Gesture Recognition using Microsoft Kinect®

K K Biswas
Department of Computer Sc. & Engineering
Indian Institute of Technology, Delhi
kkb@cse.iitd.ernet.in

Saurav kr Basu
Department of Electronics & Communication Engineering
Birla Institute of Technology, Mesra
sauravkumarbasu@gmail.com

Abstract

Gesture recognition is essential for successful and complete human – machine interaction. In this paper we propose a method to recognize human gestures using a depth camera, i.e. Kinect®. The camera views the subject in the front plane and generates a depth image of the subject in the plane towards the camera. This depth image is then used for background removal, followed by generation of the depth profile of the subject. In addition to this, the difference between subsequent frames gives the motion profile of the subject and is used for recognition of gestures. These allow the efficient use of depth camera to successfully recognize multiple human gestures. The result of a case study involving 8 gestures is shown. The system was trained using a multi class SVM.

1. Introduction

Gestures are an important means of communicating in our day-to-day life. In this demonstration we present our experience of using a Kinect® depth camera for recognition of some common gestures. The system grabs frames from the camera and identifies the gesture embedded. For our study we picked up the following eight different gestures:

- CLAP: Clapping
- CALL: Hand gesture to call someone
- GREET: Greeting with folded hands
- WAVE: Waving hand
- NO: Shaking head sideways – "NO"
- YES: Tilting head up and down – "YES"
- CLASP: Hands clasped behind head
- REST: Chin resting on Hand

The following steps were used to extract the features from the video clips of each type of gestures.

1.1. Pre-processing

The first step that we need to do is to isolate the human making the gestures from the background scene. This is done by background subtraction from the depth image of the scene using auto thresholding, proposed by Riddler and Calvard [1], on the depth histogram. Fig. 1(b) shows a typical depth histogram corresponding to video frame shown in fig. 1(a). The threshold is found from the valley following the first large peak. This enables the foreground to be extracted as shown in fig. 1(c).

The next step is to figure out the position of hand with respect to rest of the body. The histogram of the extracted foreground from the image is shown in fig. 1(d). As it relates to body parts very close to each other, the histogram does not reveal any more details. To bring in more clarity, we carry out histogram equalization. The result is shown in fig. 1(e). It is found that for different gestures the equalized histogram patterns turn out to have distinct distribution.

Figure 1(a): Depth image

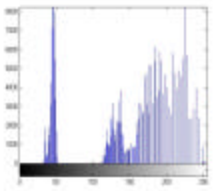

Figure 1(b): Depth Histogram

Figure 1(c): Extracted foreground

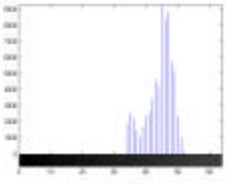

Figure 1(d): Foreground histogram

Figure 1(e): Equalized foreground

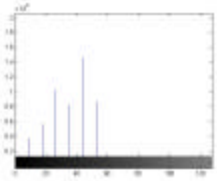

Figure 1(f): Equalized histogram

1.2. Features from ROI

A region of interest (ROI) is created by placing a 14x14 grid on the extracted foreground. The gesture is parameterized using depth variation and motion information content of each cell of the grid. For each cell the number of pixels lying in each bin is counted, and is normalized. This can be visualized as segmenting the histogram of each cell into 10 specified levels and storing the normalized count of pixels in each bin. Fig. 2 illustrates this part.

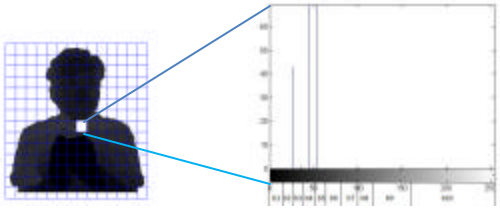

Figure 2(a): Segmented ROI

Figure 2(b): Histogram of the Highlighted Box (b1 to b10: bins)

The motion information is extracted by noting the variation in depth between each pair of consecutive frames. It is obtained by subtracting a depth image from the preceding depth image. The difference image gives the path of motion of the body part. Fig. 3(a) shows a typical difference image. An adaptive threshold was used to suppress the noise in the image and convert it into a binary image (fig. 3(b)). The 14x14 ROI grid used above was placed over this image. A normalized count of white pixels in each cell was used to depict the motion content of the corresponding frame.

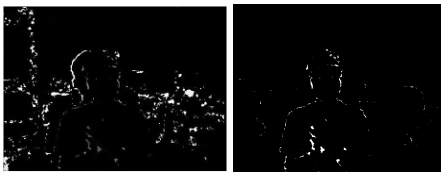

Figure 3(a): Difference Image

Figure 3(b): Binary image after noise removal

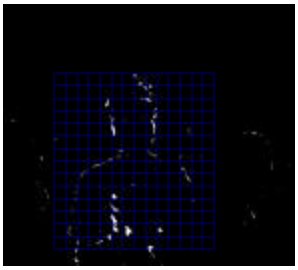

Figure 3(c): ROI grid placed over Binary Image

1.3. Training and Testing

A multiclass SVM was used to train the system for the classification of the 8 gestures. A matrix was generated for the entire training data set.

Each frame of the video was represented by a row of the matrix. The columns represent the feature points.

The training data was created with 5 subjects. The number of training and testing data frames for each of the gestures are given below.

Gesture	Training Set Frames	Testing Set Frames
Clap	2081	593
Cal	1727	603
Greet	1185	512
Wave	1136	316
No	1087	337
Yes	1247	343
Clasp	2156	549
Rest	1797	518

Table 1: Number of Frames for each Activity during Training & Testing

The confusion matrix (Table 2) shows the results of our experiment. A detailed description can be obtained in [2].

2. Working

The system functions by grabbing frames from the camera or a video and applying the gesture recognition algorithm to each frame. However, each frame should contain a subject with one of the following gestures.

3. Conclusions

It is shown that using simply the depth images, it is possible to classify hand gestures. For illustration we picked up 8 gestures. But it appears fairly easy to extend it to larger number of gestures. The accuracy of the results could be improved by making use of the skin color information of the color camera. The method is not

compute intensive, as very few calculations are involved to extract the features. This would be much faster than a method dealing with RGB components for shape and optical flow for motion.

References

[1] T.W. Ridler, S. Calvard, Picture Thresholding Using an Iterative Selection Method, IEEE Trans. System, Man and Cybernetics, SMC-8 ,pp. 630-632, 1978.

[2] Biswas K K and Basu S K, "Gesture Recognition using Microsoft Kinect®", International Conference on Automation, Robotics and Applications, 2011, pp. 100-103.

Gestures	Clap	Call	Greet	Wave	No	Yes	Clasp	Rest
Clap	449	29	19	0	0	0	2	0
Call	35	516	4	3	2	4	22	17
Greet	0	28	464	8	8	3	0	1
Wave	2	12	0	302	0	0	0	0
No	0	85	0	46	195	9	1	1
Yes	7	98	0	7	14	186	25	6
Clasp	7	19	0	6	0	0	516	1
Rest	1	8	1	19	3	0	2	484

Table 2: Confusion Matrix for Frames