IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 3, Ver. IV (May - Jun. 2014), PP 85-89
www.iosrjournals.org

Performance and Analysis of Face Recognition Using Skin Color Segmentation Algorithm

Riyazahammad Nadaf[1], Mr. B. K. Saptalakar[2]
M.Tech (digital electronics)[1], Assistant Professor[2],
Dept.of E&CE, S.D.M. College of Engineering and Technology, Dharwad

Abstract: *Face Recognition System has been an important and active research field because it offers many applications, especially in video surveillance, biometrics, or video coding. The goal of this project is to detect a human's face. The face detection algorithm involved color-based skin segmentation and image filtering. The face location is determined by calculating the centroid of the detected region. However, it is difficult to develop a complete robust face detector due to various light conditions, face sizes, face orientations, background and skin colors. This detects skin regions over the entire image, and then generates face candidates based on a connected component analysis then the face candidates are divided into human face and non-face images.*
Keywords: *Face Recognition, Segmentation, Centroid.*

I. Introduction

Face Recognition System is the process of determining whether or not a face is present in an image. Unlike face recognition which distinguishes different human faces, face detection only indicates whether or not a face is present in an image. In addition, Face Recognition System determines the exact location of the face. Face Recognition System has been an active research area for a long time because it is the initial important step in many different applications, such as video surveillance, faces recognition, image enhancement, video coding, and energy conservation. The applicability of face detection in energy conservation is not as obvious as in other applications. However, it is interesting to learn how a Face Recognition System allows power and energy to be saved. Suppose one is watching a television and working on other tasks simultaneously. The face detection system is for checking whether or not the person is looking directly at the TV. If the person is not directly looking at the TV within some time period (i.e. 15 minutes), the TV's brightness is reduced to save energy. When the person turns back to look at the TV, the TV's brightness can be increased back to original. In addition, if the person looks away for too long (i.e. more than one hour), then the TV will be automatically turned off. Different approaches to detect and track human faces including feature-based, appearance-based, and color based have been actively researched and published in literature. The feature based approach detects a human's face based on human facial features such as eyes and nose. Because of its complexity, this method requires lots of computing and memory resources. Although compared to other methods this one gives higher accuracy rate, it is not suitable for power-limited devices. Hence, a color-based algorithm is more reasonable for applications that require low computational effort. In general, each method has its own advantages and disadvantages. More complex algorithm typically gives very high accuracy rate but also requires lots of computing resources. In this project in order to detect the face region a skin-color segmentation method is used.

II. Design And Implementation Algorithm

The skin detection algorithm is derived. Color segmentation has been proved to be an effective method to detect face regions due to its low computational requirements and ease of implementation. Compared to the featured based method, the color-based algorithm required very little training. First, the original image was converted to a different color space, namely modified YUV. Then the skin pixels were segmented based on the appropriate U range. Morphological filtering was applied to reduce false positives. Then each connected region of detected pixels in the image was labeled. The area of each labeled region was computed and an area-based filtering was applied. Only regions with large area were considered face regions. The centroid of each face region was also computed to show its location.

1. Modified YUV Color Space

Converting the skin pixel information to the modified YUV color space would be more advantageous since human skin tones tend to fall within a certain range of chrominance values (i.e. U-V component), regardless of the skin type. The conversion equations are shown as follows.

$$Y = (R+2G+B)/4 \quad (1)$$
$$U = R - G \quad (2)$$
$$V = B - G \quad (3)$$

These equations allowed thresholding to work independently of skin color intensity.

2. Thresholding/Skin Detection

After skin pixels were converted to the modified YUV space, the skin pixels can be segmented based on the following experimented threshold.

$$10 < U < 74$$
$$-40 < V < 11$$

As seen in fig 2, the blue channel had the least contribution to human skin color. Leaving out the blue channel would have little impact on thresholding and skin filtering. This also implies the insignificance of the V component in the YUV format. Therefore, the skin detection algorithm using here was based on the U component only. Applying the suggested threshold for the U component would produce a binary image with raw segmentation result, as depicted in fig 1.

Figure 1 Result after thresholding

3. Morphological Filtering

Realistically, there are so many other objects that have color similar to the skin color. As seen in fig. 2, there are lots of false positives present in the raw segmentation result. Applying morphological filtering including erosion and hole filling would, firstly, reduce the background noise and, secondly, fill in missing pixels of the detected face regions, as illustrated in fig. 4. MATLAB provided built-in functions—imerode and imfill for these two operations.

outp = imerode(inp, strel('square', 3));

The command imerode erodes the input image inp using a square of size 3 as a structuring element and returns the eroded image outp. This operation removed any group of pixels that had size smaller than the structuring element's.

outp = imfill(inp, 'holes');

The command imfill fills holes in the binary input image inp and produces the output image outp. Applying this operation allowed the missing pixels of the detected face regions to be filled in. Thus, it made each face region appear as one connected region.

Figure 2 Result after morphological filtering

4. Connected Component Labeling and Area Calculation

After each group of detected pixels became one connected region, connected component labeling algorithm was applied. This process labeled each connected region with a number, allowing us to distinguish between different detected regions. The built-in function bwlabel for this operation was available in MATLAB. In general, there are two main methods to label connected regions in a binary image known as recursive and sequential algorithms.

[L, n] = bwlabel(inp);

The command bwlabel labels connected components in the input image inp and returns a matrix L of the same size as inp. L contains labels for all connected regions in inp. N contains the number of connected objects found in inp.

The command regionprops can be used to extract different properties, including area and centroid, of each labeled region in the label matrix obtained from bwlabel.

face_region = regionprops(L, ‘Area’);
face_area = [face_region.Area];

The two commands above performed two tasks (1) extract the area information of each labeled region (2) store the areas of all the labeled regions in the array face_area in the order of their labels. For instance face_area(1) = 102 would mean the area of the connected component with label “1” is 102 pixels.

5. Area-based Filtering

Morphological filtering only removed some background noise, but not all. Filtering detected regions based on their areas would successfully remove all background noise and any skin region that was not likely to be a face. This was done based on the assumption that human faces are of similar size and have largest area compared to other skin regions, especially the hands. Therefore, to be considered a face region, a connected group of skin pixels need to have an area of at least 26% of the largest area. This number was obtained from experiments on training images. Therefore, many regions of false positives could be removed in this stage, as depicted in fig. 3.

face_idx = find(face_area > (.26)*max(face_area));
face_shown = ismember(L, face_idx);

These two commands performed the following tasks (1) look for the connected regions whose areas were of 26% of the largest area and store their corresponding indices in face_idx (2) output the image face_shown that contained the connected regions found in (1).

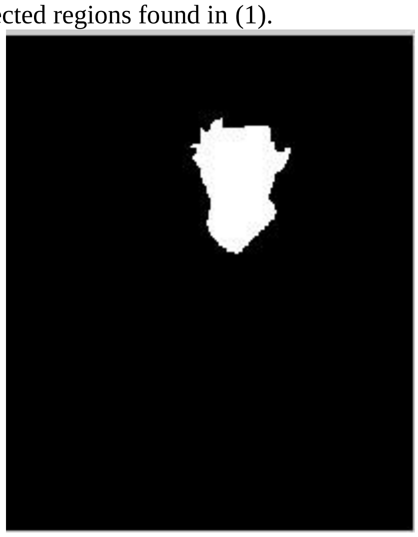

Figure 3. Result after area-based filtering

6. Centroid Computation

The final stage was to determine face location. The centroid of each connected labeled face region can be calculated by averaging the sum of X coordinates and Y coordinates separately. The centroid of each face region in fig 4 is denoted by the blue asterisk. Here the centroid of each connected region was extracted using regionprops.

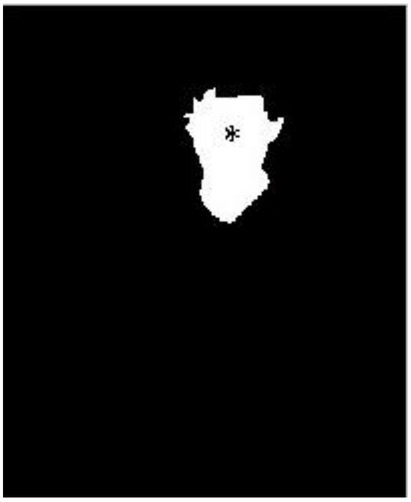

Figure 4. Result after calculating centroid

III. Face Recognition Applications

Face recognition is also useful in human computer interaction, virtual reality, database recovery, multimedia, computer entertainment, information securituy e.g operating system, medical records, online banking, biometric e.g personal identification – passports, driving licences, automated identity verification border controls, law enforcement e.g video surveillances, investigation, personal security driver monitoring system, home video surveillances system.

1. Face Identification

Face recognition systems identify people by their face images. Face recognition systems establish the presence of an authorized person rather than just checking whether a valid identification (ID) or key is being used or whether the user knows the secret personal identification numbers (Pins) or passwords.

Consider To eliminate duplicates in a nationwide voter registration system because there are cases where the same person was assigned more than one identification number. The face recognition system directly compares the face images of the voters and does not use ID numbers to differentiate one from the others. When the top two matched faces are highly similar to the query face image, manual review is required to make sure they are indeed different persons so as to eliminate duplicates.

2. Access Control

In many of the access control applications, such as office access or computer logon, the size of the group of people that need to be recognized is relatively small. The face pictures are also caught under natural conditions, such as frontal faces and indoor illumination. The face recognition system of this application can achieve high accuracy without much co-operation from user. The following are the example. Face recognition technology is used to monitor continuously who is in front of a computer terminal. It allows the user to leave the terminal without closing files and logging out. When the user leaves for a predetermined time, a screen saver covers up the work and disables the mouse & keyboard. When the user comes back and is recognized, the screen saver clears and the previous session appears as it was left. Any other user who tries to logon without authorization is denied.

3. Security

Security is a primary concern at airports and for airline staff office and passengers. Airport protection systems that use face recognition technology have been implemented at many airports around the world. The system is designed to alert airport public safety officers whenever an individual matching the appearance of a known terrorist suspect enters the airport's security checkpoint. Anyone recognized by the system would have further investigative processes by public safety officers. Computer security has also seen the application of face recognition technology. To prevent someone else from changing files or transacting with others when the authorized individual leaves the computer terminal for a short time, users are continuously authenticated, checking that the individual in front of the computer screen or at a user is the same authorized person who logged in.

IV. Conclusion

In this paper, the goal is to detect human faces is achieved. A software implementation of the algorithm was examined in MATLAB to verify its accuracy. Although it required some modification to the original algorithm, the initial goal is still accomplished. The face detection algorithm is derived from a skin detection method. Face detection is achieved by computing the centroid of each detected region. Different types of filter were applied to avoid flickering and stabilize the output displayed on the VGA screen.

References

[1]. A. J. Goldstein, L. D. Harmon, and A. BLesk,“Identification of Human Faces,” Proc.IEEE, May 1971, Vol. 59, No. 5, 748-760.
[2]. J. Nagi, “Design of an Efficient High-speed Face Recognition System”, Department of Electrical and Electronics Engineering, College of Engineering, Universiti Tenaga Nasional, March 2007.
[3]. R. Brunelli and T. Poggio, "Face Recognition: Features versus Templates", IEEE Trans. on PAMI, 1993, (15)10:1042-1052.
[4]. E. Hjelmås, and B. K. Low, “Face detection: A survey”, Computer Vision and Image Understanding, Vol. 83, No. 3, Sept. 2001, pp. 236-274.
[5]. L. Sirovich and M. Kirby, "A Low-Dimensional Procedure for the Characterization of Human Faces," J. Optical Soc. Am. A, 1987, Vol. 4, No.3, 519-524.
[6]. M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces," Proc. IEEE, 1991, 586-591.
[7]. D. Bolme, R. Beveridge, M. Teixeira, and B. Draper, “The CSU Face Identification Evaluation System: Its Purpose, Features and Structure,” International Conference on Vision Systems, Graz, Austria, April 1-3, 2003. (Springer-Verlag) 304-311.