Location and segmentation of facial features combining PSO algorithm and skin color

Li Zhi-jie

School of Computer Science and Engineering, Dalian Nationalities University, Dalian 116600, China,
lizhijie@dlnu.edu.cn

Abstract

A new method using PSO algorithm and skin color for the location and segmentation of face and facial features is proposed. In the preprocessing stage, segmented face image is obtained from initial color image. To achieve this goal, PSO algorithm is applied to search for the best face region. Then, based on the edge density of face image, eye region is located with PSO. Then, lips region is located using color component of skin segmentation. Finally, nose region is segmented based on the result of eyes and lips. The Simulation results show that this hybrid method is accurate and effective.

Keywords: Face features; edge detection; particle swarm optimization; skin segmentation; cubic spline

1. Introduction

Face localization and feature extraction are fundamental steps in the process of face recognition [1]. The accuracy of the located face center coordinates and orientation has a heavy influence on the recognition performance [2]. To ensure a robust and accurate feature extraction that distinguishes between face and non-face regions in an image, we require the exact location of the face in two-dimensional images. To address this issue, many different approaches and models are developed to detect human face. They can be divided into three categories: including using shape[3,4], color information [5], and motion[6]. However, face detection is still a handicap to the facial feature extraction algorithms[7-9] due to its high complexity.

For the approach of using shape, the face edge is often used for the shape information of detection. In practical applications, it is found that some discontinuous edge points are difficult to form meaningful contour. For the approach of using color, its information has been commonly used to assist face detection. Numerous color spaces have been developed and suggested for various applications and purposes. They are usually obtained via a transform from RGB space or other 'basic' color space. In some particular color space, segmentation method based on the color of the skin area can separate the skin color region from other image parts, since the face skin area is different from other regions in terms of color component. But, use of color in facial skin detection cannot necessarily distinguish faces from other objects (e.g., hands, arm) with a similar appearance. Other cues are therefore needed to verify that the selected area is indeed a face.

The above mentioned methods have their own shortcomings. Also, because all the pixels within the face region were evaluated to find the best ones, the existed algorithms couldn't achieve fast searching. Therefore, combined with particle swarm optimization (PSO) [10-14], this paper deeply studies the fast and accurate segmentation of human face, as well as the facial features. The proposed algorithm in this paper comprehensively utilizes the characteristics of skin color, shape information and PSO optimization to confirm the edge points of face and facial features. Though extracting the real edge points and removing false edge points, the human face segmentation is completed. For the stage of facial features, PSO algorithm is used to locate eyes based on edge information of the face image. Then combining with the HSI color space, our method realize the rapid and accurate positioning of the lips region. Finally according to the facial features location as well as their relationship, nose region is well located.

2. Face Location in Color Space

People have been constantly looking for a color space can express the skin color in unique statistical characteristics. Some studies found that when human skin color is in the C_r component of YC_bC_r space, the distribution of skin samples and non-skin samples are relatively concentrated. Although Skin and non-skin areas have small amount of overlap, their peaks have certain dislocation, which can be deal with threshold segmentation. In practice, color images are generally in RGB color space. The example image is transformed from RGB color space to the YC_bC_r color space.

Then the gray image of C_r component is saved as a one-dimensional matrix with the size of 136×170, as shown in Figure 1(a). From Figure 1(a) we can see that, face region has been clearly distinguished from other background color area. In order to quantitatively detect the face area, the gray image of C_r component needs binary processing. In the YC_bC_r color space, the skin color areas mainly concentrate between 140 and 160 on the C_r component. Based on this threshold interval, the gray image of C_r component can be converted into binary image, which makes the face area more prominent. Thus, we can get the formula for judging face color as follows:

$$f(i,j)=\begin{cases}1, & C_r \in [a,b]\\ 0, & else\end{cases} \tag{1}$$

,in which f(i,j) is the image after threshold segmentation; a, b on behalf of the selected threshold range; i, j represent the two-dimensional coordinates of each pixel. The binary image is shown as Figure 1(b). Black pixel corresponds to the value 0 and white pixel corresponds to the value 1. In this paper, we use close operation to get face region image without holes, shown as Figure 1(c). Now, although the skin region has been extracted, but the facial part still can not be distinguished strictly from other body parts such as neck.

(a) C_r component (b) binary image (c) image close operation

Figure1. Comparison of Cr component and its binary image

3. Face Segmentation using PSO searching

3.1 Edge Detection

Before edge detection, the image that will be dealt with should be converted to gray image. Gray image is defined as a binary function $f(x,y): D \to G$, where $(x,y) \in D$ represents the pixel coordinates in the two dimensional space, and $f \in G$ represents the gray value of pixel (x, y). Gray value can be calculated as the integer which denotes the lightness degree of pixel. The binary image of face contour obtained in the previous steps is a special gray image. Each pixel in it is either black or white, without intermediate gray value. Hence, the binary image don't have to be transformed, and can be used directly to extract edge. As we know that most information of image is saved in the edge, face position will also be detected based on the edge information. Edge exists between the nearest adjacent regions with different gray values. Edge image can be represented with binary image that includes only edge location without degree. In binary image, there are two gray levels (0, 1). 0 denotes black, on the contrary, 1 denotes white. In practice, edge detection can be interpreted as an operation with "point" in local area. Here, the Sobel algorithm is used for generating edge image.

3.2. Feature Extraction

After edge detection, feature extraction method will be used to detect the face position. Human face is rather special because it contains more details than other parts of the body. This is fully reflected in the edge image by the fact that face area contains high density edge points. Certainly, if the background is not complicated, feature extraction of face would be easier. How to define the face region? We have the following assumptions. The face position detection is preceded under the assumption that the face region can be surrounded by a rectangle. This method works well even under the environments when the people pose is complex. It is commonly assumed in literature that the ratio of the length to width of the rectangle area is 1.5: 1(shown as Fig.2), once the face region is located. But there is no limit on the size of rectangle since we don't know the scale of face in the image. The rectangle has three parameters: the abscissa (*x*) and ordinate (*y*) of the upper left corner, the length of the diagonal (*r*) (shown as Figure 2).

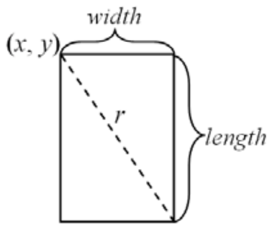

Figure 2. Rectangle parameters of face region

The criterion of feature extraction is defined by the number of edge pixels in the rectangle area to the rectangle perimeter. The ratio is large when the rectangle contains edge points at more frequent. The criterion is calculated below.

$$criterion = \frac{1}{2(x' - x + y' - y)} \sum_{i=x}^{x'} \sum_{j=y}^{y'} e(i, j) \tag{2}$$

$$\text{where } \begin{cases} e(i, j) = 1 & \text{if pixel is black} \\ e(i, j) = 0 & \text{if pixel is white} \end{cases}$$

e is edge image of the original image and *e(i, j)* represents a pixel point with value 1 or 0. *(x,y)* denotes the coordinate of the upper left corner of the rectangle, while *(x', y')* denotes the lower right corner of the rectangle. Thus, according to the ration of length and width, the following relationship exists.

$$x' = x + 3r/\sqrt{13} \tag{3}$$

$$y' = y + 2r/\sqrt{13} \tag{4}$$

Based on the above analysis, as long as the largest ratio of rectangle is found, the task of face detection is completed. The immediate question is how to search such rectangle, which is a very difficult combinatorial optimization problem. Next section, we will utilize PSO algorithm to search the optimal rectangle region.

3.3 PSO searching for face contour

Particle swarm algorithm (PSO) is a global optimization algorithm. Individual in the swarm represents a possible solution. Each particle has two measurements of position and velocity. The objective function value corresponding to the particle position coordinates can be regarded as fitness. Particle swarm algorithm measures particle through fitness, so a remarkable feature of PSO is that it has few parameters to adjust. However, these key parameters have significant impact on the accuracy and efficiency of PSO algorithm. We first study the particle representation. In the PSO algorithm, each particle corresponds to a possible optimal solution. For the optimal problem of face detection, particle representation is relatively simple, i.e., 3-dimensional vector $X_i = (x_{i1}, x_{i2}, x_{i3})$, where elements x_{i1} and x_{i2} denotes the values of parameter *x* and *y* respectively. x_{i3} denotes the value of parameter *r*.

In our PSO algorithm, each particle represents three parameters value (*x, y, r*). Particles search for the optimal location in 3-dimensional space. Corresponding to the location coordinates of each particle, there is a fitness used to measure the property of particle. The particle representing the maximum of $|f'(x, y, r)|$ possesses the best fitness. During the process of iteration, particles will study from the optimal solution of themselves and the optimal solution of the peers in the swarm. Through learning, each particle will adjust its location. Therefore, in the 3-dimensional space of parameters, particles continue flying and eventually towards the optimal solution of derivative. Obviously, objective function $|f'(x, y, r)|$ could be defined as the criterion to measure the property of particle.

According to the idea of PSO, particles fly in a swarm. Each particle represents a candidate solution of the optimization problem in the 3-dimensional space. Suppose that $X_i=(x_{i1}, x_{i2}, x_{i3})$ is position (rectangle) the *i*-th particle visited, *lBest* is personal best position (rectangle) found so far by the *i*-th particle, *gBest* is global best position discovered so far by any of the particles in the swarm, $V_i=(v_{i1}, v_{i2},\ldots, v_{iN})$ is particle velocity. During each iteration, each particle accelerates in the direction of *lBest*, as well as in the direction of *gBest*. Let $X_i(t)$ denote the current position in the search space, $V_i(t)$ is the current velocity. The position and velocity of each particle in the swarm is updated using the following equations

$$V_i(t+1)=\omega\cdot V_i(t)+\sigma_1\cdot rand\cdot(lBest(t)-X_i(t))+\sigma_2\cdot rand\cdot(gBest(t)-X_i(t)) \quad (5)$$

$$X_i(t+1)=X_i(t)+V_i(t+1) \quad (6)$$

where *rand* is used to maintain the diversity of the population, and they are uniformly distributed in the range [0, 1]. The parameters σ_1 and σ_2 are the acceleration coefficients. Proper fine-tuning of these two acceleration coefficients may result in faster convergence and alleviation of local minima. ω is the weight coefficient and $0.1\le\omega\le 0.9$. A suitable value for the weight ω usually provides balance between global and local exploration abilities and consequently results in a reduction of the number of iterations required to locate the optimum solution. The personal best allocation of each particle, *lBest*, is updated by

$$lBest_i(t+1)=\begin{cases} lBest_i(t) & if \quad f(X_i(t)) > f(lBest_i(t)) \\ X_i(t+1) & if \quad f(X_i(t)) \le f(lBest_i(t)) \end{cases} \quad (7)$$

Function *f* is the fitness function that used for judging the update. The global best position, *gBest*, is the best position among all particles in the swarm during all previous steps. It means

$$gBest\ (t+1)=\arg\max_i\ f(lBest_i(t+1)) \quad \forall i\in N \quad (8)$$

Generally, the PSO algorithm terminates when the condition of maximum iteration is satisfied. The *gBest* with best fitness is designated as global best solution of whole particle swarm. The results of searching and segmentation process utilizing PSO is shown in Figure 3.

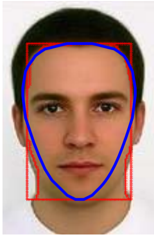

Figure 3. Segmentation process using PSO searching

4. Facial features location and segmentation

4.1 Eyes location using PSO searching

The eyes location is similar to the face location, since they all use the PSO algorithm to find the best position of rectangle containing the target. But eyes location has two differences from the face location. One is that the rectangle shape is different. Previously, we suppose that the ratio of the length to width of the face rectangle area is 1.5: 1. But for eyes rectangle, the ratio of the length to width should be 1: 1.5 according to common sense. The detailed searching process utilizing PSO is similar with that in section 3. Limited by the length of this paper, we're not going to detail the process steps, but show the result as follows. We display the searching results of left eye and right eye on the same face image. The general searching steps is shown in Figure 4.

Figure 4. Eyes detection using PSO searching

4.2 Lips location using skin color segmentation

It is well known that human's lips have a color of natural red. Even if it is black, the lips color is dark red also. Some studies found that when human skin color is in the H component of HSI space, the distribution of lips samples and non-lips samples are relatively concentrated. Images in HSI color space are stored in the computer as a three-dimensional matrix. In this paper, we use a complex posture image containing human face to perform the experiment. The image size is 136×170×3 bit. The image is transformed from RGB color space to the H component of HSI color space, then to the binary image.

The binary image of H component contains a lot of noise after the color segmentation. Therefore, we need to use the morphology to preprocess the binary image to get rid of these holes existed in the binary image. In this paper, we use open operation to get lips region image without holes. Based on the above steps, we can get a complete lip contour. But in order to extract the nose region and keep consistent with eyes, the lips region is needed to be marked with rectangular, as shown in Figure 5.

Figure 5. Illustration of lips detection

4.3 Nose location

In localization process of facial features, the selection and position of basic points is very important, because it determines the accuracy of other features. The section II and section III have successfully extracted the feature regions of eyes and lips. Based on these feature regions, we can locate other feature regions of the face. This paper only deals with the positioning of nose region. Because facial features distribution is regular, other feature regions can be approximately located by facial basic parameters. That is to say, as long as the two corners of the lips and the two corners of the eyes are located, we can calculate the nose region, as shown in Figure 6. We first select two internal corners of rectangular as the two upper vertices of nose region. The two internal corners are the lower right corner of left eye rectangle and the lower left corner of right eye rectangle. Then we select the upper left corner and the upper right corner of lip rectangle as the two lower vertices of nose region. The quadrilateral formed by the four vertices is the nose region.

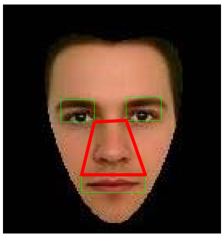

Figure 6. Nose detection based on the location of eyes and mouth

5. Conclusion

In this paper, a face features location method based on PSO algorithm and skin color segmentation technology is proposed. In the preprocessing stage, face area is roughly distinguished from background with the use of color in facial skin detection. Different thresholds of color component are defined for different facial images. Firstly, the face is located by approximating face to a rectangle. The facial features are extracted from edge information of color component image. Then, particle swarm optimization is utilized to search the optimal position of rectangle containing human face. In the phase of face features location, the features detection method mainly focuses on the detection of eyes, lips and nose on human face image. For the eyes detection, the eye is located by approximating eye to a rectangle. After the edge information is extracted from original color image, particle swarm optimization is utilized to search the optimal position of rectangle containing eye. The rectangle region with best fitness value will be detected as the eyes. For the lips detection, color component is used to locate lips region. Different thresholds of color component are defined for different facial images. For the nose detection, the feature region is extracted based on the rectangular of eyes and lips. Simulation results show that PSO algorithm was able to search effectively and reduce computational complexity, therefore reduce the search time.

References

[1] J. Hddadnia, K. Faez, M. Ahmadi. An Efficient Human Face Recognition System Using Pseudo Zernike Moment Invariant and Radial Basis Function Neural Network, *International Journal of Pattern Recognition and Artificial Intelligence*, 2003, 17(1): 41-62.

[2] Hamidreza Rashidy Kanan, Mohammad Hassan Moradi. A Genetic Algorithm based Method for Face Localization and pose Estimation. *Proceedings of the third International Conference: Sciences of Electronic, Technologies of Information and Telecommunications*. 2005.

[3] J. Wang, T. Tan. A new face detection method based on shape information. *Pattern Recognition Letters*. 2000, 21(6-7): 463-471.

[4] K.J. Kirchberg, O. Jesorsky, W. Robert. Frischholz. Genetic Model Optimization for Hausdorff Distance-Based Face Localization. *Proceedings of the International ECCV 2002 Workshop on Biometric Authentication*, 2002, pp. 103-111.

[5] K. Sobotta, I. Pitas. Face localization and facial feature extraction based on shape and color information. *Proceedings of the International Conference on Image Processing*, 1996: 483-486.

[6] R. Herpers, G. Verghese, K. Derpains et al. Detection and tracking of face in real environments. *Proceedings of the International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, 1999, pp. 96-104.

[7] W. Zhao, R. Chellappa, P.J. Phillips, et al. Face recognition: A literature survey. *ACM Computing Surveys*, 2003, 35(4): 399-458.

[8] Jin Hong-wei. The research and Realization of face detection technology based on the color and structural features. *Doctor of Engineering thesis*, National University of Defense Technology. 2007: 1-5.

[9] LIU Xiang-lou， ZHANG Ming， DENG Yan-ru. A Kind of Segmentation Method of Face Object. *Science Technology and Engineering*, 2011, 11(12): 2686-2690.

[10]J. Kennedy, R.C. Eberhart. Particle Swarm Optimization. *Proceedings IEEE International Conference on Neural Networks*, 1995, pp. 1942-1948.

[11]R.C. Eberhart, Y. Shi. Particle swarm optimization: developments, applications and resources. *Proceedings of the IEEE Congress on Evolutionary Computation*, 2001, pp. 81-86.

[12]X.D. Duan, C.R Wang, X.D. Liu. Particle Swarm Optimization and Application. *Liaoning University Press*, Shenyang, 2005.

[13]W.F. Abd-El-Wahed, A.A. Mousa, M.A. El-Shorbagy. Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems, *Journal of Computational and Applied Mathematics*, vol. 235, no. 5, pp. 1446-1453, 2011.

[14]Jian Zhang, Experimental Parameter Investigations on Particle Swarm Optimization Acceleration Coefficients, *International Journal of Advancements in Computing Technology*, Vol. 4, No. 5, pp. 99-105, 2012.

[15]Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins. Digital Image Processing Using MATLAB. *Publishing House of Electronics Industry*, 2012. 4.