Real Time Hand Gesture Recognition

Rogers Jeffrey Leo John(rl2689) and Minhaz Palasara(mup2101)

December 11, 2013

Abstract

Here we implement a method of recognizing human hand and gestures using convex hull and convexity defects form by the hand contour. We segment out the hand region from the live video feed captured using web cam on the browser. By applying heuristic we recognize Right and Left Hand with count of fingers in each hand which are further extended to identify the gesture type.

1 Introduction

Hand gesture recognition is a problem that has elicited significant attention and research as computer vision and the related algorithms have improved over the past few years. Such research is driven by the tremendous growth and variety in development of applications that require some form of gesture comprehension: these include remote hardware control, game controls, affective computing, and other endeavours in enhanced human-computer interaction . There are, however, fundamental limitations to most current systems for gesture detection based off training on a set of pre-defined gestures. Non-uniform lighting conditions and less-than-ideal camera resolution and depth of color limit the number and accuracy of possible gesture classifications in practice[3]. Moreover, the modelling and analysis of hand gestures is complicated by the variegated treatment required for adequate detection of static gestures, which represent a combination of different finger states, orientations, and angles of finger joint that are often hidden by self-occlusion [4,5]. In computer vision (CV) based solutions such as ours, hand gestures are captured by web cameras which offer resolutions that allow only a general sense of the figure state to be detected [4]. On the other hand emphasizing an understanding the movement of the hand as a pointer to the nature of the gesture allows for greater accuracy and better differentiation of gestures [5]. Additionally, the problem of hand-gesture recognition usually occurs in contexts where gestures involving finger conformation are accompanied by movement of the hands relative to the body.

Thus we implement a system that can identify simple gestures made by the hand. We implement a fast online processing algorithm for processing gestures from the live web cam feeds based in javascript. The detection process uses skin segmentation, convex-hull and convexity defect detection, hand localization based on the defects. We are tracking fingers on the both the Hands. We have implemented basic gesture for a single hand, and can be easily extended to complex gesture recognition without additional processing.

2 Motivation

Most of the existing gesture recognition systems require specialized hardware which is expensive or , require high computational power. Our goal was to develop a hand gesture recognition system that was computationally light-weight and could operate out of images and videos from the web-camera of a laptop on a light environment like Internet Browsers or Mobiles.

Another motivation for our project was http://revealjs.herokuapp.com, where simple swipe gestures are tracked using web cam based on the center of binary image formed after applying skin segmentation.

3 Investigation / Literature Review

We investigated the set of possible tools that we could use for our project. Most of the investigations were done, having the goal of our project in priority i.e. developing a hand gesture recognition system from live webcam feeds a computationaly inexpensive way.The papers and publications referenced in this report were analyzed to identify the methodology to be used for the operation.

[1] used 3D moments to identify Human Hand from the videos. But their approach required use of depth sensors like kinect for obtaining information in 3-dimensions. [9] suggests the use of color spaces like rgb and hsv for skin tone detection. Furthermore [12] compares the various color spaces like RGB, HSV, HSL and advocates the use of HSV for human skin detection as it results is more accurately segmented regions.

Investigating further into the mode of classification we found that [13] suggest the use of finite state automaton for the detection of hand gestures. Though this method uses inexpensive (computationally) black and white images, this system models each user's hand position as a state. So for a user to use a system implemented under thus model the user has to spend some time positioning the hand in each state, not enabling the seamless transition of the gesture. Furthermore maintaining data about each state is expensive.

Most of the literature we reviewed seem to suggest that the distinct shape of the human hand is valuable in identifying it from different objects. [14] had implemented the use of convex hull. Since we have only 5 fingers and two points at the base of the hand to be considered we can find the convex hull easily in $log(n)$ time and there for is computationally efficient. [14] also suggested the use of convexity defects for detecting the number of fingers and this set of features fit wel into our model

4 Technologies Used

As we wanted computationally lightweight system, we chose java script which would run with a small memory footprint within the Internet Browser. extensive research and investigations were carried out for choosing the technology and identifying the libraries necessary for this purpose

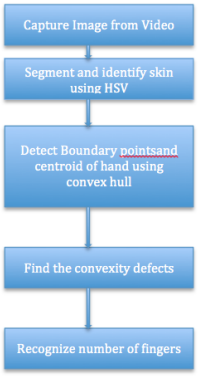

Figure 1: Flow Chart

The main challenge was to find a library for this purpose as most of the javascript libraries we found were for static image processing. We then decided to use and existing image processing library and tweak it to suite our purpose and found CV.js to be a perfect match, which provides implementation of contour, convex hull and convexity detection. See Table 1 for the list of tools investigated and a brief analysis about them.

5 Methodology

The methodology used can be seen from the figure. the project was divided into four major phases including,

1) Data Collection
2) Skin Detection/Segmentation
3) Identification of Features
4) Hand Detection

5.1 Data Collection

The main task involved in the data collection was the collection of static images of the human hand with various gestures, Around 90 images were captured using the web camera present in laptop. The hand images were taken in different orientation of the hand with different type of gestures(different fingers exposed/hidden). The dataset collected comprises of 6 folders marked 1 to 5 each containing images corresponding to the digits of the finger Figure 3 and Figure 2 are sample images that are images taken under different lighting conditions. This dataset was used as the benchmark for different heuristic based classifiers we build.

Figure 2: Improper lighting.

Figure 3: Proper lighting.

5.1.1 Issues Encountered

The first major issue we encountered was lighting. The hand recognition in static images was not good under average conditions of lighting. The second issue was presence of faces in the image. The removal of faces became critical for the application as the presence of faces was being detected as the skin tone of it was almost same as that of the hand. We had to gather more images with faces in order to prevent the recognition of them during hand recognition.

Backgrounds creating darker shade, leads to complex contour creation which might look a like hand and hence are selected by filter for further processing. We are assuming constant background, which can be easily removed from the consecutive frames.

5.2 Skin Detection

For separating the skin pixels from other objects we used the HSV Color Space. We run filter on all the image that collected to see that value of H and V in HSV space that provides high recognition rate. The HSV values from the various captured images to identify the optimal settings for these values. The optimal value that we found out are, V between 15 and 250, H between 3 and 33. In following image, screen on the bottom of canvas shows the binary image formed after segmentation.

5.3 Image Pre - processing

The resultant image form the segmentation is further processed to fill any holes in an image. We are using Dilate and Erode operation to perform this task. We are using implementation provided by CV.js.

6 Features Extraction

For recognizing the human hand from the live video feed the first step was to identify features that defined the presence of human hand. For identifying the features that would help us in the task we decided to use static images(as collected in the data collection section).

6.1 Hand contour

The unique characteristic shape of Hand has been used to detect them from images and videos. The shape of the hand is independent of the the viewpoint and lighting and can be used as a significant feature in combination with the skin color. This shape feature also helps in isolating the feature from the background. In order to extract any information from an image its essential to get contour of outer boundary of shapes. Figure 5 shows the plot of the hand contour in green

6.2 Convex Hull

To recognize that a finger is inside of the palm area or not, we used a convex hull algorithm. The convex hull algorithm is used to solve the problem of finding the biggest polygon including all vertices. Using this feature of this algorithm, we can detect finger tips on the hand. We used this algorithm to recognize if a finger is folded or not.Figure **??** shows the convex hull formed in red polygon for given hand image.

6.3 Convexity Defects

Convexity defects are valley points. In particular, convexity defects are sequences of contour points between two consecutive contour vertices on the contour Hull. Every defect has a start and an end point .The start and the end points correspond to the coordinates where the valley begins and ends . There are certain defects that occur around the edges of the hand. Such defects are generally the first and last defects in the list of defects which are sorted in increasing order of their X-coordinates. Such defects are ingnored.Figure 6 shows the hull marked as blue squares. We are also calculating the depth of the defects in the contour which is the average formed by the start and end point.

7 Building Classifier From the Found Basic Features

In this section we are explaining the processing on the features calculated above done in order to extract only Hand shapes and exclude any other erroneous

Figure 4: Convex Hull.

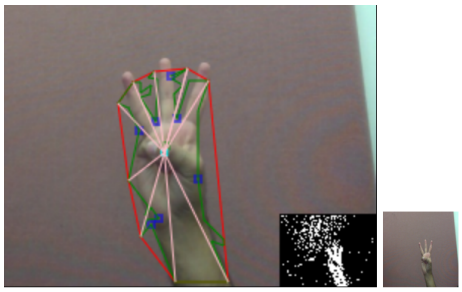

Figure 5: Convex Hull with Contour

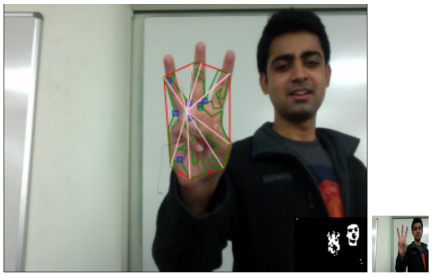

Figure 6: Hull with Defects.

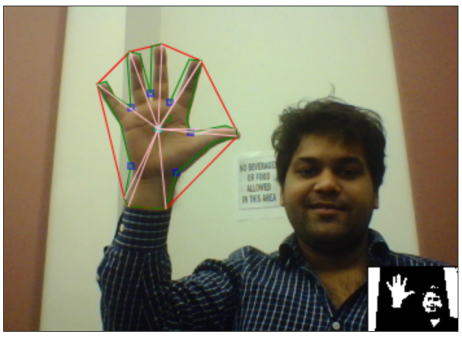

Figure 7: Locus Depth Features.

contour including face.

7.1 Locus of Defects/ Hull and Defects Depth

The locus of the defects was one of the important features we considered in identifying the hand. During the initial stages of experiments, for feature selection we found that the locus of the defects with respect center of the convex hull formed a circle. As shown in Figure 7 on projecting lines form the center of contour to hull. But this approach doesn't perform well as there are multiple convex hull point on the single fingers. It also counts the convex hull points which are at the bottom of hand. To avoid this problem we considered hull points which are larger than the locus center. Performance increased with some accuracy. Additionally I considered defects that are having atleast depth of 0.7 times distance from the locus.

Still this features fails due to multiple hull on single finger,leading to erroneous finger count and different depth proportion for different fingers. In this approach I removed the faces by percentages of defects that are having depth below threshold of 10. Here I followed the approach of removing face from the image, nstead of selecting hand from the image. Demo at the end of presentation video http://screencast.com/t/2WdCF1n58va8 shows the result by this method.

Note: File HandTracker.js has the code for this, it will display the locus on running DynamicFeaturesDisplay.html, but it is no longer used in the final version of classification.

7.2 Sorting of Defects to Find the Defects By Fingers

In order to improve finding of finger tip, I tried to remove the defects that are not between thumb and fingers(erroneous defects). Further this meaningful defects can be used to find the finger tip by considering starting and ending point of each defects - this approach completely removed the use of convex hull for finger tip detection. And hence the Hand type and Finger count can be derived.

But this approach fails to find out the correct order of for defects as it assumes that there are only two below locus of hull as shown in Figure 7. Since,

we are not removing any defects that lies below locus in this process its difficult to separate finger defects in case when there are multiple defect below the locus.

Note: File HandTracker.js has the code for this in functions sort and filter. It is commented out in the file and no longer used.

7.3 Counting Angle OF Defect and Bounding Box

As shown in Figure 8 the structure of Hand that can be exploit to filter out hand from the set of contours. It shows that the maximum angle that can two fingers be stretched is nearly 100. This angle can be only possible for defect between thumb and index finger. Hence I am using this property to filter out all the defects that are having angle greater than this value. Additionally, depth length is included in for filtering out the defects. Valid depth length that we have considered is than half of the Bounding box height inscribed by the hull.

Defect angle is calculated by cosine rule after calculating the distance of defect point from its starting and ending point.

As we are inscribing the bounding box, faces are filtered by counting the proportion of the convexity defects that are toward the bottom. I case of hand there will be few of the defects that will be at the bottom, while for the face the count increase drastically As shown in Figure 9 shows such instance that shows the number of defect are more toward bottom of box for face.

This filteration gives the optimal result and provides all the finger defects. After sorting all the defects, defect with highest angle would be at the end of hand, helping in identifying the type of hand. By moving in one direction for each defect its end point is only considered as the finger point, apart from first one for which both the points are finger points. Shared video on the link http://screencast.com/t/Y2RPTYeOvpaL shows the result from this classfication approach.

Figure 8: Hand Structure.

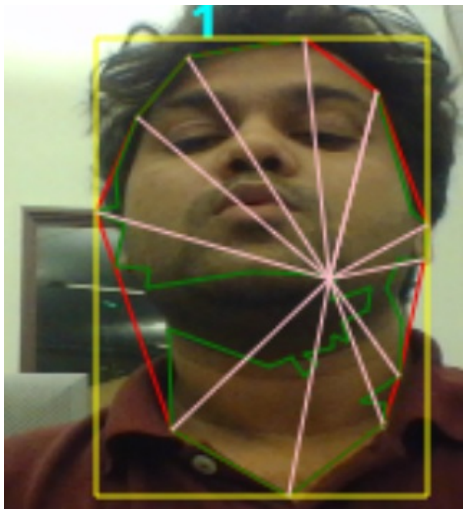

Figure 9: Face Box.

Figure 10: Hull center.

Software	Analysis
MatLab	Rich in functionality but computationally expensive and required installation
Open CV	Rich in functionality but computationally expensive and required installation
Open CV.js	Rich in functionality but requires open cv installed

Table 1: Software Analysis.

8 Gesture Recognition

video link http://screencast.com/t/I7oV8qUKh is the gesture recognition system that we have build using this classifier in real time.
This system used the open palm with five finger as the staring point and based on its movement it recognized swipe left, swipe right, swipe up, swipe down, zoom in, zoom out and stopped gestures. Explained in detail in coming sections.

9 Project Files

We are using library CV.js for image processing task like contour detection and hull detection, but it has been modified as per out need. We attached four html files for different result to see.

StaticHandTracking.html for applying tracking on an image.
DynamicHandTracking.html for applying tracking in real time, it will display all the features used. DynamicHandTracking.html for tracking hand at runtime,

No of fingers	Hand	Number of Times	Accuracy
1	Left	30	93.4
2	Left	30	95
3	Left	30	98
4	Left	30	100
5	Left	30	100
1	Right	30	91.4
2	Right	30	94
3	Right	30	97
4	Right	30	100
5	Right	30	100

Table 2: Recognition Results.

it will display the type of hand and number of fingers in it. GestureRecognitionhtml for displaying the gestures that are performed like swipe explained in above section and shown in videos.

10 Work Done By Me

1) Data collection(collecting photos of hands different people), Done after multiple trial and errors with differnt lighting conditions. Clic lab was found to provide optimal lighting for photo capture.
2) Using HSV for identification of human skin .
3) Feature indeitification. We worked together in trying out various combinations of features from the set of features identified above. Literature survey was conducted by us independently and we consolidated the results to identify features.
4) Feature selection, identifications and performance measure was divided among both the members.
5) After the selection of defect by researching, classifier was majorly build by my other partner. I was involved in the tuning and optimization part where I worked on Identifying the optimial set of defects and and the use maximum angle in the hand to identify the type of the hand.

11 Demo

The demo videos can be found at. First video is presentation video shows first classifier.

- http://screencast.com/t/2WdCF1n58va8
- http://screencast.com/t/Y2RPTYeOvpaL
- http://screencast.com/t/I7oV8qUKh

12 Results

We can see from table 2 that the overall accuracy rate is as high as 97

13 References

- Soontranon, N.; Aramvith, S.; Chalidabhongse, T.H.; , "Improved face and hand tracking for sign language recognition," Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on , vol.2, no., pp. 141- 146 Vol. 2, 4-6 April 2005
- Paulraj, M.P.; Yaacob, S.; Desa, H.; Hema, C.R.; Ridzuan, W.M.; Majid, W.A.; , "Extraction of head and hand gesture features for recognition of sign language," Electronic Design, 2008. ICED 2008. International Conference on , vol., no., pp.1-6, 1-3 Dec. 2008

- Dardas, N.H.; Petriu, E.M.; , "Hand gesture detection and recognition using principal component analysis," Computational Intelligence for Measurement Systems and Applications (CIMSA), 2011 IEEE International Conference on , vol., no., pp.1-6, 19-21 Sept. 2011
- Zhu, Hong-Min, Pun, Chi-Mun, "Movement Tracking in Real-Time Hand Gesture Recognition" Information Technology: Computer and Information Science 2010. IEEE/ACIS 2010. International Conference on , vol.1, no., pp. 135- 137 Vol. 2, 18-20 Aug 2010.
- Y. Wu, and T. S. Huang, "Hand modeling, analysis and recognition," IEEE SIGNAL PROCESSING MAGAZINE, vol. 18, no. 3, pp. 51-60, 2001.
- A Method for controlling mouse movement using relaltime camera,Hojoon Park Department of Computer Science Brown University, Providence, RI, USA
- A Hand Gesture Recognition Based Virtual Touch Wall,International Journal of Information and Education Technology, Vol. 2, No. 1, February 2012
- Visual Interpretation of Hand Gesturesfor Human-Computer Interaction: A ReviewVladimir I. Pavlovic, Student Member, IEEE,Rajeev Sharma, Member, IEEE and Thomas S. Huang, Fellow, IEEE,IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 19, NO. 7, JULY 1997
- Hand Sign Recognition through Palm Gesture and Movement Image Processing, EE 368, Spring 2012,Oleg Rumyantsev, Matt Merati, Vasant Ramachandran Stanford University
- Real-Time 3D Hand Gesture Recognition from Depth Image Lin Song1,a, Ruimin Hu1,b, Yulian Xiao2,c and Liyu Gong2 National Engineering Research Center for Multimedia Software, Computer School, Wuhan University, China
- http://www.cescg.org/CESCG-2004/web/Sedlacek-Marian/
- Davis James,Shah Mubarak,Recognizing Hand Gestures, University of Central Florida Orlando
- http://www.cpmr.org.in/CPMR-IJT/vol2/issue2/Articles/1.pdf