INTRODUCING DEEP MACHINE LEARNING FOR PARAMETER ESTIMATION IN PHYSICAL MODELLING

Leonardo Gabrielli *

A3LAB,
Università Politecnica delle Marche
Ancona, IT
l.gabrielli@univpm.it

Stefano Tomassetti *

A3LAB,
Università Politecnica delle Marche
Ancona, IT
tomassetti.ste@gmail.com

Stefano Squartini

A3LAB,
Università Politecnica delle Marche
Ancona, IT
s.squartini@univpm.it

Carlo Zinato

Viscount International SpA,
c.zinato@viscount.it

ABSTRACT

One of the most challenging tasks in physically-informed sound synthesis is the estimation of model parameters to produce a desired timbre. Automatic parameter estimation procedures have been developed in the past for some specific parameters or application scenarios but, up to now, no approach has been proved applicable to a wide variety of use cases. A general solution to parameters estimation problem is provided along this paper which is based on a supervised convolutional machine learning paradigm. The described approach can be classified as "end-to-end" and requires, thus, no specific knowledge of the model itself. Furthermore, parameters are learned from data generated by the model, requiring no effort in the preparation and labeling of the training dataset. To provide a qualitative and quantitative analysis of the performance, this method is applied to a patented digital waveguide pipe organ model, yielding very promising results.

1. INTRODUCTION

Almost all sound synthesis techniques require a nontrivial effort in the selection of the parameters, to allow for expressiveness and obtain a specific sound. The choice of the parameters depends on tone pitch, control dynamics, interpretation, and aesthetic criteria, with the aim of producing all the nuances required by musicians and their taste. Hereby, interest is given to the so-called *physically-informed* sound synthesis, a family of algorithms[1, 2] usually inspired by acoustic physical systems or derived from the transform in the digital domain of their formulation in the continuous-time domain. Such acoustic systems (e.g. strings, bores, etc.) often require simplifying hypotheses to limit the modeling complexity and to separate the acoustic phenomenon into different components. Notwithstanding this, the number of *micro*-parameters that control the sound and its evolution may be extremely large (see e.g. [3]) and if the effects of the parameters is intertwined the estimation effort may grow.

In the past some algorithms have been proposed to estimate some of the parameters of a physical model in an algorithmic fashion (see e.g. [4, 5]). These, however, require specific knowledge of physics, digital signal processing and psychoacoustic in order to provide an estimate in a white-box approach. Furthermore, specific estimation algorithms must be devised for each parameter. To solve these issues, a black-box approach could be undertaken to provide a good estimate of all the parameters at once. The goal of this preliminary work is to support the thesis that adequate machine learning techniques can be identified to satisfactorily estimate a whole set of model parameters without specific physical knowledge or model knowledge.

In the past some works extended the use of early machine learning techniques to the parametrization of nonlinearities in physical models [6, 7], or employed nonlinear recursive digital filters as physical models and employed parameter estimation techniques mutuated from the machine learning literature for the estimate of the coefficients [8, 9]). More in the spirit of this paper comes the work of Cemgil et al. on the calibration of a simple physical model employing artificial neural networks [10]. This work, however, to the best of our knowledge saw no continuation. Recently another computational intelligence approach for the estimation of a synthesizer parameters using a multiple linear regression model has been proposed, employing hand-crafted features [11]. To the best of our knowledge, however, no further attempt has been made to the estimation of a physical model parameters for sound synthesis employing other machine learning approaches. From this point of view, the swift development of deep machine learning techniques, and the exciting results obtained by these in a plethora of application scenarios, including musical representation and regression [12] suggests their application to the problem at hand. Following the recent advances of deep neural networks in audio applications, we propose here an end-to-end approach to the parameter estimation of acoustic physical models for sound synthesis based on Convolutional Neural Networks (CNN). The training can be conducted in a supervised fashion, since the model itself can provide audio and ground-truth parameters in an automated fashion. To evaluate the approach, this concept is applied to a valuable use case, i.e. a commercial flue pipe organ physical model, detailed in [13]. The estimation yields promising results which call for more research work.

The paper outline follows. Section 2 provides a mathematical formulation of the problem and the machine learning techniques employed to provide a general solution to it. Section 3 describes a real-world use case for validation of the proposed techniques.

* This work is partly supported by Viscount International SpA

Section 4 reports the implementation details and the experiments conducted, while Section 5 provides the results of these experiments and discusses them. Finally in Section 6 conclusions are drawn and open avenues for research are suggested.

2. THE PROPOSED METHOD

A physical model solves a set of differential equations that model a physical system and requires a set of of parameters $\boldsymbol{\theta}$ to generate a discrete time audio sequence $s(n)$. The goal of the model is to approximate acoustic signals generated by the physical system in some perceptual sense. If the model provides a mapping from the parameters to the discrete sequence $s(n)$, the problem of estimating the parameters $\boldsymbol{\theta}$ that yield a specific audio sequence identified as a target (e.g. an audio sequence sampled from the physical system we are approximating), is equivalent to finding the model inverse. Finding an inverse mapping (or an approximation thereafter) for the model is a challenging task to face, and a first necessary condition is the existence of the inverse for a given $s(n)$. Usually, however, in physical modelling applications, requirements are less strict, and generally it is only expected that audio signals match in perceptual or qualitative terms, rather than on a sample-by-sample basis. This means that, although, a signal $r(n)$ cannot be obtained from the model for any $\boldsymbol{\theta}$, a sufficiently close estimate $\hat{r}(n)$ may. Evaluating the distance between the two signals in psychoacoustic terms is a rather complex task and is out of the scope of this work

Artificial neural networks, and more specifically, deep neural networks of recent introduction, are well established to solve a number of inverse modelling problems. Here, we propose the application of a convolutional neural network that, provided with an audio signal of maximum length L in a suitable time-frequency representation, can estimate model parameters $\hat{\boldsymbol{\theta}}$ that fed to the physical model obtain an audio signal $\hat{s}(n)$ close to $s(n)$.

To achieve this, the inverse of the model must be learned employing deep machine learning techniques. If a supervised training approach is employed, the network must be fed with audio sequences and the related model parameters, also called *target*. The production of a dataset D of such tuples $D = \{(\theta_i, s_i(n)), i = 1, ..., M\}$ allows the network to try learn the mapping that connects these. The production of the dataset is often a lengthy task and may require human effort. However, in this application, the model is known and, once implemented, it can be employed to automatically generate a dataset D in order to train the neural network.

The neural network architecture proposed here allows for end-to-end learning and is based on convolutional layers. Convolutional neural networks globally received attention from a large number of research communities and found application into commercial applications, especially in the field of image processing, classification, etc. They are also used with audio signals, where, usually, the signal is provided to the CNN in a suitable time-frequency representation, obtained by means of a Short-Time Fourier Transform (STFT) with appropriate properties of time-frequency localization. The architecture of a CNN is composed of several layers in the following form,

$$\boldsymbol{Z}^{(m)} = h(\sigma(\boldsymbol{Q}^{(m)})), \tag{1}$$

$$\boldsymbol{Q}^{(m)} = W^{(m)} * \boldsymbol{Z}^{(m-1)}, \tag{2}$$

and $\boldsymbol{Z}^{(0)} \equiv \boldsymbol{X}$, where M denotes the total number of layers, $W^{(m)}$, $m = 1, ..., M$ are the filter weights to be learned, $\sigma(\cdot)$ is a non-linear sigmoid activation function, $\boldsymbol{Z}^{(m-1)}$ is the output of layer $m-1$, called *feature map*, $\boldsymbol{Q}^{(m)}$ is the result of convolution on the previous feature map and $h(\cdot)$ is a *pooling* function that reduces the feature map dimensionality. After M convolutional layers, one or more fully connected layers are added. The final layer has size p and outputs an estimate of the model parameters $\hat{\boldsymbol{\theta}}$.

Learning is conducted according to an update rule, which is based on the evaluation of a loss function, such as

$$\ell(W, e) = \|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}^{(\mathbf{e})}\|_2 \tag{3}$$

where e is the training epoch. Training is iterated until a convergence criterion is matched or a maximum number of epochs has passed. To avoid overfitting and reduce training times early stopping by validation can be performed, which consists in evaluating after a constant number of epochs the loss, called validation loss, calculated against a validation set, i.e. a part of the dataset that is not used for training and is, hence, *new* to the network. Even if the training loss may still be improving, if the validation loss does not improve after some training epochs, the training may stop avoiding a network overfit.

Finally, once the network is trained, it can be fed with novel audio sequences to estimate the physical model parameters that can obtain a result close to the original. In the present work we employ additional audio sequences generated by the model in order to measure the distance in the parameter space between the parameters θ_i and $\hat{\theta}_i$. If non-labelled audio sequences are employed (e.g. sequences sampled from the real-world), it is not straightforward to validate the generated result, that is why in the present work no attempt has been made to evaluate the estimation performance of the network with real-world signals.

3. USE CASE

The method described in the previous section has been applied to a specific use case of interest, i.e. a patented digital pipe organ physical model. A pipe organ is a rather complex system [14, 15], providing a challenging scenario for physical modelling itself. This specific model, already employed on a family of commercial digital pipe organs, exposes 58 macro-parameters to be estimated for each key, some of which are intertwined in a non linear fashion and are acoustic-wise non-orthogonal (i.e. jointly affect some acoustic features of the resulting tone).

We introduce here some key terms for later use. A pipe organ has one or more keyboards (*manuals* or *divisions*), each of which can play several *stops*, i.e. a set of pipes, typically one or more per key, which can be activated or deactived at once by means of a *drawstop*. When a stop is activated, air is ready to be conveyed to the embouchure of each pipe, and when a key is pressed, a valve is opened to allow air flow into the pipe. Each stop has a different timbre and pitch (generally the pitch of the central C is expressed in feet measuring the pipe length). From our standpoint, the concept of stop is very important, since each key in a stop will sound similar to the neighboring ones in terms of timbre and envelope, and each key will trigger different stops which may have similar pitch but different timbre. In a pipe organ it can be expected that pipes in a stop have consistent construction features (e.g. materials, geometry, etc.) and a physical model that mimics that pipe stop may have correlated features along the keys but this is not

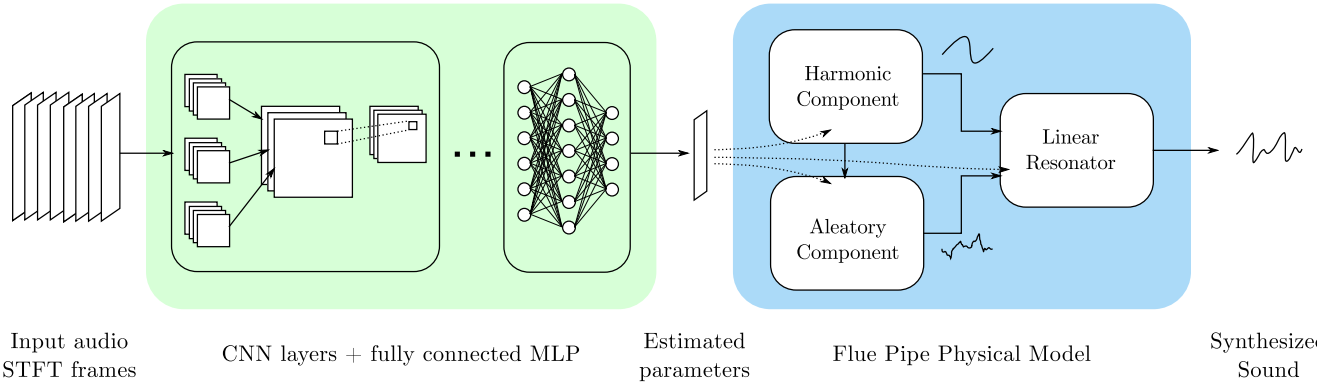

Figure 1: Overview of the proposed system including the neural network for parameter estimation and the physical model for sound synthesis.

an assumption that can be done, so it is necessary to conduct an estimate of the parameters for each key in a stop.

The physical model employed in this work is meant to emulate the sound of flue pipes and is described in detail in the related patent. To summarize, it is constituted by three main parts:

1. exciter: models the wind jet oscillation that is created in the embouchure and gives rise to an air pressure fluctuation,
2. resonator: a digital waveguide structure that simulates the bore,
3. noise model: a stochastic component that simulates the air noise modulated by the wind jet.

The parameters involved in the sound design are widely different in range and meaning, and are e.g. digital filters coefficients, non-linear functions coefficients, gains, etc. The diverse nature of the parameters requires a normalization step which is conducted on the whole training set and maps each parameter in a range [-1, 1], in order for the CNN to learn all parameters employing the same arithmetic. A de-normalization step is required, thus, to remap the parameters to their original range.

Figure 1 provides an overview of the overall system for parameter estimation and sound generation including the proposed neural network and the physical model.

4. IMPLEMENTATION

The CNN and the machine learning framework has been implemented as a python application employing Keras[1] libraries and Theano[2] as a backend, running on a Intel i7 Linux machine equipped with 2 x GTX 970 graphic processing units. The physical model is implemented as both an optimized DSP algorithm and a PC application. The application has been employed in the course of this work and has been modified to allow producing batches of audio sequences and labels for each key in a stop (e.g. to produce the dataset, given some specifications). Each audio sequence contains a few seconds of a tone of specific pitch with given parameters.

A dataset of 30 *Principal* pipe organ stops, each composed of 74 audio files, one per note, has been created taking the parameters from a database of pre-existing stops hand-crafted by expert musicians to mimic different hystoric styles and organ builders. The dataset has been split by 90% and 10% for the training and validation sets respectively, for a total of 1998 samples for the former and 222 for the latter. The only pre-processing conducted is the normalization of the parameters and the extraction of the STFT. A trade-off has been selected in terms of resolution and hop size to allow a good tracking of harmonics peaks and attack transient. Figure 2 shows the input STFT for a A4 tone used for training the network.

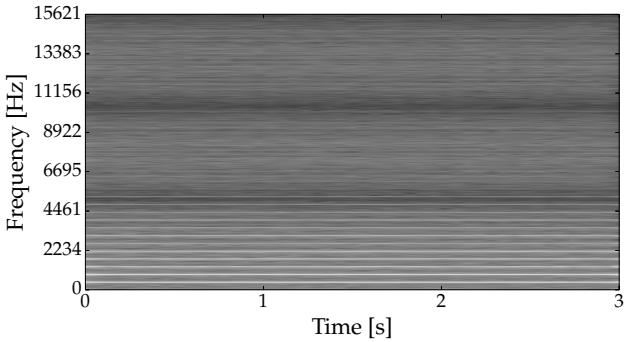

Figure 2: The STFT for an organ sound in the training set. The tone is a A4.

The CNN architecture is composed of up to 6 convolutional layers, with optional batch normalization [16] and max pooling, up to 3 fully connected layers and a final activation layer. For the training, stochastic gradient descent (SGD), Adam and Adamax optimizers have been tested. The training made use of early stopping on the validation set. A coarse random search has been performed first to pinpoint some of the best performing hyperparameters. A finer random search has been, later, conducted keeping the best hyperparameters from the previous search constant. Tests have been conducted with other stops not belonging to the training set and averaged for all the keys in the stops.

5. RESULTS

The loss used in training, validation and testing is the Mean Square Error (MSE) calculated at the output of the network with respect

[1] http://keras.io

[2] http://deeplearning.net/software/theano/

to the target, before de-normalization. Results are, therefore, evaluated in terms of MSE.

Table 2 reports the 15 best hyperparameters combinations in the fine random search. The following activation function combinations are reported in Table 2:

1. A: employs *tanh* for all layers,
2. B: employs *tanh* for all layers besides the last one, that uses a Rectified Linear Unit (*ReLU*)[17],
3. C: employs ReLU functions for all layers,
4. all other combinations of the two aforementioned activation functions obtained higher MSE score and are not included here.

Results are provided against a test set of 222 samples from three organ stops, and have same learning rate ($1E-5$), momentum max (0.9), pool sizes (2x2 for each convolutional layer), receptive field sizes (3x3 for each convolutional layer) and optimizer (Adamax [18]). These fixed parameters have been selected as the best ones after the coarse random search.

Figure 4 shows the training and validation loss plots for the first 200 training epochs for the first combination in Table 2. The loss is based on the MSE for all parameters before denormalization. This means all parameters contribute to the MSE with the same weight and makes results clearer to evaluate. Indeed, if the MSE would be evaluated after de-normalization, parameters with larger excursion ranges would have a larger effect on the loss (e.g. a delay line length versus a digital filter pole coefficient). Validation Early Stopping is performed when the minimum validation loss is achieved to prevent overfitting and reduce training times. In Figure 4, e.g., the validation loss minimum (0.027) occurs at epoch 122, while the training loss minimum (0.001) occurs at epoch 198. Two spectra and their waveforms are shown in Figure 3 showing the similarity of the original tone and the estimated one, both obtained from the flue pipe model.

Results provided in terms of MSE, unfortunately, are not acoustically motivated: not all errors have the same effect, since parameters affect different perceptual features, thus large errors on some parameters may not result as easily perceived as small errors on other parameters. To the best of the authors' knowledge there is no agreed method in literature to objectively evaluate the degree of similarity of two musical instruments spectra. Previous works suggested the use of subjective listening tests [19, 20, 21, 22], but an objective way to measure this distance is still to be addressed.

In order to provide the reader with some cues on how to evaluate these results, we draw from the psychoacoustic literature, as an example, the work from Caclin et al. [23], where spectral irregularity is proposed as a salient feature in sound similarity rating. Spectral irregularity is modelled, in their work, as the attenuation of even harmonics in dB (EHA). The perceptual extremes are chosen to be a tone with a regular spectral decay and a tone with all even harmonics attenuated by 8dB. The mean squared error calculated for these two tones (HMSE) for the first 20 harmonics (as done in their work) is 32dB. In our experiments, results vary greatly, depending on the pipe sounds to be modeled by the CNN. As an example, Figure 3 shows time and frequency plots of two experiments. They both present two A4 signals created by the physical model with two different parameter configurations handcrafted by an expert musician, called respectively "Stentor" and "HW-DE". The peak amplitude of the harmonics for the tones in Figure 3 are evaluated in terms of HMSE for the first 10, 20 and 35 harmonics in Table 1 [3].

	HMSE10	HMSE20	HMSE35
Stentor	5.2 dB	9.4 dB	18.9 dB
HW-DE	0.3 dB	10.1 dB	12.3 dB

Table 1: Harmonics MSE (HMSE) for the first 10, 20 and all 35 harmonics for the tones shown in Figure 3.

Activations class	Minibatch size	Internal layers size	MSE
A	40	(16, 16, *512*, 58)	0.261
B	50	(4, 6, 8, 10, *512*, 58)	0.203
A	40	(16, 16, 32, 32, *512*, 58)	0.164
A	40	(16, 16, 32, 32, *512*, 58)	**0.139**
A	25	(16, 16, 32, 32, *1024*, 58)	0.161
A	40	(16, 16, 32, 32, *1024*, 58)	0.266
A	25	(16 ,16, 32, 32, *512*, 58)	0.156
A	40	(16, 16, 32, 32, *1024*, 58)	0.252
A	50	(4, 6, 8, 10, *512*, 58)	0.166
A	40	(16, 16, 32, 32, *256*, 58)	0.179
A	25	(2, 2, 4, 4, *128*, 58)	0.254
C	740	(16, 16, 32, 32, *512*, 58)	0.252
B	50	(4, 6, 8, 10, *512*, 58)	0.214
C	740	(16, 16, 32, 32, *512*, 58)	0.257
B	40	(16, 16, *512*, 58)	0.179

Table 2: The best 15 results of the fine random hyperparameters search. Activation classes are described in the text. The MSE is evaluated before denormalization, thus, all parameters have the same weight. Please note: all the layers are convolutional with kernel size as indicated, exception made for the second to last which is a fully connected layer. The output layer has fixed size equal to the number of model parameters.

The first tone, shown in Figure 3(a) has a spectrum with a good match for the first harmonics, but with some outliers and a generally bad match for harmonics higher than 12. The latter, shown in Figure 3(b) has a good match, especially in its first 10 partials, but the error raises with higher partials, especially from the 12th up. This is reflected by an HMSE10 of 5.2 dB vs. 0.3 dB and an error on the whole spectrum (HMSE35) of 18.9 dB vs. 12.3 dB. HMSE20 values for the two tones do not differ significantly, due to the averaging done on spectral ranges with different results, but we leave them to the reader so that they can be compared to the experiments of Caclin et al. The HMSE20 values are somewhere between the two extremes, "same" and "different", tending more towards the former. Informal listening tests conducted with expert musicians suggest that the estimated "Stentor" tone does not match well to the original, while the "HW-DE" does match sufficiently. We hypothesize that the spectral matching of the first harmonics is more relevant in psychoacoustic terms to assess similarity, but we leave this to more systematic studies as a future work. The tones are made available to the reader online[4].

[3] The sampling frequency of the tones is 31250 Hz, thus, 35 is the highest harmonic for a A4 tone.

[4] http://a3lab.dii.univpm.it/research/10-projects/84-ml-phymod

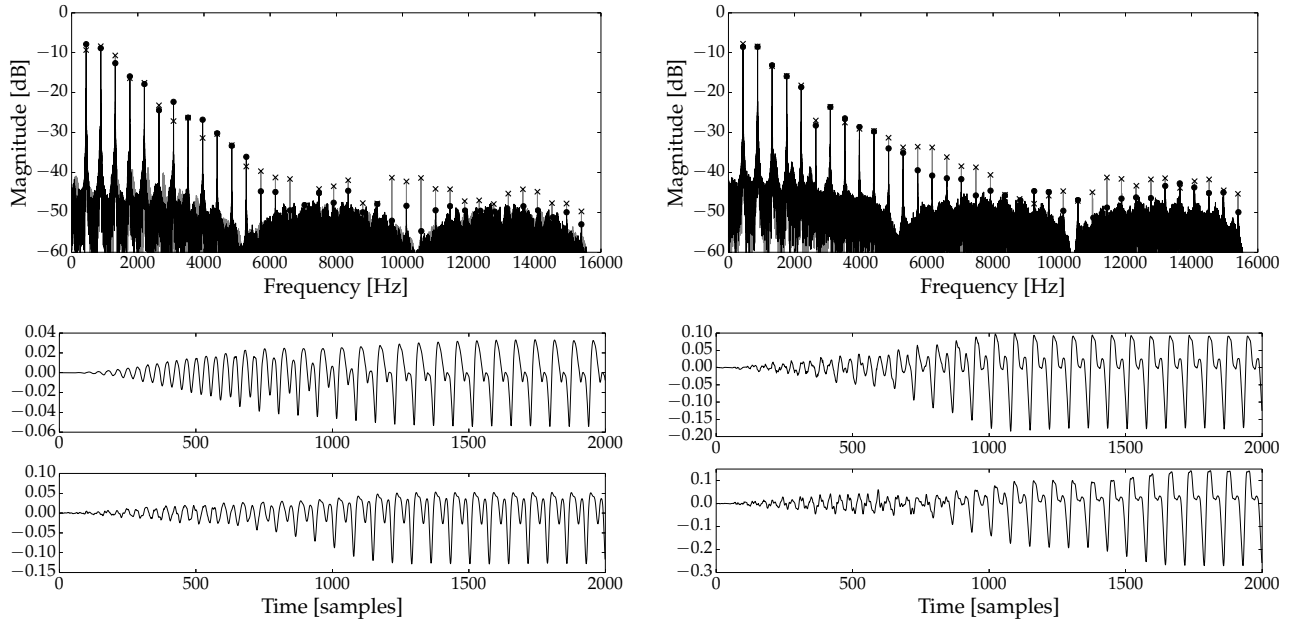

Figure 3: Spectra and harmonic content for two A4 tones from (a) *Principal* stop named "Stentor", and (b) *Principal* stop named "HW-DE". The gray lines and crosses show the spectrum and the harmonic peaks of $\hat{s}(n)$, while the black line and dots show the spectrum and the harmonic peaks of $s(n)$. In the waveform plots, the upper ones are obtained by the target parameters, while the lower ones are obtained with the estimated parameters.

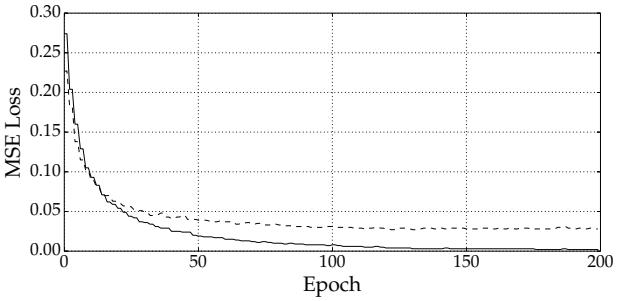

Figure 4: Training (solid line) and validation loss (dashed line) for the best combination reported in Table 2. Please note that the validation loss minimum (0.027) occurs at epoch 122, while the training loss minimum (0.001) occurs at epoch 198.

6. CONCLUSIONS

In this paper a machine learning paradigm that is general and flexible is proposed and applied to the problem of estimating the parameters for a physical model for sound synthesis. To validate the idea a specific use case of a flue pipe physical model has been employed. Results in term of MSE are good and tones spectra have a good match in terms of harmonic content, although results vary. Such results, coming from a real-world application scenario motivate the authors in believing that a machine learning paradigm can be employed with success for the problem at hand. Nonetheless, this first achievement calls for more research works. First of all a validation is required with data sampled from a real pipe organ for further assessment and to evaluate the robustness of this method to noise, reverberation and such.

During the evaluation of the results, it has been discovered that results may greatly vary depending on the stops acoustic character. A rigorous approach to machine learning requires understanding whether the training set, which is a sampling of the probability distribution of all flue pipe stops obtained by the model, is representative of that probability distribution. Furthermore, it can be expected that the organ stops that can be obtained from the model are a subset of all organ stops that could physically built, due to model limitations and simplifying hypotheses. On the other hand, due to its digital implementation, the model could circumvent some physical limitation of real flue pipes, thus, yielding stops that are not physically feasible. This calls for a better understanding of how different stops are related to each others in the modelled and the physical realms, to understand before trying the machine learning approach, whether satisfying results can be obtained. As a last remark, these are general issues that apply also to other physical model or musical instruments.

7. REFERENCES

[1] Vesa Välimäki, Jyri Pakarinen, Cumhur Erkut, and Matti Karjalainen, "Discrete-time modelling of musical instruments," *Reports on progress in physics*, vol. 69, no. 1, pp. 1, 2005.

[2] Julius O. Smith, "Virtual acoustic musical instruments: Re-

view and update," *Journal of New Music Research*, vol. 33, no. 3, pp. 283–304, 2004.

[3] Stefano Zambon, Leonardo Gabrielli, and Balazs Bank, "Expressive physical modeling of keyboard instruments: From theory to implementation," in *Audio Engineering Society Convention 134*. Audio Engineering Society, 2013.

[4] Janne Riionheimo and Vesa Välimäki, "Parameter estimation of a plucked string synthesis model using a genetic algorithm with perceptual fitness calculation," *EURASIP Journal on Advances in Signal Processing*, vol. 2003, no. 8, 2003.

[5] Vasileios Chatziioannou and Maarten van Walstijn, "Estimation of clarinet reed parameters by inverse modelling," *Acta Acustica united with Acustica*, vol. 98, no. 4, pp. 629–639, 2012.

[6] Carlo Drioli and Davide Rocchesso, "Learning pseudo-physical models for sound synthesis and transformation," in *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*. IEEE, 1998, vol. 2, pp. 1085–1090.

[7] Aurelio Uncini, "Sound synthesis by flexible activation function recurrent neural networks," in *Italian Workshop on Neural Nets*. Springer, 2002, pp. 168–177.

[8] Alvin WY Su and Liang San-Fu, "Synthesis of plucked-string tones by physical modeling with recurrent neural networks," in *Multimedia Signal Processing, 1997., IEEE First Workshop on*. IEEE, 1997, pp. 71–76.

[9] Alvin Wen-Yu Su and Sheng-Fu Liang, "A new automatic IIR analysis/synthesis technique for plucked-string instruments," *IEEE transactions on speech and audio processing*, vol. 9, no. 7, pp. 747–754, 2001.

[10] Ali Taylan Cemgil and Cumhur Erkut, "Calibration of physical models using artificial neural networks with application to plucked string instruments," *PROCEEDINGS-INSTITUTE OF ACOUSTICS*, vol. 19, pp. 213–218, 1997.

[11] Katsutoshi Itoyama and Hiroshi G Okuno, "Parameter estimation of virtual musical instrument synthesizers," in *Proc. of the International Computer Music Conference (ICMC)*, 2014.

[12] Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio, "SampleRNN: an unconditional end-to-end neural audio generation model," in *5th International Conference on Learning Representations (ICLR 2017)*, 2017.

[13] C. Zinato, "Method and electronic device used to synthesise the sound of church organ flue pipes by taking advantage of the physical modeling technique of acoustic instruments," Oct. 28 2008, US Patent 7,442,869.

[14] NH Fletcher, "Sound production by organ flue pipes," *The Journal of the Acoustical Society of America*, vol. 60, no. 4, pp. 926–936, 1976.

[15] Neville H Fletcher and Suszanne Thwaites, "The physics of organ pipes," *Scientific American*, vol. 248, no. 1, pp. 94–103, 1983.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[17] Vinod Nair and Geoffrey E Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[18] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] Simon Wun and Andrew Horner, "Evaluation of weighted principal-component analysis matching for wavetable synthesis," *J. Audio Engineering Society*, vol. 55, no. 9, pp. 762–774, 2007.

[20] H. M. Lehtonen, H. Penttinen, J. Rauhala, and V. Välimäki, "Analysis and modeling of piano sustain-pedal effects," *J. Acoustical Society of America*, vol. 122, pp. 1787–1797, 2007.

[21] Brahim Hamadicharef and Emmanuel Ifeachor, "Objective prediction of sound synthesis quality," *115th Convention of the AES, New York, USA*, p. 8, October 2003.

[22] L. Gabrielli, S. Squartini, and V. Välimäki, "A subjective validation method for musical instrument emulation," in *131st Audio Eng. Soc. Convention, New York*, 2011.

[23] Anne Caclin, Stephen McAdams, Bennett K Smith, and Suzanne Winsberg, "Acoustic correlates of timbre space dimensions: A confirmatory study using synthetic tones a," *The Journal of the Acoustical Society of America*, vol. 118, no. 1, pp. 471–482, 2005.