

MLI. Mouse: a new computer vision-based cursor control software

Pavel Peresunko ¹, Ekaterina Pleshkova ¹, Anastasiya Semizorova ¹ and Igor Kovalev ¹

¹ Institute of Space and Information Technologies, Siberian Federal University, 26 Kirensky st., 660074 Krasnoyarsk, Russia

Abstract

The personal computer opens tremendous opportunities for people, whether it is entertainment or work. However, people with disabilities cannot successfully use usual methods of interacting with a computer like a mouse and a keyboard. There are analogs that allow individuals who had disabilities to control the computer using computer vision. They can use their limbs, head, eyes, color markers as reference point, that is used to move a cursor on their computer screen. One of the most popular cursor control software is The Camera Mouse, in which the user selects the area that will serve as a reference point. However, people may find it difficult to adjust the sensitivity of the reference point movement. For example, if the sensitivity is low, some area around the edges of the screen may become inaccessible. At high sensitivity, not all the pointer movement potential is realized. And the high sensitivity, the small movements, noise, errors affect the swing of the reference pointer more. In this study, the MLI. Mouse proposed. It allows user to control the computer cursor using computer vision techniques. To control the cursor, a person needs a webcam, from which the current position of the reference point will be read. A hand or a QR code can be used as the reference point. A useful feature of this program is a convenient way to visualize and set the active area of the cursor movements. This is the area where the reference pointer movement is broadcast on the computer screen. With this, the program can be configured faster and more conveniently. The left mouse button clicking is implemented using Dwell click.

Keywords

Human computer interface, computer vision, cursor control, MLI. Mouse

1. Introduction

There are many systems in the world that are a way of connecting a person and the world. For a person, the quality of the world can be both the real world and the virtual world. A computer can serve as a virtual world. The computer gives many opportunities to make our life more productive and interesting. For example, on the computer, people can do their work, automate daily routine, play games, watch various videos, read books and articles. In addition, the computer gives us the opportunity to communicate through emails, various messengers like Telegram, WhatsApp, Viber, and others. And the area that explores the way a computer and a person interact is called the Human Computer Interface (HCI). HCI is actively used to implement different ways of human-computer interaction [1, 2, 3]. Some of them can even reduce the spread of COVID 19 [4], as they offer a contactless way of entering information. As a way of interaction between a person and a computer, we will consider the control of a computer cursor. With a graphical user interface (GUI), we can do almost all the necessary operations to explore a computer with a click. For example, in Windows and various Linux distributions, we can

Proceedings of MIP Computing-V 2022: V International Scientific Workshop on Modeling, Information Processing and Computing, January 25, 2022, Krasnoyarsk, Russia

EMAIL: pvperesunko@yandex.ru (Pavel Peresunko); Katya.pleshkova.1@gmail.com (Ekaterina Pleshkova); semizorova2205@mail.ru (Anastasiya Semizorova); kovalev.fsu@mail.ru (Igor Kovalev)

ORCID: 0000-0003-3708-129X (Pavel Peresunko); 0000-0002-7719-4064 (Ekaterina Pleshkova); 0000-0001-8476-1859 (Anastasiya Semizorova); 0000-0003-2128-6661 (Igor Kovalev)



© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

enable the one-click option. With this option, one click can launch files and directories. We can even enter text with the cursor using the on-screen keyboard.

In this work, we will consider a cursor control system for sick people who have movement restriction due to disease that interferes with normal interaction with a computer.

Cursor control for such people can be used to communicate with the real world. For example, in [5], you can use the movement of your arms to control a wheelchair and thus interact with the real world. But cursor control can connect people to the virtual world, where a person can satisfy one of their basic needs - the need to communicate.

The main cursor control method in the world is the computer mouse. However, if a person cannot use a regular computer mouse, then he can use alternative ways to control the cursor. Researchers identify several alternative ways to control the cursor - accelerometer [6], neural interface system [7] and computer vision techniques. For the first two methods, there is a significant drawback - this is the purchase of equipment, some of which are significantly expensive. In addition, the assistant must adjust the accelerometer on the patient's body from time to time.

One of the popular ways to control the cursor using computer vision is gaze control. However, to use them, people usually buy special equipment that determines the position of the eyes more accurately. A relatively new article [8] lists prices for such equipment, which range from 100\$ to 40,000\$. Therefore, this paper discusses ways to manage a computer without buying expensive devices. The control takes place using conventional video cameras like a web camera.

In this study, it is proposed to define an active area in which the user can move a reference point (RP). This area can be a rectangle with any proportions. Therefore, in this study, the "MLI. Mouse" is presented - the software that allows person to conveniently configure the active area in which the RF will move.

2. Literature review

In the literature, many articles on cursor control with a colored marker. A colored marker for such systems acts as the RP. For example, [9] uses special-colored labels to find the reference point. The labels are placed on the fingers. RP detection is done by dividing video into three channels (red, green, and blue) and processing the channels, converting them to grayscale. In [10] there is a transformation from RGB to HSV image, and then the selection is performed by color. A similar procedure occurs in [11]. But with this approach, similar colors in the background interfere, which can interfere with RP detection.

To detect RP, instead of selecting RP by color, selection of the hand can be made. For example, in [12] the hand is detected by SSD with the bounding box, and then the center of this box is used as RP. Another option for defining a hand is an approach based on obtaining a skeleton of a hand [13]. However, some patients are poorly able to actively move their arm, so this approach is not possible. Some articles [14] suggest using the head to control the cursor. But with long-term use, users may experience neck pain.

A more general way to control the cursor is to control it with any part of the body. For example, in MarkerMouse [15] xproposed to print a mark that can be places to the head. The cursor is controlled using this mark. But of course, this mark can be placed on any part of the body.

One of the most popular cursor controls is Camera Mouse [16]. The Camera Mouse website says that over 3,300,000 downloads have been registered so far. In this system, the user or assistant selects a point on video that will be the RP. Thanks to this approach, Camera Mouse can use any part of the body to control the cursor. However, people may find it difficult to adjust the sensitivity of the referent point movement. For example, if the sensitivity is low, some areas around the edges of the screen may become inaccessible. At high sensitivity, not all the pointer movement potential is realized. And the higher the sensitivity, the smaller movements, noise, errors more affect the swing of the reference pointer. The system allows user to adjust the sensitivity γ_x and γ_y of the cursor to movement along the X-axis and the Y-axis respectively. The higher the sensitivity, the more a change in the RP position on the camera will affect the change in the position of the cursor on the computer screen. To adjust the sensitivity, you need to select the sensitivity level, click on save, check the sensitivity level, change the sensitivity level, and so on. In fact, by changing the sensitivity γ_x and γ_y of the reference point on Y

and X respectively, we change the size of the active area rectangle along the X-axis and Y-axis. The Figure 1 shows the transformation of the active area on the computer screen. As a result of the sensitivity adjustment some errors can occur. For example, if the sensitivity is low, some areas around the edges of the screen may become inaccessible. At high sensitivity, not all the pointer movement potential is realized. And the higher the sensitivity, the smaller movements, noise, errors more affect the swing of the reference pointer.

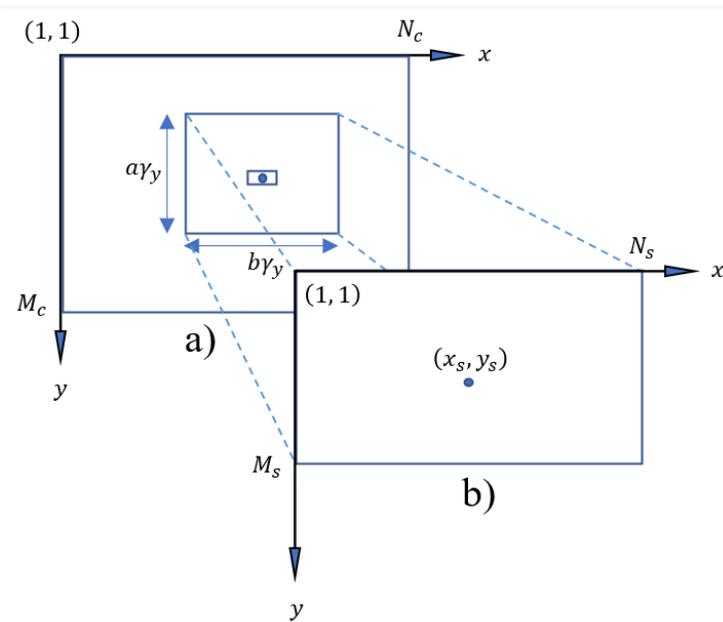


Figure 1: Transformation of the active area on the computer screen. The a) part shows a camera with resolution $M_c \times N_c$. The reference point that the user selects by drawing a small rectangle with coordinates (x_v, y_v) in the center. The user configures the sensitivity parameters γ_y and γ_x . Further, the position of the cursor on the active area affects the position (x_s, y_s) of the cursor on the screen with resolution $M_s \times N_s$

For this reason, this paper presents a new software that can speed up and simplify the way to configure the active area. This is achieved by visualizing the active area settings, which can be conveniently superimposed directly on the video from the camera. The transformation of the position of the cursor on the screen is given by a simple formula.

3. Methodology

This program was implemented in Python using the PyQt module. Many libraries are implemented in Python, for example, OpenCV, in which video is read and processed. The software in this version uses a conventional video camera. This can be either a webcam built into a laptop, or an external camera that connects via USB. These cameras return RGB video to the program input, which, in fact, are snapshots every m milliseconds, depending on the camera's FPS. In this study, cameras with 30 FPS were used. The autopy module is used to click and move the mouse cursor.

To use this software, the user must be able to use any part of the body, such as head, shoulder, elbow, hand, fingers, feet, toes, and so on. This software uses the center of the QR code as a reference mark. QR codes have many advantages. First, robust, and fast code detection, even on mobile devices. This means that even on weak computers, the system will work quickly. Secondly, the code can be used to identify the user in situation where there are many patients. Also, different codes can be used for different click actions, for example, as in [17]. The software also implements cursor control using the skeleton of a hand. The skeleton was detected using mediapipe module. Any finger can be used as the Reference point, but the default is the middle finger.

In this work, it is assumed to use positional (absolute) positioning of the cursor, since in some studies, for example, in [14] it is indicated that in their conditions this method of positioning proved to be better.

Click are realized with Dwell time. A click occurs when the cursor moves in an area of radius R for more than S seconds, where R and S are adjustable parameters in the program.

To find the coordinates x_a , y_a of the RP relative to the active area, the formulas (1) and (2) is used.

$$y_a = y_c - y_1, \quad (1)$$

$$x_a = x_c - x_1, \quad (2)$$

where y_a – the y-coordinate of the cursor relative to the active area, x_a – the x-coordinate of the cursor relative to the active area, y_1 – y coordinate of the upper left corner of the active area, x_1 – x coordinate of the upper left corner of the active area, y_c – y coordinate of RP on the camera, x_c – x coordinate of RP on the camera.

Further, after, to get the normalized coordinates of the cursor, it need to be normalized so that they lie in the range from 0 to 1. Formulas (3) and (4) are used for it.

$$y_n = \frac{y_a}{y_2 - y_1}, \quad (3)$$

$$x_n = \frac{x_a}{x_2 - x_1}, \quad (4)$$

where y_1 – y coordinate of the upper left corner of the active area, x_1 – x coordinate of the upper left corner of the active area, y_2 – y coordinate of the bottom-right corner of the active area, x_2 – x coordinate of the bottom-right corner of the active area, y_a – relative y position of the RP along X-axis, x_a – relative x position of the RP along Y-axis, y_n – normalized y-coordinate of the cursor, x_n – normalized x-coordinate of the cursor.

To obtain the coordinates of the cursor on the screen, the normalized points are multiplied by the screen resolution using the formulas (5) and (6).

$$y_s = y_n \times N_s, \quad (5)$$

$$x_s = x_n \times M_s, \quad (6)$$

where y_n – normalized y-coordinate of the cursor, x_n – normalized x-coordinate of the cursor, N_s – screen width, M_s – screen height.

All these transformations can be reduced to two formulas (7) and (8).

$$y_s = \frac{y_c - y_1}{y_2 - y_1} \times N_s, \quad (7)$$

$$x_s = \frac{x_c - x_1}{x_2 - x_1} \times M_s, \quad (8)$$

where y_1 – y coordinate of the upper left corner of the active area, x_1 – x coordinate of the upper left corner of the active area, y_2 – y coordinate of the bottom-right corner of the active area, x_2 – x coordinate of the bottom-right corner of the active area, y_c – y coordinate of RP on the camera, x_c – x coordinate of RP on the camera, N_s – screen width, M_s – screen height.

Figure shows a schematic representation of different positions of the active area on the same camera with the cursor in the same position. Changing the location of the active area changes the location of the cursor. And it doesn't matter what proportions the active area rectangle has, the cursor still covers the entire area of the monitor screen. And the narrower the rectangle, the higher the sensitivity of the cursor when moving it along the X axis. The same thing happens with moving the cursor along the Y axis.

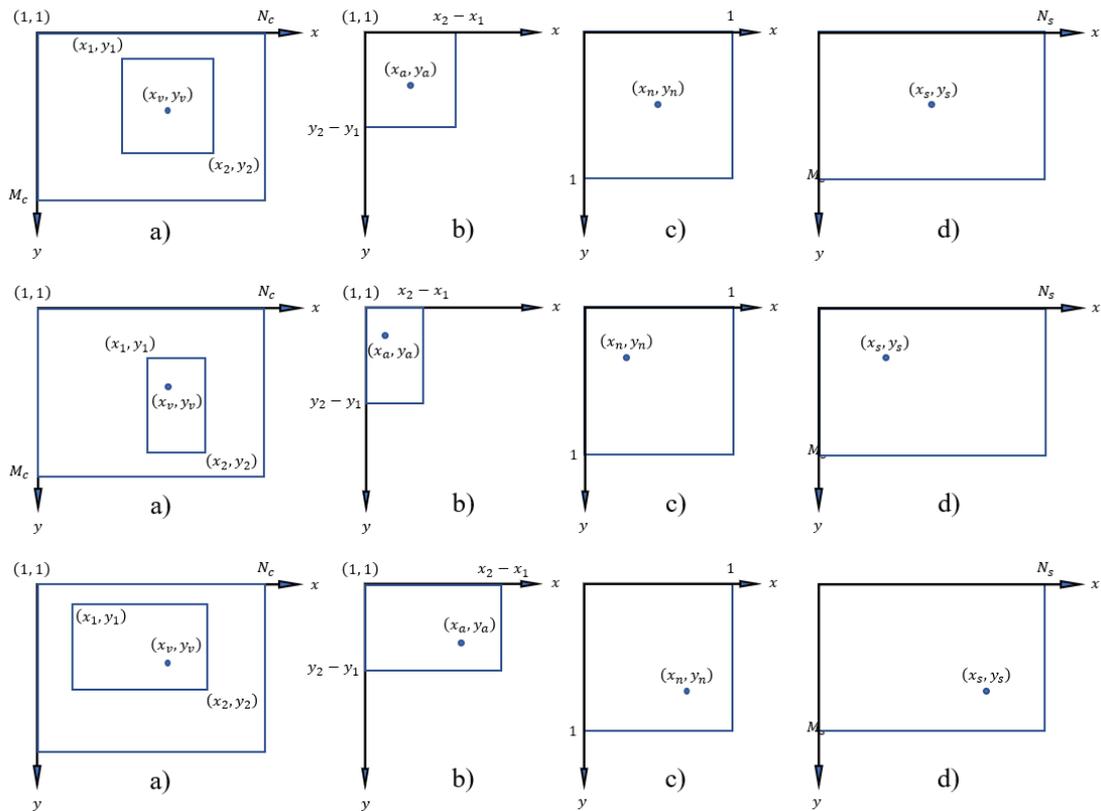


Figure 2: Examples of transforming the same cursor position for three different active area. The conversion progress is shown by row. Step is a) – reference point in video camera coordinates, b) – reference point in active area coordinates, c) reference point in normalized coordinates, d) cursor on screen

4. Result and discussion

In this study, a new software was presented that allows user to control the computer cursor using computer vision. The user, having a camera built into a laptop or any video camera that can be connected to a computer using a USB cable, can use this program.

This software allows to conveniently set the active area, the movement along which sets the movement of the cursor on the computer screen. The assistant selects the active area where the user can move the RP. The RP position on the camera is then transformed into the cursor position on the screen. For this, spatial transformations are used for cursor position, which is set by a special formula.

Using the cursor and GUI, you can open files, run programs, surf the Internet, communicate, write text using the on-screen keyboard. Thus, this program allows people with disabilities to contact with the world around them.

In addition, such important mouse functionality as pressing the right mouse button, holding the left mouse button, scrolling has not yet been implemented.

5. Acknowledgements

This work was initiated at the request of Professors Semyon Vladimirovich Prokopenko and Alexandra Alekseevna Khomchankova, who described the situation with cursor control systems.

6. References

- [1] R. E. O. S. Ascari, R. Pereira, L. Silva, Towards a Methodology to Support Augmentative and Alternative Communication by means of Personalized Gestural Interaction. in Proceedings of the 17th Brazilian Symposium on Human Factors in Computing Systems 1-12 (ACM, 2018). doi:10.1145/3274192.3274230.
- [2] D. Lee, Y. Park, Vision-based remote control system by motion detection and open finger counting, *IEEE Transactions on Consumer Electronics*. 55(4) (2009) 2308-2313. doi:10.1109/TCE.2009.5373803
- [3] P. N. Saindane, R. Patil, S. Pagar, Controlling Mouse Movements Using hand Gesture And X box 360, *International Research Journal of Engineering and Technology (IRJET)*. 4(2) (2017) 982-986. <https://irjet.net/archives/V4/i2/IRJET-V4I2190.pdf>
- [4] S. Shriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay, Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread, *Journal of Healthcare Engineering*. 1-8 (2021).
- [5] J. Zeng, Y. Sun, F. Wang, A Natural Hand Gesture System for Intelligent Human-Computer Interaction and Medical Assistance. in 2012 Third Global Congress on Intelligent Systems. 382-385 (IEEE, 2012). doi:10.1109/GCIS.2012.60.
- [6] M. D. Groll, S. Hablani, J. M. Vojtech, C. E. Stepp, Cursor Click Modality in an Accelerometer-Based Computer Access Device, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 28(7) (2020) 1566-1572. doi:10.1109/TNSRE.2020.2996820.
- [7] S. Kim, J. D. Simeral, L. R. Hochberg, J. P. Donoghue, G. Friehs, M. J. Black, Point-and-Click Cursor Control With an Intracortical Neural Interface System by Humans With Tetraplegia, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 19 (2011) 193-203.
- [8] I. Rakhmatulin, A Review of the Low-Cost Eye-Tracking Systems for 2010-2020, *SSRN Journal* (2020). doi:10.2139/ssrn.3765930.
- [9] A. Shastrakar, J. Raman, M. Paul, N. Ramteke, P. V. Sathwane, Cursor Movement Control Using Colour Detection. 4(1) (2018) 1298-1302.
- [10] F. Khan, B. Halim, A. Rahman, Computer Vision Based Mouse Control Using Object Detection and Marker Motion Tracking. 9(5) (2020) 35-45.
- [11] S. Ashvini, R. Jitesh, P. Mamun, R. Neha, S. Vishal. Cursor Movement Control Using Colour Detection. *International Journal of Scientific Research in Science, Engineering and Technology*. 4(1) (2018) 1298-1302.
- [12] Y. Foroutan, A. Kalhor, S. M. Nejati, S. Sheikhaei, Control of computer pointer using hand gesture recognition in motion pictures. (2020) 1-8. <http://arxiv.org/abs/2012.13188>.
- [13] M. Oudah, A. Al-Naji, J. Chahl, Hand Gesture Recognition Based on Computer Vision: A Review of Techniques. *Journal of Imaging*. 6(8) (2020). <https://doi.org/10.3390/JIMAGING6080073>.
- [14] H. A. Alhamzawi, Control Mouse Cursor by Head Movement: Development and Implementation. *Applied Medical Informatics*. 40(3-4) (2018) 39-44.
- [15] M. Betke, J. Gips, P. Fleming, The Camera Mouse: visual tracking of body features to provide computer access for people with severe disabilities. *IEEE Trans. Neural Syst. Rehabil. Eng.* 10(1), (2002) 1-10.
- [16] A. Shastrakar, J. Raman, M. Paul, N. Ramteke, P. V. Sathwane, Cursor Movement Control Using Colour Detection. 4(1) (2018) 1298-1302.
- [17] A. Khaliq, A. S. Khan, Virtual Mouse Implementation Using Color Pointer Detection. *International Journal of Electrical Electronics & Computer Science Engineering*. 2(4) (2015) 23-32. www.ijeecse.com.