NOTE ONSET DETECTION USING ECHO STATE NETWORKS

Peter Steiner, Simon Stone, Peter Birkholz

Institute of Acoustics and Speech Communication, Technische Universität Dresden
peter.steiner@tu-dresden.de

Abstract: In music analysis, one of the most fundamental tasks is note onset detection – detecting the beginning of new note events. It is the basis for more high-level tasks, such as beat tracking or tempo detection. The main outline of all approaches for onset detection is roughly the same: The audio signal is transformed into an Onset Detection Function (ODF), which is zero for most of the time but has pronounced peaks in case of onsets. Applying peak picking algorithms on the ODF, the onset times can be extracted. Currently, Convolutional Neural Networks (CNNs) define the state of the art. In this paper, a first exploration of Echo State Networks (ESNs) to obtain an ODF is presented. ESNs have achieved comparable results to CNNs in several recognition tasks, such as speech and image recognition. Features were extracted using a bank of filters with a logarithmic frequency spacing. The feature vectors were fed into the ESN that computed the ODF. Applying a simple threshold-based peak picking algorithm on the ODF, the onsets were detected. For the hyperparameter optimization, a dataset with pre-defined splits for an 8-fold cross validation was used. With all hyperparameters optimized, we reached an F-Measure of 0.812 using a bidirectional ESN with 8000 neurons.

1 Introduction

In music analysis, one of the most fundamental tasks is note onset detection. The onset is defined as the beginning of a new note event in an acoustic signal. It serves as the basis for more high-level tasks such as beat tracking or tempo detection, which need information about the temporal evolvement of a note sequence. The main outline of all approaches for onset detection is roughly the same: The audio signal is transformed into an Onset Detection Function (ODF), which is zero for most of the time, and has pronounced peaks in case of onsets. Thus, applying peak picking algorithms on the ODF, the onset times can be extracted.

The most common algorithms for onset detection are based on spectral differences or phase deviations [1] between adjacent frames. Therefore, different signal transformations, such as short term spectra [1] or filterbanks [2, 3], can be used. Depending on the kind of onsets to be detected, spectral differences or phase deviations lead to clear peaks and have low computational costs.

More recent approaches to obtain an ODF are based on machine learning techniques. Marolt et al. [4] used neural networks to improve a peak picking process applied on their ODF. However, this approach was restricted to the piano. Lacoste and Eck [5] were the first ones to let a neural network learn the ODF from feature vectors, in their case it was a simple feed forward network. Later, Eyben et al. [6] used a neural network with Bidirectional Long-Short-Term Memory cells (BLSTMs), which achieved an F-measure (see below) of 0.873 , which was the state of the art. Schlüter and Böck [7] used Convolutional Neural Networks (CNNs) for the same task. This approach improved the results for onset detection again significantly, reaching an F-Measure of 0.903.

Echo State Networks (ESNs) by Herbert Jaeger [8] are a special kind of Recurrent Neural Networks (RNNs). Although they are rather unknown, in the past years, they have achieved comparable results to CNNs in several recognition tasks, for example in speech and image recognition [9, 10]. Furthermore, they achieved the best ranking during the last MIREX challenge [11] for multipitch tracking. In this paper, the potential of ESNs for onset detection was explored. They have several beneficial properties for this task:

- Because of their recurrent connections, ESNs are suitable for processing temporal information.
- The training procedure is much easier than for concurrent approaches, due to less free parameters.

2 Onset Detection with Echo State Networks

Echo State Networks (ESNs) are a kind of Recurrent Neural Network (RNN). Usually, RNN architectures consist of sequential layers with connection weights to be trained using time-dependent Backpropagation. The fundamental difference of ESNs is their simple architecture, consisting of input and recurrent connection weights, which are fixed random values. Only the output weights are trained using linear regression.

Because of the recurrent connections inside the reservoir, information from previous inputs is retained inside the ESN for a certain amount of time. Depending on the choice of hyper-parameter values, the reservoir acts as a long- or short-term memory. The neurons inside the reservoir are non-linear, typically using sigmoidal activation functions. Thus, the reservoir acts as a non-linear transformation of the low-dimensional input space into a high-dimensional features space, where the desired output is a multi-linear function of the transformed features. The main outline of the proposed ESN-based model for onset detection is depicted in Figure 1.

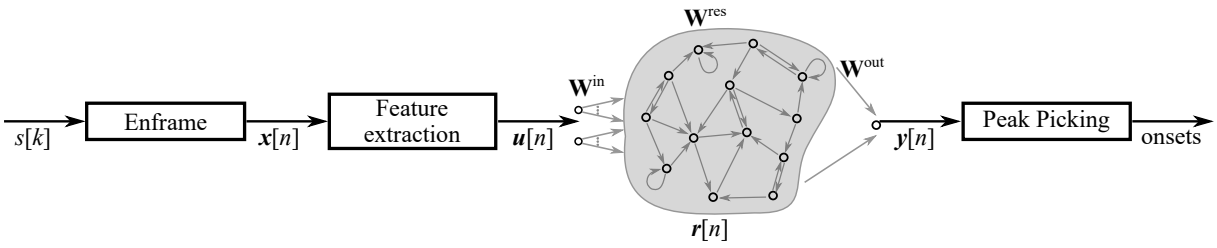

Figure 1 – Outline of the ESN-based proposed model: The input signal $s[k]$ with the sample index k was divided into overlapping frames, from which normalized feature vectors $\mathbf{u}[n]$ were extracted and fed into the reservoir using the input weight matrix $\mathbf{W}^{\text{in}}$. The reservoir consists of unordered and via the reservoir matrix $\mathbf{W}^{\text{res}}$ sparsely connected neurons. The one-dimensional output $\mathbf{y}[n]$ is a linear combination of the reservoir states $\mathbf{r}[n]$ and the output weight matrix $\mathbf{W}^{\text{out}}$, which was trained using linear regression. The output serves as the ODF, in which onsets were extracted usign a peak picking algorithm.

2.1 Feature extraction

The input signal $s[k]$ with the sample index k and a sampling frequency of 44.1 kHz was divided into overlapping frames with a frame rate of 100 Hz and the frame index n. Each frame was windowed using a Hann window with 2048 samples. The windowed frames were transformed into the frequency space using the short-term Fourier transform. Next, a triangular filterbank with 12 filters per octave and the frequency range of 27.5 Hz to 16 000 Hz was applied to every short-term spectrum to reduce the dimension and to introduce a semitone frequency spacing.

At last, the logarithm of the magnitude was taken. Because onsets are strongly correlated with energy changes in frequency bands, in many cases, such as [6], the spectral flux is added to the feature vector. Therefore, the first order difference between adjacent feature vectors was computed. Negative differences were set to zero, and this half-wave rectified vector appended to the features described above. This led to a feature vector size of $N^{\text{in}} = 160$.

2.2 Echo State Network (ESN)

The main outline of an ESN is depicted in the center of Figure 1. It consists of the input weights $\mathbf{W}^{\text{in}}$, the reservoir weights $\mathbf{W}^{\text{res}}$ and the output weights $\mathbf{W}^{\text{out}}$.

The input weight matrix $\mathbf{W}^{\text{in}}$ has the dimension of $N^{\text{res}} \times N^{\text{in}}$ where $N^{\text{in}} = 160$ and N^{res} are the size of the input feature vector and the size of the reservoir, respectively. All values in this matrix were initialized from a uniform distribution between ± 1.0. Next, each node of the reservoir was only connected to $K^{\text{in}} = 10$ randomly selected input entries. The other connections were set to zero, leading to a very sparse matrix $\mathbf{W}^{\text{in}}$. The input weight matrix was then scaled using the input scaling factor α_{U}, which was a hyper-parameter to be tuned.

The reservoir weight matrix $\mathbf{W}^{\text{res}}$ is a square matrix of the size $N^{\text{res}} \times N^{\text{res}}$, which was also initialized from a standard normal distribution. Each reservoir node received values from only $K^{\text{rec}} = 10$ randomly selected other nodes. The other connections were set to zero. The reservoir matrix $\mathbf{W}^{\text{res}}$ was normalized by its largest absolute eigenvalue to achieve a spectral radius $\rho = 1.0$, because it was shown in [8] that the echo state property holds as long as $\rho \leq 1.0$. By tuning α_{U} and ρ, it is possible to balance, how strongly the network memorizes past inputs compared to the present input.

If $\mathbf{r}[n]$ represents the reservoir state, the basic equations to describe the ESN can be written in the following way:

$$\mathbf{r}[n] = (1-\lambda)\mathbf{r}[n-1] + \lambda f_{\text{res}}(\mathbf{W}^{\text{in}}\mathbf{u}[n] + \mathbf{W}^{\text{res}}\mathbf{r}[n-1] + \mathbf{W}^{\text{bi}}) \tag{1}$$

$$\mathbf{y}[n] = \mathbf{W}^{\text{out}}\mathbf{r}[n] \tag{2}$$

Equation (1) is a leaky integration of the reservoir neurons. Depending on the leakage $\lambda \in [0,1]$, the reservoir can act as a long-term or a short-term memory. The reservoir activation function $f_{\text{res}}(\cdot)$ controls the non-linearity of the system. Here, the tanh-function was used, because its lower and upper boundaries of ± 1 ensure stable reservoir states. The bias vector $\mathbf{W}^{\text{bi}}$ with N^{res} dimensions is an additional bias term, which consists of fixed random values from a uniform distribution between ± 1.0 and multiplied by the hyper-parameter α_{B}, which was used to scale the non-linearity of the system.

Equation (2) shows how to compute the N^{out}-dimensional output vector $\mathbf{y}[n]$ from a given reservoir state $\mathbf{r}[n]$, which was expanded by one bias term. The output is obtained by a linear combination of the reservoir state and the output weight matrix $\mathbf{W}^{\text{out}}$. For training, all reservoir states were collected in the reservoir state collection matrix $\mathbf{R}$, and expanded by one bias term.The desired outputs $\mathbf{d}[n]$, which was 0.0 for non-onsets and 1.0 for onsets were collected into the desired output collection matrix $\mathbf{D}$. Afterwards, $\mathbf{W}^{\text{out}}$ was obtained using regularized linear regression (3), i.e. ridge regression to prevent overfitting to the training data. The regularization parameter $\varepsilon = 0.01$ penalized large values in $\mathbf{W}^{\text{out}}$, and $\mathbf{I}$ is the identity matrix.

$$\mathbf{W}^{\text{out}} = \left(\mathbf{R}\mathbf{R}^{\text{T}} + \varepsilon\mathbf{I}\right)^{-1}(\mathbf{D}\mathbf{R}^{\text{T}}) \tag{3}$$

The size of the output weight matrix $(N^{\text{out}} \times N^{\text{res}} + 1)$ determines the total number of free parameters to be trained in ESNs. The output $\mathbf{y}[n]$ corresponded to the onset detection function ODF.

Bidirectional reservoirs

In the case of bidirectional reservoirs, the input was first fed through the ESN as described before. Before the linear regression, the inputs were reversed in time, and again fed into the same reservoir. Afterwards, the reservoir states were again reversed in time. The reservoir state collection matrix $\mathbf{R}$ was finally built by combining the states from the forward and backward pass. This doubled the number of free parameters for the linear regression. For example, the number of features for a reservoir with 500 neurons is 500 in the unidirectional and 1000 in the bidirectional case. The final training remained the same as before.

2.3 Peak Picking

After the linear regression, the output indicated an onset or non-onset. Ideally, it would be zero for a non-onset and one for an onset. However, due to the linear regression, the output is neither binary nor bounded between zero and one. Furthermore, treating onset detection as a classification problem, the ratio between onsets and non-onsets is highly imbalanced. Thus, it is likely that note onsets are characterized by peaks, which had not always the same height.Thus, the output was considered to be an onset detection function (ODF), in which onsets can be detected using a peak picking algorithm. In this paper, the simple threshold-based peak picking algorithm proposed in [7] is used. At first, the ODF is smoothed using a Hanning window with 5 samples. Next, local maxima greater than a tunable threshold δ were detected. The locations of the resulting peaks were considered to be onsets.

3 Experimental setup

3.1 Dataset

To evaluate the capability of ESNs for onset detection, the dataset introduced by Böck in [12] was used. It consists of around 102 min of audio files sampled at 44.1 kHz and 27 700 annotated onsets. The database is already split into eight subsets for an 8-fold cross validation. We used six subsets to train the ESN and one subset as a validation set to tune the hyperparameters. After fixing the hyperparameters, the final model was trained using seven subsets and tested on the eigth unseen subset.

The audio data consists of all important types of onsets, e.g. hard, soft and complex mixtures. The excerpts are from various genres, such as rock, pop, jazz and classical music. Because the dataset was already used for several evaluations of algorithms for onset detection, we could directly compare the results of our cross-validation with state of the art algorithms, e.g. the CNN-Onset-Detector [7], which is the best performing algorithm.

3.2 Measurements

We compared our results with the state of the art algorithms and report different error measures using the `madmom`-library [13] with the same settings as used in [7]. Therefore, the detected onset times were compared to the reference onset times. If an onset was detected in a time-window of ± 25 ms around a reference, it was considered as a true positive (TP). If no onset was detected in the window around a reference, it was considered as a false negative (FN). If any

onset was detected outside the window, it was a false positive (FP). With these notations, the following three measurements can be defined:
The Precision P is the ratio of all correctly detected onsets to all detected onsets. It is 1.0 if no false positives were recognized, and 0.0 if no correct onsets were recognized at all.

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{4}$$

The Recall R is the ratio of all correctly detected onsets to all correctly detected and forgotten onsets. It is 1.0 if no false negatives were recognized, and 0.0 if no correct onsets were recognized at all.

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5}$$

The F-measure F is the harmonic mean of Precision and Recall.

$$F = 2 \cdot \frac{P \cdot R}{P + R} \tag{6}$$

In this paper, F served as the objective function to determine the peak picking threshold δ.

3.3 Implementation and optimization strategy

The algorithm was developed in Python 3. Table 1 shows the hyper-parameters to be optimized and the final result. The optimization process was conducted using a sequence of grid and line searches.

Hyperparameter	Range	Step	Final value
Input scaling α_{U}	$[0.0, 1.5]$	0.1	0.3
Spectral radius ρ	$[0.0, 1.5]$	0.1	0.7
Bias scaling α_{B}	$[0.0, 1.5]$	0.1	0.1
Leakage λ	$(0.0, 1.0]$	0.1	1.0
Threshold δ	$[0.2, 0.4]$	0.02	0.3

Table 1 – Overview over all hyperparameters to be tuned. The values show the search range and the step size, in which the exhaustive grid search took place. The final values were fixed for the evaluation.

The optimization workflow to fix the ESNs hyperparameters consisted of three steps:

1. The starting point was a grid search across α_{U} and ρ. These two hyperparameters needed to be optimized together to determine a trade-off between forward and recurrent connections. Therefore, α_{B} and λ were fixed to their default values 0.0 and 1.0.

2. Next, a line search was conducted to optimize α_{B}, while the default value for λ was kept constant. This parameter changes the default operating point of the non-linear neurons in the reservoir, which led to additional non-linearity for the system. Thus, for more non-linear tasks, we expect α_{B} to increase.

3. Finally, λ was optimized using a line search. This parameter determines the different temporal evolutions of the input compared to the output.

For every parameter combination during this optimization workflow, the cross-correlation between the target and the computed output was reported on the validation set. This was done separately for each of the eight folds. Next, the mean cross-correlation over all folds was computed. After each step, the hyper-parameters leading to the highest mean cross-correlation were fixed and used for the optimization of the next parameter.

The reservoir size was fixed to 500 during this optimization process. It has been shown that the reservoir size tends to be independent from all other hyperparameters [14, 10]. For the later evaluation, it was increased up to 8000 neurons.

After fixing all hyper-parameters, the peak picking threshold was found by a line search to maximize the F-measure.

4 Results

Table 2 presents the detection result of the proposed ESN-based model with different reservoir sizes and the current state of the art models. The results show that increasing the reservoir size clearly improved the recognition result. Using a bidirectional instead of an unidirectional reservoir also increased the F-measure because of more free parameters. For now, the bidirectional model with 8000 reservoir neurons performs best. Compared to the state of the art [6] and [7], which use BLSTMs and CNNs, respectively, the ESN still has a lower F-measure.

Model	Threshold	unidirectional			bidirectional		
N^{res}	δ	P	R	F	P	R	F
500	0.28	0.843	0.694	0.761	0.841	0.716	0.773
1000	0.28	0.852	0.720	0.781	0.872	0.721	0.789
2000	0.28	0.855	0.744	0.796	0.861	0.747	0.800
4000	0.3	0.879	0.740	0.804	0.860	0.762	0.808
5000	0.3	0.877	0.741	0.803	0.857	0.764	0.808
8000	0.3	0.870	0.750	0.806	0.854	0.774	0.812
BLSTM [6]	–				0.892	0.855	0.873
CNN [7]	–				0.917	0.889	0.903

Table 2 – P, R and F for different models evaluated using the 8-fold cross validation. The reference models were evaluated on the same dataset by the authors.

Figure 2 visualizes the input features, ground truth, ODF and detected onsets of the song "sb_Albums-Chrisanne2-01(16.0-26.0)" from the test set. It was a typical example from the test set and obtained with the unidirectional model with 500 reservoir neurons. Obviously, many onsets were forgotten in this example. However, the number of false positives is quite small. It is almost representative for the current performance of the smallest model.

5 Conclusions and outlook

We presented a new approach for onset detection in musical signals based on ESNs. The results show that this very first attempt still falls short to the state of the art, but is promising given

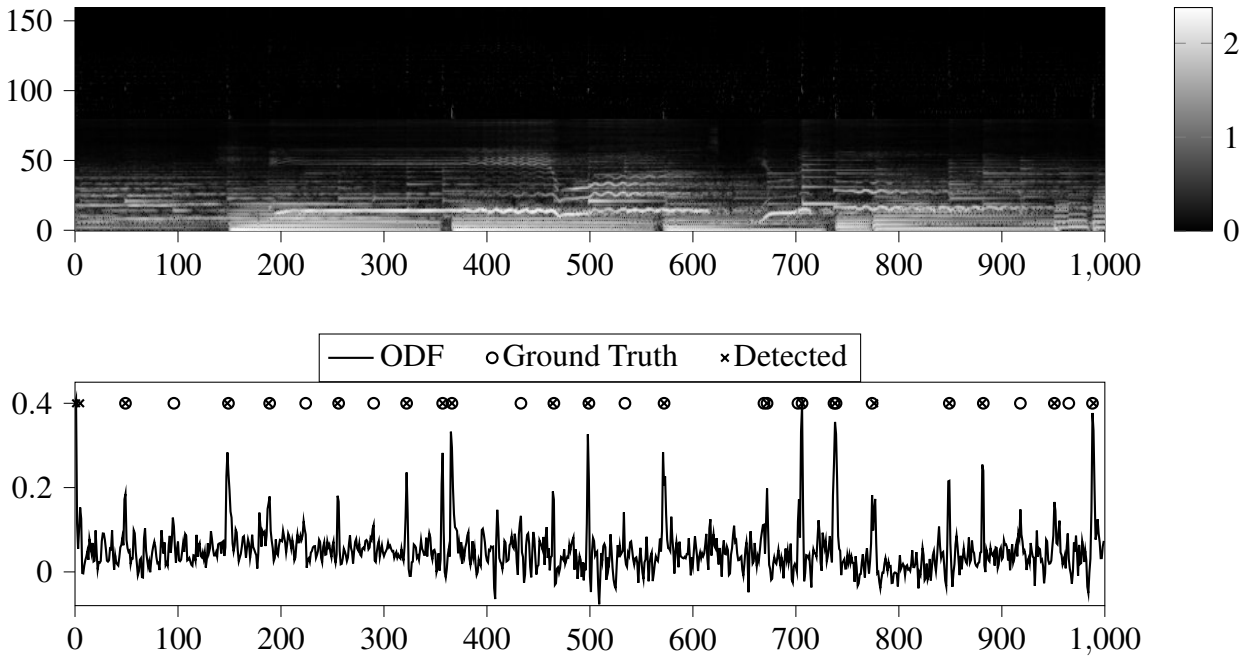

Figure 2 – Input features, ODF, ground truth and detected onsets for the example "sb_Albums-Chrisanne2-01(16.0-26.0)" from the test set. The unidirectional model with 500 reservoir neurons achieved an F-measure of 0.750. There were in total 18 correct, 2 false positive and 10 false negative decisions.

the early stage of the investigations into ESNs. There are a lot of improvements that can be incorporated in the current system. We noticed that the F-measure did not yet stop increasing with the current number of neurons. Thus, the number of reservoir neurons can still be increased until the reservoir starts overfitting on the training data. Furthermore, the input features can be expanded. Both state of the art systems are using a kind of multi-resolution features, e.g., the feature extraction as described in this paper is done with three different window sizes for the FFT. Furthermore, we did not incorporate any kind of feature normalization. In [9, 10, 11], multiple reservoirs are stacked and it has been shown that additional reservoirs can correct errors from previous layers.

6 Acknowledgements

The parameter optimizations were performed on a Bull Cluster at the Center for Information Services and High Performance Computing (ZIH) at TU Dresden.
This research was financed by Europäischer Sozialfonds (ESF) and the Free State of Saxony (Application number: 100327771).

References

[1] DIXON, S.: *Onset detection revisited.* In *Proceedings of the International Conference on Digital Audio Effects (DAFx-06)*, pp. 133–137. Montreal, Quebec, Canada, 2006. URL `http://www.dafx.ca/proceedings/papers/p_133.pdf`.

[2] PERTUSA, A., A. KLAPURI, and J. M. IÑESTA: *Recognition of note onsets in digital music using semitone bands*. In A. SANFELIU and M. L. CORTÉS (eds.), *Progress in Pattern Recognition, Image Analysis and Applications*, pp. 869–879. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[3] ZHOU, R. and M. MATTAVELLI: *A new time-frequency representation for music signal analysis: Resonator time-frequency image*. In *2007 9th International Symposium on Signal Processing and Its Applications*, pp. 1–4. 2007. doi:10.1109/ISSPA.2007.4555594.

[4] MAROLT, M., A. KAVCIC, and M. PRIVOSNIK: *Neural networks for note onset detection in piano music*. In *Proceedings of the 2002 International Computer Music Conference*. 2002.

[5] LACOSTE, A. and D. ECK: *A supervised classification algorithm for note onset detection*. *EURASIP Journal on Advances in Signal Processing*, 2007(1), p. 043745, 2006. doi:10.1155/2007/43745. URL https://doi.org/10.1155/2007/43745.

[6] EYBEN, F., S. BÖCK, B. W. SCHULLER, and A. GRAVES: *Universal onset detection with bidirectional long short-term memory neural networks*. In *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, Netherlands, August 9-13, 2010*, pp. 589–594. 2010. URL http://ismir2010.ismir.net/proceedings/ismir2010-101.pdf.

[7] SCHLÜTER, J. and S. BÖCK: *Musical onset detection with convolutional neural networks*. In *6th International Workshop on Machine Learning and Music (MML), Prague, Czech Republic*. 2013.

[8] JAEGER, H.: *The "echo state" approach to analysing and training recurrent neural networks*. Tech. Rep. GMD Report 148, German National Research Center for Information Technology, 2001. URL http://www.faculty.iu-bremen.de/hjaeger/pubs/EchoStatesTechRep.pdf.

[9] TRIEFENBACH, F., A. JALALVAND, K. DEMUYNCK, and J.-P. MARTENS: *Context-dependent modeling and speaker normalization applied to reservoir-based phone recognition*. In *Proc. Interspeech 2013*, pp. 3342–3346. 2013.

[10] JALALVAND, A., K. DEMUYNCK, W. D. NEVE, and J.-P. MARTENS: *On the application of reservoir computing networks for noisy image recognition*. *Neurocomputing*, 277, pp. 237 – 248, 2018. doi:https://doi.org/10.1016/j.neucom.2016.11.100. URL http://www.sciencedirect.com/science/article/pii/S0925231217314145. Hierarchical Extreme Learning Machines.

[11] STEINER, P., A. JALALVAND, and P. BIRKHOLZ: *MIREX 2019: Multiple-f0 Estimation using Echo State Networks*. 2019. URL https://www.music-ir.org/mirex/abstracts/2019/SBJ1.pdf.

[12] BÖCK, S., F. KREBS, and M. SCHEDL: *Evaluating the online capabilities of onset detection methods*. In *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*, pp. 49–54. 2012. URL http://ismir2012.ismir.net/event/papers/049-ismir-2012.pdf.

[13] BÖCK, S., F. KORZENIOWSKI, J. SCHLÜTER, F. KREBS, and G. WIDMER: *Madmom: A new python audio and music signal processing library*. In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 1174–1178. ACM, 2016.

[14] TRIEFENBACH, F., K. DEMUYNCK, and J.-P. MARTENS: *Large vocabulary continuous speech recognition with reservoir-based acoustic models*. *IEEE Signal Processing Letters*, 21(3), pp. 311 – 315, 2014. doi:10.1109/LSP.2014.2302080.