ROTATION-INVARIANT FACE DETECTION IN GRAYSCALE IMAGES

W. Zhang [a], Y. T. Chan [b]

weizhang@ee.cuhk.edu.hk, chan-yt@rmc.ca
[a] Department of Electronic Engineering, Chinese University of Hong Kong
[b] Department of Electrical and Computer Engineering, Royal Military College of Canada

ABSTRACT

This paper presents a two-stage, feature-based approach to face detection on grayscale images. It can detect in-plane rotated human faces in complex backgrounds. In the first stage, the Canny edges of an image are turned into smooth curves, and then merged to form ellipse-like face candidates. This avoids the standard exhaustive search in the image pyramid. In the second stage, the detector verifies the face candidates by checking the horizontal and vertical projections and then comparing the binarized face candidates against a set of binary face models. Once the face is detected, the facial features are also located. Experimental results are provided to illustrate the performance of this new detector.

1. INTRODUCTION

Face detection and recognition problems have received more and more attention due to the potential applications in, for example, surveillance, video conference, image and video indexing, etc, with many of which needing real-time face detection. Most traditional face detection algorithms use auxiliary information, such as image difference in video sequences, or pixel color in color images to achieve detection at high frame rates. For grayscale images, a detector needs to search the Gaussian Pyramids of the images exhaustively [1] because of the difficulty in segmenting the face-like regions from the images. If the faces in the images are allowed to be rotated, the detector has to determine the face orientation, which is another difficult task. Joen, *et al* [2] proposed a model-based clustering algorithm to detect the rotated faces in the rotated set of CMU test images and reported a better result than the original detector [3]. Later, this group proposed another method based on the 1st-order reduced Coulomb energy (RCE) classifier and reported a 91% detection rate without false alarms on the CMU rotated set [4]. Although the detection rate is high, this detection approach is not suitable for real-time applications due to its excessive computational requirement. Rather than searching exhaustively in the image pyramid, Maio and Maltoni [5] proposed a two-stage face detection method consisting of a generalized Hough transform and a set of directional image templates. The proposed face detector is claimed to be real-time, but their method is limited to locating a single near-upright frontal face in the image because of the limitations of the Hough transform [6].

In this paper, we present a new face detector which locates frontal views of human faces with arbitrary in-plane rotation in complex scenes. The detector first locates the face candidates by conditioning the Canny edges of the input image. It does not require exhaustive searches in the image pyramid. It then verifies the face candidates by comparing them to binary face models. Due to simplicity at the algorithm level, this detector has a high potential for real-time implementation.

2. OVERVIEW OF THE DETECTOR

Figure 1 shows the structure of the face detector. The edge map is obtained by the standard Canny edge detector. After breaking the “cross points” and the “sharp corners” in the edge map, we obtain only the smooth curves. Then these smooth curves are merged according to the smoothness of the new connections and the curvature of the newly merged curve. An ellipse is fitted on each merged curve. If the fitting error is small, the image content bounded by the fitted ellipse is extracted from the input image to form a face candidate. Finally, the candidates are verified with several binary face models.

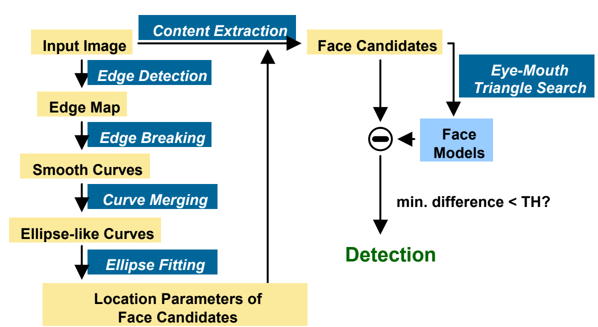

Figure 1. The structure of the detector

3. AN EDGE MERGING ALGORITHM

The standard Canny edge detector [7] can detect very weak edges and provide sufficient information for locating face candidates. Figure 2a shows the Canny edges of the image “cast1” in the CMU rotated image set.

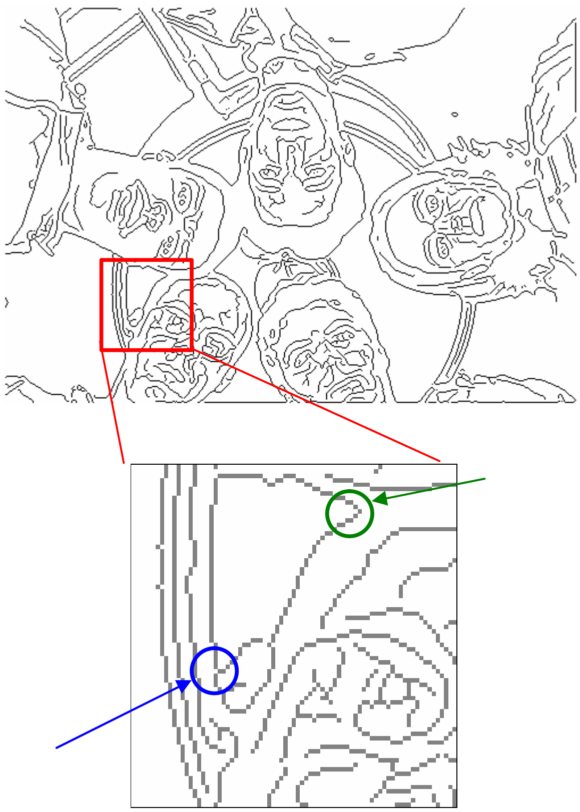

Figure 2a. Canny edges of the input image
2b. Example of “cross point” and “sharp corner”

In the Canny edge map, a pixel that has more than two neighbors in 8-connectivity is defined as a “cross point”, as indicated by the left arrow in Figure 2b. The “sharp corner”, as indicated by the right arrow in Figure 2b, can be detected by analyzing the chain code differences [8] of the edge within a certain range of the neighborhood. After breaking the “cross points” and “sharp corners”, we eliminate the relatively short curves. The resulting smooth curves are shown in Figure 3.

The *Curvature* (or circularity) of a curve is defined as $4pA/P^2$, where A is the area bounded by the curve and the line joining its endpoints, and P is the perimeter of this area. The Curve Merging process begins with the long curves whose curvatures exceed a predefined threshold. Figure 4 illustrates a merging process in the area bounded by the rectangle in Figure 3.

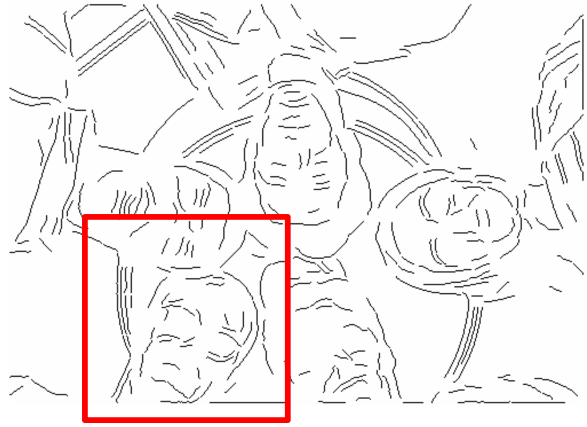

Figure 3. Smooth curves

Figure 4. a b c d e f Curve Merging process

In each iterative step of the Curve Merging process, a Search Region is determined according to the current curve length and the tangent direction at both endpoints of the current curve, as indicated by the dark regions in Figure 4 a to e. Only the curves inside the Search Region will be considered for merging with the current curve. As the curvature of a curve increases, the size of the Search Region decreases. Thus, as the merging process goes on, the number of curves to be analyzed decreases rapidly. The criteria of choosing the correct curve to be merged include the smoothness of the intermediate merged curve and its curvature. After the merging process, an ellipse is fitted for this curve (Figure 4f). Rather than storing all the coordinates of the points of the merged curve, ellipse fitting can reduce the data storage, since only 5 parameters (2 for ellipse center, 1 for major axis, 1 for minor axis and 1 for rotation angle) are needed to determine the location of a face candidate. A fitted ellipse is taken to be a face candidate if the fitting error is below a threshold. Figure 5 shows the face candidates of the image "cast 1".

Figure 5. Face candidates

4. A FAST FACE VERIFIER

To compensate for the small deviations between the outline of a face candidate and the true face, a parameter fine tuning process is performed on each face candidate. In this process, we do a symmetry test on the pixels inside the fitted ellipse at a total of 39 orientations of the original ellipse. These orientations are, for the minor axis shift, at $0.1a$, 0, $-0.1a$, and for the rotation, at $\theta \pm 18^{o}$ at 3^{o} steps, where a is the minor axis length and θ is the rotation angle of the original ellipse. For each orientation, a lighting correction is performed prior to the symmetry test. The orientation that has the largest symmetry is chosen for the next face verification step. Figure 6 shows the contents of a face candidate after parameter fine tuning and boundary removal.

Before applying the binary face model onto the face candidate, the Face Verifier searches for the minimum enclosure (or box) of all the facial features, including eyebrows, eyes, nose and mouth. It is necessary to align this box properly with respect to the center of the ellipse (face candidate). To determine the box width and its horizontal alignment, firstly, the face candidate is lighting corrected [1] and histogram equalized to improve the contrast. Secondly, the horizontal edges are strengthened (Figure 7a). Thirdly, the edge strengthened version of the face candidate is converted into a binary image (Figure 7b), where the threshold is set to be 0.25/1, i.e., any pixel intensity in the range 0 to 1 is set to 1 of it is over 0.25 and 0 if below. Finally, the sum of the pixel values is projected onto the horizontal axis (Figure 7c). The box width is taken as the range of the non-zero part of this projection.

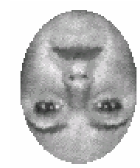

Figure 6. Image contents extracted from a face candidate after symmetry test

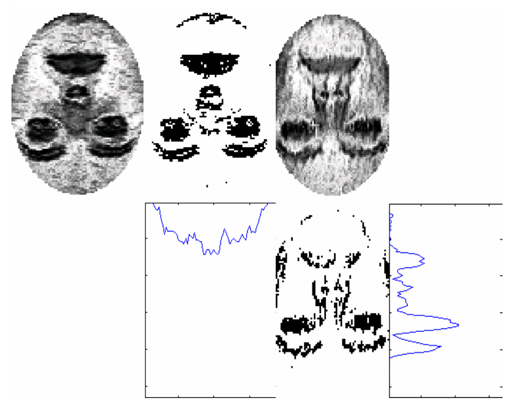

Figure 7. a b d
c e f
Horizontal and Vertical Projections after edge strengthened

Similarly, we strengthen the vertical edges (Figure 7d), binarize the image (Figure 7e), and project onto the vertical axis (Figure 7f). To determine the box height and its vertical alignment, we first note from experimental results that the box is close to a square. Hence the height of the box is selected to be equal to the width. If the height turns out to be smaller, then the box is not square. But this happens only rarely. More generally, the height is larger than the width. We then truncate the height by taking the segment that contains the largest area under the projection curve in Figure 7f.

Locating the box using the projection information is efficient and also makes it feasible to compare a binarized face candidate with a binary face model by a simple bit-wise AND operation. Figure 8 shows some examples of the detected boxes. By analyzing the geometrical alignments of the dark regions in the binarized box, we can detect the triangles formed by the eyes and the mouth, or the eyebrows and the mouth (Figure 8). The detected triangle can increase the confidence of detection and can also help locate the facial features. Figure 9 shows four binary face models that are currently used by the Face Verifier to compare against the box.

Figure 8. Examples of eye-mouth triangles in the boxes

Figure 9. Binary face models

Figure 10. Sample detection results on CMU images

5. EXPERIMENTAL RESULTS

Some detection results are shown in Figure 10 to illustrate the performance of this face detector. The proposed face detector has been tested using part of the test images in CMU rotated set, which consists of 49 images, containing 86 faces of arbitrary scales and in-plane rotations. By carefully selecting the thresholds, the proposed approach could detect 80 correct faces with no false alarms, which yields a detection rate of 93.0% in this small database.

The double thresholds (TH_{low} and TH_{high}) used in the Canny edge detector are the most significant thresholds in this approach. Currently TH_{high} is selected to produce 30% of the total number of pixels in an image as edge pixels, and TH_{low} is $0.4TH_{high}$. The produced edges are sufficient for the edge merging algorithm to locate the correct face candidates except for some extreme cases such as the two misses shown in Figure 10, where the faces are wearing strange headgears and under very bad illumination conditions. We could reduce TH_{low} to produce more weak edges for edge merging, but this will also bring more false face candidates hence making the face verification process more difficult and time-consuming. More binary face models could be constructed to cover faces that have glasses or beards.

6. CONCLUSION

This work proposes a two-stage feature-based approach to detect the in-plane rotated faces on grayscale images with complex backgrounds. In the first stage, edges from the Canny edge detector are conditioned and then merged. Ellipses are next fitted to the merged curves. Those whose fitting errors are below a predetermined threshold are then passed to the face verification stage. This stage matches binary face models to the face candidates for detection.

The proposed method avoids the need for exhaustive searches in the image pyramid. The algorithms are simple and the computational requirement is low. Hence the detector has a high potential for real-time applications.

REFERENCES

[1] B. K. Low and E. Hjelmas, “Face detection: a survey,” *Computer Vision and Image Understanding 83*, pp. 236-274, 2001.

[2] B. H. Jeon, K. M. Lee, and S. U. Lee, “Rotation invariant face detection using a model-based clustering algorithm,” *IEEE Int’l Conference on Multimedia and Expo*, pp. 1149-1152, 2000.

[3] H. A. Rowley, S. Baluja, and T. Kanade, “Rotation invariant neural network-based face detection,” *Proc. of IEEE Conference on Comp. Vision and Pattern Recog.*, pp. 38-44, 1998.

[4] B. H. Jeon, K. M. Lee, and S. U. Lee, “Face detection using the 1st-order RCE classifier,” *IEEE Int’l Conference on Image Processing*, vol. 2, pp. 125-128, 2002.

[5] D. Maio and D. Maltoni, “Real-time face location on gray-scale static images,” *Pattern Recog. 33,* pp. 1525-1539, 2000.

[6] T. M. Breuel, “Recognition by adaptive subdivision of transformation space: practical experiences and comparison with the Hough transform,” *IEE Colloquium on ‘Hough Transforms’*, pp. 7/1-7/4, 1993.

[7] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-698. 1986.

[8] H. Freeman, “Computer processing of line-drawing images,” *Computing Surveys*, vol. 6, no. 1, pp. 57-97, 1974.