PERGAMON

Pattern Recognition 34 (2001) 1067–1073

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Adaptive skin-color filter

Kyung-Min Cho, Jeong-Hun Jang, Ki-Sang Hong*

Department of Electrical Engineering, POSTECH, San 31 Hyoja-dong, Nam-ku, Pohang, Kyungbuk 790-784, South Korea

Received 30 March 1999; received in revised form 14 February 2000; accepted 14 February 2000

Abstract

In this paper, we propose a new method, called an *adaptive skin color filter*, for detecting skin color regions in a color image. The use of skin color provides an efficient way to find candidate regions for faces or hands in color images. However, it is not easy to find skin color regions because the color of skin regions varies from image to image due to a variety of reasons. Since most of the previous methods adopt a fixed threshold scheme, they are useful only in the restricted (i.e., controlled) environment. However, our method is applicable to images in more general situations since it is capable of adaptively adjusting its threshold values and effectively separating skin color regions from similar background color regions. Our method consists of basically two stages. In the first stage, a thresholding box in HSV color space is updated adaptively using a color histogram under the assumption that the area of skin color regions is comparable to that of similar background color ones. In the second stage, color vectors inside the thresholding box are classified into two groups: skin color vectors and background color vectors. Our method and other conventional methods were tested with 379 images obtained from the Internet. Experimental results show that our method is robust to the variations of skin regions' color compared to the conventional methods.

Keywords: Skin-color region detection; Face detection; Adaptive thresholding; Cluster analysis

1. Introduction

One of the research topics in color image analysis is to find regions of specific color in a given color image. Such a technique is very useful, for example, to find a face in a human image, lane-markers in a road image, the ground in a soccer game image, etc. In this paper, we focus on developing a method for finding *skin color* regions because of its increasing usefulness in detecting and tracking faces or hands in color images. The use of color to find target objects in color images has proven to be robust to the variations of objects in scale or orientation, partial occlusion by other objects, etc. In general, however, it is not an easy task to extract regions of specific color from a given color image, since the color of an object varies with changes in illuminant color, illumination geometry (i.e., angle of incidence), viewing geometry (i.e., angle of reflectance), and miscellaneous sensor parameters. The method of extracting specific color regions using fixed threshold values in color space often fails except for simple cases. Therefore, an algorithm which copes with the variations of a target object's color must be developed.

There are several published papers related to this problem. Buluswar and Draper [1] proposed a method where a multivariate decision tree (MDT) is used to determine whether the color of each pixel matches with the specified color or not, but this method requires the time-consuming processes of gathering samples and training the MDT. Sobottka and Pitas [2] transformed color vectors in RGB space into ones in HSV space and used only the hue and saturation components of them to find candidate face regions. However, their method lacks adaptation capability since the upper and lower threshold values for each component are fixed. The method of Fleck et al. [3], which is similar to that of Sobottka

* Corresponding author. Tel.: + 82-562-279-2216; fax: + 82-562-279-5586.
E-mail address: hongks@postech.ac.kr (K-S. Hong).

0031-3203/01/$20.00
PII: S0031-3203(00)00034-0

and Pitas, also has the same limitation. Yang and Waibel [4] showed that skin color distributions of different people under different lighting conditions have similar Gaussian distributions in normalized RG space. They estimated the parameters of a Gaussian distribution of normal skin color from sample face images and classified each pixel of a test image using the estimated distribution. But they did not consider the variation of the skin color distribution in the test image. Recently, Yang et al. [5] and McKenna et al. [6] proposed adaptive color distribution models for robust object tracking. Unfortunately, their methods are valid only when dealing with image sequences, and not applicable to the case of static images.

In this paper, we propose a new skin color filter that is capable of adaptively adjusting its threshold values and effectively separating skin color regions from seemingly similar, but different color regions. A brief description on our method is given as follows. We first choose rough upper and lower threshold values for each color component (hue, saturation, and value) by observing the skin color distributions of several sample images. These threshold values define a 3D box, called the *thresholding box*, in HSV color space. The color vectors inside the thresholding box play the role of standard color vectors representing normal skin color. When a new image is tested, in order to get robustness against illumination changes, the threshold values of the S and V components are updated adaptively (whereas those of the H component are fixed because we assume that the user-defined threshold values for hue constitute absolute boundaries of a skin color region in color space) based on a color histogram built in SV space. Then, a new thresholding box is formed with the updated threshold values. The assumption behind this adaptation procedure is that the area of skin color regions is comparable to (or larger than) that of the background regions whose color is similar to skin color. In other words, we assume that skin color vectors constitute a dominant cluster in HSV space within the range predefined by the upper and lower thresholds of a hue component. This assumption is valid in most applications. The adjustment of threshold values is, in fact, to search for the dominant cluster.

In most cases, color vectors of the pixels belonging to skin color regions are likely to be included in the new thresholding box. But this box is not enough to separate skin color regions from background regions of slightly different colors. We will use the term *background colors* to indicate similar but different colors which need to be distinguished from the true skin color. The background color vectors tend to coexist with the skin color vectors in the thresholding box. Cluster analysis is performed to determine if dominant background color vectors exist in the box. If they do exist, they are separated from the skin color vectors by a linear classifier. The linear classifier divides the thresholding box into two parts (i.e., classes) with a plane. Finally, the vectors that belong to the class whose mean vector is closer to the mean of the skin color vectors obtained in advance from a variety of sample images are chosen as skin color vectors.

This paper is organized as follows. In Section 2, a description of how to adaptively update the threshold values for each color component is given. Separating skin color vectors from background color vectors is considered in Section 3. In Section 4, experimental results are shown. Conclusions are given in Section 5.

2. Determining threshold values for each color component

2.1. Setting initial threshold values

In general, the color of each pixel in a color digital image is represented in terms of a combination of R, G, and B values. But treating color quantities in RGB space is not preferred because R, G, and B values are coupled together so that varying illumination affects all the R, G, and B values of pixels. In this paper, all of our work is carried out in HSV (hue, saturation, and value) space instead of in RGB space [7,8]. The HSV representation of color is known to be more compatible with human perception. In other words, the role of each component in HSV space is more understandable by human beings. Therefore, color analysis can be performed more easily in HSV space.

Initially, upper and lower threshold values for each color component are chosen manually by observing the skin color distributions of several distinctive sample images that were obtained under various illumination conditions. These threshold values define a 3D box, called the *thresholding box*, in HSV space. The color vectors inside the thresholding box play the role of standard color vectors representing normal skin color. Note that this box should be large enough to cover most skin color vectors appearing in the sample images. Initial threshold values used in this paper are shown in Table 1, where we shifted hue values by 0.5 (i.e., 180°) to make the center of the hue axis indicate pure red. Because most of the sample images used in our experiments were photographs of yellow people, the threshold values of Table 1 were chosen to reflect the characteristic of skin color of yellow people. If, for example, we need to find skin

Table 1
Initial upper and lower threshold values for each color component

Threshold	Hue	Saturation	Value
Upper	0.7	0.75	0.95
Lower	0.4	0.15	0.35

regions of white people, the threshold values should be adjusted to cover the range of skin color of white people.

2.2. Updating the threshold values

Since the threshold values in Table 1 have been roughly selected from several sample images, we cannot be assured that they are well suited for most images containing skin color. Therefore, they need to be adjusted to fit into the new image to be tested. In our method, however, thresholds for hue are fixed to the values in Table 1, because we assume that these threshold values constitute absolute boundaries for skin color. That is to say, color vectors whose hue values are outside the range bounded by the threshold values are not considered to represent the skin color we want to find. This assumption is necessary because there is no clear definition of skin color. A user needs to define an absolute range of skin color in color space. In our method, restrictions are imposed only on the hue values of skin color vectors.

The threshold values for S and V components are updated iteratively based on a color histogram built in SV space. A detailed procedure for the update of the threshold values is given as follows:

1. Construct a color histogram in SV space with color vectors whose hue values are within the range bounded by the threshold values in Table 1. In our experiments, the histogram was made on 100×100 cells.
2. Construct a *thresholding rectangle* defined by the upper and lower threshold values of the saturation and value components in Table 1.
3. Let m denote a maximum value of the histogram inside the thresholding rectangle. Calculate the center of gravity of the histogram inside the rectangle for the cells whose values are greater than $0.1\,m$.
4. Move the thresholding rectangle so that its center is placed on the center of gravity calculated in Step 3. The size of the rectangle does not change.
5. Repeat Steps 3 and 4 until the distance between the previous and current thresholding rectangles is less than some threshold value.
6. Final threshold values are determined by the four sides of the new thresholding rectangle that surrounds the region of the histogram whose cell values are greater than $0.1\,m$. Therefore, the new thresholding rectangle becomes smaller (i.e., tighter) than the initial one. The new thresholding rectangle together with the fixed threshold values for hue constitutes a new thresholding box in HSV space.

One of the results obtained with the above method is shown in Fig. 1. An input image is given in Fig. 1(a) and its skin color regions extracted using the initial thresholds in Table 1 are shown in Fig. 1(b). Since the brightness of the face region in Fig. 1(a) is lower than that of the normal skin color, only small portions of the region were detected. Fig. 1(c) shows the thresholds update result and Fig. 1(d) shows the new skin color regions extracted with the updated threshold values. According to our experiments, Steps 3 and 4 of the above procedure were repeated almost once on average for the test images used. One can see that the face region has been found almost correctly. It should be noted that what the above method really does is to look for the *dominant color* of an input image that is in the range of hue given in Table 1. By dominant color, we mean that the region of that color is larger than those of other colors. In the case of Fig. 1(a), the color of the face is dominant because the face region occupies most of the image's area. If an image is given whose background color is much more dominant than the color of the true skin regions and the difference between the two colors is relatively large, our method will fail.

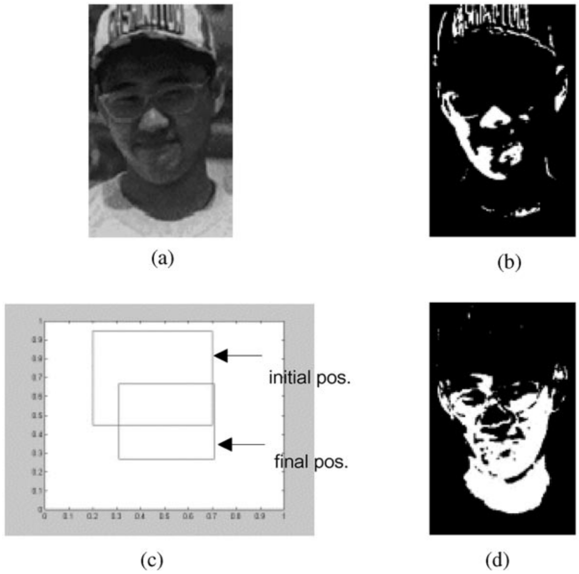

Fig. 1. Adaptation of threshold values: (a) input image; (b) skin color regions extracted using the initial threshold values in Table 1; (c) adjustment of threshold values; (d) skin color regions extracted using the updated threshold values.

3. Separating skin color vectors from background color vectors

If an input image contains a background whose color is similar to (but distinguishable from) the color of a true skin region (e.g., a picture of a person wearing skin colored clothes like Fig. 2(a)), the method in Section 2.2 will yield a result where the updated thresholding box

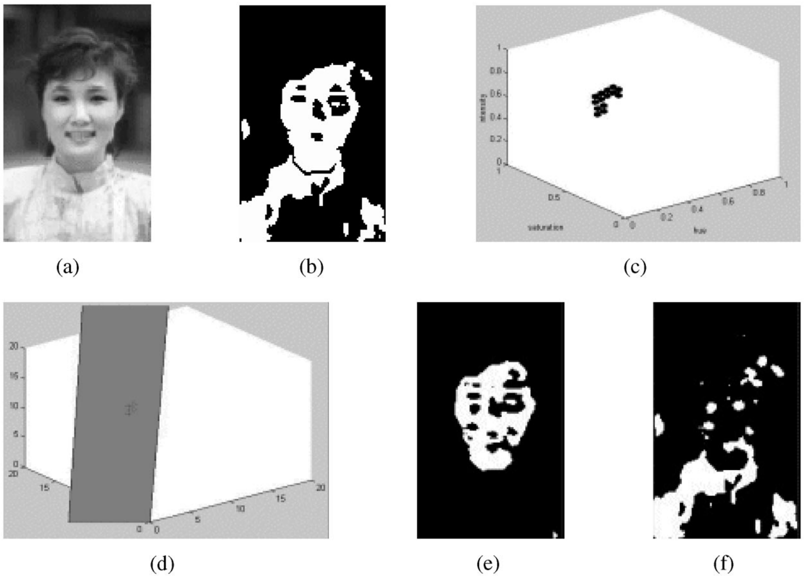

Fig. 2. Separation of skin color vectors from background color vectors: (a) input image; (b) skin color regions extracted using the updated threshold values; (c) 3D histogram of the pixels belonging to the regions of (b); (d) separating plane; (e) resulting skin color regions found; (f) background color regions.

contains both the background color and skin color vectors. If the size of a background color region is comparable to that of a skin color region, two dominant clusters of color vectors will be formed in color space. In such a case, the background color vectors cannot be simply ignored and need to be separated from the skin color vectors.

3.1. Cluster analysis

To find out if there exist two dominant groups of color vectors in the thresholding box, cluster analysis is carried out as follows:

1. Construct a color histogram in HSV space with the color vectors belonging to the thresholding box. In our experiments, the histogram was made on $100 \times 100 \times 100$ cells.
2. Perform 3D CRE (connected region extraction) for the cells whose values are greater than T_l (T_l is usually small). Then, each connected region is assigned a unique label, and subsequently the cells and color vectors belonging to the same region have the same label as that of the region.
3. End this procedure if there exists only one label (i.e., all the cells are connected). In this case, the procedures in Sections 3.2 and 3.3 are skipped, and all the color vectors in the thresholding box are considered to represent skin color.
4. Calculate for each pair of connected regions a distance between their centers if there exist more than two labels.
5. Merge differently labeled regions until only two labels remain. Closer regions are merged first.
6. Count the labels for the cells whose values are greater than T_u ($T_u \gg T_l$). If there is only one label, the procedures in Sections 3.2 and 3.3 are skipped. In this case, no dominant background color is supposed to exist.

3.2. Computing a separating-plane

Let us suppose that by the cluster analysis in Section 3.1, it has been revealed that two dominant clusters of color vectors exist in the thresholding box. This means that the color vectors in the thresholding box can be classified into two classes. Note that the color vectors belonging to the cells whose values are greater than T_l have already been classified owing to the procedure in Section 3.1. What we have to do now is to classify other color vectors that are still undetermined.

We use a linear classifier for this classification. It divides the thresholding box into two parts (each part corresponding to a different class) with a separating plane in HSV space. Therefore, training the linear classifier is equal to computing a separating plane. The linear classifier is trained with the color vectors labeled in Section 3.1 and the training can be done easily by the *perceptron learning* method introduced in Ref. [9]. The perceptron learning is to find appropriate weights of a linear classifier, given training samples, by minimizing some energy function with the gradient-descent optimization technique.

An example for a real image is shown in Fig. 2. An input image and its skin color regions extracted with the updated threshold values are shown in Figs. 2(a) and (b), respectively. Note that since the color of the woman's clothes is similar to the normal skin color, some parts of the clothes were detected as well as the face region. A 3D histogram constructed with the color vectors belonging to the updated thresholding box is shown in Fig. 2(c). In this figure, only cells whose values are above T_l are displayed as black cubes. One can see two distinct clusters in the histogram. These clusters are divided by a separating plane as shown in Fig. 2(d). Fig. 2(e) or (f) shows the regions of the image corresponding to one of the two sides of the separating plane. As can be seen in Fig. 2(e), the face region has been extracted more accurately. Note that actually we do not know yet which side of the plane corresponds to the skin color class. It will be decided in the next section.

3.3. Selecting the part corresponding to the skin color class

As mentioned in the previous subsection, the plane separates the thresholding box into two parts, but it has not been decided yet which part corresponds to the skin color class. To do this, we could use a geometric property of a target object to select the part corresponding to the skin color class. For example, look at the two binary images in Figs. 2(e) and (f). These images were constructed by examining which side of the plane the color vector of each pixel in Fig. 2(a) belongs to. Assuming that our

Table 2
Skin-color region detection results of the proposed method

Condition	Input (#)	Success (#)	Failure (#)	Success rate (%)
(a) Backgrounds of the test images do not include colors similar to skin color				
Normal	171	160	11	93.6
Reddish	3	3	0	100.0
Bright	29	27	2	93.1
Dark	8	8	0	100.0
Total	211	198	13	93.8
(b) Backgrounds of the test images contain colors similar to skin color				
Normal	146	127	19	87.0
Reddish	1	1	0	100.0
Bright	15	13	2	86.7
Dark	6	5	1	83.3
Total	168	146	22	86.9

Table 3
Average execution time under the Matlab environment

	Proc. of Section 2.2	Proc. of Section 3.1	Total
Execution time	13.74 s	15.25 s	41.68 s

(a) (b) (c) (d)

Fig. 3. Examples of the skin color region detection results: (a) input images; (b) our method; (c) Fleck et al.'s method [3]; (d) Yang and Waibel's method [4].

target object is a face, it is possible to choose (e) as representing the face region by measuring how close the shape of the region is to an oval [2]. However, this technique seems not to work well if the face is partly obscured by hands or other objects.

In our method, shape information is not used at all. Instead, we use the mean of skin color vectors obtained from many sample images. The side of the separating plane that has a mean vector closer to the mean of the sample skin color vectors is selected as corresponding to the skin color class. Two hundred sample images were used in our experiments and sample skin color vectors were extracted automatically using the threshold values in Table 1. In the example of Fig. 2, case (e) has been selected correctly with this method.

4. Experimental results

Test images used to measure the performance of our method are photographs of people that can be obtained easily from the Internet. Since our method is intended to detect regions of only one skin color, we collected most of the test images from one race, i.e., yellow people in this experiment. Since the images were taken and digitized under various conditions, it can be said that no special illumination or other constraints are imposed on our test images. A total of 379 images were collected and categorized manually. We first divided them into two groups according to whether the background of each image contains colors similar to skin color or not, and then we subdivided each group according to the conditions of the skin regions' color. After categorizing all the test images, we applied our algorithm to them in batch and obtained the results in Tables 2(a) and (b), where "Success" means that at least half of the true skin color regions were found in a given image so that the result does not raise a serious problem in performing the next step. Of the failure images in Table 2, seven images were unsuccessful due to the failure of the selection method of Section 3.3 in the case of (a), and in the case of (b), eight images were unsatisfactory due to the same reason. From the results in Tables 2(a) and (b), it is obvious that our method is robust to the variations of a skin region's color regardless of whether the background of an image contains colors similar to skin color or not. The main cause for the failure of our method in finding normal skin color regions is the presence of reddish or yellowish background regions in an image which are much larger than true skin regions. As mentioned in Section 2.2, if the background color of an image is within the range bounded by the initial thresholds and much more dominant than the color of skin regions, our method fails since the thresholding rectangle moves to the cluster center of the dominant background color vectors during the adaptation procedure and consequently excludes true skin color vectors.

We also tested our method with several tens of images of white and black people, and found that our method worked well as in the case of yellow people. Our method sometimes failed to detect the skin color regions of white

(a) (b)

Fig. 4. Other results of our method: (a) input images; (b) skin color regions found.

people when the regions are too shiny due to the reflection of light so that they almost look white. One of the limitations of our method is that it cannot find the skin color regions at the same time if an input image is taken from a group of people that are composed of several different kinds of races. However, it is theoretically possible to detect them if our method is applied sequentially with different initial settings.

Average execution time of our method is shown in Table 3. All the experiments were performed on a Pentium II (350 MHz) PC and all the codes were written in Matlab 5.3. It should be noted that Matlab codes are usually 9 or 10 times slower than the C/C + + equivalents.

We implemented other conventional methods and applied them to the same test images [3,4]. Fig. 3 shows some results of our method and the conventional ones. Since threshold values of the conventional methods are fixed, the boundaries of the detected skin color regions were usually inaccurate compared to the results of our method, also they tended to detect many unnecessary regions, especially when the background of a test image contains colors similar to skin color. Furthermore, they failed to extract skin color regions when the color of the skin regions was far different from the standard skin color. On the other hand, our method was successful in such cases. Other results of our method are shown in Fig. 4, where the input images multiple faces.

5. Conclusions

In this paper, we proposed a new skin color filter that is useful for finding candidate regions for faces or hands in color images. Since our method is capable of adaptively adjusting its threshold values and effectively separating skin color regions from similar background color ones, it is applicable to images with various conditions. Our method and other conventional methods were tested with a variety of sample images obtained from the Internet and experimental results show that the proposed method is robust to the variations of skin regions' color compared to the conventional ones.

References

[1] S.D. Buluswar, B.A. Draper, Color recognition in outdoor images, Proceedings of the International Conference on Computer Vision, 1998, pp. 171–177.

[2] K. Sobottka, I. Pitas, Extraction of facial regions and features using color and shape information, Proceedings of the International Conference on Pattern Recognition, Vol. 3, 1996, pp. 421–425.

[3] M.M. Fleck, D.A. Forsyth, C. Bregler, Finding naked people. Proceedings of the European Conference on Computer Vision, Vol. 2, 1996, pp. 592–602.

[4] J. Yang, A. Waibel, A real-time face tracker, Proceedings of the IEEE Workshop on Applications of Computer Vision, 1996, pp. 142–147.

[5] J. Yang, W. Lu, A. Waibel, Skin color modeling and adaptation, Proceedings of the Asian Conference on Computer Vision, Vol. 2, 1998, pp. 687–694.

[6] S.J. McKenna, Y. Raja, S. Gong, Object tracking using adaptive colour mixture models, Proceedings of the Asian Conference on Computer Vision, Vol. 1, 1998, pp. 615–622.

[7] J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, R.L. Phillips, Introduction to Computer Graphics, Addison-Wesley, Reading, MA, 1993.

[8] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison-Wesley, Reading, MA, 1992.

[9] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.

About the Author—KYUNG-MIN CHO received the B.S. and M.S. degrees in Electrical & Electronic Engineering from POSTECH, Korea, in 1996 and 1998, respectively. He currently works in Mirero System. His research interests include face detection and recognition.

About the Author—JEONG-HUN JANG was born in Seoul, Korea, in 1970. He received the B.S. degree in Electrical Engineering from Hanyang University, Korea, in 1994 and the M.S. degree in Electrical & Electronic Engineering from POSTECH, Korea, in 1996. He is now in the Ph.D. program at POSTECH. His research interests include feature extraction, character recognition, and face recognition.

About the Author—KI-SANG HONG received the B.S. degree in Electronic Engineering from Seoul National University, Korea, in 1977, and the M.S. degree in Electrical & Electronic Engineering from KAIST, Korea, in 1979. He also received the Ph.D. degree from KAIST in 1984. During 1984–1986, he was a researcher in the Korea Atomic Energy Research Institute and in 1986, he joined POSTECH, Korea, where he is currently an associate professor of Electrical & Electronic Engineering. During 1988–1989, he worked in the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA, as a visiting professor. His current research interests include computer vision and pattern recognition.