HUMAN COMPUTER INTERACTION USING KINECT

Sri Sai Ranga Malisetti

B.Tech, Electronics and Communication Engineering, University College of Engineering, Osmania University

Abstract - Computers have truly revolutionized the world but the way we interact with them hasn't changed much in a while. We traditionally use keyboards and mice but these interaction devices are primitive and have their own set of disadvantages, touchscreen technology has added another great feature of gesture recognition but we are still limited by the two degrees of freedom. Adding another degree of freedom give new possibilities in interaction with the computers. Developing an interface to interact with the computer using the three-dimensional nature of our hands can truly enable uses such as virtual reality, scientific visualization, robotics, gaming, etc. Microsoft Kinect is a great hardware to get started with in terms of its optical and infrared sensors which can be used in detecting our hands and thus can be programmed to get the most out of the onboard sensors. Once Kinect is successfully interfaced with the computer using Microsoft Visual Studio, the raw data output of Kinect can be programmed to detect, track and extract features from the hands placed above the sensor using object detection algorithms. Finally, hand gestures and controls can be programmed and triggered in the Windows operating system.

Keywords: Human computer Interaction, Microsoft Kinect, virtual reality.

I. INTRODUCTION

Humans interact with computers in many ways; the interface between humans and computers is crucial to facilitating this interaction. Desktop applications, internet browsers, handheld computers, and computer kiosks make use of the prevalent graphical user interfaces (GUI) of today. Voice user interfaces (VUI) are used for speech recognition and synthesizing systems, and the emerging multi-modal and Graphical user interfaces (GUI) allow humans to engage with embodied character agents in a way that cannot be achieved with other interface paradigms. The growth in human–computer interaction field has been in quality of interaction, and in different branching in its history. Instead of designing regular interfaces, the different research branches have had a different focus on the concepts of multimodality rather than unimodality, intelligent adaptive interfaces rather than command/action-based ones, and finally active rather than passive interfaces. Humans have come a long way with computer but the way we interact with computer is still traditional, using the devices such as keyboards and mice. The world is moving forward with technologies such as virtual reality and augmented reality and there is a strong need for devices to be capable of supporting the virtual environment. This aspect of virtual interaction motivated us to work with visual technology to make control much simplified using the Microsoft Kinect.

With the growing demand for greater interaction with the technological devices we use, the diminishing borders of user interaction call for greater innovation in the input devices. The innovating technology calls for a faster interaction of users with the technology. The traditional devices of computer input enable for two degrees of freedom and the user is restricted to using two dimensions but the real world is comprised of three dimensions in total. With the inclusion of the third dimension the user can interact with much more ease and interact with the computer. This has a large range of applications from 3D development to prototype designing.

Human-Computer Interaction

Human-Computer Interaction (HCI) research attempts to improve computer usability and meet the needs of users. The goal is to minimize barriers between how humans are able to share their needs and how computers are able to interpret and act on these needs. An interface is collection of methods and techniques to interact with something in the real world. it allows the user to communicate with computer or allows the computer to communicate with end user. In order to communicate with each other we required input and output devices. The use interface allows the user to enter data via input devices such as keyboard, mouse and allows the computer to interact with user in the form of displaying output on screen with help of monitor.

The User Interface broadly classified in to two categories as command line interface and graphical user interface (GUI). In command line interface the user has to enter commands as input at command prompt (for example UNIX shells). The second interface GUI consists of Window, Icon, Menus and Pointers. The GUI provides interaction via graphical elements. GUI attracted more to the end user's than command line interface. The icon is a kind of image that represents command, files and folders. Icons are generally having Standardisation. For example, Microsoft word contains different kinds of icons such as save, print, Undo, open a file icon etc. the same icons are available in MS-Power Point and MS-Excel applications. Command line interfaces are faster than graphical user interface. Other user interfaces are Menu driven and Natural language Interface. Menu Driven interface are Pull down and Pop-up Menu's.

The Human-Computer Interaction for the most part provides communication between human and computer. Still the computer is tool, which is controlled by human [1]. Communication involves exchange of different kinds of symbols and actions between human and computer. To exchange information between human and computer used various input devices like keyboard, mouse, showing symbols or graphs on the display [2]. The primary way to interact with computer is based on Graphical user interface. Human computer interaction is one of the most power full branches of computer science. It is the combination of computer science and cognitive science. The importance of HCI in computer science to develop the interactive products for computer-based system, which can be used to human with in effective manner. Traditional interaction manners are based up on command interface or GUI [3]. Recent technology the cognitive performances of human has been analysed using methods of physiological Parameters for objective and quantitative performance Measurement. The physiological parameters those are widely used is ECG (Electrocardiogram), EOG (Electrooculogram), Heart Rate, Eye blinking, and Heart Rate Variability (HRV) [4].

II. RELATED WORK

A. GESTURE RECOGNITION

Gesture recognition is a subject in language technology that has the objective of understanding human movement via mathematical procedures. Hand gesture recognition is currently the field of focus. This technology is future based. This new technology magnitudes an advanced association between human and computer where no mechanical devices are used. This new interactive device might terminate the old devices like keyboards and is also heavy on new devices like touch screens.

B. SPEECH RECOGNITION

The technology of transcribing spoken phrases into written text is Speech Recognition. Such technologies can be used in advanced control of many devices such as switching on and off the electrical appliances. Only certain commands are required to be recognized for a complete transcription. However, this cannot be beneficial for big vocabularies. HCI device help the user in hands free movement and keep the instruction-based technology up to date with the users.

C. VOICE-GUIDED USER INTERFACE

A voice-user interface (VUI) makes spoken human interaction with computers possible, using speech recognition to understand spoken commands and questions, and typically text to speech to play a reply. A voice command device (VCD) is a device controlled with a voice user interface. Voice user interfaces have been added to automobiles, home automation systems, computer operating systems, home appliances like washing machines and microwave ovens, and television remote controls. They are the primary way of interacting with virtual assistants on smartphones and smart speakers. Older automated attendants (which route phone calls to the correct extension) and interactive voice response systems (which conduct more complicated transactions over the phone) can respond to the pressing of keypad buttons via DTMF tones, but those with a full voice user interface allow callers to speak requests and responses without having to press any buttons.

D. MICROSOFT KINECT

Kinect (codenamed Project Natal during development) is a line of motion sensing input devices produced by Microsoft. Initially, the Kinect was developed as a gaming accessory for Xbox 360 and Xbox One video game consoles and Microsoft Windows PCs. Based around a webcam-style add-on peripheral, it enabled users to control and interact with their console/computer without the need for a game controller, through a natural user interface using gestures and spoken commands. While the gaming line did not gain much traction and eventually discontinued, third-party developers and researches found several after-market uses for Kinect's advanced low-cost sensor features, leading Microsoft to drive the product line towards more application-neutral uses, including integrating the device with Microsoft's cloud computing platform Azure.

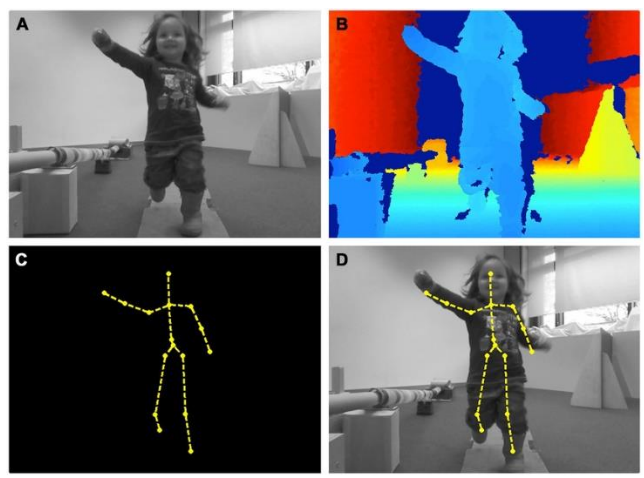

Figure 1 Various Output Streams from Kinect

The features of Kinect include the following.

- **Raw sensor streams**: Access to low-level streams from the depth sensor, color camera sensor, and four-element microphone array.

- **Skeletal tracking**: The capability to track the skeleton image of one or two people moving within Kinect's field of view for gesture-driven applications.
- **Advanced audio capabilities**: Audio processing capabilities include sophisticated acoustic noise suppression and echo cancellation, beam formation to identify the current sound source, and integration with Windows speech recognition API.

E. HUMAN COMPUTER INTERACTION USING KINECT

Many applications produced by companies and researchers used the Kinect sensor because of its motion-based software technology as an input method for face, motion and gesture tracking or recognition. This technology provides users and researchers with a facility to do 2D and 3D motion and gesture detection and skeleton motion tracking [5].

To recognize human motion and gestures, researchers used depth information and skeleton data to implement their work [6]. In our previous work with Kinect sensor V2 device [7], we presented an approach to detect the facial emotion expressions and for each expression which part of the brain is responsible for it. This is especially useful for a patient, who suffered a stroke if the patient cannot communicate verbally. A few papers similar to this work are listed below.

M. Ether, H. Wan and J. Lee [5], presented a mechanism to use the human body to control a virtual or real robotic arm by using the Kinect sensor and its SDK. A C# program was built by them to get the skeleton and the joints in the human motion from the Kinect sensor, measuring the distance between two joints of the skeleton and then generating eight simple commands, UP, DOWN, LEFT, RIGHT, FORWARD, BACKWARD, OPEN, and CLOSE. Through socket communication, commands can be passed to either a virtual robotic arm or a real robotic arm which implemented the commands.

A. Shingade and A. Ghotkar [7], presented a survey of motion, skeleton tracking techniques and different depth cameras with many NUI. Their system for skeleton recognition includes separating the foreground from the background and segmenting the human body into many regions. They used 20 joint positions for the skeleton depending on Shotton and et.al. [8]. To build a 3D character they used an open source S/W (MakeHuman) and applied a rigging process for the attached skeleton to a human character by applying the algorithm which was proposed by I. Baran and J. Popovic [9]. A database set was built for each gesture with its related joints in the system.

D. Alexiadis, P. Daras, P. Kelly, N. E. O'Connor, T. Boubekeur and Maher Ben Moussa [10], presented a system by using the Kinect sensor to align the performance dance for a student to a teacher, compute a general and an instant score for the student performance, and provide comments on the performance compared to the teacher.

They developed C++ OpenNI-based skeleton tracking software and a MATLAB engine to capture and record the body positions of 17 joints of the student performance (Head, Neck, Torso, Left and Right Collar, L/R Shoulder, L/R Elbow, L/R Wrist, L/R Hip, L/R Knee and L/R Foot) for each frame in a 3D vector signal. They captured dynamic movement for the dancer from the output of a 3D skeleton tracked character joints which were calculated from the convolution of the discrete time position signals with a first order Derivative of Gaussian. To provide the score, a comparison between the joint position of the student and the professional was made [11].

J. Lee, M. Hong, and S. Ryu [12] proposed a system that used Kinect V2 sensor, instead of other attached professional medical devices, to monitor the sleep movement, posture, and to extract the sleep information. It proved that the Kinect sensor can provide a significant sleep-related information from the human body. Their system was implemented by using C++ with the SDK for the sensor and used the OpenCV library for detection and tracking the movement of the sleeping human body. The Kinect sensor required a distance between 0.5 meters to 4.5 meters, so they located the Kinect at 2 meters on the body.

A critical 19 joints from 25 joints were selected which were relating to the sleep movements. The values of the movements were calculated every 0.5 sec. through Euclidean Distance between the former position of joints and the current position of joints from image sequences. The system continues to accumulate the values and make the comparison with 5 references sleep postures to determine user's current sleep posture until the wake up of the sleeper.

III. METHOD

A. INPUT IMAGE PRE-PROCESSING

Digital image processing allows the use of much more complex algorithms, and hence, can offer both more sophisticated performance at simple tasks, and the implementation of methods which would be impossible by analog means.

1) CLASSIFICATION

In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. Examples are assigning a given email to the "spam" or "non-spam" class, and assigning a diagnosis to a given patient based on observed characteristics of the patient (blood pressure, presence or absence of certain symptoms, etc.). Classification is an example of pattern recognition. In the terminology of machine learning, classification is considered an instance of supervised learning, i.e., learning where a training set of correctly identified observations is available. The corresponding unsupervised procedure is known as clustering, and involves grouping data into categories based on some measure of inherent similarity or distance.

Often, the individual observations are analysed into a set of quantifiable properties, known variously as explanatory variables or features. These properties may variously be categorical (e.g. "A", "B", "AB" or "O", for blood type), ordinal (e.g. "large", "medium" or "small"), integer-valued (e.g. the number of occurrences of a particular word in an email) or real-valued (e.g. a measurement of blood pressure). Other classifiers work by comparing observations to previous observations by means of a similarity or distance function. An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm, that maps input data to a category. In unsupervised learning, classifiers form the backbone of cluster analysis and in supervised or semi-supervised learning, classifiers are how the system characterizes and evaluates unlabelled data. In all cases though, classifiers have a specific set of dynamic rules, which includes an interpretation procedure to handle vague or unknown values, all tailored to the type of inputs being examined.

2) IMAGE SEGMENTATION

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super-pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s). When applied to a stack of images, typical in medical imaging, the resulting contours after image segmentation can be used to create 3D reconstructions with the help of interpolation algorithms like Marching cubes. The edges extracted from a two-dimensional image of a three-dimensional scene can be classified as either viewpoint dependent or viewpoint independent. A viewpoint independent edge typically reflects inherent properties of the three-dimensional objects, such as surface markings and surface shape. A viewpoint dependent edge may change as the viewpoint changes, and typically reflects the geometry of the scene, such as objects occluding one another.

3) MULTI – SCALE SIGNAL ANALYSIS

Signal processing is a subfield of mathematics, information and electrical engineering that concerns the analysis, synthesis, and modification of signals, which are broadly defined as functions conveying "information about the behaviour or attributes of some phenomenon",[1] such as sound, images, and biological measurements.[2] For example, signal processing techniques are used to improve signal transmission fidelity, storage efficiency, and subjective quality, and to emphasize or detect components of interest in a measured signal.

4) PROJECTION

In mathematics and statistics, random projection is a technique used to reduce the dimensionality of a set of points which lie in Euclidean space. Random projection methods are powerful methods known for their simplicity and less erroneous output compared with other methods. According to experimental results, random projection preserves distances well, but empirical results are sparse. They have been applied to many natural language tasks under the name of random indexing.

5) THRESHOLDING

Thresholding is a non-linear operation that converts a gray-scale image into a binary image where the two levels are assigned to pixels that are below or above the specified threshold value. The simplest thresholding methods replace each pixel in an image with a black pixel if the image intensity is less than some fixed constant T or a white pixel if the image intensity is greater than that constant. In the example image on the right, this results in the dark tree becoming completely black, and the white snow becoming completely white. Colour images can also be thresholded. One approach is to designate a separate threshold for each of the RGB components of the image and then combine them with an AND operation. This reflects the way the camera works and how the data is stored in the computer, but it does not correspond to the way that people recognize colour. Therefore, the HSL and HSV colour models are more often used; note that since hue is a circular quantity it requires circular thresholding. It is also possible to use the CMYK colour model. Automatic thresholding is a great way to extract useful information encoded into pixels while minimizing background noise. This is accomplished by utilizing a feedback loop to optimize the threshold value before converting the original grayscale image to binary. The idea is to separate the image into two parts; the background and foreground.

6) EDGE DETECTION

Edge detection includes a variety of mathematical methods that aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed edges. The same problem of finding discontinuities in one-dimensional signals is known as step detection and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction. A typical edge might for instance be the border between a block of red color and a block of yellow. In contrast a line (as can be extracted by a ridge detector) can be a small number of pixels of a different color on an otherwise unchanging background. For a line, there may therefore usually be one edge on each side of the line.

B. DATA ANALYSIS AND MACHINE LEARNING

1) PATTERN RECOGNITION

Pattern recognition is the automated recognition of patterns and regularities in data. Pattern recognition is closely related to artificial intelligence and machine learning, together with applications such as data mining and knowledge discovery in databases (KDD), and is often used interchangeably with these terms. However, these are distinguished: machine learning is one approach to pattern recognition, while other approaches include hand-crafted (not learned) rules or heuristics; and pattern recognition is one approach to artificial intelligence, while other approaches include symbolic artificial intelligence. A modern definition of pattern recognition is:

"The field of pattern recognition is concerned with the automatic discovery of regularities in data through the use of computer algorithms and with the use of these regularities to take actions such as classifying the data into different categories"

This article focuses on machine learning approaches to pattern recognition. Pattern recognition systems are in many cases trained from labelled "training" data (supervised learning), but when no labelled data are available other algorithms can be used to discover previously unknown patterns (unsupervised learning). Machine learning is the common term for supervised learning methods and originates from artificial intelligence, whereas KDD and data mining have a larger focus on unsupervised methods and stronger connection to business use. Pattern recognition has its origins in engineering, and the term is popular in the context of computer vision: a leading computer vision conference is named Conference on Computer Vision and Pattern Recognition. In pattern recognition, there may be a higher interest to formalize, explain and visualize the pattern, while machine learning traditionally focuses on maximizing the recognition rates. Yet, all of these domains have evolved substantially from their roots in artificial intelligence, engineering and statistics, and they've become increasingly similar by integrating developments and ideas from each other.

In machine learning, pattern recognition is the assignment of a label to a given input value. In statistics, discriminant analysis was introduced for this same purpose in 1936. An example of pattern recognition is classification, which attempts to assign each input value to one of a given set of classes (for example, determine whether a given email is "spam" or "non-spam"). However, pattern recognition is a more general problem that encompasses other types of output as well. Other examples are regression, which assigns a real-valued output to each input; sequence labelling, which assigns a class to each member of a sequence of values (for example, part of speech tagging, which assigns a part of speech to each word in an input sentence); and parsing, which assigns a parse tree to an input sentence, describing the syntactic structure of the sentence.

Pattern recognition algorithms generally aim to provide a reasonable answer for all possible inputs and to perform "most likely" matching of the inputs, taking into account their statistical variation. This is opposed to pattern matching algorithms, which look for exact matches in the input with pre-existing patterns. A common example of a pattern-matching algorithm is regular expression matching, which looks for patterns of a given sort in textual data and is included in the search capabilities of many text editors and word processors. In contrast to pattern recognition, pattern matching is not generally a type of machine learning, although pattern-matching algorithms (especially with fairly general, carefully tailored patterns) can sometimes succeed in providing similar-quality output of the sort provided by pattern-recognition algorithms.

2) FEATURE EXTRACTION

In machine learning, pattern recognition and in image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Feature extraction is a dimensionality reduction process, where an initial set of raw variables is reduced to more manageable groups (features) for processing, while still accurately and completely describing the original data set. When the input data to an algorithm is too large to be processed and it is suspected to be redundant (e.g. the same measurement in both feet and meters, or the repetitiveness of images presented as pixels), then it can be transformed into a reduced set of features (also named a feature vector). Determining a subset of the initial features is called feature selection. The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

Feature extraction involves reducing the amount of resources required to describe a large set of data. When performing analysis of complex data one of the major problem's stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power, also it may cause a classification algorithm to overfit to training samples and generalize poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy. Many machine learning practitioners believe that properly optimized feature extraction is the key to effective model construction.

IV. IMPLEMENTATION AND RESULTS

The hand tracking model is implemented through a seven-step procedure. This results in finite edge mapping and the location of the centroid on the hand which can be used as point of reference. The output is channelled to an external source through a python script which enables the successful movement of the cursor in the computer. The seven-step process is explained in detail in the coming sections.

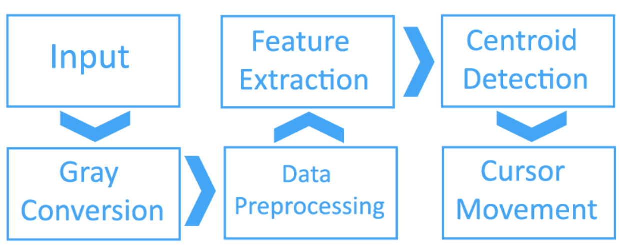

Figure 2 Flowchart of the Implementation

1) INPUT

Kinect video camera aids in facial recognition and other detection features by detecting three color components: red, green and blue. Microsoft calls this an "RGB camera" referring to the color components it detects. The Kinect camera is an RGB device with a resolution of 640×480 and a 24-bit color range (Red- Green- Blue channels). Working at a rate of 30 frame captures per second this camera, this camera is similar to the run of the mill webcam or the sensors in your digital camera and it is, in most regards, very commonplace. The camera is capable of capturing 1280×960 resolution however it seems that the fastest frames per second (fps) achievable at this higher definition is about 15fps. Conversely faster frame rates are possible by setting the resolution of the camera lower. The Kinect sensor is limited in the distance that it can see and has a working range of between 1.2 and 3.5 meters. The horizontal field of view is 57° wide, which means at its maximum range it will be able to scan a scene 3.8 meters wide.

Sensor Specs.: 640×480 pixels @ 30 Hz

2) GRAYSCALE CONVERSION

Average method is the simplest one. You just have to take the average of three colors. Since it's an RGB image, so it means that you have add r with g with b and then divide it by 3 to get your desired grayscale image. This problem arises due to the fact, that we take average of the three colors. Since the three different colors have three different wavelength and have their own contribution in the formation of image, so we have to take average according to their contribution, not done it averagely using average method. We are taking 33% of each, that means, each of the portion has same contribution in the image. But in reality, that's not the case. The solution to this has been given by luminosity method.You have seen the problem that occur in the average method. Weighted method has a solution to that problem. Since red color has more wavelength of all the three colors, and green is the color that has not only less wavelength then red color but also green is the color that gives more soothing effect to the eyes. It means that we have to decrease the contribution of red color, and increase the contribution of the green color, and put blue color contribution in between these two. So, the new equation that form is:

New grayscale image = ((0.3 * R) + (0.59 * G) + (0.11 * B)).

According to this equation, Red has contributed 30%, Green has contributed 59% which is greater in all three colors and Blue has contributed 11%.

3) THRESHOLDING

The input to a thresholding operation is typically a grayscale or color image. In the simplest implementation, the output is a binary image representing the segmentation. Black pixels correspond to background and white pixels correspond to foreground (or vice versa).

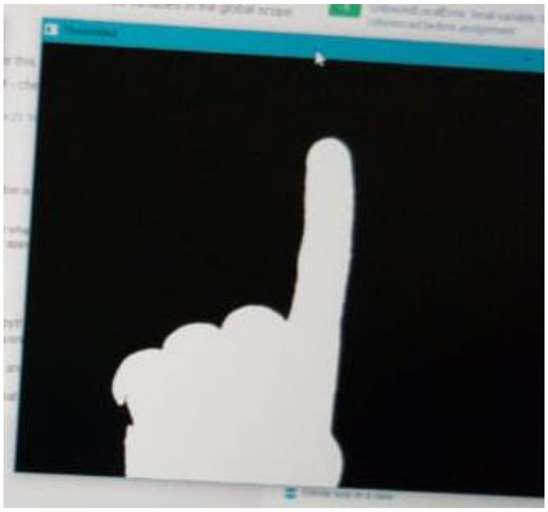

Figure 3 Image after Thresholding

In simple implementations, the segmentation is determined by a single parameter known as the intensity threshold. In a single pass, each pixel in the image is compared with this threshold. If the pixel's intensity is higher than the threshold, the pixel is set to, say, white in the output. If it is less than the threshold, it is set to black.

4) DRAWING LANDMARKS

Figure 4 Image after real-time landmark tracing

The threshold image is essential in the feature extraction of the object. Essentially, the boundary is drawn over the hand which can be used as a viewpoint for the boundary algorithm to draw a line over the edge of the hand. The boundary determines the area of the hand and can be used to comprehend the open or the closed nature of the fist.

5) CENTROID DETECTION

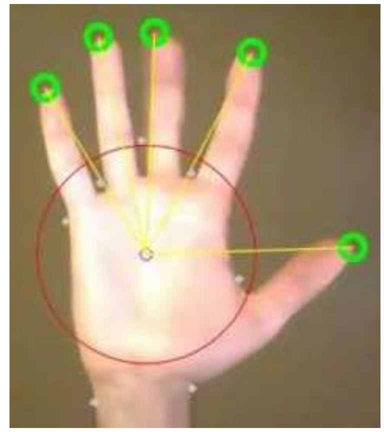

Figure 5 Centroid Tracking

The centroid of the hand is mapped in the similar way as shown in the above image which is calculated from the boundary from the feature extraction segment. This results in the pin-point feature map of the hand with its centroid, used further in the location coordination of the cursor in the computer. The distance from the camera plays an important role as the further the hand moves the area of the image on sensor. Therefore, for accurate results it is essential for the hand to be at a finite distance from the sensor.

6) CURSOR MOVEMENT

A mouse cursor, also known as a mouse arrow, or mouse pointer, is a graphical image that is used to activate or control certain elements in a graphical user interface. More plainly, it indicates where your mouse should perform its next action, such as opening a program, or dragging a file to another location. The mouse pointer follows the path of the user's hand as they move their mouse. The relative movement of the hand is tracked and the cursor movement is tracked.

7) GUI DISPLAY

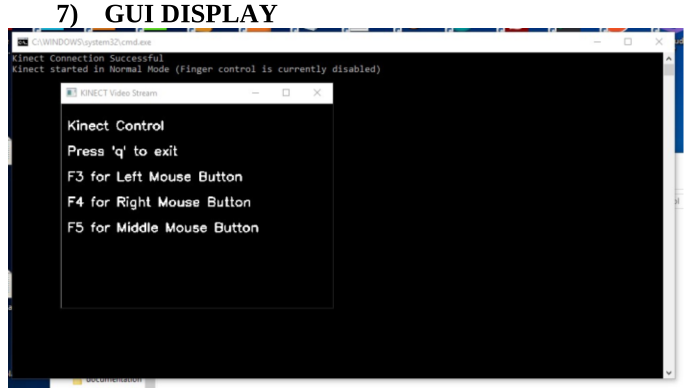

Figure 6 GUI window in Windows 10

The final output of the project is implemented as a Graphical User Interface developed using the python-OpenCV window generation interface. This helps in the user to directly interact with the computer and enable the user to move the mouse with the movement of the house. The finger movement option is also enabled but is given as a development feature. The mouse buttons are embedded in to the function keys using the AutoHotkey script which is started as a subprocess through the python subprocess module. The keys can be controlled using the left hand and the cursor can be controlled using the right hand. The inclusion of relative mouse speeds helps in the better movement of the cursor. The gestures are also implemented in the finger model with the finger count determining the mouse controls directly through a single hand.

V. CONCLUSION

The proposed approach to control the cursor in the aim of the project has been accomplished and the result is tested with various lighting conditions and the inclusion of the various hand gestures has also been successful. The application has been created with the solid principles in consideration and made available with and a GUI output. The application can be further certified and can be used as a product with runtime executable supplied to the user. Although the two-dimensional axis control has been integrated but with the inclusion of depth map the three-dimensional control can be integrated. The further inclusion of the 3D map feature in the new version of the Kinect helps in the generation of a 3D mesh around the hand model which can be used in the multiple finger point calculation. Using a better close-range IR sensor can be influential in the close-range distance determination and the inclusion of the Z axis controls. The spatial mapping of the coordinates can be improved with the IR camera as there is no much interference in the IR Spectrum.

VI. REFERENCES

[1] Bo Cui, Tongze Xue, "Design and realization of an intelligent access control system based on voice recognition ",Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on, Volume: 1Page(s): 229 – 232, 8-9 Aug. 2009

[2] Lu Ji xiang; Wang Ping; Yu Sui huai; Lu Chang de; , in "An improved spectral subtraction algorithm based on auditory masking in voice human-computer interaction ," Mechatronics and Automation (ICMA), 2010 International Conference on, 4-7 Aug. 2010,page 1938 – 1941.

[3] Zhen Zhu , Jing-Yan Wang, in "Multi-Agent Based Approach to Support HCI," Machine Learning and Cybernetics, 2006 International Conference on , 13-16 Aug. 2006.

[4] Rabbi, A.F, Ivanca, K,Putnam, A.V, Musa, A, Thaden, C.B,Fazel-Rezai, R," Human performance evaluation based on EEG signal analysis: A prospective review", Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE, 3-6 Sept. 2009.

[5] S.Hyuk Lee and S.Hyun Oh, "A Kinect Sensor based Windows Control Interface", International Journal of Control and Automation, vol.7, no.3, 2014, pp.113-124.

[6] Hesham A. ALABBASI, Florica Moldoveanu, Alin Moldoveanu and Zaid Shhedi," Facial Emotion Expression Recognition With brain Activates Using Kinect Sensor V2", International Research Journal of Engineering and Technology (IRJET), vol. 02, Issue 02, May-2015, pp 421-428.

[7] M. Eltaher, H. Wan and J. Lee," Motion Detection Using Kinect Device for Controlling Robotic Arm", 2013 ASEE Northeast Section Conference, Reviewed Paper, March 14-16, 2013.

[8] A. Shingade and A. Ghotkar," Animation of 3D Human Model Using Markerless Motion Capture Applied To Sports", International Journal of Computer Graphics & Animation (IJCGA), vol.4, no.1, January 2014.

[9] A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman and A. Blake, "Real-time human pose recognition in parts from single depth images", Machine Learning for Computer Vision book, vol. 411, pp 119-135, 2013.

[10] I. Baran and J. Popovi,"Automatic rigging and animation of 3d characters" ,ACM Transactions on Graphics (TOG), SIGGRAPH 2007, vol.26, issue 3, July 2007

[11] D. Alexiadis, P. Daras,P. Kelly, N. E. O'Connor, T. Boubekeur and Maher Ben Moussa, "Evaluating a Dancer's Performance using Kinectbased Skeleton Tracking", the 19th ACM international conference on Multimedia, November 2011, Pages 659-662.

[12] J. Lee, M. Hong, and S. Ryu,"Sleep Monitoring System Using Kinect Sensor", Hindawi Publishing Corporation, International Journal of Distributed Sensor Networks ,Article ID, April 2015.

[13] R . C Gonzalez and R. E Woods, "Digital Image Processing", Pearson, 3 edition, 2007.

About the Author:

Sri Sai Ranga Malisetti received the B.E. degree from the University College of Engineering, Osmania University, India. He is currently working as a Cloud Engineer at Edgetensor Technologies Inc. His current research interests include Cloud Security, Cryptography and Risk Analysis.