



Sun, Y., Liang, X., Fan, H., Imran, M. and Heidari, H. (2019) Visual Hand Tracking on Depth Image Using 2-D Matched Filter. In: 4th International Conference on UK - China Emerging Technologies (UCET 2019), Glasgow, UK, 21-22 Aug 2019, ISBN 9781728127972.

There may be differences between this version and the published version. You are advised to consult the publisher's version if you wish to cite from it.

<http://eprints.gla.ac.uk/191228/>

Deposited on: 29 July 2019

Enlighten – Research publications by members of the University of Glasgow
<http://eprints.gla.ac.uk>

Visual Hand Tracking on Depth Image using 2-D Matched Filter

Yongdian Sun¹, Xiangpeng Liang¹, Hua Fan², Muhammad Imran¹ and Hadi Heidari¹

¹James Watt School of Engineering, University of Glasgow, G12 8QQ, UK

²School of Microelectronics, University of Electronic Science and Technology of China, Chengdu, China
Hadi.Heidari@glasgow.ac.uk

Abstract—Hand detection has been the central attention of human-machine interaction in recent researches. In order to track hand accurately, traditional methods mostly involve using machine learning and other available libraries, which requires a lot of computational resource on data collection and processing. This paper presents a method of hand detection and tracking using depth image which can be conveniently and manageably applied in practice without the huge data analysis. This method is based on the two-dimensional matched filter in image processing to precisely locate the hand position through several underlying codes, cooperated with a Delta robot. Compared with other approaches, this method is comprehensible and time-saving, especially for single specific gesture detection and tracking. Additionally, it is friendly-programmed and can be used on variable platforms such as MATLAB and Python. The experiments show that this method can do fast hand tracking and improve accuracy by selecting the proper hand template and can be directly used in the applications of human-machine interaction. In order to evaluate the performance of gesture tracking, a recorded video on depth image model is used to test theoretical design, and a delta parallel robot is used to follow the moving hand by the proposed algorithm, which demonstrates the feasibility in practice.

Keywords— *Hand Detection and Tracking; Matched Filter; Depth Image; Human-machine interaction; Delta robot.*

I. INTRODUCTION

With the rapid development in computer science and sensing technology, human-machine interaction has been widely explored for various applications in the past decades [1]. As the fundamental of a gesture control application, a great diversity of hand detection methods had been proposed along with the evolution in computer science. KINECT ONE, the second-generation product of Kinect, consists of one RGB color camera, one depth sensor (IR camera and IR projector) and a microphone array which provides three-dimensional image data for the applications such as gesture-controlled video games and human-machine interaction [2]. The schematic of KINECT ONE is shown in Fig. 1.

In the past few years, various methods have been created and applied for hand detection and tracking using Kinect [3]. The most common and convenient approach is to use standard Windows SDK package from Microsoft, which can realize multiple functions on Kinect included hand detection [4]. Besides, some other frequently-used methods for Hand detection with Kinect include machine learning and deep learning on depth or color image. Additionally, based on various advanced image processing technologies, the Kinect control libraries can be directly used for hand detection and tracking under controlled by corresponding drivers like PrimeSense's open NI series drivers [5].

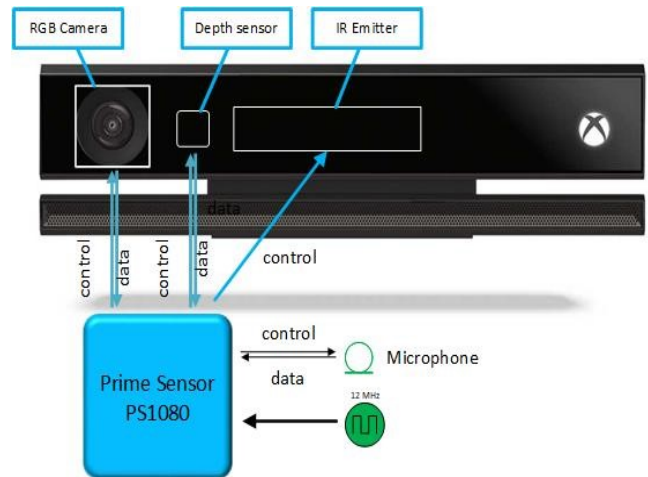


Figure 1. KINECT architecture: RGB camera can provide color image and Depth sensor provides a depth image.

Meanwhile, for object detection in image processing, a common idea is that finding significant points in the image. Likewise, human face detection is looking for special features like eyes, nose, and mouth in an image. However, in contrast to the human face, hands don't have those distinguishing features for detection. One solution for that is to mark special reflective points on hand to separate it from the environment [6-8]. This paper addressed this problem by choosing hand shape as a special feature for detection using 2-D matched filter.

Generally, compared with the above-mentioned methods, this paper presents an efficient solution based on the two-dimensional matched filter without huge data collection and training. The system needs to find the object among the pre-processed image using convolution with a hand template. There are five steps to achieve hand detection through this method. The flow chart is shown in Fig. 2. In the end, hand position can be marked on the image after a series of comparison.

In practical application of gesture control for a delta robot, additional steps are needed to correct the hand tracking results. Considered about the design of delta robot, the extra coordinate transformation is to adjust the hand position onto the coordinate of the moving platform of the delta robot, which can protect the delta robot in case the hand position goes beyond the activity space of the robot arm[9].

The details of this method will be introduced in this paper as followed: Section II presents the procedure of image pre-processing to remove noise as much as possible. Theoretical template and filter design and delta robot control design are

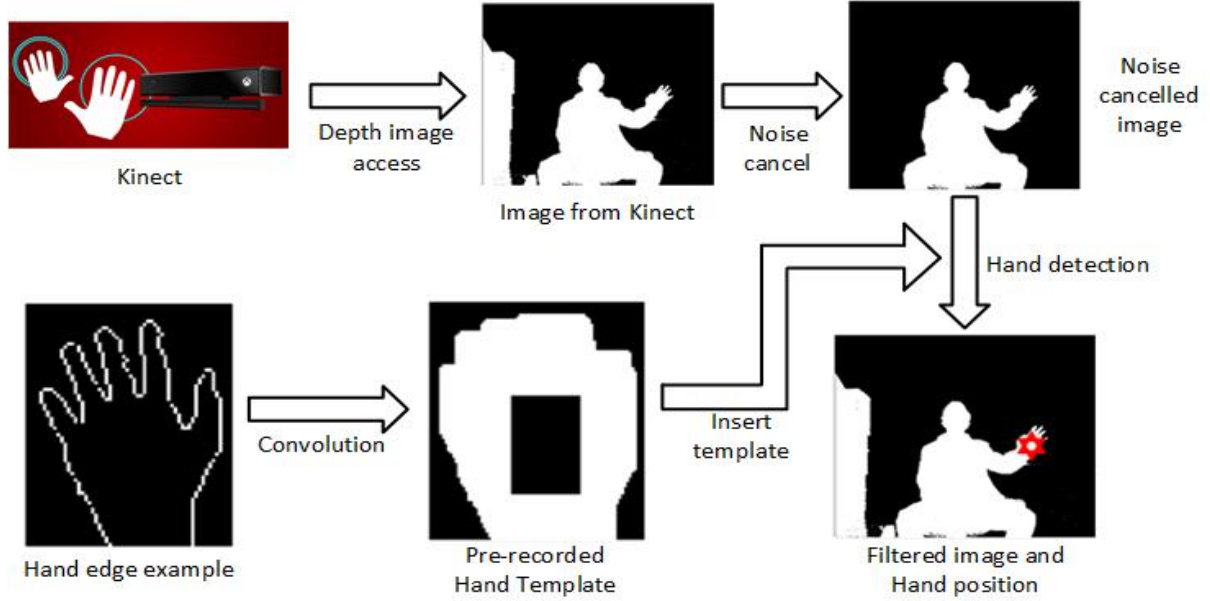


Figure 2. Hand Detection Flow Chart. Image is output from Kinect and input in 2-D matched filter with a hand template. The hand position is finally marked on the Image as the red star.

shown in section III. Finally, experiment results including a demonstration of real-time delta robot control and conclusion will be discussed in section IV and section V respectively.

II. IMAGE PRE-PROCESSING

A. Depth image obtain

In this method, Kinect control and data processing use different program platforms LabVIEW and MATLAB. Two program platforms combine together for hand detection and tracking. LabVIEW can be programmed as an image received platform from KINECT since it can directly acquire the image and hand position at the same time without other requirements. Later on, depth image data is processed with MATLAB script in the LabVIEW [10].

B. Noise canceling

Noise canceling is to reduce the interference signal as much as possible before gesture tracking for higher accuracy [11, 12]. Ideally, all the experimental test should be carried out in a wide-open area to reduce the background noise. However, due to the limit of the experiment environment, noise influence is much larger than expected. As shown in Fig. 2, the image from Kinect has been pre-processed. Nevertheless, the image still contains the user's body and other dopants that are close to the user like a cabinet or other objects in the background. Therefore, noise contains two parts, background noise, and noise around the human body. The former one is canceled before image output from Kinect, while the latter one is canceled after image output.

By setting the detection range of the camera, the Kinect can quickly eliminate most irrelevant objects in the background. The second canceller is to distinguish the user and the dopants close to him. In this process, the 2-D binary image connected pixel labeled technology is applied. As shown in Fig. 3, gray cells are assigned pixels. The same number will be labeled on the all 4-connected assigned pixels that is greater than 0. According to the experimental setup, the largest labeled area is the researcher's body, and other labeled areas are noise which needs to be fully removed.

Edge detection is used for removing dopants. With labeled pixels, edge detection can accurately contour the body based on Eq. (1)

$$Edge(s, t) = \sum_a \sum_b Body(a, b) Operator(m, n) \quad (1)$$

where $Edge(s, t)$ denotes the detected edge coordinates of the body whose dimension is $s \times t$, and $Body(a, b)$ is the matrix of data on the depth image. $Operator(m, n)$ is an $m \times n$ matrix used as edge detection coefficient. In this experiment, the edge detection operator is a Sobel operator shown in Eq. (2)

$$Sobel\ Operator = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (2)$$

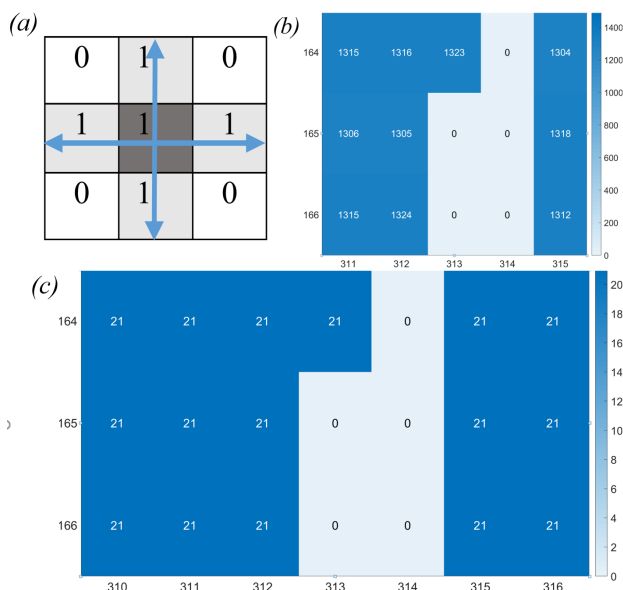


Figure 3. (a) Labeled 2-D 4-Connected pixels. (b) Segment of body pixels in image. (c) Segment of labeled body pixels in image.



Figure 4. Result after 2-D matched filter. The hand is located at the marked point.

Afterward, noise canceled image is the original image multiplied by body edge using Eq. (3).

$$\begin{aligned} \text{NoiseCanceledImage}(x, y) \\ = \text{Image}(x, y) * \text{Edge}(s, t) \end{aligned} \quad (3)$$

where x, y is the dimension of the noise cancel image and image, and (s, t) is the edge coordinate given in Eq.(1). In this equation, the matrix sizes of noise canceled image and image are the same to keep the edge points on the same coordinates.

III. DESIGN OF TEMPLATE AND FILTER

A. Template design:

For a matched filter, the template or known as filter coefficients is the most important element which can influence the accuracy of the result. In order to get an accurate hand position, a clear and complete open hand template is strongly suggested [13].

As shown in Fig. 2, the template is an open hand edge example which had been widened edge by another convolution with a unit matrix. The purpose of this operation is that magnifying the edge to adapt to the various patterns of gesture.

In treating process, original hand edge example need to be convoluted with a proper matrix of ones to increase the width of the hand edge according to the two-dimension discrete convolution as Eq. (4).

$$\begin{aligned} \text{Template}(i + c - 1, j + d - 1) \\ = \sum_x \sum_y \text{HandEdge}(c, d) \text{Ones}(i, j) \end{aligned} \quad (4)$$

As shown in Eq.4, formulas $i+x-1$ and $j+y-1$ are used to adjust the template matrix size where x, y and i, j represent the matrix dimensions of hand edge image and matrix of ones.

At the same time, in the template, the zero (black) area on hand palm can also affect the detection accuracy, small hand palm increases the wrong detection rate when the hand is around the human face. Obviously, in this case, the template is more like a circular template rather than a human hand template.

B. Filter design

As shown in Fig. 2, the template and noise canceled image are the two inputs of the hand detector algorithm. Since the human hand is always ahead of the body, the hand depth is smaller than body value. In order to outpoint the hand, the depth image needs to do a value transfer to make the hand's

$$\begin{aligned} \text{Pixel}(e + x - 1, f + y - 1) = \\ \sum_e \sum_f \text{Template}(e, f) \text{NoiseCanceledImage}(x, y) \end{aligned} \quad (5)$$

value larger than the body. Without using any existing filter function, the convolution is the basic operation of this filter [14]. Since the depth image and the hand template are 2-D arrays and discrete distribution as one pixel in storage, convolution needs to be satisfied with these two conditions. Equation. (5) shows the process of 2-D discrete convolution.

where template and image are two discrete 2-D variables and pixel yields the convolution result of each pixel. x is the column number, and y is row number of image data.

After convolution, a set of pixel data will be generated from the filter as shown in Fig. 4, where the color density means the numerical value of each pixel. The hand locates at the largest numerical value pixel which is the darkest point on the image since hand should be the best-matched point of the template after Eq. (5). The final pixel array is larger than the original image. Therefore, the hand position needs to be transferred to get the real position in the original image.

IV. EXPERIMENTAL RESULTS

Experiments were carried out in term of the accuracy of hand detection on depth image data and real-time system using KINECT to do gesture tracking of a delta robot.

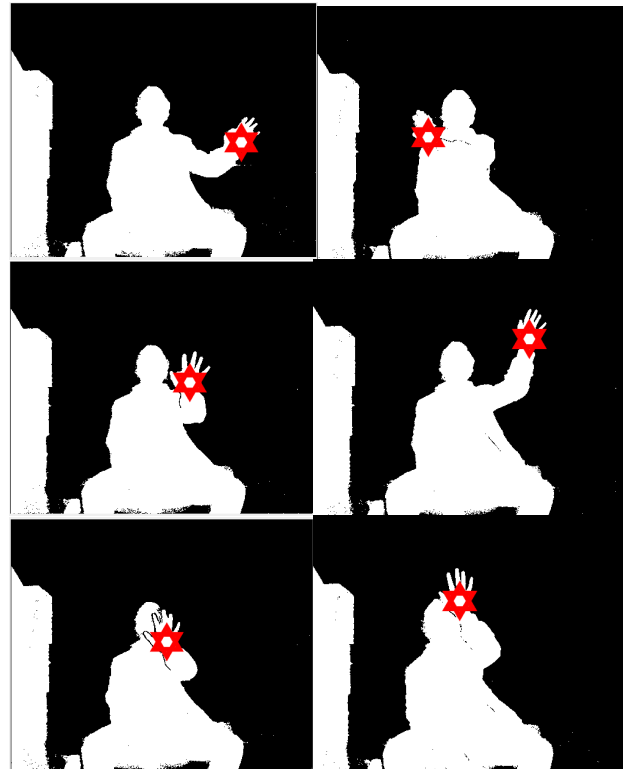


Figure 5. Hand Tracking Results of Sobel operator. Red point is the hand position



Figure 6. Real-time delta robot control using the hand tracking algorithm.

A. Experiments on collected depth image data

The hand template is the crucial variable of this method and closely connects to the hand detection and tracking results. Also, edge detection operators could influence accuracy because of noise canceled image changed.

Additionally, under the consideration about the differences between human hand's shapes, although any hand shape can do the hand detection, the higher accuracy can be achieved if the hand template is acquired from tester's hand.

A hand template can do both right and left hands detection, which means single hand template can be used to for any hand detection. However, it is worth noting that the right-hand template has higher accuracy for right-hand detection and lower accuracy for the left hand. With the same hand template, different matrix *ones* have a significant influence on hand detection. Based on the analysis of the results, if the size of matrix *ones* is set to be 21×21 with Sobel operator for body edge detection, as shown in Fig. 5, the accuracy of the result is over 90%.

B. Experiment on a gesture-controlled delta robot

In order to evaluate gesture tracking performance, a real-time delta robot is controlled by the gesture. Delta robot is a parallel robot arm that controlled by three servo motors for three degrees of freedom. With coordinate transfer using delta robot's reverse kinematics, hand position data can be converted to the angles of the servo motor. As shown in Fig.6, the delta robot is controlled by tester's hand. The moving platform follows the hand moving track. To attenuate the effect of incorrect tracking and protect the robot, the controlled program includes the mechanisms that prevent the delta robot from the angle values of 1) negative or 'NaN' calculation result and 2) data mutation that causes dramatical movement.

V. CONCLUSION

This paper aims to show a lightweight hand detection and tracking method without using libraries and machine learning, which can be used directly for hand detection on different

platforms. In addition to simulations, its performance is also evaluated with a delta robot as an example of application. With the error attenuation coding, the delta robot perfectly follows the hand movement. Multiple experiments results show that this method is highly feasible for real-time system.

According to the experiments, this method is suitable for specific gesture detection and tracking, although it is available to be used for gesture recognition which needs the set of hand templates.

The future work focuses on the accuracy improvement in the real-time system and different gesture tracking by optimizing hand templates.

REFERENCES

- [1] X. Liang, H. Heidari, and R. Dahiya, "Wearable Capacitive-Based Wrist-Worn Gesture Sensing System," in IEEE New Generation of Circuits and Systems (NGCAS), Sept. 2017, pp. 181-184.
- [2] Z. Zhang, "Microsoft Kinect Sensor and Its Effect," IEEE MultiMedia, vol. 19, no. 2, pp. 4-10, 2012.
- [3] J. Suarez and R. R. Murphy, "Hand gesture recognition with depth images: A review," in IEEE RO-MAN: the IEEE Int. Symposium on Robot and Human Interactive Communication, 2012, pp. 411-417.
- [4] A. Jana, Kinect for windows SDK programming guide. Packt Publishing Ltd, 2012.
- [5] A. Davison, Kinect open source programming secrets: Hacking the Kinect with OpenNI, NITE, and Java. McGraw-Hill New York, 2012.
- [6] V. Frati and D. Prattichizzo, "Using Kinect for hand tracking and rendering in wearable haptics," in IEEE World Haptics Conference, 21-24 June 2011 2011, pp. 317-321.
- [7] X. Liang, R. Ghannam, and H. Heidari, "Wrist-Worn Gesture Sensing With Wearable Intelligence," IEEE Sensors Journal, vol. 19, no. 3, pp. 1082-1090, 2019.
- [8] S. Zuo, K. Nazarpour, and H. Heidari, "Device Modeling of MgO-Barrier Tunneling Magnetoresistors for Hybrid Spintronic-CMOS," IEEE Electron Device Letters, vol. 39, no. 11, pp. 1784-1787, 2018.
- [9] J. Zhao, R. Ghannam, M. Yuan, H. Tam, M. Imran, and H. Heidari, "Design, Test and Optimization of Inductive Coupled Coils for Implantable Biomedical Devices," Journal of Low Power Electronics, vol. 15, no. 1, pp. 76-86, 2019.
- [10] A. Pfister, A. M. West, S. Bronner, and J. A. Noah, "Comparative abilities of Microsoft Kinect and Vicon 3D motion capture for gait analysis," Journal of Medical Engineering & Technology, vol. 38, no. 5, pp. 274-280, 2014.
- [11] J. Fu, S. Wang, Y. Lu, S. Li, and W. Zeng, "Kinect-like depth denoising," in IEEE International Symposium on Circuits and Systems, 20-23 May 2012 2012, pp. 512-515.
- [12] K. O. Htet, R. Ghannam, Q. H. Abbasi, and H. Heidari, "Power Management Using Photovoltaic Cells for Implantable Devices," IEEE Access, vol. 6, pp. 42156-42164, 2018.
- [13] J. L. Raheja, A. Chaudhary, and K. Singal, "Tracking of Fingertips and Centers of Palm Using KINECT," in Third International Conference on Computational Intelligence, Modelling & Simulation, 20-22 Sept. 2011 2011, pp. 248-252.
- [14] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834-848, 2018.