Gyroscope-Based Conducting Gesture Recognition

Andreas Höfer, Aristotelis Hadjakos, Max Mühlhäuser
Department of Computer Science, Technische Universität Darmstadt
Hochschulstraße 10
64289 Darmstadt, Germany
hoefer@rbg.informatik.tu-darmstadt.de, telis, max@tk.informatik.tu-darmstadt.de

Abstract
This paper describes a method for classification of different beat gestures within traditional beat patterns based on gyroscope data and machine learning techniques and provides a quantitative evaluation.

Figure 1. The user holds the gyro mouse in her right hand while conducting.

1. Introduction

Traditional beat patterns are used by conductors to indicate the beat to the musical ensemble. A beat pattern describes one cycle of the conductor's motion, which represents a musical time unit like a measure.

Automatic conducting gesture recognition is needed for conducting teaching applications and interactive conducting systems, like the Personal Orchestra system [1]. Recent conducting gesture recognition methods have been based on visual tracking [2, 3], magnetic tracking [4], accelerometer signals [5] etc. Gyroscopes are well-suited for interactive conducting systems because of their high accuracy, low latency and low cost. Conducting gesture recognition methods based on gyroscope signals have been studied [6, 7], but to our best knowledge a quantitative evaluation of a gyroscope-based method has not yet been carried out.

The recognition problem that we want to solve is classification of different beat gestures within traditional beat patterns. For example, in the 3/4 beat pattern there are three beat gesture classes to distinguish: (1) the motion from beat one to beat two, (2) the motion from beat two to beat three and (3) the motion from beat three to beat one. Such a classification method would allow an interactive conducting system to synchronize the conducting and the audio playback on a per measure level (additionally to beat tracking).

2. Related Work

There has been a wealth of research on conducting recognition systems. For an extensive overview refer to [2]. In this section we provide a short overview of gyroscope-based conducting recognition systems.

Dillon et al. use input from a 2D gyro mouse to recognize beat patterns [6]. Their method is based on modeling beat patterns as sequences of impulses, which are sudden changes of direction in the baton trajectory. Dillon et al. have evaluated their method in a user study with children. However, they provide no quantitative data about the achieved recognition rates.

The Gesture Follower by Bevilacqua et al. can synchronize a performed gesture to a recorded gesture [7]. It was used to synchronize conducting movements based on a 2D gyroscope and a 3D accelerometer signal. The system is not trained for multiple users, which limits its applicability for an interactive conducting system.

3. Conducting Gesture Classification

As input device we use a Gyration cordless mouse with integrated 2D gyroscope (Figure 1). When the gyro mouse is moved in 3D space, the gyroscope measures the rotational velocities around the mouse's horizontal axis (up-down signal) and vertical axis (left-right signal).

The stream of pairs of rotational velocities is split into data segments, which represent single beat gestures. Each gyro data segment is mapped to a sequence of discrete symbols. These sequences are classified using previously trained discrete hidden Markov models (HMM). HMMs have already been utilized for accelerometer- and vision-based conducting recognition systems [5, 2].

3.1. Computation of Symbol Sequences

The vertical minima in the mouse trajectory, which correspond to the beats in the beat pattern, are found in the up-down gyroscope signal. The stream of 2D rotational velocities is split at the computed times into segments, which represent single beat gestures.

After the data stream is segmented, the length per gesture data segment is normalized by downsampling to 15 elements per segment. A data point of the normalized sequence is computed by linear interpolation from the two nearest neighbors in the original sequence.

To map the data points on discrete symbols, two different methods are used interchangeably. The first method has been adapted from a similar approach by Kolesnik [2]. For a given 2D data vector the phase of its polar coordinate representation is computed and quantized in the appropriate number of levels. The second method is vector quantization. In a precomputation step, a codebook is obtained by executing a k-means clustering algorithm on sample data and using the cluster centers as codebook elements. Each codebook element represents a discrete symbol. The actual quantization is done by computing the nearest codebook element for each data vector.

The sequences of two consecutive beat gestures are concatenated and classified as a whole, similar to a method by Ilmonen and Takala [4]. As some beat gesture classes are very similar, e. g., the motion from beat one to two and the motion from beat three to four in the 4/4 beat pattern, a concatenation of sequences simplifies the recognition task so that higher recognition rates are achieved.

3.2. HMM Classification

The main classification step is done with discrete HMMs. A HMM is a stochastic model, especially suited for sequential data like speech or gesture data [8]. In preparation, one HMM per class is trained with sample sequences. For classification, the sequence is scored with the trained HMMs by computing the likelihood of the sequence for each HMM. The HMM with the highest likelihood indicates the class of the sequence.

We use HMMs with 10 states, 20 observation symbols and a strict left-to-right topology (i. e., only transitions from each state to itself and to its direct successor are allowed). The state transition probabilities and the output symbol probabilities of each model are determined by training with the Baum-Welch algorithm for multiple input sequences. Uniform distributions are used as initial estimates for the Baum-Welch algorithm.

4. Evaluation

The necessary sample data for training and testing were acquired by recording conducting gestures of three musically educated people (former music university students of HfMDK Frankfurt) conducting with the gyro mouse. From each test person we recorded gesture data in 2/4, 3/4 and 4/4 beat pattern with tempi of 60, 80, 100 and 120 beats per minute. Furthermore, we let the test persons conduct a recording of "Serenade in G", KV 525 by W. A. Mozart. The data of the metronome sessions were used for training and testing, the data of the Serenade sessions were used for testing only. We asked the test persons to conduct in the beat patterns they had learned through their studies, which differed from student to student, and in the beat patterns by McElheran [9] to obtain gesture data with uniform beat patterns. With some sporadic exceptions both quantization methods (computation of phase and vector quantization) led to similar recognition rates during our evaluation; hence, we only report the results of the experiments in which computation of phase was used.

First, we used training and test data from the same conductor. Performing a cross-validation with the four metronome recordings from one conductor, we obtained a average recognition rate of 99.62%. Using the metronome data as training data and the Serenade data as test data, we obtained a average recognition rate of 98.74%.

Second, we performed across-user experiments, where we trained the models with data from two conductors and used the data from the third as test data (all three conducting in their individual beat patterns). This resulted in average recognition rates of 95.67% in case of metronome data as test data and 90.04% in case of Serenade data as test data.

Finally, we repeated the across-user experiments, using the gesture data in which uniform McElheran beat patterns (instead of individual beat patterns) were used. This led to average recognition rates of 99.40% for metronome data and 98.88% for Serenade data.

References

[1] J. O. Borchers, E. Lee, W. Samminger and M. Mühlhäuser, "Personal Orchestra: A Real-Time Audio/Video System for Interactive Conducting," *Multimedia Systems*, vol. 9, no. 5, pp. 458–465, 2004.

[2] P. Kolesnik, *Conducting Gesture Recognition, Analysis and Performance System*, Master's thesis, McGill University, Montreal, 2004.

[3] E. Lee, I. Grüll, H. Kiel and J. O. Borchers, "conga: A Framework for Adaptive Conducting Gesture Analysis," *NIME*, 2006, pp. 260–265.

[4] T. Ilmonen and T. Takala, "Conductor Following With Artificial Neural Networks," *ICMC*, 1999, pp. 367–370.

[5] S. Usa and Y. Mochida, "A Multi-Modal Conducting Simulator," *ICMC*, 1998, pp. 25–32.

[6] R. Dillon, G. Wong and R. Ang, "Virtual Orchestra: An Immersive Computer Game for Fun and Education," *Proc. of the Conf. on Game Research and Development (CyberGames)*, 2006, pp. 215–218.

[7] F. Bevilacqua, F. Guédy, N. Schnell, E. Fléty and N. Leroy, "Wireless Sensor Interface and Gesture-Follower for Music Pedagogy," *NIME*, 2007, pp. 124–129.

[8] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[9] B. McElheran, *Conducting Technique: For Beginners and Professionals*, 3rd ed., Oxford University Press, 2004.