Computer control in human-machine interaction systems by hand movements

Hoa Tat Thang[1*], Doan Van Hoa[2], Do Ngoc Tuan[2]

[1]Le Quy Don Technical University;
[2]Academy of Military Science and Technology, Vietnam.
*Corresponding author: hoatatthang@gmail.com.
Received 16 September 2021; Revised 9 December 2021; Accepted 15 December 2021.
DOI: https://doi.org/10.54939/1859-1043.j.mst.csce5.2021.42-48

ABSTRACT

Computers have become popular in recent years. The forms of human-computer interaction are increasingly diverse. In many cases, controlling the computer is not only through the mouse and keyboard, but humans must control the computer through body language and representation. For some people with physical disabilities, controlling the computer through hand movements is essential to help them interact with the computer. The field of simulation also needs these interactive applications. This paper studies a solution to build a hand tracking and gesture recognition system that allows cursor movement and corresponding actions with mouse and keyboard. The research team confirms that the system works stably, accurately and can control the computer instead of a conventional mouse and keyboard through the implementation and evaluation.

Keywords: Human-machine interaction; Gesture recognition; Body language; Movement of hands.

1. INTRODUCTION

Many studies have shown that for traditional human-machine interaction without using mouses and keyboards, webcams are the most effective devices to recognize human commands for computer systems. Many systems use different ways to represent mouse clicks, grabs, releases, or keystrokes. Erdem et al. used fingertip tracking to control mouse movements in one study. A mouse button click is implemented by defining a screen such that a click occurs when the user's hand hovers over the area [1]. Another approach was developed by Chu-Feng Lien [2]. This method only uses the fingertips to control the mouse pointer and click. Clicking is based on image density and requires the user to hold the mouse pointer at the desired location for a short time. Kamran Niyazi et al. [3] used a webcam to detect the color of the cloth tape applied to the fingertip. The movements of the fingers wearing this colored cloth will be equivalent to the movements of a computer mouse. Clicks are calculated based on the distance between the colored cloth tapes on the fingertips. The method of using colored cloth tape glued on the finger to simulate the actions of a computer mouse is also used in [4], [5]. These methods all accurately simulate the actions of a computer mouse, such as mouse movement, left and right-click, left double click, or drag operations. Still, the disadvantages of these methods are that specific colored cloth tapes must be used at the tips of the fingers and these devices are not always available.

For people with disabilities or ordinary people with some special working conditions, interacting with the computer in the usual form with a mouse or keyboard can be challenging. In military simulation systems, human-machine interaction is often used in interactive control applications by hand movements used in many simulation problems such as simulation in bomb and mine disposal.

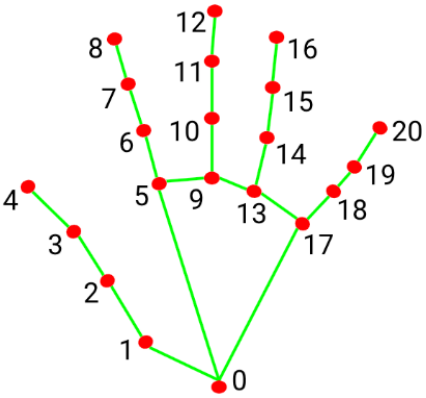
Gesture recognition research is to identify a specific human gesture and transmit the information to the computer. The general goal is to make computers understand human body language, thereby bridging the gap between machine and human. Hand gesture recognition can be used to enhance human-computer interaction without relying on traditional input devices such as keyboards and mouses. The proposed interactive system uses a webcam to identify and recognize hand gestures based on hand gestures. In this system that eliminates the sensor, we only analyze the hand image to control the pointer's movement and the computer keyboard.

In the new proposal of the research team, we propose a new method to simulate mouse activities and some critical actions on the keyboard just by recognizing the human hand through the camera without any equipment. Colored cloth tape does not need to be used in this case. The proposed method confirms high accuracy, is capable of flexibly simulating mouse movements and some basic keyboard operations such as left, right, up, down, page up, page down.

2. A METHOD OF DETERMINING THE KEY POINTS OF THE HANDS

2.1. Theoretical foundations

In our research, we propose a method to determine the key points of the hands based on the open library of Google MediaPipe.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Figure 1. *Key points of the hands*

(https://google.github.io/mediapipe/solutions/hands).

MediaPipe allows recognizing 21 essential key points on the hand. After determining the key points, the next task is to determine the opening and closing actions of the fingers, determine the distance between the fingers, and finally, based on the relative position of the fingers to determine the corresponding mouse and keyboard actions. The functional diagram of the system is shown in figure 2.

The main contents that the system needs to resolve to include:

- Directly capture hand image data from the webcams.
- Pre-processing, flipping the obtained images.
- Control the mouse pointer through key points movement
- Perform left-click, right-click, pick-and-drop, and double-click operations.

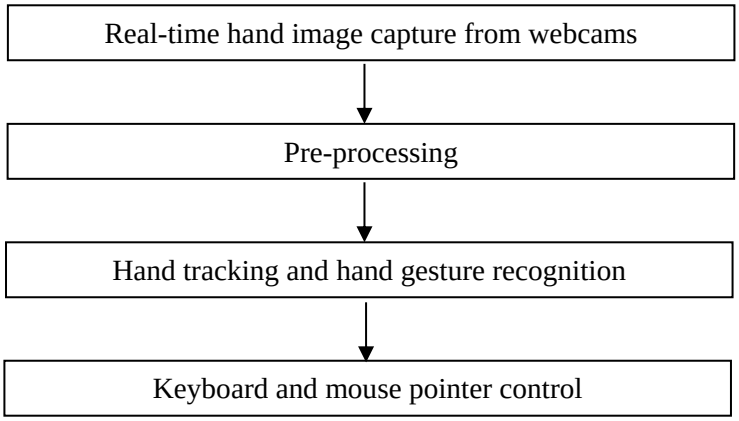


Figure 2. *The working diagram of the system.*

2.2. Experiment preparation

2.2.1. Directly capture hand image data from webcams.

To obtain hand images, we use High-definition (HD) resolution webcams. The webcam captures continuous video at a steady speed. We use laptops with configuration: Core i7 CPU, 8Gb Ram with Windows 10 operating system installed in our experiments. The measured processing speed ranges from 35-40 frames/sec, entirely suitable for controlling the computer in real-time.

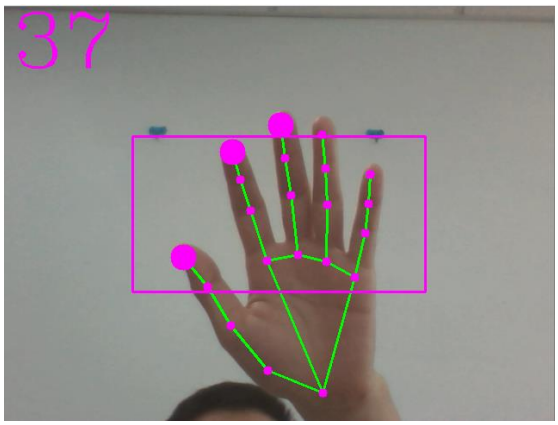


Figure 3. *Processing the received hand image.*

2.2.2. Pre-processing, flipping the obtained images

Images obtained from the webcam must be preprocessed. The received image is not the same as the image we stand in front of the mirror, so if the hand moves to the left, the image of the receiving hand also moves to the left of the screen, but from our viewing angle, it moves to the right relative to us. This makes it challenging to use the hand image to move the mouse, so it is necessary to process the obtained image and flip the image horizontally so that the obtained image is like when we stand in front of a mirror. This will be more convenient when controlling the hand into the correct position.

2.2.3. Controlling the mouse pointer through key point movement

The mouse pointer's position will be assigned a critical point on the hand. When this key point moves, the mouse pointer moves accordingly. Some problems when choosing

a pivotal point to control the mouse pointer should be noted as follows:

When the fingers move (clicks, keyboard controls), the key points vibrate accordingly. If the critical point is selected incorrectly, the click operations will be incorrect. When the mouse is clicked, the cursor will be moved to another position, so choosing the key point least affected when performing finger movements is necessary. The experiment shows that key point number 17 (see location in Figure 1) is less affected even when performing finger movements, so in our study, we choose key point number 17.

Controlling the mouse pointer in the corners of the screen without a reasonable method will not be possible because part of the hand will not be on the screen, and then the hand will not be recognized. To overcome this situation, we propose a solution that only detects key point 17 in a rectangle, with each edge receding from the edges of the frame received from the webcam by about 150 pixels, as shown in Figure 1.

The condition for the mouse pointer to move we choose is when the distance between the two key points on the tip of the thumb and middle finger is less than α value (Experimentally, we choose this value to be 30 pixels). The reason for the selection will be presented in the content of the action of picking and releasing the mouse. Normally, the mouse pointer will move when the three thumbs, index, and middle fingers are pinched, as shown in Figure 4.

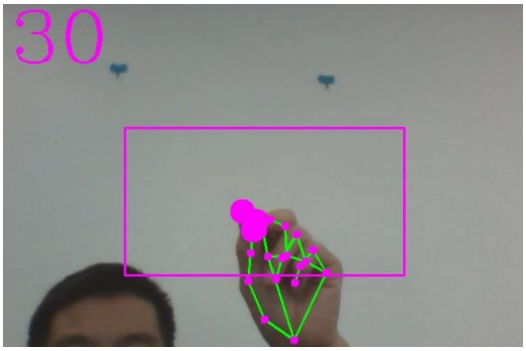


Figure 4. *Mouse pointer movement.*

With the obtained x, y position of the key point 17 obtained on the screen, the $xloc$ and $yloc$ positions of the mouse pointer on the screen are calculated according to the following formula:

$$xloc = wScr * \frac{x-offset}{wCam-2*offset};$$
$$yloc = hScr * \frac{y-offset}{hCam-2*offset}.$$

Where:

$xloc, yloc$: The position of the cursor on the screen;

$wScr, hScr$: The respective width and height of the display screen.

$wCam, hCam$: The corresponding width and height of the frame obtained from the webcam.

$offset$: The shortened distance from the edges of the frame is obtained; in our case, this value is equa to 150 pixels.

2.2.4. Performing left-click, right-click, pick-and-drop, and double-click operations

The left-click operation is activated by calculating the distance between two key points on the index finger and thumb tips. The mouse down operation is activated when this distance is less than α value. The mouse up operation is activated when this distance is greater than α value.

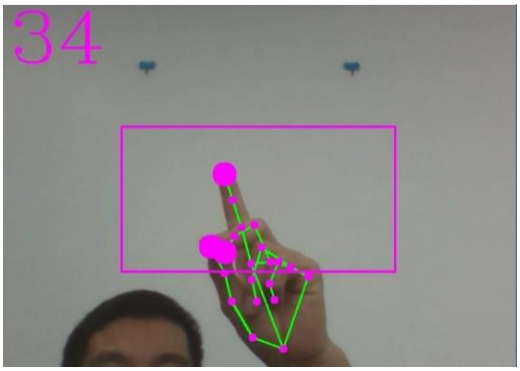


Figure 5. *Left mouse button down operation.*

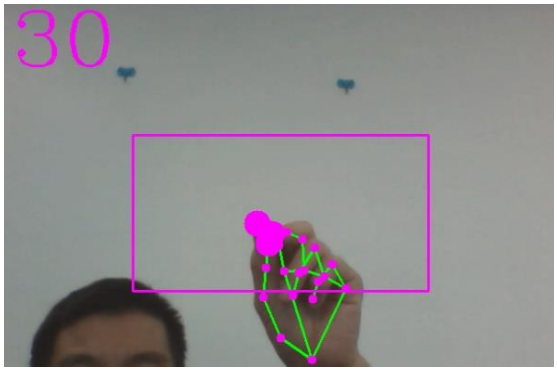


Figure 6. *Left mouse button up operation.*

Similar to left-clicking, right-clicking is enabled by calculating the distance between two key points on the tip of the thumb and middle finger. The right-click is activated when this distance is greater than α value. In our experience, α value is also 30 pixels, as in the case of left-clicking.

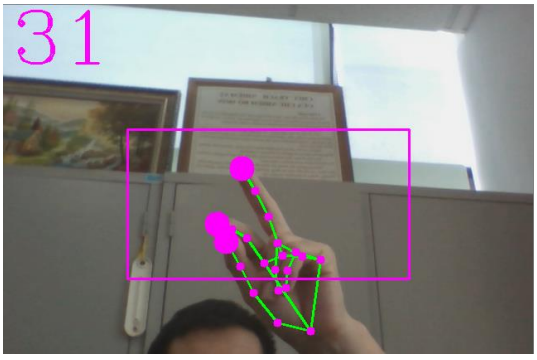


Figure 7. *Right-click operation.*

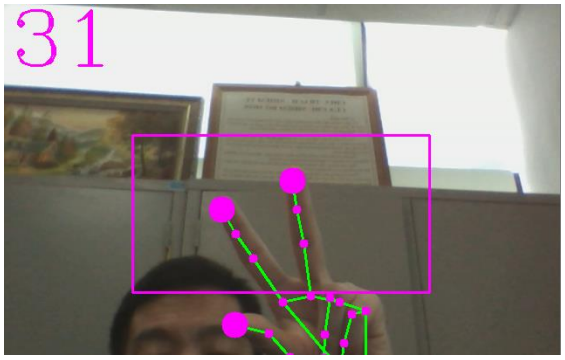


Figure 8. *Double click operation.*

Double click is done in 2 ways:

- Method 1: Perform 2 times left-click operation.

- Method 2: Move the tips of the index and middle fingers away from the thumb, as shown in figure 8.

Pick and drop operations are performed when the mouse pointer moves while the left mouse button is pressed down (Figure 5). When the pick operation is complete, the release operation is performed when the left mouse button is released (Figure 6). For this reason, we choose the condition for the mouse pointer to move only when the distance between the two key points on the tip of the thumb and middle finger is less than α value, not necessarily all three pinched fingers.

3. EXPERIMENTAL SIMULATION RESULTS

We perform our experiment on a regular laptop, with a core i7 CPU and, 8Gb Ram with Windows 10 operating system. Python programming environment, using MediaPipe library to extract key points of the hands and Autopy library to control a computer mouse. We perform a series of mouse pointer actions such as moving the cursor, left-click, right-click, double-click, drag and drop operations and are compared with using colored tape on the fingertips [7]. The results show that the proposed method has much higher accuracy, faster processing speed and allows working in real-time. The distance of the hands to the camera in both cases is 1 metre. The results are shown in Figure 9. In Figure 9, we see that the correct recognition rate is over 95% for the proposed method in all actions. In comparison, the color tape method on the finger has a maximum correct recognition rate of 95% in the case of cursor movement. The accuracy of the proposed method is greater than the hand tape method by at least 2.5% in case of cursor movement action and up to 11% in case of drag action.

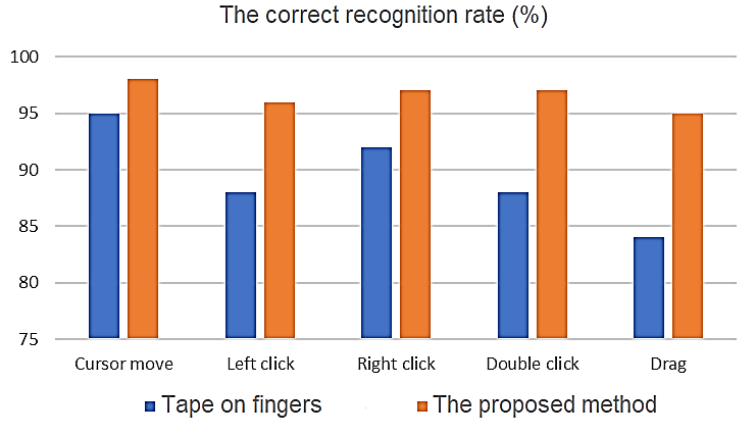


Figure 9. *The correct recognition rate of mouse pointer actions.*

4. CONCLUSION

The proposed method has been shown to be effective in controlling the mouse pointer, and the basic functions are shown in real-time. We have implemented some basic operations commonly used by computer mouse such as mouse movement, left-click, right-click, double-click, and pick-and-drop operation. The recommended method using key points of the hand extracted from deep learning techniques and Google's MediaPipe library shows superior performance than the old ways using image processing. The system is suitable for projection, remote computer control without direct impact on the computer but only needs to move the hand.

Currently, we have researched to handle some basic keyboard operations (left, right, up, down), but in the future, we will continue to study to use more keyboard functionality and better remote desktop control.

REFERENCES

[1]. Erdem, E. Yardimci, Y. Atalay, V. Cetin, A. E. "*Computer vision based mouse*", Acoustics, Speech, and Signal Processing, Proceedings. (ICASS). *IEEE International Conference*, 2012.

[2]. Chu-Feng Lien, "*Portable Vision-Based HCI – A Realtime Hand Mouse System on Handheld Devices*", National Taiwan University, Computer Science and Information Engineering Department.
[3]. Kamran Niyazi, Vikram Kumar, Swapnil Mahe, Swapnil Vyawahare, "*Mouse Simulation Using Two Coloured Tapes*", Department of Computer Science, University of Pune, India, International Journal of Information Sciences and Techniques (IJIST) Vol.2, No.2, March 2012.
[4]. Akshay Ishwar Pawar, Sahil Prakash Tumbare, Tejal Avinash Godse, Tejaswini Wagh, "*Mouse Control using a Web Camera and Hand Gestures with Colour Tapes*", International Journal of Science Technology & Engineering, Volume 2, Issue 11, May 2016.
[5]. Riza Sande, Neha Marathe, Neha Bhegade, Akanksha Lugade, Prof. S. S. Jogdand, "*Virtual Mouse using Hand Gestures*", International Journal of Advanced Research in Science, Engineering and Technology, Vol. 8, Issue 4 , April 2021.
[6]. Abhik Banerjee, Abhirup Ghosh, Koustuvmoni Bharadwaj, "*Mouse Control using a Web Camera based on Color Detection*" ,IJCTT, vol.9, Mar 2014.
[7]. Rachit Puri , "*Gesture Recognition Based Mouse Events*", International Journal of Computer Science and Information Technology, 2014.
[8]. Angel, Neethu.P.S, "*Real Time Static & Dynamic Hand Gesture Recognition*", International Journal of Scientific & Engineering Research Volume 4, Issue3, March-2013.

TÓM TẮT

ĐIỀU KHIỂN MÁY TÍNH TRONG CÁC HỆ THỐNG TƯƠNG TÁC NGƯỜI MÁY BẰNG CỬ ĐỘNG CỦA BÀN TAY

Máy tính đã trở nên phổ biến trong những năm gần đây. Các hình thức tương tác giữa con người với máy tính càng ngày càng đa dạng. Việc điều khiển máy tính không chỉ thông qua chuột và bàn phím mà trong nhiều trường hợp con người cần thiết điều khiển máy tính thông qua ngôn ngữ và biểu diễn của cơ thể. Đối với một số người khiếm khuyết về cơ thể thì việc điều khiển máy tính thông qua cử động bàn tay là rất cần thiết giúp họ tương tác với máy tính. Trong lĩnh vực mô phỏng cũng cần các ứng dụng tương tác này. Bài báo này nghiên cứu giải pháp xây dựng hệ thống theo dõi bàn tay và nhận dạng cử chỉ để cho phép di chuyển con trỏ và hành động tương ứng với chuột và bàn phím. Qua triển khai và đánh giá chúng tôi khẳng định hệ thống làm việc ổn định, chính xác, có thể điều khiển máy tính thay chuột và bàn phím thông thường.

Từ khóa: Tương tác người-máy; Nhận dạng cử chỉ; Ngôn ngữ cơ thể; Cử động của bàn tay.