피부 영역 분할과 신경 회로망에 기반한 갈라 영상에서 얼굴 검출

Face Detection in Color Images Based on Skin Region Segmentation and Neural Network

이연숙, 김영봉
부경대학교 전자컴퓨터정보통신공학부
Young-Sook Lee(yuliasia@pknu.ac.kr), Young-Bong Kim(ykim@pknu.ac.kr)

모약

많은 연구 다모종 프로그램들과 상업적 응용들이 얼굴 검출과 얼굴 인식 시스템들을 개발하기 위해 시도하고 있다. 인간의 얼굴 검출은 접근 제어 및 비디오 감시 시스템, 유인 컴퓨터 비디오, 신경인증 등과 같은 많은 응용 프로그램들과 중요한 역할을 한다. 일반적으로 스킨 영역 분할 후 피부와 연결된 얼굴, 스킨 투프로 인한 연결된 얼굴, 여러 개의 작은 부분으로 분할된 하나의 얼굴과 같은 얼굴과 특별한 문제들이다. 많은 얼굴 검출 기법들이 첫 번째 외 두 번째 문제를 해결하기도 하였으나, 그러나 이번 문제점에서 다른 측면의 원리로 인해 여러 영역의 분할된 하나의 얼굴이 검출되며 이것은 심각하다. 그러므로 우리는 기존 영역 분할 알고리즘을 이용할 수 없기 때문에 이 문제를 해결하기 위해 효율적인 수렴된 스킨 분할 알고리즘을 제안한다. 본 논문에서는 총 3장으로 총 25장의 내용을 설명한다. 각 장을 기준으로 각장에서 그래프와 그림 등의 도입을 통해 각 장의 얼굴이 여러 영역으로 분할되는 경우를 처리하기 위해 동작 영역간의 연결성을 활용하여 하나의 큰 영역으로 만든 랜덤 작업을 시도하였다. 틀로 크기의 얼굴 검출을 위해 다양한 기본 크기의 영역 모양 및 채택된 각 얼굴 근처 영역에 얼굴이 존재하는지를 판단하기 위해 영역과 알고리즘에 기반한 얼굴 검출 분류기를 사용하였다.

Abstract

Many research demonstrations and commercial applications have been tried to develop face detection and recognition systems. Human face detection plays an important role in applications such as access control and video surveillance, human computer interface, identity authentication, etc. There are some special problems such as a face connected with background faces connected via the skin color, and a face divided into several small parts after skin region segmentation in general. It can be allowed many face detection techniques to solve the first and second problems. However, it is not easy to detect a face divided into several parts of regions for reason of different illumination conditions in the third problem. Therefore, we propose an efficient modified skin segmentation algorithm to solve this problem because the typical region segmentation algorithm cannot be used.

Our algorithm detects skin regions over the entire image, and then generates face candidate regions using our skin segmentation algorithm. For each face candidate, we implement the procedure of region merging for divided regions in order to make a region using adjacency between homogeneous regions. We utilize various settings searching window sizes to detect different size faces and a face detection classifier based on a back-propagation algorithm in order to verify whether the searching window contains a face or not.

■ keyword : Face Detection, Skin Color Modeling, Region Merging

declare : #261123#003  
문재인 : 2009년 12월 18일
문재인 : 2009년 11월 23일
문재인 : 이연숙, e-mail : yuliasia@pknu.ac.kr
I. INTRODUCTION

Human faces have various inherent features and people have different appearances. Many research demonstrations and commercial applications have been tried to develop face detection and recognition systems. For example, they will work for surveillance systems, which have to recognize all employees of a company or a public institution, alert to any intruders.

Face detection is the first imperative step of human face recognition systems which identify a human face automatically using inherent facial feature information in computer vision, face tracking systems which track a human face, and facial expression recognition system that is to recognize human facial expressions of emotion. Face detection is to localize and extract human face in still images or digital video sequences. The development of a fast and exact face detection system plays an important role in the improvement of system performance.

Numerous methodologies have been proposed to locate or detect faces in gray-scale[1-3] or color images[4][5]. Gray-scale based methods utilize primitive feature values such as the luminance in detecting a face or faces on gray-scale images. The luminance is very sensitive to illumination. This method can result in not detecting a face which is least a part of important facial features because of the influence of the light.

Human skin color has been used and proven to be an effective feature in applications from face detection to hand tracking. Although different people have different skin colors, several studies have shown that the color difference lies largely between their intensity rather than their hue. Some color spaces that have been used in skin color detection include CIE's XYZ[6], HSV, HSI, YES, normalized RGB[7], RGB[8], YCrCb[9][10-11], and so forth.

Face detection systems on color images use different color ranges according to different races. There are four different races consisting of Caucasian, which refer to white American with light color skin, and it is also called the white race. We also have Mongoloids, which refer to an Asian race consisting of the Oriental and also called the yellow race. Another race is Negroid, which consists of colored people from Africa and which is also called the Black race. The last of the four human races is called Australoid which consists of people from Australia which is also called the Red race. We use skin colors of three races consisting of Caucasian, Asian, and Negroid.

Face classification method according to skin colors has been developed based on a fact that the identical human race is very similar skin colors and skin color pixels are localized in a small portion in chrominance space. Color is very sensitive to the light, thus there is a problem that we feel differently for the same color by illumination changes. Even though a color is within skin colors, it classifies the color into non-skin color. Therefore, we need to develop a robust and efficient face detection algorithm for extracting diverse skin colors.

Recently, many researchers have interested in the reduction of false alarm rates, high detection rates, and speedup of the system performance. Their systems also have focused on detecting human faces in still images and digital video sequences. Most of previous methods spend plenty of the running time on reducing false alarm rates and vice versa. It is very difficult to control trade-off between the reduction of false alarm rates and the speedup of the running time.

Therefore, we propose a frontal and near-frontal face detection algorithm in still color images using skin color segmentation algorithm and neural network. We tried to improve the system performance by speeding up the running time and reducing false alarm rates.
For each face candidate region, we utilize a variety of different searching window sizes for detecting faces in different sizes and a face detection classifier based on a back-propagation algorithm in order to verify whether the searching window contains a face or not.

The remainder of the paper is organized as follows: The description of our algorithm described in Section II. Experimental results in our algorithm are given in Section III. Conclusions and future work are discussed in Section IV.

II. FACE DETECTION

We describe an overview of our proposed face detection algorithm and then give details of the algorithm. We will demonstrate the performance and experimental results on several image databases.

An overview of our face detection algorithm is depicted in Fig. 1, which consists of two major modules: (1) skin region segmentation and region merging for finding face candidates; and (2) face detection for verifying detected face candidates.

![Fig. 1. Architecture of our face detection algorithm](image)

1. Conversion into YCbCr space

It is essential to choose an appropriate color space and a cluster associated with skin color in this space for extracting skin color regions in color images. Most existing approaches have been considered not luminosity component but chromaticity components to extract skin color pixels in color images. However, this paper employs not only chromaticity component in order to detect skin regions but also luminosity component in order to remove white noises and dark light on the images.

Color information is often effective in image segmentation. Since the skin-tone color depends on luminance, we continuously transform the YCbCr color space to make the skin cluster luma-independent. If the luminance value of a given image is not within defined ranges to remove dark lights or white noises, set the value to 0. For any color, its chrominance keeps almost constant under different lighting conditions. We consider the YCbCr color space as the interesting color space because it is robust against skin of various races and detection of dark and light skin ton colors. We convert the RGB color information to YCbCr color information:

\[
\begin{align*}
    r &= 0.2990R + 0.5870G + 0.1140B \\
    c_{\text{b}} &= -0.1697R - 0.3312G + 0.5000B \\
    c_{\text{r}} &= 0.5000R - 0.4187G - 0.0813B
\end{align*}
\]

where Y represents the luminance and Cb and Cr represent the chrominance components respectively.

2. Illumination control based the luminance component

Many previous works have been performed a preprocessing step such as illumination correction and histogram equalization. The histogram equalization is to assign a uniform histogram to the input image. This equalization can correct different camera gains and improve the contrast. Thus, it makes the images independent on the environments. Illumination correction subtracts a best-fit brightness plane from a
given resized window pixels, it can reduce the effect of heavy shadows caused by extreme lighting angles. However, we did not apply to illumination correction in our algorithm. We only add the histogram equalization to the preprocessing step of our algorithm. If we apply to illumination correction, our algorithm will improve the performance more. Initially, an original image goes through the preprocessing step using histogram equalization.

The input image contains a variety of noises under different illumination conditions. Detection to skin color is intimately associated with white noises among them. That is to say, illumination with high luminance has an effect on detecting skin color pixels. To remove these kinds of white noises, we utilize a transformed luminance component of a color image and remove some regions where can be contained white noises on the color image.

The luminance value of white noise is over a constant threshold in general. In case of dark illumination, this problem that cannot be detected skin color pixels or can be extracted incorrect color pixels.

As above-mentioned reasons, we utilize the luminance component to process high luminance or dark lighting. A binary image can be computed by

\[ r_{i,j} = \begin{cases} 1 & \text{if } T_H \geq 1.7 \\ 0 & \text{otherwise} \end{cases} \]

\[ P_{i,j} = 1 - r_{i,j} \]

where \( i \) is in and \( j \) coordinates on the image. The threshold value was chosen empirically.

If the luminance value of \( r_{i,j} \) is within defined ranges \( TH_{low} \) and \( TH_{high} \), set the value of \( r_{i,j} \) on a binary image to 1 (black), otherwise 0 (white).

3. Selection of Human Skin Color Model

The most important goal of skin color detection is to build a decision rule, which will discriminate between skin and non-skin pixels. There are many approaches such as explicitly defined skin region, LUT (Normalized Lookup Tab.), Bayes classifier, SOM (Self-Organizing Map), Gaussian distribution and so on.

Explicitly defined skin region method to build a skin classifier is to define explicitly through a number of rules the boundaries skin cluster in some color space. The simplicity of this method has attracted and still does many researchers. The obvious advantage of this method is simplicity of skin detection rules that leads to construction of a very rapid classifier. It is difficult to achieve high recognition rates with this method and needs to find both good color space and adequate decision rules empirically.[10][11] Chai and Ngan[10] proposed a simple and fast method using a fixed-range skin color map that is derived and used on the chrominance component of the input image to detect pixels with skin color appearance. It is basically classified pixels as skin color if its chrominance values are within defined ranges.

The human skin color model used in this paper is based on the explicitly defined skin region method. Skin colors from various races of the world are collected from the internet in the different size of skin samples pixels. We cropped skin regions for each individual from each image by hand in [Fig. 2]. There are a total of 35,668 skin samples from 31 color images to determine experimentally the color distribution of human skin in chromatic color space. Skin colors were grouped into three classes: Caucasian, Asian, and Negroid. The skin pixels in Cb-Cr space are confined in a small region as shown in [Fig. 3]. We employ the ranges of skin pixels obtained skin region distribution under experiment.
Fig. 2: Skin samples for different races such as Caucasian (the white race), Asian (the yellow race), and Negroid (the black race).

Fig. 3: Histograms of Cb and Cr values of different skin colors:
(a) the white race (b) the yellow race (c) the black race (d) the different races containing the white, the yellow, and the black race.

To model skin color, one has to look for color spaces in which the distribution of color components is concentrated in a small area. Researchers have searched for color spaces. It is a fact that skin pixels are localized in a small portion in Cb and Cr space for each different facial skin color as shown in [Fig. 3].

To extract skin pixels for each race, we suggested ranges for this rectangular model are found from one-dimensional histograms for Cb and Cr. The decision rule is as follows:

\[
\text{decision rule} = \begin{cases} 
1 & \text{if } C_b > TH_{\text{pe}} \text{ and } C_r > TH_{\text{pe}} \text{ (white race)} \\
0 & \text{otherwise} 
\end{cases}
\]

where \( C_b \) and \( C_r \) are the value whether a skin pixel or not, \( TH_{\text{pe}} \) and \( TH_{\text{pe}} \) are thresholds for extracting Cb and Cr values on color spaces. \( P_w \) and \( P_y \) represent Cb and Cr values on original image, respectively.

For each race, skin pixels computed as following:

\[
I_{\text{race}}(i,j) = \begin{cases} 
1 & \text{if } (P_w(i,j) \leq TH_{\text{pe}}) \text{ and } \left( P_y(i,j) \leq TH_{\text{pe}} \right) \\
0 & \text{otherwise} 
\end{cases}
\]

For each pixel, the final decision of skin pixel will be determined by following decision rule using extracted skin pixels for each race. Therefore, possible skin pixels \( I_{\text{race}}(i,j) \) can be determined as follows:

\[
I_{\text{final}}(i,j) = I_{\text{race}}(i,j) \cap I_{\text{white}}(i,j) \cap I_{\text{cancer}}(i,j) 
\]

where \( I_{\text{white}}(i,j) \) and \( I_{\text{cancer}}(i,j) \) represent a possible skin pixel which is computed in equation (6) for Caucasian, Asian, and Negroid, respectively.

4. Skin color segmentation algorithm

In order to segment skin color regions, it classified a skin pixel into a non-skin pixel using skin color model. This procedure used pixel values based on YCbCr color space. Connected components labeling scans an image and groups its pixels into components based on pixel connectivity, i.e., all pixels in a connected component share similar pixel intensity values and are in some way connected with each other. Once all groups have been determined, each pixel is labeled with a gray scale or a color (color labeling) according to the component it was assigned to.

Extraction and labeling of various disjoint and connected components in an image is crucial to many automated image analysis applications. We use and modify a "grassfire" connected component labeling algorithm in general [11]. After labeling algorithm, there
are some special cases such as a face connected with background, faces connected via the skin color, and a face divided into several small parts. In case of a face connected with background or faces connected via the skin color, it can detect faces by a face detector. The third case has a problem that can not detect faces by the face detector. Thus, we need a new method to solve this problem because the typical region segmentation algorithm can not be used to. Therefore, we propose an efficient modified skin segmentation labeling algorithm. If a face divides into several small parts, it can merge several parts into a big region using the modified skin region labeling algorithm.

Proposed skin region segmentation algorithm

Step 1: Start at location (0,0) on the input image. Scan the image pixel by pixel moving across from left to right until search for labeling the first skin color pixel which is not labelled and is within skin ranges at location x, y.
Step 2: Label all the 4-connected components of the input image and assign labels to pixels in the image.
Step 3: Compute the area for each labelled connected region.
Step 4: If all pixels are labelled, go to 5; otherwise, go to step 4.
Step 5: If the area of Cb and Cr values in each labelled region is less than a constant threshold (150), remove this region containing non-face. The threshold was chosen empirically.

\[ S_{Cb} = \frac{C - B}{2} \text{ and } S_{Cr} = \frac{C - R}{2} \]  \tag{8}

\[ S_{Cr} = \frac{C - R}{2} \text{ and } S_{Cb} \geq TH. \]  \tag{9}

Step 6: Although a face is segmented one part, the face can be divided into several small parts. In that case, these parts are located within 1 or 2 pixels. We assume that homogenous regions are very close to. Since it is difficult to compute distance between pixels within 1 or 2 pixels, we utilize to merge these regions to use dilate/erosion operator. We can employ the procedure of region merging, and so we do not need to compute distance between pixels.

Step 7: After merging region, we need to apply labeling algorithm again.

As shown in [Fig. 1][Fig. 5], we shows results for extracting face candidate regions using the proposed modified skin region segmentation algorithm. [Fig. 4(b)][Fig. 5(d)] show blue color bounding boxes which are segmented skin regions. The segmented skin region which is the area of Cb and Cr values in each region is over a constant threshold.

A left arm of a woman on the image in [Fig. 4(c)] and neck of a man or a part of a left arm of a woman in [Fig. 5(c)] are removed because these regions are less than the threshold of summation of the area for Cb and Cr and rejected from face candidate. These are not marked blue boxes in these rejected regions. [Fig. 4(d)] shows that the woman’s face with her hair is selected a skin region because her hair colors are similar to skin color pixels. However, at the next step, the face detector based neural network can reject this skin candidate region as a non-face. We will not need to a special procedure for rejecting this region.

![Fig. 4: Skin color segmentation](image)

(a) original image  (b) skin color map
(c) skin color classification  (d) face region candidate
5. Bounding Box of face candidate region

As shown in [Fig. 4(c)][Fig. 5(c)], we can see the results for proposed skin color segmentation algorithm and some parts of facial features such as eyes or eyebrows and the nose are removed due to varying light conditions. If it applies each face candidate in [Fig. 4(e)][Fig. 5(e)] as an input to the face detector based on neural network in the verification step, it can be possible that the result can be rejected the face candidate containing a face.

In order to fill holes for face candidate regions, we build a rectangle box as an input to the face detector. This rectangle box has pixel values on the original image. A starting point and a ending point of the bounding box are the minimum of and the maximum of x-coord position and y-coord position, respectively. The rectangles drawn the blue color indicates bounding boxes for skin regions. These rectangle regions as the input to the verification step are used.

6. Searching window size and searching interval

In general, most of face detection systems are applied at each pixel location and utilize a variety of searching window sizes because the size of a face in an image can vary according to the distance of the person to the camera and faces of various sizes can also be present. In order to detect faces in various sizes on the given input image, we have implemented a pyramid approach that scans for faces at various image resolutions. However, if face detection systems search for potential face candidates on the given image at each pixel position, it can result in slowing down the performance of these systems. Thus, we did not search for potential face candidates on the given image at all each pixel position. Proposed skin region segmentation algorithm can remove non-skin regions and face face candidate regions which contain potential faces are only input to neural network classifier on image pyramids.
In order to detect face region candidates, we implemented the skin region segmentation algorithm as mentioned above in Section 2.2 and then each face candidate went through the basic face detector algorithm based on neural network verification stage. Image pyramids from an image is formed to detect faces of different size. Scaling by 1.1 to 1.2 at each level yields a pyramid of adequate scaling granularity. We chose that each level of the pyramid is given by applying a scale factor of 1.2(14). We utilize that pyramidal images created from a result image contain face candidate regions after the procedure of skin region segmentation. We use variety of searching window sizes ranging from 20 by 20 pixels to about the size of each detected face candidate. The resulting image realizes each searching window to 20×20 pixels. This resized image is considered as input to a neural network detector at the verification stage.

At each level of image pyramid, the resized image within each face candidate is first passed through the neural network detector trained to detect a whole face. Using these face candidate regions, we can eliminate lots of windows that do not potentially contain a face. Thus, the speedup of our algorithm is greatly depended on the efficient generation of the face candidates on the result image at the skin color region segmentation algorithm stage. Most faces are detected at multiple nearby positions or scales, while false detections often occur with less consistency. If we search for a certain searching interval, it can eliminate many false detections and improve the system performance. To search for the searching window interval, we investigated 12 test images containing a frontal- or near-frontal face. The input test images went through face detector based neural network. We chose one of detected face regions. For each location and scale at which a face is detected, we investigate the detection results at 20 by 20 pixels positions. As a result, the interval in pixel between each searching window is 4 or 5 pixels in order not to miss a face.

7. Face detection classifier based on Neural Network

We have a whole face detection classifier using a typical three-layer back-propagation network. Back propagation has been the most widely used of the neural networks paradigm and has been successfully utilized in applications such as image classification, facial target recognition and pattern analysis problems [3].

Back propagation of the system is layered with each layer fully connected to the above layer and the below layer. Initially, the weights are set to small-randomized values. The output of this detector is a real number.

The face detector is to search for a whole face in the input image. We defined the detector as a whole face detector. We used a training set of 300 face images that includes 100 positive examples and 200 negative examples as shown in Fig. 7, each 20×20 pixel window in size. This whole face detector using back propagation network is trained to classify 20×20 intensity images as face or non-face and produced real values between 1 and 0, indicating whether or not a window contains a face, respectively.

![Fig. 7. Training sets](image)

(a) face examples  (b) non-face examples
III. EXPERIMENTAL RESULTS

We have tested our system on two face image databases.

There is no standard face detection database in color images. The available databases which are commonly used by face recognition for FR3T face databases or face detection for CMU database and M/T databases are grayscale image databases. There are not suitable for evaluating face detection algorithm for color images. Therefore, we have collected our databases composed of 108 still color images with 240 faces. The images have various different sizes, with complex background and indoor/outdoor and are collected from photographs, some test pictures from existing methods, the World Wide Web, and digitized television pictures. The images contained various races consisting of Caucasian, Asian, and Negroid. We present detection results in Table 1 and Table 2 for our test sets.

We have tested our system with the Champion database set, which consists of 227 images containing 227 faces. Table 3 shows the face detection results of our system and method in Hsu[1]. The detection rate (DF) and the number of false positive obtained in Hsu[1] are 91.63 percent and 14 respectively. The detection rate in our system is higher than Hsu[1]. However, false detection rate is high with 15.25 percent.

| Table 2: Detection rate and # Detect for our system on multiple faces in an image |
|---------------------------------|--------|---------|---------|---------|---------|
| age                            | race   | Our system without region merging | Our system with region merging | # Detect | FR%    |
| Two or more people             | 80     | 79      | 97.7    | 64      | 93.3    |

| Table 3: Comparison of Detection rate on the Champion database |
|----------------|--------|---------|--------|
| race           | Hsu[1] | Our system |
| Black          | 0.227  | 0.227   |
| Dark           | 0.183  | 0.183   |
| Fair           | 0.275  | 0.275   |
| Tinted         | 0.123  | 0.123   |

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented an efficient frontal and near-frontal face detection algorithm for color images based skin region segmentation and neural network. We utilize the YCbCr color space to model skin color in order to detect the different size of faces because the color space is robust to extract skin colors for various races. Existing works did not use the luminance because it is very sensitive to the illumination. However, our algorithm employed a modified skin color segmentation algorithm to remove some white noises caused by illumination and dark light affects.

The proposed skin segmentation algorithm is added into the region merging procedure using distance between regions when a face is divided into several parts of regions by too brighter or darker illumination and showed the good results.

For each face candidate regions, we utilize different sizes of a searching window and a face detection classifier based on a back-propagation algorithm in order to verify whether this searching window contains a face or not. In order to speed up the performance of
our system, we used limitations of size and the interval of the searching window under experimental results.

Although we tested test sets including near-frontal, rotated faces, diverse races with complex background and various illumination conditions, we obtain high detection rates between 91.1% and 99% of faces for our collected test sets and 91.65% for the Champion database. Fig. 8 shows result images from our algorithm with region merging and detect multiple faces of different sizes with a wide variety of facial variations and races. Fig. 9 shows some false detection results. After changing the interval of searching window, some images of false detection can detect faces. We need to develop more adaptive intervals of the searching window.

As future work, one of our goals is to detect multi-view faces detection using our algorithm. The performance can be improved by applying proposed algorithm in order to reduce tremendous non-face candidates in test images with very light effects and complex backgrounds.

References


이 영숙 (Young-Sook Lee)
- She received her BS degree in computer engineering from the University of Dongsan, Busan in 1985 and received the MS degree in computer science from Pukyong National University in 1990. She is currently a PhD student at Pukyong National University in computer science department.
- Her research interests include computer vision, machine learning, and computer animation.

김 영성 (Young-Song Kim)
- He received his BS degree in computer science and statistics from Seoul National University, Seoul, in 1987. He received the MS and PhD degrees in computer science from KAIST in 1989 and 1994, respectively. He is currently a full professor in the department of electronics, computer and telecommunication engineering at Pukyong National University.
- His current research interests are game programming, computer animation, and 3D object.