19th INTERNATIONAL CONGRESS ON ACOUSTICS
MADRID, 2-7 SEPTEMBER 2007

Combining Performance Actions with Spectral Models for Violin Sound Transformation

PACS: 43.66.Jh

Perez, Alfonso; Bonada, Jordi; Maestre, Esteban; Guaus, Enric; Blaauw, Merlijn
Music Technology Group; Ocata 1, Barcelona, Spain;
{aperez, jbonada, emaestre, eguaus, mblaauw}@iua.upf.edu

ABSTRACT
In this work we present a violin timbre model that takes into account performance gestures. It is built by analysis of performance data using machine learning methods and it is able to predict the timbre given a set of performance actions. Gestural data and sound are synchronously captured by means of 3D motion trackers attached to the instrument and a bridge pickup. The model is used for sample transformation within a spectral concatenative synthesizer informed by gestures.

INTRODUCTION

Spectral concatenative synthesis models [1], [2] generate sound by concatenation of spectrally transformed samples. Sample concatenation is crucial for the quality of sound produced, and sometimes transitions between two samples do not sound natural, especially in the case of sustained excitation instruments such as the violin, because they have a wider timbre space and need a continuous control. One manner in which to improve these models in order to provide better controllability and expressive capabilities is to take into consideration performance gestures, that is, informing the model with "how is the instrument played".

Performance actions are sound producing gestures articulated by the musician that control/drive the production of sound (see in [8] for a categorization of musical gestures). When performing with a violin, one can produce a wide range of different timbre variations, by applying a complex combination of actions controlled by bow and fingers. Bowing actions are the most relevant concerning timbre and therefore we will focus on them.

We have developed a sensing system [14] by making use of two *Polhemus* 3D-motion trackers. Using data provided by this system we obtain bowing performance actions with great accuracy. Sound is acquired by means of a 4-channel bridge pickup that is then spectrally analyzed.

With this setup, we are able to synchronously collect large amounts of performance data (gestures and sound), that is used to train a set of neural networks. The trained networks are finally used in the transformation stage of a spectral concatenative synthesizer.

The paper is structured as follows: First we describe which data is acquired and how. In the case of sound recording, we discuss why are we using a bridge pickup instead of another device. Then we present the neural network that models the timbre, detailing its structure, inputs, output, the dataset for training and its performance. Finally we outline its use in the transformation procedure by the synthesizer and conclude by commenting some evaluation results and presenting further developments of the model.

DATA AQUISITON

With our measuring system consisting of the bridge pickup and the motion trackers we can capture an enormous amount of performance data. The main advantages over other systems like bowing machines [4] are that the range of the bowing actions is not constrained by the machine and that we can capture real performance data.

Measuring performance actions

Gestural data is captured with two 3d-motion trackers, one mounted on the violin and the other on the bow. We are able to estimate with great precision and accuracy the position of the strings, the bridge and the bow. With the data collected can calculate the following bowing performance parameters:

- *Bow-bridge distance* (*BBD* from now on), is the distance from the bow to the bridge. The normal range of values is from close to the bridge (less than 10 mm) to close to the fingerboard (around 50 mm).
- *Bow position*, bow transversal position that ranges from the tip (around 65 cm) to the frog (0cm).
- *Bow speed or bow velocity* is the derivative of bow position.
- *Bow pressure* or *bow force* is a measure proportional to the deformation of the bow and is dependant on *bow position*.
- *String* being played

According to the literature [11], [12] the bowing parameters that affect timbre the most seem to be *BBD*, *bow speed* and *bow force*. Notice that we consider additionally the *string* being played and *bow position*.

Recording the Sound

As stated by Cremer [3], in a simplified model of violin sound production, we can consider all the elements of sound transmission from the bridge to the listener as lineal. We could then assume that the sound pressure that arrives to our ears is proportional to the transversal force exerted by the string on its anchorage on the bridge as result of the Helmholtz motion when bowing. This means that we can separate the violin sound signal into two parts: bowed string vibration and a linear filter composed mainly by resonances of the bridge and the sounding box. The former can be measured with piezoelectric transducers [6] or deconvolved from a microphone recording [9], and the latter can be measured as an impulse response [9], [10].

The main advantages of measuring directly the string vibration are (1) we avoid problems with violin body resonances and sound radiation, and (2) we can obtain one signal per string. After trying several transducers we decided to use a 4-channel *Barbera* piezoelectric bridge transducer [7] (*BTS* from now on), because it captures a signal close to ideal string velocity signal. Given that transversal force on the bridge is proportional to string displacement [3], we can translate from string velocity to said force by integration.

In fig.1 we show the signal paths from the vibration of the bowed string to the radiated sound. The *BTS* picks the velocity of the string that is then integrated and finally convolved with the impulse response of a violin body. The resulting sound should be perceptually the same as the direct radiation.

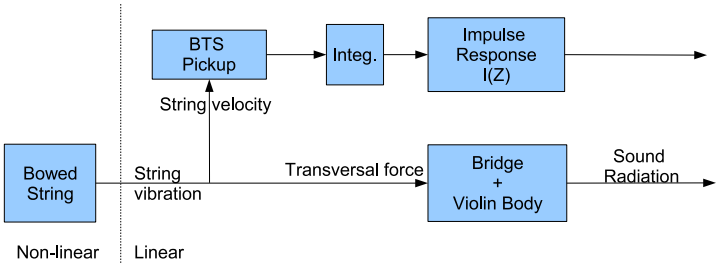


Figure 1: Signal Paths

MODELING THE TIMBRE

In this section we describe how the training dataset is built from collected performance data, then we show some results of the analysis of the data and finally we describe the set of neural networks that is proposed and used by the synthesizer.

Building the training dataset

The input to the model are the bowing actions described previously. The output is the corresponding spectrum. After dealing with different spectrum representations we arrived at the following: spectrum is divided into frequency bands and for each band we calculate the average harmonic energy. Bands have been fixed inspired in perceptual scales of the auditory system. The limits of the bands are [1000, 2000, 4000, 7000, 10000, 16000, 22000] Hz.

In order to have enough data a performer was asked to play open strings combining different values of bow *force*, bow *speed* and *BBD* covering the whole parameter space. After a segmentation of the recordings we obtained a dataset of around 122.500 analyzed frames corresponding to note sustains. In fig. 2 we show the distribution of each input parameter for the A-string. In the case of *BBD,* we can see how the performer played mainly at three distances: close to the bridge, middle and close to the fingerboard. For the other strings the distribution was similar.

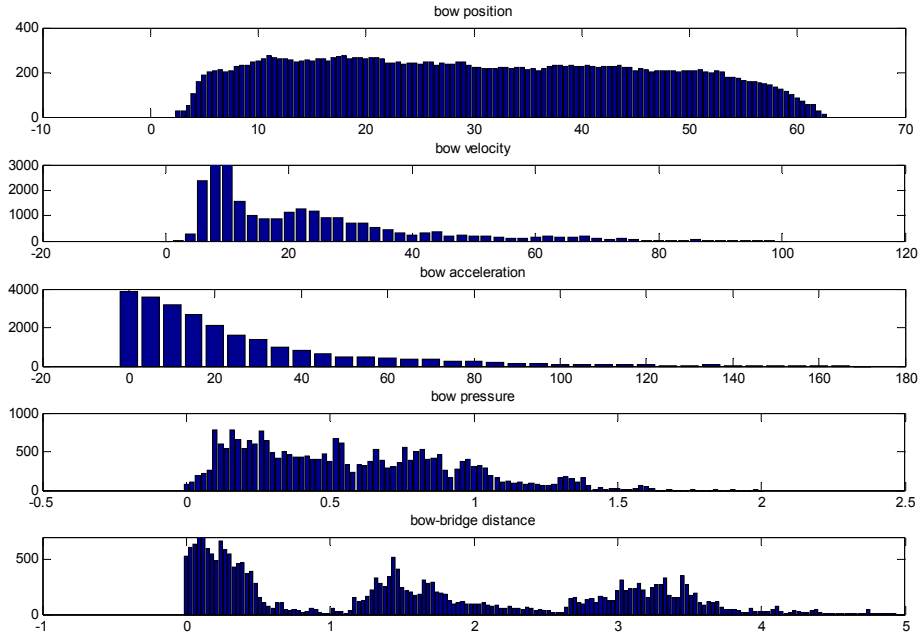


Figure 2: Distribution of input parameters

Data Analysis and Visualization

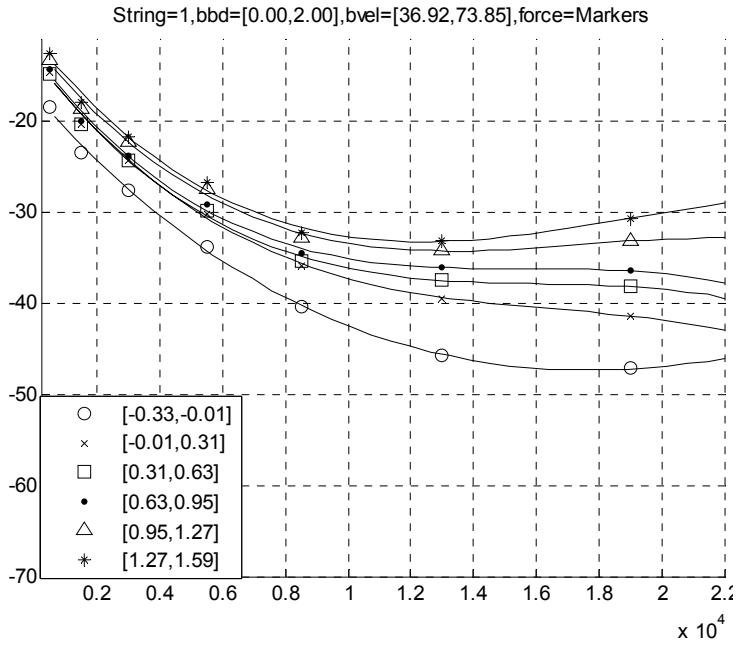
Before deciding the type of statistical model to use, data was analyzed in order to detect some patterns on the data. Here we describe the main characteristics observed.

In fig.3 and fig.4 there are several spectrum envelopes represented by markers indicating the average harmonic energy at each frequency band, and 3-rd degree polynomial fitting the markers. Notice that input parameters are discretized into categories (range of values). Fig.3 shows the evolution of the envelope when increasing *bow force* and in fig.4 when increasing *bow velocity.*

Input parameters, *string*, *bow force*, *bow speed* and *BBD* are affecting the spectrum in the following manner:

- Lower strings have higher spectral decay. See how spectra in fig.4 corresponding to the A-string have higher decay than in fig. 3 corresponding to the E-String.
- By increasing *bow force,* spectral energy shows a frequency dependent gain: Gain is higher for higher frequencies, whereas for low frequencies is almost inexistent. We can see this behaviour in fig. 3 for six different force category values.
- With increasing bowing speed there is an energy gain almost constant for all frequencies. This is depicted in fig. 4: we can see how envelopes are almost parallel.
- When bowing a string, harmonic nodes of the string under the bow are not excited. This is noticed in the ideal string velocity spectrum that has an *abs(sinc)* shape with nodes at those harmonics [12]. Conversely to string velocity spectrum, *BTS* spectrum does not

have those characteristic *abs(sinc)*-nodes. *BBD* seems to affect BTS spectrum in a similar way as *bow speed* does.



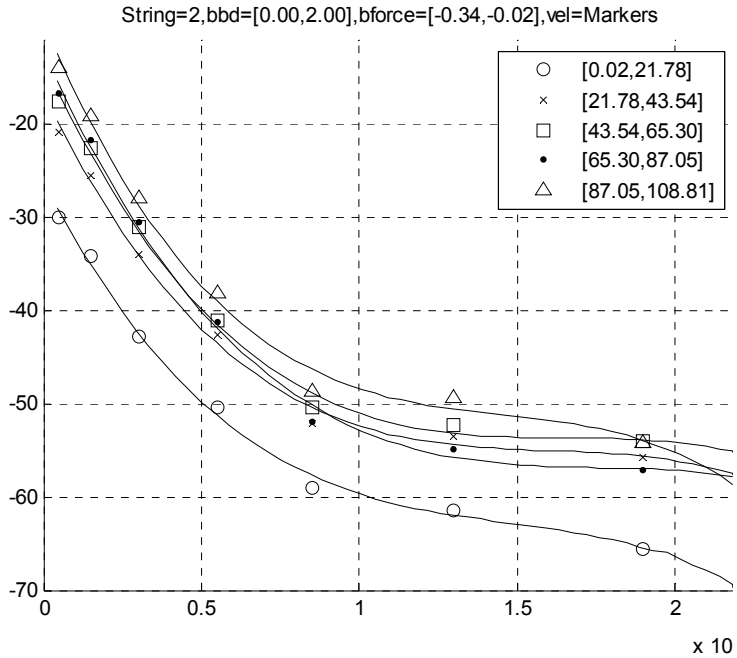
Fixed parameters:
- *BBD*
- *bow velocity*
- *string*

Bow force range [-0.5,1.5] is divided into 6 categories.

At low frequencies, spectrum remains almost constant.

Higher forces boost high frequencies.

Figure 3:Changes in spectrum by increasing bow force



Fixed parameters:
- *BBD*
- *bow force*
- *string*

Bow velocity range [0,110]cm/seg. is divided into 5 categories.

There is a constant energy gain at all frequencies when increasing bow velocity

Figure 4:Changes in spectrum by increasing bow velocity

The Model

Neural networks are non-linear statistical data modelling tools used to model complex relationships between input and output parameters. We need a model to predict numerical values (energy of the bands) given some numerical inputs and we choose neural networks

because they fit quite well to these requirements. They have been used previously for prediction of harmonic energy in [13].

For simplicity we build separate models (*model_$s_i b_j$*), each one predicting the harmonic energy of a specific band(*j*) given a specific string(*i*). Input parameters to each network are *string*, *BBD*, *bow position*, *bow velocity* and *bow force*. The output parameter is the average energy at a predefined band. Each network has a hidden layer with two neurons as represented in fig. 5. We used a feed-forward neural network trained by back-propagation.

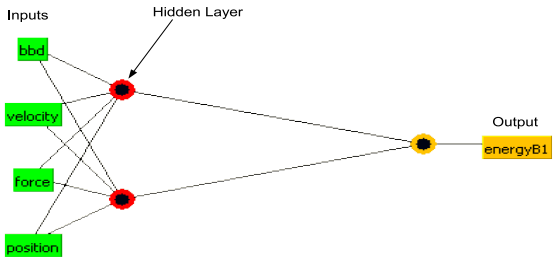


Figure 5: Neural Network architecture

The parameters for training the networks are:

- learning rate=0.3
- momentum=0.2
- number of epochs=500

For each network we get a similar regression performance. In the third column of table1 we show the results of a ten-fold cross-validation of the training data for the network corresponding to the first string and first frequency band (*model_$s_1 b_1$*).

As a comparison we also show the performance of a linear regression in the second column of table1. We can see how the use of the neural network improves a lot the performance. The obtained linear regression is as follows:

*energyB1= -2.3475 * bbd + 0.2148 * velocity + 2.6835 * force + 0.0117 * position -27.9791*

	Linear Regression	Neural Network
Correlation coefficient	0.8889	0.9708
Mean absolute error	2.5754	0.8893
Relative absolute error	41%	14%
Total Number of Instances	30,625	30,625

Table 1: Prediction errors for Linear Regression and Neural Network

SOUND TRANSFORMATIONS

Transformations are essential in a synthesizer that concatenates recorded samples because (1) they extend the parameter space that is not sampled and (2) they allow smoothed transitions when concatenating two samples. Our model is intended to complement other transformations within a spectral concatenative synthesizer. The synthesizer makes use of a database of samples containing both the sound and the control parameters that produced the sound. With this model we are able to modify the sound as if it was produced with other parameters.

In fig.6 we depict the transformation procedure: For each temporal frame, we predict the energy in the bands for both source and target actions. The difference envelope between source and target spectra is used to define the filter that is applied to the sound, obtaining the transformed sound. Target actions come from a performance model or from another stored sound. Notice that we do not use source energy values stored in the database, but we predict them with the model. This way the applied filter is not so sensible to prediction error, and furthermore, the model can be applied to non-sustained parts of the sound (attack, release and transition segments). Preliminary results are very promising.

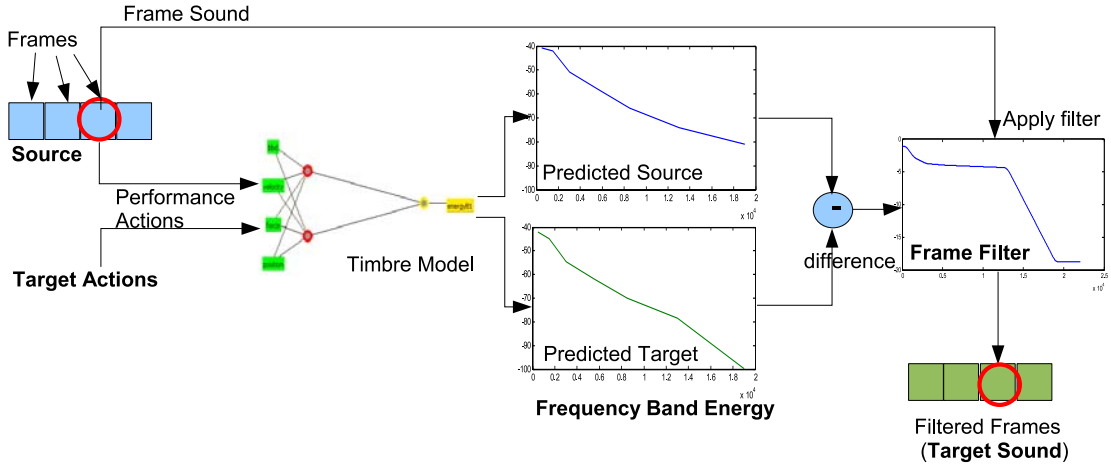


Figure 6:Transformation procedure of a frame

CONCLUSION

We presented a methodology for transforming the timbre of violin sound samples driven by performance actions. It is being tested as a complement to other transformations in a spectral concatenative synthesizer and the initial results are successful.

Further developments of the model will include refining the structure of the neural network and contrasting it with other machine learning methods, inform the model with other performance actions such as fingering and increasing the resolution of the timbre model (number of frequency bands).

Additionally, although sound signal captured with the *BTS* fits well for synthesis purposes, it is of interest to measure a signal directly related to string vibration so we could inform our model with physical formulae.

ACKNOWLEGMENTS

This work has been supported by Yamaha Corp.

References:

[1] Jordi Bonada and Xavier Serra. Synthesis of the singing voice by performance sampling and spectral models. IEEE signal processing magazine, 24:67, 2007.
[2] Diemo Schwarz. Corpus-based concatenative synthesis. IEEE signal processing magazine, 2007
[3] Lothar Cremer. Physics of the Violin. The MIT Press, November 1984.
[4] Cronhjort, A. **C**omputer-controlled bowing machine (MUMS), *STL-QPSR 2-3/1992,* pp. 61-66. 1992
[6] Jim Woodhouse and Claudia Fritz. Virtual violins project. http://www2.eng.cam.ac.uk/
[7] Richard Barbera. Resonant pick-up system, US patent 4,867,027, 1989.
[8] Musical Gestures Project, http://www.hf.uio.no/imv/forskning/forskningsprosjekter/musicalgestures/
[9] Farina A. and Langhoff A. and Tronchin L. Realisation of virtual musical instruments: measurements of the Impulse Response of Violins using MLS technique. 1995
[10] Perry R. Cook and Dan Trueman. A Database of Measured Musical Instrument Body Radiation Impulse Responses, and Computer Applications for Exploring and Utilizing the Measured Filter Functions.
[11] Anders Askenfelt. Measurement of the bowing parameters in violin playing II: Bow–bridge distance, dynamic range, and limits of bow force. 1989
[12] K. Guettler, Erwin Schooderwaldt, and Anders Askenfelt. Bow speed or bowing position-which one influences spectrum the most? In Proceedings of the Stockholm Music Acoustics Conference, 2003.
[13] Eric Lindeman. Music Synthesis with reconstructive phrase modelling. IEEE signal processing magazine, 80:92, 2007.
[14] E. Maestre, M. Blaauw, J. Bonada, A. Perez, E. Guaus. Acquisition of violin instrumental gestures using a commercial emf tracking device. Submitted to International Conference on Music Information. 2007