to skin detection”, In Proc. of the CVPR ’99, Fort Collins, CO , USA, 23-25 Jun 1999, Vol. 1, pp. 274–280.

[58] SKARBEK, W., AND KOSCHAN, A., “Colour image segmentation – a survey”, Institute for Technical Informatics, Technical University of Berlin, October 1994.

[59]POYNTON, C. A., “Frequently asked questions about colour”, In ftp://www.inforamp.net/pub/users/poynton/doc/colour/ColorFAQ.ps.gz, 1995.

[60] BROWN, D., CRAW, I., AND LEWTHWAITE, J., “A som based approach to skin detection with application in real time systems”, In Proc.of the British Machine Vision Conference, Manchester, UK, 10-13 September 2001.

[61] PHUNG, S. L., BOUZERDOUM, A., AND CHAI, D, “A novel skin color model in ycbcr color space and its application to human face detection”, In IEEE International Conference on Image Processing(ICIP’2002) , 10 December 2002, Vol. 1, pp. 289–292.

[62] ZARIT, B. D., SUPER, B. J., AND QUEK, F. K. H, “Comparison of five color models in skin pixel classification”, In ICCV’99 Int’l Workshop on recognition, analysis and tracking of faces and gestures in Real-Timesystems, Corfu , Greece, 26-27 Sep 1999, pp. 58–63.

[63] HSU, R.-L., ABDEL-MOTTALEB, M., AND JAIN, A. K., “Face detection in color images”, IEEE Trans. Pattern Analysis and Machine Intelligence, May 2002, Vol 24, no 5, pp. 696–706.

[64] Jain, A. K., Murty, M. N., and Flynn, P. J., "Data Clustering: A Review", ACM Computing Surveys, September1999, Vol. 31, no. 3, pp. 264-323.

[65] Garrett-Mayer,E. and Parmigiani,G., "Clustering and Classification Methods for Gene Expression Data Analysis", Johns Hopkins University, Dept.of Biostatistics Working Papers, Johns Hopkins University, The Berkeley Electronic Press(bepress), http://www.bepress.com/jhubiostat/.,2004.

[66] Jiang, D., Tang, C., and Zhang, A., "Cluster Analysis for Gene Express Data: A Survey", IEEE Transactions on Knowledge and data Engineering, November 2004, Vol. 16, no. 11, pp. 1370-1386.

[67] Blum, C. and Roli, A., "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison", ACM Computing Surveys, September 2003, Vol. 35, no. 3, pp. 268-308.

[68] Ruspini, E. H., "A New Approach to Clustering", Information and Control, 1969, Vol. 15, no. 1, pp. 22-32.

[69] Dunn, J. C., "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", Journal of Cybernetics, 1973, Vol. 3, no. 3, pp. 32-57.

[70] Bezdek, J., “Pattern Recognition With Fuzzy Objective Function Algorithms”, Plenum. New York, 1981.

[71] Lampinen, T., Koivisto, H., and Honkanen, T., "Profiling Network Applications with Fuzzy C-Means Clustering and Self-organizing Map", in Proceedings of 1st International Conference on Fuzzy Systems and Knowledge Discovery: Computational Intelligence for the E-Age, Orchid Country Club, Singapore, 18 -22 November 2002, Vol. 1, pp. 300-394.

[72] Zhao, Y. and Karypis, G., "Soft Clustering Criterion Functions for Partitional Document Clustering: A Summary of Results", in Proceedings of CIKM-04, 13th ACM International Conference on Information and Knowledge Management, Hyatt Arlington Hotel, Washington D.C., USA, 8-13 November 2004, pp. 246-247.

[73] Hamerly, G. and Elkan, C., "Alternatives to the k-means algorithm that find better clusterings", in Proceedings of CIKM-02, 11th ACM International Conference on Information and Knowledge Management, McLean, VA (near Washington, DC), 4-9 November 2002, pp. 600-607.

[74] Xie, X. L. and Beni, G., "A Validity Measure for Fuzzy Clustering", IEEE Trans.Pattern Analysis and Machine Intelligence, 1991, Vol. 13, no. 8, pp. 841-847.

[75] Halkidi, M., Batistakis, Y., and Vazirgiannis, M., "On Clustering Validation Techniques", Journal of Intelligent Information Systems, December 2001, Vol. 17, no. 2, pp. 107-145.

[76] P.Viola, M. Jones, “Robust Real-time Object Detection”, Second International Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing, and Sampling VANCOUVER, CANADA, 2001.

[77] NOBUYUKI OTSU , “A Threshold Selection Method from Gray-Level Histograms”, IEEE Transactions on Systems, Man and Cybernetics, Jan 1979, Vol 9, Issue 1, pp. 62 – 66.