Labeling Algorithm for Face Detection
Using Skin and Hair Characteristics

Pouya Ghofrani, Zahra Neshat, and Hassan Aghaeinia

Abstract—This research presents an algorithm for face detection based on color images using three main components: skin color characteristics, hair color characteristics, and a decision structure which converts the obtained information from skin and hair regions to labels for identifying the object dependencies and rejecting many of the incorrect decisions. Here we use face color characteristics that have a good resistance against the face rotations and expressions. This algorithm is also capable of being combined with other methods of face recognition in each stage to improve the detection.

Index Terms—Edge detection, hair region, label, object dependencies, skin region, threshold.

1. Introduction

Detection of the human face is one of the most important fields of computer vision and due to its wide applications in human computer interaction (HCI) and video surveillance, it is becoming more popular with each passing year. The main challenge of face recognition is the wide range of variations in colors and illuminations. Indeed, we must obtain our desired information from the ones that can also lead us to incorrect results. To reach this goal, lots of methods have emerged, classified, and developed over the recent years. Some methods are based on color images and due to the use of color characteristics, they are not applicable on black-white peers. In this article, we would like to use statistical color ranges of the face and hair separately besides an object oriented algorithm to detect faces more reliably. Here, we use the “face detection” and “face recognition” with the same meaning.

There are several views for detecting the skin and hair color regions in different color spaces such as YCbCr (Y: luminance, Cb: chroma blue, Cr: chroma red), RGB (red, green, blue), and HSI (hue, saturation, intensity).

Since the skin color region varies from one person to another, it is difficult to obtain a unique solid region that includes all colors possible and at the same time, it is small enough to prevent adding false information. However, even if we are able to find such a region, there is no guarantee that this region only detects the skin pixels and no more data from other parts of the image with this color in it. This problem also exists about the hair. So we need a decision algorithm to identify the face characteristics and reject others as much as possible. In this article, we have proposed a labeling algorithm that converts the face elements, i.e. hair and skin, to several labeled objects. By this virtue, it is easier and faster to work with the chosen elements. One of the advantages of this algorithm is the object scale comparability. This property lets us remove the small size objects with respect to a threshold determined arbitrarily by a fraction of the greatest object size. Another advantage is preserving the combinability with any other method of face recognition in any step to improve the reliability. Here, we deplete the objects of the information matrices quickly with special filters such that the remained edges contain the same useful information as the primary matrix had before. Therefore the load of processing decreases and subsequently the speed of detection can increase. This algorithm also has a suitable logic to determine the object dependencies.

2. System Overview

The input image is divided into two parts which are determined by the conditions of the skin and hair color regions. Depending on the chosen color space, we can use different methods for skin detection. The intensity element in RGB model allows us to find a suitable region for dark color hairs. The outcomes from here will be segmented by 5×3 or 5×5 blocks. Hereafter, we use edge detection with the “Canny” filter to recognize the external and marginal edges. This filter plays an important role in the detection procedure because it preserves the edge continuity better in comparison with other filters such as “Sobel” and “Prewitt”. Then each result matrix will be labeled and converted to a matrix of objects. Now the algorithm makes decision which of the labeled objects has correlation with others. The performance of the algorithm is explained in the next parts in details. Fig. 1 exhibits the total view used here for recognition.
3. Detecting the Skin and Hair Regions

Crowley and Coutaz\(^1\) said one of the simplest algorithms for detecting skin pixels was to use the skin color algorithm. There are several proposed methods to find the skin color region, however, as the intensity and illumination changes affect the skin detection procedure and subsequently the face recognition result immensely, we need to use histogram normalization before applying any detection method. Here, we assume that the normalization procedure is performed as a primary step in both of the skin and hair region detection modules. We may use the skin characteristics in each of the HSI, RGB, or YCbCr color spaces, however as in \cite{1}, it seemed that combining them will be more efficient.

On the other hand, the real skin pixels are just several members of the pixels in the region obtained above and so this algorithm fails singly when there are more skin regions like legs, arms, etc. Hence, if we are able to detect the hair color region as well, we will be able to remove many of the undesired parts. The problem with detecting the hair region is the wide range of variations in color and illumination as we had known about the skin before. In addition, we cannot use the hair detection for all humans. All in all, as Chen and Lin\(^2\) suggested, we can use the intensity element in the HSI color model to detect the dark range of hair.

4. Segmentation

Segmentation is a module in which images are quantized by \([m] \times [n]\) pixels blocks such that the image resolution decreases; nonetheless, some of the geometrical calculations are speeded up\(^2\). Furthermore, it can also eliminate some of the insignificant areas of pixels detected in both hair and skin regions. The amounts of ‘m’ and ‘n’ in the above can be suitably chosen according to the image size. In this article, we prefer \([5 \times 3]\) or \([5 \times 5]\) blocks for quantization. Using the non-uniform quantization (e.g. \([5 \times 3]\)) is more efficient than using the uniform quantization (e.g. \([5 \times 5]\)) because of the natural face dimensions.

Fig. 2 exhibits the results of another kind of quantization done by a programming code such that it just eliminates the pixels which cannot participate in the construction of any \([5 \times 5]\) pixels block. It dose not miss much information.

5. Edge Detection

After the quantization procedure, each of the acquired images is comprised of many distinct areas which we call them “objects”. In addition, each of them has one color and so we have not any internal edges. Therefore, the result of the edge detection can provide almost the same information that the whole object does. This step is not necessary; nonetheless, it decreases the amount of unnecessary information in matrices and subsequently speeds up the face detection. It is also capable of providing brief but

![Fig. 2. Quantization by a programming code: (a) sample image, (b) quantization of the skin matrix, and (c) quantization of the hair matrix.](image-url)
useful information about the location of the probable faces for other methods. The important point here is preserving the edge continuity as much as possible, such that the solidarity of the object survives. So, we need to use a suitable filter for edge detection like the “Canny” filter in MATLAB software. The objects we have at the end of this step are in the form of closed curves.

6. Labeling

In order to identify the distinct objects and refer to them afterwards, we need to label them. Also, we define a scale for each of them by counting the number of their elements, so that we can sort them with respect to this property to remove the small parts in size by an arbitrary threshold.

7. Decision Algorithm

The thresholds for skin and hair labels are included in the decision algorithm and preferred to be as follows respectively.

1) THS: the threshold for skin labels (1/5 of maximum scale of skin labels, preferred).
2) THH: the threshold for hair labels (1/3 of maximum scale of hair labels, preferred).
3) \([Y]\) Hair+3: is the same as that certain hair label matrix, when we shift all of its ‘Y’ components by 3 pixels.

Objects (labels) with the scale less than the THS or THH are not eligible for face locations and will have their information removed from the next considerations. As shown in Fig. 4, the algorithm has a simple logic and so it is fast enough to respond to the multi-face images properly. The main core of the algorithm starts from the section which checks the intersection of the skin and the matrix \([Y]\) Hair+3. The reason that we shift all of the ‘Y’ components of all eligible hair labels by the number 3 originates from the properties of quantization procedure done before. In spite of other methods\(^2\), we do not use the direct intersection of skin and hair labels to confirm a fifty percent correlation, because throughout the quantization by \([5\times5]\) or \([5\times3]\) pixels blocks, we may lose this intersection feature; so, we choose an average model as illustrated in Fig. 3.

The above figure shows simply why we use 3 pixels as an average form to shift the hair labels and then make decision whether there is a common pixel with the skin label or not as a preparation for next steps. It is clear that why this consideration solitarily is not enough to have a fair judgment about the complete correlation of two chosen labels; so, we use another condition in the next step to find out whether there is a relationship between two labels. \(M_{\text{ys}}\) and \(M_{\text{yh}}\) are two parameters defined by the following statements that we use for this purpose:

\[
\begin{align*}
M_{\text{ys}} &= \frac{\sum \text{All (non-zero) pixels of the lable } S_i Y(S_i)}{\text{Number of the pixels of the lable } S_i} \\
M_{\text{yh}} &= \frac{\sum \text{All (non-zero) pixels of the lable } H_j Y(H_j)}{\text{Number of the pixels of the lable } H_j}
\end{align*}
\]

where \(\lfloor X \rfloor = \max \{n \in \mathbb{Z} | n \leq X\}\) or the floor of \(X\).

---

![Diagram](https://via.placeholder.com/150)

Fig. 3. \([5\times5]\) and \([5\times3]\) blocks of pixels with their central pixels.

![Diagram](https://via.placeholder.com/150)

Fig. 4. Decision algorithm flowchart.
In other words, each of the above parameters shows the average of the whole ‘Y’ components of its correspondent label. Now, we can understand what the condition \( M_{15} - 10 > M_{65} \) does. This condition just allows those hair labels to proceed through the algorithm that are placed in a higher location (or equivalently lower ‘Y’ component) averagely.

Choosing a constant number for comparison is quite arbitrary and here we used the number ‘10’ due to some outcomes from statistical experiments and observations done for a series of sample images. The point of Fig. 4 that should be paid attention is the time element which has not been shown in the flowchart. This means that the algorithm in the dashed-gray color runs as follows.

For each skin label, we examine the eligibility of the whole hair labels, and after finished, we call other skin labels regularly and follow the same procedure.

We did not show this subject in the flowchart due to complexity.

### 7.1 Border and Center Detection

After all of the labels, including both skin and hair labels of each of the distinct faces, are found. We start to use the border and center detection module to obtain the borders of any face in isolation. This can be found from the following rule simply:

\[
I_{F_i} = \sum_{\text{all label matrices-}F_i} \text{All label matrices}
\]

where \( F_i \) means the \( i \)th face detected and ‘~’ shows the assumed labels belong to \( F_i \). Also, in the following equations, the mathematic symbol ‘\(|\)’ means “such that”.

In continue, when we say for an example \( X \in I_{F_i} \) or \( Y \in I_{F_i} \), we have assumed that \( X \) or \( Y \) contains non-zero information.

Upper border line equation:

\[
Y = \text{Min} \{ Y | Y \in I_{F_i} \}
\]

\[
\text{Min} \{ X | X \in I_{F_i} \} < X < \text{Max} \{ X | X \in I_{F_i} \}.
\]

Lower border line equation:

\[
Y = \text{Max} \{ Y | Y \in I_{F_i} \}
\]

\[
\text{Min} \{ X | X \in I_{F_i} \} < X < \text{Max} \{ X | X \in I_{F_i} \}.
\]

Right border line equation:

\[
X = \left[ \text{Max} \{ X | X \in I_{F_i} \} - \frac{\Delta}{6} \right]
\]

\[
\text{Min} \{ Y | Y \in I_{F_i} \} < Y < \text{Max} \{ Y | Y \in I_{F_i} \}
\]

Left border line equation:

\[
X = \left[ \text{Min} \{ X | X \in I_{F_i} \} + \frac{\Delta}{6} \right]
\]

\[
\text{Min} \{ Y | Y \in I_{F_i} \} < Y < \text{Max} \{ Y | Y \in I_{F_i} \}
\]

where

\[
\Delta = \text{Max} \{ X | X \in I_{F_i} \} - \text{Min} \{ X | X \in I_{F_i} \}
\]

and the center is easily found as follows:

\[
X_{\text{center}} = \frac{\text{Min} \{ X | X \in I_{F_i} \} + \text{Max} \{ X | X \in I_{F_i} \}}{2}
\]

\[
Y_{\text{center}} = \frac{\text{Min} \{ Y | Y \in I_{F_i} \} + \text{Max} \{ Y | Y \in I_{F_i} \}}{2}.
\]

As we will see in the next part, these centers and borders move to a new location; however, we should know the first characteristics of the center to do this movement. In other words, we use the difference between the new and old centers to find out how much we should move.

### 7.2 Centre Shifting

The reason that we use the centre shifting module is the fact that we would like to mask the detected faces better and we believe that the effective features in determining the face location by this algorithm are the skin labels (objects). With this assumption, we transfer the previous obtained mask to a new location without any change in its proportions, such that the previous center is placed in the location of the statistical center of the skin label matrix calculated as follows:

\[
X_{\text{New center}} = \frac{\sum \text{All pixels of the skin lable } S_i \text{ of } F_i \cdot X(S_i)}{\text{Number of the pixles of the lable } S_i \text{ of } F_i}.
\]

\[
Y_{\text{New center}} = \frac{\sum \text{All pixels of the skin lable } S_i \text{ of } F_i \cdot Y(S_i)}{\text{Number of the pixles of the lable } S_i \text{ of } F_i}.
\]

Now the detection process is complete. (Note that we have defined all of the label (object) matrices to be the same as the original image matrix in size).

### 7.3 Proposed Combinatorial Algorithm

As mentioned in Section 2, we can use the information provided by other methods such as principal component analysis (PCA), linear discriminant analysis (LDA) or Gabor wavelets besides this algorithm to enhance the system reliability. This is practical because of the object oriented property of the algorithm that quickly prepares the raw information of face locations via labeling such that the methods mentioned above can focus on them instead of the whole image. Also, we note that the data within the hollow objects (labels) that are obtained from this algorithm can be recovered easily in many cases depending on how much edges have kept the continuity; therefore, if other methods need these data, they are reachable.
In the combinatorial algorithm, we change the input of the PCA, LDA, or Gabor wavelet algorithms simply as explained below:

\[
P \cong \text{Primary (original) image matrix}
\]

\[
m_i = \text{Max} \left\{ Y \mid Y \in I_{F_i} \right\} - \text{Min} \left\{ Y \mid Y \in I_{F_i} \right\}
\]

\[
n_i = \text{Max} \left\{ X \mid X \in I_{F_i} \right\} - \text{Min} \left\{ X \mid X \in I_{F_i} \right\}
\]

\[N = \text{Number of the } I_{F_i} \text{ matrices}
\]

\[
J_{F_i} = \begin{bmatrix} J_{F_{i1, X, Z}} \\
J_{F_{i2, X, Z}} \\
\vdots \\
J_{F_{i3, X, Z}} \end{bmatrix}_{mn \times 3}
\]

\[\text{Input for PCA/LDA/Gabor wavelet } = \bigcup_{i=1}^{N} \left\{ J_{F_i} \right\}.
\]

Now the obtained method uses more conditions including skin labels, hair labels, the correlation between them, and PCA/LDA/Gabor wavelet decisions that make the system more reliable.

In addition to the methods mentioned, we can also use the mouth color and its position to have a better color based face recognition. As Bian and Du suggested\(^3\), using a restrictive condition and a distance model from mouth to the center of the eyes seems suitable for this purpose. The condition they used in RGB color space is as in the following\(^3\):

\[0.2 \theta < 0.5(2 \text{arc cos} \left( \frac{0.5(2R-G-B)}{\sqrt{(R-G)(B-G)+(R-B)(G-B)}} \right) \}.
\]

7.4 Acceleration

The acceleration here in the combinatorial method occurs because of the elimination of improbable regions for faces. In addition to this, we can ignore the hair label condition for those cases including persons who have no hair; of course in this case, we should use a potent skin detection method like the method discussed in [4]. Overall, the experience has shown that finding color regions for face detection is usually more convenient and faster than applying strong and sophisticated mathematic conditions in color images. However, we can use the mobility/movement of the human faces for real-time face recognition in comparison with background, too\(^5\).

8. Results

In this section, we would like to depict how the defined scales are used for detection and what labels look like in forms of images. The detection here for the both examples has been done by the skin and hair models proposed in [2]; however, as mentioned earlier, it is not mandatory to use just a specific model and for instance, the model discussed in [1] based on a combination of HSI-RGB-YCbCr color spaces is also suitable.

Fig. 5 and Fig. 6 show the scales of the labels and compare them from this viewpoint. These chart demonstrations have been done just for the image in Fig. 2 (a) as an example. (Note that the skin and hair detection for this figure has been done in Fig. 2 (b) and Fig. 2 (c), respectively.)

Hereafter, we apply the thresholds THS and THH for the skin and hair labels respectively to select the most probable labels and reject the others. Fig. 7 shows the results.

As shown in Fig. 7, the label matrices are so simple and this is the reason that we can easily connect them to other methods for face detection. The final result of detection has been shown in Fig. 8.

The second example also follows the same procedure. Note that the face here is under illumination variations. Also as Fig. 9 (c) shows, the hair region recognized by the model discussed in [2] is a large region; however, the decision algorithm works well due to the suitable conditions defined in it.

![Fig. 5. IDS: the scale for skin labels in the terms of number of pixels (a) table of exact values and (b) comparative view.](image-url)
Fig. 6. IDH: the scale for hair labels in the terms of number of pixels (a) table of exact values and (b) comparative view.

Fig. 7. Labeled objects eligible for face detection in Fig. 2 (a): (a) the most probable skin label selected for the face of the woman, (b) the previous part, i.e. (a), when the most probable hair label has been added to it after successfully satisfying the conditions of correlation (the 3 pixels intersection condition and $M_{xs} - 10 > M_{sh}$), and (c) the same as part (b) but for the child.

Fig. 8. Result of detection.

8. Conclusions

This article introduces a labeling algorithm for face recognition in color images that uses the skin and hair color regions proposed in [1], [2], [6], and [7], an area-label conversion system, and a suitable logic to identify the correlation between labels quickly. This algorithm is capable of detecting several human faces at the same time and sharing information with other methods to increase the overall system reliability. As could be seen from the algorithm flowchart, it uses simple logical operations on a small amount of data which is the labeled objects to recognize the faces.

Fig. 9. Second example: (a) original image, (b) skin region detected and quantized, (c) hair region detected and quantized, (d) combination of all eligible labels detected for a face, and (e) final result after center shifting.

Note that the amounts of $A$, THS, THH, and the shift value of ‘10’ pixels in the inequality $M_{xs} - 10 > M_{sh}$ have been chosen arbitrarily and so we can choose other values for them; however, they have shown a statistically appropriate behavior.

References


Pouya Ghofrani was born in Sanandaj, Iran, in 1989. He received the B.S. degree in electrical engineering from the Amirkabir University of Technology (Tehran Polytechnic) of Iran (AUT), Tehran in 2012. He is currently pursuing the M.S. degree with the Faculty of Electrical Engineering and Information Technology, RWTH AACHEN, Germany. His research interests include image processing, electromagnetics, and wireless communications.

Zahra Neshat was born in Tehran, Iran in 1988. She is now pursuing her B.S. degree in telecommunications engineering with AUT, Tehran. Her research interests include image processing, face recognition, and electronics.

Hassan Aghaeinia (photograph not available at the time of publication) was born in Khonsar, Iran. He received his B.Sc. degree in electronic engineering from AUT in 1987. In 1989, he finished his M.Sc. in Amirkabir University, in 1992. He received the M.Sc. from Valenciennes University (UVHC), Valenciennes, France. Then he continued his studies towards the Ph.D. degree with UVHC in electronic engineering, and received his Ph.D. degree in 1996. From 1996 till present, he has been a faculty member with Amirkabir University, where he is an associate professor in Communication Engineering Group. His research includes digital communications & spread spectrum systems, advanced communication systems, digital image processing, and image communication.