Air Violin: A Machine Learning Approach to Fingering Gesture Recognition

David Dalmazzo
Music and Machine Learning Lab, Music Technology Group, Universitat Pompeu Fabra
Barcelona, Spain
david.cabrera@upf.edu

Rafael Ramirez
Music and Machine Learning Lab, Music Technology Group, Universitat Pompeu Fabra
Barcelona, Spain
rafael.ramirez@upf.edu

ABSTRACT

We train and evaluate two machine learning models for predicting fingering in violin performances using motion and EMG sensors integrated in the *Myo* device. Our aim is twofold: first, provide a fingering recognition model in the context of a gamification virtual violin application where we measure both right hand (i.e. bow) and left hand (i.e. fingering) gestures, and second, implement a tracking system for a computer assisted pedagogical tool for self-regulated learners in high-level music education. Our approach is based on the principle of *mapping-by-demonstration* in which the model is trained by the performer. We evaluated a model based on Decision Trees and compared it with a Hidden Markovian Model.

CCS CONCEPTS

• **Applied computing** → **Sound and music computing**; Interactive learning environments; • **Computing methodologies** → *Instance-based learning*;

KEYWORDS

Gestures, Machine Learning, Hand Tracking, HMM, Gamification, Violin, Music education

ACM Reference Format:
David Dalmazzo and Rafael Ramirez. 2017. Air Violin: A Machine Learning Approach to Fingering Gesture Recognition. In *Proceedings of 1st ACM SIGCHI International Workshop on Multimodal Interaction for Education (MIE'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3139513.3139526

1 INTRODUCTION

Acquire high-level musical skills in a professional context requires a strong dedication and practice. It is a task mostly based on the master-apprentice relation, which is typically based on some hours per week supervised-correction. Therefore, bad habit development on students is a common issue in high-level musical education, where a solution is needed especially on self-practice focused periods to reduce the high percentage of students facing overuse-syndrome, stressfulness or frustration. Although motion capture technologies have been widely used in sports science, detecting and correcting musical gestures in real-time has been much less explored. The automatic detection and analysis of music gestures in a computer assistant system would be of great use for enhancing music instrument learning. The aim of this work is to apply low-cost sensor technology to the automatic detection and analysis of gestures related to music performances. Taking the violin as a case study, we measure both right hand (i.e. bow) and left hand (i.e. fingering) gestures. Our aim is to detect bow velocity and fingering by using off-the-shelf EMG and motion capture devices such as the Myo and Riot sensors positioned in the forearms of the musician. In this paper, we describe two models based on machine learning to detect left-hand finger selection and performance in real-time. Furthermore, our approach has two lines of development, 1) A virtual Violin as a gamification, which pretend to give feedback to novel users about musical fundamentals related to gestures. 2) A tracking system to record experts performers then trains a model which will have a good-practice scheme, to then compare students performance and give in real-time feedback about how far from the *good-practice* model they are. To develop such a system, we started tracking left-hand finger performance by applying and comparing two common Machine Learning models described in the literature, which are Decision Trees using the implementation *Weka*. The second ML model is a Hidden Hierarchical Markovian Model implemented using *Mubu* external objects [9] previously developed by Ircam.

2 RELATED WORK

Motion tracking systems based on sensors and machine learning techniques for gesture classification and regression has been widely covered, where Hidden Markov Models (HMM) represents one of the most used methods. Matthew Brand et al. (1997) [2] proposed a time-flexible classifier of multidimensional data captured using computer vision to classify hand gestures performance over time windows. Wilson and Bobick (2000) proposed an online HMM model as an adaptive algorithm for learning hand gestures [10] to perform *"air drumming-sounds"* mapping those gestures. In terms of real-time classification and regression, Chad Peiper et. al (2003) [7] proposed a software for detection of violin bow-strokes articulations, such as *Détaché*, *Martelé*, *Spiccato* and *Staccato* being performed on a special tracking-system room based on cameras. Kolesnik and Wanderley (2005) implemented a Max external object

MIE'17, November 13, 2017, Glasgow, UK

ACM ISBN 978-1-4503-5557-5/17/11…$15.00
https://doi.org/10.1145/3139513.3139526

that uses low-cost USB cameras to track hand motion and determine gestures applying HMM [6]. Several approaches to gesture recognition use the models to trigger events or select sounds but do not assess more precise time-related gesture expression. Bevilacqua et al. (2009) proposed a model where continues recognition follows the gesture performance, giving a state of the intrinsic time-windows steps as likelihood values of the gesture [1]. Françoise et al. (2014) presented a set of probabilistic models which compare algorithms for gesture recognition such as Gaussian Mixture Models, Gaussian Mixture Regression and HMM implemented in Max to create a gesture-based sound mixing where motion and sounds are learned from *direct-demonstration*. Fiebrink on 2010 presented Wekinator [3], a software focused on implementation of machine learning algorithms to classify and recognise patterns where sound and gesture mapping is commonly used for electronic music performers and artists. Fried and Fiebrink (2013) [5] presented a method based on Deep Learning to automatically extract gestural features and map motion vectors to sound representations with a cross-modal mapping that allows gesture-to-speech, gesture-to-visualization and sound-to-haptic mappings. In this paper, we apply machine learning techniques from *Wekinator* and HMM presented by Françoise et al. (2014) [4] to implement Max externals to compare and track musician's hand gestures using the Myo and Riot sensors especially focused on finger tracking recognition for an Air Violin application. Hence, this study is pointing to the idea of developing a framework that estimates finger gesture performance on specific musical pieces to provide precise feedback to the students practising those pieces in real-time. As well, develop a virtual violin that generates sound based on gestures which can be used as an extension for musical interactive performances adding this framework to the real violin interpretation or a gamification on learning music by practising predefined levels.

3 MATERIALS AND METHODS

We recorded the 8 EMG sensors signals and the orientation quaternion from the Myo device placed on the left forearm. To filter the raw EMG data we applied a Bayesian filtering [8] to estimate the activity of muscles. We capture the accelerometer and gyroscope from the Riot sensor placed in the right hand. The Myo raw signal is captured by a C++ application that sends the filtered data to *Wekinator* application and the signal is captured and recorded in Max buffer using *Mubu* external objects. Based on the *mapping-by-demonstration* principle [4], the players have to record the finger gestures every time they use the framework. In DT case, each finger is recorded in a range of 5s. In the HMM case, each finger gesture is recorded in 500ms time-window (no overlapping the recordings). The training set in both cases takes around 1s, to then run the model to estimate which finger is performed. Hence, the interaction procedure is based on two phases: a) The recording phase where the users record single finger gestures performed over a surface or in the violin's arm. First, pressing one violin string with the index finger (f1), middle finger (f2), ring finger (f3) and little finger (f4) (reference in figure 3). For DT we recorded 1000 samples per finger performing the gesture during 500 ms, having a total of 5 gestures (f1, f2, f3, f4, non-gesture). b) In the training phase, we apply two machine learning models to predict which finger is being performed by the musician in real-time. In a MacBookPro mid-2009, the precise estimation takes around 200-250ms, however, in a newer computer, this time is reduced to 100ms.

3.1 Motion Sensors

Myo device is composed of 3Dof Gyroscope, 3Dof Accelerometer, orientation quaternion which can be translated to a 3Dof Euler-Angular position, and 8 Electromyogram. We use as well, one Riot sensor for the right hand, which has 3Dof Accelerometer, 3Dof Gyroscope, 3Dof Magnetometer to map the gestures of the right hand into the sound synthesis generation.

3.2 Models

The model induced by the decision tree algorithm (as implemented in the Wekinator application) outputs a real-time fluctuating number describing the finger being performed having a range of 5 values: from 1 to 4 for each finger (see fig.1) and 5 as no-finger gesture. Using a Modulus operator with a stream of 15 values we normalise and stabilise the output applying as well a histogram within a range of 250 ms precision.

We train an HMM using the recording of the 8 EMG sensors plus the orientation quaternion for gestural position reference of the arm, during 1-second per gesture plus non-finger gesture. As an output, the HMM model continuously estimates the intrinsic time-window steps (20 states) of each finger gesture. That means, in a range of 1s, the model has 20 internal states of Gaussian probabilistic distribution (50ms) of the time progression within the gesture, giving a correspondent value per gesture called *likelihood* (Fig 4). It reports the time progression of the gesture within the 20 states of the original training phrases. As well, a normalised *likelihood* reports which are the predominant gesture being recognised.

3.3 Implementation

The *Air Violin* application is programmed on Max. Max is a visual programming language for music, video and graphics, commonly used for interactive creative applications development for music or interactive media. It is suitable for quick prototyping. We have implemented an external library called *Mubu* (multi-buffer), which is a container of multiple objects for sound and motion analysis, developed at Ircam [9]. With the help of Myo-SDK, a custom application made in C++ to read Myo signals, Max and Wekinator application, we developed a first model of an interactive music instrument as a virtual violin that gives musical feedback of gestural performance to novice users, by giving sound output directly mapped from motion data, as well learning from user gestures to adapt the sound manipulation. It is known that little differences on gestures may produce big differences in performed sound. Therefore, this interactive model can be applied as an ecological validation model on learning and performing musical instruments, giving both sonic and haptic feedback to the users on how they are performing.

4 RESULTS

Decision trees produced a correctly classified instances percentage of 87.74% with a rate of incorrect classified instances of 12.26%. The lower accuracy for detecting finger-gestures was for the index

finger (f1) (see Figure 1) giving an f-measure of 0.76. The real-time accuracy on the Max implementation is 82.77%.

Precision	Recall	F-Measure	Class
0,818	0,797	0,760	1
0,854	0,824	0,799	2
0,938	0,896	0,897	3
0,943	0,918	0,914	4
0,840	0,952	0,866	5
0,879	0,877	0,847	Weighted Avg

Figure 1: Precision, Recall and F-Measure of the five gestures recorded, recalling the value in percentage of the correct estimations over all related-gesture samples; F-Measure is expressed as the *harmonic mean* between precision and recall, expressing the average probability of correct estimations. The class column is the gesture performed, where 5 correspond to non-gesture, and the numbers from 1 to 4 corresponds to the finger performed.

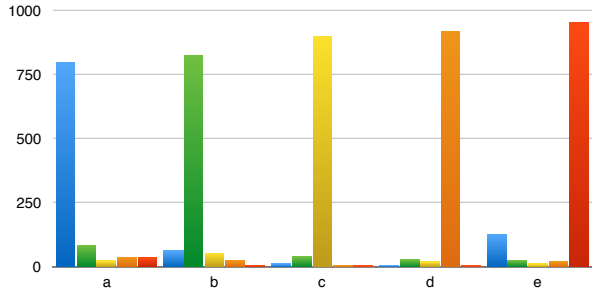

Figure 2: Graph representing the confusion matrix produced by the decision trees algorithm. Correct Predictions per each single gesture recorded

In the HMM model, we measure 100 samples with a time-window of 1.5s per gesture and the average accuracy of each gesture is f1 = 97.1%, f2 = 98.9%, f3 = 94.5%, f4 = 95.1%, f5 = 99.3%. With an average normalised likelihood of f1 = 0.966, f2 = 0.935, f3 = 0.894, f4 = 0.927, no-finger = 0.99. To measure how good the model estimates the finger gesture, we recorded 100 samples per gesture. Using a metronome with 120 bpm, the gesture was performed pressing each finger and non-gesture over each quarter note, recording three times the exercise to fulfil more than 100 samples per gesture.

5 CONCLUSION AND FUTURE WORK

We presented two approaches to predict left-hand finger gestures for two purposes: a) Gamification of an interactive musical tool for novice musical learners b) An off-line probabilistic measurement model to estimate finger disposition on a musical peace based on

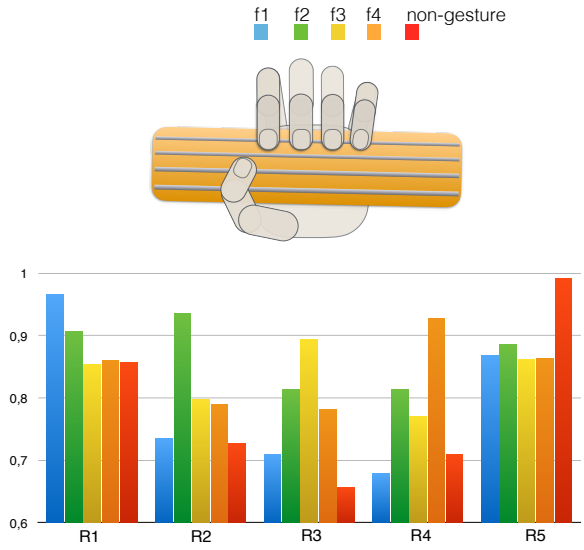

Figure 3: Left-hand setup (top): f1: Index finger, f2: Middle finger, f3: Ring finger, f4: Little finger, 5: no-finger. HMM Regression (bottom): A probabilistic model recognition for each gesture phase. R1 corresponds to recognition of gesture f1, R2 corresponds to f2, and so on. The graph shows a probability estimation of the HMM model after recording 100 samples per gesture. The probability graph shows that the difference is small among finger-gestures, however, the model is highly accurate on estimating the correct finger performed

f1	f2	f3	f4	non-gesture
0,9662	0,9067	0,8539	0,8595	0,8573
0,7347	0,9357	0,7977	0,7892	0,7277
0,7090	0,8133	0,8944	0,7812	0,6572
0,6783	0,8136	0,7707	0,9276	0,7089
0,8675	0,8849	0,8619	0,8630	0,9923

Figure 4: Likelihood as a normalised value describing probability of the finger performed. Likelihood value is useful in real-time scenarios, as it describes the internal time-window state of the gesture, it is suitable for more accurate motion over time analysis.

EMG sensor information, hence, give feedback in real-time to self-practice learners. This system constitutes part of the main idea of developing a computer-based motion self-learning assistant. Tracking both left and right-hand muscular activity gives insights on how performers are executing fine gestural movements. Having precise models of "*good practices*" may provide useful real-time feedback

to learners on their deviation from a target performance. The precision of finger gesture estimation using decision trees has a high accuracy in time intervals larger than 200-250ms and drops down in time intervals smaller than 200ms. Accuracy detecting f1 and f2 is lower than for f3 and f4, perhaps due to the sensor disposition of the EMG in the Myo device. The alternative positioning of the MYO device must be further explored. On the other hand, HMM seems to be a suitable model, especially for more gestural-expressive mappings, where the intrinsic timing of the gesture is estimated as well. As a clarification, this test was executed on a computer assembled on 2009 with a Core2Duo processor, with newer laptops, those 200-250ms is reduced to 100ms, enhancing the real-time possibilities of the framework. The results show a high accuracy using Weka framework for the DT model, however, this calculation are made in cold, in the real-time scenario HMM has a better performance accuracy and gestural description over time than DT which gives, as a result, an output that needs a range of 200ms time-window to give the new state of the gesture. However, this first results can be improved in DT as well, integrating the model into the same framework, avoiding sending and receiving values through OSC messages, and fine tuning the parameters of the algorithm. As a future work, we plan to apply deep learning models on real-time gestural analysis, with the *mapping-by-demonstration* principle.

ACKNOWLEDGEMENT

This work has been partly sponsored by the Spanish TIN project TIMUL (TIN 2013-48152-C2-2-R), the European Union Horizon 2020 research and innovation programme under grant agreement No. 688269 (TELMI project), and the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502).

REFERENCES

[1] Frédéric Bevilacqua, Bruno Zamborlin, Anthony Sypniewski, Norbert Schnell, Fabrice Guédy, and Nicolas Rasamimanana. 2009. Continuous realtime gesture following and recognition. In *International gesture workshop*. Springer, 73–84.

[2] Matthew Brand, Nuria Oliver, and Alex Pentland. 1997. Coupled hidden Markov models for complex action recognition. In *Computer vision and pattern recognition, 1997. proceedings., 1997 ieee computer society conference on*. IEEE, 994–999.

[3] Rebecca Fiebrink and Perry R Cook. 2010. The Wekinator: a system for real-time, interactive machine learning in music. In *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)(Utrecht)*.

[4] Jules Françoise, Norbert Schnell, Riccardo Borghesi, and Frédéric Bevilacqua. 2014. Probabilistic models for designing motion and sound relationships. In *Proceedings of the 2014 international conference on new interfaces for musical expression*. 287–292.

[5] Ohad Fried and Rebecca Fiebrink. 2013. Cross-modal Sound Mapping Using Deep Learning.. In *NIME*. 531–534.

[6] Paul Kolesnik and Marcelo M Wanderley. 2005. Implementation of the Discrete Hidden Markov Model in Max/MSP Environment.. In *FLAIRS Conference*. 68–73.

[7] Chad Peiper, David Warden, and Guy Garnett. 2003. An interface for real-time classification of articulations produced by violin bowing. In *Proceedings of the 2003 conference on New interfaces for musical expression*. National University of Singapore, 192–196.

[8] Terence D Sanger. 2007. Bayesian filtering of myoelectric signals. *Journal of neurophysiology* 97, 2 (2007), 1839–1845.

[9] Norbert Schnell, Axel Röbel, Diemo Schwarz, Geoffroy Peeters, Riccardo Borghesi, et al. 2009. MuBu and friends–assembling tools for content based real-time interactive audio processing in Max/MSP. In *ICMC*.

[10] Andrew D Wilson and Aaron F Bobick. 2000. Realtime online adaptive gesture recognition. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, Vol. 1. IEEE, 270–275.