

A Computer Interface for the Disabled by Using Real-Time Face Recognition

Cheng-Yao Chen, and Jyh-Horng Chen
Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

Abstract-In this paper we present a computer interface for the disabled by using real-time face recognition algorithm. Based on an adaptive color model, the system allows complex backgrounds (including objects and onlookers) and varying illuminations. Our refined control technique is also robust when suffering the user position shifts. Experiment shows that our system can significantly increase the efficiency for the disabled to deal with internet information and multimedia entertainment.

Keywords- Computer interface, the disabled, real-time, face recognition

I. INTRODUCTION

An efficient computer interface for the disabled has long been thought as a crucial topic in the field of medical electronics. Professionals around the world have proposed many solutions. However, most of them require wearing extra instruments, such as infrared appliance, headset with camera, artificial feature attachment, and etc. Thanks to the progresses in the field of face recognition [1] and computer hardware, we build up a man-machine interface by simply using a USB digital PC camera (Fig. 1.). The disabled can not only be trained to use it easily and comfortably, but also enrich their lives by the information exchanged through the internet and multimedia entertainment.

Our system can be functionally divided into two stages: face recognition and mouse/action control (Fig. 2.). We discuss each block in details in the following paragraph.

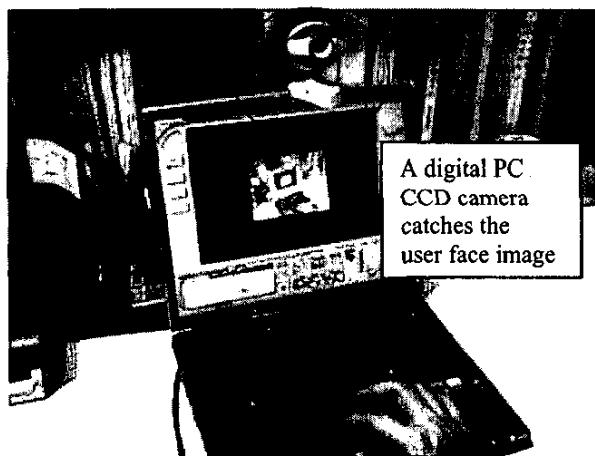


Fig. 1. System Overview

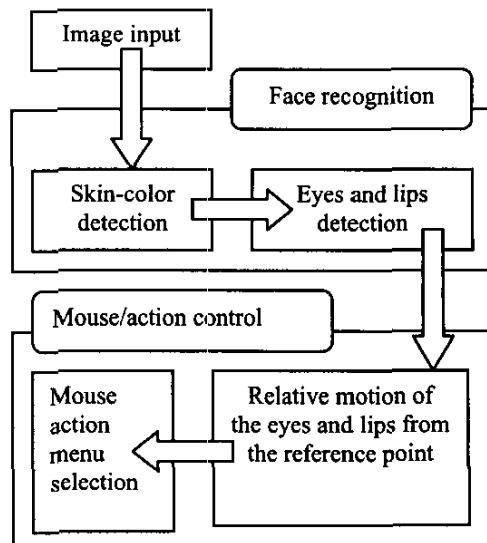


Fig. 2. System block diagram

II. METHODOLOGY

RGB model is natural to human visual perception, but because of its intensity inherited property it is not suitable for computer vision. When the illumination changes, the skin color will cause a non-linear effect in the RGB representation. As a result, we use the normalized RGB (range from 0 to 1), and transfer it to the HSI (hue, saturation and intensity) color system [2,3] which observes the following equations:

$$H = \begin{cases} \theta & \text{if } B \geq G \\ 360^\circ - \theta & \text{if } B < G \end{cases} \quad (1)$$

where

$$\theta = \cos^{-1} \left\{ \frac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)(R-G)+(R-B)(G-B)}} \right\} \quad (2)$$

and

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \quad (3)$$

and

$$I = \frac{R + G + B}{3} \quad (4)$$

HSI model can significantly remove the white illuminant effect on skin color. However, in order to remove the non-white illuminant effects, we add an adaptive shifting color factor in the detection. Thus,

$$F(x, y) = \begin{cases} 1 & \text{if } -R_{H,S} \leq R_{H,S}(x, y) \leq R_{H,S} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $F(x, y)$ represents the possible face point candidate, and $R_{H,S}$ represents the skin color cluster with non-white illumination compensation updating with time.

After defining the face point candidates, we use a linking and searching algorithm to determine the boundaries of the face candidates in the window. Here we assume that the user sits nearest to the camera, and would be the largest face region in the window. Also by using the motion estimation, we can block out the other face candidate.

Within the face, we again use the color information to find the lips and check its ellipse shape by pattern comparison and record the last lips location to filter out impossible movement. Since the color of black and white are easily confused by the illumination, we use another technique to find the eyes.

Eyes are surely the most complicated region in the face region especially in the horizon line [4]. Thus, we first apply the edge detector to find all the edges within the face, and then use morphological operation to separate all the candidate clusters. Afterwards, we use the shape matching to find the eye candidates, and check some knowledge-based information (ex. typical eye-to-lip distance, eye separation distance, containing both dark and bright points, and etc.). Also, the motion estimation is used here to decrease the system loading.

In order to avoid user position shifts, we propose a relative motion vector between the moving reference and the face reference to control the computer mouse (Fig. 3). Here, the face reference is defined as the mass center of the face. We observe that with less than 20 degree head rotation, the face mass center would only suffer 5% variation. Thus, we use the temporal low pass filter to avoid the vibration of the face reference. That will also make the mouse cursor more stable. Then we use a weighted center of eyes and lips to serve as a moving reference $M(x, y)$ as follows:

$$M(x, y) = w_1 E_{right}(x, y) + w_1 E_{left}(x, y) + w_2 L(x, y) \quad (6)$$

where $E_{right}(x, y)$, $E_{left}(x, y)$, and $L(x, y)$ represent the iris locations and lips mass center respectively. Moreover, we set a subtle strategy to let the eyes achieve small movement of the mouse and lips achieve the large movement. The mouse speed S can be defined as follows:

$$S = a(M(x, y) - F_{reference}) + b \quad (7)$$

where $F_{reference}$ represents the face reference and a, b are velocity factors changing with the user location. By such linear modification, we can avoid abrupt mouse velocity change when suffering user position shifts.

We set our mouse with single-click as the default action, and we also add double-click, scroll-up, scroll-down, and pause/restart commands into our action menu to facilitate the user. If the mouse pointer holds still for more than 1.5 sec, the action menu will pop up. If no action selected within 1.5 sec, the mouse will single click the target automatically. The above proposed waiting time is changeable to fit different user tolerances.

III. RESULTS

We use a laptop equipped with Intel Pentium⁴ 1.0 GHz CPU, Logitech QuickCam Pro 3000, and Microsoft Windows XPTM to test our system for the disabled. Our system only take up roughly 40% of the windows resource, and thus allow the user to operate more than two applications simultaneously. The results show that the disabled can smoothly use the common applications such as Microsoft Internet Explorer, Windows Media Player, Word, and Excel in the 1024 x 768 screen resolution. When the user wants to type, we adopt the "Screen Keyboard" to serve as the input device.

We also test our system by the other obvious face candidates and complex background (Fig. 4).

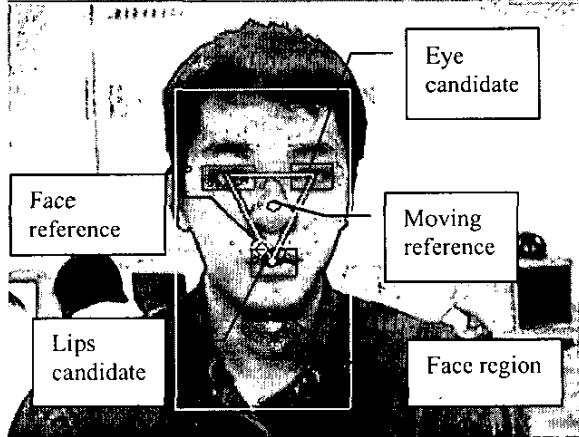


Fig. 3. Control strategy diagram

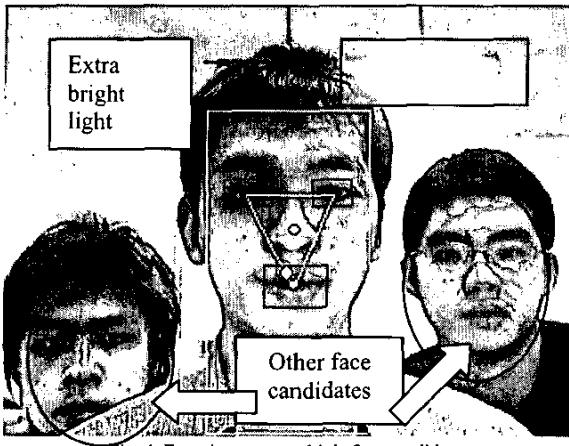


Fig. 4. Experiment on multiple face candidates

The result shows that our system maintains its robustness and precision despite the presence of other faces, but the system performance is decrease due to the computation complexities. Finally, in order to test the partial varying illumination, we put more than two light sources in different directions, and our system still successfully work with roughly 10% wrong decisions (Fig. 5.).

IV. DISCUSSION

When the user suffers from the partial varying illumination, our system is found some failure operations. Such wrong decisions are resulted form the non-linear shifts of skin color and abrupt artificial edges in the face region, we may adopt a high-order adaptive color model to solve this problem.

Every PC CCD camera has its own pre-process procedures, and may somehow deteriorate the performance of our system. If more precision required, we should include the camera characteristic (such as compression loss, white balance, auto gain control, and etc.) into our color model.

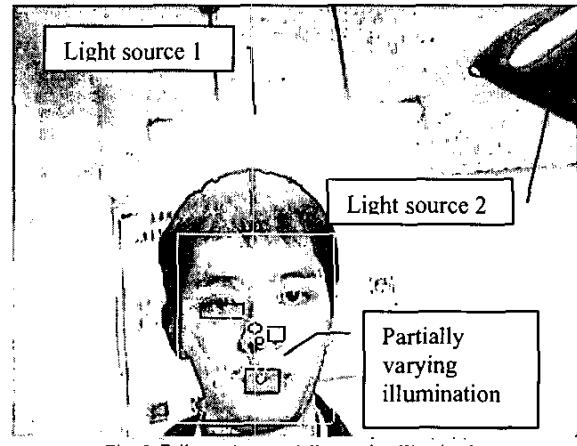


Fig. 5. Failure when partially varying illumination

V. CONCLUSION

A reliable computer interface system for the disabled by using real-time face recognition is presented. Although some failure operations are found because of the partial varying illumination, the robustness in the normal office light condition with complex background is significant. Furthermore, if we incorporate the camera pre-process characteristic into our color model, we can achieve an even higher precision and make it more suitable for the computer control.

REFERENCE

- [1] Erik Hjelmas and Boon Kee Low, "Face Detection: A Survey", *Computer Vision and Image Understanding* 83, 2001, pp. 236-274.
- [2] C.H. Lee, J.S. Kim, and K.H. Park, "Automatic human face location in a complex background", *Patter Recog.* 29, 1996, pp. 1877-1889.
- [3] J. C. Terrillon, M. Shirazi, H. Fukamachi, and S. Akamatsu, "Comparative performance of different skin chrominance models and chrominance spaces for automatic detection of human face in color images", *Proc. 4th IEEE Int. Conf. on Automatic Face and Gesture Recognition*, 2000, pp. 54-61.
- [4] L.C. De Silva, K. Aizawa, and M. Hatori, "Detection and tracking of facial features by using a facial feature model and deformable circular template", *IEICE Trans. Inform. Systems E78-D(9)*, 1995, pp.1195-1207.