A Method for Controlling Mouse Movement using a Real Time Camera

Nitin Sakhare & Prof. Pratyoosh Rai

nitinsakhare@live.com; Pratyoosh.rai@yahoo.co.in

Department of Computer Science and Engineering Bhabha Engineering Research Institute, Bhopal

Abstract

This paper introduces another methodology for controlling mouse development utilizing an ongoing camera. Most existing methodologies include changing mouse parts, for example, including more catches or changing the following's position ball. Rather, we propose to change the equipment outline. Our system is to utilize a camera and PC vision innovation, for example, picture division and motion acknowledgment, to control mouse errands (left and right clicking, double tapping, and looking over) and we demonstrate how it can perform everything current mouse gadgets can. This paper demonstrates to assemble this mouse control framework.

Introduction

As PC innovation keeps on creating, individuals have littler and littler electronic gadgets and need to utilize them universally. There is a requirement for new interfaces outlined particularly for utilization with these littler gadgets. Progressively we are perceiving the significance of human figuring association (HCI), and specifically vision-based motion and protest acknowledgment. Straightforward interfaces as of now exist, for example, implanted console, organizer console and smaller than usual console. Then again, these interfaces require some measure of space to utilize and can't be utilized while moving. Touch screens are likewise a decent control interface and these days it is utilized all inclusive as a part of numerous applications. Be that as it may, touch screens can't be connected to desktop frameworks on account of expense and other equipment impediments. By applying vision innovation and controlling the mouse by characteristic hand signals, we can lessen the work space required. In this paper, we propose a novel approach that uses a video gadget to control the mouse framework. This mouse framework can control all mouse errands, for example, clicking (right and left), double tapping and looking over. We utilize a few picture preparing calculations to execute this.

Related Work

Numerous analysts in the human PC cooperation and mechanical technology fields have attempted to control mouse development utilizing video gadgets. Then again, every one of them utilized distinctive techniques to make a clicking occasion. One methodology, by Erdem et al, utilized fingertip following to control the mouse's movement. A mouse's tick catch was actualized by characterizing a screen such that a tick happened when a client's hand ignored the district [1, 3]. Another methodology was produced by Chu-Feng Lien [4]. He utilized just the fingertips to control the mouse cursor and snap. His clicking technique depended on picture thickness, and required the client to hold the mouse cursor on the craved spot for a brief timeframe. Paul et al, utilized still another strategy to click. They utilized the thumb's movement (from a 'thumbs-up' position to a clench hand) to stamp a clicking occasion thumb. Development of the hand while making an exceptional hand sign moved the mouse pointer.

System Flow

The framework request is appeared in Figure 1. At first, when we get a picture from the camera, we change over the shading space RGB to YCbCr. At that point, we characterize a scope of hues as 'skin shading' and change over these pixels to white; every single other pixel are changed over to dark. At that point, we figure the centroid of the dorsal district of the hand. After we distinguish the hand, we discover the circle that best fits

this area and increase the range of this circle by some worth to acquire the most extreme degree of a 'non-finger locale'. From the double picture of the hand, we get vertices of the raised structure of every finger. From the vertex and focus separation, we get the dynamic's positions fingers. At that point by expanding any one vertex, we control the mouse development.
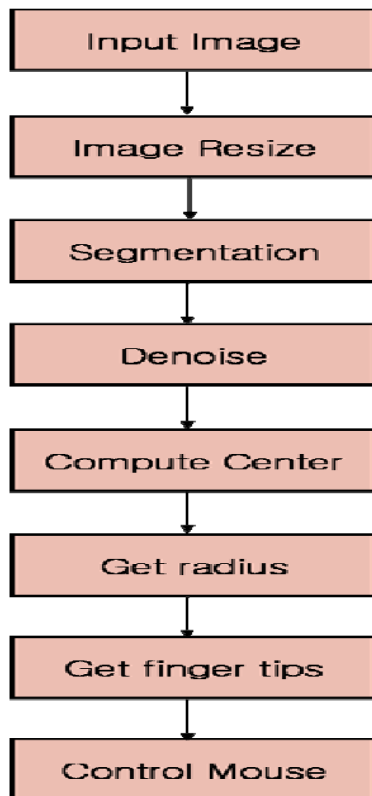
Deleting noise

Utilizing this methodology, we can't get a decent gauge of the hand picture in light of foundation commotion. To show signs of improvement appraisal of the hand, we have to erase boisterous pixels from the picture. We utilize a picture morphology calculation that performs picture disintegration and picture enlargement to dispense with commotion [1]. Disintegration trims down the picture region where the hand is not present and Dilation extends the Image's zone pixels which are not dissolved. Numerically, Erosion is given by,

$$A\Theta B = \{x \mid (B)_x \cap A^c = \varphi\}$$

where An indicates information picture and B signifies Structure components. The Structure component is worked on the Image utilizing a Sliding window and definite matches are stamped. Figure 3 demonstrates a graphical representation of the calculation. Widening is characterized by,

$$A\oplus B = \{x \mid (\hat{B}\,) \cap A \neq \varphi\} = \{x \mid [(\hat{B}\,)_x \cap A] \subseteq A\}$$

where An indicates the information picture and B means the structure component. The same structure component is worked on the picture and if the middle pixel is coordinated, the entire zone around that pixel is stamped. Figure 4 demonstrates the calculation. Disintegration and Dilation are changed by the state of B. In this way, B ought to be chosen ahead of time. We performed disintegrate capacity with structure of 10x10 square pixels three times and enlarge capacity with 6x6 square pixels structure three times to get a clearer hand. Figure 5 is the consequence of this calculation.
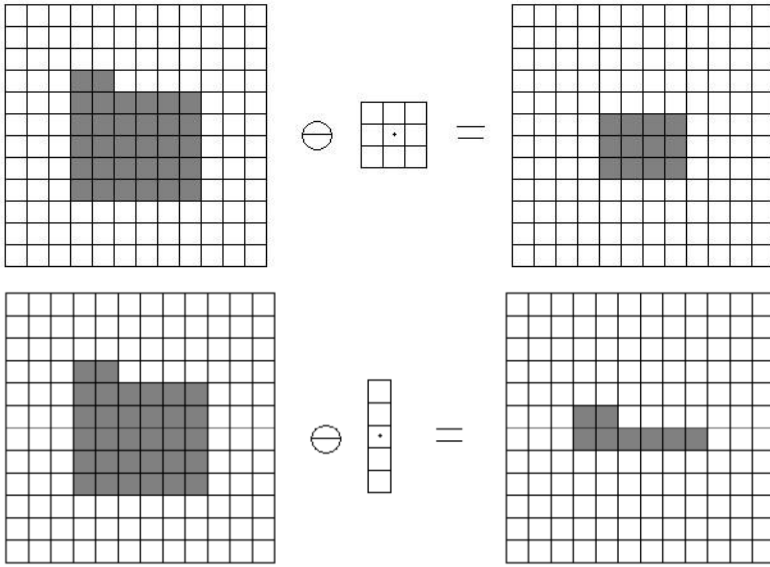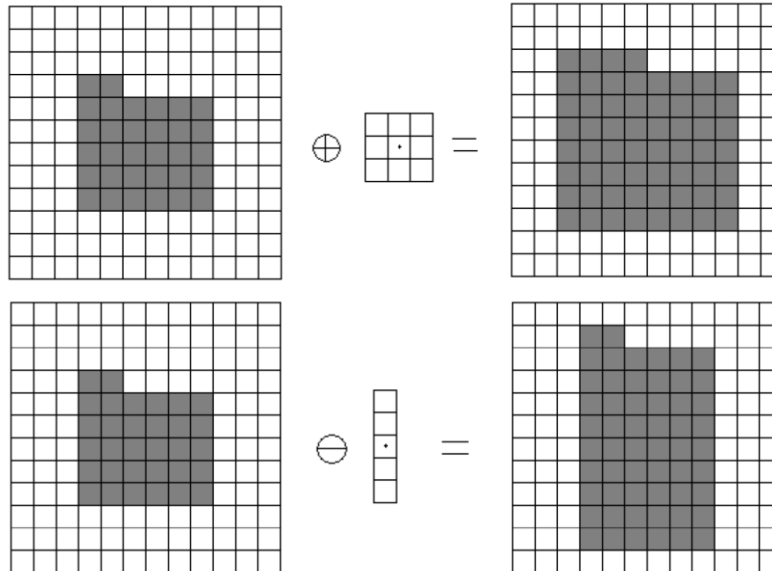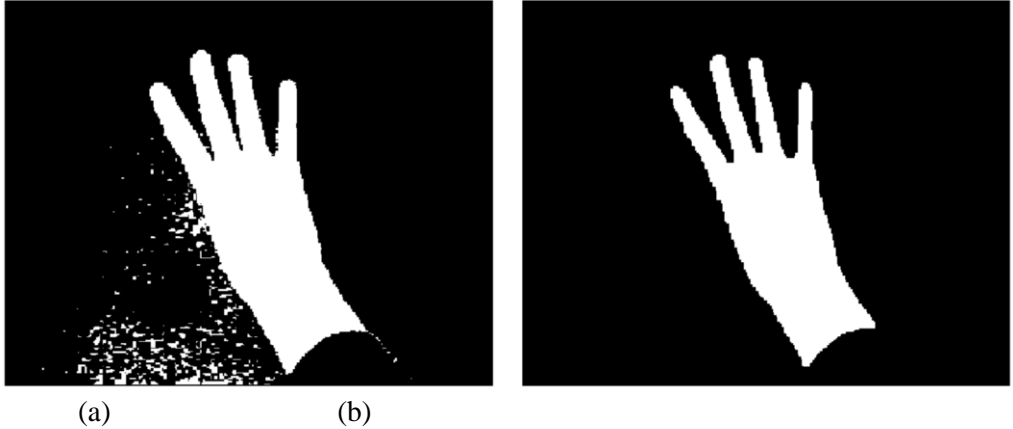
Figure 3. The result of image erosion. The structure element scans the input image. When the structure element matches, the central pixel is kept; when it does not match,all pixels are discarded.

The result of image dilation. The structure element scans the input image. When the center of the structure matches any pixels, the bin of the structure element is filled.

(a)   (b)

The result of image morphology. Perform erosion and dilation on binary image. (a) Original image. (b) The result of image morphology.

1.1 Finding finger tip

To perceive that a finger is within the palm region or not, we utilized a curved body calculation. The arched frame calculation is utilized to tackle the issue of discovering the greatest polygon including all vertices. Utilizing this component of this calculation, we can identify fingertips on the hand. We utilized this calculation to perceive if a finger is collapsed or not. To perceive those states, we increased 2 times (we got this number through various trials) to the hand range esteem and check the separation between the inside and a pixel which is in raised structure set. On the off chance that the separation is longer than the hand's sweep, then a finger is spread. Moreover, if two or all the more fascinating focuses existed in the outcome, then we viewed the longest vertex as the pointer and the hand signal is clicked when the outcome's quantity vertex is two or more.

The aftereffect of curved structure calculation has an arrangement of vertexes which incorporates all vertexes. In this way at times a vertex is put close different vertexes. This case happens on the finger's side tip. To take care of this issue, we erased a vertex whose separation is under 10 pixels when contrasting and the following vertex. At last, we can make them interest point on every finger. Figure 7 demonstrates the outcomes.
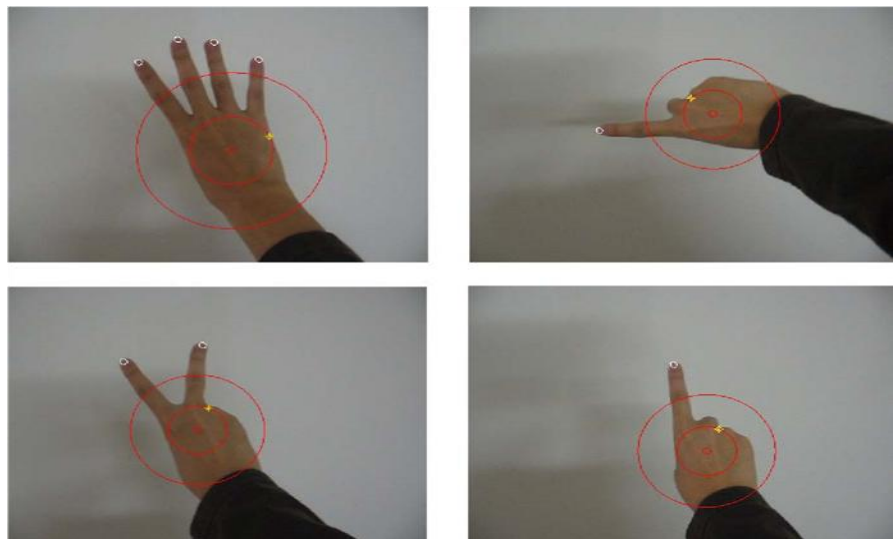
Moving Mouse Cursor

We utilized the pointer as a cursor controller to control mouse cursor. We utilized two distinctive methodologies for moving the mouse cursor. The primary system is mapping cursor control. It implies that the forefinger on a camera screen can position maps to a desktop screen position. At the end of the day, the mouse cursor is put on the desktop window alongside the pointer tips position showed on the camera screen position. This system has an issue. On the off chance that the desktop's determination window has a higher determination than the camera determination, then the cursor position can't be exact in light of the fact that when the camera determination believers to the desktop window determination we lose middle of the road esteem. We expect the proportion of hopping pixel is up to 4 pixels. The second strategy is weighted rate cursor control. We get a finger's distinction of the present picture and the past picture and process the separation between the two. Next, we move the mouse cursor if the hole between the two finger pictures (present and past casing) is far then the mouse cursor moves quick or, if the hole is close then the cursor moves moderate. This calculation additionally has an issue. Since we are working with constant picture preparing, the customer machine ought to be quick. The picture information rate is 15 outlines for every second and we need picture preparing time on CPU. It implies that a few machines which can't accomplish picture preparing 15 pictures for each sec don't work easily in light of the fact that figuring the picture focus and the hand shape requires significant investment. Therefore, this calculation does not work legitimately. In this paper, we utilized the first strategy which utilizes total position of fingertips on the grounds that it is more precise than the second technique.

Left Clicking and Double-Clicking

To call framework occasion for left clicking, no less than two raised body vertexes must be off the palm zone which was registered in the past part. Also, the x position of one of two vertexes ought to be lower than the other to limit location of other fingertips. At the point when the file's level finger and thumb is 70 to 90 degree then we can perceive that the signal is left clicking. Really, if the thumb is set outside the hand's circle area, then the signal is left clicking. The double tapping happens when the thumb moves 0 to 90 degree and back two times quick.

Right Clicking

We basically executed this part utilizing past signals. On the off chance that we make the hand posture left clicking for 3 seconds, then the framework calls the right clicking occasion.

Looking over

Looking over is an exceptionally helpful assignment with hand motions. We likewise actualized this element in our framework. As figure 5 appears, we appointed a 10% edge as a parchment region on the right half of the desktop screen area and partitioned considerably. In the event that the forefinger put the upper region of the parchment region with clicking stance then the framework rang the parchment capacity or else the framework called the look down capacity.

Analyses and Results

We tried all mouse errands such that left snap, right snap, double tap, dragging, and looking on windows. The tried framework is that Core2-Duo T8300, 2GB memory, GeForce 8300, Microsoft Windows XP, Microsoft LifeCam VX-1000 (640x480 determination, 15fps). We couldn't contrast and the mouse gadget on the grounds that this hand signal framework dependably shows lower execution than genuine mouse gadget. Rather than contrasting and a genuine mouse, we permitted to utilize this framework to four analyzers to know how it can be adjusted effortlessly.

Our system for assessing execution is that we checked the time that an errand, for example, clicking, double tapping and looking over, is finished. We planned four tests to get execution. In the first analysis we set a symbol on the focal point of desktop window and put the cursor in the upper left corner. We then measured the time in to what extent it takes to choose the symbol. In the second test the symbols' position is same and we just measured an ideal opportunity to demonstrate the drop down menu on the symbol. In the third investigation the symbols' position is same with the second trial and we quantified an ideal opportunity to open the symbol. In the last analysis we opened a

site((http://news.yahoo.com) on the focal point of the desktop window and measured the time until the parchment bar moves through and through. The outcomes are demonstrated as follows: (time = sec)

	Trial 1	Trial 2	Trial 3	Average
User1	1.2	1.0	1.1	1.10
User2	1.5	1.3	1.0	1.26
User3	1.2	1.1	1.1	1.33
User4	1.0	0.9	1.3	1.06

	Trial 1	Trial 2	Trial 3	Average
User1	4.2	4.1	4.2	4.16
User2	3.8	4.0	4.2	4.00
User3	4.2	4.1	4.1	4.13
User4	4.0	4.0	4.2	4.06

	Trial 1	Trial 2	Trial 3	Average
User1	2.5	3.1	2.2	2.6
User2	2.6	1.9	1.9	2.13
User3	1.7	1.9	2.0	1.86
User4	1.6	2.3	2.1	2.00

	Trial 1	Trial 2	Trial 3	Average
User1	3.7	2.5	7.4	4.5
User2	2.3	4.9	2.6	3.2
User3	3.2	5.2	4.1	4.1
User4	3.4	2.1	4.8	3.4

	Left-Clicking	Right-Clicking	Double-Clicking	Scrolling
User1	1.10	4.16	2.6	4.5
User2	1.26	4.00	2.13	3.2
User3	1.33	4.13	1.86	4.1

User4	1.06	4.06	2.00	3.4

Discussion

In this venture, the issue was the point at which the finger shook at part. Since we utilized continuous video, the enlightenment changes each casing. Thus the hand's position changes each edge. In this manner, the fingertip position recognized by raised body calculation is likewise changed. At that point the mouse cursor pointer shakes quick. To alter this issue, we included a code that the cursor does not move if the distinction of the past and the present fingertips position is inside of 5 pixels. This imperative functioned admirably yet it makes it hard to control the mouse cursor delicately. Another issue by enlightenment issue is division of the foundation for removing the hand shape. Since the hand mirrors every light source, the hand shading is changed by spot. On the off chance that hand shape is bad then our calculation can't function admirably in light of the fact that our calculation expect the hand shape is all around portioned. In the event that the hand shape is bad then we can't gauge the length of range of the hand.

For discovering the focal point of hand, it has an issue to locate the inside precisely. In the event that the camera demonstrated a hand with wrist, then the middle will move a little towards the wrist on the grounds that the wrist's shade is the same as hand shading. In this way, it causes that calculation framework to fizzle in light of the fact that if the middle is moved down, then the hand's range can be littler than the genuine size. Besides, the circle will move down and get to be littler so when a client evildoers his or her hand then the discovering fingertips calculation can likewise come up short in light of the fact that a thumb can be put outside of the circle inevitably.

Conclusion

We added to a framework to control the mouse cursor utilizing a continuous camera. We actualized all mouse undertakings, for example, left and right clicking, double tapping, and looking over. This framework depends on PC vision calculations and can do all mouse undertakings. Be that as it may, it is hard to get stable results in view of the assortment of lighting and skin shades of human races. Most vision calculations have brightening issues. From the outcomes, we can expect that if the vision calculations can work in all situations then our framework will work all the more effectively. This framework could be helpful in presentations and to lessen work space. Later on, we plan to include more elements, for example, amplifying and using so as to contract windows, shutting window, and so on the palm and various fingers.

References

[1] Hojoon Park. A Method for Controlling Mouse Movement using a Real-Time Camera. ww.cs.brown.edu/research/pubs/theses/masters/2010/park.pdf2010.

[2] Computer vision based mouse, A. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E. Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASS). IEEE International Conference.

[3] Vision based Men-Machine Interaction http://www.ceng.metu.edu.tr/~vbi/.

[4] Chu-Feng Lien, Portable Vision-Based HCI - A Real-time Hand Mouse System on Handheld Devices.

[5] Ben askar Chris Jordan, Hyokwon Lee mouse free. http://www.seas.upenn.edu/cse400/CSE400-2009- 2010/nal-report/Jordan-lee.pdf, 2009-2010.

[6] Pranavmistry, Sixthsense. http://www.youtube.com/watch?v=ZfV4R4x2SK0, 2010. [Online; accessed28-november-2010].

[7] Mark dot lee Java bindings for the vlc media player.http://www.capricasoftware.co.uk/vlcj/index.php ,2011. [Online; accessed 1-may-2011].

[8] Pranavmistry, Mouseless http://www.pranavmistry.com/projects/mouseless/,2010 [Online; accessed28-november-2010].

[9] Shaker, N.; Abou Zliekha, M.;Damascus Univ., Damascus,