Detection of Face Direction by Implementing Face Edge Patterns

Jarjees A. Khidir

College of Science Education

University of Salahaddin

Received on:24/6/2008 Accepted on:17/3/2009

ABSTRACT

In this paper, a method for detecting the direction of a human face is developed; regardless of its age or sex. The method involves creating a set of five face patterns representing the front, up, down, left, and right directions of a face. The face patterns are produced by applying Canny’s edge detection algorithm on some face files. The direction of the input face is found by first applying the above algorithm on the input file and comparing it with the five face patterns. The face pattern that gives minimum difference will represent the direction of the input face. Excellent results were reported when applied on images with relatively clear background and the head were centered at the image area.

Keywords: Detection, Face Direction, Face Edge Patterns.

الكشف عن اتجاه الوجه من خلال تنفيذ أنماط حافة الوجه

جرجيس خضر

كلية علوم التربية، جامعة صلاح الدين

تاريخ الاستلام: 200/06/24 تاريخ القبول: 2009/03/17

الكلمات المفتاحية: الكشف عن اتجاه الوجه، أنماط حافة الوجه.

1. Introduction

The interface between people and computers has developed over the years from the early days of switches and LEDs to punched cards, interactive command-line interfaces, and the direct manipulation style of graphical user interfaces (GUI). The “desktop metaphor” of graphical user interfaces, also known as WIMP interfaces (for Windows, Icons, Menus, and Pointing devices), has been the standard interface between people and computers for many years. Of course, software and technology for human-computer interaction (HCI) is not isolated from other aspects of computing. In addition, there are many non-GUI (or “post-WIMP”) technologies, such as virtual reality, speech recognition, computer vision, and spatial sound, which promise to change the status quo in computer-human interaction [1].

Human face detection plays an important role in many applications such as intelligent human computer interface (HCI), biometric identification, and face recognition. The goal of any face detection technique is to identify the face regions within a given image. The reliable detection of faces has been an ongoing research topic for decades [2].

Current face detection methods can be categorized into image-based methods and feature-based methods. Image-based methods use the intensity array as template to match the standardized candidate regions. Feature-based methods detect facial features using the skin color, the specific patterns of various facial features, and their relative positions [3].

2. Human face tracking

Recently, there has been a great deal of interest and progress in the analysis of faces. Potential applications are many; including recognition, gesture recognition, face-spotting in images, gaze detection and HCI [4].

Of the various vision-based interaction (VBI) technologies, human face tracking (HFT) is perhaps the most useful, as it can be used to support several other technologies as well as be used directly by various applications. Face tracking can serve as input to various other VBI modules, such as dynamic face recognition, facial expression analysis, audio-visual speech processing, body tracking and modeling, gesture recognition and activity analysis. The location, pose, and expression of the face are the keys to both extracting and interpreting human body information. Since faces are arguably the most stable and identifiable component of the body under various transformations, the face can serve as an anchor from which to relate other VBI tasks. Additionally, an understanding of face pose or expression can be vital to interpret high-level information such as facial identity or body gesture [5].

Estimation of face direction is one of the most important technologies for useful man-machine interface. For example, Ballard and Stockman proposed the system that facial aspect is used to control the cursor. For the purpose of tele-conference, facial movement estimation has been studied to send codes at high speed. Human gaze estimation also needs the information of face direction. In most cases, the estimation of face direction requires a lot of time, or needs the information about the position of features [6].

2.1 Face tracking review

In 1997, Takuya Minagawa, *et al.*, developed a system which can estimate the face-direction of a person with two cameras using stereo vision. In this system, the features of the face, the eyes and the mouth, are extracted in each image to measure their 3D positions. Then, face direction is determined by the 3D coordinates of the extracted features [6].

In 1999, Vera Bakic and George Stockman, designed a real-time program that detects the face features of a workstation user for controlling a computer using head movements and gaze direction. Based on the face pose, it determines where on the display the subject is looking. Button selection is done by opening the mouth [4].

In 2006, R. Shikata , *et al.*, designed a new meeting support system that helps the hearing impaired to understand the contents of a meeting. The proposed system distinguishes the mainstream of the discussion from other chattering based on the utterances. The situation of the meeting is acquired as a captured image using an omni-directional vision sensor. The system then detects the speaker from the captured image by using face directions of the participants. As a method to detect face direction, the system utilizes Sobel edge detection filter [7].

3. Detection of face direction

To detect the face direction of an input image file, a program using *MATLAB V7.0* language is made. The program reads an input face file which is a color JPG image with the dimensions of 300x300 pixels containing a human face of any age or sex and outputs a statement telling the face direction. The face size should be occupying at lease 60% of the whole image space; (*Fig-1a*). The input face file is converted to gray-scale (intensity) and stored in a matrix of 300x300; (*Fig-1b*).

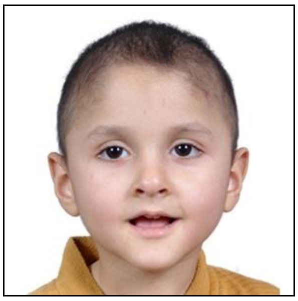


Figure 1a: The input face (Dyar.jpg)

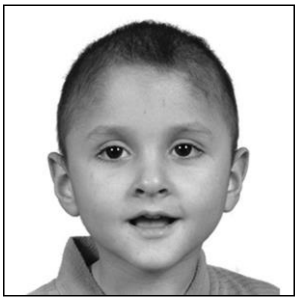


Figure 1b: The gray-scale face

Then, Canny’s method for edge detection is applied on the gray-scale matrix to produce an edge matrix of 300x300 of 1’s and 0’s; (*Eq-1* and *Fig-2a*).

EM* = edge (*IM* , 'canny' , [0.1 0.15])** ***…Equation (1)

Where: ***EM*** is the edge matrix of the input face
IM is the intensity (gray-scale) matrix of the input face

By experimenting various threshold values on Canny’s method to get the suitable amount of edges from the gray-scale face, a low threshold=0.1 and a high threshold=0.15 is recommended. These two thresholds are then used to detect strong and weak edges, and to include the weak edges in the output only if they are connected to strong edges. Canny’s edge detection method is preferred rather than other methods (like Sobel, Prewitt, Roberts, Laplacian of Gaussian, and Zero-Cross methods) because it is less likely than the others to be fooled by noise, and more likely to detect true weak edges [8].

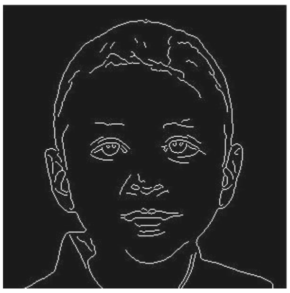


Figure 2a: The edge face

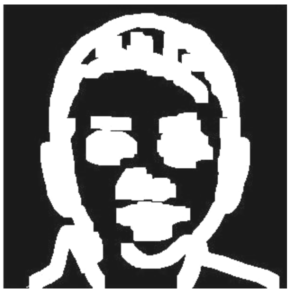


Figure 2b: The Thickened edge face

The edge matrix of the input face is then compared to a set of five, already created, face edge patterns. Because the edge matrix; as well as the five face edge patterns; consist of 1's and 0's, the comparisons are done by intersecting the edge matrix with each face pattern and counting the number of non-intersected 1's. In order to increase the probability of intersection and then similarity, the edges of the edge matrix are thickened by 6 pixels in each direction; (*Eq-2* and *Fig-2b*):

$$\boldsymbol{EM}[\boldsymbol{i}-6:\boldsymbol{i}+6,\boldsymbol{j}-6:\boldsymbol{j}+6]=\begin{bmatrix}1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1&1&1\end{bmatrix}$$

…Equation (2)

For each pixel value ***EM(i , j)***=1 ; Where: ***EM*** is the edge matrix of the input face.

The XOR operation is used to perform the intersection by applying it on each corresponding pixels; (*Eq-3*):

$$\boldsymbol{CR}=\sum_{\boldsymbol{i}=1}^{300}\sum_{\boldsymbol{j}=1}^{300}\left(\boldsymbol{EM}(\boldsymbol{i},\boldsymbol{j})\oplus\boldsymbol{EP}(\boldsymbol{i},\boldsymbol{j})\right)$$ ***… Equation (3)***

Where: ***CR*** is the comparison result
EM is the edge matrix of the input face
EP is a face edge pattern matrix

The face edge patterns represent the five basic face directions which are front, up, down, left, and right. The face edge pattern ***EP*** that produces the minimum difference ***CR*** to the edge matrix of the input face ***EM*** will represent the direction of the input face.

3.1 Creating face edge patterns

The position of the area containing the eyes, the nose, and the mouth within an image can specify which direction the face is looking at. Depending on that, five face direction patterns using some face files are created in five directions; front, up, down, left, and right using SAMSUNG DIGIMAX 360 digital camera; (*Fig-3*).

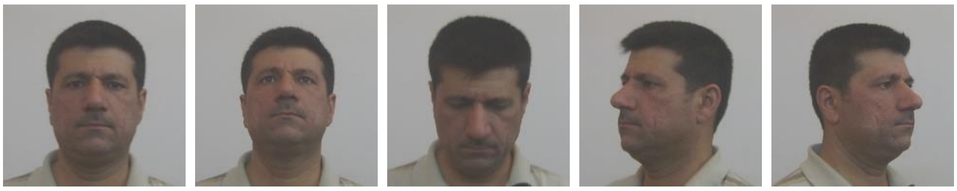


Figure 3: The five directions photos

Adobe Photoshop is used to add some donut shapes to these photos to give weight and density pixels to the eyes, nose, and mouth areas. These colored photos are then converted to gray-scale matrices; (*Fig-4*).

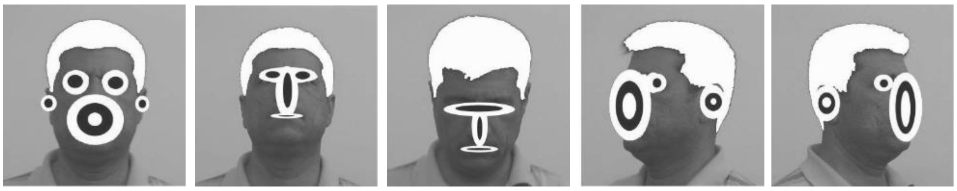


Figure 4: The photos with added shapes in gray-scale

Canny's edge detection is applied then on the resulting gray-scale photos of (*Fig-4*) using the same low and high thresholds as used in the input face edge; (*Eq-1*). The results are stored in five edge matrices. Finally, the thickened mask processing of (*Eq-2*) is applied on the five edge matrices to produce the required set of five face edge patterns; (*Fig-5*).

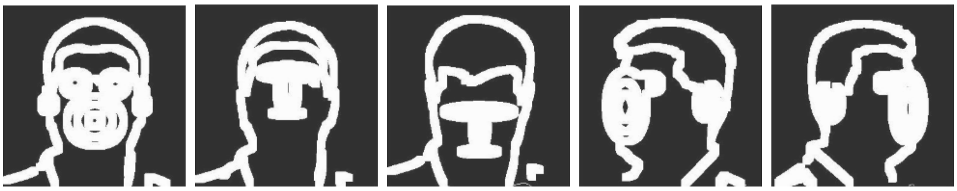
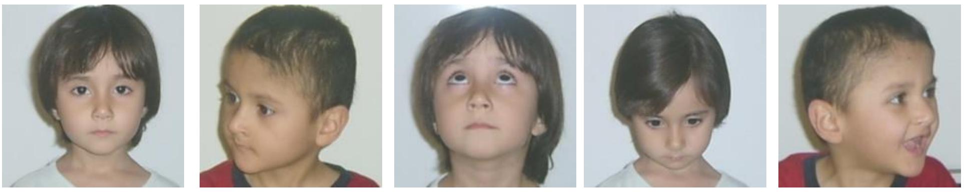


Figure 5: The resulting face edge patterns

3.2 Testing and Results

To test the method, several photos for many kids ('Dyar' and 'Sara') with various face directions using the mentioned camera are applied to the program and the results were excellent; (*Fig-6*).



Face direction is Front Face direction is Left Face direction is Up Face direction is Down Face direction is Right

Figure 6: Tested photos with their results

Note: The program considers the right and left directions to be relative to the viewer's point of view.

Then, randomly selected 31 face photos are applied (with dimensions 300x300); (*Fig-7*).

Figure 7: Some downloaded photos

The program successfully detected the direction of 27 faces and failed for only 4 of them. If we added the 10 face directions of Dyar and Sara that were successfully detected, the result will be 37 successful detections. This makes the percentage of successful detection about 90% and failed detection about 10% of the cases.

4. Summary and Conclusions

In this paper, a method for detecting face direction has been developed. A program is designed to input a colored face image and modify it by some edge managements. A set of five face patterns is used for comparison to output the direction of the input face. Detecting the face direction depended on the position of the eyes, nose, and mouth. Canny's method for edge detection is used to find the edges of the face.

We conclude that the face direction can be detected by carefully preparing some set of face edge patterns to be compared with the input image. By increasing the number of face patterns representing various directions, more accurate direction detection; and even an estimated direction angle can be measured. Also, better detection is done on images with relative clear background because a complex background would cause increased edges that affect the comparison results.

REFERENCES

[1] A. Van Dam, “Post-WIMP user interfaces: the Human Connection”, Communications of the ACM, Vol. 40, No. 2, pp. 63-67, Feb. 1997.

[2] Li Tao, Ming-Jung Seow, and Vijayan K. Asari, “Nonlinear Image Enhancement to Improve Face Detection in Complex Lighting Environment”, International Journal of Computational Intelligence Research. ISSN 0973-1873 Vol. 2, No. 4, pp. 327-336, 2006.

[3] Yi Zhu and Florin Cutu, “Face Detection using Half-Face Templates”, Journal of Vision, ISSN 1534-7362 Vol. 3, No. 9, pp. 839, October 22, 2003.

[4] Vera Bakic and George Stockman, “Menu Selection by Facial Aspect”, Proceedings of Vision Interface 99, Trois Rivieres, Quebec, CAN (19-21 May 1999).

[5] Matthew Turk, Changbo Hu, Rogerio Feris, Farshid Lashkari, Andy Beall, “TLA Based Face Tracking”, Proceedings International Conference on Vision Interface (VI’2002), pages 229–235, Calgary, May 2002.

[6] Takuya Minagawa, Hideo Saito, Shinji Ozawa, “Face-Direction Estimating System Using Stereo Vision”, 23rd International Conference on Industrial Electronics, Control and Instrumentation, 1997 (IECON 97), 9-14 Nov 1997, On page(s): 1454-1459 vol. 3.

[7] R. Shikata, T. Kuroda, Y. Tabata, Y. Manabe and K. Chihara, “Development of vision based meeting support system for hearing impaired”, Proceedings of the 6th International Conference on Disability, Virtual Reality and Associated Technologies, pp. 53-58, Esbjerg, Denmark, 2006.

[8] MATLAB V7.0 Product Help.