

## A MICRO-GESTURE RECOGNITION ON THE MOBILE WEB CLIENT

Chengfeng Jian<sup>1\*</sup>, Wei Zhang<sup>1</sup> and Yue Ying<sup>2</sup>

<sup>\*1</sup> Computer Science and Technology College, Zhejiang University of Technology, Hangzhou, China;

<sup>2</sup> Zhejiang Radio & Television University, Hangzhou, China.

Email: jiancf@zjut.edu.cn

### ABSTRACT

Micro-gesture recognition is the key technology on the vision-based sktech on the mobile client. This paper introduces one kind of micro-gesture recognition algorithm for the mobile Web client. Micro-gesture means the micro-motion of fingers like swing or bending. Research based on HTML5, Javascript and webRTC framework, and with the help of open source Javascript package: objectdetect.js and JsCV Core. The video information is acquired by the Web camera, and the hand is divided into the palm and the finger domains. Palm domain is used to locate the position of the hand, finger domain is used to extract the finger information after the processes of color extraction, region segmentation and contour extraction.

**Keywords:** Web client, WebRTC, Color extraction, Region segmentation, Contour extraction.

## 1. INTRODUCTION

### 1.1 Motivation

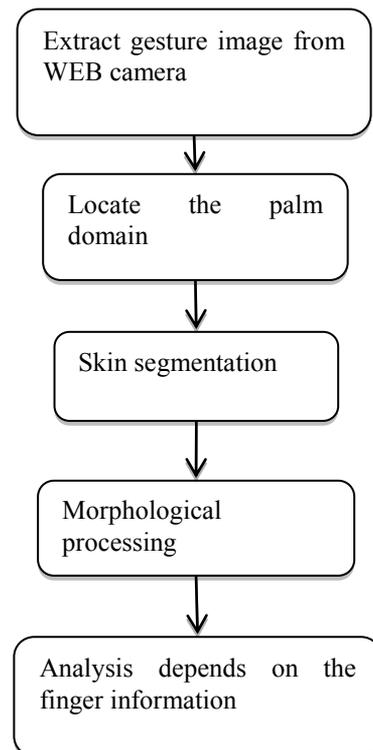
The popularity of mobile devices, resulting in the focus of consumer's device gradually moved to the mobile terminal. With the development of the internet application, consumers are no longer limited to the original interactive mode, so that the developer of internet application realized that the interactive needs to be friend, convenient.

For developers, there are three systems: IOS, Android, Windows Phone which makes developers need to develop at least three versions of the application, this shows that the importance of cross platform. The reasonable way to solve the problem of platform limitation is developing the WEB application.

Gesture recognition is the focus of computer vision research [1] [2]. Hand gesture has the natural ability to show the personal ideas. According to different gestures, the system identifies the corresponding events which demonstrate the potential of gesture recognition in human-computer interaction.

## 2. RECOGNITION PROCESS

The flowchart of micro-gesture recognition as shown in Figure 1 and its main steps are discussed in the following:



**Figure 1.** The process of the micro-gesture recognition

## 2.1 Image acquisition and location

Image acquisition based on the WebRTC framework proposed by Google, which can achieve the browser without plug-in video play by calling the JavaScript API and HTML5 standards. At present, Opera, Chrome, and Firefox browser have all supported this decoding standard. JavaScript code can be used to achieve the call to the camera, the image information is reflected in the canvas, then you can use the image information to the gesture positioning. Location using open source project 'objectdetect.js'. This API uses the Haar feature and adopts the cascade classifier to detect the target, and the Haar feature can be trained by a large number of target images. In this paper, the target image is the palm area image, as shown in Figure 2.



Figure 2. Palm domain position

## 2.2 Skin segmentation

Relative to RGB, HSV is more able to show the contact of the perception of color [3], so transform the color space from RGB to HSV. H (hue) is hue, means color representation of the basic attributes, such as red, yellow, and so on; s (saturation) is saturation, means color purity. V (value) is the lightness (also called luminance).

G, B, R information obtained from the image is 0~255, where the three values are converted to 0 to 1, as follows:

$$\begin{cases} r = \frac{R}{255} \\ g = \frac{G}{255} \\ b = \frac{B}{255} \end{cases}$$

Set max for R, G, B, the maximum value, min for R, G, B, the minimum value. Since h is the angle of hue angle, in order to make it easy for computer to calculate. Therefore, the overall h divided by 60°, to locate in the range of 0 to 1, and s and V are located in the range of 0 to 1. H, S, V transform as follows:

$$H = \begin{cases} 0, & \text{if max} = \text{min} \\ \frac{1}{6} \times \frac{g - b}{\text{max} - \text{min}}, & \text{if max} = r \text{ and } g \geq b \\ \frac{1}{6} \times \frac{g - b}{\text{max} - \text{min}} + 1, & \text{if max} = r \text{ and } g < b \\ \frac{1}{6} \times \frac{b - r}{\text{max} - \text{min}} + \frac{1}{3}, & \text{if max} = g \\ \frac{1}{6} \times \frac{r - g}{\text{max} - \text{min}} + \frac{2}{3}, & \text{if max} = b \end{cases}$$

$$S = \begin{cases} 0, & \text{if max} = 0 \\ \frac{\text{max} - \text{min}}{\text{max}} = 1 - \frac{\text{min}}{\text{max}}, & \text{otherwise} \end{cases}$$

$$V = \text{max}$$

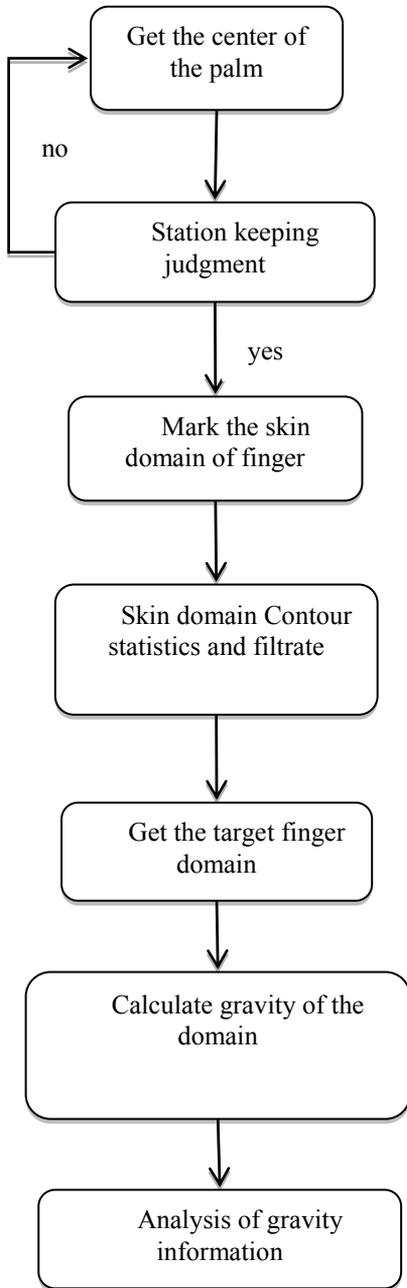
Because there are many kinds of human skin, different light also make color range expanded in the RGB images in. In HSV, H component means the color information of image, the effects of light intensity is slow. In the hue component, the skin color is concentrated in a small area. Because of this feature, using the HSV model can make the skin segmentation more perfect. Roughly case, H satisfies 0.59<H<1 or 0<H<0.1, s satisfies 0<S<1 and V satisfy the 0.4<V<1, can be considered skin. The skin color pixels value is set to 255, the non-skin color pixels value is set to 0. as shown in Figure 3.



Figure 3. Skin color binary graph

## 3. THE RECOGNITION ALGORITHM

The flowchart of recognition algorithm as shown in Figure 4 and its main steps are discussed in the following.



**Figure 4.** The process of recognition algorithm

### 3.1 Calculate gravity of the palm domain

For the skin color binary graph, we have set the skin color pixels to 255, so we can obtain the coordinate point of palm domain pixel, according to the center of gravity of the below formula to figure out the palm domain:

$$C(x, y) = \left( \frac{\sum_{i=0}^{N-1} P_i(x)}{N}, \frac{\sum_{i=0}^{N-1} P_i(y)}{N} \right)$$

Where  $C(x, y)$  is Coordinates of center of gravity,  $P_i(x)$  is cross coordinates of the pixels of the skin color,  $P_i(y)$  is vertical coordinates of the pixels of the skin color,  $N$  is total number of pixels of skin color.

### 3.2 Station keeping judgment

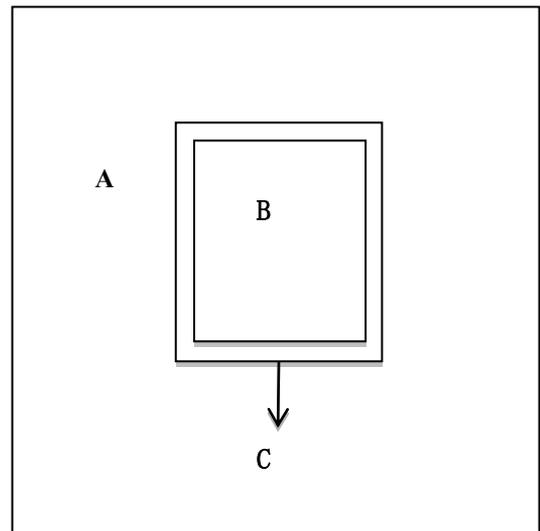
Web client gesture recognition, in pursuit of high efficiency and real-time performance, will lose the accuracy. in order to achieve the efficiency and accuracy of the phase balance, the use of a station keeping judgment method. After the last step, we can get a center of gravity  $(x, y)$ , with the passage of time, we will get a lot of center of gravity. Then the center of gravity coordinates will be grouped into a group of 50, if there are 30 points in a range, means the hand station keeping.

### 3.3 Mark the skin domain

In order to effectively deal with the color information, we need mark the pixel area which can extract areas we need and exclude areas we don't need. The general idea is: first, the number value of the whole gesture image is initialized; then, in accordance with the way from left to right, from top to bottom, mark the skin color pixel area.

#### (1) Initialize number

The image region is shown in Figure 5: A is the whole image region; B is the gesture image region; C is the image region extend 2 pixels size for B region.



**Figure 5.** Image region

Image initialization is initialized all the number value of the pixels within the region C to 0, as shown in Figure 6: the black border area is B region, gray box represents the skin pixels, white boxes represent the non-skin color pixels.

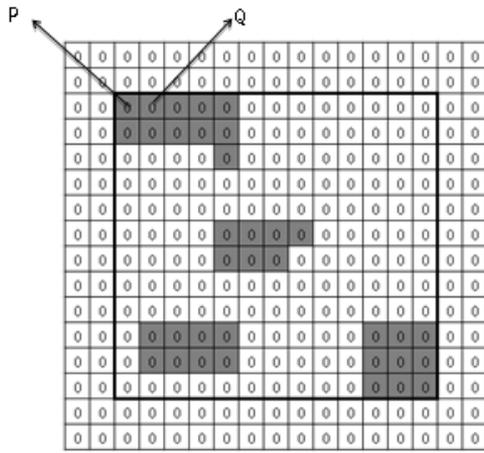


Figure 6. Initialize number

(2) Mark method

As shown in the picture, we mark from left to right, top to bottom, assuming that the coordinates of the skin pixels are  $(x, y)$ , the number value is num. We need to determine whether  $(x-1, y)$ ,  $(x-1, Y-1)$ ,  $(x, Y-1)$ ,  $(x+1, Y-1)$  four points have been changed number value. If all the number value is 0, it means the pixel regions were not renumbered, this region should be the assigned the mark number as  $num=num+1$ ; if one point has been renumbered, we should assigned the mark number as the mark number of the point has been renumbered. According to the above principles, as shown in Figure 6, skin pixels we first mark is P, the current  $num=0$ . Because the point P on the four direction point number = 0, so the number value of P should be  $num=num+1$ , which is 1. For Q, because P points on the left front and P has been renumbered 1, so Q number value should renumber as the same value, which is 1. Mark the skin area as shown in Figure 7:

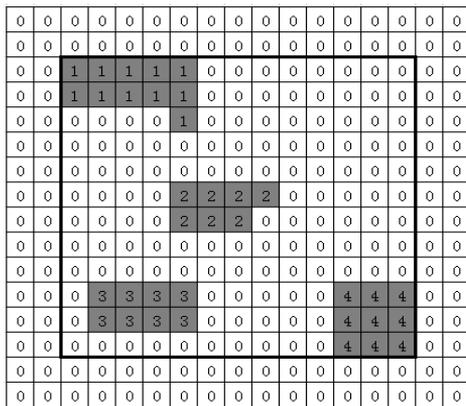


Figure 7. Mark skin region

3.4 Contour statistics

There are interference regions in the skin color segmentation, as shown in Figure 8. The purpose is to screen out the color contour we need.

This procedure first count the number of each mark value which we can know the size of the area. Based on experimental statistics, finger contour size probably accounted for regional location of 5% to 15%. so we can

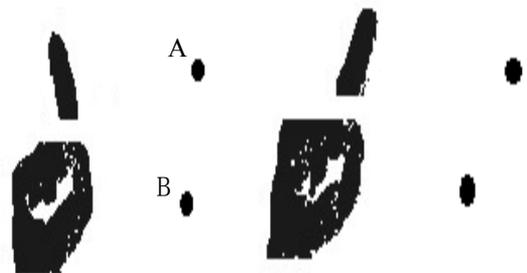
obtain skin contour region and exclude the too large or too small interference regions.



Figure 8. Interference regions

3.5 Position of center of gravity

We use the same method to obtain the center of gravity of finger region, and abstract the center of gravity, as shown in Figure 9. Start means motion start, End means motion end, A presents barycenter of finger, B presents barycenter of palm.



Motion start Motion end

Figure 9. The center of gravity of finger region

Bulid the barycentric coordinates triangle, as shown in Figure 10.

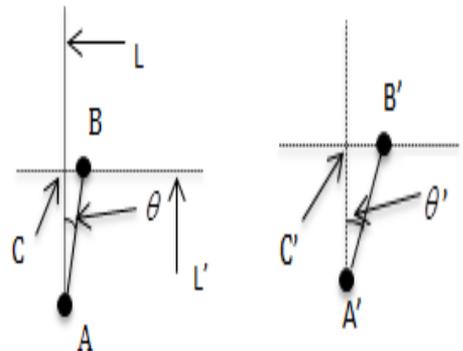


Figure 10. Barycentric coordinates triangle

(1) Calculate finger length

In the finger region, we assumed that the minimal point of the vertical coordinate is E, the maximum point of the vertical coordinate is F. E and F each do a horizontal line, Horizontal line spacing as d is shown in Figure 11. The fingers have two conditions: elongation and bending, the length of the bending is roughly 1/2 of the elongation. So we assumed the length of motion start as d1, length of motion end as d2, max for d1, d2 in the larger,  $m = m = |d1 - d2|$ .

If  $m > 0.4max$ , finger maybe do bending motion; if  $m < 0.4max$ , finger don't do bending motion. Based on this situation, we need analyze finger motion information from aspects of angle.

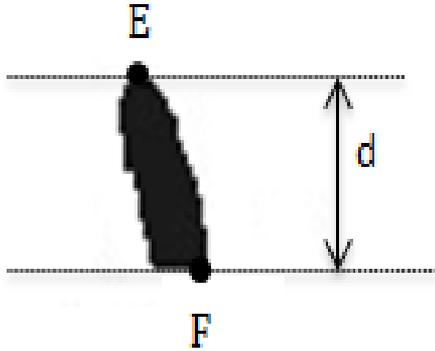


Figure 11. Image of finger length

(2) Computation angle of  $\theta$ ,  $\theta'$

Assuming that the A point coordinate is  $(x_1, y_1)$ , the B point coordinate is  $(x_2, y_2)$ , the C point coordinate is  $(x_1, y_2)$ , so

$$\theta = \arccos\left(\frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}\right)$$

In the same way, we can calculate the  $\theta'$ .

If  $|\theta - \theta'| < 10^\circ$ , it shows that the finger don't swing.

If  $|\theta - \theta'| > 10^\circ$ , and if  $\theta > \theta'$ , it shows that the

finger do swing motion to the right, if  $\theta < \theta'$ , it shows that the finger do swing motion to the left.

Now, we have got 4 movements of the finger: bending, holding, left, right. We can analyze the gesture with the movements of finger and the coordinate of barycenter.

#### 4. CONCLUSIONS

The importance of human-computer interaction is more prominent, and the diversity of interaction is more important. This paper is based on this starting point to research a real-

time and efficient micro gesture recognition algorithm based on Web client, and the language of the web client has the characteristics of cross platform which can run in most circumstances and solve the limitations of the platform. With the development of the mobile devices, gesture recognition will make an important contribution to human interaction.

#### ACKNOWLEDGMENTS

This work was supported in part by the Project of the Science and Technology Department of Zhejiang Province under Grant No.2014C31081 and No.2014C31068 and part by the Project of the Open University of China.

#### REFERENCES

1. Lingchen Chen; Feng Wang; Hui Deng; Kaifan Ji, A Survey on Hand Gesture Recognition, Computer Sciences and Applications (CSA), 2013 International Conference on Computer Sciences and Applications, pp. 14-15, Dec. 2013.
2. J. Yang, Y. Xu and C. S. Chen, Gesture Interface: Modeling and Learning, IEEE International Conference on Robotics and Automation, Vol.2, pp. 1747-1752, 1994.
3. P. Kakumanu, S. Makrogiannis and N. Bourbakis, A Survey of Skin-Color Modeling And Detection Methods, Pattern Recognition, vol. 40, pp. 1106-1122, 2007. DOI: [10.1016/j.patcog.2006.06.010](https://doi.org/10.1016/j.patcog.2006.06.010).
4. LEE J. S., LEE Y. J., LEE E. H., et al., Hand Region Extraction and Gesture Recognition from Video Stream with Complex Background through Entropy Analysis [C], San Francisco, CA, USA: Proceedings of the 26th Annual International Conference of the IEEE EMBS, 2004.
5. Panwar M., Hand Gesture Recognition Based on Shape Parameters, Computing, Communication and Applications (ICCCA), 2012 International Conference on Computing, pp. 1,6, 22-24, Feb. 2012. DOI: [10.1109/ICCCA.2012.6179213](https://doi.org/10.1109/ICCCA.2012.6179213).
6. Chenglong Yu, Xuan Wang; Hejiao Huang; Jianping Shen; Kun Wu, Vision-Based Hand Gesture Recognition Using Combinational Features, , 2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), pp. 543,546, 15-17, Oct. 2010. DOI: [10.1109/IIHMSP.2010.138](https://doi.org/10.1109/IIHMSP.2010.138).
7. Tingfang Zhang, Zhiquan Feng, Dynamic Gesture Recognition Based on Fusing Frame Images, Intelligent Systems Design and Engineering Applications, 2013 Fourth International Conference on Intelligent Systems Design and Engineering Applications, pp.280,283,6-7 Nov. 2013. DOI: [10.1109/ISDEA.2013.468](https://doi.org/10.1109/ISDEA.2013.468).
8. R. Lockton. A.W. Fitzgibbon, Real-Time Gesture Recognition Using Deterministic Boosting, Proceedings of British Machine Vision Conference, 2002. DOI: [10.5244/C.16.80](https://doi.org/10.5244/C.16.80).
9. N.D.Binh, E.Shuichi, T.Ejima, Real-Time Hand Tracking and Hand Recognition System, ICCI 2006, 5th IEEE International Conference, 2006.

10. J. Raheja, C. Ankit, S. Singal. Tracking of Fingertips and Centers of Palm Using KINECT, *2011 Third*

*International Conference on Computational Intelligence Modelling Simulation*, pp. 248-252, 2011.