Journal of Network Intelligence ©2020 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **5**, Number **4**, November **2020**

Design of Real-time Face Position Tracking and Gesture Recognition System based on Image Segmentation Algorithm

Teh-Lu Liao[1], Han-Chang Chen[1], Jun-Juh Yan[2,*]

[1]Department of Engineering Science,
National Cheng Kung University, Tainan 701, Taiwan

[2]Department of Electronic Engineering,
National Chin-Yi University of Technology, Taichung 41107, Taiwan

[*]Email: jjyan@ncut.edu.tw (corresponding author)

Received November 2018; revised November 2019

Abstract. *Human vision is one of the most advanced senses; therefore, image for the human?s sense is very important. Along with the rapid improvement in the development of computer technology and execution speed, image processing techniques have also matured. However, in the past, positioning cameras have been used nearly exclusively for detecting and tracking moving objects. If the moving objects move outside the lens? view area, the object cannot be tracked. In order to improve this weakness and reduce blind spots, this paper proposes a real-time object tracking gesture recognition system based on image segmentation algorithm.*
Keywords: Object tracking, Object detecting, Image processing, Gesture recognition

---

1. **Introduction.** For many years, the mouse and keyboard have been the bridge between people and computers. Nowadays, a new approach that uses human gestures has been developed [1-3]. Actions or touch commands can allow the computer to distinguish instructions, such as Wii games and smartphones. This gesture operation has changed the interaction between humans and the machines, allowing users to do virtual activities similar to the real actions without the need for remote controls or hand-held tools. Thus machine vision has become a focus of attention [4]. Among the various types of machine vision, moving object detection and object tracking have been considered interesting research topics. In fact, researches in this area have an extremely wide range of applications, such as video conferencing and distance learning. Combining face tracking [5] and gesture recognition [6] techniques can maintain a speaker?s face in the screen?s center, saving the trouble of manual operation of the camera. Through a simple gesture, a variety of other gestures can be defined. As to the current computer technology development, the personal computer already has lots of advantages, such as a high processing speed, large storage space, powerful hardware and software platforms. It has gradually become a primary tool for processing images, sound and so on. Since each type of multimedia information processing and transmission has become digitalized, the digital multimedia data has the merits of portability and durability, which adds convenience for use. As to the hardware selection, among them, the background subtraction method has better detection results due to less computational requirements, and has been widely used in a variety of visual systems. However, simple subtraction is easily affected by variations of

light source, causing miscalculations of moving objects. In addition, it cannot achieve the dynamic detection effect, so it is not applicable to the real-time human position tracking system.

In view of this, moving object detection in the present study will firstly detect the skin color [7] [8], and then in order to reduce miscalculations caused by light source changes, the skin color detection image must be processed by dilation and erosion to remove small noises. The operation results are processed by Sobel mask image marginalization [9]. Finally, coordinates of the whitest spot numbers using the histogram statistics calculate the center of the moving objects, and track the object. Consequently, in this study the target color analysis is critical, and also through the use of white balance correction, color image processing can be made to understand how human?s visual perception identifies colors. As for gesture recognition, the use of vector magnitudes and angles are used as the basis for judgment.

2. Material and methods.

2.1. System architecture. The system architecture and the main flow chart are shown in Figures 1 and 2, respectively. A web camera is connected to the computer through a USB interface, and transmits the scene to the GUI interface. Through calculation and the Bluetooth interface, the center position of the target is transferred to the Arduino Mega 2560 development board, and the control circuit receiving the coordinates converts to the motor moving direction. Through rotation of the servo motor, the visual angle of the camera can be changed so that the object is maintained in the center of the picture.

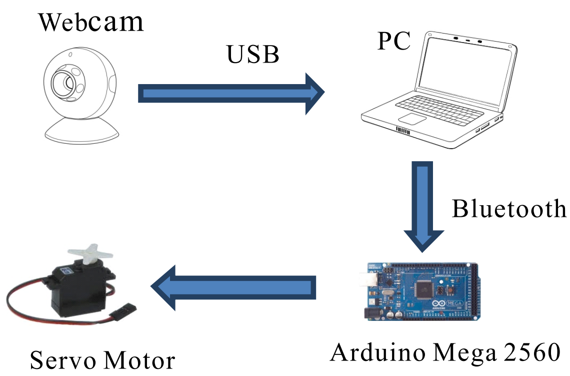

FIGURE 1. System architecture

2.2. Part 1: Face detection. Since face detection makes it easier and more direct to find the initial position of the face and palms, this study is based on detecting the location of the face. The purpose of locating the left and right sides of the face is to check whether there is a certain range of face size to reduce the range of image processing [10-14]; consequently, the speed of system processing is also accelerated. The face detection procedure is given in the flowchart shown in Figure 3.

- **A. Skin color histogram** As shown in Figure 4, the face region can be very clearly seen after the noise reduction. In order to obtain the face position, two histogram statistics of the horizontal and vertical directions are conducted. Figure 5 refers to the horizontal direction histogram, while Figure 6 represents the vertical direction histogram. The algorithm can calculate the total pixel value of the whole image color region, and then obtain the average value of the horizontal and vertical images.

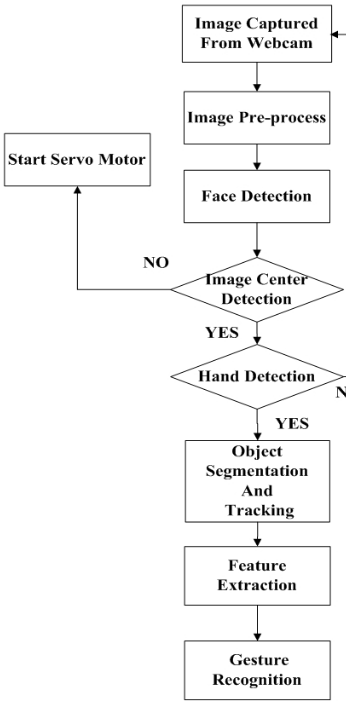

FIGURE 2. Main flow chart

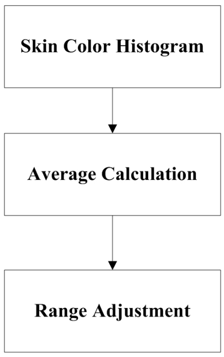

FIGURE 3. Flow chart of face detection

FIGURE 4. After noise reduction

B. Average calculation The histogram statistics shown above indicate that the middle face area in the histogram statistics is more concentrated, and forms a wave crest. Therefore, this paper uses this feature to calculate the total pixel value of the whole image face region, as in formulas (1) and (2). Then, the average values of the

Figure 5. Horizontal direction histograms

Figure 6. Vertical direction histograms

horizontal and vertical images can be calculated, as in formulas (3) and (4). The average value shows that if some left values of the pixels in the horizontal direction are less than the average value and the right values are greater than the average value, then this value is set as coordinate vector X_1. If some left values of the pixels in a horizontal direction are greater than the average value, the right pixel values are less than the average value; this is set as coordinate vector X_2. By applying the same method, the coordinates vectors of Y_1 and Y_2 can also be obtained. The descriptions mentioned above can be seen through the discriminate formulas (5) and (6) to realize the results as shown in Figure 7.

$$\mathbf{Row}_{sum}[y] = \sum_{x=0}^{width-1} P(x,y), y = 0, ..., height - 1 \tag{1}$$

$$\mathbf{Col}_{sum}[x] = \sum_{y=0}^{height-1} P(x,y), x = 0, ..., width - 1 \tag{2}$$

$$\mathbf{Col}P_{Ave} = \frac{1}{width} \sum_{x=0}^{width} \sum_{y=0}^{height} P(x,y) \tag{3}$$

$$\mathbf{Row}P_{Ave} = \frac{1}{height} \sum_{x=0}^{width} \sum_{y=0}^{height} P(x,y) \tag{4}$$

$$\begin{cases} X_1 = \{x \,|if\, (\mathbf{Col}_{sum}\,[x-1] < \mathbf{Col}P_{Ave}) \cap (\mathbf{Col}_{sum}\,[x+1] > \mathbf{Col}P_{Ave})\}\,, \\ X_2 = \{x \,|if\, (\mathbf{Col}_{sum}\,[x-1] > \mathbf{Col}P_{Ave}) \cap (\mathbf{Col}_{sum}\,[x+1] < \mathbf{Col}P_{Ave})\}\,. \end{cases} \tag{5}$$

$$\begin{cases} Y_1 = \{y \,|if\, (\mathbf{Row}_{sum}\,[y-1] < \mathbf{Row}P_{Ave}) \cap (\mathbf{Row}_{sum}\,[y+1] > \mathbf{Row}P_{Ave})\}\,, \\ Y_2 = \{y \,|if\, (\mathbf{Row}_{sum}\,[y-1] > \mathbf{Row}P_{Ave}) \cap (\mathbf{Row}_{sum}\,[y+1] < \mathbf{Row}P_{Ave})\}\,. \end{cases} \tag{6}$$

where $P(x, y)$ denotes the pixel value at (x, y) coordinate. The vectors X_1, X_2, Y_1 and Y_2 are used to determine the face region. The Figure 7 shown above indicates the face position, but the range extraction is too big and a little offset. Thus, the next step is to display the exact face range through adjustment.

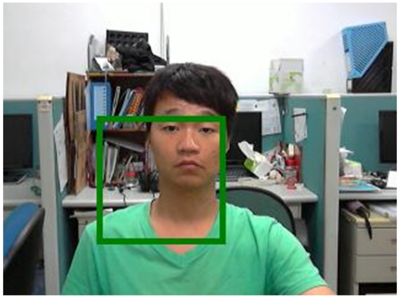

FIGURE 7. Result image

C. Range adjustment The horizontal and vertical directions of the face region can be obtained from the previous step. First, find the maximum pixel using formulas (7) and (8) of the horizontal direction within the range, then, based on the left and right sides of the maximum pixel, search for the more appropriate coordinate within the face region. Repeat this procedure for the vertical direction. The aforementioned description via the discriminate formulas (9) and (10) can be seen below.

$$\mathrm{Col}P_{\max} = \max\{X_1, X_2\} \tag{7}$$

$$\mathrm{Row}P_{\max} = \max\{Y_1, Y_2\} \tag{8}$$

$$\begin{cases} X_1' = \{x \,|if\, (\mathbf{Col}_{sum}\,[x-1] \leq Col_{max}) \cap (\mathbf{Col}_{sum}\,[x+1] \geq Col_{max})\}\,, \\ X_2' = \{x \,|if\, (\mathbf{Col}_{sum}\,[x-1] \geq Col_{max}) \cap (\mathbf{Col}_{sum}\,[x+1] \leq Col_{max})\}\,. \end{cases} \tag{9}$$

$$\begin{cases} Y_1' = \{y \,|if\, (\mathbf{Row}_{sum}\,[y-1] \leq Row_{max}) \cap (\mathbf{Row}_{sum}\,[y+1] \geq Row_{max})\}\,, \\ Y_2' = \{y \,|if\, (\mathbf{Row}_{sum}\,[y-1] \geq Row_{max}) \cap (\mathbf{Row}_{sum}\,[y+1] \leq RowP_{max})\}\,. \end{cases} \tag{10}$$

where X_1', X_2', Y_1' and Y_2' denote the coordinate vector after quantification. $\mathrm{Col}P_{max}$ and $\mathrm{Row}P_{max}$ are defined as the returned coordinate data from image. After the range adjustment, the results are shown in Figure 8. It can find that the new face range is more precise than that in Figure 7.

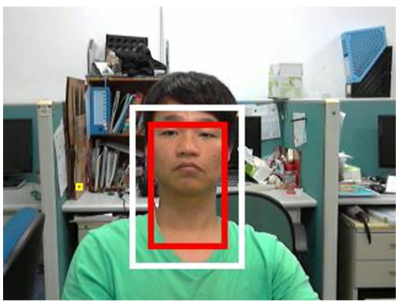

FIGURE 8. After adjustment

2.3. **Part 2: Object segmentation and tracking.** In this topic, the Sobel operation is first applied within the searching range to obtain the edges of hand/palm; the number of the palm edge is greater than that of the arm edge. Thus, the palm can be quickly identified for subsequent judgments. To realize target segmentation and tracking, the flowchart in Figure 9 is used.

A. Edge histogram The calculation steps mentioned above can determine the positions of the hands; this paper conducts edge detection on the entire image. The aim of using edge detection is to clearly distinguish the palms and arms since there are more edges associated with the palms than with the arms Therefore, we use this feature to check the image after marginalization, as shown in Figure 10.

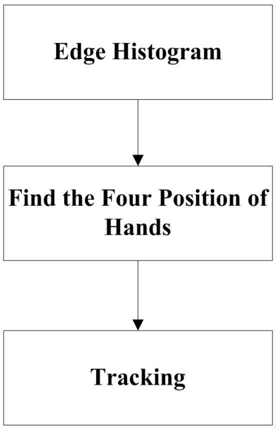

FIGURE 9. Flow chart of object segmentation and tracking

FIGURE 10. Result of Sobel edge detection

The image marginalization histogram statistics is shown in Figure 11, which refers to the histogram statistics in the horizontal direction, while Figure 12 and Figure 13 represent the histogram statistics of left and right hand in the vertical direction, respectively.

B. Find the four positions of hands The Figure 14 shows that the maximum peak value, regardless of it being the horizontal or vertical histogram; further, this paper uses the formulas (11)-(18) to determine the maximum peak value of pixels in the horizontal direction.

$$\mathrm{Col}_{sum1}[x] = \sum_{y=S_{Y1}}^{S_{Y2}} P(x, y), x = S_{X1}, ..., S_{X2} \tag{11}$$

FIGURE 11. Histogram of Sobel edge detection

FIGURE 12. Statistical results of marginalization for the left hand range

FIGURE 13. Statistical results of marginalization for the right hand range

$$\mathrm{Col}_{sum2}\left[x\right] = \sum_{y=S_{Y3}}^{S_{Y4}} P(x,y), x = S_{X3}, ..., S_{X4} \tag{12}$$

$$\mathbf{Max}X_1' = \mathbf{max}\left\{\mathbf{Col}_{sum1}\left[S_{X1}\right], ..., \mathbf{Col}_{sum1}\left[S_{X2}\right]\right\} \tag{13}$$

$$\mathbf{Max}X_2' = \mathbf{max}\left\{\mathbf{Col}_{sum2}\left[S_{X3}\right], ..., \mathbf{Row}_{sum2}\left[S_{X4}\right]\right\} \tag{14}$$

The position of the maximum peak value is firstly obtained, and then adding and deducting a constant to the position of the maximum peak value can provide the range of both hands, respectively, for example, NewX_1 and NewX_2 in Figure 14 are horizontal coordinates for the left hand and NewX_3 and NewX_4 are the vertical coordinates for the right hand.

$$\mathrm{Row}_{sum1}\left[y\right] = \sum_{x=S_{x1}}^{S_{x2}} P(x,y), y = S_{Y1}, ..., S_{Y2} \tag{15}$$

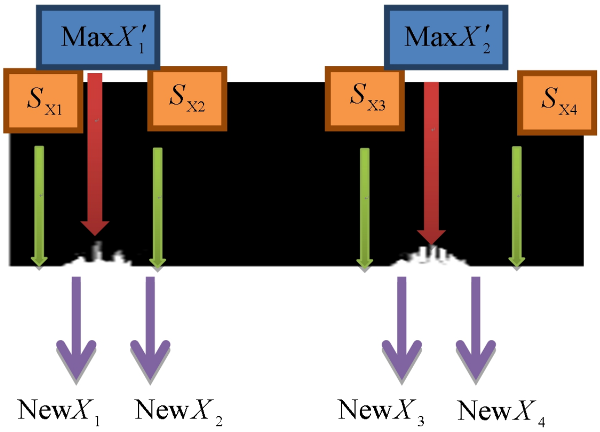

FIGURE 14. Histogram of the horizontal direction

$$\text{Row}_{sum2}[y] = \sum_{x=S_{x1}}^{S_{Y1}} P(x,y), y = S_{Y3}, ..., S_{Y4} \tag{16}$$

$$\mathbf{MaxY}'_1 = \mathbf{max}\left\{\mathbf{Row}_{sum1}[S_{Y1}], ..., \mathbf{Row}_{sum1}[S_{Y2}]\right\} \tag{17}$$

$$\mathbf{MaxY}'_2 = \mathbf{max}\left\{\mathbf{Row}_{sum2}[S_{Y3}], ..., \mathbf{Row}_{sum2}[S_{Y4}]\right\} \tag{18}$$

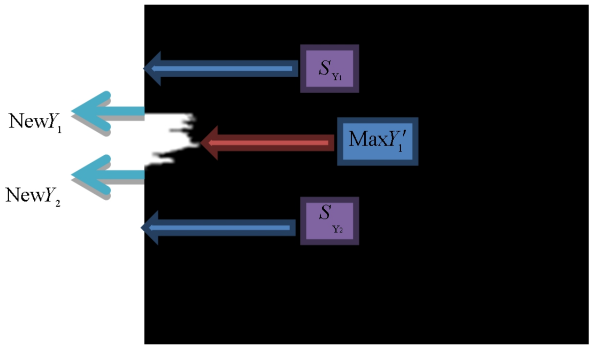

FIGURE 15. Left hand of the vertical direction histogram

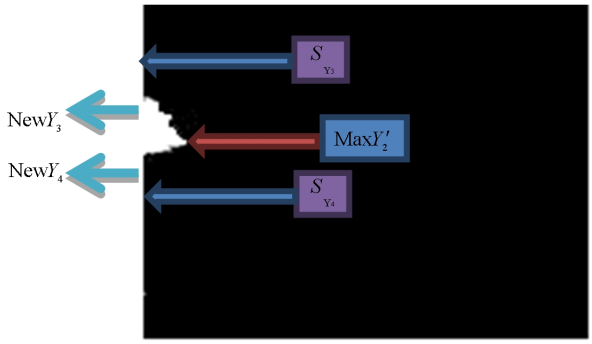

FIGURE 16. Right hand of the vertical direction histogram

From Figures 15 and 16, the range from the left and right hands can be identified in the vertical coordinates. The coordinates in the left hand are NewY1 and NewY2, while those for the right hand are NewY3 and NewY4. The more detailed description can be found in [16].

C. Tracking The purpose of this section is to split the palm and trace the palms exact location. After the discussion above, the left and right hand position in one image can be successfully segmented. In the following, tracking of the hands is elaborated.

The last step captures the left and right hands, i.e. a total of eight coordinates are needed in order to realize the tracking. Therefore, each of the eight coordinates is extended for a fixed proportion, which is the white frame shown in Figure 17. The white range, which is a search range, moves over time along with the change in the palm; it means that the tracking effect is realized.

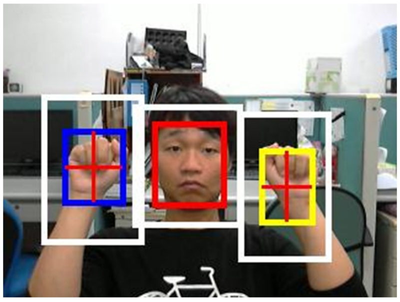

FIGURE 17. Result of object segmentation and tracking

2.4. **Part 3: Feature Extraction.** Since the tracking effect is achieved, this paper does not directly use identification and feature extraction can be also done. Because thus far only the extracted information can be identified, identification results might contain errors. To address this issue, the recognition rate must be improved. To solve this problem, in addition to the characteristics of the palm, we add additional conditions for detecting whether there are fingers or not. Therefore, this system captures the location of both hands, which can be explained by the flowchart in Figure 18.

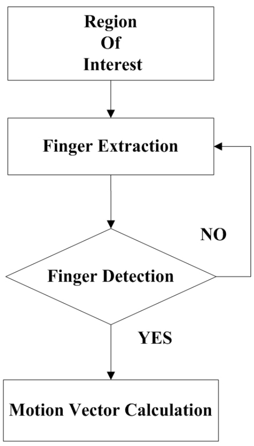

FIGURE 18. Flow chart of feature extraction

The hand range can be captured as mentioned in the last subsection, so searching in this range can determine whether there are any features about the hands. Results are shown in Figures 19 and 20. When the fingers show up, the histogram will have a corresponding peak value within the range when the fingers appear. By using this feature, the object detection method mentioned in the last subsection can determine and judge successfully whether the fingers show up or not, as shown in Figure 21. Finger positions will be framed out in red and with the coordinates of the fingers, identification can be done accordingly.

Figure 19. Fingers detection

Figure 20. Histogram of finger edge detection

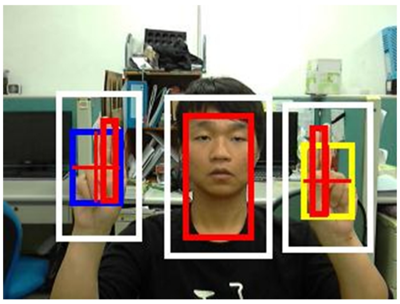

Figure 21. Result of feature extraction

2.5. **Part 4: Hands gesture recognition.** The move process at each time interval usually produces a series of moving sequential information. Before identification, a moving trajectory will transform the moving sequences of each time point [15]. This paper only considers two kinds of dynamic information: (1) the angle of displacement; and, (2) the displacement intensity.

First of all, by subtracting the fingertips of the image from the finger tips in the previous image, the gradient value from the points can be calculated. According to the gradient value, the angle of displacement and the displacement intensity can be calculated.

First, the identification method used in this paper makes judgments from a sequence of finger movements. The data of the moving distance will be neglected slightly, as the slight moving distance will not influence the recognition. Figure 22 is the quadrant diagram. If there is a sequence of finger movements, much angle information can be obtained to categorize the information. If most angles are in Region I there is a tendency that the fingers are moving toward Region I. The detailed calculation method can be found in [16].

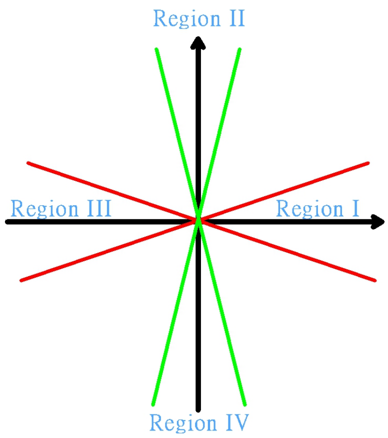

FIGURE 22. Quadrant diagram

3. **Experimental results and discussion.** The next hand gesture recognition test item is dynamic hand gestures, where actions such as translations, zooming in and out and rotations test whether the system can recognize the corresponding instructions or not, as shown from Figures 23 to 28.

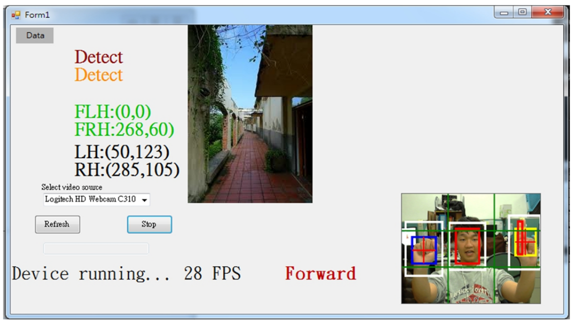

FIGURE 23. Translate (Forward)

After testing, a significant relationship between the dynamic hand-gestures recognition rate and the correctness of cutting out the hand shape was found. In this paper, if the hands are not correctly detected in 5 seconds, this testing will be identified as a failure. Gestures shown by users and correctness of the gesture standard set by the system also play a role in recognition results. A different noise tolerance set will have different results. If the fingers which protrude out slightly are judged as fingers, they can be correctly identified as the noise after the noise tolerance is lowered. In this experiment, the average recognition rate was 85%, in which the translate identification rate was 95%, the zoom identification rate was 7%, and the rotation identification rate was 90%.

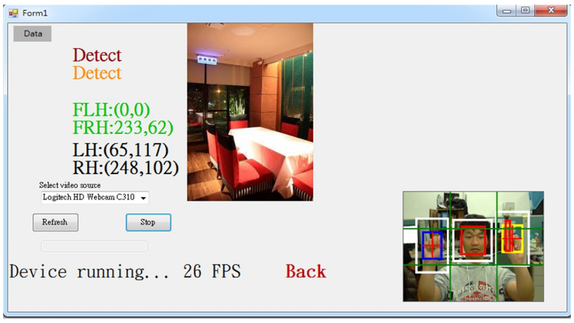

FIGURE 24. Translate (Back)

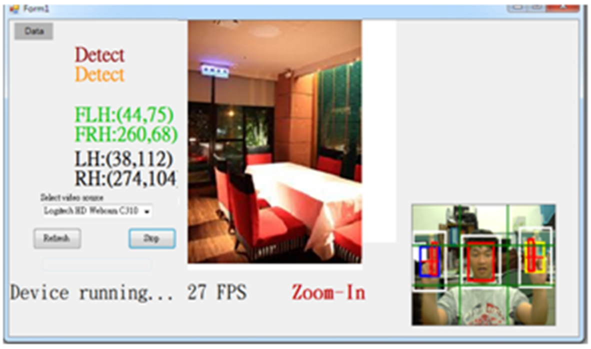

FIGURE 25. Zoom-in

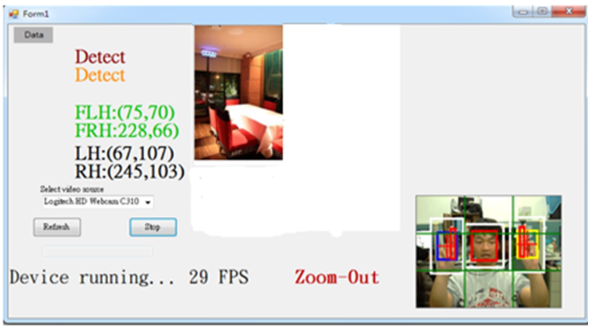

FIGURE 26. Zoom-Out

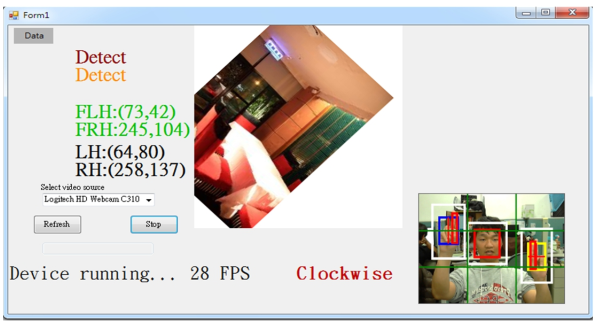

FIGURE 27. Rotate (Clockwise)

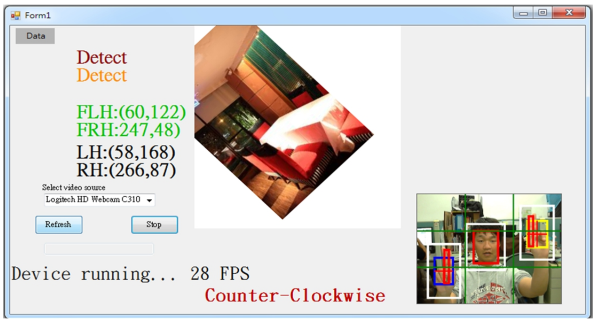

Figure 28. Rotate (Counter-Clockwise)

4. **Conclusion.** This paper links basic theoretical analysis, the system architecture design and the application. USB transmission acts as a communication platform to realize the moving target tracking system. The system design mainly includes the hardware setup and software algorithm design. For the hardware part, this study adopted Arduino Mega 2560 as the control unit, and the tracking platform was driven by a servo motor to follow the image motion. Although Arduino Mega 2560 itself provided the USB hardware interface, the associated firmware design must be integrated by the user according to different needs. The present study used a C language compiler firmware program to realize USB transmission operation, and after the image data was uploaded to the PC, it was sent back to the next stage to control the motor. The other part is the algorithm and graphical user interface (GUI) used in this study. It applies the Mean-shift and background subtraction method, and proposes the moving object detection algorithm to detect the location of the moving object. By counting the number of white spots, the palm center can be estimated; then, through histogram statistics, the finger features can be analyzed and tracked. In addition, via the discriminate quadrant conditions, gesture miscalculations can be improved by tracking only the palm. Finally, calculating the relative location of the object center point coordinates in the image can be used for judging mobile objects. During the implementation and testing process, the algorithm performance was validated using MATLAB, and related algorithms can be realized and a GUI user control interface developed with Visual C#2010. After experimental verification, this system can accurately detect, track and identify the palm and finger positions, and therefore achieves the system target of object tracking and recognition of a real-time image system.

REFERENCES

[1] Z.H. Chen, J. T. Kim, J. L. Liang, J. Z., Y. B. Yuan, Real-time hand gesture recognition using finger segmentation, *The Scientific World Journal*, vol. 2014, Article ID 267872, 9 pages, 2014.

[2] H. M. Ravindu T. Bandara, K. S. Priyanayana, A. G. Buddhika P. Jayasekara, D. P. Chandima, R. A. R. C. Gopura, An intelligent gesture classification model for domestic wheelchair navigation with gesture variance compensation, *Applied Bionics and Biomechanics*, vol. 2020, Article ID 9160528, 11 pages, 2020.

[3] H. M. R. T. Bandara, B. M. S. S. Basnayake, A. G. B. P. Jayasekara, and D. P. Chandima, Cognitive navigational command identification for mobile robots based on hand gestures and vocal navigation commands, *in Proceedings of the 2nd International Conference on Electrical Engineering(EECon)*, IEEE, 2018.

[4] Z. Li, P. P. Ling , G. A. Giacomelli, Machine vision monitoring of plant growth and motion. *science: international journal of earth space*, vol. 5, no.2, pp. 263-270, 1998.

[5] S. Thakur, S. Paul, A. Mondal, S. Das, A. Abraham, Face detection using skin tone segmentation, *2011 World Congress on Information and Communication Technologies*, Mumbai, pp. 53-60, 2011.

[6] M. Pantic , L. J. M. Rothkrantz, Automatic analysis of facial expressions: the state of the art, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424-1445, Dec. 2000.

[7] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, M. A. Abidi, Recent advances in visual and infrared face recognition-a review. *Computer vision and image understanding*, vol. 97, no. 1, pp.103-35, 2005.

[8] B. Zhang, L. Zhang, D. Zhang, L. Shen, Directional binary code with application to PolyU near-infrared face database. *Pattern recognition letters*, vol. 31, no. 14, pp. 2337-44, 2010.

[9] K. TSUKADA, M. YASUMURA, Ubi-finger: A simple gesture input device for mobile and ubiquitous environment. *Journal of Asian Information, Science and Life (AISL)*, vol. 2, pp. 111-120, 2004.

[10] A.B.M. Ramadan, A. M. El-Garhy, F. Z. Amer, M. Hefnawi, Forecasting gamma radiation levels using digital image processing. *Life Science Journal*, vol. 9, no. 1, pp.701-10.

[11] T. Hanafy, Design and validation of real time neuro fuzzy controller for stabilization of pendulum-cart system design and validation of real time neuro fuzzy controller for stabilization of pendulum-cart system. *Life Science Journal*, vol. 8, no. 1, pp. 52-60, 2011.

[12] M. Arif, M. Saqib, S. Basalamah, A. Naeem, Counting of moving people in the video using neural network system. *Life Science Journal*, vol. 9, no. 3, pp. 1384-92, 2012.

[13] H. Yousefi, M. Hirvonen, H. Handroos, A. Soleymani, Application of neural network in suppressing mechanical vibration of a permanent magnet linear motor. *Control Engineering Practice*, vol.16, no. 7, pp. 787-97, 2008.

[14] Y. Gao, C. Deng, G. Jiang, Application of BP neural network in the digital recognition system, *2010 2nd International Conference on Computer Engineering and Technology*, Chengdu, pp. V6-522-V6-526, 2010.

[15] J. L. Raheja, R. Shyam, U. Kumar, B. P. Prasad, Real-time robotic hand control using hand gestures. *2010 Second Intl. Conf. on Machine Learning and Computing (ICMLC)*, 2010; pp.12-16.

[16] H. C. Chen, Real-time human position tracking and gesture Recognition System Based on Image Segmentation Algorithm and Its Application to Image Browser, *Cheng Kung University Master thesis.* 2013.