Volume 2, Issue 2, February 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Simple Algorithm for using Face as a Pointing Device using OpenCV

V. Pradeep
Assistant Professor
Department of Computer Science and Engineering
JCT College of Engineering and Technology
Coimbatore, India.

Abstract— **Computer has influenced our life in such a way that it is very difficult to sustain without a computer. For physically challenged persons, especially persons without hands and legs, it is impossible to use the computers without an assistive technology. Keyboard and mouse are the most essential input devices to work with a computer. By the use of on-screen keyboard, a pointing input device such as mouse is sufficient to operate a computer with GUI software. The basic actions of a mouse are Mouse Movement and Mouse Button Click. This paper is on developing an assistive technology that replaces the mouse movement by head movement using OpenCV. The Mouse Button Click is implemented by any facial expression such as blinking eye, opening mouth and head movement.**

Keywords— **alternative mouse, assistive technology, hands free computing, gesture recognition, user interface, disabled users.**

I. Introduction

Computers changed the world a lot. It helped man step forward into the future. Thanks to computers, space exploration came true, new designs of vehicles and other transportation were made; entertainment became more entertaining, medical science made more cures for diseases, etc. The computers impacted our lives in many ways. They did make life a lot easier. Without computers, the world would be a harder place to live in [1]. There are many people who find the standard computer input devices - the keyboard and mouse - difficult to use due to a motor disability. The mobility impairment prevents them from moving the mouse or typing on a keyboard. A number of keyboard and mouse configuration options designed to overcome physical difficulties exist so-called adaptive or assistive technology – hardware or software that eliminates barriers to using a computer [2]. The “mobility-impaired” people have trouble using the hardware of their computers rather than understanding or interpreting information [3].

II. Components

This paper brings out an innovative idea to use the camera as an alternative to the mouse. The mouse operations are controlled by the head movement and the eyes-blink that is captured by the camera. The components are shown in the figure 1.

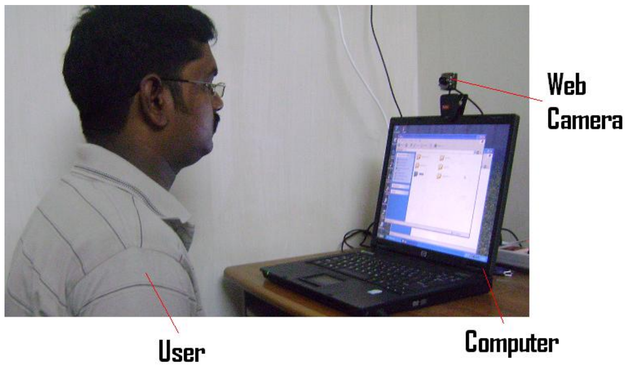

Figure. 1 Components for using Face as a Pointing Device

III. OpenCV

OpenCV is an open source computer vision library designed for computational efficiency with a strong focus on real-time applications. One of OpenCV’s goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly [4]. This paper uses OpenCV library to detect a frontal face in an image using its Haar Cascade Face Detector.

IV. Method

In computing, a cursor is an indicator used to show the position on a computer monitor or other display device that will respond to input from a text input or pointing device [5]. The cursor is moved on the screen by setting the coordinates for the desired position on the screen. The top-left corner is (0,0) and the bottom-right is (X_MAX,Y_MAX). In the proposed system, the cursor is moved left, right, up and down by moving the head right, left, towards the camera and backwards the camera respectively.

V. Algorithm

```
Step 1:  Take a snapshot from the moving picture and
         detect a face using OpenCV haar cascades
Step 2:  Find the dimensions of the face and calculate the
         rectangular points pt1 and pt2
Step 3:  Calculate the initial x-axis mid point
         i_m.x = (|pt1.x - pt2.x|) / 2
         Calculate the initial y-axis length
         i_l.y = |pt1.y - pt2.y|
Step 4:  Take a snapshot from the moving picture and
         detect a face using OpenCV haar cascades
Step 5:  Find the dimensions of the face and calculate the
         rectangular points pt1 and pt2
Step 6:  Calculate the current x-axis mid point
         c_m.x = (|pt1.x - pt2.x|) / 2
         Calculate the current y-axis length
         c_l.y =|pt1.x - pt2.x|
Step 7:  Calculate the difference between initial and
         current x-axis mid point
         m.x = i_m.x - c_m.x
         Calculate the difference between initial and
         current y-axis length
         l.y = i_l.y - c_l.x
Step 8:  Calculate the current position of the mouse
         pos = GetMousePosition();
Step 9:  Calculate the change of mouse position in x-axis
         if ( -3 ≤ m.x ≤ 3)
         // Ignore slight movements of head
            if(-10 ≥ m.x ≥ 10)
            //Ignore unexpected movement of head
                   ch.x = 0
            else
                   ch.x = m.x
Step 10: Calculate the change of mouse position in y-axis
          if ( -3 ≤ l.y ≤ 3)
          // Ignore slight movements of head
          if(-10 ≥ l.y ≥ 10)
          //Ignore unexpected movement of head
                   ch.y = 0
         else
                   ch.y = l.y
Step 11: Move the mouse pointer to the new position
         (pos.x, pos.y) = (pos.x + ch.x , pos.y + ch.y)
Step 12: Ignore if values crosses the boundary
         if (pos.x >X_MAX)
                   pos.x= X_MAX
         if (pos.x < X_MIN)
                   pos.x= X_MIN
         if (pos.y >Y_MAX)
                   pos.y= Y_MAX
         if (pos.y < Y_MIN)
                   pos.y= Y_MIN

Step 13: Update the initial values
         i_m.x = ch.x
         i_l.y = ch.y
         Go to Step 4.
```

VI. Snapshots

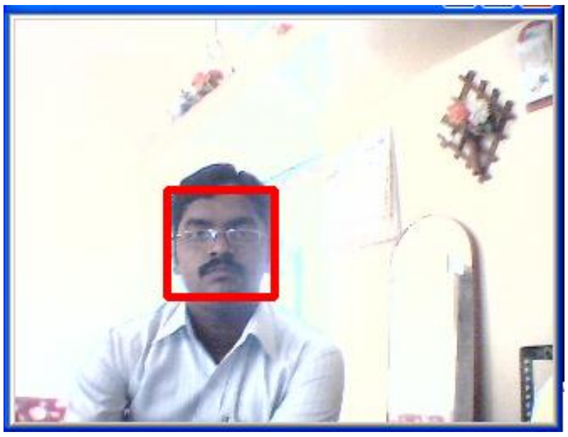
Figure. 2 Moving Face behind to move the cursor down

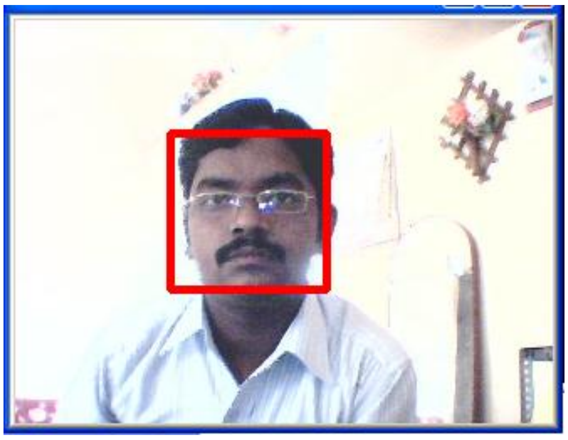
Figure. 3 Moving Face front to move the cursor up

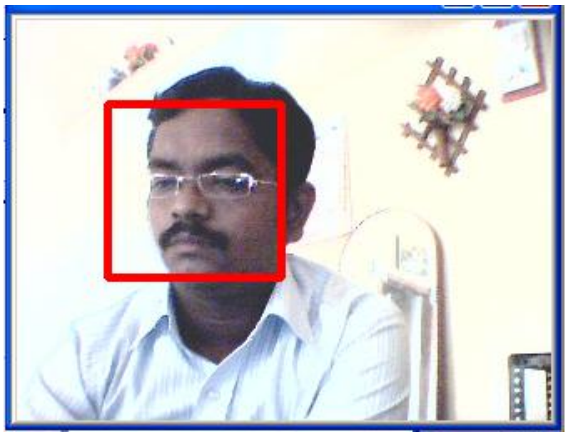
Figure. 4 Moving Face right to move the cursor left

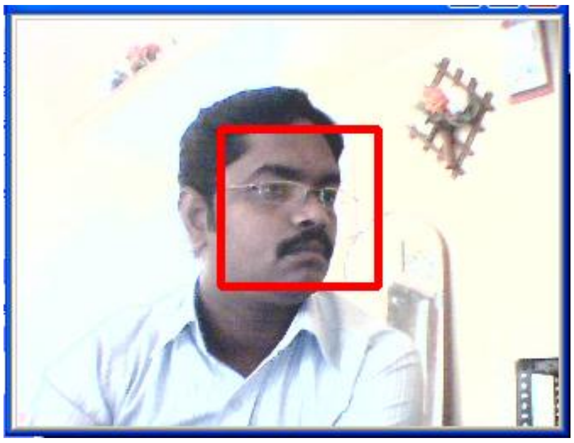

Figure. 5 Moving Face left to move the cursor right

VII. CONCLUSION

To the person who only knows how a hammer works, every problem is some kind of nail. This is to say, getting the right tool can make the difference between a successful task and a smashed-up effort. Since you can't always get it all done with a blunt object, it pays to know what else is available [6]. This paper is an innovative idea to provide the advantage of mouse GUI operations available to the "mobility impaired" people.

REFERENCES

[1] http://tatooine.fortunecity.com/vonnegut/320/history.htm.

[2] Trewin, S. and Pain, H. 1999. Keyboard and mouse errors due to motor disabilities. Int. J. Human-Comput. Stud. 50, 2, 109–144.

[3] http://joeclark.org/book/sashay/serialization/Chapter03 .html.

[4] Gary Bradski and Adrian Kaehler. Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media, 2008.

[5] http://en.wikipedia.org/wiki/Cursor_(computers)

[6] http://www.christopherreeve.org/site/c.mtKZKgMWKwG/b. 4453181/k.7884/Assistive_Technology.htm