**MAJOR PROJECT REPORT**

(Project Semester June-December 2017)

**HAND GESTURES RECOGNITION SYSTEM**

Submitted by:

**Taresh Dewan (14103053)**

**Archit Guleria (14103054)**

**Amisha Jindal (14103061)**

**Divya Sharma (14103062)**

Under the Guidance of:

**Prof. Anoop Dobhal & Prof. Shefali Aggarwal**

**Assistant Professors**

**Department Of Computer Science and Engineering**

**PEC University of Technology, Chandigarh**

**DECLARATION**

We hereby declare that the project work entitled **“Hand Gestures Recognition System”** is an authentic record of our own work carried out at PEC University of Technology, Chandigarh, under the guidance of **Prof. Anoop Dobhal and Prof. Shefali Aggarwal**, for Major Project offered by PEC University of Technology, Chandigarh, during the academic year 2017-18.

**Taresh Dewan (14103053) Archit Guleria (14103054)**

**Amisha Jindal (14103061) Divya Sharma (14103062)**

**CERTIFICATE**

Certified that the project work entitled **“Hand Gestures Recognition System”** submitted by **Taresh Dewan, Archit Guleria, Amisha Jindal and Divya Sharma**, for the partial fulfilment of Major Project offered by PEC University of Technology, Chandigarh, during the academic year 2017-18 is a genuine work carried out by the students in my supervision and has not been tampered with.

Date: 30 November 2017 **Prof. Anoop Dobhal & Prof. Shefali Aggarwal**

**Assistant Professors, CSE Department**

**ACKNOWLEDGEMENT**

It gives us utmost pleasure as we acknowledge and extend our heartfelt gratitude to the ones who believed in us and helped us in making this project a success. Without their cooperation, we wouldn’t have been able to achieve the milestones that we had aimed for.

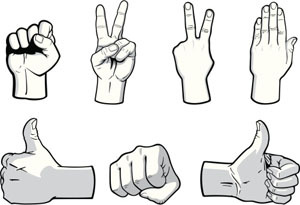
We would like to express our thanks to **Prof. Anoop Dobhal** and **Prof. Shefali Aggarwal** for their constant guidance, valuable suggestions and moral support. They have always been ready with ways to improve the proceedings and quality of output. Their consistent energy and enthusiasm helped us get everything on track.

We would also like to other faculty members and friends for their constant support during the course of the project.

**ABSTRACT**

Hand gesture recognition is very significant for human-computer interaction. Hence, we present a system that will use a real-time method for hand gesture recognition. Our first module includes getting the system to recognize gestures and testing it with a real-time application such as volume control and the second module is innovating the technology by fusing augmented reality with it and developing a video chat feature.

As we know, the vision-based technology of hand gesture recognition is an important part of human-computer interaction (HCI). In the last decades, keyboard and mouse play a significant role in human-computer interaction. However, owing to the rapid development of hardware and software, new types of HCI methods have been required. In particular, technologies such as speech recognition and gesture recognition receive great attention in the field of HCI.



*Figure (i): Some common gestures that are used in real life*

This project implements hand gesture recognition using OpenCV and Python and uses Euclidean distance method to extract the image of the hand from a video.

**TABLE OF CONTENTS**

**Declaration ii**

**Certificate iii**

**Acknowledgement iv**

**Abstract v**

**Table of Contents vi**

**List of Figures vii**

**List of Abbreviations viii**

**1. Introduction 1**

**2. Background 2**

**3. Proposed Work 6**

**4. Implementation Details 7**

4.1 OpenCV 7

4.2 Python 7

4.3 NumPy 8

4.4 Augmented Reality 8

**5. Phases Of The Project 9**

5.1 Capture frames and convert to grayscale 9

5.2 Blur image 10

5.3 Thresholding 11

5.4 Contouring 11

5.5 Find convex hull and convexity defects 12

5.6 Interpretation of gestures 13

**6. Results and Discussion 14**

6.1 Background Estimation 14

6.2 Background Updating 14

6.3 Problem with Euclidean Distance Approach 15

**7. References 16**

**LIST OF FIGURES**

1.1: The overview of the proposed method for hand gesture recognition 1

2.1: The web cam used in [6] was HP-101 Specs-YUY2\_320x240 4

4.1 OpenCV logo 7

4.2 Python logo 7

5.1 Phases of the system 9

5.2 Greyscale image of a hand 10

5.3 Blurred image of a hand 10

5.4 Output after Greyscale, Blurring and Thresholding 11

5.5 Output after Contouring 12

5.6 Finding convex and defect points 12

5.7 Sample output 13

6.1 Formula 14

6.2 Formula 14

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **Abbreviation** | **Full-Form** |
| HCI | Human-Computer Interaction |
| PUI | Perceptual User Interface |
| OpenCV | Open Source Computer Vision Library |
| ROI | Region of Interest |
| RGB | Red Green Blue |
| ToF | Time of Flight |

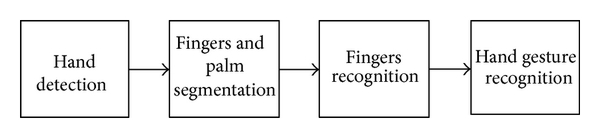
**1. INTRODUCTION**

Ever since the dawn of time, man has worked tirelessly to make life better. With the advent of computers and the internet, communication and information transfer has become extremely easy. But there’s always been one common way of interacting with these machines.

No matter how powerful and complex, you always have to be near a machine and somehow in physical contact with it to interact with it. But gesture recognition technology could change all that. If perfected and used correctly, it could actually render traditional input devices like keyboard, mice and touch screens redundant. In simple words, gesture recognition is the science of interpreting human gestures as input commands using mathematical algorithms. This may include full body motion recognition, or something small like a change in facial expression. Although it might seem small, however, reading facial expressions is in fact a more difficult task than recognizing more pronounced gestures.

Using the concept of gesture recognition, we present a system that will not only recognize gestures in real-time but also improve user experience by converting the gestures to animated figures during video chat. The project will make use of OpenCV and Python along with algorithms such as Euclidean Distance and Background Subtraction for the recognition.

The system will be developed in two phases, the first one being the recognition of gestures, where we will develop algorithms to effectively get the system to recognize the number of fingers and give count dynamically. The efficacy of the system will be tested by implementing a real-time application on it. The second module will include using this system to recognize gestures that are stored in the database and give animated figures during video chat in order to make user experience better.



*Figure 1.1: The overview of the proposed method for hand gesture recognition*

**2. BACKGROUND**

From our analysis of the existing systems and research papers, we found that:

In [1], the author is mainly concerned with image processing and computer vision concepts for interpretation of gestures. Instructions or commands can be conveyed to a machine or robot with the use of gestures. This phenomena is known as Human Machine Interaction or Human Computer Interaction (HCI). Hang gestures are an ideal way of exchanging information between humans, computers, robots and other devices. The study is calculating skeleton of the hand using distance transformation technique and using it for recognition instead of using the entire hand, as it robust against translation, rotation and scaling. Skeleton is computed for each and every hand posture in the entire hand motion and superimposed on a single image called as Dynamic Signature of the particular gesture type. Gesture is recognized by using the Image Euclidean distance measure by comparing the current Dynamic Signature of the particular gesture with the gesture Alphabet set.

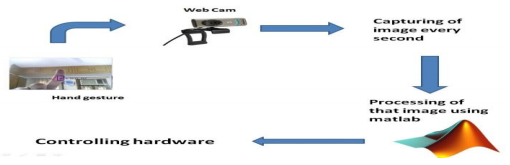
In [2], the author presents a new Euclidean distance for images, which is called image Euclidean distance (IMED). Unlike the traditional Euclidean distance, IMED takes into account the spatial relationships of pixels. Therefore, it is robust to small perturbation of images. The author argues that IMED is the only intuitively reasonable Euclidean distance for images. IMED is then applied to image recognition. The key advantage of this distance measure is that it can be embedded in most image classification techniques such as SVM, LDA, and PCA. The embedding is rather efficient by involving a transformation referred to as standardizing transform (ST). The study shows that ST is a transform domain smoothing. Using the face recognition technology (FERET) database and two state-of-the-art face identification algorithms, the author demonstrates a consistent performance improvement of the algorithms embedded with the new metric over their original versions.

In [3], the image Euclidean distance (IMED) considers the spatial relationship between the pixels of different images and can easily be embedded in existing image recognition algorithms that are based on Euclidean distance. IMED uses the prior knowledge that pixels located near one another have little variance in gray scale values, and defines a metric matrix according to the spatial distance between pixels. In this paper, the author proposes an adaptive image Euclidean distance (AIMED), which considers not only prior spatial knowledge, but also prior gray level knowledge from images. The most important advantage of the proposed AIMED over IMED is that AIMED makes the metric matrix adaptive to the content of the concerned images. Two ways of using gray level information are proposed. One is based on gray level distances, and the other is based on cosine dissimilarity of gray levels. Experiments on two facial databases and a handwritten digital database show that AIMED achieves the highest classification accuracy when it is embedded in nearest neighbor classifiers, principal component analysis, and support vector machines.

In [4], hand gesture interpretation is an open research problem in Human Computer Interaction (HCI), which involves locating gesture boundaries (Gesture Spotting) in a continuous video sequence and recognizing the gesture. Existing techniques model each gesture as a temporal sequence of visual features extracted from individual frames which is not efficient due to the large variability of frames at different timestamps. In this paper, the author proposes a new sub-gesture modelling approach which represents each gesture as a sequence of fixed sub-gestures (a group of consecutive frames with locally coherent context) and provides a robust modelling of the visual features. This approach is further extended to the task of gesture spotting where the gesture boundaries are identified using a filler model and gesture completion model. Experimental results show that the proposed method outperforms state-of-the-art Hidden Conditional Random Fields (HCRF) based methods and baseline gesture spotting techniques.

In [5], we see that the information conveyed in seminars, project presentation or even in class rooms can be effective when slideshow presentation is used. There are various means to control slides which require devices like mouse, keyboard, or laser pointer etc. The disadvantage is one must have prior knowledge about the devices in order to operate them. This paper proposes two methods to control the slides during a presentation using bare hands and compares their efficiencies. The proposed methods employ hand gestures given by the user as input. The gestures are identified by counting the number of active fingers and then slides are controlled. Unlike the conventional method for hand gesture recognition which makes use of gloves or markers or any other devices, this method does not require any additional devices and makes the user comfortable. The proposed method for gesture recognition does not require any database to identify a particular gesture. The experiment was tested under different kinds of light sources like incandescent bulb, fluorescent lamp and natural light.

In [6], the author focuses on the fact that humans communicate with one another not only through their vocal abilities but also through the gestures that they make. A gesture can go a great way in putting through our point and making the other person understand us and computers are no different in this and can be controlled and made to respond to hand gestures using a gesture recognition system. The primary goal of a gesture recognition system is to identify specific human gesture and use the gesture for controlling the device in the manner specified for that gesture in the gesture recognition system. By implementing real time gesture recognition system, a user can control a computer by simply doing gestures in front of the web cam which is attached with the laptop/computer. And to implement this system, the study has proposed real-time hand tracking algorithm, extraction algorithm and feature extraction.



*Figure 2.1: The web cam used in [6] was HP-101 Specs-YUY2\_320x240*

In [7], the author emphasises that a gesture recognition system should identify specific human gestures and use them to convey information for device control and that a user can control a computer by doing a specific gesture in front of a video camera linked to the computer by implementing a real time gesture recognition. This study covers various issues like what are gestures, their classification, their role in implementing a gesture recognition system, system architecture concepts for implementing a gesture recognition system, major issues involved in implementing a simplified gesture recognition system, exploitation of gestures in experimental systems, importance of gesture recognition system, real time applications and future scope of gesture recognition system.

In [8], the author proposes a non-complex and greater speed motion history image related system to recognize hand gestures. Hand gesture recognition in aspect to human machine interface is being developed vastly in recent days. Because of the disturbance of lighting and background being not plain, many visual hand gesture recognition systems operate or show successful results only in restricted background. In this study, the main focus is on applying pointing behaviour for the human machine interface. Now days, the gesture recognition has been a new developmental and experimental thing for most of the human related electronics. This system allows people to operate electronic products more conveniently. A gesture recognition method is suggested which will be an interface between human machine interaction i.e. HMI. Some non-complex algorithm and hand gestures to decrease the hand gesture recognition complexity have been proposed and would be more easy and simple to control real-time computer systems.

**3. PROPOSED WORK**

The whole project aims at successfully recognizing gestures and using the system in a video chat feature to enhance user experience. The project development has been divided into two phases:

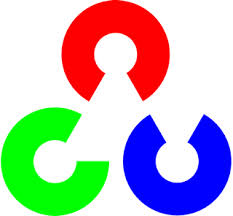
* The first phase included getting the system to start recognizing gestures. For this, we have used different algorithms including Euclidean Distance, Background Estimation, Background Update and others. When our system started giving finger count dynamically, we validated our system’s results by designing a volume control module where we controlled the volume using hand gestures.
* The second phase involves the fusion of augmented reality and gesture recognition and using the gestures recognized by our system to give animated visuals to a user during video chat and enhance his/her experience. Gestures would be written to a file and read by the augmented reality software and different visuals will be triggered.

**4. IMPEMENTATION DETAILS**

Technologies like OpenCV, Python, NumPy and Augmented Reality have been used in this project.

**4.1 OpenCV**

**OpenCV (Open Source Computer Vision Library)** is a [library of programming functions](http://en.wikipedia.org/wiki/Library_%28computing%29) mainly aimed at real-time [computer vision](http://en.wikipedia.org/wiki/Computer_vision), developed by [Intel](http://en.wikipedia.org/wiki/Intel_Corporation), and now supported by [Willow Garage](http://en.wikipedia.org/wiki/Willow_Garage) and Itseez. It is free for use under the [open source](http://en.wikipedia.org/wiki/Open_source) [BSD license](http://en.wikipedia.org/wiki/BSD_license). The library is [cross-platform](http://en.wikipedia.org/wiki/Cross-platform). It focuses mainly on real-time image processing.



*Figure 4.1 OpenCV logo*

**4.2 Python**

**Python** is a widely used [general-purpose](http://en.wikipedia.org/wiki/General-purpose_programming_language), [high-level programming language](http://en.wikipedia.org/wiki/High-level_programming_language). Its design philosophy emphasizes code [readability](http://en.wikipedia.org/wiki/Readability), and its syntax allows programmers to express concepts in fewer [lines of code](http://en.wikipedia.org/wiki/Lines_of_code) than would be possible in languages such as [C](http://en.wikipedia.org/wiki/C_%28programming_language%29). The language provides constructs intended to enable clear programs on both a small and large scale.



*Figure 4.2 Python logo*

**4.3 NumPy**

**NumPy** is the fundamental package for scientific computing with Python. NumPy’s main object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. It contains among other things:

* a powerful N-dimensional array object
* sophisticated (broadcasting) functions
* tools for integrating C/C++ and Fortran code
* useful linear algebra, Fourier transform, and random number capabilities

**4.4 Augmented Reality**

The origin of the word *augmented* is *augment*, which means *to add something*. In the case of augmented reality (also called AR), graphics, sounds, and touch feedback are added into our natural world. Unlike virtual reality, which requires you to inhabit an entirely virtual environment, augmented reality uses your existing natural environment and simply overlays virtual information on top of it. As both virtual and real worlds harmoniously coexist, users of augmented reality experience a new and improved world where virtual information is used as a tool to provide assistance in everyday activities.

**5. PHASES OF THE PROJECT**

The work on the project was divided into a number of phases. The aim was to develop a working system that could recognize gestures.

*Fig. 5.1: Phases of the system*

**5.1 Capture frames and convert to grayscale**

Our ROI (Region of Interest) is the hand region, so we capture the images of the hand using the web camera and convert them to grayscale.

**Why grayscale?**

We convert an image from RGB to grayscale and then to binary in order to find the ROI i.e. the portion of the image we are further interested for image processing. By doing this our decision becomes binary: “yes the pixel is of interest” or “no the pixel is not of interest”.



*Fig. 5.2: Greyscale image of a hand*

**5.2 Blur image**

We’ve used Gaussian Blurring on the original image. We blur the image for smoothing and to reduce noise and details from the image. We are not interested in the details of the image but in the shape of the object to track.

By blurring, we create smooth transition from one color to another and reduce the edge content. We use thresholding for image segmentation, to create binary images from grayscale images.

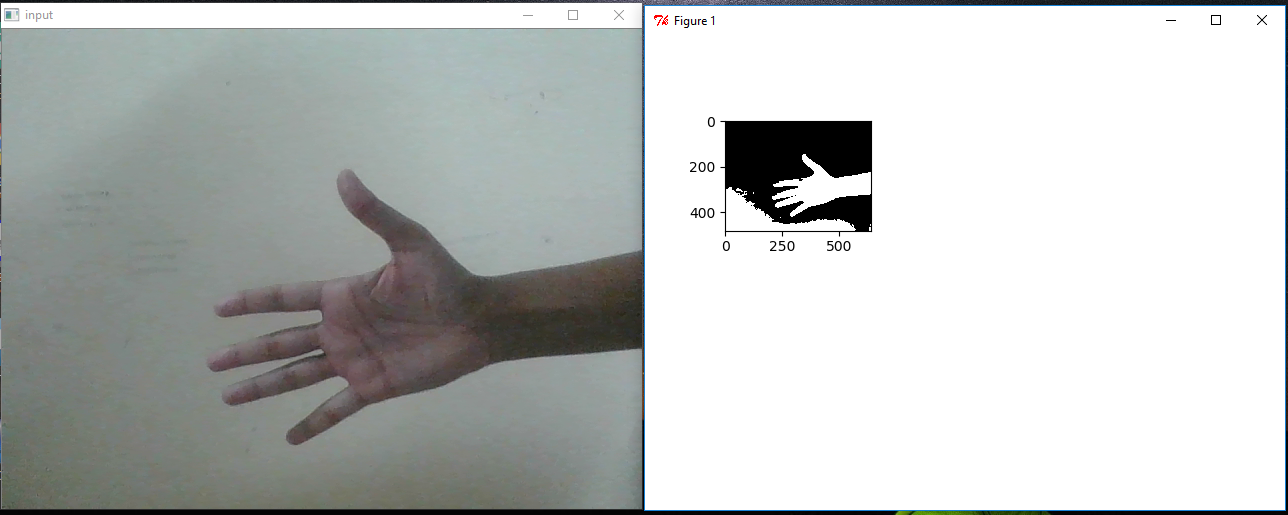


*Fig. 5.3: Blurred image of a hand*

**5.3 Thresholding**

In very basic terms, thresholding is like a Low Pass Filter by allowing only particular color ranges to be highlighted as white while the other colors are suppressed by showing them as black.

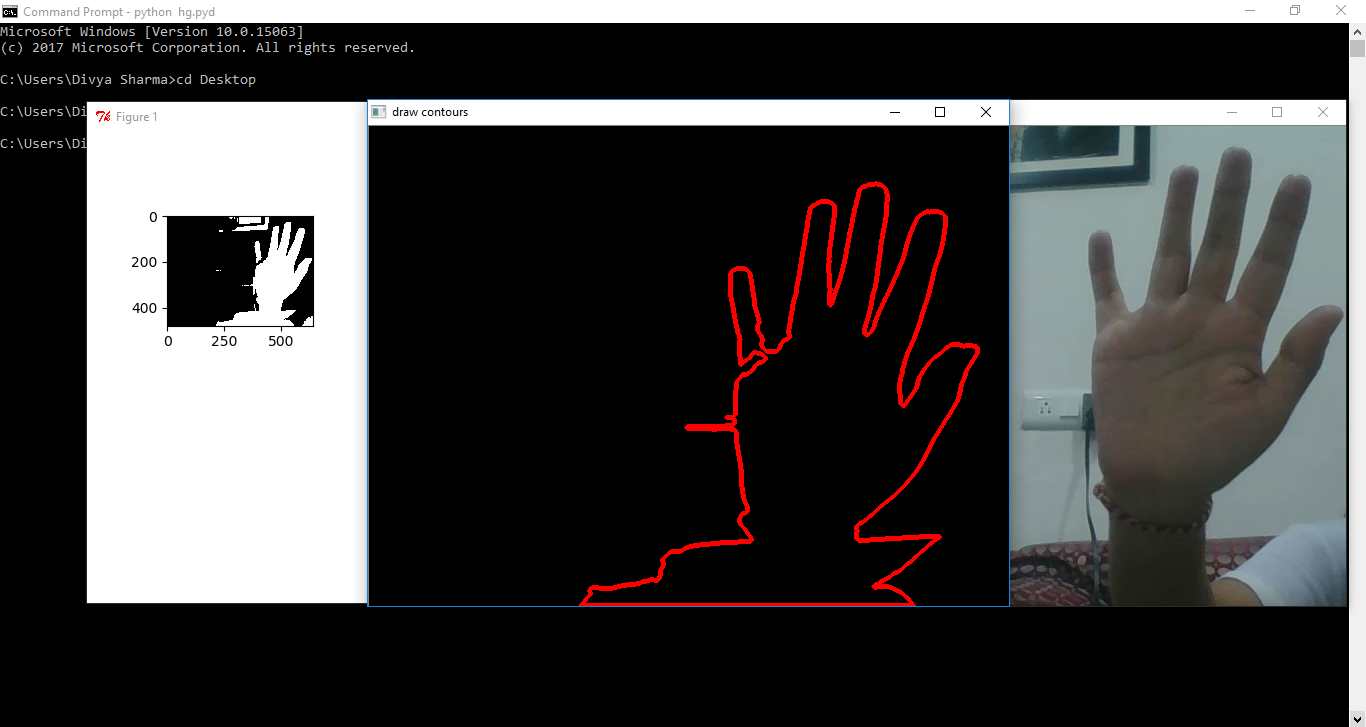
We’ve used Otsu’s Binarization method. In this method, OpenCV automatically calculates/approximates the threshold value of a bimodal image from its image histogram. But for optimal results, we may need a clear background in front of the webcam which sometimes may not be possible.



*Fig. 5.4: Output after Greyscale, Blurring and Thresholding*

**5.4 Contouring**

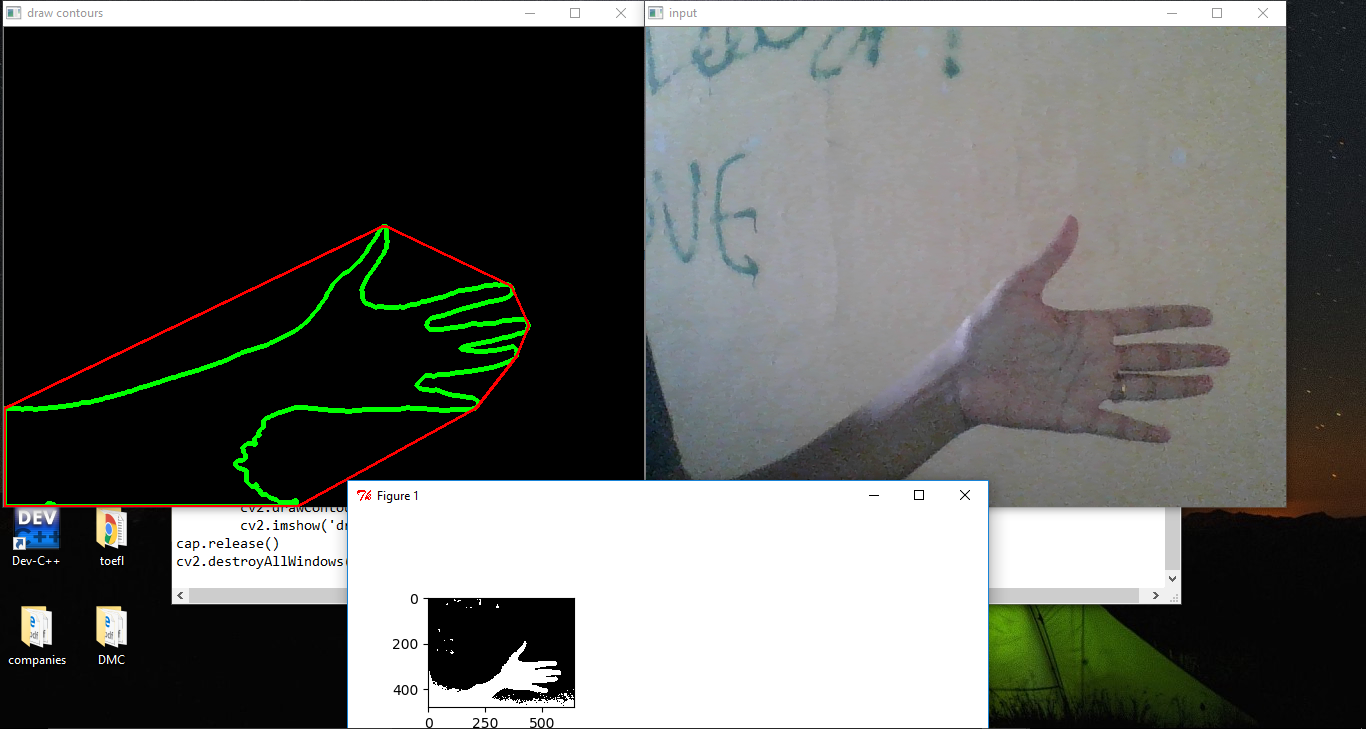
Contour can be explained simply as a curve joining all the continuous points (along the boundary), having same color or intensity. The result of findContours function is a Python list, where it contains all objects boundary points as separate lists. So to find number of objects, find length of this list.



*Fig. 5.5: Output after Contouring*

**5.5 Find convex hull and convexity defects**

We now find the convex points and the defect points. The convex points are generally, the tip of the fingers. But there are other convex point too. So, we find convexity defects, which is the deepest point of deviation on the contour. By this we can find the number of fingers extended and then we can perform different functions according to the number of fingers extended.

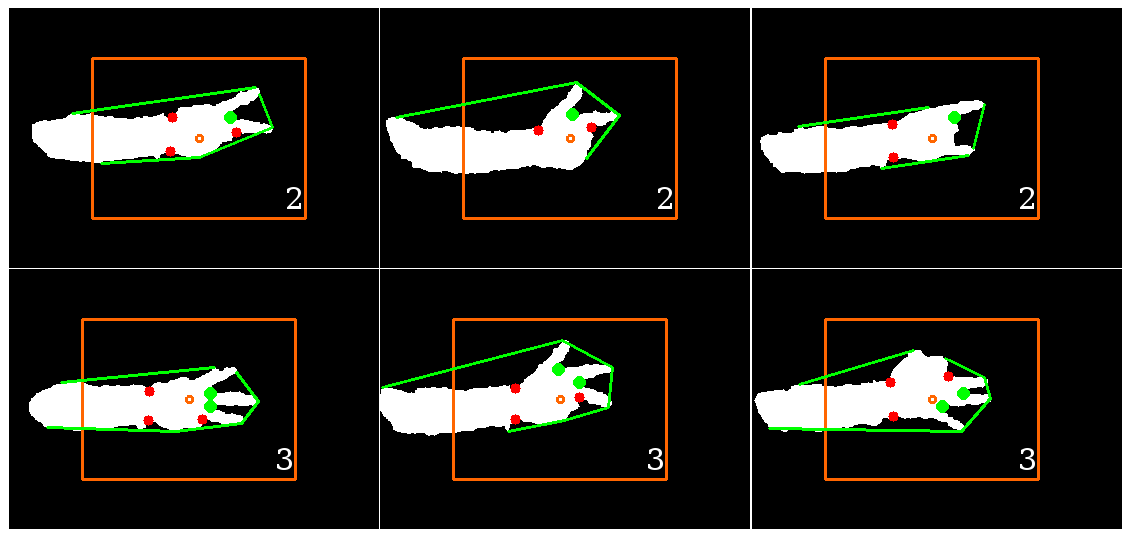


*Fig. 5.6: Finding convex and defect points*

**5.6 Interpretation of gestures**

As the final step, our system will be able to recognize the following gestures:

1. Counting number of fingers
2. Volume Control



*Fig. 5.7: Sample output*

**6. RESULTS AND DISCUSSION**

Our system successfully recognizes gestures and gives finger count. We validated the results using volume control. We intent to fuse augmented reality with the system and make a video feature to enhance user experience.

**6.1 Background Estimation**

It is proposed to estimate an initial background image using a large number of pictures and the median filter operating in the temporal axis. The assumption is that the pixel stays in the background for more than half of the frames used for the estimation.



*Fig. 6.1: Formula*

Where B is the Background Image, I is the Frame Image and n is the number of frames to be used for the approximation of the background.

**6.2 Background Updating**

To avoid false object detection (false negatives) that might be caused by slight but steady background lightning changes, a knowledge-based background updating algorithm is proposed. The background image that has been given or estimated with the background estimation algorithm has to be updated in regular time intervals. Using the knowledge of the detected objects in a scene, it is possible to define a selective update, which computes a new background value only if a point is not marked as an object pixel.

The new background is the weighted average between the current background objects and the old background.

  
*Fig. 6.2: Formula*

Where Bj is the jth updated background image (j = I mod K), Ii is the current image, and a (0<=a<=1) is a scalar representing the importance of the current data and the learning rate of the model. K controls the update frequency of the background to avoid the updating for every frame.

**6.3 Problem with Euclidean Distance Approach**

The Euclidean Distance approach calculates distance of the centre of the hand to the convex points. Convex points are actually the fingertips of our hand. Hand gesture is recognized by using the Image Euclidean distance measure by comparing the current Dynamic Signature of the particular gesture with the gesture Alphabet set. There is a new Euclidean distance for images, which is called image Euclidean distance (IMED). Unlike the traditional Euclidean distance, IMED takes into account the spatial relationships of pixels. Therefore, it is robust to small perturbation of images. It is argued that IMED is the only intuitively reasonable Euclidean distance for images. IMED is then applied to image recognition. The key advantage of this distance measure is that it can be embedded in most image classification techniques such as SVM, LDA, and PCA. The embedding is rather efficient by involving a transformation referred to as standardizing transform (ST). The study shows that ST is a transform domain smoothingthe image Euclidean distance (IMED) considers the spatial relationship between the pixels of different images and can easily be embedded in existing image recognition algorithms that are based on Euclidean distance. IMED uses the prior knowledge that pixels located near one another have little variance in gray scale values, and defines a metric matrix according to the spatial distance between pixels. In this paper, the author proposes an adaptive image Euclidean distance (AIMED), which considers not only prior spatial knowledge, but also prior gray level knowledge from images. The most important advantage of the proposed AIMED over IMED is that AIMED makes the metric matrix adaptive to the content of the concerned images. Two ways of using gray level information are proposed. One is based on gray level distances, and the other is based on cosine dissimilarity of gray levels. Experiments on two facial databases and a handwritten digital database show that AIMED achieves the highest classification accuracy when it is embedded in nearest neighbour classifiers, principal component analysis, and support vector machines.

The limitation of this approach is that it considers objects other than the hand in the scene.

**7. REFERENCES**

1. Reddy K.S., Latha P.S., Babu M.R. (2011) Hand Gesture Recognition Using Skeleton of Hand and Distance Based Metric. In: Wyld D.C., Wozniak M., Chaki N., Meghanathan N., Nagamalai D. (eds) Advances in Computing and Information Technology. Communications in Computer and Information Science, vol 198. Springer, Berlin, Heidelberg.
2. Wang, L., Zhang, Y., Feng, J.: On The Image Euclidean Distance. Center for Information Sciences School of Electronics Engineering and Computer Sciences, Peking University Beijing, 100871, China.
3. Li, J., Lu, B.-L.: An adaptive image Euclidean distance. China Pattern Recognition 42, 349–357 (2009).
4. M. R. Malgireddy, J. J. Corso, S. Setlur, V. Govindaraju, and D. Mandalapu, “A framework for hand gesture recognition and spotting using sub-gesture modeling,” in Proceedings of the 20th International Conference on Pattern Recognition (ICPR '10), pp. 3780–3783, August 2010.
5. Ram Rajesh J, Nagarjunan D, Arunachalam RM, and Aarthi R. Distance Transform Based Hand Gestures Recognition For Powerpoint Presentation Navigation. Advanced Computing: An International Journal ( ACIJ ), Vol.3, No.3, May 2012.
6. Shivanku Mahna, Ketan Sethi, Sravan Ch. Controlling Mouse using Hand Gesture Recognition. International Journal of Computer Applications (0975 – 8887), Volume 113 – No. 5, March 2015.
7. Viraj Shinde, Tushar Bacchav, Jitendra Pawar, Mangesh Sanap. Hand Gesture Recognition Using Web Camera. International Journal of Advanced Engineering & Innovative Technology, Volume 1, Issue 1, April-2014.
8. Neha S. Rokade, Harsha R. Jadhav, Sabiha A. Pathan, Uma Annamalai, Controlling Multimedia Applications Using Hand Gesture Recognition. International Research Journal of Engineering and Technology (IRJET). Volume: 02, 05 Aug 2015.
9. <https://gogul09.github.io/software/hand-gesture-recognition-p1>
10. <https://www.slideshare.net/shounakkatyayan/hand-gesture-recognition-42105322>
11. <http://www.edgefxkits.com/blog/what-is-gesture-recognition-technology-glimpse-of-the-future/>
12. <https://www.google.co.in/search?tbm=isch&q=hand+gestures&chips=q:hand+gestures,g_5:nonverbal+communication&sa=X&ved=0ahUKEwjkuveV1-HWAhWJuY8KHftSDMEQ4lYIMCgA&biw=1366&bih=637&dpr=1#imgrc=nPAFarYBu9KVSM>:
13. <https://gogul09.github.io/software/hand-gesture-recognition-p1>
14. https://sci-hub.bz/