International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST

Hand Gesture Recognition Using OpenCV

Rohini M[1], Abishek Leo Kingston. J[2], Shriram.G.S[2], Siva Sankaran.S[2], Vasuki. G[2]
[1]Assistant Professor, Department of Computer Engineering, Coimbatore Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India
[2]UG Scholar, Department of Computer Engineering, Coimbatore Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India

Article Info
Volume 8, Issue 2
Page Number : 272-277

Publication Issue
March-April-2021

Article History
Accepted : 20 March 2021
Published : 28 March 2021

ABSTRACT

In human-computer interaction or sign language interpretation, recognizing hand gestures and detecting fingertips become ubiquitous in computer vision research. Hand gesture recognition provides an intelligent and natural way of human computer interaction. Its applications range from medical rehabilitation to consumer electronics control. In order to distinguish hand gestures, various kinds of sensing techniques are utilized to obtain signals for pattern recognition. This system can be divided into three parts according: Hand detection, hand gesture recognition, Edge detection for images, Integrating media. The system has two major advantages. First, it is highly modularized, and each of these steps is capsuled from others; second, the edge/contour detection of hand as well as gesture recognition is an add-on layer, which can be easily transplanted to other applications. The techniques used for image processing are hand gesture detection, edge detection, thresholding, contour detection. Using OpenCV, which provides a library collection of functions for different image processing techniques, these input images can be processed and corresponding key strokes will be generated.

Keywords : Fingertip detection, Gesture recognition, Handy algorithm.

I. INTRODUCTION

Hand gesture recognition provides an intelligent and natural way of human computer interaction (HCI). Its applications range from medical rehabilitation to consumer electronics control (e.g. mobile phone). In order to distinguish hand gestures, various kinds of sensing techniques are utilized to obtain signals for pattern recognition. Acceleration-base and electromyogram-based techniques are two research branches in the field of hand gesture pattern recognition. Acceleration-based (ACC-based) gesture control is usually studied as a supplementary interaction modality. It is well suited to distinguish noticeable, larger scale gestures with different hand trajectories of forearm movements. With ACC-based techniques some subtle finger or hand movement may be ignored whereas electromyogram-based (EMG-

based) gesture recognition techniques use multi-channel EMG signals which contain rich information about hand gestures of various size scales. Due to some problems inherent in the EMG measurements, including the separability and reproducibility of measurement, the size of discriminable hand gesture set is still limited to 4-8 classes. In order to realize a natural and robust gesture-based HCI system, the selection of input hand gestures that are well discriminable from each other is of crucial importance. Considering the complementary features of ACC- and EMG-measurements, we believe that their combination will increase the number of discriminable hand, wrist and forearm gestures and the accuracy of the recognition system. This paper describes in IP Gaming we are proposing a system in which without using sensors and Devices, we are detecting the hand and gesture with Simple Web camera and performing the Image Processing technique in which using those gesture, we can control the media on console. In Image Processing the motions are detected through a web camera. These images are then passed for the image processing. The techniques used for image processing are hand gesture detection, edge detection, thresholding, contour detection. Using OpenCV, which provides a library collection of functions for different image processing techniques, these input images can be processed and corresponding key strokes will be generated.

II. LITERATURE SURVEY

Motion capture and depth sensing are two emerging areas of research in recent years. With the launch of Kinect in 2010, Microsoft opened doors for researchers to develop, test and optimize the algorithms for these two areas. J Shotton proposed a method to quickly and accurately predict 3D positions of the body joints without using any temporal data. Key prospect of the method is they are considering a single depth image and are using an object recognition approach. From a single input depth image, they inferred a per pixel body part distribution. Leyvand T discussed about the Kinect technology. His work throws light on how the Identity of a person is tracked by the Kinect for XBox 360 sensor. Also a bit of information about how the changes are happening in the technology over the time is presented. With the launch of Kinect, there is a sea change in the identification and tracking techniques. The authors discussed the possible challenges over the next few years in the domain of gaming and Kinect sensor identification and tracking. Kinect International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 3, March - 2013 ISSN: 2278-0181 IJERT identification is done by two ways: Biometric sign-in and Session tracking. A method to track fingertips and the center of palms using Kinect was presented by Raheja. It applied thresholding on the depth of hand regions for segmentation. Then the palm was filtered and subtracted from the hand, so that only the fingers were left in the image. Under most situations when the hand was in front of the user, the fingers should be closest to the Kinect with the shallowest depth. Therefore, by determining the minimum depth, fingertips were found. The center of the palm was determined by finding the maximum of distance within the image of the hand. When fingers were extended, the accuracy of detecting fingertips was nearly 100% accuracy, and that of palm center was around 90%. However, this method did not attempt at gesture recognition. He proposed another approach using depth data provided by Kinect to detect fingertips. First, it found hand points by thresholding on depth elata, and then generated the convex hull containing the hand by Graham Scan. Fingertips were detected by calculating the angle between candidate points. After fingertips were found, the mouse clicking motion was recognized and tested on the popular game Angry Bird; that is, it recognized only one gesture.

1. EDGE DETECTION

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image and data extraction in areas such as image processing, computer vision, and machine vision. Since computer vision involves the identification and classification of objects in an image, edge detections is an essential tool.

2.HANDY ALGORITHM

Handy algorithm is escpecially used to detect the hands and gestures.One of its key features is the ability to display the spatialization process,aiming at transforming the network into a map,and ForceAtlas2 is its default layout algorithm.The latter is developed by the Gephi user's typical networks.

3. THRESHOULD

Frequently we have done many layers of processing steps and want either to make a final decision about the pixels in an image or to categorically reject those pixels below or above some value while keeping the others. The OpenCV function cvThreshold() accomplishes these tasks. The basic idea is that an array is given, along with a threshold, and then some-thing happens to every element of the array depending on whether it is below or above the threshold. The cvThreshold() function handles only 8-bit or floating-point grayscale source images.

1. Capture the images using web camera. 2. On applying Image Processing technique we will be able to detect the hands and Gesture and perform some console based operation.

i. Edge Detection: Edges are the sharp black shadow surrounding the objects.

ii. Threshold Control: for controlling sharpness of edges.

iii. Finding Contours, contours are nothing but shadow areas of hand.

iv. Set the proper beginning of the contours.

v. Detect Convexity Defect in the picture. Defects are the points which are having thick edges.

vi. Detect Convexity Defect ending points for the tip of our hand detection.

vii. Draw Circles on the defects we obtained.

viii. Save the Co-ordinates of the Defects obtained in each areas.

3. Trigger Image Capture.

4. Background Subtraction is done for clearing background.

III. ALGORITHM FOR HAND DETECTION

Finger Identification Algorithm

1. The first and easiest step is to identify the thumb and the index finger, since the distance between them is the largest among all neighboring fingers.

2. The little finger is identified as the farthest finger away from the thumb; meanwhile the middle finger is identified as the closest one to the index finger.

3. The remaining one is the ring finger. The process of detecting hands and identifying fingers are performed every time when the input data source changes. If the same object still exists in the current frame with some transformation compared to the previous frame, all properties of this object is mapped from the previous frame to the current frame; otherwise the disappeared object is collected by an unmapped item collector.

Fingertip Detection, The distance between Po and the line made by PI and P2 is apparently larger than that of p˜ and the line made by p˜ and p;. Thus a threshold distance could be used to differentiate fingertip points and non-fingertip points.

IV. SYSTEM ARCHITECTURE

The architecture diagram consists of:

- System
- OpenCV Library
- Background Subtraction
- Hand Gesture Detection

The motions are detected using web camera and passed to the system for processing.

The system consists of OpenCV and Java Application, using these Hand Detection and Background subtraction is done. This processed image in the checked in which area it has occurred and its corresponding key event is called for action to be performed. These steps are repeated till the end of the recognition.

Fig.1.Process of Hand Gesture Recognition

V. RESULT AND DISCUSSIONS

The purpose of this project is to recognize hand gesture with more accuracy. The design is very simple and the user doesn't need to wear any type of hand gloves. Although this recognition application can be run in an ordinary computer having a web camera, but ideally it requires system having camera with at least 1.4GHz processor and at least 1GB RAM. The algorithm can detect the hand gestures with 99.95% recognition rate if the user interacts in right way.

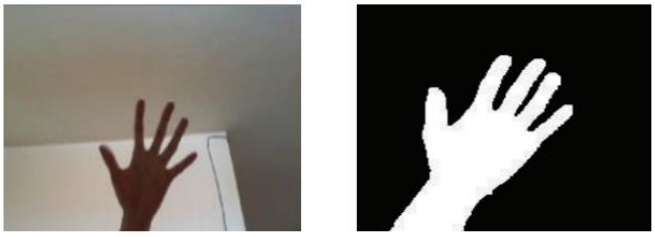

Fig.2. Hand Gesture Recognized

VI. CONCLUSION

In this paper, we have discussed how camera can be used for Detection hand gestures and can be applied to any . We are using camera as a detecting device as well as input device for Augmented Reality System. The proposed system helps reduce the burden on experts to look into few regular activities. Instead, they can use our system for such activities. Also, the work simplifies the documentation process. The supervisor can keep track of current status of activity from his desk. Also, stepwise verification is possible as the system keeps track of each step. Through the introduction of our system, we will bring new opportunities for mechanical engineering based companies to use Augmented Reality for simplification of their complex tasks. This will add new dimensions to the conventional way of maintenance activities.

VII. REFERENCES

[1]. A. Bandini, J. Zariffa, Analysis of the hands in egocentric vision: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[2]. C. Von Hardenberg, F. Be´rard, Bare-hand human-computer interaction, in: Proceedings of the 2001 workshop on Perceptive user interfaces, ACM, Orlando, Florida, USA, 2001, pp. 1–8.

[3]. P. Xu, A real-time hand gesture recognition and human-computer interaction system, arXiv preprint arXiv:1704.07296.

[4]. A. Haria, A. Subramanian, N. Asokkumar, S. Poddar, J. S. Nayak, Hand gesture recognition for human computer interaction, Procedia computer science 115 (2017) 367–374.

[5]. R. Rastgoo, K. Kiani, S. Escalera, Sign language recognition: A deep survey, Expert Systems with Applications (2020) 113794.

[6]. T. Sahana, S. Paul, S. Basu, A. F. Mollah, Hand sign recognition from depth images with multi-scale density features for deaf mute persons, Procedia Computer Science 167 (2020) 2043–2050.

[7]. M. Flasin´ski, S. Mys´lin´ski, On the use of graph parsing for recognition of isolated hand postures of polish sign language, Pattern Recognition 43 (6) (2010) 2249–2264.

[8]. B. Garcia, S. A. Viesca, Real-time american sign language recognition with convolutional neural networks, Convolutional Neural Networks for Visual Recognition 2.

[9]. J. P. Sahoo, S. Ari, D. K. Ghosh, Hand gesture recognition using dwt and f-ratio based feature descriptor, IET Image Processing 12 (10) (2018) 1780–1787.

[10]. M.-Y. Wu, P.-W. Ting, Y.-H. Tang, E.-T. Chou, L.-C. Fu, Hand pose estimation in object-interaction based on deep learningfor virtual reality applications, Journal of Visual Communicationand Image Representation (2020) 102802.

[11]. M. M. Alam, S. M. Rahman, Detection and tracking of fingertips for geometric transformation of objects in virtual environment, in: Proc. IEEE/ACS 16th Int. Conf. on Computer Systems and Applications (AICCSA), IEEE, Abu Dhabi, United Arab Emirates, 2019, pp. 1–8.

[12]. P. Song, H. Yu, S. Winkler, Vision-based 3d finger interactions for mixed reality games with physics simulation, in: Proceedings of the 7th ACM SIGGRAPH Int. Conf. on virtual-reality continuum and its applications in industry, ACM, Singapore, 2008, p. 7.

[13]. C. Weichel, M. Lau, D. Kim, N. Villar, H. W. Gellersen, Mixfab: a mixed-reality environment for personal fabrication, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, Toronto, Ontario, Canada, 2014, pp. 3855–3864.

[14]. M. M. Alam, S. M. Rahman, Affine transformationof virtual 3d object using 2d localization of fingertips, Virtual Reality & Intelligent Hardware 2 (6) (2020) 534–555.

[15]. M. Bhuyan, D. R. Neog, M. K. Kar, Fingertip detection for hand pose recognition, International Journal on Computer Science and Engineering 4 (3) (2012) 501.

[16]. J. Shotton, 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. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1297 –1304, June 2011.

[17]. T. Leyvand, C. Meekhof, Y.-C. Wei, J. Sun, and B. Guo. Kinect identity: Technology and experience. Computer, 44(4):94 –96, April 2011.

[18]. S. Henderson and S. Feiner. Exploring the benefits of augmented reality documentation for maintenance and repair. Visualization and Computer Graphics, IEEE Transactions on, 17(10):1355 –1368, Oct. 2011.

[19]. Saket Warade, Jagannath Aghav, Petitpierre Claude, Sandeep Udayagiri. Real-Time Detection and Tracking with Kinect. International Conference on `Computer and Information Technology (ICCIT'2012) June 16-17, 2012, Bangkok

[20]. R. Owens, "Lecture 6", Computer Vision IT412, 10/29/1997. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/ OWENS/LECT6/node2.html

[21]. S. Price, "Edges: The Canny Edge Detector", July 4, 1996. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/ MARBLE/low/edges/canny.html

[22]. Ehsan Nadernejad, Sara Sharifzadeh, Hamid Hassanpour. Edge Detection Techniques: Evaluations and Comparisons. Applied Mathematical Sciences, Vol. 2, 2008, no. 31, 1507 – 1520.

Cite this article as :

Rohini M, Abishek Leo Kingston. J, Shriram.G.S, Siva Sankaran.S, Vasuki. G, "Hand Gesture Recognition Using OpenCV", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 8 Issue 2, pp. 272-277, March-April 2021.
Journal URL : https://ijsrst.com/IJSRST218247