AI And Hand Gesture Recognition Based Virtual Mouse

Telagarapu Prabhakar[1*], Gurram Sunitha[2], Gudavalli Madhavi[3]

Abstract

The mouse is one of the wonderful inventions of Human-Computer Interaction (HCI) technology. Currently, wireless mouse or a Bluetooth mouse still uses devices and is not free of devices completely since it uses a battery for power and a dongle to connect it to the PC. In the proposed AI virtual mouse system, this limitation can be overcome by employing webcam or a built-in camera for capturing of hand gestures and hand tip detection using computer vision. The algorithm used in the system makes use of the machine learning algorithm. Based on the hand gestures, the computer can be controlled virtually and can perform left click, right click, scrolling functions, and computer cursor function without the use of the physical mouse. The algorithm is based on deep learning for detecting the hands.

Key Words: Artificial Intelligence, Python, Hardware, Arduino and OpenCV

DOI Number: 10.14704/nq.2022.20.8.NQ44252 **NeuroQuantology 2022; 20(8):2316-2326**

Introduction

Software technology has improved enormously in the last decade becoming an indispensable part of daily life. The cursor was first piece of Field Of human - computer Interaction (HCI) equipment. This same mouse is unsuitable for HCI sometimes in scenarios, including such Mortal Machine Systems. Several findings on diverse means of managing the desktop computer for Human – computer interaction have been undertaken. Hand signals are by far the most normal and instinctive HCI strategy, and a plausible swap for the computer mouse [1]. Hand movements would be used to perform mouse actions like right clicking, left clicking, scrolling up and scrolling down, as well as hand trailing will be used to manoeuvre the Desktop Workstation cursor [2]. The research has been conducted on a NUI/NUX (natural user interface/natural user experience) recognition algorithm is needed for the gestures. The virtual monitor concept has recently been introduced to eliminate the disadvantages found in the study on an NUI/NUX framework.

Worked on previous work on an NUI/NUX related to hand mouse and the hand-mouse interface design and implementation. Process includes pre-processing, normalization, and feature extraction [3]. Explains how to create the virtual monitor, coordinate the transformation algorithm. In this only a user who was aware of more than 15 gestures could use the interface. The increased interest in human–computer interaction has encouraged more research. A natural user interface/natural user experience (NUI/NUX) that uses a user's motions and voice has also been studied [4]. This approach involves Image processing and Image acquisition technique. In this a green box will be appear on the screen if the code matches the object will be detected with red border around it and makes function. After the camera resembles the hand then it moves to Convex Hull detection. It calculates the contour or say it does the contour analysis for detecting the fingertips. Any sudden movement of the hand or object in front of the camera the program will not consider it as a detectable object as a 1 second delay. There are total four different kind of gesture, one is used to move the cursor, another one is used to do the right click , another one is used to do left click , and another gesture for scrolling up and down. The simulated monitor concept has been established to solve these deficiencies [5].

Corresponding author: Telagarapu Prabhakar
Address: [1,2,3] Electronics and Communication Engineering, GMR Institute of Technology, Andhra Pradesh, India
E-mail: prabhakar.t@gmrit.edu.in

A virtual screen is a "virtual world where mouse arrow could be aimed directly by the fingers here between Touch id as well as the user." The simulated presentation seems to have the capability of giving a frame of reference for attempting to control the move the mouse. Besides that, the digital values precisely represent real position information [6]. The current system consists of a generic mouse and trackpad monitor control system with no hand gesture control. It is not feasible to access the monitor screen with a hand motion from a distance. In the virtual mouse field, the scope is simply limited, even though it is primarily seeking to accomplish. Virtual Mouse Control Using Colored Fingertips and Hand Gesture Recognition [7] One of the research projects in human-computer interaction is a virtual mouse with fingertip identification and hand motion tracking based on image in a live video. The use of fingertip identification and hand motion recognition to control a virtual mouse is proposed in this research. The researchers used two approaches for tracking [8] the fingers in this study: coloured caps and hand gesture detection.

A survey on Hand Gesture Recognition for Simple Mouse Control Human-Computer Interaction (HCI) is an intriguing field of study. It is the study of how people interact with computers and to what extent computers are or are not designed to interact successfully with humans. The accelerometer-generated hand motion acceleration signals are collected by the micro - controller. Hand noise is produced because of slight quivering movements [9]. The convex hull algorithm is used to solve the problem of determining the largest polygon with all vertices. This algorithm can detect fingertips on the hand. This algorithm produces whether a finger is folded. To acknowledge this, the hand radius value is multiplied twice and the distance between the centre and a pixel is checked to see if it is in the convex hull set [10].

The purpose of linear discriminant analysis (LDA) is to reduce the dimensionality of the feature space while improving recognition performance (i.e., to reduce the size of m). The goal of feature selection and feature extraction is not only to reduce computational burden but to improve classification accuracy. The reduced features are fed into classifiers as inputs. LDA is used to reduce the dimension of the selected features to reduce computational load and increase the recognition performance of the classifier [11]. The reduced feature vectors are then fed into a PNN classifier to determine which motion the feature vector belongs to If the distance exceeds the radius of the hand, larger is spread. A check is performed when the convex hull produces one more result vertex. The convex hull method computes the vertices of a circle greater than multiple times the size of the hand as the threshold. One of the most prominent areas of research in the field of human-computer interaction is the design and implementation of virtual mice. The technological advancements that have occurred in the camera industry have greatly reduced the cost of these systems [12].

The K-means clustering algorithm is used in the paper to partition the input image for segmentation. To determine the orientation, the bounding box is used. For detection, features such as centroid and Euclidean distance are measured. Seven bits are used to represent the hand. The first bit represents the hand's orientation [13]. The second bit accounts for the thumb's presence in the figure. The next three bits are for displaying the number of raised fingers. Furthermore, they can readily offer the precise location of palm and finger placements, orientation, and combinations utilising embedded sensors to the gloves. However, this method requires the user to be firmly attached to the laptop, which restricts the ease with which humans and computers can interact. Furthermore, these devices are rather expensive [14].

In this paper, we propose an algorithm for detecting fingers, recognising gestures, and controlling mouse operations. This study is divided into three sections: colour recognition, gesture recognition, and virtual mouse control, as well as a general system description. The Python programming language is used to implement the algorithm in this research. For image processing, we utilise the OpenCV library, and for mouse control, we use the pyautogui package. For mouse control, the algorithm employs two different approaches. One involves the employment of colour caps, while the other involves the recognition of bare hand gestures. The primary goal is to create a solution for finger tracking in the real world, while cursor control on a computer is still done physically. In most applications, physically controlling the mouse may be tricky. We can utilise a webcam and some algorithms to control cursor activities without physically touching the mouse. This paper describes the development and analysis of real-time finger tracking for making gestures, so that gestures can be utilised in a variety of mouse applications such as movement, single click, double click, right click, and scrolling.

Methodology

The objective of this paper is to create a Virtual Mouse that focuses on a few key development areas. To begin, this project intends to do away with the requirement for a physical mouse while allowing users to interact with the computer system via webcam utilising various image processing techniques. Aside from that, the goal of this project is to create a Virtual Mouse that can work on a variety of surfaces and environments.

To programme the camera to capture images in a continuous loop, which will then be analysed using various image processing techniques. As previously stated, the Virtual Mouse application will continuously capture photos in real time, undergoing a variety of processes such as Binary Image conversion, HSV conversion, and others.

To create a system that works with a webcam. The Virtual Mouse application will work with the help of a webcam, which will capture images in real time. If there is no webcam detected, the application will not work.

To create an input that works on any surface. If the user is facing the webcam while performing the motion gesture, the Virtual Mouse programme will work on any surface and in any interior situation.

To transform a hand gesture or motion into mouse input for a certain screen position. The position of the mouse points will be detected by the Virtual Mouse application, which will be set as the position of the defined colours. Furthermore, a combination of different colours can cause a variety of mouse events, including right/left clicks, scrolling up/down, and more.

An algorithm to identify fingers, recognising gesticulations, and attempting to control cursor operational processes in just this document. This paper is divided into three sections: hand recognition, motion tracking, and virtual transgenic mice regulation, as well as a summary of the mechanism. The Python language is being used in the manuscript to enforce the automated system. Designers are using the Opencv for image analysis and the pyautogui library for mouse cursor. The method computes two approach to implementing mouse power. First one requires the use of fingers, whereas the second requires the acknowledgment of bare hand signals. Techniques and algorithms used Hand gesture detection and tracking are handled by the MediaPipe structure, and technology is handled by the OpenCV library. The algorithm uses computational concepts to track and recognise hand movements and fingertips.

Media Pipe is a Try googling popular open-source schema used in deep learning pipework. The Media Pipe guideline is best suited for cross-platform advancement because it is based on time series data. The Media Pipe blueprint is multimodal, that means it can operate with such a diverse variety of sound - visual styles. Developers use the Media-Pipe framework to build and analyse systems using graphs, as well as to develop systems for application purposes. The pipeline configuration is where the steps in a Media Pipe-based system are carried out. The pipeline created can run on a variety of platforms, allowing for mobile and desktop scalability. The Media Pipe approach has three main steps: performance assessment, a template for fetching sensor readings, as well as a reuse compendium of structures known as graphs. A keystone xl is a plot constituted of computer modules that are provided a link by channels thru which packets of data flow.

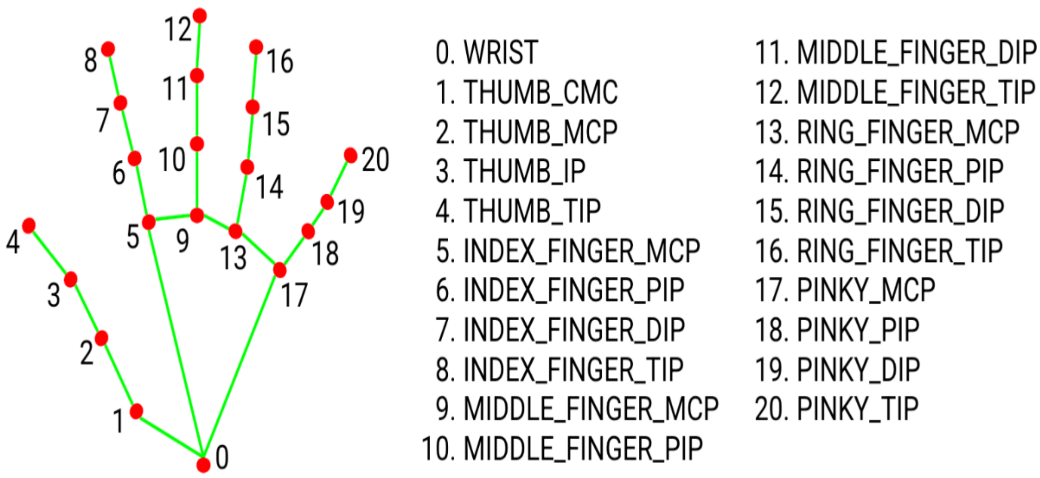

Fig. 1: In the hand, co-ordinates, or landmarks

Developers can create their own application by replacing or defining custom calculators anywhere in the graph. The calculators and streams combined create a data-flow diagram; the graph was created with MediaPipe, and each node is a calculator, with streams connecting the nodes.In order to detect and recognise a hand or palm in real time, a single-shot detector model is used. The MediaPipe employs the single-shot detector model. It is first trained for a palm detection model in the hand detection module because palms are easier to train. Furthermore, on small objects like palms or fists, the non-maximum suppression works much better. Locating joint or knuckle co-ordinates in the hand region is a model of hand landmark. OpenCV is a computer vision library that includes object detection image processing algorithms. The computer vision library OpenCV is a python programming language library that can be used to create real-time computer vision applications. The OpenCV library is used to process images and videos as well as perform analysis such as face and object detection.

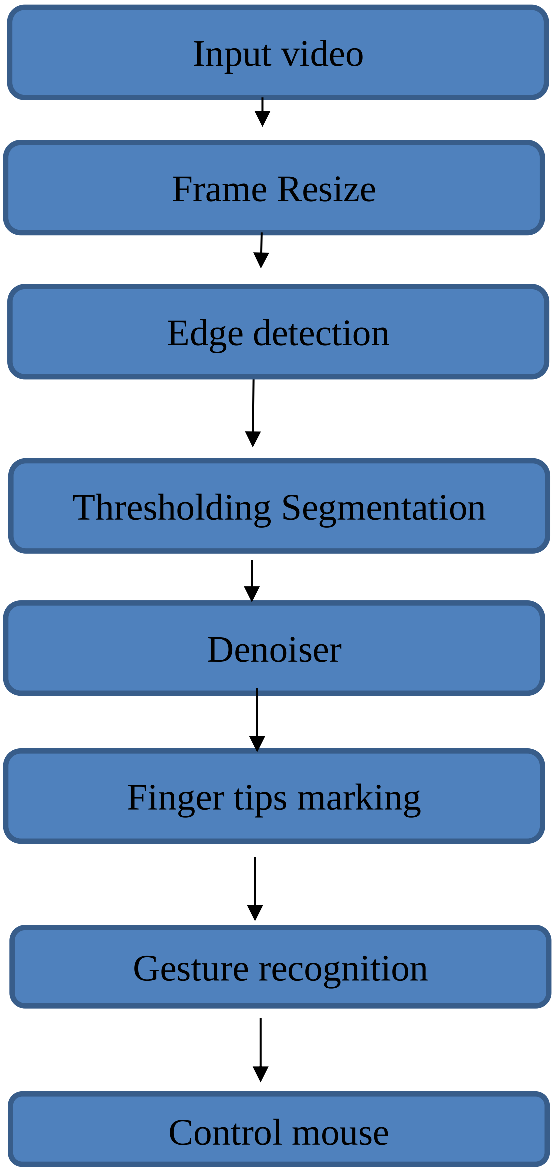

Fig. 2 Overview of the System

Figure 2 shows Overview of the system. It is divided into two methods: "fingertip detection" with coloured caps and "gesture recognition." It entails video interfacing and image processing via background subtraction. Background subtraction aids in ignoring stable objects in favour of focusing on the foreground. Fingertip detection entails colour identification, circle identification, and finger guessing. Gesture recognition entails identifying skin colour, contour detection, and convex hull formation before guessing the gesture. The appropriate mouse operation can be carried out. The following procedure provides the information about Overview of the system.

1) Video Extraction: Initially, the webcam video is captured using the Video Capture function in the OpenCV library. The extracted video is 512x512 pixels in size. Then, starting from A frame from the video is extracted and used for further processing the image's processing The ideal number of frames. There are 12 extracted per second. The frame that was extracted the image is in the BGR (Blue, Green, and Red) format. This is an illustration. HSV (Hue, Saturation, Value) format was used. OpenCV's BGR to HSV conversion function.

2) Background Removal: The extracted frame image It contains several objects. However, it is only the foreground that Processing necessitates the presence of the hand with colour caps. As a result, the background must be ignored. So, first, some context. OpenCV subtraction algorithm So, first, some context. The OpenCV subtraction algorithm is required for the Object extraction in the foreground Background subtraction is used in the current frame image is compared to a 21st century image using an algorithm. In the past, there was a prior frame image. The objects that lacked Motion in 20 or more frames is the best. As a result, background objects are overlooked. Since then, the motion While performing the, one of the colour caps is always present mouse operations are in the foreground objects.

The hand shadows in the frame image are omitted using the command detect shadows in the algorithm for background subtraction following the background we get a masked greyscale image because of the subtraction algorithm containing the objects in the foreground This is the bitwise 'and' of this. The combination of a greyscale image and the original frame image yields So, first, some context. The OpenCV subtraction algorithm is required for the object extraction in the foreground Background subtraction is used in the current frame image is compared to a 21st century image using an algorithm. In the past, there was a prior frame image. The objects that lacked Motion in 20 or more frames is the best. As a result, background objects are overlooked. Since then, the motion While performing the, one of the colour caps is always present. mouse operations are in the foreground. objects. The hand shadows in the frame image are omitted using the command detect shadows in the algorithm for background subtraction Following the background We get a masked greyscale image because of the subtraction algorithm. containing the objects in the foreground This is the bitwise 'and' of this. The combination of a greyscale image and the original frame image yields.

3) Finger Identification: The background-subtracted image is scanned to obtain the colour caps of a required colour. In HSV (Hue, Saturation, Value) format, the lower and upper ranges of a specific colour are given as the 'in range' function in the OpenCV library is used to extract a greyscale image containing only the objects of the required colour. The bitwise 'and' of this greyscale image with the background subtracted frame image yields the image with only the objects of the required colour. This is used to identify the required colour cap among a variety of fingers.

4) Tips Identification: After identifying the objects of a specific colour that are in motion, the shape of the objects that coincide with the colour caps is identified. The fingers used in the project are tips in shape.As a result, objects with tips shapes are identified. To identify the tips, OpenCV's Hough Circles function is used. The radius of the circle is specified in a specific range to identify colour caps placed at the optimal distance from the camera. The range values range from 50 to 200 pixels, corresponding to 0.5 m to 1 m. The Hough tip function returns a two-dimensional list containing the radius and centres of all the coloured circles in the image, which can be drawn using the co-ordinates.

5) Recognition of Gestures: Finger identification can be accomplished by smoothing frames with the bilateral filter function and then subtracting the background of images. Each frame is then checked against the colour range, and the contours' function is used to obtain the palm's bounding shape, and the convex hull is formed using the convex hull function. The Contours function is used to get the outline of a hand's palm. It is done by comparing the colour of the palm to the colour range specified, and then an approximate boundary of the palm is detected and drawn, yielding an outline of the palm as a contour. The convex envelope of a point set A is the smallest convex that contains A. To form the hull of the entire contour, the convex hull technique is used. The points on the contour are checked to ensure that all of them are inside the hull, with the extreme, outermost points serving as some of the hull's points. The normal image of a palm is depicted HSV

image created from a normal image. The image for finger detection, where green lines are contours along the palm and red lines are obtained using the convex hull technique. The contour area and convex hull area formed by the palm are used to identify various hand gestures. The area ratio illustrates a specific gesture. The number of peaks formed in the contour indicates the number of fingers. This is the number of fingers that can be opened freely. The areacnt is the palm of the hand, and the area hull is the hull formed by the palm.

6) Cursor moment: The final step before deciding is to move the cursor. Movement is implied by the centroid in the above image. It has a set(0,'PointerLocation',[x,y]) function for moving the cursor to desired (X,Y) coordinates. It doesn't have any functionality for clicking events. Python class is used to move the mouse and simulate the mouse click event. All these skills will be used by a robot. The sensor decision is commensurate to the negotiated settlement of the pointing device. As both an outcome, putting money in a relatively high photograph will be more helpful. The image is 640x480 in size, and the computer is 1280x800 in size. If the resolution of the webcam is lower than with the monitors, it should be. Anatomy is a collection of filtering techniques that control images based on their shape and size. A constructing element is supposed to apply to an information image to obtain an equivalent image representation. The input image is compared to its neighbours, yielding a value for the output image. The neighbourhood factors, such as shape and size, can be determined by the programmer, resulting in the construction of programmer-defined morphological operations for the input image.

A most foundational segmentation algorithms are bulging and depletion. In a picture, deformation adds screens towards the edges of objects, whilst erosion takes away pixels first from outsides of things. Guess it depends on the scaling factor used, the number of images added to the system will vary. During morphometric erosion and dilation processes, the attributes of any image pixel have been determined utilizing a strict provision to the commensurate neighbouring pixels in the source images. The action seems to be dilation or erosion contingent on the rule implemented to the image pixel. Hand signals can be used to provide feedback to the video camera in the proposed system.

The web camera caught the photos of the arm movements. Gestures are acknowledged and transferred into messages by image sensor. As an information, the impulses will be sent to the control system. The signals are thoroughly checked by the controller. As a result, the final processed input is passed to the hardware, which performs the given task and produces the required output. Waiting, collecting, manipulating, and executing are some of the main processes that occur during processing. Whereas waiting refers to the user's inability to perform a gesture, collecting refers to the machine's gathering of information, processing refers to information manipulation, and execution refers to task execution.

7) Left and right click: Both fingers are used for left clicks and two fingers together is used for right clicks in this work. At the beginning, the hand is fully opened so that the fingers can be expanded to their maximum length, which can be calculated during the acquisition of the first frame, which is treated as the mother image. Now, a threshold length for clicking is set to 60%, and whenever a left click is required, a blue colour finger is folded, which is detected in the image and treated as a baby image. If the length of the finger in the baby image is less than the threshold length, it will perform a left click. A lively motion is the one which changes constantly, while a static sign is one that is observed at such a fixed moment in time.

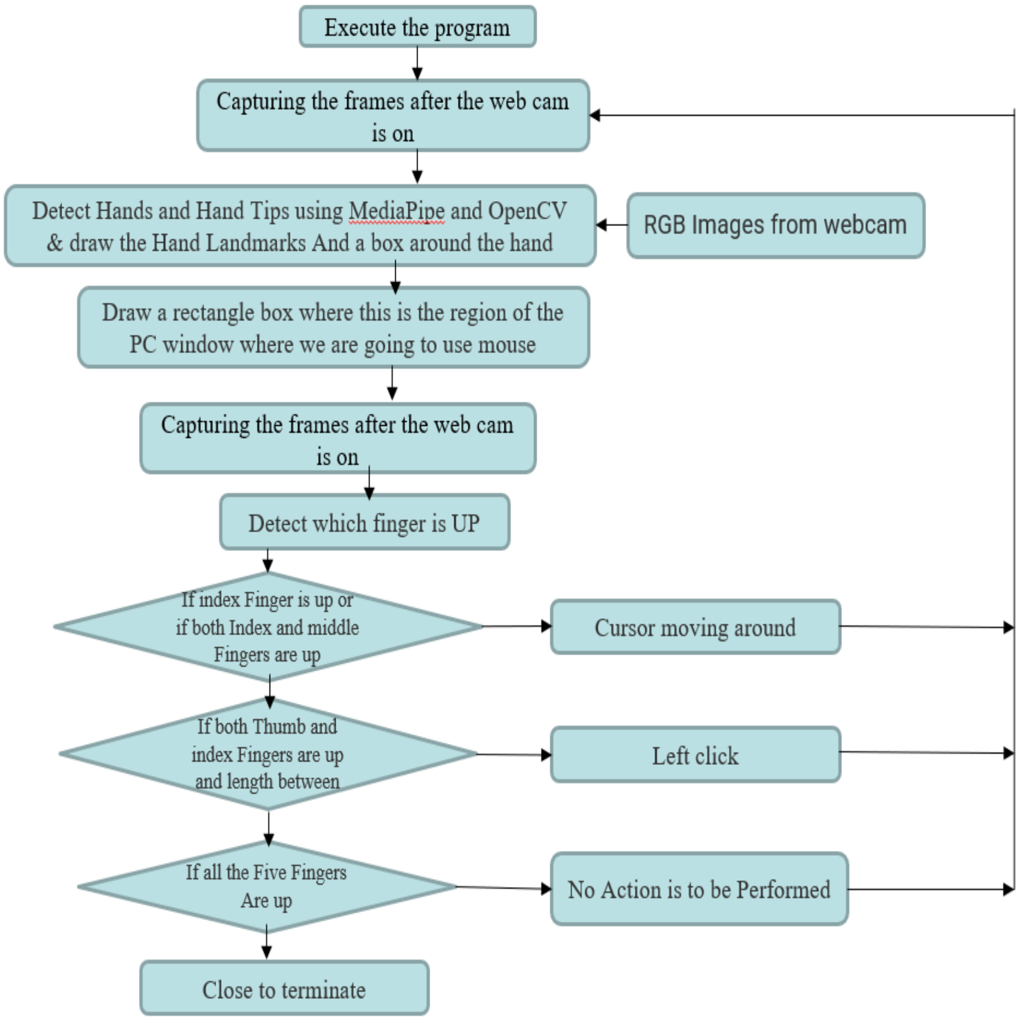

Fig. 3: Flow chart for virtual mouse

Approach to accuracy: To Obtain accuracy the following steps need to follow. Figure 3 shows the flow chart of virtual mouse and figure 4 shows Process for extraction and execution.

Extraction of Characteristics: The movement of the artefact does provide important data for object localization but also retrieval. To obtain mobility info, we suppose that its input sign is semi. A motion sensor could perhaps track an object's mobility in space to hold by analysing locations modifications.

Thresholding: We use frame disparity standards and criteria to derive the possible moveable region in the cluttered backgrounds after extracting the motion region. For order to detect motion distinction, Mean shift binarization is inefficient. As a result, tissue pixel recognition with Grayscale is personally prefer for background subtraction.

Skin colour recognition: This same restriction R>G>B could be used to find color of skin regions, that can include red, fuchsia, brown, as well as bright yellow. A tech of tissue sections is used to capture the dynamic kinds of skin pixels with ring principles for R, G, and B pixel resolution. To discover skin areas, try comparing the colour combinations inside the area to the targeted sample colour. If the regions seem to be related, the province can be alluded to as the surface layer.

Detecting the edges: It's being used to differentiate between the hand and arm areas. The arm area has very few outer edges than just the left-hand region, which would be obvious. To obtain different route edges, the Kirsch tool called tactic is being used, and then the maximum available value of each camera sensor is selected to form the image representation.

Hand gesture tracking on a local scale: To locate a rather more precise hand territory, are using the current frame region details. The touch stance was decided utilising motion, ethnic background, but also texture features. This same edge, body movement, and complexion hand placement is not always in centre of the ungloves hand area. So tracking is done by hand. Motion data and

foreground data are separated into two stages. The following is the local tracking method.
1. Choose the high contrast and skin image area near the inaccurate central core.
2. Choose foreground boundary points.
Recognition of hand signs: The Process of extraction and execution shown in figure 4.

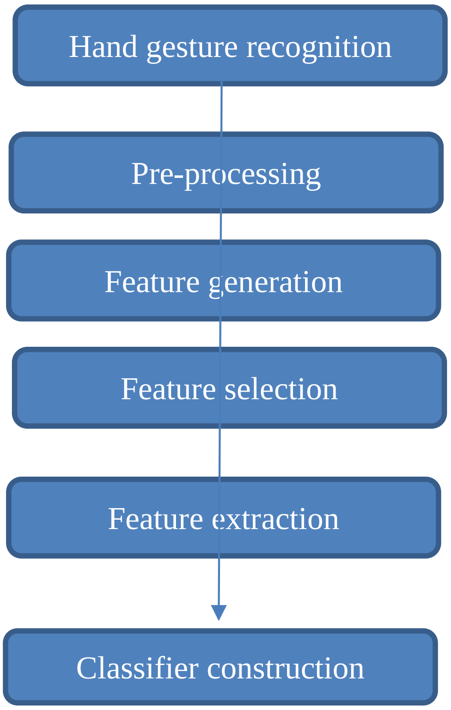

Fig. 4 Process of extraction and execution

Hand Detection: We studied gesture recognition system using simple cursor manipulation. The palm is by far the most major part, so it would be simple to separate it using segmentation methods. This type of separation is more like human understanding as the eyes will simply differentiate skin from its background. This classic technique for separating skin pixels, which specifies top and bottom bound values, was used to segment the hand.
Recognition of Gestures: The image's biggest contour is extracted first. The centre coordinate of the hand would be determined. The size of the hand can be determined by sketching a circle and expanding the diameter from the centre coordinate. The fingertips are recognised using the convex hull technique. The cubic spline algorithm is used to tackle the challenge of finding the biggest polygon containing all vertices. This algorithm can detect the tips of the fingers on the hand. This algorithm determines if any finger is folded or not. To identify this, multiply the hand radius by two and verify the length from the midpoint to pixel set.
Noise Reductions: To acquire a better estimation of the hand, we need to eliminate noisy pixel from the image. A photo morphing algorithm is used to eliminate noise by performing picture erosion and dilation. Erosion reduces the input image where the palm is not present whereas Dilation widens the region of the pixel that are not eroded. The microcontroller collects palm motion acceleration signals generated by the accelerometer.
Slight trembling motions cause hand noise to be created. Signal pre-processing includes calibrating, a moving filter, a high pass filter, along with normalisation. To remove errors and offset from the raw data, the accelerations are first calibrated. An average filter is used to decrease the high - frequency components of calibrated accelerations. High-pass filters are utilized to take away gravitational acceleration from filtered acceleration to retrieve accelerations induced by hand movement.
After correctly filtering the data signal, segment each movement to obtain the exact motion interval. Finally, each motion period is adjusted into equal sizes using interpolation. The properties of distinct hand movement signals can be determined by extracting eight elements of tri axis accelerating signals, comprising average, STD, VAR, IRQ (cross range), correlations across axis, MAD,

RMS, and energy. The variable selection, attribute selection, or variable set selection are all terms for the process of picking a set of attributes for use in model creation.

It's a method based on the assumption that the information includes a lot of duplicate or useless material. Redundant features provide no further data than the one that are picked. Irrelevant features don't supply any useful information. The process of selecting a subset of relevant features for use in model construction is also known as variable selection, attributes selection, or variable subset selection. It's a technique that assumes the data contains a lot of redundant or irrelevant information.

Features that are redundant provide no more information than the ones that are currently selected. Features that are irrelevant provide no useful information. LDA is a technique for reducing the dimension of the feature space while improving recognition performance (i.e., to reduce the size of m). The goal of feature selection and feature extraction methods is to reduce computational load while also improving classification accuracy.

The reduced features are used as classifier inputs. LDA is used to reduce the dimension of the selected features to reduce the computational load and improve the classifier's recognition accuracy. The reduced feature vectors are then fed into a PNN classifier, which determines which motion the feature vector belongs to. sign is a red light. To understand fully a signal, all stationary and non - stationary gestures must be understood bar stools. kitchen time. This complex method is known as gesture control.

Results

The proposed AI virtual mouse system is based on the frames captured by a laptop or computer's webcam. The video capture object is created using the Python computer vision library OpenCV, and the web camera will begin capturing video, as shown in figure 2. The frames are captured by the web camera and sent to the AI virtual system. The AI virtual mouse system makes use of a webcam to capture each frame until the programme is finished. To find the hands in the video frame by frame, the video frames are converted from BGR to RGB colour space. Rectangular Region for Moving Through the Window (Virtual Screen Matching). The transformational algorithm is used by the AI virtual mouse system to convert the coordinates of the fingertip from the webcam screen to the full-screen computer window for controlling the mouse. When the hands are detected and we determine which finger can perform the desired mouse function, a rectangular box is drawn in the webcam region in relation to the computer window, where we move the mouse cursor around the window, as shown in figure 4.

To detect finger, tip of the finger we found using MediaPipe as well as the corresponding coordinates of a fingers that are raised, as in Fig 6, and conduct the necessary mouse function. For navigating the computer with pointer. The cursor pointer is made to travel all around computer using Python's AutoPy if the forefinger's tip-Id 1 is up or both the forefinger with Id 1 and centre finger with Id 2. For moving the mouse cursor around the computer window. If the index finger with tip Id 1 is up, or both the index finger with tip Id 1 and the middle finger with tip Id 2, the mouse cursor is made to move around the computer window using Python's AutoPy package.

The computer performs the scroll up cursor function uses Python PyAutoGUI both index and the middle finger are up and distance between fingers is greater than 40px, as illustrated in figure 4. To perform a left button, click with the mouse. The computer is configured to control a left mouse click click using Python pynput, if the thumb-tip Id 4, index-tip Id 8, middle Id 12, ring-tip id16, and little with tip id 20 seem to be up as well as the distance between two pointing fingers is below 30px, as in Fig 6 and 7.

In the proposed AI virtual mouse system, the concept of advancing the human-computer interaction using computer vision is given. Cross comparison of the testing of the AI virtual mouse system is difficult because only limited numbers of datasets are available. The hand gestures and fingertip detection have been tested in various illumination conditions and been tested with different distances from the webcam for tracking of the hand gesture and hand tip detection. An experimental test has been conducted to summarize the result shown in below table.

Table 1: Hand Gesture Analysis

Hand tip Gesture	Mouse function performed	Success rate	Failure	Accuracy
Tip ID 1 is up	Mouse moment	100	0	100%
All five tip IDs 0, 1, 2, 3, and 4 are up	No action to be performed	95	5	95%
Tip IDs 0 and 1 are up and the distance between the fingers is <30	Left click	90	10	90%

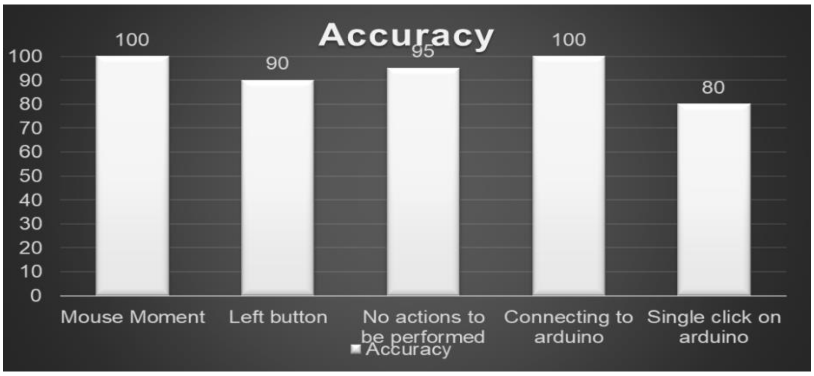

Fig. 5 Performance accuracy

Accuracy = no. of predicted outcomes/ total no. of instance taken

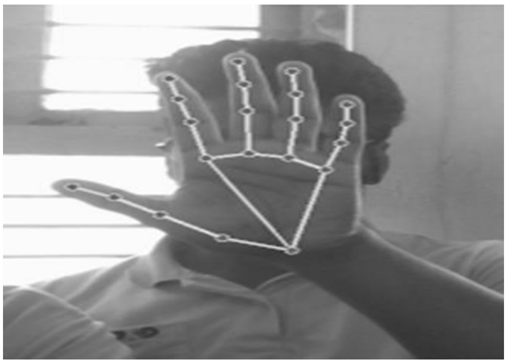

Fig.6 No Action to be Performed on the Screen

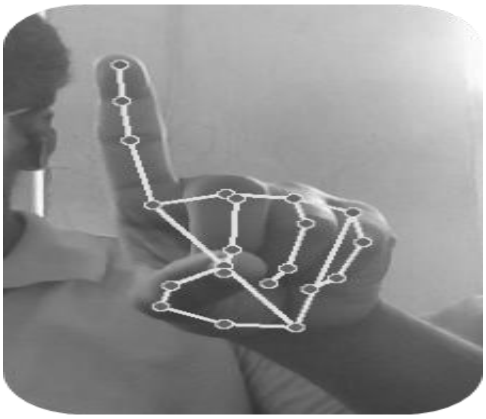

Fig. 7 Detecting Which Finger Is Up and Performing the Particular Mouse Function.

Fig.8 Connecting to Arduino through virtual mouse

The signals are thoroughly checked by the controller. As a result, the final processed input is passed to the hardware, which performs the given task and produces the required output as shown in figure 8. The proposed AI virtual mouse system had achieved an accuracy of about 99%. From this 99% accuracy of the proposed AI virtual mouse system, we come to know that the system has performed well. As seen in table, the accuracy is low for "left Click" as this is the hardest gesture for the computer to understand. The accuracy for right click is low because the gesture used for performing the mouse function is harder. Also, the accuracy is very good and high for all the other gestures. Compared to previous approaches for virtual mouse, our model worked very well with 99% accuracy.

References

Chowdhury, Sugnik Roy, Sumit Pathak, and MD Anto Praveena. "Gesture Recognition Based Virtual Mouse and Keyboard." 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184). IEEE, 2020.

Shriram, S., et al. "Deep learning-based real-time AI virtual mouse system using computer vision to avoid COVID-19 spread." Journal of healthcare engineering 2021 (2021).

Varun, Kollipara Sai, I. Puneeth, and T. Prem Jacob. "Virtual mouse implementation using open CV." 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI). IEEE, 2019.

Shetty, Monali, et al. "Virtual Mouse Using Object Tracking." 2020 5th International Conference on Communication and Electronics Systems (ICCES). IEEE, 2020.

Soroni, Faria, et al. "Hand Gesture Based Virtual Blackboard Using Webcam." 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2021.

Jayasathyan, S., V. Shanjai Sethupathy, and M. Bharathi. "Implementation of Real Time Virtual Clicking using OpenCV." 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS). Vol. 1. IEEE, 2022.

Joshi, H., R. Litoriya, and D. Mangal. "Design of a Virtual Mouse Using Gesture Recognition and Machine Learning." (2022).

Zhang, Zixuan, et al. "Artificial Intelligence‐Enabled Sensing Technologies in the 5G/Internet of Things Era: From Virtual Reality/Augmented Reality to the Digital Twin." Advanced Intelligent Systems (2022): 2100228.

Raj, Shaorya, et al. "Human Computer Interaction using Virtual User Computer Interaction System." International Journal of Performability Engineering 18.6 (2022).

Souza, Kennedy Edson Silva de, et al. "An Evaluation Framework for User Experience Using Eye Tracking, Mouse Tracking, Keyboard Input, and Artificial Intelligence: A Case Study." International Journal of Human–Computer Interaction 38.7 (2022): 646-660.

Souza, Kennedy ES, et al. "User experience evaluation using mouse tracking and artificial intelligence." IEEE Access 7 (2019): 96506-96515.

Tula, Antonio Diaz, et al. "Heatmap explorer: An interactive gaze data visualization tool for the evaluation of computer interfaces." Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems. 2016.

da Silva, João Batista, and Artur Henrique Kronbauer. "A new approach to identifying the potentialities of User Experience techniques." Proceedings of the 24th Brazilian Symposium on Multimedia and the Web. 2018.

Veriscimo, Erico de Souza, Joao Luiz Bernardes Junior, and Luciano Antonio Digiampietri. "Evaluating user experience in 3D interaction: a systematic review." XVI Brazilian Symposium on Information Systems. 2020