International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.4., 2015
(July-Aug)

RESEARCH ARTICLE

ISSN: 2321-7758

DESIGN OF REAL TIME VISION BASED POINTER INTERFACE SYSTEM USING MATLAB

JADHAV PRIYANKA P[1], Dr. KANSE Y.K[2]

[1]Student, Department of Electronics Engineering, Karmveer Baurao Patil College of Engineering and Polytechnic, Satara, Maharashtra, India.

[2]Associate Professor, Department of Electronics Engg., Karmveer Baurao Patil College of Engineering and Polytechnic, Satara,Maharashtra, India.

JADHAV PRIYANKA P

ABSTRACT

In the present world, the interaction with the computing devices has advanced to such an extent that as humans it has become necessity and we cannot live without it. The technology has become so embedded into our daily lives that we use it to work, shop, communicate and even entertain our self. To efficiently use the existing techniques, most computer applications require more and more human to computer interaction. For that reason, human-computer interaction (HCI) has been a lively field of research in the last few years. Real time vision-based pointer interface systems can identify different hand gestures from video input and use them as artificial commands, which computers can understand and respond to. Real time vision-based pointer interface systems can provide an intuitive communication channel for human to computer interaction and able to substitute computer mouse. This reduces hardware impact of the system and also increases range of usage of physical word objects instead of digital word objects such as mouse.

Keywords- HCI, VE.

I. INTRODUCTION

Human-computer interfaces (HCI) have developed from text-based interfaces through 2D graphical-based interfaces, multimedia-supported interfaces, to fully fledged multimodal based 3D virtual environment (VE) systems. While providing a new sophisticated paradigm for communication, learning, training and entertaining, VEs also provide new challenges for human-computer interaction. The traditional 2D HCI devices such as keyboards and mouse are not adequate for the latest VE applications. Instead, VE systems provide the opportunity to integrate different communication modalities and sensing technologies together to provide a more immersive user experience [1].

The devices that can sense body position and orientation, speech and sound, facial and gesture expression and other aspects of human behavior can be used for more powerful and effective interactions between human and computers. To achieve natural and immersive human-computer interaction, the human hand could be used as an interface device. Hand gestures are a powerful human to human communication channel, which forms a major part of information transfer in our everyday life. Hand gestures are an easy to use and natural way of interaction. For example, sign languages have been used extensively among speech-disabled people. People who can speak also use many kinds of gestures to help their communications. However, the expressiveness of hand gestures has not been fully explored for human to computer interaction. Compared with traditional HCI devices, hand gestures are less intrusive and more convenient for users to interact with computers and explore the 3D virtual worlds. Hand gestures can be used in a wide range of applications such as conventional human to computer interactions through linguistic gestures (e.g. various sign languages) and objects manipulation in VEs[1].

Human-computer interaction (HCI) involves a variety of problems related to designing the communication between humans and machines. First of all, the HCI interfaces are based on hardware devices (e.g. a mouse or a keyboard), but they also require a software component which converts the signal delivered by the device to information that controls the machine. Recently, the touch-screens have gained on popularity. Here, the interaction is based on an illusion that the user manipulates physical objects on the screen, which provides a considerable speedup between the user's intentions and machine's action. There are also a number of attempts to rely exclusively on the visual information and create a touch-less HCI. This would reduce the hardware layer to a simple camera, but it requires advanced computer vision algorithms for gesture analysis [3].

As computers become more pervasive in society, facilitating natural human computer interaction (HCI) will have a positive impact on their use. Hence, there has been growing interest in the development of new approaches and technologies for bridging the human–computer barrier. The ultimate aim is to bring HCI to a regime where interactions with computers will be as natural as an interaction between humans [1]. Conventionally keyboard or mouse is being used as mode of input. To achieve natural and immersive human-computer interaction, the human hand could be used as an interface device. Hand gestures are a powerful human to human communication channel, which forms a major part of information transfer in our everyday life. Hand gestures are an easy to use and natural way of interaction. For example, sign languages have been used extensively among speech-disabled people. People who can speak also use many kinds of gestures to help their communications [2].

The latest computer vision technologies and the advanced computer hardware capacity make real-time, accurate and robust hand tracking and gesture recognition promising. Many different approaches have been proposed such as appearance-based approaches and 3D hand model-based approaches. Most of these approaches deal the hand gesture as a whole object and try to extract the corresponding mathematical description from a large number of training samples. These approaches analyze hand gestures without breaking them into their constituent atomic elements that could simplify the complexity of hand gestures. As a result, many current approaches are still limited by the lack of speed, accuracy and robustness.

They are either too fragile or demand too many prerequisites such as markers, clean backgrounds or complex camera calibration steps, and thus make the gesture interaction indirect and unnatural. Currently there is no real-time vision-based pointer interface system that can track and identify hand movements in a fast, accurate, robust and easily accessible manner. The goal of this work is to build a real-time vision based pointer interface system for the purpose of human-computer interaction. To achieve this goal, the hand movements need to be taken into account and the principles that can improve the system's performance in terms of speed, accuracy and robustness need to be applied[3].

The real time vision based pointer interface system presents a new solution for human-computer interaction that utilizes hand gestures for intuitive operation without any devices restraining the hand freedom. This system is sufficiently fast and effective to substitute a computer mouse. Virtual interaction between laptop and user without using hardware like mouse makes this entire system user friendlier and simpler. Adding sound response to the system makes this system helpful to the visually impaired persons. Solution is placed between conventional pointing devices and touch-screens in terms of user experience, giving the users an illusion that they are interacting with physical objects [3].

II. LITERATURE REVIEW

The hand gestural equivalents of direct manipulation interfaces are those which uses hand gesture alone. It can be ranged from interfaces those recognize a few symbolic hand gestures to those that implement fully fledged sign language interpretation. Interfaces can recognize static hand poses, dynamic hand motion, or a combined combination of both. In all cases each gesture has an unambiguous semantic meaning associated with it for recognition. The development of hand gestures recognition has been travelled a quite a long path in a very short period of time with the improvement of science and technology and research of many scientists.

The gesture-based HCI development in recent years introduced several interesting interaction interfaces. They are based on two main approaches to the interaction. The first one relies on the hardware and sensors, for example a data glove. Precision and accuracy is a big advantage of this equipment. It measures directly the hand position, its orientation, fingers directions and angles. However, its price is very high and out of range for an ordinary computer user. Moreover, the equipment reduces the comfort and naturalness of the interaction due to the hardware that must be worn [4].

The second approach is based on the computer vision techniques. This does not require any special hardware for the interaction, hence such solutions are potentially available to all computer users. Such an interface is contact-free, so the naturalness and comfort can be preserved. The weakest point of such systems is their high dependence on the environment conditions (mainly lighting) [5], [6].

An interesting vision-based interaction concept was presented by Wilson and Cutrell. Here, the hand movements are captured using optical flow to control the mouse cursor position, but for other operations (e.g. clicking) the conventional hardware must be used. In addition, the authors outline how to extract basic hand features like size or tilt, but the hand shape or trajectory is not taken into account. The vision-based systems involve several image processing stages, i.e. skin region segmentation and hand detection, hand feature extraction and pose estimation, and finally, hand landmarks tracking and trajectory analysis [7].

In general, among the skin color models, statistical and parametric solutions can be distinguished. The parametric skin models are based on fixed decision rules defined in color spaces after analysis of skin-tone distribution. These rules are applied to determine if a pixel color value belongs to the skin [8].

Statistical skin color modeling is based on analysis of skin pixel values, distribution for a training set of images, in which skin and non-skin areas are already identified and annotated. This creates a global skin color model which makes it possible to determine the probability that a given pixel value belongs to the skin class. The main difficulty of skin region segmentation lies in high dependency of the skin models from the lighting conditions [9], [10].

There are many approaches to extracting hand features, based on which the recognition can be performed. Tan and Wu described hand state by image moments [11]. Boreki and Zimmer employed contour and its curvature analysis [12], while Manresa et al. used contour and convex hull [13]. Both methods allow for extracting information on the actual hand shape (e.g. convexity defects which can be used for finger detection). MacLean employed skeletonization concept for hand recognition, especially for finger detection [14].

Bhuyan et al. presented a different approach, where a vector of certain features (values), is extracted from the gesture trajectory. This vector forms a mathematical description that represents a certain class of gestures. A set formed from such vectors defines a knowledge-base that issued for gesture matching [15].

III. METHODOLOGY

A. Proposed System

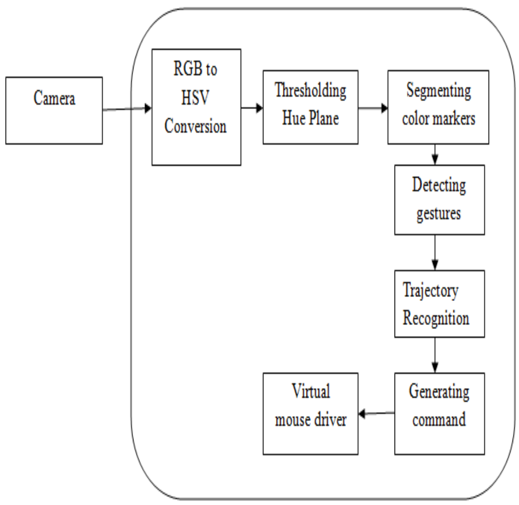

Fig. 1: Block diagram of implemented system

Now-a-days, mouse is the most popular input device, used for human-computer interaction to interact with the digital world through user's hand.

This kind of communication restricts the naturalness of the interaction because the control of the presentation keeps the user in the proximity of the computer. It would be more comfortable and effective if the user could point directly to the display device without any hardware equipment.

The block diagram of the proposed real time vision pointer interface system is shown in Fig. 1. The interaction with the physical world is done by camera. Camera is used as a sensor. Camera is attached to laptop by using USB. Here, USB webcam with 640 by 480 resolution used as a source of input. Camera takes the video and starts recording the live video and in continuation of recording it sends the live video to MATLAB which is installed in laptop. MATLAB contains software routines of camera initialization, rgb2hsv conversion, morphological operations, and centroid locating, virtual mouse driver and projector interface. In MATLAB, code is prepared which convert the incoming live video from camera into frames of images or slicing of video is done in the form of images. These images that are obtained from the slicing of video are then flipped and processed for color recognition process. The outputs of the color recognition process are the images that contains only those colors of which color caps are present at the fingertips of the user. Neither the fingers of user are shown in the output images nor any background colors, there in the output images from the color recognition process.

For this purpose, RGB values of the color caps are set prior in the code so that no other color will be detected in the image after color recognition except the caps colors. The output images are displayed in continuation and at the same speed as the speed at which slicing of video is done, so that it looks like a continuous movie in which the input is physical world and the output is only those colors which are present at the fingertips of the user. The colour is then associated with the mouse cursor in code so that whenever the colour moves in the output image from one position to another, the mouse cursor gets attached at the same position where the colour is now displayed. Depending upon gesture, command will be picked up from look up table and will be fed to virtual mouse driver. Virtual mouse driver will execute the command and will move the cursor physically.

The main application of this system is in educational institution or in any corporate presentation to make the work easy, simple and independent of lots of hardware requirements.

B. Hardware Details

The hardware components required are only Camera**,** Laptop/PC & Color marker.

I. Camera:

A webcam captures and recognises an object in view and tracks the user's fingers with colour marker using an algorithm. It sends the data to the laptop. The camera, in a sense, acts as a digital eye, seeing what the user sees. It also tracks the movements of the thumbs and index fingers of both of the user's hands. The camera recognizes objects around you instantly, with the projector overlaying the information on surface.

II. Laptop:

The laptop transmits and receives the data. A Web-enabled laptop processes the video data. Other software searches the Web and interprets the hand gestures.

III. Color markers:

It is at the tip of the user's fingers. Marking the user's fingers with red, yellow, green, and blue tape helps the webcam recognize gestures. The movements and arrangements of these makers are interpreted into gestures that act as interaction instructions for the projected application interfaces.

C. Software Details

For this project tools such as OpenCV, Matlab can be used. But I have used MATLAB software and windows 2007 operating system.

I) Comparison of Tools

a) OpenCV - Open source computer vision is a library of a programming functions mainly aimed at real time computer vision. It is free for use under the open source BSD license. The library is cross platform. It focuses mainly on real time image processing. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real time applications. The library is written in optimized C/C++ can take advantage of multi-core processing. Enabled with openCL, it can take advantage of the hardware acceleration of the underlying heterogeneous computer platform. OpenCV is damn fast when it comes to speed of the execution. In OpenCV atleast 30 frames per second can be analyzed resulting in real time detection[1].

b) MatLab- Matrix laboratory is a multi-paradigm numerical computing environment and fourth-generation programming language, developed by MathWorks. MatLab allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python. Although MatLab is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink, adds graphical multi-domain simulation and Model-Based Design for dynamic and embedded systems [1].

Ease of use-

Matlab is a relatively easy language to get to grisp with. Matlab is high level scripting language, meaning that we don't have to worry about libraries, declaring variables, memory management or other low level programming issue. As such ,it can be very easy to through together some code to prototype your image processing idea [1].

Memory management-

OpenCV is based on C.As such, every time we allocate a chunk of memory we will have to release it again. If we have loop in our code where we allocate a chunk of memory in that loop and forget to release it afterwards we will get 'leak'. This is where the program will use a growing amount of memory untill it crashes from no remaining memory. Due to the high level nature of matlab ,it is a smart enough to automatically allocate and release memory in the background.

Development environment-

Matlab comes with its own development environment. Matlab has workspace window available, so we can see the variables while program is getting developed. Hence, program developing and debugging becomes easier. For OpenCV, there is no particular IDE to use [1].

IV. RESULTS

As this project mainly focuses on having control of mouse through color marker, here from Fig. 2 and Fig. 3 it can be seen that a cursor is moving according to movement of blue color marker in front of camera.

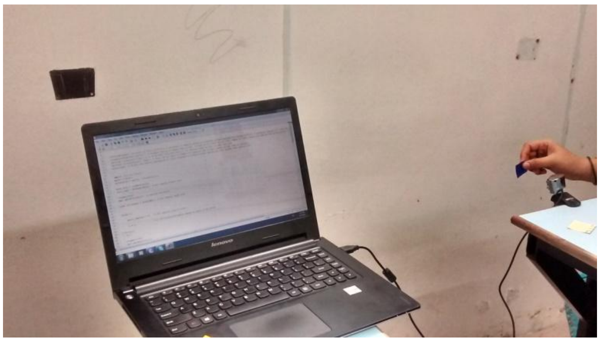

Fig. 2: Movement of blue color marker

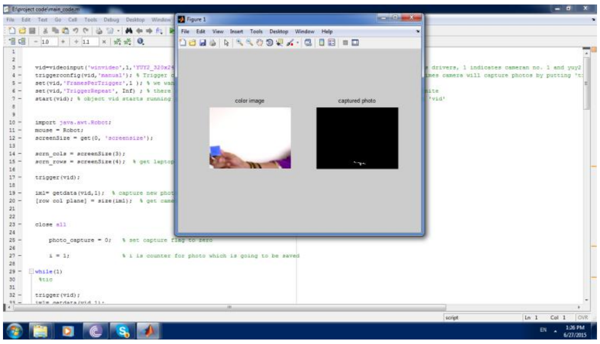

Fig. 3: Movement of blue color marker

It can be seen from Fig. 4, that a text is selected as camera is capturing yellow color, in Matlab window it is showing searching word status. In Fig. 4, the selected text is being searched in Google.

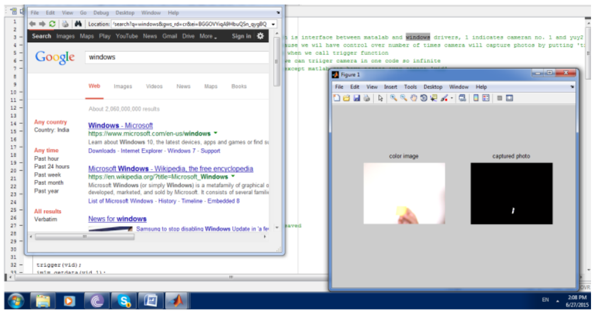

Fig. 4: Movement of yellow color marker

Fig. 5, is showing right click event of mouse as green color marker is detected.

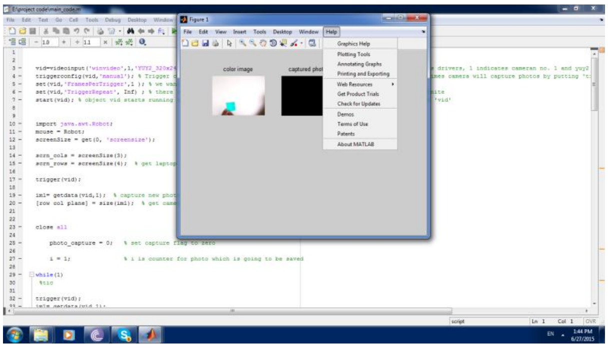

Fig. 5: Movement of green color marker

Fig. 6, shows photo is captured and it is stored on hard disc as arrangement of green and yellow color marker is shown.

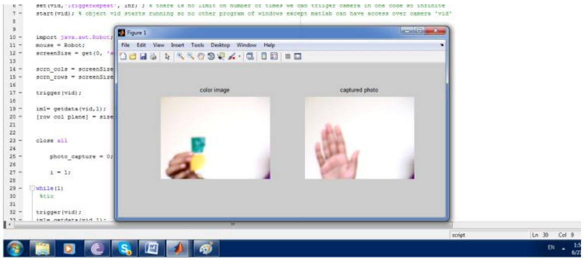

Fig. 6: Arrangement of green and yellow color marker

V. ADVANTAGES AND DISADVANTAGES OF THE SYSTEM

A. Advantages

- It uses hand gestures to interact with digital information.
- Supports multi-touch and multi-user interaction.
- Data access directly from machine in real time.
- It is cost effective and one can mind map the idea anywhere.
- It is portable and easy to carry .
- The system could be used by anyone without even a basic knowledge of a mouse.

B. Disadvantages

- This technology will affect the hardware market and will result in less revenue being generated and lower the cash inflow.
- The system gives better results at mornings and bright places as compare to night time and in dark areas

VI. FUTURE ENHANCEMENTS

Early research on vision-based hand tracking and gesture recognition usually needs the help of markers or colored gloves. In current state-of-the-art vision-based hand tracking and gesture recognition techniques, research is more focused on tracking the bare hand and identify hand gestures without the help of any markers and gloves.

- To get rid of color markers.
- Applying this technology in various interest like gaming, education systems etc.
- To have 3D gesture tracking.
- To make this work as helpful for disabled person.

VII. CONCLUSION

The key of this project is to make Human machine interface simple and easily interactive. Virtual interaction between laptop and user without using hardware like mouse, makes this entire system user friendlier and simpler. Adding sound response to the system makes this system helpful to the visually impaired persons.

Real time vision based pointer interface system helps to replace the computer's mouse and extend the interaction capabilities with the help of color markers. The system can operate using a simple camera and it allows for intuitive and non-restraining interaction. The real time vision based pointer interface system can improve the performance in terms of speed, accuracy and robustness. Clearly, it has the potential of becoming the ultimate "transparent" user interface for accessing information about everything around us. Having access to internet easily anywhere, effectively adds weightage to this entire system. Now on words the entire information will be in every ones hand. This is an user friendly interface which integrates digital information in to the physical world. It does not change human habits but causes computer and other machines to adapt to human needs.

VIII. ACKNOWLEDGMENT

I am using this opportunity to express my gratitude to everyone who supported me for writing this research paper. I am thankful for their guidance and invaluably advice during this work. I am sincerely grateful to them for sharing
their truthful and illuminating views on a number of issues related to this paper.

I express my warm thanks to Dr. Kanse Y.K for his support and guidance at Department of Electronics Engineering; Karmaveer Bhaurao Patil

College of Engineering, Satara, 415001; Maharashtra, India.

REFRENCES

[1]. Anupam Agrawal; Siddharth S. Rautaray; "Vision based hand gesture recognition for Human computer interaction: a survey", Springer Science, Business Media Dordrecht, 2012.

[2]. Darya Frolova; Helman Stern and Sigal Berman; "Most Probable Longest Common Subsequence for Recognition of Gesture Character Input" IEEE TRANSACTIONS ON CYBERNETICS, VOL. 43, NO. 3, JUNE 2013.

[3]. Maciej Czupryna, Michal Kawulok, "Real-Time Vision Pointer Interface" 54th International Symposium ELMAR- 2012, 12-14 September 2012.

[4]. J. Yang, K. Wang, X. Zhang, X. Chen and Y. Li, V. Lantz, "A framework for hand gesture recognition based on accelerometer and EMG sensors," IEEE Trans. on Systems, Man and Cybernetics, vol. 41, no. 6, pp. 1064-076, 2011.

[5]. Beifang Yi, Frederick C. Harris Jr., Ling Wang, and Yusong Yan, "REAL-TIME NATURAL HAND GESTURES", published by the IEEE CS and the AIP 1521- 9615, 2005, IEEE.

[6]. L. Dung; M. Mizukawa; "Fast Hand Feature Extraction Based on Connected Component Labeling, Distance Transform and Hough Transform", J. of Robotics and Mechatronics,vol.21, no. 6, pp. 726-738, Fuji Publishing, 2009.

[7]. A. Al. Hamadi; M. Elmezain; "Gesture Recognition for Alphabets from Hand Motion Trajectory Using Hidden Markov Models", IEEE Int. Symp. on Signal Process. and Information Tech, pp. 1192-1197, 2007.

[8]. A.D. Wilson; E. Cutrell; "FlowMouse: A Computer Vision-Based Pointing and Gesture Input Device" LNCS, vol. 3585, pp. 565-578, 2005.

[9]. N.G. Bourbakis; P. Kakumanu; S. Makrogiannis; "A survey of skincolor Modeling and detection methods", Pattern Recogn., vol. 40, no. 3, pp. 1106-1122, 2007.

[10]. M. H. Yang; N. Ahuja; "Gaussian Mixture Model for Human Skin Color and Its Applications in Image and Video Databases", SPIE Conf. on Storage and Retrieval for Image and Video Databases, pp. 458-466, 1999.

[11]. J. M. Rehg; M. J. Jones; "Statistical color models with application to skin detection", Int. J. Of Comp. Vision, vol. 46, no. 1, pp. 81-96, 2002.

[12]. C. Wu; S. Zhao; S. Chen; W. Tan; "Hand extraction using geometric moments based on active skin color model", IEEE Int. Conf. on Intell. Computing and Intell. Systems, vol. 4, pp. 468-471, 2009.

[13]. A. Zimmer; G. Boreki; "Hand geometry: a new approach for feature extraction", Fourth IEEE Workshop on Automatic Identification Advanced Technologies, pp.149-154, 2005.

[14]. C. Manresa; F.J. Perales; J. Varona; R. Mas; "Hand Tracking and Gesture Recognition for Human-Computer Interaction", Electronic Letters on Computer Vision and Image Analysis, vol. 5, no. 3, pp. 96-104, 2005.

[15]. C. Pantofaru; D. Topalovic; J. Tsotsos; J. MacLean; K. Derpanis; L. Wood; R. Herpers, "Fast hand gesture recognition for real-time teleconferencing applications", IEEE ICCV Workshop on Recogn., Analysis, and Tracking of Faces and Gestures in Real-Time Systems, pp. 133-140, 2001.