WIIMOTE GESTURE RECOGNITION

Jozef Mlích
Doctoral Degree Programme, DCGM, FIT BUT
E-mail: imlich@fit.vutbr.cz

Supervised by: Benedikt Hörnler
E-mail: hbe@mmk.ei.tum.de

ABSTRACT

This paper presents a method of a gesture recognition with Wiimote controller. The presented approach includes topic of data acquisition, their filtering and further explotation. Additionally, construction of data processing pipeline based on Hidden Markov Models is presented. Finally, the evaluation of gesture recognition is discussed.

1 INTRODUCTION

Nowadays, there are a many research groups working on design of advanced user interfaces. The specification of such kind of user interfaces is not well defined, though. One of very common concepts assumes, that the interface should be controlled by movement of hands. This hand gesture recognition task in video sequences is very complex and includes image segmentation, object tracking and recognition it self.

The integration of such interfaces into various computer systems require cheap, very accurate black box with small consumption of system resources. Wiimote is wide available device which allow to create advanced user interfaces, as shown Johnny Chung Lee, et. al. in [1].

Accelerometer based gesture recognition is known approach and was discussed by Hofmann, et. al. [5]. The gesture recognition on Wiimote controller is also known task and it was discussed by Schlömer, et. al. [6]. This paper introduces similar approach which tries to improve accuracy of gesture recognition using more features for classification and different design of recognition pipeline.

2 DATA EXPLOTATION

Wiimote controller can provide information about its position, including distance from computer. These information is provided by embedded IR camera. However, this requires more complex workplace setup and disallows use gesture recognition in general cases, therefore in this paper only analysis of the data from accelerometers is discussed.

Data retrieved from accelerometer sensor in a Wiimote controller are defined as timed triplet $(a_x, a_y, a_z)_t$, which represent device acceleration in axes X, Y and Z Data are streamed with sampling rate about 80 Hz. For experimental reasons, 8 gestures were defined: *circle*, *down*, *left*, *right*, *shake*, *shake to side*, *square* and *up*. Examples of the filtered data received from a Wiimote controller for selected gestures are shown in the Figure 1.

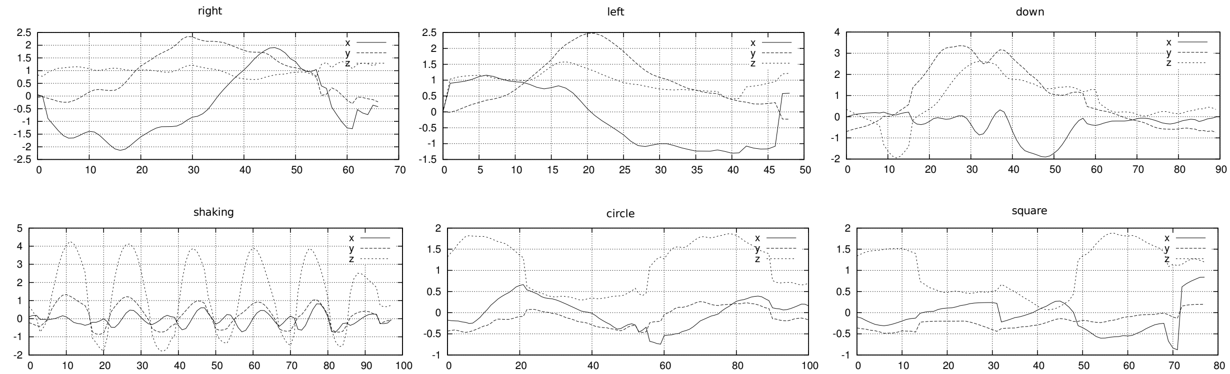

Figure 1: Output of acceleromaters for different gestures

For the recognition purposes three features were proposed. First, a temporal derivation of acceleration $da_x = \frac{a_x}{\delta t}$. The second, axes differences $(DXY, DYZ, DZX) = (a_x - a_y, a_y - a_z, a_z - a_x)$. And third, size of a vector $|a| = \sqrt{a_x^2 + a_y^2 + a_z^2}$. The resulting feature vector therfore contains this features: $(a_x, a_y, a_z, |a|, da_x, da_y, da_z, DXY, DYZ, DZX)_t$.

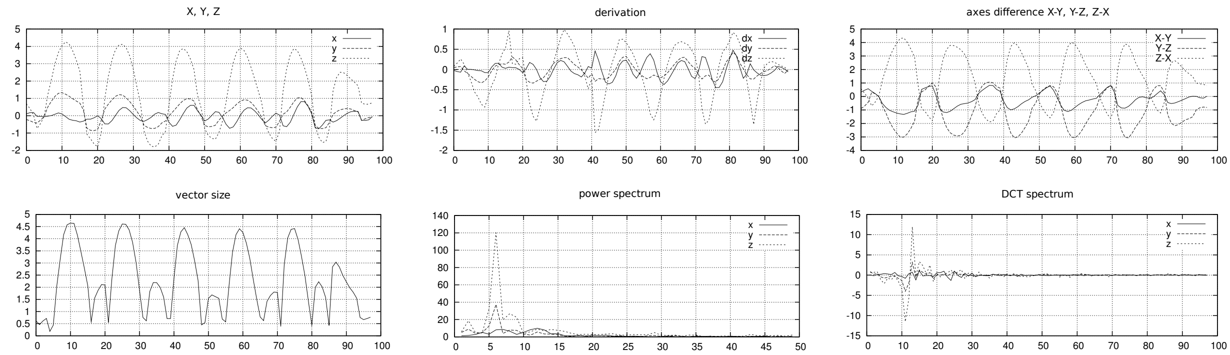

Figure 2: Features for gesture *shaking*

The features are chosen ad-hoc with purpose to have significantly different values for different gestures. For better understanding of data and feature selection the power spectrum and dft spectrum was made. The features for gesture *shaking* and its power spectrum and dft spectrum are shown in Figure 2.

3 CONCEPT

Design of processing pipeline is critical for gesture recognition system. The performances of two different approaches were explored. Both of them shares the same modules. First module obtains the data from Wiimote controller. Next, are data filtered by two simple filters. The *Hidden Markov Models* module computes likelihood of analyzed gesture. Finally, the decision about gesture with *argmax* module is done.

The sketch of basic processing pipeline is shown in Figure 3. The first approach analyses a set of overlapped windows with static size as a single gesture. The second approach tries to detect gesture boundaries and to analyze the data vector all at once.

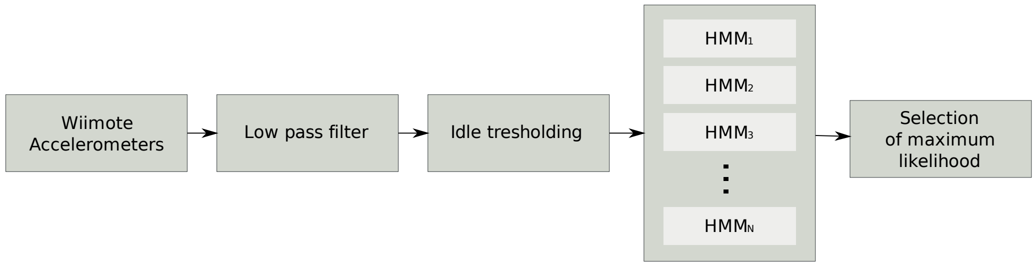

Figure 3: Processing pipeline for data with dynamic length

There are a many APIs which allows to control and retrieve data from Wiimote controller [3]. The library WiiYourself! [2] met all requirements and was used for purposes of this work.

Before classification, data are preprocessed by low pass filter, which is commonly used for noise removal. It was chosen after explotation of power spectrum. The data obtained from Wiimote controler was containing gestures and high freqency noise. The data are then processed by idle thresholding filter, which removes samples with low importance.

The thresholding is applied on absolute value of acceleration vector size $|a|$ decreased by gravity acceleration g (See Equation 1). The threshold value was set to $t = 0.2$.

$$||a| - g| < t \tag{1}$$

A common sense suggests that a gesture has dynamic length. Recognition of this kind of patterns, *Hidden Markov Models* (HMMs) [4] are frequently used. One HMM model represents a single gesture.

The recognition in streamed data (i.e. first approach) could be realized by analyzing recently retrieved block of samples. By shifting of this window in time could be analyzed whole data stream. Each incoming block is processed by HMMs corresponding to all gestures and the output likelihoods are passed to decision module. The gesture class is selected according to maximal likelihood in all windows.

In second case, the size of analyzed window is determined by idle thresholding module or by button on the Wiimote controller. The drawback of this approach is that it cannot be extended to recognition of overlapping gestures, but it provides very realible results and it is easy to integrate.

4 EVALUATION

There are two assumptions for approach with static overlapping windows. First, the gesture have approximately same size like a window. Second, the performed gesture should have significantly higher likelihood than noise or part of neighbor gesture. The problem with evaluation of this approach is in annotation. It is not easy to annotate real data, because human visual recognition of gesture is too much difficult. Artificial data set will not contain idle parts of stream between gestures. For valuable evaluation is necessary to create specialized data set considering all these problems which is expansive, therefore was evaluation of first approach skipped.

For the evaluation of recognition with dynamic sized non overlapping windows a training and testing sets were created. Both, the training and testing set contains about 50 examples for each gesture. The average length of gesture was 54 samples. The dataset was created by single person in lab environment. Tests of recognition performance with other persons shows that the training set is sufficient for the task. For evaluation single calibrated Wiimote device was used. But the recognition will also work on other devices.

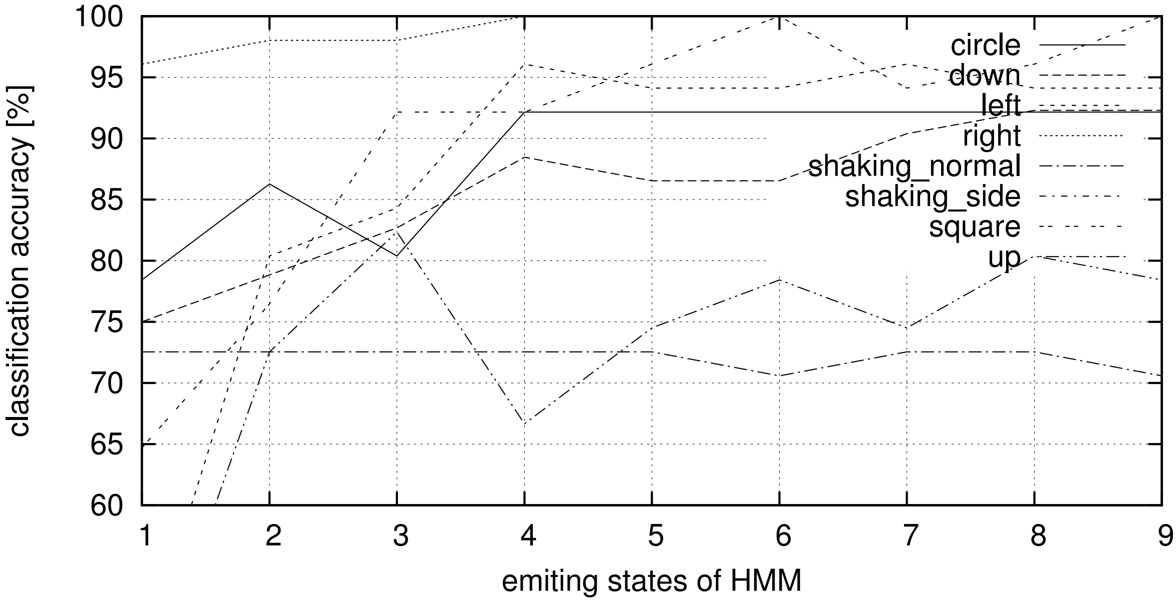

Figure 4: Accuracy of classification for given gestures with N-states HMM

Training of gestures was done with Baum-Welch estimation (HTK toolkit [7]) with different size of hmm initial transition matrix (single Gaussian mixture for each state). For the recognition, Viterbi algorithm is used.

The evaluation was focusing on accuracy of gesture recognition which is defined as a ratio of correctly recognized gestures to all gestures. Detailed overview of recognition accuracy with different model sizes is shown in the Figure 4. The most interesting part of graph is shown in Table 1.

states	circle	down	left	right	shake	shake to side	square	up
3	80.39	82.69	84.31	98.03	72.54	100.00	92.15	82.35
4	92.15	88.46	96.07	100.00	72.54	100.00	92.15	66.66

Table 1: Accuracy of classification for given gestures with N-states HMM

5 CONCLUSIONS

The proposed gesture recognition system seems to be ready for an interactive industrial environment. The accuracy of recognition is oscillating around 85 % depending on gesture type. That is in agreement with theoretical predictions. Similar results were reported by Schlömer, et. al. [6]. The resource consumption is immeasurable on Windows XP SP3, Intel Pentium 4, 2.8GHz, 2 GB RAM.

Interesting areas of further research are recognition of overlapping gestures, interaction of multiple devices, compound gestures, finding the best HMM configuration (number of Gaussian Mixtures per state), experiments with discriminative training of HMM, experiments with large data sets and gesture recognition on video based or 3D data.

ACKNOWLEDGEMENTS

This work has been supported by AMIDA Augmented Multi-party Interaction with Distance Access, EU-6FP-IST, IST-033812-AMIDA.

REFERENCES

[1] *Hacking the Nintendo Wii Remote*, volume 7, Piscataway, NJ, USA, 2008. IEEE Educational Activities Department.

[2] Wiiyourself! library. http://wiiyourself.gl.tter.org/, 2008.

[3] Wiili.org. http://www.wiili.org/, 2009.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] Frank G. Hofmann, Peter Heyer, and Günter Hommel. Velocity profile based recognition of dynamic gestures with discrete hidden markov models. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*, pages 81–95, London, UK, 1998. Springer-Verlag.

[6] Thomas Schlömer, Benjamin Poppinga, Niels Henze, and Susanne Boll. Gesture recognition with a wii controller. In *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages 11–14, New York, NY, USA, 2008. ACM.

[7] S. Young, J. Odell, D. Ollason, P. Woodland, and J. Jansen. *The HTK Book*. Machine Intelligence Laboratory, Cambridge University Engineering Department, 2006.