

Hand Gesture Recognition using Depth Data

Xia Liu
Ohio State University
Columbus OH 43210

Kikuo Fujimura
Honda Research Institute USA
Mountain View CA 94041

Abstract

A method is presented for recognizing hand gestures by using a sequence of real-time depth image data acquired by an active sensing hardware. Hand posture and motion information extracted from a video is represented in a gesture space which consists of a number of aspects including hand shape, location and motion information. In this space, it is shown to be possible to recognize many types of gestures. Experimental results are shown to validate our approach and characteristics of our approach are discussed.

Index Items: *real-time depth data, shape analysis, gesture recognition.*

1. Introduction

Gesture recognition has been a subject of much study lately as a promising technology for man-machine communication. Various methods have been proposed to locate and track body parts (e.g., hands and arms) including markers, colors, and gloves. We propose a method that does not require color information or extra device to be worn by the user. Such a gesture recognition module would have many applications including man robot communication, intelligent rooms, and interactive games.

We approach this problem by analyzing real-time depth data obtained by a time-of-flight camera. Examples of depth data are illustrated in Fig. 1, where (a) and (b) contain gray and depth images, respectively. For many gesture recognition methods based on gray images, it is not a trivial task to segment the hand in the first place, while in the depth image, it is quite feasible by selecting an appropriate depth level to separate the hand from the rest of the image. Another advantage of our approach is that we can track the hand location in 3-dimension. Furthermore, as you can see in Fig. 1(b), it is also possible to locate the hand position relative to the head. Our algorithm makes good use of these characteristics to classify many kinds of gesture.

The remainder of the paper is organized as follows. After a brief survey of related work in Section 2, our method is presented in Section 3. Section 4 contains experimental results and analysis of our method and Section 5 contains a description of the sensor we use. Section 6 concludes the paper.



a. Gray image



b. Depth (The closer the object is, the brighter the intensity.)

Figure 1. Gray and depth image examples

2. Related work

For hand detection, which is an essential component for gesture recognition, many approaches use color or motion information [6,7]. For these approaches, however, tracking hand motion is a non-trivial task under challenging lighting conditions [12,14]. Some use special equipment such as gloves, while some use a background with specific color to make the task feasible [2,11]. For the ease of hand shape analysis, a magnified view can also be used [1, 13], in which case body and arm posture information might be lost. To capture the whole body, some use multiple cameras to extract 3D information of the subject [5,9]. To understand the 3D volume obtained, 3D model fitting has also been proposed, yet it is a complex and computation-intensive process and frequently unstable. Stereo vision, which is a popular choice in depth sensing [9], usually does not provide image resolution sufficient for hand shape analysis due to lack of texture on the subject. Thus, depth has primarily been used in limited gestures such as large pointing motions [5]. Some use high resolution 3D data obtained by using coded lighting for static hand posture analysis [15].

As compared with existing approaches, salient features of our method are as follows. (i) Multiple features can be analyzed simultaneously including hand location, hand shape, movement, orientation, and speed, thereby forming a feature space sufficient for a large vocabulary. (ii) Gestures are recognized without using any special background, marks, or gloves. (iii) Each computation module is computationally inexpensive, thereby achieving real-time gesture recognition. (iv) Due to depth sensing using infrared, the method is less susceptible to illumination changes.

3. Algorithm

The key idea of our gesture recognition is to make full use of 3D trajectory as well as shape information. Our algorithm has the following components.

- I. Hand detection
- II. Hand shape analysis
- III. Hand trajectory analysis
- IV. Classification of <shape, location, trajectory, orientation>

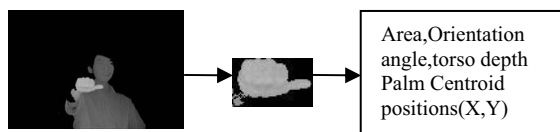


Figure 2. Flow of the algorithm

In the following, we assume that only one person is present in the video, facing toward the camera. This simplifies human and hand detection. We also assume in our gesture recognition that the hand position is distinctly apart from the body.

3.1 Hand and head detection

Human detection is relatively simple by using depth images, since the foreground can be separated from the background by a depth constraint. Under the assumption that there is a single person in the camera view, the task is further simplified. The major blob in the image with distance smaller than a pre-defined threshold value can be treated as a human. After a human has been detected in the scene, we detect his head position as well. This information is valuable, since certain gesture patterns require knowledge of the hand location relative to the rest of the body. In a depth image, the head can be detected by using vertical and horizontal histograms as in Fig. 4. As far as the image contains a single user, this method works well.

For hand gesture, the hand is usually at a distance different from that of the body (Fig. 3). To detect the hand, we also use an aspect ratio constraint to eliminate the arm part from the hand.



Figure 3. Hand and head detection

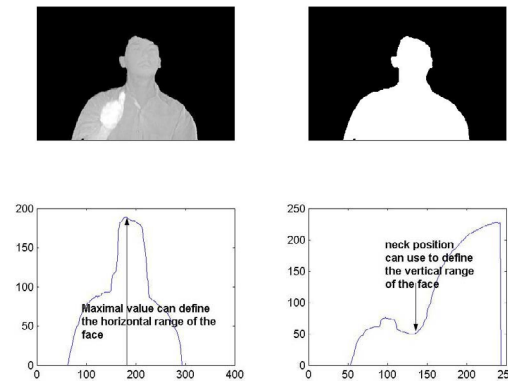


Figure 4. Simple head detection

3.2 Hand shape analysis

For our hand shape analysis, we establish a dataset that includes various hand shapes for different gestures. Figure 5 contains various hand patterns we have collected so far.

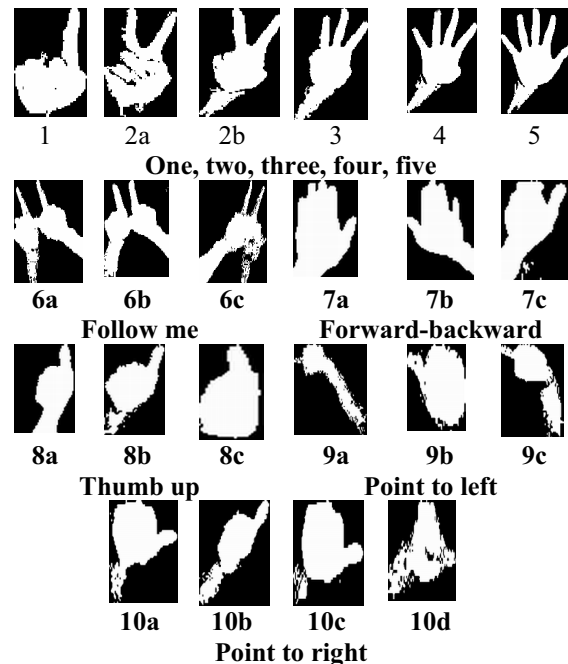
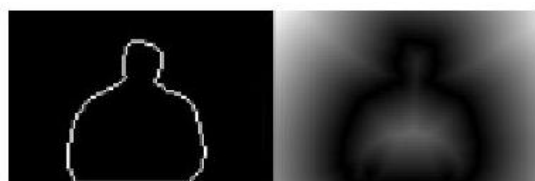


Figure 5. Various hand patterns.

To match an unknown hand shape against our hand shape data, we use the Chamfer Distance (CD) between two images to measure shape similarity. CD is defined by using Distance Transform for shape comparison. The result of

distance transform is a gradient image where each pixel corresponding to a part of an edge contains 0 while other pixels contain the distances to the closest edge points. To calculate CD between an unknown image X and template Y , we take correlation between the edge image of X and DT of Y .



Edge image DT image
Figure 6. Example of Distance Transform (gradient image)

The smaller the value is, the higher the similarity. We can define such a similarity measure S as:

$$S = \sum_i \sum_j E(i, j) \times DT(i, j)$$

where E stands for the edge image of X and DT stands for the gradient image of template Y . We store distance-transformed images for all the shapes in the template dataset. For an unknown hand shape, the similarity calculation as above is conducted and the pattern with the smallest value is chosen as the pattern for this hand. Temporal voting is used in decision making to reduce the bias created by particular images in the sequence. Alternatively, for hand shape analysis, a method such as [1] could also be used to strengthen the finger-counting part of the algorithm.

3.3 Hand trajectory analysis

The hand moving in space creates a curve (trajectory) in 3-space with time information associated to it (Fig. 7). To simplify our matching process, we decouple the trajectory into three parts, namely, X , Y , and Z movements. Here, X and Y are the horizontal and vertical coordinates of the hand in the image plane, while Z is represented by the mean depth value of the hand to indicate the distance to the camera. We use the following processing steps.

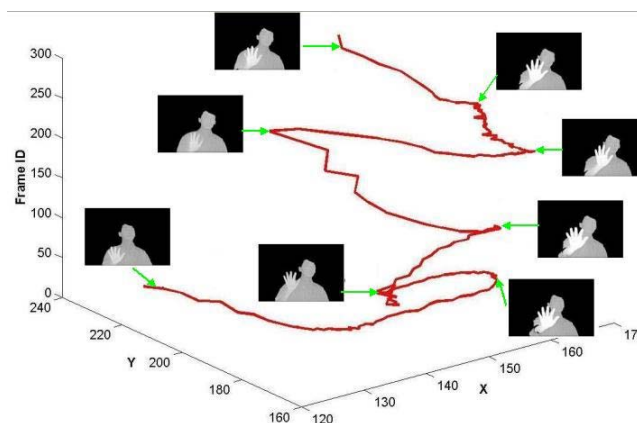


Figure 7. Spatial trajectory of hand position.

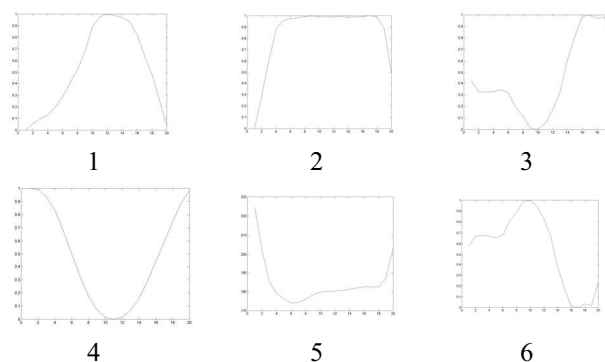


Figure 8. Curve templates

1. The X , Y , and Z components of each hand movement are matched against template curves (which are normalized to a uniform time interval $[0, N]$, where N is the number of sampled points). Figure 8 contains some example template patterns for X , Y , and Z . These correspond to actions such as “push and return (pattern 1, Fig. 8)”, “push, hold, and return (pattern 2)”, “push, return, pull, and return (pattern 3)”. The curves have been smoothed by using FFT to reduce local noise before normalization. (Currently, we do not consider amplitude information of hand movements.)
2. For curve matching, we only consider significant movements in some dimension. For example, for the “back up” gesture, the hand mainly moves forward from torso to the camera direction. However, there might also be some horizontal and vertical displacement of the hand in the image plane, but such a displacement is negligible, thus we exclude such a movement from consideration.
3. For any non-negligible movement, we compare an given curve with all templates in the curve dataset. We use the minimum square error to compare similarity

between curves. Namely, for an unknown curve $Y(t), t = 0, 1, \dots, N$ and template curve $S(t), t = 0, 1, \dots, N$, we use:

$$MSE = \sum_{t=1}^N [Y(t) - S(t)]^2$$

as the measurement for similarity. The smaller MSE is, the higher the degree of similarity.

3.4 Hand orientation

Currently, we measure hand orientation for selected hand shapes only. Figure 9 shows the result of hand orientation analysis for the “thumb-up” hand pattern. Orientation information has been obtained by examining the major axis of the region for the hand shape.

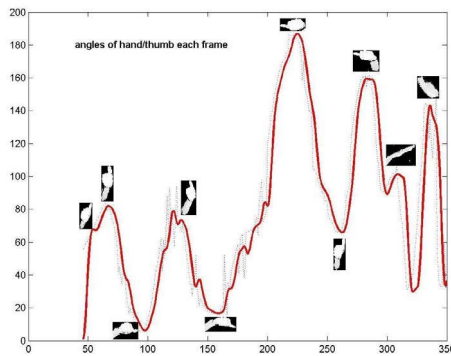


Figure 9. Hand orientation analysis example

3.5 Pattern classification

After we have collected information on hand shape and its trajectory, we now classify gestures based on all feature information. Our gesture representation takes the form :

$G = \langle \text{shape, location, trajectory, orientation, speed} \rangle$

Since each attribute can distinguish a few to several patterns, we can see that this space G can easily contain several dozens of gesture patterns.

One important remaining issue from a pragmatic viewpoint is how to tell when a certain gesture begins and ends. One solution is to introduce a gesture delimiter that is to be used always at the beginning and the end of a gesture (such as hand pattern 1 as in Fig. 1d). Some times, the hand disappears after the end of one gesture, which can also be used as a natural delimiter. Figure 10, for instance, contains a snapshot consisting of 300 frames (10 seconds). For frames 1~130, the hand moves forward and backward in the Z direction; for frames 130~180, no hand is detected. After frame 180, the hand again moves forward and keeps still at

a fixed depth value from frames 200~260. Thus, frames 130~180 can be interpreted as a delimiter between two gestures.

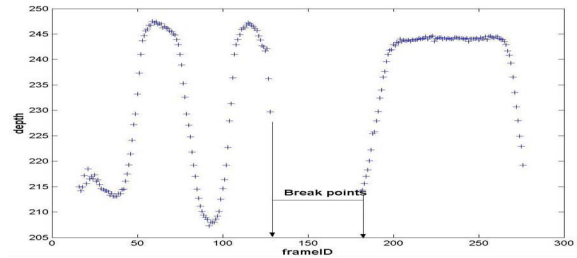


Figure 10. Gesture separation example

To understand natural continuous human gestures, methods such as HMM are to be considered. Figure 11 contains a flow diagram to illustrate the steps for gesture recognition.

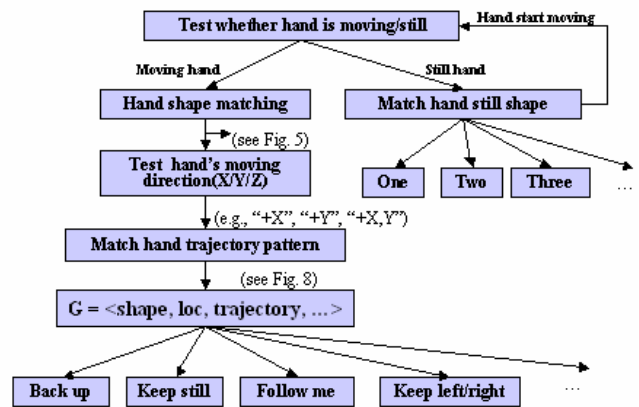






















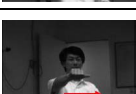




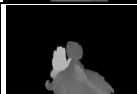




Figure 11. Diagram for gesture recognition

4. Experimental results

We now include our gesture recognition sample set. (See Table 1). This sample set has been designed to give motion instructions to a mobile robot. By using gesture commands listed here, we can give instruction such as “[Back-up] [three steps] and [stop]” and “[Step to the side] by [this much] and [Turn left]”. We could also use a similar command set with some modification to control home device such as TV (e.g., “volume down”, “move to the next channel”).

Gesture (gray & depth)		Function	Hand shape	X,Y,Z curve
		One/Look here/Begin	1	0,0,0
		Two	2a,2b	0,0,0
		Three	3	0,0,0
		Four	4	0,0,0
		Five/Stop	5	0,0,0
		Back up	5,7a	0,5,1
				
		Keep still	5,7a	0,5,2
				
		Follow me (ASL)	6a,6b,6c	2,0,0
				
		Step to left	10a,10b,10c,10d	1,0,0
		Step to right	9a,9b,9c	4,0,0
		Thumb up (Well done!)	8a,8b,8c	0,3,0
		Keep moving	7a,7b,7c	0,0,0




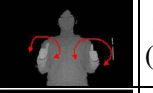


				
		Show size (this much)	1	0,0,0
		No, Not good	8a,8c	0,0,0
Note two hand shapes are detected in one frame in some gestures.				

Table 1: Sample gesture patterns.
(X, Y, Z curve type 0 means “static” in this direction.)

Assuming that gesture commands begin with a delimiter, our gesture commands have been interpreted correctly (error rate 5% or less). The majority of error occurs currently in hand shape analysis. Due to temporal voting, we are likely to get correct answers most of the time. Note that there are many unused gesture combinations. For our gesture space $G = \langle \text{shape, loc, trajectory, orientation, speed} \rangle$, we only use shape and trajectory information in the above example of Table 1 (hand orientation as in Fig. 9 is not used here.) Additional gesture interpretations can be easily obtained by using ‘loc’ attribute (i.e., hand location relative to the head). For “back-up” gesture, for instance, if the hand is located far left relative to the face, then it could mean ‘Back-up into the left direction’. If we make a full combination of our gesture space attributes, the vocabulary can be increased up to one hundred.

5. Sensing technology

We now describe the sensor that we have used for this work. Recently, a new type of depth perception method has been introduced based on the time-of-flight principle [8]. This method can capture depth and colour information simultaneously using the same optical axis in real-time. Moreover, it allows us to specify a range $[D_{min}, D_{max}]$ for which object depth information is to be recorded. We take advantage of this unique feature so as to eliminate background clutters by setting D_{max} immediately behind the subject being imaged (i.e., remove any object beyond certain depth from the image acquired).

The image capturing method is based on active sensing, where a pulse of infrared illumination is projected to the object and the sensor reads its echo signal from the object. Furthermore, the device has a high-speed shutter by which it controls the sensing scope for objects whose signal can enter the sensor. For a far object, the echo can reach the device only after the shutter has been closed. Thus, no signal is recorded for the corresponding object (thus, most

background objects disappear from the recorded scene). See [8] for more details.

Compared to stereo systems, this device has the following characteristics about depth information obtained:

- Illumination-invariant in in-door environments, as long as the environment does not contain light of the same wavelength used for pulse of the sensor.
- Error in depth is approximately 5~15mm for 0.5~3m depth window. Depth information is recorded in the separated depth image (8 bits a pixel). This gives sufficient separation between the hand and the body used for gesture recognition.
- Both depth and colour images are captured at real-time speed (30 frames/sec).
- Depth for objects with no texture (such as walls) can still be obtained. However, for certain objects that does not return any signal (e.g., light absorbing materials), distance cannot be obtained.

6. Concluding remarks

We have presented a method for gesture recognition based on depth image sequences. Strengths of our approach are as follows. (1) The 3D data acquisition is in large part insensitive to illumination changes as long as the environment does not contain light component that interferes with that of the active illumination. (2) Face and hand segmentation is done with ease using 3D, which has been the bottleneck for many algorithms based on gray image analysis. (3) A number of attributes are considered, thereby forming a large gesture space. Experimental results have been shown to illustrate cases where this approach is promising.

Due to the structure of our gesture space, our method allows interpretation of many gesture patterns. Many of them would appear quite artificial, however, they would be easier to understood by machines. Our further study includes incorporation of more movement patterns for further enriching the vocabulary possibly by using gray as well as depth information. Another topic is to understand gesture patterns that are easily understood by humans such as sign languages. We believe that our method is extendible to cases where both the user as well as the camera (e.g., robot) are in motion.

References

- [1] K. Oka, Y. Sato, and H. Koike, Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems, *Proc. of the 5th Intl. Conf. on*

- Automatic Face and Gesture recognition*, May 2002, Washington D.C.
- [2] E. Polat, M. Yeasin, and R. Sharma, Robust tracking of human body parts for collaborative human computer interaction, *Computer Vision and Image Understanding* 89(1): 44-69, 2003.
- [3] D.M.Gavrila and V.Philomin, Real-time Object Detection for Smart Vehicles, *Proc. of IEEE International Conference on Computer Vision*, pp. 87-93, Kerkyra, Greece, 1999.
- [4] A. Wilson and A. Bobick, Parametric hidden markov models for gesture recognition, *IEEE Trans. on Pattern Anal. Mach. Intel.* 21(9):884-900, 1999.
- [5] N. Jovic, B. Brumitt, B. Meyers, S. Harris, and T. Huang, Detection and estimation of pointing gestures in dense disparity maps, *Proc. of the 4th Intl. Conf. on Automatic Face and Gesture Recognition*, 2000, Grenoble, France.
- [6] L. Bretzner, I. Laptev, and T. Lindeberg, Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering, *Proc. of the 5th Intl. Conf. on Automatic Face and Gesture Recognition*, May 2002, Washington D.C., 423-428.
- [7] V. Pavlovic, et al. Visual interpretation of hand gestures for human-computer interaction: a review, *IEEE Trans. on Pattern Anal. Mach. Intel.* 19(7):677-695, 1997.
- [8] G.J. Iddan and G. Yahav, 3D imaging in the studio, *SPIE Vol.* 4298, 2000, pp.48.
- [9] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, Intelligent ASIMO: System overview and integration, *IEEE Intelligent Robots and Systems*, Genova, Swiss, 2478-2483, 2002.
- [10] C. Vogler and D. Metaxas, ASL recognition based on a coupling between HMMs and 3D motion analysis, *Proc. Int. Conf. Computer Vision*, Bombay, 1998.
- [11] T. Sarner, J. Weaver, and A. Pentland, Real-time American Sign Language recognition using desk and wearable computer based video, *IEEE Trans. Pattern Anal. Mach. Intel.* 20, 1998, 1371-1375.
- [12] Y. Zhu, G. Xu, and D.J. Kriegman, A real-time approach to the spotting, representation, and recognition of hand gestures for human-computer interaction, *Computer Vision and Image Understanding* 85(3), 189-208, 2002.
- [13] V.Athitsos and S. Sclaroff, An appearance-based framework for 3D handshake classification and camera viewpoint estimation, *Proc. of the 5th Intl. Conf. on Automatic Face and Gesture Recognition*, May 2002, Washington D.C., 45-50.
- [14] X. Zhu, J. Yang, and A. Waibel, Segmenting hands of arbitrary color, *Proc. of the 4th Intl. Conf. on Automatic Face and Gesture Recognition*, March 2000, Grenoble, 446-453.
- [15] S. Malassiotis, N. Aifanti, and M.G. Strintzis, *1st Intl. Symp. On 3D Data Processing, Visualization, and Transmission*, June 2002, Padova, Italy.