iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56691

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

A Smart Device with Hand Gesture using Webcam

Mr. Krishna Gudi[1], Bhumika Mahadevappa Umachagi[2], Edumudi Abhinaya[3], Guru Kiran S[4], Guru Kashyap N S[5]
[1]*Assistant Professor, Dept of Computer Science, K S Institute of Technology, Bengaluru, Karnataka*
[2, 3, 4, 5]*Dept of Computer Science, K S Institute of Technology, Bengaluru, Karnataka*

Abstract: Over time, Human-Computer Interaction (HCI) has experienced tremendous growth. An entirely new era in the AI domain of human-computer interaction was ushered in by hand gesture recognition technologies. One of the easiest and most natural forms of communication is hand gestures. It might be challenging for novice
With hand gestures, the user can control cursor functions on the system. The input device for the suggested system is a camera. this application, HCI deployment for educational purposes will enter a completely new age.

I. INTRODUCTION

This design will become more portable and tiny as Bluetooth and other wireless technologies advance, along with AR and other everyday gadgets. This research develops a virtual mouse system powered by AI simulate mouse movements on a computer.
Gesture Controlled Virtual Mouse over conventional input techniques are also covered in the study, including how user-friendly and more natural the interaction is. T
he information provided in this article will benefit academics, developers, and everyone else interested in the most recent developments in gesture-based interface technologies, as well as the expanding.
The implementation of the system is based on models like the Convolutional Neural Network (CNN) implemented by Media Pipe running on top of Python, and it doesn't require any additional hardware.

II. RELATED WORKS

A. Gesture with Hands

The user's hand gestures are recorded via the webcam.
Ying and Thomas (2001) have identified various gestures that fulfill distinct functions. "Controlling gestures" are the gestures used for pointing and navigation in virtual environments and other display control applications. The virtual mouse interface (Tsang et al., 2018) is one application of gesture control that allows users to move the cursor with their hands in place of a conventional mouse. Conversational gestures, like pointing to emphasize a point, are essential to human connection.

B. Neural Networks

The Neural networks are a powerful tool for modelling complicated relationships between input and output using non-linear statistical data. Using non-linear statistical data, An effective tool for modeling intricate relationships between input and output.. Sakr (2000) The neural network is a popular option for pattern recognition classification due to its superiority in noise immunity. One well-known method for training a feed forward neural network is the back propagation learning algorithm. Two more learning algorithms train a neural network are the Perceptron and Delta Rule. Alsmadi et al.'s 2009 study examined temporal hand gestures.

C. Palm Detection Model

In this a single shot detector model it is a single shot detector model for hand recognition and position like detection utilizing face mesh. The initial stage is to train a palm detector rather than a hand detector, because identifying hands with moveable fingers is much more difficult to do than estimating the bounding boxes of stiff objects like fists and palms.
Because the hands involved are little, the non-maximum suppression strategy works particularly well in social and self-scenarios like handovers.
Reducing the number of anchors by 4-5 is possible by modeling palms with square anchor boxes (connections in machine learning terminology) and disregarding other aspect ratios. Furthermore, to understand the entire scene context, even for little objects, a feature extractor based on a codec pair is employed (similar to the Retina Net technique).

©IJRASET: All Rights are Reserved

D. Hand Positions

Mediapipe approach overcomes these problems by employing of Once the palm has been identified throughout the entire image, 2D and 3D handknuckle positions inside the observed hand regions can be determined using regression, direct coordinate prediction, or the mediapipe hand landmark model. Even in situations where the hand is partially concealed by the model's body or only visible in part, the model.

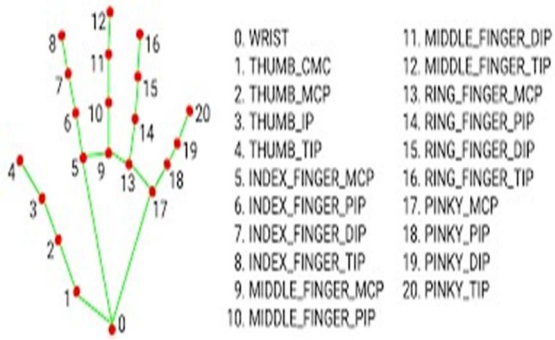


Fig : Hand positions

III. LITERATURE SURVEY

In the previously proposed system by Kabid Hassan Shibly's Hand gestures are not supported in the current environment , so we are trying to design a smart device to control the mouse wirelessly in order to perform this operation , we need all the function such as left ,right and movement of the cursors so , we need python autopy to develop this project and also we are adding new features such as volume and brightness controlling functions its becoming more difficult to use mouse in modern world because of virus . From the paper International Journal of Engineering Research and Applications we referred the content. A hand gesture cannot be used to reach the monitor's display screen from a distance. Even though this is what it mainly seeks to do, the breadth is only limited to the virtual mouse region.

IV. OBJECTIVES

1) We designed and implemented a hand-mouse interface that introduces a new concept called a ‘‘virtual monitor’’, to extract a user's physical features through Kinect in real time.
2) Instead of using more expensive sensors , a simple web camera can identify the gesture and perform the action
3) It can control computer mouse operations.
4) We are implementing cursor function such as Netural Gesture,

- ❖ Move Cursor,
- ❖ Left Click,
- ❖ Right Click,
- ❖ Double Click,
- ❖ Scrolling,
- ❖ Drag and Drop,
- ❖ Multiple Items Selection,
- ❖ Volume Control,
- ❖ Brightness Control.

V. METHODOLOGY

The artificial intelligence virtual mouse system utilizes data from camera images on a personal computer or laptop, the Python computer vision package OpenCV will do video capture object, which in turn triggers the web cam to begin recording.

A. Hand Tracing

A specific number of frames per second, or FPS, is continuously provided by the webcam. A Python library's first is found using Mediapipe. The user's hand gestures are recorded via the webcam. A specific number of frames per second, or FPS, is continuously provided by the webcam.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

B. Cursor control Module

We can use just our two fingers to control the computer mouse and carry out standard mouse operations like dragging, clicking, and moving. The Python language's built-in "autopy" package facilitate cursor operations. Basically, this library is to make mouse actions simple and effective..

C. Hand Detection

Using the mediapipe framework, set the major or minor hand using points, receive the maximum number of hands (2), the minimum detection and tracking confidences (50%) and the finger status.

Pseudo code algorithm for virtual mouse using hand gesture

```
1. start
2.  gesturecontroller_mode=[0,1]                          FIST = 0
3.  capturing videoframe using opencv                     PINKY = 1
4.  Identifying major and minor hand if mutiple hands     RING = 2
5.  classify the left and right hand                      MID = 4
6.  converting image to rgb and flip the image            LAST3 = 7
7.  capturing the fingers state                           INDEX = 8
8.  obtaining gesture with fingers state                  FIRST2 = 12
9.  handling fluctations due to noise                     LAST4 = 15
10. obatained euclidean distance between points x and y axis   THUMB = 16
11. obtained previous mouse location x and y axis         PALM = 31
12. if PINCH_MAJOR
      on x-axis Controller.changesystembrightness   PINCH_MAJOR = 35/right hand
      on y-axis Controller.changesystemvolume
13  else_if PINCH_MINOR
      on x-axis  Controller.scrollHorizontal        PINCH_MINOR = 36 /left hand
      on y-axis  Controller.scrollVertical
14  else_if  GESTURE==FIST
      NO_action
15  else_if  GESTURE==v_gesture                       V_GEST = 33
      find Absolute points of axis
      Mouse_movement
16  else_if  GESTURE==MID
      left_click
17  else_if  GESTURE==index
      Right_click
18  else_if  GESTURE==mid and index finger closed     TWO_FINGER_CLOSED = 34
      Double_click
19  stop
```

D. Hand Gesture

Points for each finger are as follows: fist = 0, pinky finger = 1, ring finger = 2, mid finger = 4, last 3 fingers is open = 7, index = 8, first 2 fingers is open = 12, last 4 fingers = 15, thumb is open = 16, palm finger = 3. Additional mappings are v_gest = 33 v_gesture if index and mid finger in v shape, two_finger_closed = 34 if index and mid finger is join, pinch_major = 35 if right hand with thumb and index finger is join with fingertip, and pinch_minor = 36 if left hand with thumb. Find the current finger state: 1 if the finger is open, 0 otherwise.'Finger' is ratio of the distances between the base, middle, and tip knuckles of the finger. Points for [[8,5,0],[12,9,0],[16,13,0],[20,17,0]] are awarded for the thumb from index. Calculating Euclidean distance between 'point' for finding state of finger and ratio.

$d = \sqrt{[\ (x2 - x1)\ 2 + (y2 - y1)\ 2\ ]}$

E. To control Position of Pointer

With media pipe module, the mouse pointer computer window by raising either the middle finger (tip Id = 4) or the index finger (tip Id = 8). This is where the tracked and detected hands are gathered, with a list of 21 locations representing each hand. The scales applied to the x and y coordinates are [0.0, 1.0] and (x, y, and z coordinates). It is determined by the width and height of the image. Z represents a landmark's depth, with the wrist serving as the beginning point. The lower the value, the closer the landmark is to the camera. A treasure trove of hands that have been tracked and detected, each represented by a list of 21 global coordinate landmarks. X, Y, and Z components of each landmarks are three-dimensional real-world coordinates expressed in meters, where the origin is the approximate geometric center of the hand. data on handedness from hands that have been detected and tracked (e.g., is it a left or right hand).

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

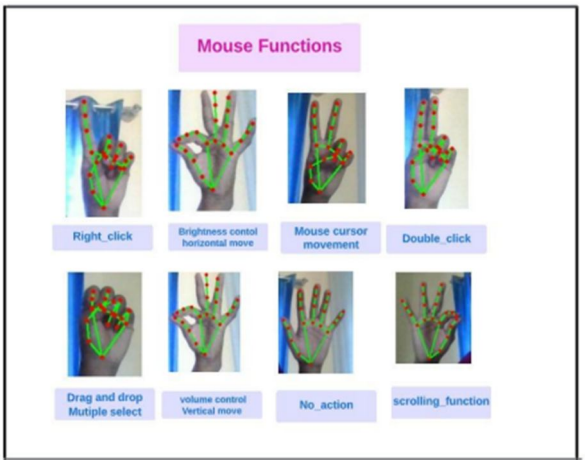


Fig:5.1 Mouse Functions

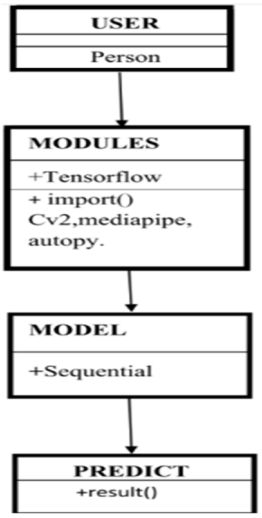


Fig:5.2 Flow Chart

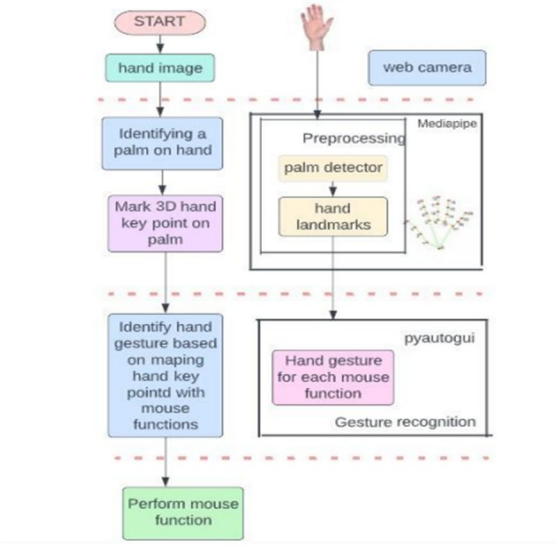


Fig: 5.3 Working Model

1) Application Requriments

a) OpenCV: OpenCV is a library of programming functions mainly for real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage, then Itseez. The library is cross-platform and licensed as free and open-source software under Apache License

b) MediaPipe: MediaPipe is an open-source framework for building pipelines to perform computer vision inference over arbitrary sensory data such as video or audio. Using MediaPipe, such a perception pipeline can be built as a graph of modular components. MediaPipe is currently in alpha at v0.

2) Software Requirements:

The software Requirements in this project include:

a) Python

b) Anaconda prompt

c) Spyder IDE

d) Modules (OpenCV, Mediapipe, Autopsy) The MediaPipe framework is utilised for hand motion recognition and tracking, while the OpenCV library is used for computer vision.

VI. ACKNOWLEDGEMENT

We extend our sincere gratitude to Mr.Krishna Gudi for his invaluable insights and guidance during the planning and execution phases of this project. His generous contribution of time and expertise has been immensely appreciated. Additionally, we express our thanks to all the professors at KSIT for their unwavering support and encouragement throughout this endeavor.

REFERENCES

[1] Zhengyang Wang Jason Gu 2022 Tracking for Human Computer Interface Using Multiple Sensor Data Fusion.

[2] Wahyu Rahmaniar , Alfian Ma'arif , and Ting-Lan Lin 2022 IEEE Touchless Head-Control (THC): Head Gesture Recognition for Cursor and Orientation Control.

[3] Changhyun Jeon, Oh- Jin Kwon , Dongkyoo Shin & Dongil Shin 2017 Hand-Mouse Interface Using Virtual Monitor Concept for Natural Interaction.

[4] Tsang, W.-W. M., Kong-Pang Pun. (2005). A finger-tracking virtual mouse realized in an embedded system. 2005 International Sympo- sium on Intelligent Signal Processing and Communication Systems. doi:10.1109/ispacs.2005.1595526.

[5] Tsai, T.-H., Huang, C.-C., Zhang, K.-L. (2015). Embedded vir- tual mouse system by using hand gesture recognition. 2015 IEEE International Conference on Consumer Electronics - Taiwan. doi:10.1109/iccetw.2015.7216939 10.1109/icce-tw.2015.7216939.

[6] Roh, M.-C., Huh, S.-J., Lee, S.-W. (2009). A Virtual Mouse interface based on Two-layered Bayesian Network. 2009 Workshop on Applications of Computer Vision (WACV). doi:10.1109/wacv.2009.5403082 10.1109/wacv.2009.5403082

[7] Li Wensheng, Deng Chunjian, Lv Yi. (2010). Implementation of virtual mouse based on machine vision. The 2010 Interna- tional Conference on Apperceiving Computing and Intelligence Analysis Proceeding. doi:10.1109/icacia.2010.5709921 10.1109/icacia.2010.5709921.

[8] Choi, O., Son, Y.-J., Lim, H., Ahn, S. C. (2018). Corecognition of multiple fingertips for tabletop human-projector interaction. IEEE Transactions on Multimedia, 1–1. doi:10.1109/tmm.2018.2880608.

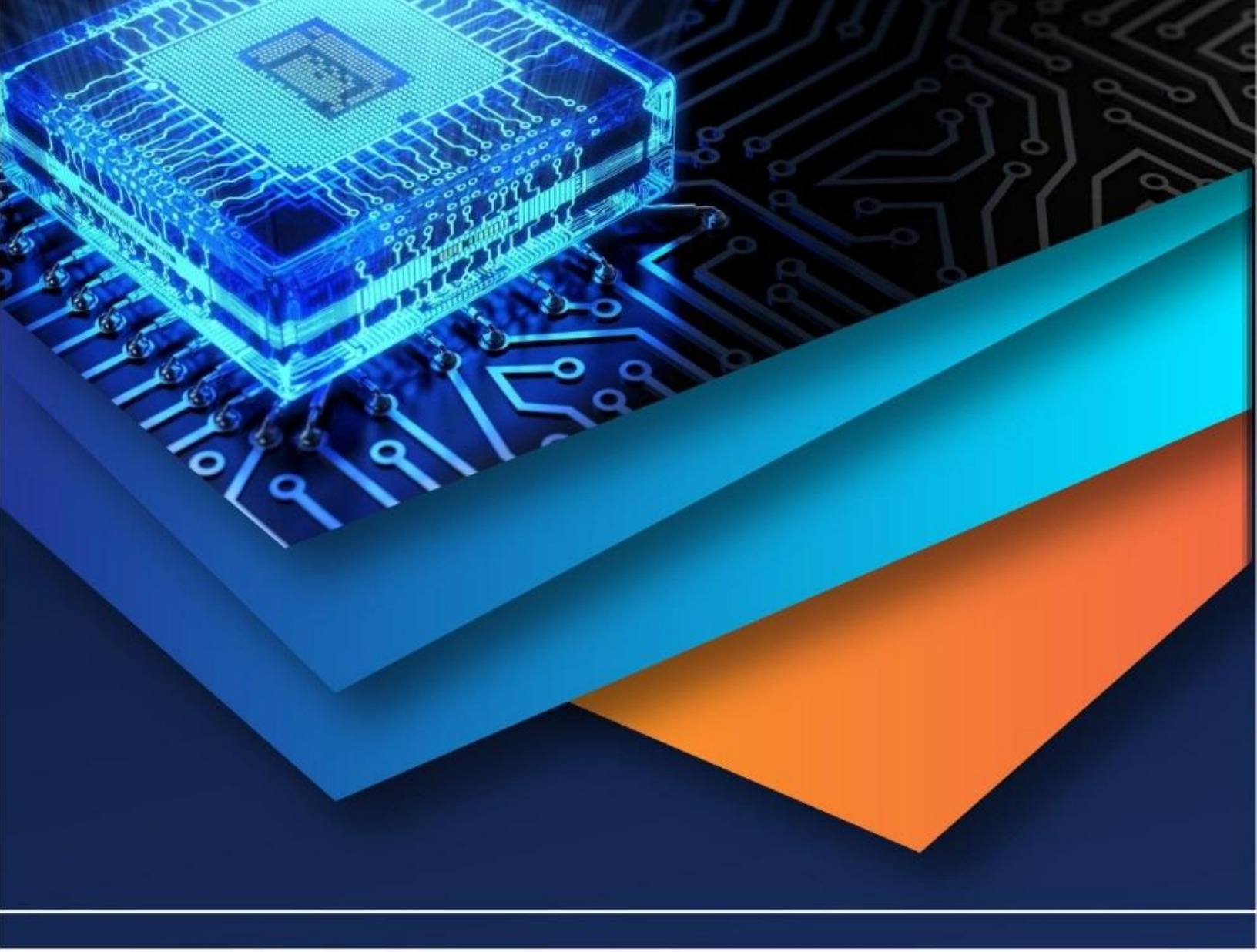
[9] Jyothilakshmi P, Rekha, K. R., Nataraj, K. R. (2015). A frame- work for human- machine interaction using Depthmap and com- pactness. 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT). doi:10.1109/erect.2015.7499060.

[10] S. Vasanthagokul, K. Vijaya Guru Kamakshi, Gaurab Mudbhari, T. Chithrakumar, "Virtual Mouse to Enhance User Experience and Increase Accessibility", 2022 4th International Conference on Inven- tive Research in Computing Applications (ICIRCA), pp.1266-1271, 2022, doi:10.1109/ICIRCA54612.2022.9985625.

[11] Shajideen, S. M. S., Preetha, V. H. (2018). Hand Gestures - Virtual Mouse for Human Computer Interaction. 2018 International Conference on Smart Systems and Inventive Technology (ICS-SIT). doi:10.1109/icssit.2018.8748401.

[12] Henzen, A., Nohama, P. (2016). Adaptable virtual keyboard and mouse for people with special needs. 2016 Future Technologies Conference (FTC). doi:10.1109/ftc.2016.7821782.

[13] Reddy, V. V., Dhyanchand, T., Krishna, G. V., Mahes h- waram, S. (2020). Virtual Mouse Control Using Colored Fin- ger Tips and Hand Gesture Recognition. 2020 IEEEHYDCON. doi:10.1109/hydcon48903.2020.9242677 .



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)