2012 International Conference on Image, Vision and Computing (ICIVC 2012)
IPCSIT vol. 50 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V50.9

A Hand-Tracking HCI Interface

Chuan-Kai Yang [+] and Yu-Chun Chen

National Taiwan University of Science and Technology

Abstract. Human computer interaction, or HCI for short, concerning how people interact with computers, has long been an important and popular research field. Traditionally keyboards and mice are the most commonly used devices. In order to have a more flexible interaction, this paper proposes a vision-based interface for human computer interaction based on hand contour tracking. As an articulated hand can provide more freedom, the versatility of possible interactions also increases. Our algorithm combines several existing methods and further improves them to provide a more accurate and responsive hand tracking system. In addition, we also propose a novel windows interface to facilitate human computer interaction.

Keywords: Hand Tracking, Hand Gesture Recognition, Human Computer Interaction, Skin Detection.

1. Introduction

Keyboards and mice have long been the main devices we used to interact with computers. Though still not completely realistic, fancy HCI applications such as those shown in the science fiction movies *Minority Report* and *Iron Man* have impressively demonstrated the potential and trend of HCI technologies that will be very soon made available. Since the invention of Wii motes and Kinect sensors, different kinds of human computer interactions are possible and have been developed. However, the need to carry an extra Wii mote device and the current inaccuracy of Kinect sensors still leave room for further improvement. Nevertheless, as one can very often observe, compared with traditional keyboard/mouse interfaces, the exclusive use of hands in the new trends of interactions has distinguished itself by offering a more intuitive and natural way for communication. Furthermore, the increasingly popular concept of *ubiquitous computing* has called for convenient and portable input devices, thus making hand-gesture inputs even more attractive. For example, a smart phone equipped with the capability of hand gesture recognition could be a good input substitute for its intrinsically small touch screen or keypad. Rather than *data gloves*, which transfer hand gestures through relatively expensive electronic devices, we are more interested in recognizing the gestures of a bare hand. In this regard, there exist works that can track a 2D articulated hand model. In this paper, we make further improvement in terms of computation efficiency and propose a novel window interface to be coupled with the hand-tracking system for more user-friendliness in human computer interactions.

The rest of the paper is organized as follows. Section 2 reviews works related to this paper. Section 3 provides background to make it easier for understanding our proposed methods. Section 4 presents our approaches for designing a hand tracking system. Section 5 describes the novel window interface we propose. Section 6 concludes this work and gives suggestions for possible future directions.

2. Related Work

Vision-based methods for detecting or tracking objects have been important research issues. Among these methods, hand tracking is what we concerned in this work. Generally speaking there are several types of approaches adopted. The first type is based on a hand's 3D model information, as proposed by Rehg [4].

[+] Corresponding author. Tel.: + 886-2-2737-6756; fax: +886-2-2737-6777.
E-mail address: ckyang@cs.ntust.edu.tw.

However, as it usually involves more cameras and more complex computation, its popularity is limited. The second type depends on skin detection, as proposed by Manresa et al. [5] and Zaletelj et al. [6]. The main problem with this kind of approaches is the need to know where the hands locates, when there may exist other parts of human bodies with skin colors, such as arms or heads. The third type is based on image difference, as proposed by [7, 8, 9]. The main idea is to make use of the difference between the current image and a pre-captured background image. The last type makes use of machine learning, such as the methods proposed by [10, 11, 12, 13]. The former two are mainly for face detection, while the latter two apply similar framework but generalize them to hand recognition and tracking.

3. Background

To ease the understanding of this work, we hereby briefly describe the involved numerous techniques in this section.

3.1. Deformable templates

Based on the approaches proposed by Blake et al. [1], we represent the hand for tracking as a set of B-splines, where each of them is controlled by a vector called state. Through such a mechanism, each B-spline can be made to scale, translate, rotate, etc. As a result, such a deformable template can be used to track a hand.

3.2. Measurement method

Figure 1 briefly shows how we perform our measurement. For each measurement point, there will be a measurement line, along which we detect the features in a frame to determine the score of this measurement point. The multiplication of all the scores of the measurement points on all measurement lines lead to a fitness value that tells how well a hypothetical contour fits the real contour.

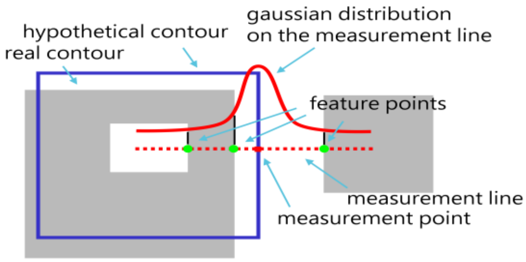


Fig. 1: measurement method

3.3. Condensation algorithm

By combining the concepts of particle filtering and deformable template, the condensation algorithm [1] treats each contour as a particle, which represents the state of a hypothetical contour, and its corresponding weight is the fitness of the contour. The main idea of condensation is to repeatedly perform the three steps of re-sampling, prediction and measurement to “evolve particles”. Due to lack of space, for more details, please refer to [1].

3.4. Sweep tracker algorithm

In our implementation, we adopt a hierarchical deformable template where the first level is the palm, the second level includes four fingers and the first knuckle of the thumb, and the third level is the second knuckle of the thumb. The condensation algorithm is used to track the palm, while we use the sweep tracker algorithm [3] to track the angle and length of fingers.

4. Improved Hand Tracking System

In this section, we describe how the hand tracking is fulfilled and how the improvements are made.

4.1. Articulated hand template

By assuming that the hand to track moves perpendicularly to where the camera is pointing at, we adopt the articulated hand template similar to [3]. There are totally 50 control points for the B-splines and 14 degrees of freedom, $(x, y, \alpha, \lambda, \theta_0, l_0, \theta_1, l_1, \theta_2, l_2, \theta_3, l_3, \theta_4, \theta_5)$ to allow a hand to move freely, as shown in Figure 2, where (x, y) is the central location, and (α, λ) are the rotation angle and size of the tracked hand, respectively, while (θ_i, l_i) are the angle and length of a particular finger.

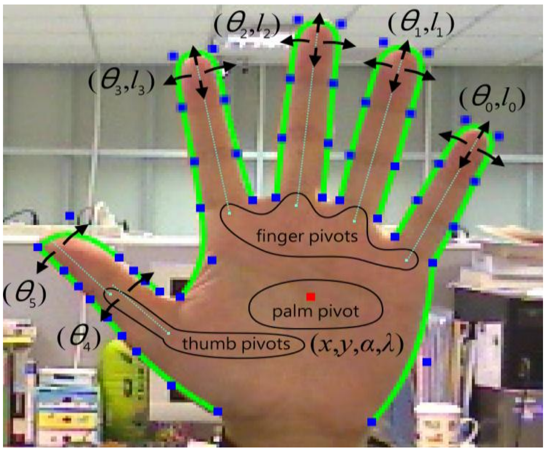


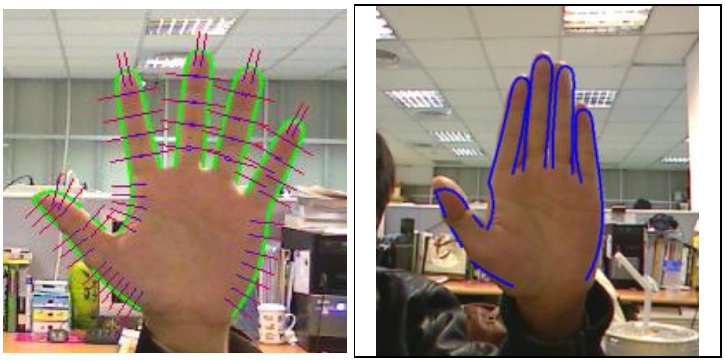
Fig. 2: the hand contour template

4.2. Dynamic model

To predict the an articulated contour model, we assume each knuckle behaves like an independent oscillator where its damping constant, natural frequency and root-mean-squared average displacement values are set according to [1].

4.3. Contour measurement model

As shown in Figure 3(a), we deploy measurement lines in the way similar to [1]. In addition, we also make use of the skin detection algorithm in [3] to facilitate the associated calculation. However, as a user may close his/her hand as shown in Figure 3(b), the measurement process may get erroneous results. As a consequence, edge detection techniques are introduced so that the contour of fingers can be correctly detected, and thus the corresponding hand gestures.



(a) (b)

Fig. 3: measure lines that are perpendicular to the hand contour

5. Novel Window Interface

In this section, we present the novel window interface we designed to facilitate friendly HCI.

5.1. Initialization

To increase the accuracy of our hand tracking system, an initialization process is designed, as shown in Figure 4. Figure 4(a) indicates the case where our system thinks the hand’s fitness is still below a desired threshold. Figure 4(b) represents the case where the fitness value has reached the desired threshold, so our system starts some parameters’ tuning and initializes the tracking process. Once this matching status has lasted for consecutive 10 frames, as shown in Figure 4(c), the hand tracking process starts to function.

(a) (b) (c)

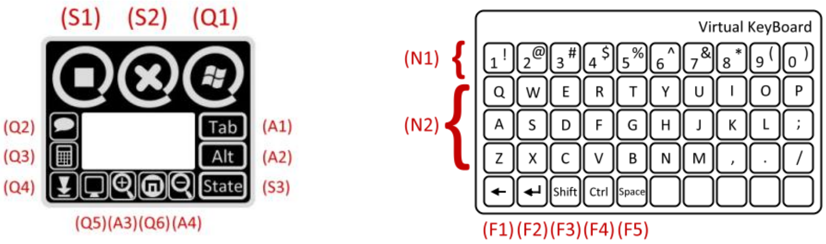
Fig. 4: hand contour initialization

5.2. Clicking/Dragging detection

For detecting a click, we need to detect the change of finger length. To do so, we applied the technique EWMA (Exponentially Weighted Moving Average) [3] to see if the length of a tracked finger falls below a threshold, and if so, a clicking is triggered. Note that the clicking position can also be determined as the whole hand is under the tracking process. To emulate a dragging, we set a finger length range; that is, once the tracked hand is moving with the finger length falling within the specified range, a dragging is triggered.

5.3. Virtual touch panel

To make our system even more useful, we have also implemented two virtual touch panels, one for the mouse, and one for the keyboard, as shown in Figure 5(a) and Figure 5(b), respectively. Some hot-keys can be used to switch on/off these virtual touch panels when necessary.



(a) (b)

Fig. 5: the mouse interface and keyboard interface

5.4. Results and system environment

This system was implemented on a machine with Intel Core 2 Duo 1.86GHz, 3GB memory, running on Windows XP. The camera in use is Logitech’s QuickCam which can process frames of size 640×480 at the speed of 30 frames per second. The involved programming language is C# and the DirectShow library is used for capturing video frames.

6. Conclusions and Future Work

This paper presents a hand tracking system that improves both the Condensation algorithm proposed by Blake and Isard [1], and the hand tracking system proposed by Tosas [3]. The proposed system can offer real-time performance, and thus is more suitable for being integrated with other applications. We have also developed a novel window interface with virtual touch panels to be coupled with the hand tracking system to further assist desired and friendly human computer interactions.

7. Acknowledgements

This work was supported in part by the National Science Council under the grant NSC 97-2221-E-011-109.

8. References

[1] Blake, A. and Isard, M. (1998). *Active contours.* Springer.

[2] Isard, M. and MacCormick, J. (2000). *Hand tracking for vision-based drawing.* Technical report, Visual Dynamics Group, Dept. Eng. Science, University of Oxford.

[3] Tosas, M., (2006). Visual Articulated Hand Tracking for Interactive Surfaces. PhD thesis, Nottingham University.

[4] Rehg, J., (1995). Visual Analysis of High DOF Articulated Objects with Application to Hand Tracking. PhD thesis, Electrical and Computer Eng., Carnegie Mellon University.

[5] Manresa, C., Varona, J., Mas, R. and Perales, F. J., (2005). *Hand Tracking and Gesture Recognition for Human-Computer Interaction.* Electronic Letters on Computer Vision and Image Analysis, 5, 3, 96-104.

[6] Zaletelj, J., Perhavc, J. and Tasic, J. F., (2007). *Vision-based human-computer interface using hand gestures.* Eight International Workshop on Image Analysis for Multimedia Interactive Services. (WIAMIS' 07)

[7] Yuan, X. and Lu, J., (2005), *Virtual Programming with Bare-Hand-Based Interaction.* Proceedings of the IEEE International Conference on Mechatronics & Automation Niagara Falls, Canada.

[8] Hardenberg, C. V. and Bérard, F. (2001) *Bare-Hand Human-Computer Interacion.* Proceedings of the ACM Workshop on Perceptive User Interfaces, Orlando, Florida, USA, Nov. 15-16 2001.

[9] Letessier, J. and Bérard, F. (2004). *Visual Tracking of Bare Fingers for Interactive Surfaces.* in: UIST '04: 17th Annual ACM symposium on User Interface Software and Technology, ACM Press, New York, NY, USA, 2004, 119–122.

[10] Rowley, H., Baluja, S. and Kanade. T., (1998), *Neural network-based face detection.* In IEEE Patt. Anal. Mach. Intell., volume 20, pages 22–38, 1998.

[11] Viola, P. and Jones, M, (2001), Robust real-time face detection. In ICCV, volume 20, 11, 1254–1259.

[12] Wu, X., Xu, L., Zhung, B. and Ge, Q. (2003) *Hand Detection Based on Self-Organizing Map and Motion Information.* IEEE Int. Conf. Neural Networks & Signal Processing Nanjing, China, December 14-17, 2003

[13] C. S. Fahn and C. N. Yang, (2009), Two-Hand Fingertip Identification and Gesture Recognition Techniques Applied for Human-Computer Interaction Systems in Real Time. National Taiwan University of Science and Technology.

[14] Isard, M. and Blake, A. (1998a). Condensation - conditional density propagation for visual tracking. Int. J. Computer Vision, 28, 1, 5-28.

[15] MacCormick, J. and Blake. A. (1999). *A probabilistic exclusion principle for tracking multiple objects.* In Proc. 7th International Conf. Computer Vision, 572-578.

[16] MacCormick, J. and Isard, M. (2000). Partitioned sampling, articulated objects, and interface-quality hand tracking. In European Conf. Computer Vision.