Gesture Recognition using Microsoft Kinect®

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

Saurav Kumar Basu
Department of Electronics & Communication Engineering
Birla Institute of Technology
Mesra, India
saurav1292.08@bitmesra.ac.in

***Abstract*— Gesture recognition is essential for human – machine interaction. In this paper we propose a method to recognize human gestures using a Kinect® depth camera. 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 Support Vector Machine.**

Keywords-gestures, segmentation, kinect, depth image, SVM.

I. INTRODUCTION

Gestures are an important means of communicating in our day-to-day life. Often we communicate by the movement of body parts like hands and head rather than speaking. So for a successful machine-human interaction consideration of these gestures is inevitable.

Lot of work has already been reported for gesture and activity recognition. Most of these are based on the use of one or more RGB (colour) cameras. The attempt was often to track a particular body part like head, as done by Madabhushi and Aggarwal [1], or to track the joints as done by Ali and Aggarwal [2] and Uddin, Thang and Kim [4] or to utilize the spatio – temporal motion patterns as proposed by Rui and Anandam [3] to identify a particular or sequence of activities. Zhu and Fujimura [5] have used a flexible Bayesian framework for integrating pose estimation results obtained by methods based on key-points and local optimization. Breuer, Eckes and Müller [6] have used a depth camera to generate a cloud of 3 – D points of human hand and the applied PCA and model fitting to estimate the location and orientation of the hand and identify the associated gesture.

In this paper we present our experience of using a Kinect® depth camera for recognition of some common gestures. Kinect® interprets a 3D scene information using a projected infrared structured light (fig.1). This 3D scanner system called Light Coding employs a variant of image-based 3D reconstruction. The Kinect® sensor is a horizontal bar connected to a small base with a motorized pivot and is designed to be positioned lengthwise above or below the video display. It also has a RGB camera.

The depth sensor consists of an infrared laser projector combined with a monochrome CMOS sensor, which captures video data in 3D under any ambient light conditions. Fig 2 shows the RGB camera image and the depth image of the same scene. The depth map is visualized here using colour gradients from white (near) to blue (far). The monochrome depth sensing video stream is in VGA resolution with 2,048 levels of sensitivity. The colour and depth images taken by Kinect® for a particular scene are shown in fig. 2 a-b.

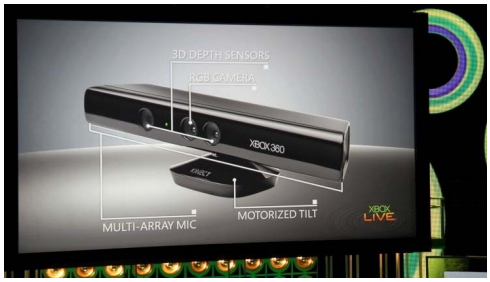

Figure 1. Microsoft Kinect®

Figure 2(a). Colour Image of a scene

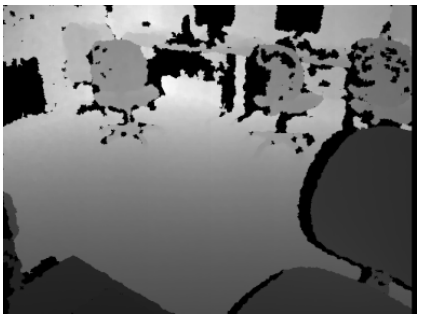

Figure 2(b): Depth Image of the same scene

II. FEATURE EXTRACTION

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 various gestures are shown in fig. 3a-h. The following steps were used to extract the features from the video clips of each type of gestures.

Figure 3(a). Clap

Figure 3(b). Call

Figure 3(c). Greet

Figure 3(d). Wave

Figure 3(e). No – Nodding head sideways

Figure 3(f). Yes – Nodding head up and down

Figure 3(g). Clasp

Figure 3(h). Rest

A. *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. This was done by using auto thresholding, proposed by Riddler and Calvard [7], on the depth histogram. Fig. 4(b) shows a typical depth histogram corresponding to video frame shown in fig. 4(a). The threshold is found from the valley following the first large peak of the histogram. This enables the foreground to be extracted as shown in Fig. 4(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. 4(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. 4(e). It is found that for different gestures the equalized histogram patterns turn out to have distinct distribution.

Figure 4(a): Depth image

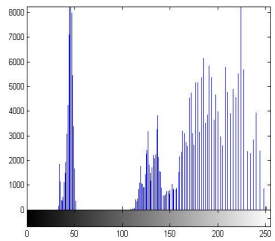

Figure 4(b): Depth Histogram

Figure 4(c). Extracted Foreground

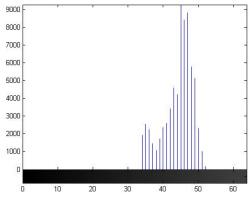

Figure 4(d). Foreground Histogram

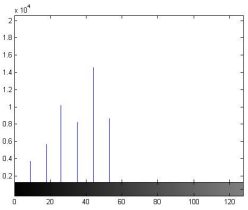

Figure 4(e). Equalized Foreground Figure 4(f). Equalized Histogram

B. *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.

The depth information is extracted by dividing the entire depth gray scale range (0-255) into 10 bins (0-20, 21-30, 31-40, 41-50, 51-60, 61-80, 81-100, 101-120, 121-165, 166-255). 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 the 10 specified levels and storing the normalized count of pixels in each bin. Fig. 5 illustrates this part.

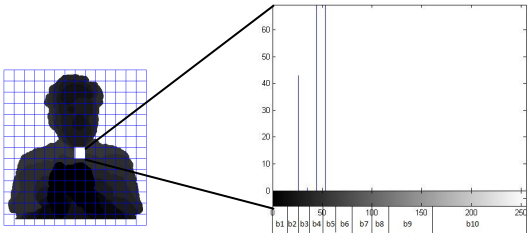

Figure 5(a): Segmented ROI

Figure 5(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. 6(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. 6(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 6(a). Difference Image

Figure 6(b). Binary image after noise removal

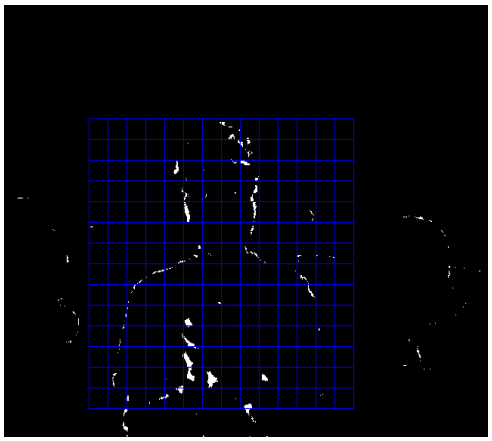

Figure 6(c). ROI grid placed over Binary Image

C. *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 number of columns was 1960 (10 bins for 196 cells) for depth related data, and another 196 for normalized count of motion related data for each cell. Thus, the dimension of SVM training matrix was N x 2156 (N being the number of images in the entire training data set). The number of classes was 8 (different gestures). The training data was created with 5 subjects. The number of training and testing data frames for each of the gestures are given below.

The confusion matrix (Table II) shows the results of our experiment.

TABLE I. NUMBER OF FRAMES FOR EACH ACTIVITY

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

III. CONLUSIONS

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] A. Madabhushi and J. K. Aggarwal, "Using head movement to recognize activity", Proceedings of 15th International Conference on Pattern Recognition, vol. 4, pp. 698 – 701, 2000.

[2] A. Ali and J. K. Aggarwal, "Segmentation and recognition of continuous human activity", Proceedings of IEEE Workshop on Detection and Recognition of Events in Video, pp. 28 – 35, 2001.

[3] Y. Rui and P. Anandam, "Segmenting visual actions based on spatio – temporal motion patterns", Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 111 – 118, 2000.

[4] Md. Z. Uddin, N. D. Thang, T. S. Kim, "Human activity recognition via 3 – D joint angle features and Hidden Markov Models", International Conference on Image Processing, pp. 713 – 716, 2010.

[5] Y. Zhu and K. Fujimura, "A Bayesian Network for human body pose tracking from depth image sequences", Sensors, vol. 10, issue 5, 5280-5293, 2010.

[6] P. Breuer, C. Eckes and S. Müller, "Hand gesture recognition with a novel time - of – flight camera – a pilot study", MIRAGE - MIRAGE , pp. 247-260, 2007

[7] T.W. Ridler, S. Calvard, "Picture thresholding using an iterative selection method", IEEE Trans. System, Man and Cybernetics, SMC-8 ,pp. 630-632, 1978.

TABLE II. CONFUSION MATRIX FOR FRAMES

Gestures	**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