THE BASIS MIXER: A COMPUTATIONAL ROMANTIC PIANIST

Carlos Eduardo Cancino Chacón **Maarten Grachten**

Austrian Research Institute for Artificial Intelligence

{carlos.cancino, maarten.grachten}@ofai.at

ABSTRACT

In this demonstration, we give an overview of the Basis Mixer, a basis modeling framework for musical expression in piano music. This framework, which was originally conceived for simple linear modeling of expressive dynamics [5], now allows for creating comprehensive computational models of expression in piano music. We focus on the representation and modeling of different aspects of expression, and demonstrate the generative capabilities of trained models through a simple web-interface.

1. INTRODUCTION

The basis modeling framework for musical expression was first proposed in [5]. This framework uses numerical encodings of musical scores (basis functions) to learn how music is performed in an expressive manner. The Basis Models (BMs) can be used for both analysis and synthesis of music performances.

Initial versions of the framework were limited to simple linear models to describe the relation between expressive dynamics and the musical score in classical piano music [5]. Subsequent versions include the use of non-linear models, like the use of feed-forward neural networks (FFNNs) to model expressive dynamics [2], and recurrent neural networks (RNNs) to model expressive tempo [4]. By now, the framework allows for modeling a *complete* representation of expressive piano performances. In addition to expressive dynamics and tempo, the framework represents articulation and timing deviations of individual notes.

In this demonstration, we give an overview of the way diverse musical score information is represented uniformly in the BM framework. We also describe how the expressive characteristics of piano performances are quantified for modeling in the framework, focusing on the combination of note-wise and onset-wise expressive parameters. We show how onset-wise parameters allow for time series modeling, while the note-wise parameters are modeled using local context models. Finally, we present the prototype for a web service that allows the user to generate, and manipulate an expressive performance of a piece of music using BM models, given a MusicXML file.

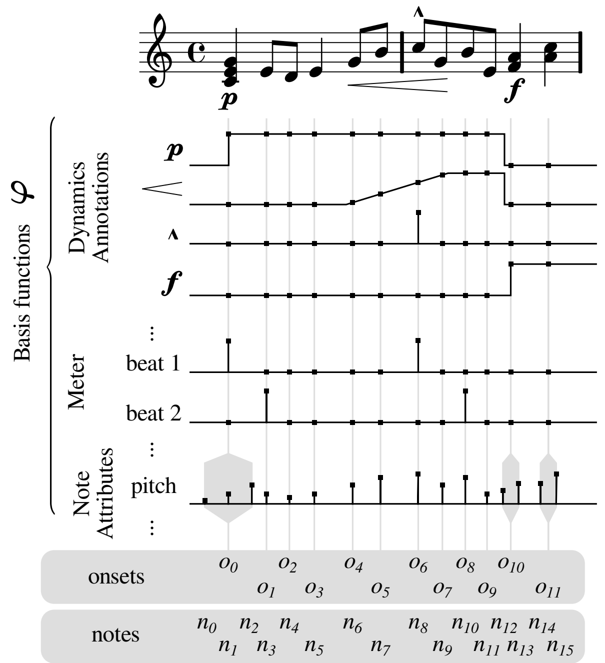

Figure 1. An example of basis functions modeling of dynamics annotations, meter and pitch.

2. SCORE REPRESENTATION

The BM framework models numerical descriptors that encode an expressive performance, referred to as *expressive parameters* $\mathbf{y}$, as a function $f(\boldsymbol{\varphi})$ of *basis functions* $\boldsymbol{\varphi}$, i.e. numerical encodings of a variety of descriptors of a musical score. In its original formulation, the BM included mostly features accounting for performance directives explicitly written in the score like pitch, dynamics markings (***ff***, ***p***) and articulation markings ($legato$ slurs). Recent work includes the inclusion of raw pianoroll information, as well as musicologically informed features, such as the harmonic tension [7], and features representing segmentation structure within the piece, based on a self-similarity matrix of the pianoroll. To the best of our knowledge, this diversity of score information is unprecedented in existing modeling approaches to musical expression.

Figure 1 illustrates schematically the numerical encoding of a musical score using the BM framework.

3. MODELING EXPRESSION IN PIANO MUSIC

In this demonstration, we describe a piano performance in terms of the following five expressive parameters, which correspond roughly to common concepts involved in musical expression.

3.1 Dynamics

Dynamics refers to the loudness of the performance. For simplicity we treat the MIDI velocity of the performed notes as a proxy for dynamics. As such, the values can be extracted per note directly from the corpus. We model dynamics using two targets:

1. **Loudness**. Average MIDI velocity for all notes performed with the same onset

2. **Loudness spreading**. Deviation of the MIDI velocity from the above temporal average for each note in the timestep

3.2 Tempo

Musical tempo is defined as the rate at which musical beats occur. Rather than the beat rate, we take the logarithm of its reciprocal, the *beat period*, as a representation of local tempo. We compute the beat periods by using a spline smoothing of the interonset intervals (IOIs). We model musical tempo using two targets:

3. **Tempo**. Computed using a spline interpolation through performed note onsets

4. **Timing**. Onset deviations of the individual notes from the spline interpolation

3.3 Articulation

Articulation refers to the lengthening or shortening of note durations in a performance, producing *legato* and *staccato* playing styles, respectively. The articulation is modeled note-wise using a single target:

5. **Articulation**. Logarithm of the ratio of the actual duration of a performed note to its reference (notated) duration according to the local IOI

The representation of a performance in terms of the above expressive parameters is lossless (up to the average beat-period), which means that an encoding of a MIDI performance in terms of these parameters allows for an exact reconstruction of the performance, given the score and the average beat period.

3.4 Regression models

Given the sequential nature of the onset-wise expressive parameters, we use dynamic models, particularly RNNs, to model loudness and tempo. Previous work on expressive timing [4] has shown that using bidirectional temporal models increases predictive accuracy. Furthermore, our current implementation allows us to use more sophisticated recurrent architectures, such as long-short term memory

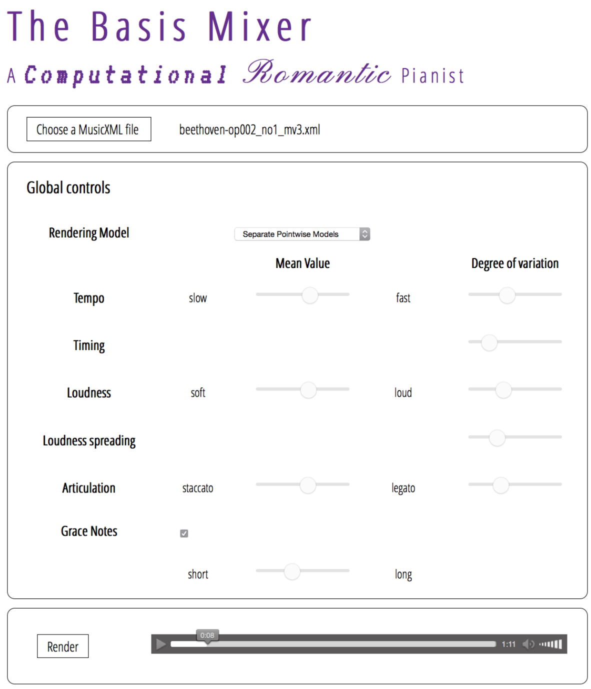

Figure 2. Prototype UI for the Web App.

networks (LSTMs). In order to jointly model these expressive parameters, we use the network outputs to parametrize a Gaussian mixture distribution, resulting in Gaussian mixture density RNNs [6].

On the other hand, to model the note-wise targets, i.e. loudness spreading, timing and articulation, we use FFNNs. To jointly model these expressive parameters we use Gaussian mixture FFNNs [1].

4. INTERACTIVE WEB INTERFACE

We present the prototype of a Web app to render expressive performances of musical pieces using the BM framework. The user can upload a score in MusicXML format and create an expressive interpretation of the score as an audio file. The user interface (UI) is shown in Figure 2.

The Web app generates a performance using models trained in a supervised fashion on romantic piano music from the late 18th and 19th centuries, including Chopin and Beethoven, performed by professional pianists, and recorded on a computer-monitored grand piano [3]. The expressive parameters of each piece are standardized, i.e. they are scaled and translated to have zero-mean and unit variance. Using the controls in the UI, the user can easily specify the mean value and standard deviation of the different expressive parameters, thereby controlling the overall expressive characteristics of the performance.

5. ACKNOWLEDGEMENTS

This work is supported by European Union Seventh Framework Programme, through the Lrn2Cre8 (FET grant agreement no. 610859).

6. REFERENCES

[1] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer Verlag, Microsoft Research Ltd., 2006.

[2] C. E. Cancino Chacón and M. Grachten. An evaluation of score descriptors combined with non-linear models of expressive dynamics in music. In Nathalie Japkowicz and Stan Matwin, editors, *Proceedings of the 18th International Conference on Discovery Science (DS 2015)*, Lecture Notes in Artificial Intelligence, Banff, Canada, 2015. Springer.

[3] Sebastian Flossmann, Werner Goebl, Maarten Grachten, Bernhard Niedermayer, and Gerhard Widmer. The Magaloff Project: An Interim Report. *Journal of New Music Research*, pages 1–24, September 2011.

[4] M. Grachten and C. E. Cancino Chacón. Temporal dependencies in the expressive timing of classical piano performances. In *Companion of Embodied Music Interaction*. Routledge, 2016. Under review.

[5] Maarten Grachten and Gerhard Widmer. Linear Basis Models for Prediction and Analysis of Musical Expression. *Journal of New Music Research*, 41(4):311–322, December 2012.

[6] Alex Graves. Generating Sequences With Recurrent Neural Networks. *arXiv*, 1308:850, 2013.

[7] Dorien Herremans and Elaine Chew. Tension ribbons: Quantifying and visualising tonal tension. In *Proceedings of the Second International Conference on Technologies for Music Notation and Representation TENOR*, pages 8–18, Cambridge, UK, May 2016.