Real Time Hand Gesture Recognition

Minhaz Palasara (mup2101@columbia.edu)

December 11, 2013

1 Abstract

This report details implementation of a light weight algorithm for locating the human hand and recognizing gestures using convex hull and convexity defects formed by it. The hand is located by segmentation on the skin color in a live video feed in web browser using a standard web-cam. By applying heuristic, algorithm not only differentiates between the types of hand (Right or Left), it also provides the count of visible fingers in each hand. This information is used in identifying a predefined set of gestures.

2 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 on training on a set of predefined 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[1]. 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 [7] [6]. 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 [7]. 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 [6]. 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.

We have implemented a system that can identify simple gestures made by hands. It is an online algorithm running on JavaScript engine for locating gestures in a live web cam feed. The detection process comprises of skin segmentation, convex-hull and convexity defect detection, and hand localization based on the defects. The detection process also supports tracking of fingers on both the hands. Currently, basic gestures using single hand is supported, but it can be easily extended to track complex gestures.

3 Motivation

Most of the existing gesture recognition systems require a specialized hardware which is expensive or requires high computational power. Our goal is to develop a light weight and computationally inexpensive hand gesture recognition system. The system is developed in JavaScript, this would allow in deploying the system as a plug-in on a browser to explore any web page using hand gestures.

4 Literature Review

We have done an extensive study on the existing gesture recognition systems. We have listed some of the literature that proved helpful in the implementation of project. [5] uses 3D moments to identify Human Hand from videos using depth sensors like kinect. [3] suggests use of color spaces like RGB and HSV for skin tone detection. Furthermore, [2] 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 classification models for gesture recognition, [4] suggest use of finite state automaton for the detection of hand gestures. Though this method uses computationally inexpensive black and white images, the system models each user's hand position as a state. As a result, for a user to use a system implemented using this model requires to spend some time positioning the hands in each state, not enabling the seamless transition of gestures. Furthermore maintaining data of each hand state is expensive.

Most of the literature we reviewed emphasises on the distinct shape of the human hand. The hand structure is valuable in separating it from different objects. [4] is based on the use of convex hull. Since, we have only 5 fingers and two points at the base of the human hand, convex hull cab easily be found in log(n) time. Also, as described in [4] convexity defects can be used to find the number of fingers.

5 Technologies Used

As we want a computationally lightweight system, we chose JavaScript which can run with a small memory footprint on any Internet Browser with JavaScript engine. It also makes the implementation device independent, as the contemporary browsers are very well supported on all possible platforms. It also eases integration of the system as a service on specific website. We have used the source code of an existing image processing library named CV.js, and modified it to suite our purpose. It provides contour, convex hull and convexity detection. The following table lists libraries/languages/tools investigated and a brief analysis about them.

Software	Analysis
Matlab	Rich in functionality but computationally expensive, platform dependant
Open CV	Rich in functionality but computationally expensive, platform dependant
Open CV.js	JavaScript based, no pre-installation dependency

6 Implementation Details

The project is divided into four major phases,

1) Data Collection
2) Skin Detection/Segmentation
3) Feature Extraction
4) Hand Detection

6.1 Data Collection

To compare performance of different features for gesture recognition, we need a static set of images. This dataset is considered for benchmark on the performance of different classification models. Around 90 images were captured using a standard web camera (available in laptops). The hand images were taken in different orientations with different type of gestures (different fingers exposed/hidden). The dataset has 6 folders, each folder has images with number of the exposed fingers same as the folder number. Figure 2 and Figure 1 show sample images taken under different lighting conditions.

6.1.1 Issues Encountered

The first major issue we encountered was lighting. Hand recognition in static images is not accurate under an average lighting conditions, shown in Fig. 1. The second issue was the presence of the human face in the images. As similar skin tone could lead to false identification, removal of human face is essential. Backgrounds with darker shade leads to complex contour creation which can resemble human hand and hence are selected by filter for further processing.

We are assuming a static background, which can easily be removed from the consecutive frames.

6.2 Skin Detection

For separating out only the skin pixels we are using the HSV Color Space. We ran this filter on all the collected images to find the values of H and V (in HSV space) that provide high recognition rate. The optimal values are, V between 15 and 250, H between 3 and 33. In the Figure 4, screen on the bottom of canvas shows the binary image formed after applying the segmentation filter.

Figure 1: Improper Lighting

6.3 Image Pre-processing

The resultant image after applying the segmentation is further processed to fill any holes caused by the segmentation. We are using the Dilate and Erode operations for this purpose. The processed dataset has images with only human skin areas, hands and faces. This dataset is used to locate hand, type of hand and number of fingers. The complete pre-processing is kept very light in computation, hence it could be easily used in a real time setting without any considerable delays.

Figure 2: Proper Lighting

6.4 Features Extraction

To separate out the human hand in an image, we need to first identify the features that uniquely define it. We have listed the features used in the project in the following section. These features were first tested on the images in the dataset, and then it was adapted for detection in a live video. For classification, we are using complex features derived from the following basic features.

6.4.1 Hand Contour

The unique characteristics of the human hand are independent of the viewpoint and lighting. These features can be used in combination with the skin color to not only locate hand, also in isolating the background. In order to extract any information from an object of interest, it is essential to first get its contour (outer boundary). Fig. 4 shows hand contour (in green color) generated by the algorithm.

6.4.2 Convex Hull

The Convex Hull for a set of points in the euclidean space is the smallest convex set that contains all the points in the set, as shown in Fig. 3 and Fig. 4 in red color. This is one of the important features to uniquely identify the human hand. The human hand when fully open (all fingers are visible) has Hexagon as a convex hull. This is one of the filtration criteria to remove the human face and complex background, as it has convex hull with many sides (almost a circle), shown in Fig. 6. The Convex hull information could also be used to count the

Figure 3: Convex Hull

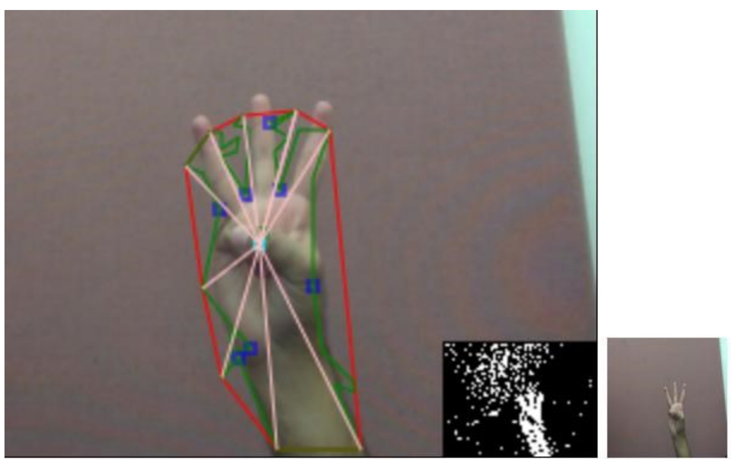

Figure 4: Convex Hull with Contour

number of visible fingers, as the polygon would change from a Hexagon to a Rhombus(one open finger).

6.4.3 Convexity Defects

The Convexity defects are the valley points of a hand. In particular, convexity defects are sequences of contour points between two consecutive contour vertices on the convex hull. The human hand has valley points at a larger depth relative to the size of hand . Every defect has a start and an end point, shown in Fig

4. in blue squares. The start and the end point corresponds to the coordinates where a valley begins and ends . There are certain defects that occur around the edges of the hand (near wrist). Such defects are generally the first and last defects in the list of defects which are sorted in in-creasing order of their X-coordinates. These defects are ignored from further processing. We are using the defect depth in addition to the contour, this models the unique structure of hand with heavy center mass and sparse boundary.

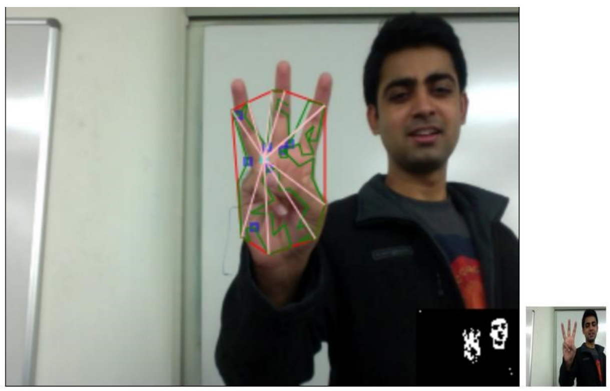

Figure 5: Convex Hull with Defects

6.5 Hand Detection

Above mentioned features are used to build a complex heuristics (filters) to locate the human hand, type of hand (right or left) and the number of fingers in each hand. The heuristics are listed in the following sections. These heuristics are used with decision tree classification model.

6.5.1 Locus of Defects/Hull and Defects Depths

The locus of the defects was one of the important features we considered in identifying the human hand. During the initial stages of experiments we found that the locus of the defects with respect to the center of the convex hull almost forms a circle. As shown in Figure 6, projection lines from the center of contour to the hull. This heuristic often fails due to false convexity points. It also considers the convex hull points which are at the bottom of hand. To avoid this problem, we consider points which are higher than the locus center. This heuristic helps in considerable increment in the accuracy. Additionally, we consider the defect depth (distance of a finger tip from the center of palm) for convex hull filtration. We consider points with depths of 0.7 times the distance from the locus center as the finger tips. However, this heuristic fails some of the times in count of fingers due to wrong hull points.

This heuristic helps in the human face removal. The human face produces almost a circular convex hull with lower defect depths. We use the threshold value of 10, chosen by run on the static images, as a criteria for consideration of a point as a defect. The hull with a higher percentage of defects with depth less than the threshold models an almost circular hull and hence filters the human face. Here we follow the approach of filtering out the human face(or objects other than hand), instead of selecting hand from an image. This heuristic is applied on a live video feed, the result is store in a video file named **locus-features.mp4** in the **result** folder at `https://github.com/MinhazPalasara/HandGesture_Recognition`

The HandTracker.js file has the code for heuristic. Run DynamicFeaturesDisplay.html to display the filter values. This heuristic is not used in the gesture recognition application.

6.5.2 Filtering Defects using Defect Depth

To improve finger tip localization, we filter out the defects that are not at almost same distance from thumb or finger tips. The resulting defects are sorted to find the starting and the ending point of a hull and finger defects. However, this approach fails in finding the correct order of defects as it assume that there are only two defects below the center of hull as shown in Figure 6. As we are not removing any defects that lie below the defect locus, it is difficult to separate out only the finger defects. It also fails in differentiating between the right and left hand.

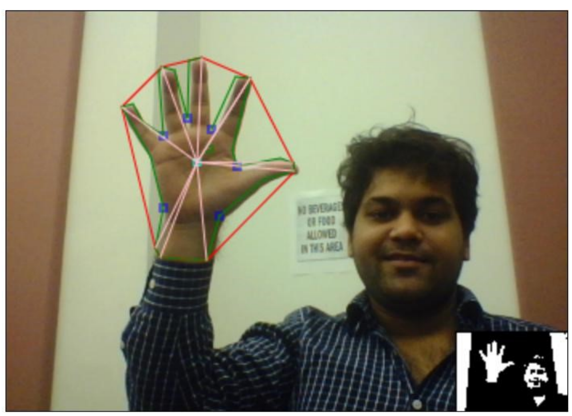

Figure 6: Locus Depth Features

The HandTracker.js file has this implementation, it is commented out in the file and is not part of final algorithm.

Verticle

Figure 7: Face Hull

6.5.3 Defect Angle and Bounding Box

In order to differentiate between the right and left hand and hence the starting and ending point, we are using the thumb as the reference point. Shown in Figure 8, the structure of human hand that can be exploited to filter out hand from the set of contour points. The maximum angle between two fingers is approximately 100 degrees. This angle is only possible between thumb and index finger. We are using this heuristic to filter out all the defects that are having angle greater than this value and the defect with value in this range is used as the reference point. The depth length is combined with this in filtering out the defects.

The defect angle is calculated by the cosine rule using the distance of defect from its starting and ending point. As we are inscribing a bounding box, faces are filtered by counting the percentage of the convexity defects toward the bottom half. For the human hand, few defects (ideally 2) are present at the bottom, while for the face the count increases drastically As shown in Figure 9.

This filtration provides fairly accurate results and locates all the finger defects. After sorting all the defects, defect with the highest angle is considered the end of hand (thumb), this helps in identifying the type of hand. While moving in one direction from the end of hand, for every defect its end point is considered as the finger tip except for the first one. For the starting point both ends are finger tips. This filtration is applied in a live video feed, the result is stored in a video file named **hull-defects-thumb.mp4** in the **result** folder at `https://github.com/MinhazPalasara/HandGesture_Recognition`.

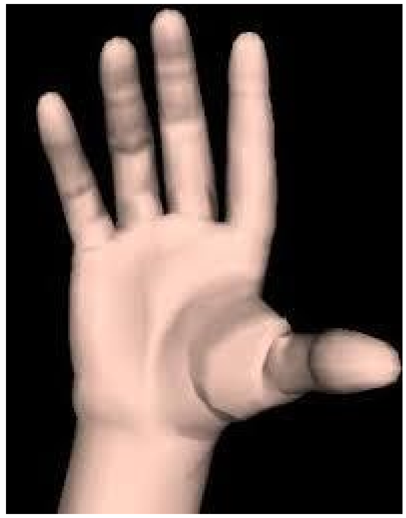

Figure 8: Hand Structure

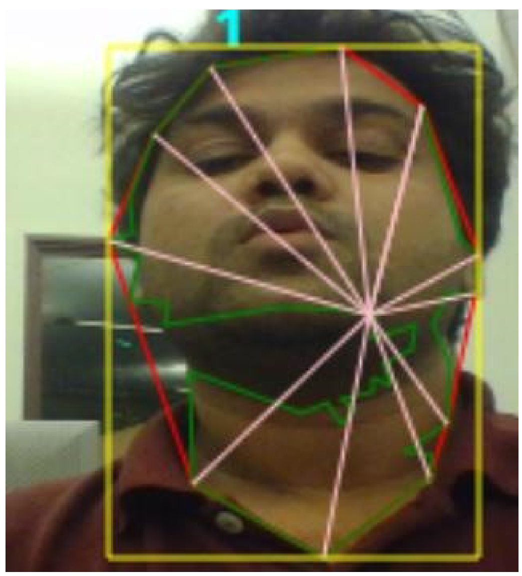

Figure 9: Face Convex Hull

7 Gesture Recognition Application

We have extended the hand localization algorithm to a simple predefined gesture recognition system. Please, see the videos named **Live-HandGesture-Recognition-1.mp4** and **Live-HandGesture-Recognition-2.mp4** in the **re-**

Figure 10: Hull Center

sult folder at `https://github.com/MinhazPalasara/HandGesture_Recognition`

This system uses an 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.

8 Conclusion

The system works with a really high success rate in locating hands and the number of visible fingers. However, the results gets affected in a poor lighting conditions. Use of a specialized hardware such as depth sensors could make the results consistent even in poor lighting conditions.

References

[1] Nasser H Dardas and Emil M Petriu. Hand gesture detection and recognition using principal component analysis. In *Computational Intelligence for Measurement Systems and Applications (CIMSA), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

[2] James David and Mubarak Shah. Recognizing Hand Gestures. `http://web.cse.ohio-state.edu/~jwdavis/Publications/eccv.pdf`, 1994.

[3] Oleg Rumyantsev, Matt Merati, and Vasant Ramachandran. Hand sign recognition through palm gesture and movement. *Image Processing*, 2012.

[4] Dr. V. Ramalingam S. Nagarajan, Dr. T.S.Subhasisni. Vision Based Real Time Finger Counter for Hand Gesture Recognition. `http://www.cpmr.org.in/CPMR-IJT/vol2/issue2/Articles/1.pdf`, 2012.

[5] N Soontranon, Supavadee Aramvith, and Thanarat H Chalidabhongse. Improved face and hand tracking for sign language recognition. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 2, pages 141–146. IEEE, 2005.

[6] Ying Wu and Thomas S Huang. Hand modeling, analysis and recognition. *Signal Processing Magazine, IEEE*, 18(3):51–60, 2001.

[7] Hong-Min Zhu and Chi-Man Pun. Movement tracking in real-time hand gesture recognition. In *Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on*, pages 240–245. IEEE, 2010.