Research of Hand Positioning and Gesture Recognition Based on Binocular Vision

Tong-de Tan，Zhi-min Guo

Information and Engineering School

Zhengzhou University

Zhengzhou, China

ABSTRACT

This paper proposes a new method of extracting feature points of hand. The method uses the center of mass of the hand as the match point to calculate the location information of the target based on Mathematical model of binocular visual positioning. The convex hull points of hand contour obtained by image segmentation can be used to identify the different gestures. Furthermore, a system with both functions of locating the three-dimensional position of hand and identifying the appropriate gestures is designed, which can serve as the interface to drive virtual hand to complete manipulation of grasping, moving and releasing virtual objects.

KEYWORDS: Hand Positioning, Hand Gesture Recognition, 3D Registration, Virtual Hand Grasping

1 INTRODUCTION

In virtual reality environment, most people interact with virtual objects using the two-dimensional mouse or three-dimensional mouse. In order to make the operation more realistic and natural, sometimes they need the help of data gloves or clothing to realize the interaction between man and virtual objects; Although the equipment of data gloves or data clothing can obtain more precise location information, which price is generally more expensive and is cumbersome to install, which application range is limited to specialized research and is difficult to promote. At the same time, optical positioning equipment, with its easy nstallation,elatively inexpensive, non-contact and high precision measurement, is increasingly subject to attention.

Human hands have not only contained a large number of interactive information with human cognitive habits but also the three-dimensional space position information from the continuous movement of hands, so we can combine the location and gestures information as a whole. The hand positioning and gesture recognition system is built up based on the binocular vision principle, which can real-time calculate the hand position and recognize gesture information through two fixed cameras. The system can be used as a natural, efficient interface driven with immersive virtual hand in training, virtual assembly, virtual surgery, robotic arm control and so on.

2 SYSTEM OVERVIEW

The whole system is divided into six parts: the operator, the image acquisition module, hand positioning module, 3D registration module, gesture recognition module and the virtual operating module. The six parts interact with each other to form a closed loop, shown in Figure 1. After the operation action is issued by the Real-world operator, the image acquisition module captures hand moving image via the left and right fixed cameras. In the hand positioning module， which images caught on left and right cameras have been input to, the three-dimensional position of center of hand mass, referred to as the feature points, relative to the left (right) camera is calculated based on principles of binocular vision. In the 3D registration module, the coordinates of center of hand mass in the camera coordinate are mapped to the coordinates of center of virtual hand mass in virtual environment. Gesture recognition module firstly selects gesture images caught on a camera, then analyze the images and identify the appropriate gestures lastly. In the virtual operating module, according to the virtual hand position and posture, we can complete grasping, moving virtual objects and other operations by operating rules in the virtual environment. According to received feedback an operator adjustments the posture and position of hand and drive the virtual hand to implement the next operation.

3 HAND POSITIONING MODULE

In the positioning module, the system, by means of high-speed industrial camera, makes hand as the target detection object and detects it from the CCD video sequence, then calculates the three-dimensional coordinates of the target position which is relative to the right camera optical center.

3.1 REAL-TIME TARGET DETECTION AND TRACKING

In order to locate the position of the hand, first, the system should detect feature points of the hand for matching in left-right cameras respectively, then, locate the hand position based on binocular stereo-visual model. Because the human hand is a complex multi-link body, which size and shape changes frequently in the process of motion, and the system requires the ability to locate target real-time, it is very necessary to find a algorithm of real-time tracking target in image according to target size and shape change. The author uses current mainstream stracking algorithm-CamShift to find the target quickly and accurately in each frame, and improves it to track targets automatically.

3.1.1 CamShift Algorithm.

CamShift algorithm is a motion tracking algorithm based on color, which inherits the MeanShift algorithm and extends it to continuous image sequence. The basic algorithm is that all frames of video images are processed by the MeanShift algorithm, and the results(the search window center and size) of the previous frame is considered as the initial value of the search window of the next frame for the MeanShift algorithm, and so iteration continues, can achieve the target tracking. The algorithm procedure is as follows:

e-mail: ttde@zzu.edu.cn
e-mail: guo_zhimin@126.com

IEEE International Symposium on Virtual Reality Innovation 2011
19-20 March, Singapore
978-1-4577-0054-5/11/$26.00 ©2011 IEEE

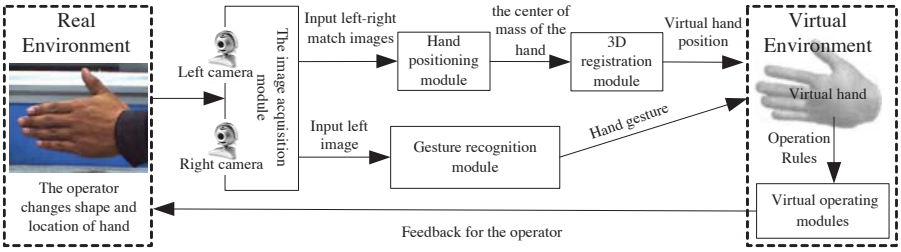

Figure 1: System block diagram

(1) Convert one frame image to HSV format and extract the value of H channel;
(2) Initiate search window size and position;
(3) Calculate the color histogram in the search window, normalize it to obtain the graph of probability distribution, and use it as a lookup table, we can get the graph of probability projection by means of replacing H-channel image pixel value of each point with its corresponding probability value;
(4) Get the new location of the search window using the MeanShift algorithm;
(5) In the next frame, use the value of position and size from the previous frame to initiate search window;
(6) Repeat (3), (4), (5) until the end of the image sequence;

3.1.2 Improved CamShift Algorithm

The CamShift algorithm, having the advantages of fast convergence and unaffected by noises, can track targets real-time. However, the algorithm is a semi-automatic tracking algorithm and in order to track the target the search window must be pre-set according to the target location and size of images. This system requires that the target of images captured by the camera is detected automatically. If the location and size of the target are treated as the initial parameters for search window, we can call the CamShift algorithm iterative to track the target. In this way, the semi-automatic tracking is converted to the automatic tracking.

Firstly, a small rectangular area (80 * 60) from hand images is selected as a pre-saved template image; Secondly, the color histogram of the template image is obtained by calculating the template image and is regarded as a match features; Thirdly, according to the size of the template image (80*60), each frame image (640*480) is divided into several small blocks of image, then the histogram information for each small block of image is calculated. Finally, The particular block of image is found out, which color histogram is most similar to the color histogram of the template image through comparing the color histogram of each image with the template image's, and its position is regarded as the search window for the CamShift algorithm. So, the target can be detected and tracked automatically, as shown in Figure 2.

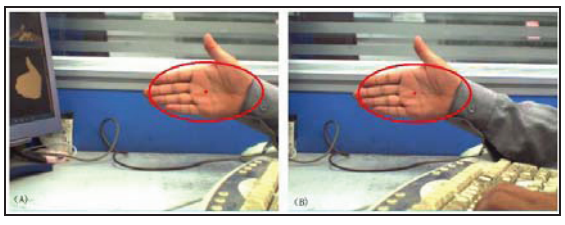

Figure 2:Tracking the center of hand mass by CamShift algorithm.

3.1.3 Extraction Feature Points of Hand

In order to achieve binocular visual positioning, stereo matching process is needed and the coordinates of the target point in camera image must be gotten. The system will regard the center points of hand mass in the left and right images as feature points to match. The execution speed of the algorithm is much quicker than the speed of traditional stereo matching algorithms (such as SAD, SSD, NCC, etc.).The system can meet the real-time requirements. The center of search window, which is achieved by the CamShift Algorithm, is considered as the center of hand mass. The coordinates is derived using the following formula:

Zero-moment:

$$M_{00} = \sum_x \sum_y I_c(x, y) \qquad (1)$$

First-moment of x and y:

$$M_{01} = \sum_x \sum_y y I_c(x, y)$$
$$M_{10} = \sum_x \sum_y x I_c(x, y) \qquad (2)$$

The center point of search window is（xc,yc）:

$$x_c = M_{10} / M_{00}, \; y_c = M_{01} / M_{00} \qquad (3)$$

Ic(x,y) is the pixel value of the coordinates(x,y), the range of x,y is limited in the size of the search window.

3.2 Implementation of Target Position

Binocular vision is to get target depth information from the two images. According to the geometric relationship of binocular imaging, you can easily get target location formula:

$$\begin{cases} X = \dfrac{b(u_1 - u_0)}{u_2 - u_1} \\ Y = \dfrac{b(v_1 - v_0)}{u_2 - u_1} \\ Z = \dfrac{bF}{u_2 - u_1} \end{cases} \qquad (4)$$

The u0 and v0 is respectively the number of rows and columns coordinates of intersection points between the optical axis and the left camera plane, u1, u2 are target point abscissa respectively for left and right camera plane coordinate, v1 is target point ordinate for left camera plane coordinate. The length of Z is the distance from the space point to the plan e of the two cameras, the length of baseline b is the distance between the two optical centers of camera and f is the focal length of camera. u0, v0, the baseline b and the focal length F can be obtained through camera calibration. The coordinate of target points in the right camera coordinate system can be calculated real-time with equation (3). (Note: The system assumes the world coordinate system and the right camera coordinate system are coincident, the calculated three-dimensional coordinates are relative to optical center of the right camera.)

4 3D REGISTRATION MODULE

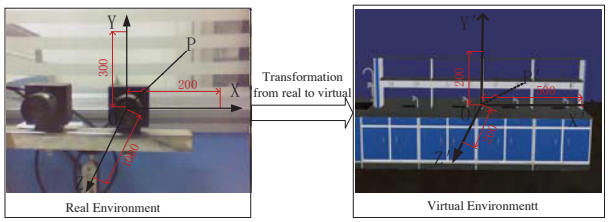

Figure 3: Coordinate system of real environment conversion to coordinate system of virtual environment

The position data of the center of hand mass is used to drive the virtual hand to move correspondingly in virtual environment. As the camera focal length and the experimental constraints, hand movement should be limited in an appropriate three-dimensional space; While the movement of the virtual hand should also be limited in scope; a mapping in space is exist between the real and the virtual. The coordinates of hand in real environment are converted to the coordinates in virtual environment by 3D registration. As shown in Figure 3, the hand of real environment moves in XYZ coordinate system of the camera. The range of motion (red arrow mark, in millimeters) in the X-axis is(-200,200), Y-axis is(0,300), Z-axis is(0,1000), the origin of coordinates is the right camera optical center O , the point P (Xc, Yc, Zc) is the center point of hand mass. A chemical desk was constructed in the virtual environment for experiment, the virtual hand move in X'Y'Z' coordinate system, the range of motion (red arrow mark, in millimeters) in the X' axis is (-500,500), Y' axis is (0,200), Z' axis is (0,200), the origin of virtual coordinates is the center of the surface desk O ', where is the initial location of the virtual hand. Point P in real environment can be converted to P ' (Xv, Yv, Zv), the location of the virtual hand, by 3D registration. The relationship from the point p in real environment to the point P ' in environment can be expressed as:

$$\begin{pmatrix} Xv \\ Yv \\ Zv \\ 1 \end{pmatrix} = M \begin{pmatrix} Xc \\ Yc \\ Zc \\ 1 \end{pmatrix} = \begin{pmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} Xc \\ Yc \\ Zc \\ 1 \end{pmatrix}$$

The M is transformation matrix of the coordinate, Sx, Sy, Sz are the scaling factor for the M matrix, the value of which are the ratio of three-component lengths from hand-movement in virtual environment to hand-movement in real environment. This experiment, Sx = 500/200, Sy = 200/300, Sz = 200/100; Depending on different applications, motion space of the virtual hand will be different, so 3D registration is needed in order to determine the M matrix before the system is used.

5 GESTURE RECOGNITION MODULE

Gesture target in this system is the real hand and the objective here is to recognize various gestures without any mark on the hand. The main function of the module is processing the image data acquired by image acquisition module and extracting hand contour feature. Therefore, it recognizes different gestures by comparing it with predefined gesture data in database.

5.1 EXTRACTION OF HAND CONTOUR FEATURE

This paper uses color based on recognition technology for extraction of hand contour feature. Neither any hand marker nor special background is needed in the experiment. First we convert RGB color space to HSV color space; then establish a complexion segmentation model, extract complexion information and contour; at last we specify hand contour according to the position of center of mass and the area of contour.

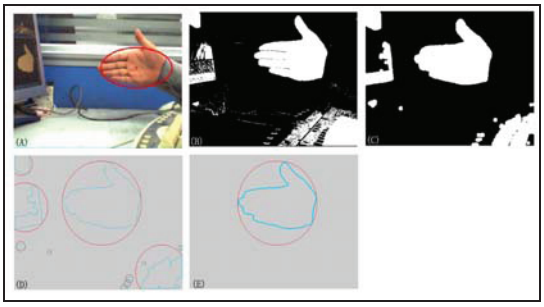

Figure 4: The process of extracting hand contour.

Camera acquired raw image in RGB color space, which contains not only the information of color red, green and blue, but also the information of brightness, extraction of complexion would be easily interfered by variation of external light in such a color space, whereas in HSV color space(H represents chrominance, S represents saturation, V represents brightness), component H and component S are determined by material and reflection characteristics of the object itself, which would not be easily interfered by external environment. So it's more reliable to extract complexion information according to component H and component S in HSV color space through color space converting. Establishing complexion segmentation model, value of component H varies from 0 to 30, while the value of S component is between 75 and 200. The complexion image extracted is illustrated in Figure 4(B). During converting process, although we restricted ranges of component H and component S, many discontinuous regions and small noise points are still produced due to the existence of many similar information of complexion in the image acquired by camera. The way to deal with this is making use of filter technology in image processing field, after experimental analysis and comparison with results of various filters, we choose opening and closing operation in mathematical morphology to filter unwanted background noise points, and finally obtain a satisfactory binary image, as shown in Figure 4(C).

There are still many small regions that similar to complexion in the filtered binary image. We can use position of center of mass and area of contour to find and extract hand contour. Using edge detection algorithm like Canny to find boundary pixels by detecting the difference between binary pixels, and constructing different contour from those pixels, as shown in Figure 4(D). In order to find hand contour in all detected contours, we use position and area of contours to determine hand contour. In public vision area of binocular camera, we can find the range of maximum and minimum value (roughly between 16376 and 100000) of various hand contour areas after experiment. The pixel coordinates of the center of mass of hand have been obtained during the stage of feature point extraction, for now we can look through areas in the range to see whether the pixel coordinates of the center of mass of hand is inside of hand contour, if so, it means that this contour is the one we are looking for, otherwise, continue to check the next contour. By doing so, we can finally find the hand contour, as shown in Figure 4(E).

5.2 GESTURE RECOGNITION

As Figure 5(A) shown below, we have summarized 12 different gestures which are commonly used in people's daily life by observing and analyzing various motions of hand. Through analyzing these 12 gestures, the contour feature data of each

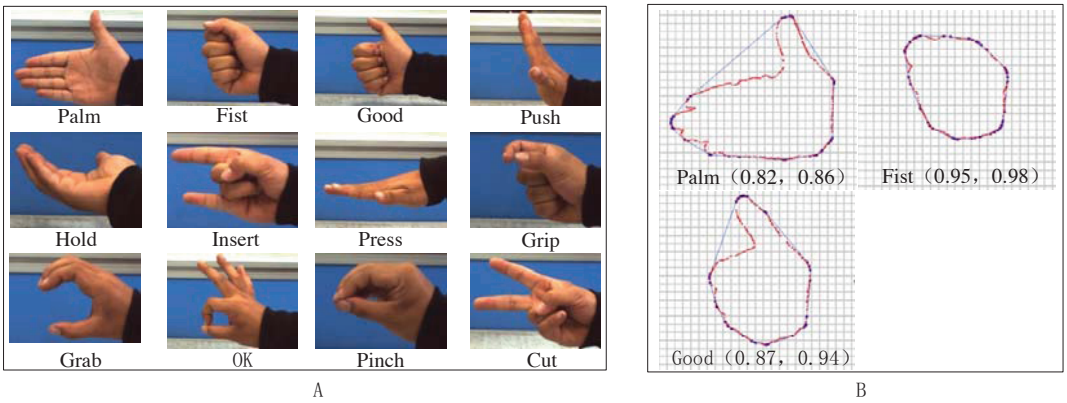

Figure 5: 12 gestures (A) and convex defect of hand contour (B).

gesture could be obtained and stored in database as samples for gesture recognition. For convenience of expression, here we use 3 typical gestures (palm form, fist form and "good" form) as examples to explain the establishment of database and the principle concerning recognition of such 3 gestures.

The 3 gestures have following meaning:

Palm form – right before grasping virtual object.

Fist form – virtual object is grasped and can be moved.

"good" form – grasping completed.

An useful method to understand gesture contour is computing its convex hull, then calculating its convex defect, gesture feature can be well demonstrated with the convex defect, As indicated in Figure 5(B), areas enclosed by red lines are hand contours, blue dots laid on red lines construct convex hull points of hand contours, blue line areas is contour of convex hull points. The method of recognition is calculating the ratio between hand contour area (enclosed by red lines) and convex hull points area (enclosed by blue lines), thus recognizing gestures by differentiating ratios of different gestures.

Hand contour ratio = hand contour area / convex hull points contour area

The value interval of each gesture is obtained (noted in brackets in Figure 5(B)) by analyzing large amount of experimental data, which can be stored in database as criterion for gesture recognition. What is known from qualitative analysis is that the wider the "gap" between red line and blue line, the larger the area of convex defect, so that the hand contour ratio is decreasing while the increasing of the "gap" area. As a result, the ratio of palm form is quite small; for the case of fist form, the ratio is relatively large due to the approximation of its two areas; and the ratio of "good" form is between two others.

6 Simulation

6.1 Virtual Hand Modeling

First we establish 3 3D hand models that correspond to 3 different gestures in gesture recognition module respectively. The virtual hand in virtual environment is driven to adjust position and form according to exported virtual hand position and gesture information during running of system.

6.2 Operation Rules

In order to make sure grasping of object can be correctly implemented by virtual hand whilst also making grasping motions authentic and natural, some grasping rules must be set. Under such rules, the object is considered to be grasped If both virtual hand and virtual object meet the conditions; at this time the object moves to specified location with virtual hand; if releasing fingers, virtual hand and virtual object are no longer meet the conditions, so then freeing relations of dependence and releasing object.

Rule 1: Recognizing gesture and calculating distance of centers of mass between virtual hand and virtual object. If distance value is smaller than the threshold set before, this is considered as grasping operation and object will be activated.

Rule 2: After activation of object, gesture changes from palm to fist and locks on the object, now the object is attached to the hand and will change its position with the movement of the hand.

Rule 3: After activation or lock of object, if gesture changes from fist to "OK", the operation is considered to be completed and object will be released.

6.3 Examples

We apply the system interface designed in this paper to virtual chemistry experiment platform, this implement grasping of graduated cylinder and horizontal movement. The process is: image acquisition module captures hand image of operator in front of camera, then the image data is sent to hand location module and gesture recognition module for processing. Location of center of mass of hand is exported from location module and then is converted to three-dimensional coordinates of virtual hand in virtual environment after passing augmented reality registration module; gesture recognition module exports current gesture information as palm. Augmented reality registration module judges whether current operation satisfy operation rules or not according to position information of virtual hand and gesture information, once it does, then activating the graduated cylinder in virtual environment and attaching it to the virtual hand. For the moment the operator's gesture changes to fist and locks on graduated cylinder, operator keeps fist form and makes movement in front of camera, this drives the virtual hand in virtual environment to make the same movement, as displayed in Figure 6.

7 Conclusion

This paper designs and implements real time gesture recognition and location system according to the principle of binocular vision, combines spatial location and gesture recognition and proposes a new method of human-computer interaction, which can be used as driver Interfaces of virtual hand and applied to virtual chemistry experiment platform. This system has reached a satisfactory result in real time, operational stability and robustness. Hand recognition and location is a very complicated project, this paper has done only a preliminary exploration, there are many unresolved problems, such as: experimental environment is sensitive to light, accuracy of recognition and location is not high, dual hand operation cannot be recognized, no three-dimensional

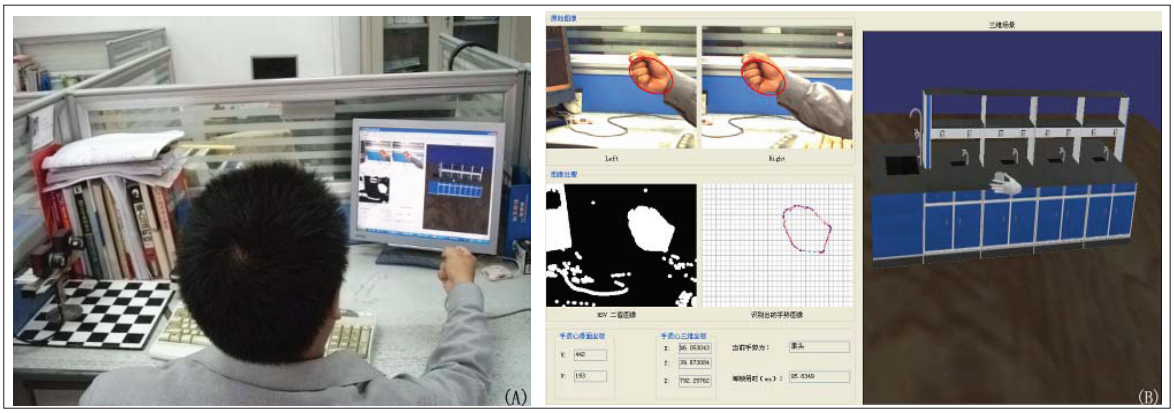

Figure 6: Fist gesture (A) drive virtual hand in virtual environment to do the moving graduated cylinder operation (B)

reconstruction of hand data model and so on. However, we believe that with the continuous development of computer vision field, the computer understanding of gesture in the future will definitely come to a new height.

REFERENCES

[1] Bradski GR. Computer Vision Face Tracking For Use in a Perceptual User Interface[M].Intel technology Journal, 1998.

[2] Hongxia Chu, Shujiang Ye, Qingchang Guo. Object Tracking Algorithm Based on Camshift Algorithm Combinating with Difference in Frame[J].International Conference on Automation and Logistics, 2007.

[3] D.Comaniciu and P.Meer. Mean shift analysis and applications. IEEE International Conference on Computer Vision(vol.2,p.1197),1999.

[4] Gray Bradski, Adrian Kaehler. Learning OpenCV[M].Published by O'Reilly Media, 2008.