

Research on Human-Computer Interaction Technology based on Visual User Gesture Recognition

Jianfeng Liao ^{1, 2, *}, Qun Zhang ² and Jianping You ²

¹ Department of Computer Science Wenhua College, Wuhan, China.

² College of Computer Science, Hubei University of Technology, Wuhan, China.

* jazz981012@163.com

Abstract. This paper deeply discusses the general situation, background and application of human-computer interaction, and proposes a new exploration-one-hand gesture recognition in human-computer interaction mode, and discusses the main research techniques in detail and comprehensively. The basic framework of vision-based gesture recognition system is studied. The various principles and methods of vision-based gesture positioning, gesture tracking, gesture segmentation and gesture recognition are analyzed. Based on the CamShift algorithm and the improved CamShift algorithm for gesture tracking, the CamShift algorithm cannot solve the large-area motion interference problem when solving complex dynamic changes. Therefore, it is proposed to add Kalman filter to estimate the next state. It proves that more effective gesture tracking is realized. This paper uses a more reasonable method to achieve the correct sense of input gestures through the visual channel by means of computer vision, digital image processing, pattern recognition and other theories and techniques. The response required to achieve natural human-computer interaction.

Keywords: Vision; User gesture recognition; Human-computer interaction; CamShift algorithm.

1. Introduction

Human-computer interaction is the study of people, computers, and the technologies that interact with each other. The development process of human-computer interaction technology is a process of adapting people from computer to computer to adapt to people. With the increasing demand for automation from human beings, traditional window, icon, mouse, and pointer (WIMP) interaction methods cannot fully meet people's needs due to defects in the naturalness and efficiency of operation, while traditional interaction methods only Support for single-user operation has been unable to meet the requirements of new multi-user systems.

The current method of gesture-based gesture interaction is to install two cameras at the two top corners of the display plane, take images with the camera, extract gestures through image processing, and calculate the coordinate position of the fingertips according to the triangle positioning and coordinate transformation. Finger input [1-3]. The shortcoming of this kind of interaction is that it can only support single-user input. In order to compensate for such defects, this paper proposes gesture tracking based on Kalman filter and CamShift algorithm. Kalman filter and CamShift algorithm are combined to realize gesture tracking, and the experimental conclusion is used to illustrate this paper. The combination of the two uses gesture tracking.

2. Vision-based Gesture Recognition System Overall Design and Experimental Preparation

2.1 System Design

The modeling of gestures is mainly to determine the position of the hand in the captured image through a certain algorithm, which can be quickly positioned, so that the subsequent work can be performed in time. There are currently algorithms that pass the action model, as well as Haar-like features and Adaboost-based face detection algorithms. These are very robust in certain environments. The analysis of gestures is mainly based on the determination of the gesture area, through the geometric algorithm to extract more specific parts, such as the finger, or the apex of the finger. This

allows relative coordinates to be identified in some systems, providing the basis for later interactions. Gesture recognition is to establish the corresponding relationship through the established mapping model on the premise that the gesture information can be obtained, so that the interaction task is more specific[4].

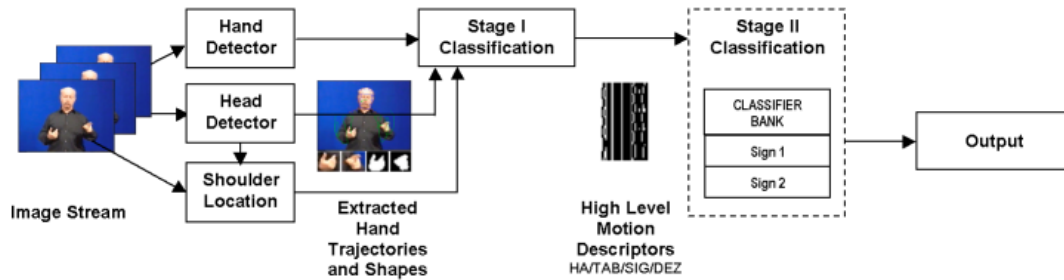


Fig. 1 Vision-based gesture recognition process

2.2 Gesture Modeling and Analysis

The first type is template matching technology, which is the simplest identification technology. It matches the feature parameters of the gesture to be recognized with the pre-stored template feature parameters, and performs the recognition task by measuring the similarity between the two. Template matching can be divided into two categories according to the choice of template: First, using a known target as a template, template matching is performed in an image to find an area close to the template, thereby identifying an object in the image. Such as points, lines, geometric figures, text and other objects; second, with an image as a template, compared with the image to be processed, to identify the existence and movement of the object[5]. The parameter that measures the difference between the template and the matching object is the sum of squared errors. Assume that the size of the template is $m \times n$ (width \times height) and the size of the image is Width \times Height (width \times height). The coordinates of a point in the template are (i, j) , and the gray value of the point is $U(i, j)$; the point coordinate in the image coincident with it is $(P-i, Q-j)$, and the gray level of the point is $V(P-i, Q-j)$. The result of a match is calculated by the following formula.

$$\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [U(i, j) - V(P-i, Q-j)]^2 \quad (1)$$

After all the images are matched, the smallest one is found. It can be seen that the amount of computation of the template matching is very large, and one-time subtraction, $m \times n$ square, and $m \times n - 1$ time addition are performed for one match, and the entire image is subjected to $(Width - m + 1) \times (Height - n + 1)$ time matching.

2.3 System Software Platform

This system uses C/C++ language to write interface and algorithm. The development platform uses Microsoft Visual C++ 6.0. In addition, since this system is a visual system, it uses OpenCV, a computer vision open source library developed by Intel Corporation. In this way, OpenCV is directly called. The library function can write a classical algorithm for visual processing, which greatly reduces the programming difficulty and improves the development efficiency. In the OpenCV library, the Cv Cam Shift Tracker class is used to implement the Cam Shift algorithm, making secondary development very simple.

2.4 Camera Video Reading

Thanks to OpenCV for system development, the camera video read-in part is also very easy, just call its specified function, create a display window, and properly install the camera driver. After

starting the program, the real-time video can be easily read from the camera and displayed in the window. The specific effect is shown in Figure 2.

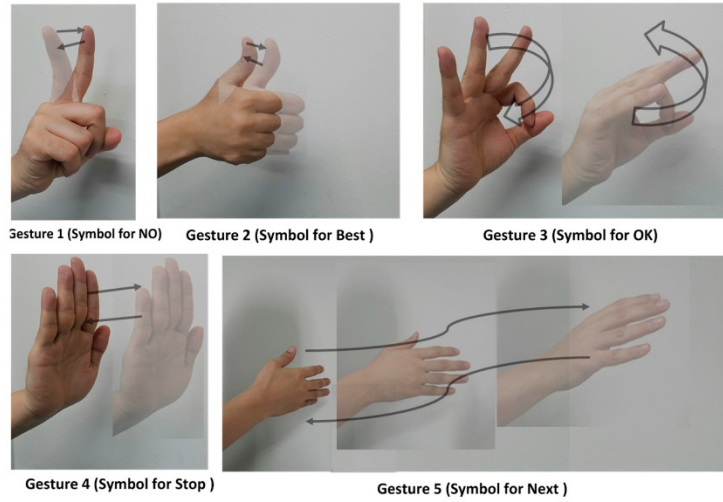


Fig. 2 Camera video read in an action screenshot

3. Gesture Tracking based on Camshift Algorithm

3.1 Camshift Basic Algorithm

The CamShift algorithm is a nonparametric method for density function gradient estimation of dynamically varying distributions. For discrete probability distributions, the algorithm process is as follows[6]:

- (1) Select a search window of size s in the color probability distribution map.
- (2) Calculate the zero moment:

$$Z_{00} = \sum_x \sum_y I(x, y) \quad (2)$$

Calculate the first moment of x and y :

$$Z_{10} = \sum_x \sum_y xI(x, y) \quad Z_{01} = \sum_x \sum_y yI(x, y) \quad (3)$$

Where $I(x, y)$ is the pixel value of coordinate (x, y) , and the range of variation of x and y is the range of the search window.

- (3) Calculate the centroid of the search window as (x_c, y_c) :

$$x_c = Z_{10} / Z_{00} \quad y_c = Z_{01} / Z_{00} \quad (4)$$

Considering the symmetry, S takes an odd number close to the calculation result.

By calculating the second moment, the long axis, short axis and direction angle of the tracked target can be obtained. The second moment is

$$Z_{20} = \sum_x \sum_y x^2 I(x, y) \quad Z_{02} = \sum_x \sum_y y^2 I(x, y) \quad Z_{11} = \sum_x \sum_y xy I(x, y) \quad (5)$$

The direction angle of the target long axis is

$$\theta = 1/2 \arctan \left[\frac{2(Z_{11}/Z_{00} - x_c y_c)}{(Z_{20}/Z_{00} - x_c^2) - (Z_{02}/Z_{00} - y_c^2)} \right] \quad (6)$$

When tracking the target of a specific color with the CamShift algorithm, it is not necessary to calculate the color probability distribution of all the pixels of each frame of image, and only need to calculate a larger area than the current search window, wherein the color probability distribution of the pixel points can be reduced. A lot of calculations. When the video sequence changes frame by frame, continuous tracking is formed.

3.2 Experimental Tracking Results

In the experiment, the improved CamShift algorithm is used to track the video sequence read by the camera. The specific experimental steps are as follows:

(1) Find a sports area. For the two-frame line difference and binarization at the beginning of the video sequence, the motion region H is found by the above clustering algorithm. The value in the action probability lookup table is appropriately reduced, and the size and center of the initial search window are calculated in the corresponding H field in the second frame image.

(2) Set the action probability calculation area. An area having a size of 1.5 times the search window width b and height h centered on the center of the search window is set to calculate the motion probability distribution area.

(3) Set the calculation area of the CamShift search window. The area where the iterative calculation is set is the area centered on the center of the search window and slightly larger than the search window[7].

(4) Calculate the size and center of the search window. The distribution centroid is calculated with the centroid as the center of the search window, and the search window size is a function of the zeroth moment calculated in (3) until convergence (centroid change is less than 2). Save the size and center of the search window and calculate the width and height of the area of interest according to the formula.

(5) Processing the next frame of image. Set the size and center of the search window to the size and center of the search window stored in (4) and return to (2).

Through the above improved CamShift algorithm, the motion sequence of the video sequence read by the camera is tracked, and the inverse probability projection image of the action is obtained as shown in FIG. 3, wherein the white area is the action probability area.

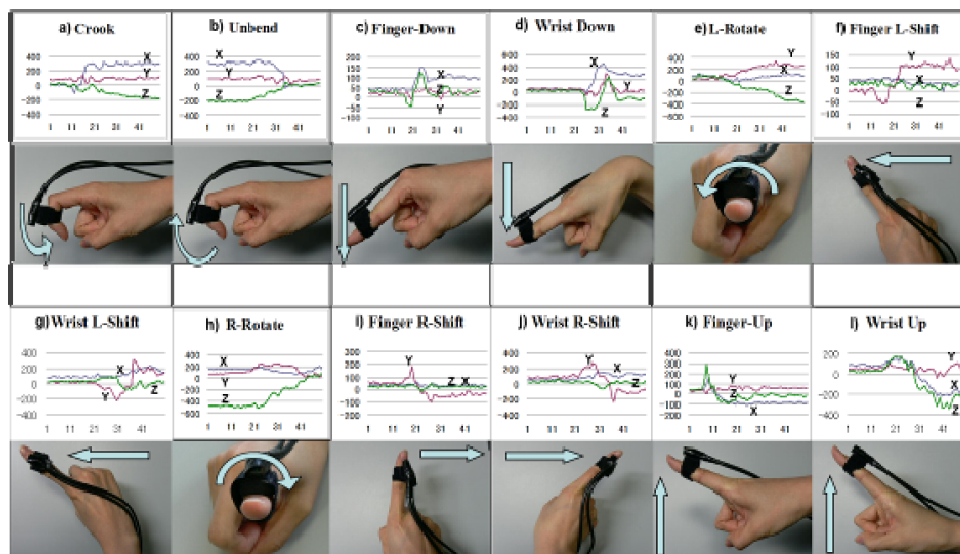


Fig. 3 Finger gestures and the corresponding output wave of acceleration

Fig. 3 gesture probability projection

4. Gesture Recognition based on Geometric Moment and Edge Detection

The system realizes simple human-computer interaction through gesture recognition. Firstly, several gestures are read by the camera to perform image preprocessing, including gesture detection using motion detection and motion detection, and gesture segmentation based on adaptive thresholding. Then the geometric moment and edge detection based on the proposed method are adopted. The combined gesture recognition method: extracting four components of the seven feature components of the geometric moment feature by the pre-processed gesture image to form a geometric moment feature vector of the gesture. The edge of the gesture image is directly detected on the basis of the grayscale image, and the boundary direction feature of the image is represented by the histogram. Finally, the distance between the images is calculated by setting the weights of the two features, and then the gesture is recognized.

Moment is a very important feature in image analysis and image recognition. The geometric moment is the simplest and most important moment in the moment function. Geometric moments are a common technique used to derive translational, telescopic, and rotational invariants. The $(p+q)$ moment of the two-dimensional image $g(x, y)$ is defined as follows.

$$m_{pq} = \sum_y \sum_x y^p x^q g(x, y) \quad (7)$$

The geometric moment function with invariance in image plane transformation is applied in the fields of target discrimination and pattern recognition. The shape feature of the image can obtain a set of geometric moment feature invariants through a series of calculations, which can be used to identify possible differences. A certain type of image of the same characteristics in size and in different directions.

After the binarization preprocessing of the gesture image, the geometric moment feature of the gesture image is extracted, and four of the seven feature components of the geometric moment feature are extracted to form a geometric moment feature vector of the gesture. The edge of the image is directly detected on the basis of the grayscale image, and the boundary direction feature of the image is represented by the histogram. The distance between the images is calculated by setting the weights of the two features, and the 30-letter gesture is recognized with a recognition rate of 90%.

5. Conclusion

In this paper, we introduce some methods for the steps of gesture recognition, and focus on the methods and experimental results used in this paper. The gesture positioning adopts the combination of motion information and motion information, which not only ensures the high efficiency and real-time performance of the calculation, but also solves the problem of large-area motion interference. The performance is better than the commonly used single positioning method; the gesture tracking uses the improved Camshift algorithm method. While ensuring real-time calculation, it better solves the occlusion problem of gesture tracking; the gesture segmentation adopts adaptive enthalpy method, which is efficient and effective, and suitable for real-time processing; gesture recognition adopts geometric moment feature and edge detection, and gesture recognition Very effective, simple to calculate, and guaranteed real-time. Combining these aspects, the method used in this paper performs gesture interaction, which is simple to calculate, guarantees real-time performance, and is superior to common methods.

Acknowledgments

This work was financially supported by Hubei Natural Science Foundation No. 2013CFC113.

References

- [1]. Ren Yaxiang. Research on Human-Computer Interaction Development Based on Gesture Recognition. *Computer Engineering and Design*, Vol.7 (2006) No.27, p.1201-1204.
- [2]. Xiao Zhiyong, Qin Huabiao. Human-Computer Interaction Based on Sight Tracking and Gesture Recognition. *Computer Engineering*, Vol.15 (2009) No.35, p.198-200.
- [3]. LI Wensheng, XIE Mei, DENG Chunjian. Human-computer interaction technology framework based on multi-point gesture recognition. *Computer Engineering and Design*, Vol.6 (2011) No.32, p.2129-2133.
- [4]. Gu Lizhong. Research on gesture recognition and human-computer interaction based on appearance. *Shanghai Jiaotong University*, Vol.4 (2008) No.28, p. 103-105.
- [5]. Du Yujun. Research on gesture recognition and human-computer interaction system based on somatosensory sensor. *Wuhan University of Science and Technology*, Vol.3(2013) No.5, p.7-15.
- [6]. Pan Jiansheng. On the Application of Gesture Recognition in Human-Computer Interaction. *Computer Knowledge and Technology*, Vol.7(2011) No.35, p.9216-9218.
- [7]. Liu Jun, TianGongLi, LiRongkuan, et al. Human-computer interaction based on gesture recognition in intelligent space. *Journal of Beijing Union University*, Vol.2 (2010) No.24, p.14-17.