8—7

Detecting Faces in Color Images using an Adaptive Color Model and Salient Features

Hang-Bong Kang[1]
Dept. of Computer Engineering
The Catholic University of Korea

Abstract

Face detection has many interesting applications such as a face recognition system, a surveillance system, and video/image indexing system. In this paper, we propose a new method of face detection using an adaptive skin color model and salient features. First, we detect skin color segments by adjusting threshold window in Hue-Saturation(HS) subspace based on the distribution of color histogram. From the detected skin color segments, salient feature points like eyes for face are extracted. After that, the possible eye feature points are compared with normalized eye features which are obtained from the training data. At this template matching stage, we use a modified Hausdorff distance. Finally, eye feature points are selected and then face segments are determined. Experimental results are presented.

1 Introduction

Face detection is an important tool in the biometric applications such as a face recognition system and a surveillance system. In detecting faces, gray scale images are usually used and face detection boils down to a pure pattern recognition task. Face template matching, image invariants, or low level features for the detection of facial features such as eyes, a nose and a mouth, are utilized [1, 2]. However, color has been suggested to be a powerful fundamental cue for face detection and fast implementations with high accuracy are also needed in modern applications [3].

Recently, an increasing body of research has addressed the problem of automatic face detection in color images. Various statistical color models are used in discriminating skin pixels and non-skin pixels. Color histograms [4, 5], single Gaussian model [6], and a Guassian mixture density model [3, 7, 8] were suggested. Jones and Rehg [4] proposed a comprehensive analysis of skin and non-skin color models and showed that histogram models were found to be slightly superior to Guassian mixture models in terms of skin color classification. In this approach, Bayesian detectors based on skin color histogram produced higher face detection results but their adaptation involved increased computational cost[3]. Tsapatsoulis et al. [3] proposed a model which combines an adaptive two-dimensional Gaussian color model and shape features with template matching. The adaptation is performed by re-estimating parameters, which are extracted from the current image for still images or from the previous images for video sequence.

There are still some limitations in face detection because the color of a face varies with changes in illuminant color, viewing geometry and miscellaneous sensor parameters. So, it is desirable to develop an adaptive algorithm to handle various situations. In this paper, we propose a new face detection approach that combines an adaptive color model and salient facial features with template matching. Section 2 describes skin color segment detection method. Section 3 explains salient feature extraction and Section 4 discusses the template matching. Section 5 presents experimental results with our proposed method.

2 Skin Color Segments Detection

Our proposed method consists of three modules such as a skin segment extraction module, an eye feature extraction module, and a template matching module. This is shown in Figure 1. The Eye feature extraction module consists of two parts. One part is based on gray scale information and the other one is based on color information. In this Section, we will describe an adaptive method to extract skin segments from images based on color histogram.

To detect skin segments from color images, we use a HSV color space. In HSV space, hue represents the color while the saturation is the purity of that color. So, a distribution in 2D HS space provides a color model with a degree of invariance to scene brightness. Even though the skin color subspace in HSV color space covers a small area of the HS space, it is very difficult to construct a skin color model to efficiently detect faces in all images. One possible solution is to set initial thresholds in HS space and adjusts the thresholds based on the distribution of color histogram. Figure 2(a) shows the initial threshold window in HS space in which the histogram is calculated on 360 x 100 bins. The size of initial window is determined from the Hue and Saturation values of training image data.

[1] Address: #43-1 Yokkok 2-dong Wonmi-Gu, Puchon, Kyonggi-Do, Korea. E-mail: hbkang@catholic.ac.kr

From the bins of the initial window, we compute the maximum value and ignore the bins having values that are less than 15% of the maximum value. Among the bins that have larger value than 15 % of the maximum value, we extract the connected regions. From the largest connected region, we compute the gravity center of that region. In this step, we assign different weights to each bin. The weights are determined depending on the value of bins. If the position of gravity center does not coincide with the center of the initial window, we slide the window such that the window's center coincides with the connected region's gravity center. If the amount of movement from the previous center to the new center does not exceed the predefined threshold value, we stop the movement. Finally, we control the size of the window. We adjust the range of Hue and Saturation values in which all histogram values larger than 15% of the maximum value in the window are included. A new threshold window is constructed at the new center. Figure 2(b) shows the final window of HS space. Skin segments are extracted using the final threshold window consisting of H and S values. The detected skin image has some holes in the face candidates and some noise in the background. To remove these, we execute morphological operations such as opening and closing with a 3X3 structuring element. At the filtered skin segments, the noises are reduced.

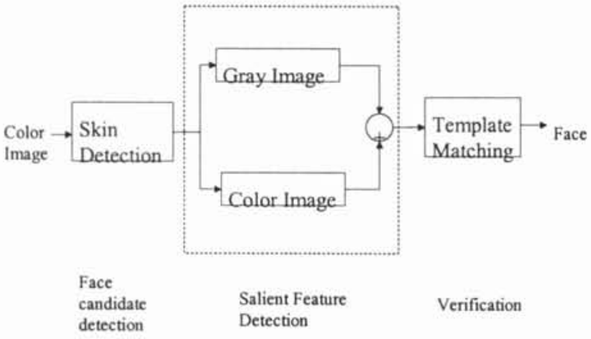

Figure 1: Face Detection Method

After detecting skin segments, we compute the shape features because extracted segments are sometimes unrelated to the human faces. Since the shape of the face is elliptical, it is desirable to compute the similarity of the shape of the extracted segment with an elliptical shape. The shape feature is computed from the bounding rectangle of each skin segment. We compute the ratio of the width (or short side) to the height (or long side) of the bounding rectangle. If the ratio is in the range between 0.3 and 0.9, we classify the segments as the candidates of face segments. The face candidates are passed to the eye feature extraction module.

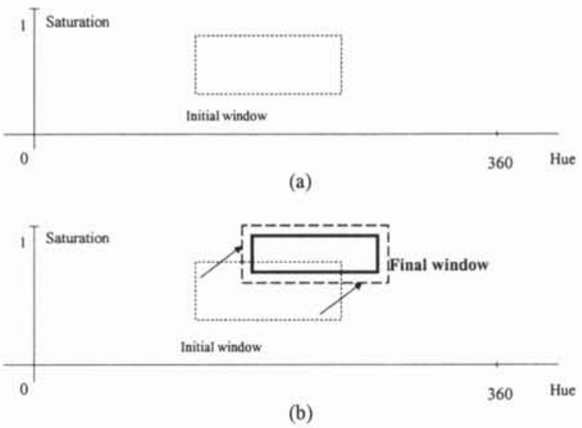

Figure 2: HS threshold window: (a) initial window (b) final window.

3 Salient Features Extraction

There are various facial features such as eyes, a nose, and a mouth. Among these features, we extract eye candidates as salient features because they are important in characterizing faces in different viewing geometry. To extract eye candidates, we use two approaches. One is based on gray values and the other is based on color information. To detect eye candidates from gray scale information, we transform the selected original face candidates into a gray scale image. From the gray scale image, we extract holes on the face candidates and then make hole areas into a binary image using a threshold value. Then, we select connected components which have enough size as salient features. From color information, we also extract holes on the initial face candidates and select connected components whose color are in the range of black. The features from gray image and color image are merged.

After that, we label the feature points. This is shown in Figure 3(a). From detected feature points, we make a ranking according to the possibility of their being eyes. To compute the possibility for being eye features, we divide the face candidate into four regions because eyes are located in the upper side of the face. First, we divide the window including a face into 3 to 2 ratio in height. Then, we compute the average of x coordinates of feature points in the window and then divide the window vertically using average value of x coordinates. This is shown in Figure 3(b). For each feature point in the upper left window, we execute dilation operation with a disk structuring element. Then, we compute the compactness value of dilated feature points. If the dilated feature point is near the circle, we choose the point as eye candidate. Finally, the feature points are sorted according to the compactness value and saved in the eye feature point list.

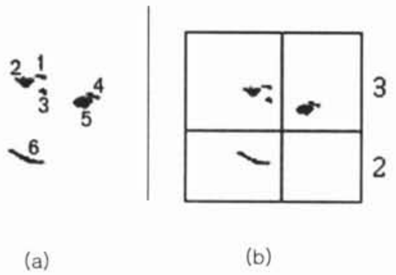

Figure 3: Eye feature candidates

4 Face Detection using Template Matching

From the eye feature list of face candidates, we verify whether the segment is face or not by template matching. For template matching, we construct normalized eye features from training data. Figure 4(a) shows the normalized eye features. To match extracted features from face candidates with templates, we first compute the gradient of the line by connecting two eye features extracted from eye feature list. If the gradient of the eye line is not equal to horizon, we transform the gradient of extracted eye line into horizontal. Then, we choose the eye region from the transformed images. After that, we compute edge information using Sobel edge operator in extracted eye region. We also compute the edge information of the template like Figure 4(b). To compute similarities between extracted eye features and a template, we measure the modified Hausdorff distance [10].

The modified Hausdorff distance is defined as follows,

$$H(A,B) = \max(h(A,B), h(B,A)) \ , \qquad (3)$$

where $h^{f}(A,B) = f^{th}_{a \in A} \min_{b \in B} \|a-b\|$

and $f^{th}_{x \in X} g(x)$ denotes the f-th quantile value of g(x) over the set X.

For a possible pair of eye features, we compute the modified Hausdorff distance from normalized eye features. We choose the eye feature candidates which have the minimum Hausdorff distance from normalized eye features. After finding eye candidates, we compute the face area using the angle between eye lines and a chin. The angle is obtained from the training image. After detecting eye and chins, we localize the face from the image data.

5 Experimental Results

We have tested our method on images which are photographs of people that can be obtained easily from the Internet. A total of 578 images were collected and tested. First, we detected face segments by adjusting threshold window in HS space. Figure 5(b) shows the detected skin segments from initial threshold HS window. The initial window size is determined from 100 training image data. Figure 5(c) shows the final detected skin segments from adjusted threshold HS window. Noises at the background are reduced. To remove some holes in the face candidates, we executed morphological opening and closing operations. Figure 5(d) shows the filtered face candidate.

Then, we extract eye features from the face candidates. Figure 6(b) shows the eye feature candidates from gray information and Figure 6(c) shows the eye feature candidates from color information. The extracted eye feature candidates are merged and then tested to detect real eye features.

We compute the gradient of eye lines and rotate the input image for template matching. Figure 7(a) shows input image and we rotate the image by 13.5 degree like Figure 7(b). Then, we cut the eye regions from the image for template matching. The size of eye regions is adjusted by expanding or shrinking based on the size of normalized eye region image like Figure 4(a). Sobel edge detection is performed and modified Hausdorff distance is computed for each pair of eye features. Finally, we select desirable eye points and compute the face area using the angle between eyes and a chin. Figure 8 and 9 shows detected faces. The accuracy of face detection result on our test images is 91 %.

Figure 4: Eye template: (a) normalized eye region image, (b) edge detection result.

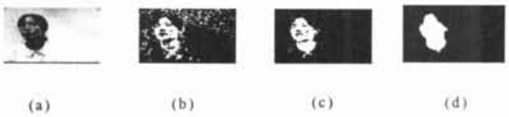

Figure 5: Detected Skin Segments: (a) image data, (b) skin segments detected from the initial threshold window, (c) skin segments detected from the final threshold window.

6 Conclusions

In this paper, we discuss a new approach for face detection using an adaptive skin color model and salient features. First, we extract skin color segments by adjusting the threshold window in HS subspace based on the distribution of color histogram. This provides a good color model with a degree of invariance to scene brightness. From extracted skin segments, we select eye candidates as salient features because they are important in characterizing faces in different viewing geometry. The template matching with eye features are performed by modified Hausdorff distance measure. By extracting eye features from face segments, our approach is useful in a face recognition system.

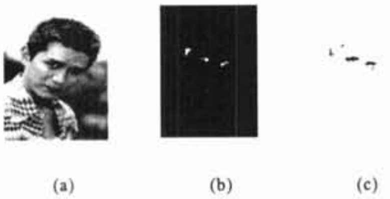

Figure 6: Eye Features: (a) image data, (b) detected eye features from gray information, (c) detected eye features from color information.

References

[1] M. Yang, D. Kreigman, and N. Ahuja, "Detecting Faces in Images: A Survey," *IEEE Trans. PAMI*, Vol. 24(1), pp. 34-58, 2002.

[2] K. Yow, C. Cipolla, "Feature-based human face detection in complex background," *Image and Vision Computing,* Vol. 15(9), pp.713-735, 1999.

[3] N. Tsapatsoulis, Y. Avrithis and S. Kollias, "Facial Image Indexing in Multimedia Databases," *Pattern Analysis & Applications,* 93-107, 2001.

[4] M. Jones and J. Rehg, "Statistical Color Model with Application to Skin Detection," *Proc. IEEE CVPR'99*, 1999.

[5] R. Kjeldsen and J. Kender, "Finding skin in color images," Proc. *2nd Int. Conf. Automatic Face and Gesture Recognition,* pp. 312-317, 1996.

[6] J. Yang, W. Lu, and A. Waibel, "Skin-color modeling and application," *Proc. ACCV'98*, pp. 687-694, 1998.

[7] T. Jebara and A. Pentalnd, " Parameterized structure from motion fro 3D adaptive feedback tracking of faces, " *Proc. CVPR,97*, pp. 144-150, 1997.

[8] S. McKenna, S. Gong, and Y. Raja, " Modeling facial color and identity with Gaussian mixtures," *Pattern Recognition,* Vol. 29(11), pp. 1883-1992, 1998.

[9] A. Hanjalic and H. Zhang, "An Integrated Scheme for Automated Video Abstraction Based on Unsupervised Cluster-Validity Analysis," *IEEE Trans. Cir. and Sys. for Video Tech.,* Vol. 9, No. 8, pp. 1280-1289, Dec. 1999.

[10] W. Rucklidge, "Locating Objects using the Hausdorff Distance," *Proc. ICCV'95*, pp. 457-464, 1995.

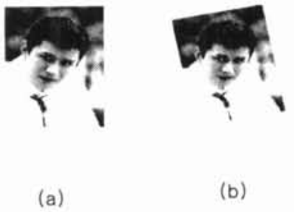

Figure 7: (a) input image, (b) rotated image

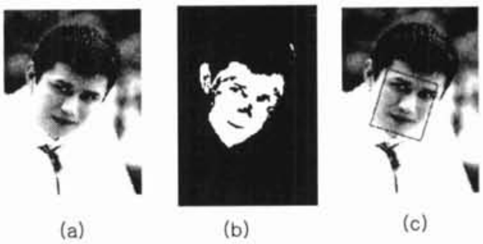

Figure 8: (a) input image, (b) skin segments, (c) detected face

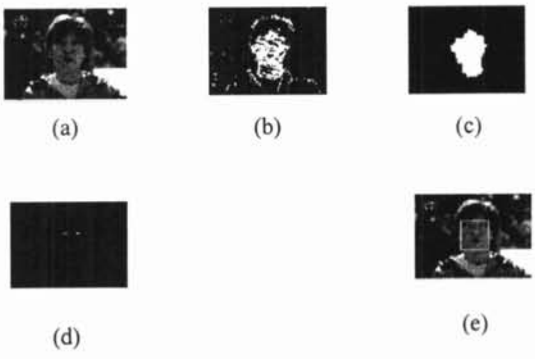

Figure 9: Experimental Results: (a) input image, (b) face segments, (c) filtered face segments, (d) eye features (e) detected face