2012 Third International Conference on Emerging Applications of Information Technology (EAIT)

Gesture Classification with Machine Learning using Kinect Sensor Data

Sambit Bhattacharya, Bogdan Czejdo and Nicolas Perez
Department of Mathematics and Computer Science
Fayetteville State University
Fayetteville, NC, USA
{sbhattac, bczejdo, nperez3}@uncfsu.edu

***Abstract*— We present approaches for gesture classification and gesture segmentation by using machine learning on the Kinect sensor's data stream. Our work involved three phases. Firstly we developed gesture classification from a known vocabulary of gestures in an edited data stream. Secondly we extended those techniques to detect and classify a gesture in an unedited stream which also captures random movements. Thirdly, we apply rules to filter out movements that were not intentional gestures and yet resembled certain gestures in our vocabulary.**

Keywords- human gesture; machine learning; gesture classification; gesture recognition; gesture segmentation; Kinect sensor

I. INTRODUCTION

Recognition and classification of human gestures is an active area of research since it is an important component of human-machine interaction. Most existing approaches use low level image processing operators to extract salient features from the moving human body in video data, which are then used to train machine learning algorithms [1,2]. Such feature extraction techniques pose a significant challenge and work under certain restrictive assumptions such as lighting conditions and even the hardware design of the video capturing device [1]. Gesture segmentation also remains a problem with multiple approaches described in the literature [2]. The Microsoft Kinect offers a hardware and software platform which solves some key computer vision problems with high levels of accuracy and well understood error characteristics and limitations [3]. In this paper we present our approach in solving the problem of gesture recognition by using machine learning on the Kinect's data stream of locations of joints on a human body, in other words skeletal data. Our approach consists of three parts. Firstly we develop gesture classification from a known vocabulary of gestures in an edited data stream. Secondly we extend those techniques to detect and classify a deliberate gesture in an unedited stream which also captures random movements. Thirdly, we apply rules to filter out movements that were not intentional gestures and yet resembled certain gestures in our vocabulary; in other words we developed techniques for both the gesture segmentation and classification problem in an unedited stream.

This work was partially supported by National Science Foundation (award id: 0959958) and an internal grant award of Fayetteville State University's Integrated STEM Academic Success (ISAS) program.

II. THE KINECT JOINT DATA STREAM AND GESTURE VOCABULARY

A. The Kinect Sensor and its Data Streams

Here we describe in brief the main functions of Kinect sensor. We next describe how the Kinect data stream is read and how we modify the data organization for processing purposes.

The Kinect sensor can measure depth data (estimates of distance from Kinect to pixels in scenery) and it can identify and locate human skeleton. This skeleton is identified through positions of skeletal joints computed by Kinect from depth data. The Kinect processes and sends the skeletal data along with the depth data at the rate of 30 frames a second. It provides data streams for skeletal joints of up to two humans.

The precision of Kinect measurements and distortions was extensively studied in [3]. This experimental study reports that the random error of depth measurements increases quadratically with increasing distance from the sensor and the depth resolution also decreases quadratically with increasing distance from the sensor. It also measured the error bounds and resolutions limits and recommends that for mapping applications the data should be acquired within 1–3 m distance to the sensor. Our experiments were performed with sufficient variations within these limitations.

The Kinect sensor sends skeletal data to the computer as a temporal sequence of X, Y, Z coordinates of all 20 tracked joints. The data is grouped into time-stamped frames and the i^{th} frame can be represented as $C_i^{1x}\ C_i^{1y}\ C_i^{1z} \ldots C_i^{20x}\ C_i^{20y}\ C_i^{20z}$, where C is a single coordinate value and the super-script has joint number followed by axis letter. When recording a data stream that starts at the 0^{th} frame for which n+1 frames have been seen so far, we organize the values as $C_0^{1x}\ C_1^{1x}\ C_2^{1x} \ldots C_n^{1y}\ C_0^{1y}\ C_1^{1y}\ C_2^{1y} \ldots C_n^{1y}\ C_0^{1z}\ C_1^{1z}\ C_2^{1z} \ldots C_n^{1z}\ C_0^{2x}\ C_1^{2x}\ C_2^{2x} \ldots C_n^{2x} \ldots$ *and so on …* We can further abbreviate this notation to $C_{0,n}^{1x}\ C_{0,n}^{1y}\ C_{0,n}^{1z}\ C_{0,n}^{2x}\ C_{0,n}^{2y}\ C_{0,n}^{2z} \ldots\ C_{0,n}^{20x}\ C_{0,n}^{20y}\ C_{0,n}^{20z}$ where the subscript *0, n* means the sequence starting at *0* and ending at *n*. We additionally perform a data scaling step as recommended in [ref] for each C along the X, Y and Z directions for every joint as $C_{scaled} = (C_{original} - min) / (max - min)$ where max and min are the maximum and minimum values of that particular feature. For running the machine learning programs described next, we consider this sequence as a single feature vector where