Face Detection Using Skin Color Segmentation

Ritu[1], K K Verma[2]
[1]*M.Tech Scholar, DPG Institute of Technology & Management, Gurgaon*
[2]*Assistant Professor, DPG Institute of Technology & Management, Gurgaon*

Abstract— Face detection is a computer coding technology that determines the locations and sizes of human faces in a given image format. Basically, it detects only the facial features and ignores the rest like trees, building, etc. This is the definition of face detection: the system acquires an arbitrary image, and it can analyze the messages included in the image accurately, and determine the exact position and region of the face [4]. Face detection can detect whether or not a human face exists on the image, and if it does, then the numbers and positions of the human face/s on the image can be determined.

Face detection remains an open problem. The task of face detection is extremely trivial for humans, but it is a challenge to enable computers to carry out the same task. The problem of face detection in still images is more challenging and difficult when compared to the problem of face detection in video since emotion information can lead to probable regions where face could be located.

Keywords— Face detection, Computer coding

I. INTRODUCTION

We may summarize the basic problem definition of Face Detection by saying that we are given an input scene. The goal is to find a set of possible locations on the image satisfying certain conditions. We are subject to the constraint that we are able to find the faces from the scene in a short period of time.

The main objective of our thesis is to study existing techniques for face detection, and propose changes or modification for improvement of current methods. Our work involves initial extraction of regions of the image based on skin colour range. Face detection is the first step of face recognition; however, it involves many complexities such as postures, lighting, background, etc. There exists many approaches towards face detection such as colour based, neural networks and feature based techniques. Our approach is skin-color based which is robust, simple and effective.

Our algorithm of face detection system consists of three steps.

- Classification of each pixel in the given image as a skin pixel or a non-skin pixel.
- Identify different skin regions or the skin spread in the skin detected binary image by using connectivity analysis.
- Determine whether each of the skin regions identifies as a face or not. This is done using two parameters i.e. the height to width ratio of the skin detected regions and the percentage of skin in the rectangle confined by the height and width.

Many researchers have proposed different methods addressing the problem of face detection. Face detection techniques are classified in to feature based and image based. The feature based techniques use edge information, skin color, motion and symmetry measures, point distribution, et al [1]. Image based techniques include neural networks, linear subspace method like Eigen faces, fisher faces etc. A number of face detection algorithms such as those using Eigen faces [2] and neural networks [3], for instance, have been developed. In these algorithms, however, a large amount of numerical computation is required, making the processing extremely time-consuming.

II. FACE MAPPING

Edge-based feature maps are the very bases several image representation algorithm [11], [12]. The feature maps represent the distribution of four-direction edges extracted from a 64x64 pixel input image. The input image is first subjected to pixel by pixel spatial filtering operations using kernels of 5x5 pixel sizes to detect edges in four directions, i.e. horizontal, +45 degree, vertical, or -45 degree. The threshold for edge detection is determined taking the local variance of luminance data into account. Namely, the median of the 40 values of neighboring pixel intensity differences in a 5x5 pixel kernel is adopted as the threshold. This is quite important to retain all essential features in an input image in the feature maps. Figure2 shows an example of feature maps generated from the same person under different illumination conditions. The edge information is very well extracted from both bright and dark images. 64-dimension feature vectors are generated from feature maps by taking the spatial distribution histograms of edge flags. In this work, three types of feature vectors, two general-purpose vectors and a face-specific vector generated from the same set of feature maps were employed to perform multiple-clue face detection algorithm [3]. Figure2 illustrates the feature vector generation procedure in the projected principal-edge distribution (PPED) [12].

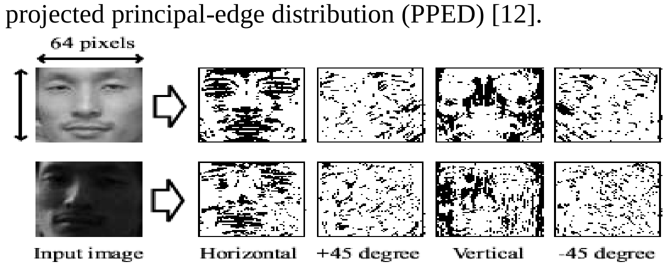


Fig.1: Feature maps generated from bright and dark images.

This provides a general purpose vector. In the horizontal edge map, for example, edge flags in every four rows are accumulated and the spatial distribution of edge flags along the vertical axis is represented by a histogram. Similar histograms are generated from three other feature maps, and a 64- dimension vector is formed by concatenating the four histograms. Generation of the other general-purpose vector called the cell edge distribution (CED) vector is illustrated in Figure3. Each feature map is divided into 4x4 cells. Each element in a CED vector indicates the number of edge flags in the corresponding cell. A face specific feature vector generation scheme called the eyes-and-mouth (EM) extraction is shown in Figure4. Two16-pixel-high bands of rows are cut from the horizontal feature map. They correspond to the location of eyes and a mouth when the 64x64 pixel window encloses a human face. Then the number of edge flags in two neighboring columns is counted to yield a single vector element in a 64-dimension EM vector.

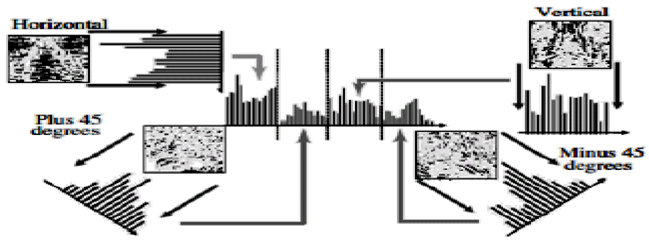


Fig.2: Feature vector generation based on projected principal-edge distribution (PPED).

Face detection is carried out in two steps: the coarse selection step and the fine selection step. In the coarse selection step, a 64x64-pixel area is taken from the input image and a feature vector is generated. Then, the feature vector is matched with all template vectors of face samples and non-face samples stored in the system and classified as a face or a non-face according to the category of the best-matched template vector. The matching is carried out using the Manhattan distance as the dissimilarity measure. Template matching is performed using three feature vector generation algorithms: PPED, CED and EM. If a local image is classified as a face by all the three vector representations, then it is adopted as a face candidate. This classification is carried out by pixel-by-pixel scanning of the 64x64-pixel window over the entire image.

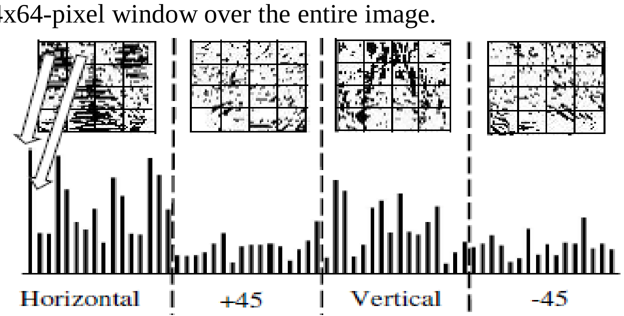


Fig.3: Feature vector generation based on cell edge distribution (CED) algorithm.

In neural networks are trained with a training set of face images. The test image is then fed into the neural network for faces to be detected. In practice the network will not receive a perfect image of face which represented by vector as input. Specifically, the network should make as few mistakes as possible when classifying images with noise of mean 0 and standard deviation of 0.2 or less [13, 14]. Advantage of using a NN is that the system can be easily be upgraded to recognize faces.

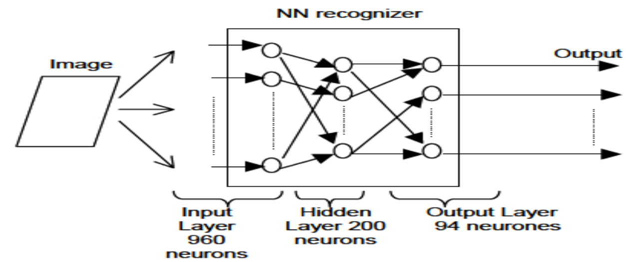


Figure 4: Neural Network based face detection

III. COLOUR BASED FACE DETECTION

For detecting face there are various skin color based algorithms have been proposed. Color is an important feature of human faces. Using skin-color as a feature for tracking a face has several advantages. Color processing is much faster than processing other facial features. Under certain lighting conditions, color is orientation invariant [15].However, color is not a physical phenomenon; it is a perceptual phenomenon that is related to the spectral characteristics of electromagnetic radiation in the visible wavelengths striking the retina. Tracking human faces using color as a feature has several problems like the color representation of a face obtained by a camera is influenced by many factors (ambient light, object movement, etc.), different cameras produce significantly different color values even for the same person under the same lighting conditions and skin color differs from person to person[16]. It is also robust towards changes in orientation and scaling and can tolerate occlusion well. A disadvantage of the color cue is its sensitivity to illumination color changes and, especially in the case of RGB, sensitivity to illumination intensity.

It would be fair to say that the most popular algorithm to face localization is the use of color information, whereby estimating areas with skin color is often the first vital step of such strategy. Hence, skin color classification has become an important task. Much of the research in skin color based face localization and detection is based on RGB, YCbCr and HSI color spaces. YCbCr and HSI are two most commonly used colour models for digital processing. In our work we have used the HIS model. Kjeldson and Kender defined a color predicate in HSV color space to separate skin regions from background [15]. Here the responsible values are hue (H) and saturation (S). If the threshold is chosen as [*H1*, *S1*] and [*H2*, *S2*], and a pixel is classified to have skin tone if the values [*H*,*S*] fall within the threshold and this distribution gives the localized skin image. Similar to above algorithm our approach is also having the same constraint.

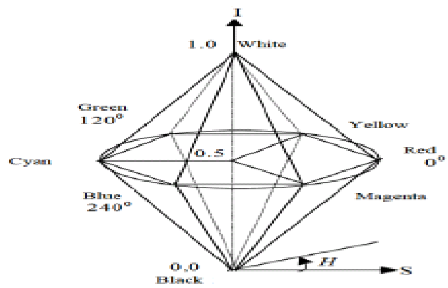


Fig.5: Double cone model of HIS colour space

Clustering is the process of division of a data set into subsets or clusters, so that the similarity of points in each partition is as high as possible, while points in different partitions are dissimilar [17]. Clustering algorithms attempt to separate a dataset into distinct regions. The aim of cluster analysis is to partition a data set into a number of disjoint groups or clusters. The members within a cluster are more similar to each other than members from different clusters. For face detection we cluster an image and then analyze the result to detect probable facial regions. In our work, we have used a grid-density based approach [18], where we attempt to cluster the objects in colour images by transforming the image data into *HSI* attributes and then performing a grid based clustering.

IV. SKIN BASED FACE DETECTION

Skin detection is the process of finding skin-colored pixels and regions in an image or a video. This process is typically used as a preprocessing step to find regions that potentially have human faces and limbs in images. Several computer vision approaches have been developed for skin detection. A skin detector typically transforms a given pixel into an appropriate color space and then uses a skin classifier to label the pixel whether it is a skin or a non-skin pixel. A skin classifier defines a decision boundary of the skin color class in the color space based on a training database of skin-colored pixels [19].

1. Skin classifier :

Skin Classifiers A variety of classification techniques have been used in the literature for the task of skin classification. A skin classifier is a one-class classifier that defines a decision boundary of the skin color class in a feature space. The feature space in the context of skin detection is simply the color space chosen. Any pixel which color falls inside the skin color class boundary is labeled as skin. Therefore, the choice of the skin classifier is directly induced by the shape of the skin class in the color space chosen by a skin detector. The more compact and regularly shaped the skin color class, the more simple the classifier. The simplest way to decide whether a pixel is skin color or not is to explicitly define a boundary. Brand and Mason [28] constructed a simple one-dimensional skin classifier: a pixel is labeled as a skin if the ratio between its R and G channels is between a lower and an upper bound. They also experimented with one-dimensional threshold on IQ plane of YIQ space where the "I" value is used for thresholding. Other methods explicitly define the skin color class boundary in a two dimensional color space using elliptical boundary models [29]. The parameters of the elliptical boundary can be estimated from the skin database at the raining phase. Baysian Approach for Skin Detection: Skin classification can be defined probabilistically as: given a pixel with color c what is the probability of it being skin pixel P(skin|c). Once this probability is computed, the pixel is labeled as a skin pixel if such probability is larger than a threshold and non-skin otherwise. Obviously we cannot compute such probabilities for every possible color (e.g., in 24 bit RGB, there are 2563 colors). Fortunately, using Bayes rule, this can be rewritten as

$$P(skin|c) = \frac{P(c|skin)P(skin)}{P(c|skin)P(skin) + P(c|notskin)P(notskin)} \quad \text{…………1}$$

Bayes rule defines the posterior probability of a pixel being skin given its color (P (skin|c)) in terms of the likelihood of observing such color given the skin class (P (c|skin)) and the prior probability of the skin class P (skin). The prior probability measures our guess about a random pixel being a skin without observing its color. The denominator in the Bayes rule is the total probability of observing the color c, a factor that does not affect the decision whether a pixel ought to be labeled as skin or non-skin. Given Bayes rule, the skin classification reduces to computing the likelihood term, i.e., P (c|skin). Given a database of skin-colored pixels we can estimate the probability density function (pdf) of P (c|skin). Several approaches have been introduced to compute this pdf including the use of histograms [30], the use of a single Gaussian model, or a Mixture of Gaussians model [31] to approximate such pdf. The skin classifier can also be posed as a two-class problem. From Bayes rule, this results in computing the likelihood ratio of observing a given color given a skin class versus a non-skin class, i.e., P(c|skin)/P(c|notskin). Such ratio can then be threshold to decide whether a pixel is a skin or non-skin pixel. Besides modeling the likelihood of an observed color given the skin class, the complementary class needs to be models. That is, modeling the probability density function of non-skin pixels P(c|notskin). Rehg and Jones [30] approximated such pdfs using 3D histograms in the RGB space based on a large database of skin and non-skin images

2. Color Space

A color space is an abstract mathematical model describing the way colors can be represented as tuples of numbers, typically as three as three or four values as color components. A wide range of colors can be created by the primary color of pigment. Those colors then define a specific color space. The resulting 3-D space then provides a unique position for every possible color that is possible by combining those three pigments.

A color space is actually a specific organization of colors which in combination with physical device profiling, allows us for reproducible representation of colors in both digital and analog representations [20]. This is a special tool for understanding the color capabilities of a particular device or digital file. When we try to reproduce color on another device, then this is the color space which can show us whether we will be able to retain shadow or highlight detail, color saturation and by how much either can be compromised[21].

We know usually color can be measured by the following attributes [22]:

- Brightness: This is the human sensation by which an area exhibits more or less light.
- Hue: This is the human sensation according to which an area appears to be similar to one, or to proportions of two, of the perceived colors red, yellow, green and blue.
- Colorfulness: This is the human sensation according to which an area appears to exhibit more or less of its hue.
- Lightness: This can be defined as the sensation of an area‘s brightness relative to a reference white in the scene.
- Chroma: This is the colorfulness of an area relative to the brightness of a reference white.
- Saturation: This is the colorfulness of an area relative to its brightness.

V. COLOR SPACE

RGB uses additive color mixing, because it describes what kind of light needs to be emitted to produce a given color. RGB stores individual values for red, green and blue. RGBA is RGB with an additional channel, alpha, to indicate transparency.

HSV (hue, saturation, value), also known as HSB (hue, saturation, brightness) is often used for artististical purposes because it is often more ideal to think about a color in terms of hue and saturation than in terms of additive and subtractive color components. HSV is a transformation of an RGB color space and its components and colorimetry are proportional to the RGB color space from which it was derived.

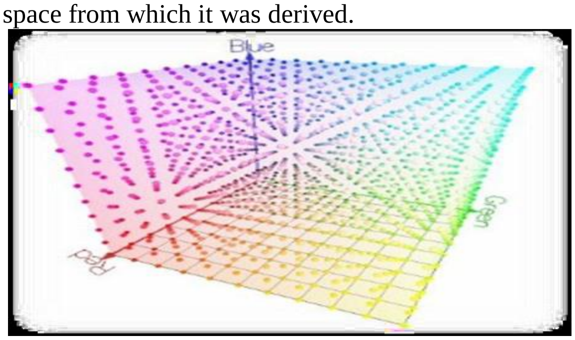


Fig.6: RGB Colors pace

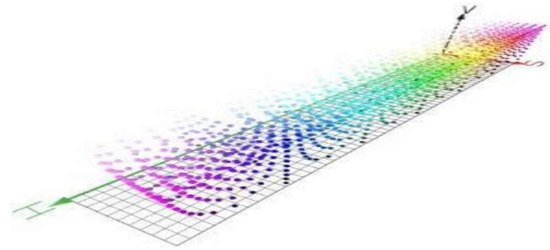


Fig.7: HSV Color space

In the HSV space, H stands for hue component, which describes the shade of the color; S stands for saturation component, which describes pureness of the hue (color) while V stands for value component, which describes the brightness. The removal of V component takes care of varying lighting conditions.

YCbCr is a family of color space used as a part of the color image pipeline in digital and video systems. Y is the luma component and CB and CR are the blue-difference and red-difference Chroma components. Y′ (with prime) is distinguished from Y which is luminance, which means that light intensity is non-linearly encoded using gamma correction. YCbCr is not an absolute color space but it is a way of encoding RGB information. The color displayed depends on the actual RGB used to display the signal.

$Y = 0.299R + 0.587G + 0.144B$

$Cb = R - Y\ ,\ Cr = B - Y$

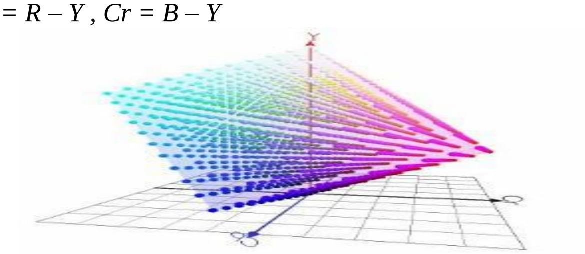


Fig.8: YCbCr Color space

This color space is originally defined by CAE and specified by the International Commission on Illumination [24] [25]. In this color space, we have one channel is for Luminance (Lightness) and other two color channels are ‘a’ and ‘b’ known as chromaticity layers. The ‘a*’ layer indicates where the color falls along the red green axis, and ‘b*’ layer indicates where the color falls along the blue-yellow axis. ‘a*’ negative values indicate green while positive values indicate magenta; and ‘b*’ negative values indicate blue and positive values indicate yellow. Most important feature of this color space is that this is device independent [26], means to say that this provides us the opportunity to communicate different colors across different devices. Following figure clearly illustrates the coordinate system of L*a*b* color space [27]:

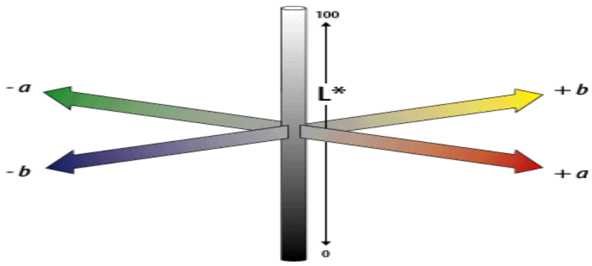


Fig.9: L*a*b Color space

In the above figure, the central vertical axis represents lightness (L*). L* can take values from 0(black) to 100(white). The coordinate axes follows the fact that a color cannot be both red or green, or both blue and yellow, because these colors oppose each other. Values run from positive to negative for each axis. Positive 'a' values indicate amounts of red, while negative values indicate amounts of green. And, positive 'b' values indicate amount of yellow while, negative 'b' values indicate blue. The zero represents neutral gray for both the axes. So, here, values are needed only for two color axes and for the lightness or gray scale axis (L*).

VI. PROPOSED ALGORITHM

The input image is processed primarily and RGB mode is being converted to L*a*b* color space. The image is then analyzed for getting pixel values for skin and non-skin region. After identification of pixel values, each color component will be given a threshold value. Then the skin and non-skin pixels will be separated via binary skin classifier where black and white represents the skin and non-skin region. Finally it will detect the face region easily and define the face boundary. The absence of the illuminance component increases performance, which also supports in finding the appropriate color space for skin detection. Combination of color components reduces the computational complexity and processing time.

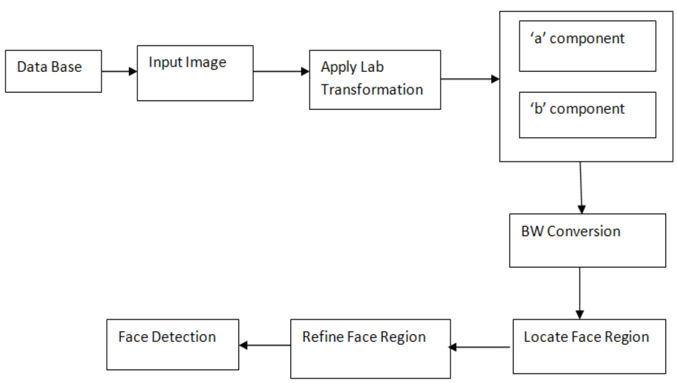


Fig.10: Block Diagram of the Process

A color represented in one space can be changed to another spatial representation by performing some linear or non-linear transformation. Here we are using CIELAB color space for color transformation .CIE L*a*b* or CIELAB consists of the components such as luma, green (or)red and yellow(or) blue are in CIELAB. If a specific threshold is crossed positive or negative direction for a* or b* values you are navigating out of RGB color space gamut and therefore RGB values do not change anymore. Color transformation equation for L*a*b defined as:

$$L^\star = 116 f(Y/Y_n) - 16$$

$$a^\star = 500\left[f(X/X_n) - f(Y/Y_n)\right]$$

$$b^\star = 200\left[f(Y/Y_n) - f(Z/Z_n)\right]$$

Where

$$f(t) = \begin{cases} t^{1/3} & \text{if } t > (\frac{6}{29})^3 \\ \frac{1}{3}\left(\frac{29}{6}\right)^2 t + \frac{4}{29} & \text{otherwise} \end{cases}$$

Here, Xn, Yn and Zn are the CIE XYZ tri stimulus values of the reference white point (the subscript n suggests "normalized"). Under Illuminant D65, the values are

$$X_n = 95.047, Y_n = 100.000, Z_n = 108.883$$

The division of the domain of the f(t) function into two parts was done to prevent an infinite slope at t = 0. f(t) was assumed to be linear below some t = t0, and was assumed to match the t1/3 part of the function at t0 in both value and slope. In other words:

$$t_0^{1/3} = at_0 + b \quad \text{(match in value)}$$

$$\tfrac{1}{3}t_0^{-2/3} = a \quad \text{(match in slope)}$$

The intercept f(0) = b was chosen so that L* would be 0 for Y = 0: b = 16/116 = 4/29. The above two equations can be solved for a and t0:

$$a = \tfrac{1}{3}\delta^{-2} = 7.787037\ldots$$

$$t_0 = \delta^3 = 0.008856\ldots$$

Where δ = 6/29[7]

Read the input image from the database. Create color transformation which converts RGB into L*a*b color. Then apply transformation which coverts RGB into L*a*b color space.

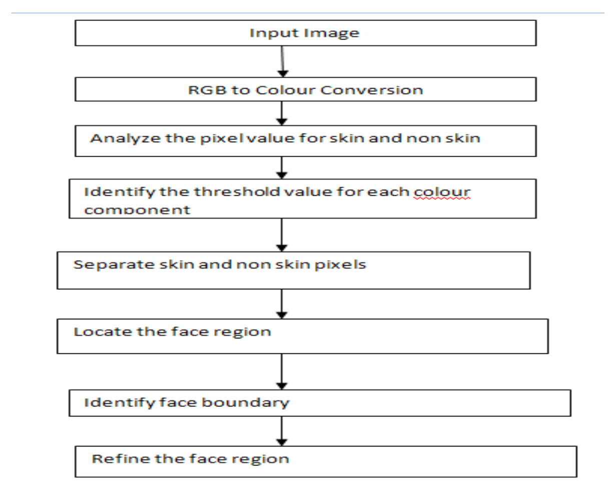


Fig.11: Flow diagram

Analyze the pixel value for skin and non-skin region through histogram analyses. Apply global threshold to convert intensity image into binary image for each color component in L*a*b. Separate skin and non-skin pixels through the conversion of grey scale image to binary image. Then locate the face region by combination of the chromo components of binary image. Define and refine the face by bounding box region properties.

Various input images with different face occlusion and poses (up, left. right, down) are provided. The images are processed under L*a*b color transformation result, the components of various elements such as lightness (L), a* and b* where a* indicates negative as green, positive as magenta and b* indicates position between yellow and blue. Conversion of a* and b* image components are converted individually leaving Luma component. The resultant images of binary component of a* and b* are combined to detect the face using region properties in MATLAB.

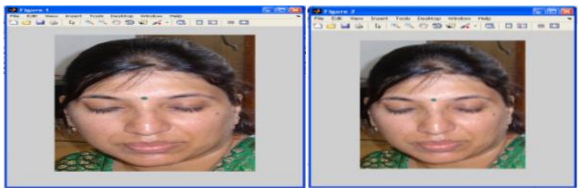
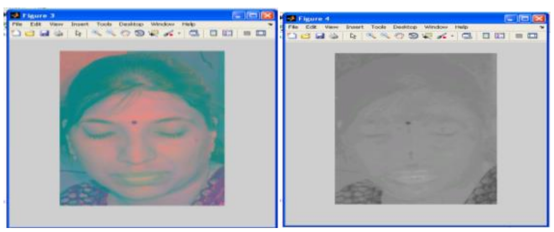
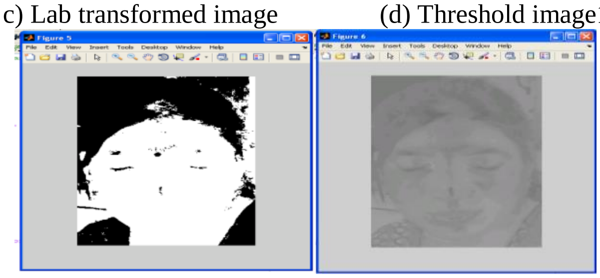


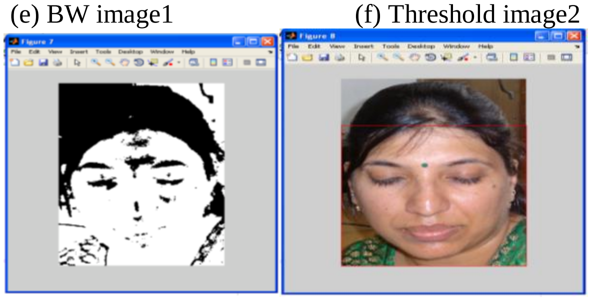
Fig.12: (a) Input image (b) Resize image



c) Lab transformed image (d) Threshold image1



(e) BW image1 (f) Threshold image2



(g) BW image2 (h) face detection

Fig.12: Face Detection of image 1 from data base

The face detection algorithm were thoroughly studied and put to practical use by simulations in MATLAB 7.10.0. Simulations were done on various images under different conditions and the error and success rates were recorded. The success rate was different for different images depending on the external factors. The overall success rate was found to be 95.33%.

Face detection is an important aspect for various fields of study such as face recognition, expression detection, video monitoring, status authentication, and others for which it remains an important research field till date. In this approach detected the face region more effectively from the given photos. The binary classifier where black and white represents the skin and non-skin region. Here the illuminant component is dropped and the chromo components were used which helps the identification better more. The absence of the light component increases the performance and that also for supporting to find the skin detection with appropriate color space. Finally it detects the face region easily and defines the face boundary. The experimental result shows that the proposed method is invariant to the lighting condition under which the image was taken. The results also revealed the robustness and efficiency of this method, for varying conditions such as pose, expression and lighting conditions.

Face detection using skin color likelihood based on facial landmarks to locally deal with face occlusion and face pose (up, down, right, left,) estimation problems in details. Another topic is to find feature descriptor that provides good representation of skin color in the presence of severe lighting condition. These feature descriptors improve this binary skin segmentation better performance and further reduce the false detecting rate in dealing with images with more complex back ground and multiple faces.

VII. References

[1] P. Kakumanu, S. Makrogiannis, N. Bourbakis, "A Survey of Skin-Color Modeling and Detection Methods", Pattern Recognition 40, pp 1106-1122, 2007. available at www.sciencedirect.com.

[2] “Face detection” , Inseong Kim, Joon Hyung Shim, and Jinkyu Yang .

[3] [3]Rafel C. Gonzalez, Richard E. Woods, “ Digital Image Processing” Third edition

[4] C.feng, Quwang, Yunming Du. “Face Detection Based on Membership and Geometry Feature”, Journal, Information Technology, Harbin, 2007, 31(8), pp.84-87.

[5] http://www.bioid.com/index.php?q=downloads/software/bioid-face-database.html accessed 2013.01.13.

[6] http://wenku.baidu.com/view/3f5c70946bec0975f465e225.htm accessed 2013.01.13

[7] Y. Ming “Detecting Human Faces in Color Images” from Beckman Institute and Department of Electrical and Computer Engineering 2009 15th.

[8] Chellappa R., Wilson C. L., and Sirohey S., “Human and machine recognition of faces: A survey”. Proceedings of the IEEE,83(5):705–740, 1995.

[9] Kondo T. and Yan H., .Automatic human face detectionand recognition under nonuniform illumination,.PatternRecognition Letters, vol. 32, pp.1707-1718, 1999.

[10] Rowley H., Baluja S., and Kanade T., Neural Network-Based Face Detection,.IEEE Trans. Pattern Analysisand Machine Intelligence, Vol. 20, no. 1, pp. 23-38, Jan.1998.

[11] Suzuki Y. and Shibata T., “Multiple-Clue Face Detection Algorithm Using Edge-Based Feature Vectors”,.accepted for presentation in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004), May 2004.

[12] M. Yagi and T. Shibata, .An Image Representation Algorithm Compatible With Neural-Associative-Processor-Based Hardware Recognition Systems-IEEE Trans. Neural Networks, Vol. 14, no. 5, pp.1144-1161, Sep. 2003.

[13] [13] Rowley H., A.,’’Neural Network-based face detection’’ PhD thesis, Carnegie Mellon University, Pittsburgh, USA, May 1999.

[14] [14] Mitchell T., ‘’Neural net & Face images” , Home work 3, CMU, Carnegie Mellon University, Pittsburgh, USA, October 1997.

[15] Kjeldsen, R. and Kender., J., “Finding Skin in Color Images,” *Proceedings of theSecond International Conference onAutomatic Face and Gesture Recognition*, pp. 312-317 (1996).

[16] *Suzuki* , Y. *and Shibata, T.*, “AN EDGE-BASED FACE DETECTION ALGORITHM ROBUST AGAINSTILLUMINATION, FOCUS, AND SCALE VARIATIONS”.

[17] Han, J., &Kamber, M. *Data Mining: Concepts and Techniques*. India: Morgan Kaufmann Publishers, 2004.

[18] Sharma, S., Das, R. & Bhattacharya, D. K., “DGDCT: Distributed Grid-Density Based Algorithm for Intrinsic Cluster Detection over Massively Spatial Data”, accepted for presentation in IEEE ICDCN, Kolkata, 2008.

[19] Skin Detection - a Short Tutorial † Ahmed Elgammal, Crystal Muang and Dunxu Hu Department of Computer Science, Rutgers University, Piscataway, NJ, 08902, USA

[20] http://en.wikipedia.org/wiki/Color_space

[21] http://www.cambridgeincolour.com/tutorials/color-spaces.htm

[22] Charles Poynton, ―A Guided Tour of Color Space‖, New Foundations for Video Technology (Proceedings of the SMTPE Advanced Television and Electronic Imaging Conference), 1995, pp. 167-180.

[23] http://in.mathworks.com/help/images/understanding-color-spaces-and-color-space-conversion.html?refresh=true

[24] Hunter, RichardSewall (July 1948). "photoelectric color-difference meter". Josa 38 (7): 661. (Proceedings of the winter meeting of the optical society of America)

[25] Hunter, RichardSewall (December 1948). "Accuracy, precision, and stability of new photo-electric color-difference meter". Josa 38 (12): 1094. (Proceedings of the thirty-third annual meeting of the optical society of America)

[26] http://en.wikipedia.org/wiki/Lab_color_space

[27] http://dba.med.sc.edu/price/irf/Adobe_tg/models/cielab.html