An adaptive fuzzy technique for real-time detection of multiple faces against a complex background

Dyut Kumar Sil
BCSE, Final year Student
Jadavpur University

Subhadip Basu
Senior Lecturer, CSE Dept.,
Jadavpur University

Mita Nasipuri
Professor, CSE Dept.,
Jadavpur University

ABSTRACT

This paper describes a real-time face detection system which is capable of processing video frames extremely rapidly while achieving high detection rate. The primary contribution of this paper is development of a fast algorithm for partitioning each frame into sub-images, detection of potential facial sub-images and real-time clustering of such potential sub-images into isolated objects/faces. A set of experiments in the domain of real-time face detection are presented. The performance of the system is comparable to some well known previous systems [5, 6, 8, 9]. Being implemented on a conventional desktop, face detection could be done at the rate of 13 frames per second.

Categories and Subject Descriptors

H.2.0 [**Image**]: Real-time image processing, computer vision, algorithm implementation and evaluation.

General Terms

Algorithms, Performance, Experimentation,

Keywords

Real-time clustering, computer vision, face detection.

1. INTRODUCTION

Face detection technology determines the location of human faces in digital images using different facial attributes. It detects facial features and ignores anything else, such as buildings, trees and bodies. *Face detection* can be regarded as a specific case of object-class detection; in object-class detection, the task is to find the locations and sizes of all objects in an image that belong to a given class. It is the first step of face recognition and is greatly used in facial recognition modules for designing automatic security systems to check whether any person has entered in a prohibited area, to design a computerized system for foot-fall analysis of particular place and other biometric authentication systems. Some recent digital cameras use face detection for auto focus feature. *Face detection* is also the first step of Face Perception which is the process by which the brain and mind understand and interpret the face, particularly the human face.

Much work has already been done on *face detection* techniques. Most of these are based on shape and relative position of eyes. The Gradient technique proposed by Kukharev, Masicz and Masicz[1] detects any object of facial shape as face which may not be a face. However, this technique has the advantage that it works indifferently under changing lighting condition. There are also techniques those use skin color to detect face. R. Féraud *et al*.[2] proposed a fast and simple face localization technique based on motion and skin color filters. C.R. Wren *et.al.* [3] have used the color and spatial similarity properties to form connected regions in which all pixels have similar image properties. However, both the techniques use high-end, costly gadgets and complex computation techniques for segmentation and tracking of human activities. Rolf P. Würtz , in his work on object recognition [4], have used Gabor wavelet features for sampling the image, amplitude thresholding, background suppression and image representation. The computational complexities of these algorithms are high for a real-time application. Technique used by Blepo[6] works well in real-time and produce satisfactory result when faces are frontal to the camera but performance becomes unsatisfactory when orientations of faces change.

Though many works have been done, real-time face detection is still an open problem. This is mainly because of the complexity of the problem with respect to the variations in lighting conditions, face orientations and limitation on computation cost etc. In the current work we have tried to address this problem by developing an adaptive fuzzy technique to address the imprecision or vagueness there in.

2. PRESENT WORK

Our face detection procedure classifies images based on the values of HSL color components of a pixel. The HSL color model is less susceptible to changing lighting conditions in comparison to the RGB color model. This is the primary reason behind the choice of the said model.

2.1. RGB to HSL Computation

From the camera driver we are obtaining RGB color value for every pixel of each frame. So, the conversion from RGB color model to HSL color model must very simple to compute otherwise that may be a big cause for bad performance of the system in real-time. To reduce computation in conversion from RGB to HSL a very simple technique [7] has been used.

Let r, g, b Є [0,1] be the red, green, and blue coordinates, respectively, of a color in RGB space. Let max be the greatest of r, g, and b, and min the least.

To find the hue angle h [0, 360] for HSL space, we have computed:

$$h = \begin{cases} 0 & \text{if } \max = \min \\ (60^\circ \times \frac{g-b}{\max - \min} + 0^\circ) \bmod 360^\circ, & \text{if } \max = r \\ 60^\circ \times \frac{b-r}{\max - \min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max - \min} + 240^\circ, & \text{if } \max = b \end{cases}$$

To find saturation and lightness s, l [0,1] for HSL space, we have computed:

$$l = \tfrac{1}{2}(\max + \min)$$

$$s = \begin{cases} 0 & \text{if } \max = \min \\ \frac{\max - \min}{\max + \min} = \frac{\max - \min}{2l}, & \text{if } l \le \frac{1}{2} \\ \frac{\max - \min}{2 - (\max + \min)} = \frac{\max - \min}{2 - 2l}, & \text{if } l > \frac{1}{2} \end{cases}$$

2.2. Training of skin color pixel

Our technique needs an initial seed value of skin color pixel and also a range. To find this initial seed value and the range we have trained many face images under a given lighting condition. This procedure has been done offline. In this procedure, for each face image, maximum, minimum, mean and standard deviation value for each component of Hue, Saturation and Lighting has been computed. Mean value of each component has been taken as the seed value for that image. The computational methodologies for the seed and the range values are shown below:

Seed$_{ic}$ = mean$_{ic}$,

where, Seed$_{ic}$ denotes seed value and mean$_{ic}$ denotes mean value of c component, c Є {H, S, L} of ith image.

A heuristic function has been applied to determine the range.

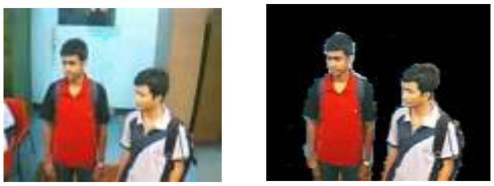
Range$_{ic}$ = (1 - μ).standard_deviation$_{ic}$ + μ.(max$_{ic}$ – min$_{ic}$)/2.

where, Range$_{ic}$ is allowed range, standard_deviation$_{ic}$ denotes standard deviation, max$_{ic}$ is maximum value, min$_{ic}$ is minimum value of c component, c Є {H, S, L} of ith image and μ is a constant obtained experimentally.

Now, final seed value of a component is obtained by taking median of all the seed values of corresponding component of all images. Similarly, final range of a component is also computed by taking median of ranges of corresponding component of all images.

2.3. Adaptive background estimation and subtraction

To reduce the amount of computation our technique subtracts the background information from the image before processing each image. This not only reduces the computational time but also helps to avoid skin colored object in background to be falsely detected as face. Initially an image of the background has been saved when no face was in the image. Background may change during the course of the processing. To be vigilant with the changing background the initially saved background image must have to be changed during the course of the processing. Our technique saves the current frame as the background image if the current frame and last η consecutive frames also had not detected any face.



(a) (b)

Figure -1: (a) before background subtraction and (b) after background subtraction

Now the question may come how the background is subtracted. The saved background image is exactly of same size that of a frame. Before processing the current frame first each pixel of the current frame is compared with corresponding pixel in the saved background image. Now if difference between the two pixels in all the 3 color components is below corresponding 3 thresholds, the pixel in the current frame is considered as a background pixel and it need not to be considered further in the process of detection of faces. The values of these pixels are set 0 for all the three color components so that they never selected as a potential skin colored pixel. The three thresholds one for each of the three color components, are obtained experimentally.

2.4. Adaptive seed value estimation

As discussed before, our technique has started with an initial seed skin color. This seed is used for processing first frame only. During processing of each frame, the seed value for next frame is being calculated. We have used a fuzzy membership function f, as shown below, to determine the membership value (Є [0, 1]) of each pixel in the fuzzy set of skin colored pixel.

$$f(x) = \frac{1}{\left(1 + \left(\frac{(x-c)}{a}\right)^{b}\right)}$$

where, x is the value of a HSL color component of the pixel, c is corresponding color component of seed pixel of the current frame, a & b are constant. We have considered b=2, a depends on the range of the particular color component.

The pixel having membership value >= δ for all three component are considered as skin pixels. Considering membership values as weights of these skin pixels, weighted average values for all three component is computed which are used as component of seed pixel for next frame. To avoid the possibility of a large transition of seed color of skin so that seed color become a complete different color from skin color, we have introduced a limitation upto which seed color may change from the initial seed color obtained from offline training.

2.5. Partitioning a test image frame

Our technique initially partitions the test image into 40X40 array of fixed size sub-image. We have worked taking image of size 320X240 which implies that the size of a sub-image is 8X6. We have assumed that size of a face to be detected is greater than 8X6. Analyzing a large scale of image taken in the location where we have tested our technique, it is revealed except in some cases all the face have size greater than the size of a sub- image. How partitioning is being done has been illustrated in the following figure using partitioning into 6X6 array of sub-images.

Figure -2: Fixed partitioning of the frame into 6X6 array

So, a face to be detected comprises of one or more sub-images. How a sub-image is selected as a part of a face and how such sub-images combined, have been explained in next two sections.

2.6. Design of a fuzzy skin color model

By the fuzzy function described in the section 2.4 we have got membership value for all components (H, S and L) for each pixel. A fuzzy function f' has been designed to get membership value of a pixel from the membership value of its components,
such that:

$$f'(p) = \alpha.f\left(p_{H}\right) + \beta.f\left(p_{S}\right) + \gamma.f\left(p_{L}\right)$$

where p is a pixel, f'(p) its membership value, α, β, γ are constant such that α + β + γ = 1, f is the function described in 2.4, p_H, p_S, p_L *are H, S, L component respectively of the pixel p. α, β, γ are obtained experimentally.*

Finally, we have considered mean of membership values of all the pixels f'' in the area of a sub-image as the membership value of the sub-image, such that:

$$f''(\xi) = \left(\frac{1}{n}\right) \sum_{(p \varepsilon \xi)} f'(p)$$

where ξ is a sub-image, n is number of pixel in a sub- image, f''(ξ) is membership value of the sub-image. If the membership value of a sub-image >= ω, it is considered (potentially) to be part of a face. The constant ω is obtained experimentally.

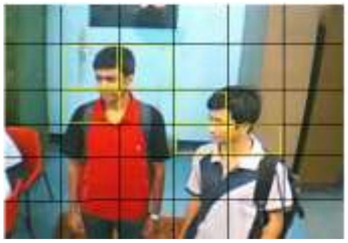


Figure -3: Partitions having membership value >= ω

2.7. Real time clustering of sub-images

After obtaining the sub-images those are potentially parts of some faces, our task is to divide the image into clusters such that each cluster contains all the sub-images that are parts of a single face, i.e. we have to combined the sub-images those are parts of a single face. For the need of the real-time application the clustering of the sub-images must have to be very fast i.e. the algorithm must be simple to compute. The algorithm we have used given following:

```
Initialize label, L Å -1
For all ξ scan the image from left to right and top to bottom
    IF f”(ξ)>=0.5 then
        Insert sub-images ξ to List and ξ.label Å L
            While List is not empty
                Remove a sub-image ψ from List
                For each 4-neighbour Φ of ψ
                    If Φ is unlabelled and f”(Φ)>=0.5 then
                        Insert Φ to List and Φ.label Å L
                End For
            End While
            L Å L – 1
End For
```

Sub-images having same label are merged to a single skin-colored cluster.

Figure -4: Partitions having same label are merged to a cluster

2.8. Elimination of noisy clusters

All the clusters obtained after clustering need not to be faces. Some may be other skin colored object such as hands, skin colored shirt those were not in the background. We have analyzed various face images to obtain a range of aspect ratio and size of face. Our technique filters out the noise clusters those having size out of the range of size of a face or those having aspect ratio out of the range of aspect ratio of a face. The clusters remained after filtering are the detected faces.

3. EXPERIMENTAL RESULT

In last chapter we have discussed about our present work. We have seen that many experimentally obtained has come in our discussion. These experimentally obtained constants and some other experimental results have been discussed in the following subsections of this chapter.

3.1. Experimentally obtained constants

In section 2.2, we have discussed about the constant µ which was used for calculation of allowed range of a color component of an image. The value of µ for which we have got best result has been given below:

$$\mu = 0.3$$

In section 2.3, we have discussed about the constant η which was used for updating of the background (basically η+1 is the number consecutive frame not containing faces causes to update the saved background to current frame). The value of η for which we have got best result has been given below:

$$\eta = 5$$

In section 2.4, we have discussed about the constants a and b which were used for evaluation of membership value of a color component of in the fuzzy set of the corresponding component of skin color pixels. Now in our experiment we have assumed b=2 for all the three color components and corresponding a has been obtained experimentally.

'a' for Hue = 5.00
'a' for Saturation = 0.23
'a' for Lightness = 0.16

In section 2.4, we have discussed about the constant δ which was used as minimum membership value for all the three color components H, S and L for the pixels which would be considered for determining the seed value for next frame. The value of δ for which we have got best result has been given below:

$$\delta = 0.5$$

In section 2.6, we have discussed about the constants α, β and γ which were used for evaluation of membership of a pixel in the fuzzy set of skin color pixel. The values of α, β and γ for which we have got best result has been given in the table below:

$$\alpha = 0.6$$
$$\beta = 0.3$$
$$\gamma = 0.1$$

In section 2.6, we have discussed about the constant ω which was used as minimum membership value of a sub-image to be considered as potential to be part of some face. The value of ω for which we have got best result has been given below:

$$\omega = 0.5$$

As discussed in section 2.2, we need an initial seed color value for skin color pixels in our present system of real-time face detection. Here in the following table we have given the values of all the three color components namely H, S and L of the initial seed color we have obtained in our experiment.

Value of Hue = 28
Value of Saturation = 0.75
Value of Lightness = 0.55

3.2. Allowed variance for considering skin color

As discussed in section 2.2, we need a variance for each color component which is allowed for a skin color pixel from the corresponding component of the seed color. Here in the following table we have given the allowed variances we have in our experiment of all the three color components namely H, S and L from the corresponding color component of the seed color.

Variance allowed for Hue = 5
Variance allowed for Saturation = 0.04
Variance allowed for Lightness = 0.08

As discussed in section 2.4, there is a limitation upto which the calculated seed color may change from the initial seed color. This limitation is obtained experimentally and given below:

Variance allowed for Hue = 2
Variance allowed for Saturation = 0.04
Variance allowed for Lightness = 0.10

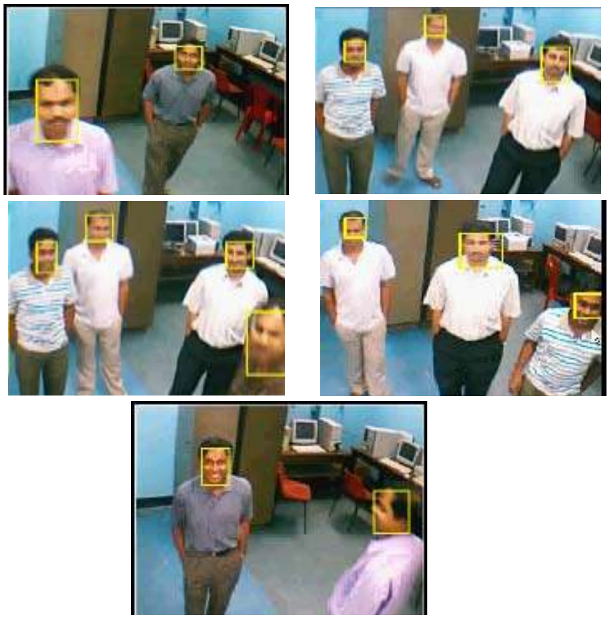


Figure -5: Some successful cases of our Real-time Face Detection System

3.3. Design of the filtering criteria

As discussed in section 2.8, noisy clusters have been filtered out using the range of size and aspect ratio of faces. After analyzing a big collection of face images, we have got the range of size and aspect ratio of faces are given below:

$$0.85 \leq \text{Aspect Ratio} \leq 1.40$$

$$8 \text{ X } 12 \leq \text{Size} \leq 72 \text{ X } 84$$

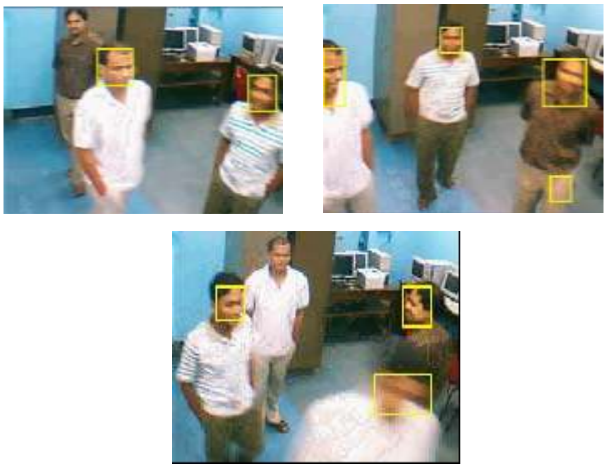


Figure -6: Some unsuccessful cases of our Real-time Face Detection System

3.4. Time complexity analysis

Speed is very crucial in any real-time system and it is the single most important criteria for such system. A real-time face detection system can't afford some extra computational time for the sake of better face detection. Our real-time face detection system is quite faster than most of such available systems. On a
2.66 GHz Pentium IV processor, the face detector of our real-time face detection system can process a 320 by 240 pixel image in about 0.078 seconds which is fast enough for real-time face detection application. This is roughly 8 times faster than the Rowley-Baluja-Kanade detector [8] and about 360 times faster than Scneiderman-Kanade detector [9].

4. CONCLUSION

We have presented a simple yet efficient approach for real-time face detection which minimizes computation time while achieving high accurate detection rate. This technique is much faster than available approaches. Unlike the available works, our approach does not search faces of different sizes iteratively but it detects faces of different sizes simultaneously and that makes our approach faster. Being faster our approach is applicable in any application where real-time face detection is essential. The range of such application is vast. The application may be used as a foot-fall analysis software for shop or it may be a security system to stop intruder. Again face detection is first step of face recognition. So, our technique can also be applied in a face recognition system where faces have to be recognized against complex background.

5. REFERENCE

[1] G. Kukharev, P. Masicz and P. Masicz "Modified gradient method for face localization", *Enhanced methods in computer security, biometric and artificial intelligence systems, pages 165-176, 2005.*

[2] R. Féraud, O. J. Bernier, J. E. Viallet and M. Collobert, "A Fast and Accurate Face Detector Based on Neural Networks", *IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no.1, pp. 42-53, January-2001.*

[3] C.R.Wren, A. Azarbayejani, T. Darrell, A.P. Pentland, "Pfinder: Real-Time Tracking of the Human Body", *IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 780-785, July-1997*

[4] R.P.Würtz, "Object Recognition Robust Under Translations, Deformations, and Changes in Background", *IEEE Trans. PAMI, vol. 19, no. 7, pp. 769-775, July-97.*

[5] P. Voila and M.J. Jones "Robust Real-time face detection", *International Journal of Computer Vision 57(2), 137-154, 2004.*

[6] http://www.ces.clemson.edu/~stb/blepo

[7] http://en.wikipedia.org/wiki/HSL_color_space

[8] H. Rowley, s. Baluja and Kanade, T. 1998. "Neural network-based face detection", *IEEE Patt. Anal. Mach. Intell., 20:22-38.*

[9] H. Schneiderman and Kanade, T. 2000. "A statistical method for 3D object detection applied for faces and cars", *International Conference on Computer Vision.*