Memoirs of the Faculty of Engineering, Okayama University, Vol.40, pp.78-82, January, 2006

An Eye-Contour Extraction Algorithm from Face Image using Deformable Template Matching

Nobuo Funabiki, Megumi Isogai
Grad. School of Natural Science & Tech.
Okayama University
Okayama 700-8530, Japan

Teruo Higashino, Masashi Oda
Grad. School of Inform. Science & Tech.
Osaka University
Suita 565-0871, Japan

(Received December 22, 2005)

A variety of studies on face components such as eyes, lips, noses, and teeth have been proceeding in medicine, psychology, biometrics authentication, and other areas. In this paper, we present an algorithm of extracting eye contours from a face image using the deformable template matching method. Our template for an eye contour is composed of three quadratic functions for the perimeter and one circle for the pupil. In our algorithm, a digital color face image is rst converted to a binary image of representing eyes, after the region around eyes is identi ed on the face image by using hues and values of the color image. Then, parameters in the template are optimized by a local search method with a tabu period and a hill-climbing, so as to t the template to the eye contour in the binary image. The accuracy of our algorithm is evaluated through sample face images of students. In addition, the application of our proposal to eye shape indices is investigated in a face image database "HOIP", where recognizable di erence exists in index distributions between males and females.

1 Introduction

A variety of studies on face components such as eyes, lips, noses, and teeth have been proceeding in medicine, psychology, biometric authentication, and other many areas. For example, in the biometric authentication, entire face images and/or face components have gradually become a focus of attention [1][2], although n-gerprints have been the major information to identify individuals. Users may sometimes resist o ering the information on their ngerprints. Besides, arti cial ngerprints can be made easily in the current technology, so that invalid users may be authenticated as valid users by using them.

In this paper, we present an eye contour extraction algorithm from a face image based on the *deformable template matching method*[3]. The deformable template matching method has gained a growing interest in locating and nding exact shapes and sizes of known objects. Actually, it has been used in many applications including boundary nding in magnetic resonance images [4], extractions of eyes and mouths [5], vehicle segmentation and classi cation for ITS [6], mouth description [7], and dental caries lesions detection [8]. In the method, the shape or contour of the object to be extracted is modeled by a combination of parametric functions such as linear functions, quadratic functions, and circles, called a *deformable template*. The parameter values to constitute the template are searched by an optimization algorithm, so that the template should t the object in a given image as best as possible.

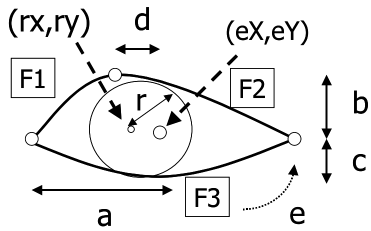
Our template for an eye contour is composed of three quadratic functions for the eye perimeter and one circle for the pupil. Using this simple template, an eye contour can be extracted with a high degree of accuracy, while the statistical processing of eye shapes becomes possible without di culty. In our algorithm, the *eye region image* that only includes the region for eyes in the image is rst extracted from the color face image by using the color information on skins, brows, and hairs. Then, the *eye edge image* for the eye contour is generated by applying *Canny edge detector* [9] to this eye region image. After initial values of parameters for the template are calculated using the eye region image, they are optimized by a local search method with a tabu period and a hill-climbing [10]. The accuracy of eye contours extracted by our algorithm is evaluated through sample face images of students. Then, two indices to describe characteristics of eye shapes are de ned, and their distributions among 100 persons are investigated in the *HOIP (Human and Object Interaction Processing)* Japanese face image database [1]. The results suggest the application of our algorithm to an eye make diagnosis system.

This paper is organized as follows: Section 2 describes the template for an eye contour. Section 3 presents our eye contour extraction algorithm. Section 4 evaluates the accuracy in sample face images. Section 5 investigates distributions of two shape indices in HOIP. Section 6 concludes this paper.

2 Eye Contour Template

The deformable template for an eye contour is composed of three quadratic functions and one circle (Fig. 1). Two quadratic functions, $F1$ and $F2$, represent the upper edge of the eye contour and another one, $F3$, represents the lower edge, where their end points are shared. In the functions, the coordinate origin is the center between the point shared by $F1$ and $F3$ and the point by $F2$ and $F3$, and the x-coordinate is upward and the y-coordinate is rightward. The circle represents the pupil. A total of 10 parameters appear in this template, where a is the half width of the eye, b is the upper height, c is the lower height, d is the distance between the end point shared by two upper functions and the center of the pupil, e is the rotation angle of the eye, (eX, eY) is the center coordinate of the rotation, r is the radius of the pupil, and (rx, ry) is the center coordinate of the pupil.

Here, we note that our template is di erent from the template in [5] that is composed of two quadratic functions and one circle with 11 parameters. Besides, their method is computationally intensive as calculating the cost function composed of ve terms where four terms require integral calculations. Furthermore, the initialization of parameter values is not speci ed, although initial values usually determine the nal solution in their search method based on the steepest descent.



	x-domain	function
F1	[-a, -d]	$y = \frac{b}{(a\ \ d)^2}(x+d)^2+b$
F2	[-d, a]	$y = \frac{b}{(a+d)^2}(x+d)^2+b$
F3	[-a, a]	$y = \frac{c}{a^2}x^2\ \ c$

Figure 1: Deformable template for eye contour.

3 Eye Contour Extraction Algorithm

In this section, we present our eye contour extraction algorithm that is composed of the preprocessing, the initial parameter calculation, and the local search of parameters.

3.1 Preprocessing

In the preprocessing, the binary *eye edge image* is generated from a digital color face image through the following steps:

1. Convert the image from the RGB color system to the HSV color system.
2. Identify the face skin area through binarization of the image by $0.35 \leq H \leq 0.65$ for 1-pixels.
3. Identify the eye area through binarization of the skin area by $V \leq 120$ for 1-pixels.
4. Compose the *eye region image* by removing the hair and the brows by counting the number of 1-pixels rst in the x-axial direction, and then in the y-axial direction.

[1] http://www.hoip.jp/ENG/index.html

5. Generate the *eye edge image* by applying Canny edge detector to the eye region image.

3.2 Initial Parameter Calculation

The initial values for template parameters in the template are calculated using the *eye region image* in the following steps:

1. Find the coordinates of the leftmost 1-pixel and the rightmost 1-pixel. Let (Lx, Ly) and (Rx, Ry) be their coordinates respectively.
2. Set $a = \frac{Rx \quad Lx}{2}$, and $(eX, eY) = (\frac{Lx+Rx}{2}, \frac{Ly+Ry}{2})$.
3. Find the coordinates of the top 1-pixel and the bottom 1-pixel. Let (Tx, Ty) and (Bx, By) be the coordinates respectively.
4. Set $b = (Ty \quad eY)$, $c = (eY \quad By)$, $d = (eX \quad Tx)$, and $e = \frac{\pi}{2} \frac{|Ly \quad Ry|}{2a}$.
5. Find the x-coordinate in which the number of successive 1-pixels with the same y-coordinate becomes maximum. Let Mx be this x-coordinate, and MTy and MBy be the top and bottom y-coordinates of 1-pixels there.
6. Set $(rx, ry) = (Mx, \frac{MTy+MBy}{2})$.
7. Count the number of successive 1-pixels from (rx, ry) with the same y-coordinate in the left and right directions, and set this number for r.

3.3 Cost Function for Parameter Optimization

The parameter values in the template are optimized through minimization of the following cost function E:

$$E = \quad \sum_{p_i \in T} f(p_i) \qquad \sum_{p_i \in F1,F3[\quad a, \quad d]} f(p_i) \quad K \quad |T| \quad (1)$$

where T is the set of 1-pixel coordinates on the template, p_i is a member in T, $|T|$ is the number of 1-pixels on the template, K is a constant coe cient ($K = 0.8$ in simulations), and $f(p)$ is a weighted function given by:

$$f(p) = L \text{ if } p \in I_E,$$
$$f(p) = L \quad l \text{ if } p \notin I_E \text{ and } |p \quad q| = l < L \text{ for } q \in I_E,$$
$$\text{and } f(p) = 0 \text{ otherwise} \qquad (2)$$

where I_E represents the set of 1-pixel coordinates in the eye edge image. The rst term in E maximizes the correspondence between the template and the eye edge image. The second term intensi es the correspondence of the inner part of the eye contour that is more important than the outer part. The last term reduces the template size.

3.4 Local Search Algorithm

The parameter values in the template are optimized by a local search algorithm, such that the cost function in (1) is minimized. In each iteration, one randomly selected parameter is checked whether its increase or decrease by a small xed value reduces the cost function or not. If either change of the parameter actually reduces it, the value is modi ed in that manner. Otherwise, a tabu period is set to this parameter, so that it is not selected during a certain period. When any parameter cannot be modi ed, the procedure is terminated.

1. Set initial values of parameters in Section 3.2, initialize their tabu periods by 0, and calculate the initial value of the cost function E.
2. Randomly select a parameter whose tabu period is 0.
3. Calculate the cost function values E_+ and E when a xed variation is added and deleted to the parameter respectively. Note that this variation is set 1 for eX, eY, rx, ry, and e, 0.2 for a, and 0.1 for b, c, d, and r.
4. Select the minimum one among E, E_+, and E . If E_+ or E is selected, update the parameter value and the cost function to the corresponding ones. Otherwise, set the tabu period (6 in simulations) to the parameter.
5. Terminate the procedure if any parameter cannot be modi ed.
6. Decrease the tabu period by 1, and go to step 2.

In addition, the following *hill-climbing procedure* is applied at the convergence, in order to further improve the solution by changing the value of a randomly selected parameter and repeating the local search.

1. Randomly select a parameter.
2. Increase or decrease it with the xed variation, where either one is selected randomly.

3. Apply the above local search procedure.

4. If the cost function is reduced, keep this new cost function and parameter values. Otherwise, retrieve old ones.

5. Repeat step 1 ∼ step 4 in 5 times.

Here, we note that our local search algorithm does not need the cost function to be di erentiable by variables to be optimized unlike the gradient descent method. Besides, the sequential update of parameters guarantees the convergence to a local minimum, while the hill-climbing procedure avoids the convergence to poor local minima.

4 Accuracy Evaluation by Sample Images

For accuracy evaluations of eye contours extracted by our algorithm, the di erence on four feature points of each eye contour (total eight points for one person) between the algorithmic extraction and the manual extraction is measured in eight student face images that were taken in our laboratory using a conventional digital camera. The four points are top, bottom, leftmost, and rightmost points of the eye contour. Table 1 summarizes the average di erence, the average of the maximum di erence of eight points in each person, and the standard deviations (SD), in addition to the average computation time on Pentium IV 1.7 GHz. The average di erence on the feature points of eye contours is less than 0.5mm, which we believe is the su cient accuracy. Fig. 2 illustrates an example of extracted eye contours.

Table 1: Accuracy and CPU time in sample images.

accuracy			time
average	max	SD	
0.326mm	0.469mm	0.277mm	25.08s

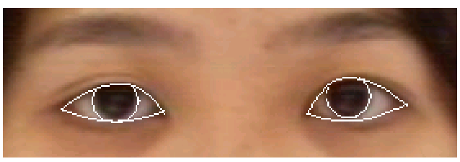


Figure 2: Example of extracted eye contours.

5 Eye Shape Indices in HOIP Database

As an application of our algorithm, we de ne two indices on the eye shape characteristic, namely the *roundness* and the *outer curve*, by using the template, and investigate their distributions among 50 males and 50 females of 15 - 40 years old in the HOIP database. The *roundness* is de ned by $(b + c)/a$ in template parameters. The *outer curve* is de ne by the coe cient of the second-order term $\frac{b}{(a\ d)^2}$ in the function $F2$, because the outer part of an eye including eyelash is often made up in females. Figs. 3 and 4 show their distributions respectively, where the F-test veri es the di erence between males and females in both cases. In the present situation, we cannot answer the question whether this di erence comes from nature or from the influence of eye makeup. It is said that eye makeup such as eyeliner, eyelash curling, and eye shadow can a ect the visual eye shape. We will investigate how eye makeup influences eye shapes quantitatively so that our algorithm can be used in an eye make diagnosis system. In addition, the eye shape check after the eye surgery will also be an interesting application. Thus, we believe a number of useful applications exist for this eye-contour extraction algorithm.

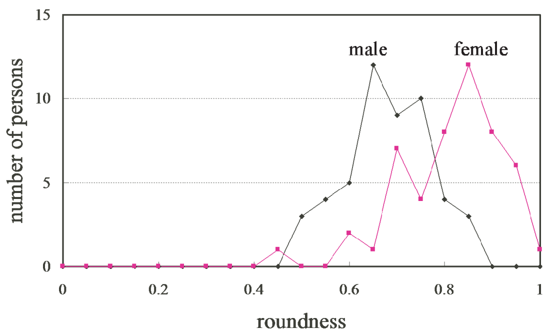


Figure 3: Roundness of left eyes.

6 Conclusion

This paper presented an eye contour extraction algorithm using the deformable template matching method, with the template composed of three quadratic functions and one circle, and its application to HOIP face image database with two eye shape feature indices. In future studies, our algorithm will be applied to non-Japanese face images.

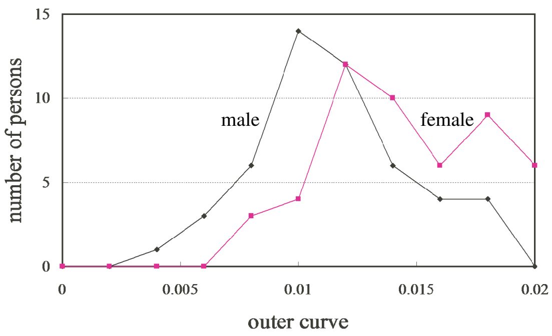


Figure 4: Outer curve of left eyes.

References

[1] R. Brunelli and T. Poggio, "Face recognition: features versus templates," IEEE Trans. Patt. Anal. Machine Intell., vol.15, no.10, pp.1042–1052, 1993.

[2] M. Nagano, M. Yamamoto, K. Suzuki and A. Ohuchi, "A face identi cation system based on the Mahalanobis-Taguchi system," Int. Trans. Operational Res., vol.8, pp.31–45, 2001.

[3] A. K. Jain, Y. Zhong and S. Lakshmanan, "Object matching using deformable templates," IEEE Trans. Patt. Anal. Machine Intell., vol.18, no.3, pp.267–278, 1996.

[4] L. H. Staib and J. S. Duncan, "Boundary nding with parametrically deformable models," IEEE Trans. Patt. Anal. Machine Intell., vol.14, no.11, pp.1061–1075, 1992.

[5] A. L. Yuille, P. W. Hallinan and D. S. Cohen, "Feature extraction from faces using deformable templates," Int. J. Computer Vision, vol.8, pp.99–111, 1992.

[6] M. -P. D. Jolly, S. Lakshmanan and A. K. Jain, "Vehicle segmentation and classi cation using deformable templates," IEEE Trans. Patt. Anal. Machine Intell., vol.18, no.3, pp.293–308, 1996.

[7] G. Park, R. Mersereau and M. J. T. Smith, "Initialization of deformable templates using weighted Gaussian approximations," Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process., pp.2231–2234, 2000.

[8] K. Kantapanit, P. Inrawongs, W. Wiriyasuttiwong and R. Kantapanit, "Dental caries lesions detection using deformable templates," Proc. IEEE Int. Symp. Circuit. Systems,pp.125–128, 2001.

[9] J. Canny, "A computational approach to edge detection," IEEE Trans. Patt. Anal. Machine Intell., vol.8, no.6, pp.679–698, 1986.

[10] Z. Michalewics and D. B. Fogel, "How to solve it: modern heuristics," Springe, 2002.