POLYPHONIC PIANO NOTE TRANSCRIPTION WITH RECURRENT NEURAL NETWORKS

Sebastian Böck, Markus Schedl

Department of Computational Perception, Johannes Kepler University, Linz, Austria
sebastian.boeck@jku.at

ABSTRACT

In this paper a new approach for polyphonic piano note onset transcription is presented. It is based on a recurrent neural network to simultaneously detect the onsets and the pitches of the notes from spectral features. Long Short-Term Memory units are used in a bidirectional neural network to model the context of the notes. The use of a single regression output layer instead of the often used one-versus-all classification approach enables the system to significantly lower the number of erroneous note detections. Evaluation is based on common test sets and shows exceptional temporal precision combined with a significant boost in note transcription performance compared to current state-of-the-art approaches. The system is trained jointly with various synthesized piano instruments and real piano recordings and thus generalizes much better than existing systems.

Index Terms— music information retrieval, neural networks

1. INTRODUCTION

Music transcription is the process of converting an audio recording into a musical score or a similar representation. In this paper we concentrate on the transcription of piano notes, especially on the two most important aspects of notes, their pitch and onset times. To detect them as accurately as possible is crucial for a proper transcription of the musical piece. We leave out higher level tasks like determining the length of a note (given either in seconds or in a musical notation like quarter note). Also we do not consider the velocity or intensity. The output of the system is a simplified piano-roll notation of the audio signal.

Traditional music transcription systems are based on a wide range of different technologies, but all have to deal with the subtasks of estimating the fundamental frequencies and the onset locations of the notes. A very basic approach formulated by Dixon [1] solely relies on the spectral peaks of the signal to detect notes; local maxima represent the onsets and the drop of energy below a minimum threshold marks the offset of the note. Bello et al. [2] additionally incorporate time-domain features to predict multiple sounding pitches assuming that the signal can be constructed as a linear sum of individual waveforms based on a database of piano notes. Raphael [3] proposes a probability-based system which uses a hidden Markov model (HMM) to find chord sequences. The states are represented by frames with labels based on the sounding pitches. Ryynänen and Klapuri [4] also use HMMs to model note events based on multiple fundamental frequency features. Transition between notes are controlled via musical knowledge.

Most of today's top performing piano transcription systems rely on machine learning approaches. Marolt [5] describes an elaborate approach based on different neural networks to recognize tones in an audio recording, combined with adaptive oscillators to track partials. Poliner and Ellis [6] use multiple support vector machine (SVM) classifiers trained on spectral features to detect the sounding fundamental frequencies of a frame. Post-processing with HMM is applied to temporally smooth the output. Boogaart and Lienhart [7] use a cascade of boosted classifiers to predict the onsets and the corresponding pitches of each note. All these systems use multiple classifiers and thus can not reliably distinguish whether a sounding pitch is the fundamental frequency of a note or a partial of another one. This results in lots of false note detections. In contrast, our system uses a single regression model and is thus able to distinguish between these states and hence lowers the number of false detections significantly.

2. SYSTEM DESCRIPTION

Figure 1 shows the proposed piano transcription system. It takes a discretely sampled audio signal as its input. The signal is transferred to the frequency domain via two parallel *Short-Time Fourier Transforms* (STFT) with different window lengths. The logarithmic magnitude spectrogram of each STFT is then filtered to obtain a compressed representation with the frequency bins corresponding to the tone scale of a piano with a semitone resolution. This representation is used as input to a bidirectional Long Short-Term Memory (BLSTM) recurrent neural network. The output of the network is a piano-roll like representation of the note onsets for each MIDI note.
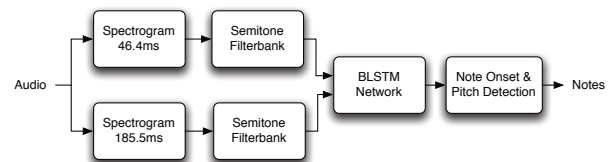

Fig. 1: Proposed piano transcription system overview.

2.1. Feature extraction

As input, the system takes a monophonic pulse code modulated (PCM) audio signal $x(n)$ with a sampling rate of $f_s = 44.1\,\text{kHz}$ in floating point representation (in the range of $[-1...1]$). The signal is split into overlapping frames with frame lengths of 2048 and 8192 samples (46.4 and 185.8 ms). Different frame lengths have been chosen to achieve both a good temporal precision and a sufficient frequency resolution for the transcription of the notes. Two consecutive frames are located 10 ms apart, resulting in a constant frame rate $f_r = 100\,\text{fps}$. A Hamming window with the same size as the frame is applied before the signals are transferred to the frequency domain with the Short-Time Fourier Transform.

Phase information of the resulting complex spectrograms $X(n, k)$ is omitted, and only the magnitude values are used for all further calculations. A logarithmic representation of the magnitude spectrograms is advantageous, compared to the linear one. To avoid negative values and to put the values in a suitable range for the following neural network stage, the spectrograms are multiplied with a factor 1000 and a fixed value of 1 is added before taking the logarithm. These values yielded the best results during preliminary tests.

To reduce the dimensionality of the input vector of the neural network, the two magnitude spectrograms $S(n, k)$ are filtered with semitone filterbanks $F(m, k)$. The frequencies m are spaced equally on a logarithmic frequency scale and are aligned according to the pitches of the 88 MIDI notes (i.e., semitone spacing). This spacing is expanded up to the maximum frequency of 16 kHz. Overlapping triangular filters are used to combine multiple spectrogram frequency bins into one. The area of each filter is normalized to 1 to compensate the overemphasis of high frequencies. Finally duplicate filters (which occur if the frequency resolution of the STFT is too coarse for low MIDI pitches) are eliminated. This results in a dimensionality reduction from 5120 values of the two spectrograms down to 183.

The use of a semitone spacing instead of a less granular one (e.g., quarter tone) has two main advantages: first, it reduces the dimensionality of the input vector for the neural network by roughly a factor of two, thus resulting in reduced training time. It also desensitizes the whole system against minor tuning variations of different pianos, hence leading to a much better generalization without the need for a manual adjustment of the piano tuning.

Since the energy of the signal rises during the note attack phase which directly follows the note onset, also the first order differences of the semitone filtered spectrograms are included. For the small window length, the difference is calculated to the preceding frame, whereas for the long window length it is calculated relative to the frame at the index $n - 4$. This measure cancels the delay of the rise in energy relative to the actual note onset position. Although adding the first order differences doubles the input vector size of the neural network from 183 to 366, it increases the overall transcription performance and simultaneously reduces the needed training epochs, since the network converges faster.

2.2. Neural Network

For the neural network stage, a bidirectional *recurrent neural network* (RNN) with *Long Short-Term Memory* (LSTM) units is used. Compared to *feed forward neural networks* (FNNs), RNNs have the advantage that they are able to model temporal contexts due to the use of recurrent connections in the hidden layers. Although theoretically able to remember any past values, they suffer from the vanishing gradient problem, i.e., input values decay or blow up exponentially over time, thus limiting their range to a maximum of a few time steps. Hochreiter and Schmidhuber [8] developed a new method called LSTM to overcome this problem. Each LSTM block has a recurrent connection with weight 1.0 which enables the block to act as a memory cell. Input, output, and forget gates control the content of the memory cell through multiplicative units and are connected to other neurons as usual.

A *bidirectional recurrent neural network* (BRNN) doubles the number of hidden layers and presents the input values to the newly created set of hidden layers in reverse temporal order. This offers the advantage that the network not only has access to past input values but can also 'look into the future'.

If BRNNs are used in conjunction with LSTM neurons, a *bidirectional Long Short-Term Memory* (BLSTM) recurrent neural network is built. It has the ability to model a wider temporal context around a given input value. For the detection of notes this is an essential feature, since the onset is not only characterized by an increase in energy during the attack phase, but also by a special energy envelope during the following decay, sustain, and release phases.

BLSTMs have been successfully implemented in systems for onset detection [9] and beat detection and tracking [10] which both showed state-of-the-art performance in their respective field. In contrast to those implementations, the neural network of this approach uses a regression output layer. The biggest advantage compared to multiple classifier system [5, 6, 7] lies in the ability of the system to correctly identify whether a sounding pitch is the fundamental frequency of a note or a partial of another one, thus reducing the number of false positive and negative note detections significantly.

The used neural network has three bidirectional hidden layers with 88 LSTM units each. The regression output layer has 88 units, each representing one MIDI pitch. The output of these units represent the activation functions for each note.

2.2.1. Network training

The network is trained with supervised learning and early stopping. The used training data set is described in Section 3. Together with the target values extracted from the MIDI data, each audio sequence is preprocessed as described above and presented to the network for learning. The network weights are initialized with random values following a Gaussian distribution with mean 0 and standard deviation 0.1. Standard gradient descent with backpropagation of the errors is used to train the network. To prevent over-fitting, performance is evaluated after each training iteration on the validation set. If no improvement on the summed squared error is observed for 20 epochs, the training is stopped.

122

2.2.2. Network testing

For the evaluation of the system, the unknown music excerpts of the test set are preprocessed as described in Section 2.1 and presented to the previously trained network. The resulting note activation regression matrix of the output nodes is used as input to the following stage.

2.3. Note onset and pitch detection

The notes onset times and pitches are derived directly from the neural network output. The activation values for each pitch are smoothed with a Hamming window of 90 ms length before being thresholded. The length of the window is not crucial as long as it is smaller than the duration between two consecutive notes of the same pitch. The threshold is determined individually per note on the validation set by: $\theta_p = \arg\max_\theta \{TP_\theta - FP_\theta - FN_\theta\}$, TP denoting true positive, FP false positive, and FN false negative detections. A standard local maximum peak picking algorithm is applied to gather the final note onset positions for each pitch.

3. DATA

Solo piano music has been chosen for training and evaluation of the described system. As a basis, the musical renderings and recordings of the MAPS database[1] introduced by Emiya [11] are used. They consist of 209 pieces rendered by seven different software synthesizers and 60 real piano recordings with an upright Yamaha Disklavier. To expand the dataset, 267 MIDI files from the same source, the Classical Piano Midi Page[2], were synthesized with the freely available Maestro Concert Grand v2[3] sound font. To compensate the emphasis towards synthesized sounds, the LabROSA Disklavier recordings[4] (used for evaluation in [6]) and real audio recordings of 13 Mozart sonatas played on a Bösendorfer SE290 computer monitored grand piano by the pianist Roland Batik were added to the set. The whole dataset is split into training, validation, and testing examples according to the original splitting in [6], thus maintaining the comparability of the results. Table 1 shows the distribution of the dataset.

Number of notes	training	validation	testing
MAPS (MIDI instruments)	854507	108778	107310
MAPS (Disklavier)	86026	16495	5675
MIDI (Maestro Concert)	519479	59838	71225
Batik (Bösendorfer)	76095	13387	16926
LabROSA (Disklavier)	47134	0	23298

Table 1: Note distribution of the datasets.

---

[1] http://www.tsi.telecom-paristech.fr/aao/en/2010/07/08/
[2] http://www.piano-midi.de
[3] http://www.linuxsampler.org/instruments.html
[4] http://labrosa.ee.columbia.edu/projects/piano/

4. RESULTS

To measure the performance of our system, standard precision, recall, and f-measure scores are used. Another measure is the accuracy, defined by Dixon in [1]. Since it counts false detections twice (both the false negative and the false positive detection), error scores as used by Poliner [6] are provided additionally; E_{subs} denote note substitutions, E_{miss} missed notes, E_{fa} false additions, and E_{tot} the sum of all errors.

4.1. Note Onset Transcription

An onset is considered as correctly identified if its pitch is correctly identified and its location is within a certain window around the ground-truth position. For onset detection usually a 100 ms window is used, and the results given in [6] use this window length as well. [7] uses a window of 68.25 ms. Although penalizing our system, we give results only for a detection window of 50 ms.

Dataset [%]	Acc	E_{tot}	E_{subs}	E_{miss}	E_{fa}
MAPS (MIDI)	84.0	15.3	2.9	6.8	5.6
MAPS (Disklavier)	68.7	32.6	6.6	17.0	9.0
MIDI	88.9	9.9	2.0	4.0	3.9
Batik	90.1	9.9	0.3	6.4	3.2
LabROSA	62.7	39.3	9.6	17.8	11.9
complete	85.6	13.7	2.5	6.1	5.1
complete (w/o octave)	89.7	9.2	2.5	5.3	1.4
Poliner [6]	62.3	43.2	4.5	16.4	22.4
Boogaart [7]	87.4	-	-	-	-

Table 2: Note onset transcription accuracy and error rates for the partial and complete test sets.

Table 2 shows the accuracy and error rates for the different test sets compared to other state-of-the-art systems. The new approach clearly outperforms the system of Poliner and Ellis [6] not only in case of the complete test set, but even for the most difficult partial test set (the LabROSA Disklavier recordings). This does not only show the good performance of our approach, but also highlights its good generalization capability. Concurrent with the rise in accuracy, all error score are significantly lower. This demonstrates the ability of the system to detect even difficult notes without adding a high number of false detections.

If only a single instrument is evaluated (i.e., the MIDI test set), the systems also performs better than the one of Boogaart and Lienhart [7], which was trained with a single MIDI instrument. This is remarkable, since our system is not trained specifically for a single instrument. Trained solely on the MIDI dataset, our system achieves an accuracy of 93.6%, exceeding their results even further.

The much better result for the Batik test set can be explained by the lower musical complexity of the Mozart sonatas compared to other musical pieces of the test set.

Tonal misalignments can have different impacts on human perception. A pitch error of an octave does usually not harm the overall impression very much, whereas other transcription errors can spoil a musical piece completely. Therefore Table 2

123

also adds results if octave errors are not considered. Since the number of errors are almost evenly distributed across all test sets, only one result for the complete test set is given. Not counted are errors occurring due to notes being added exactly one octave below or above the correct pitch, or notes that are missed if there is a detection exactly one octave apart. It can be seen that more than 70% of the spuriously added notes are pitched exactly one octave aside and hence do not harm the musical perception much.

4.2. Temporal resolution

According to Handel [12], 5 ms is the threshold of perceptual difference for musical performances. For piano transcription it is therefore highly desirable to achieve the maximum possible temporal precision. Table 3 shows the precision, recall, f-measure and accuracy results for different detection window sizes on the whole test set. The system achieves roughly the same performance for all detection windows down to a length of 50 ms. If the window size is reduced to 30 ms (which corresponds to a maximum deviation of the detection by a single frame in both directions at the used frame rate), the performance starts to decrease. Even if only the annotated frame is used for evaluation, the system is still performing decently, highlighting the exceptional temporal precision. It should be noted that the Disklavier recordings sometimes have annotation inaccuracies of up to 15 ms, which explains the much lower result if only one frame is considered.

Detection size	Precision	Recall	F-measure	Accuracy
100 ms	0.936	0.917	0.927	86.3%
70 ms	0.935	0.917	0.926	86.2%
50 ms	0.933	0.915	0.924	85.9%
30 ms	0.924	0.906	0.915	84.4%
10 ms	0.738	0.723	0.730	57.5%

Table 3: Note onset transcription results for the complete test set measured with different window detection sizes.

The use of multiple spectrograms enables the algorithm to achieve such a good temporal performance. Inspection of the internal states of the neural network shows that the network gathers timing information almost exclusively from the spectrogram and the differences obtained with the shorter STFT window length, while the information needed to determine the pitch of a note (especially the lower pitched ones) is mostly obtained from the spectrogram with the longer window.

5. CONCLUSION

In this paper we presented a new piano transcription system which is a significant step towards real audio-to-MIDI transcription. It gives exceptional temporal precision paired with state-of-the-art note onset and pitch transcription performance.

The evaluation on publicly available test sets shows that our approach greatly reduces both the number of false positive and negative note detections. The reduction is mainly due to the use of a single regression output layer to simultaneously detect note onsets and pitches compared to the one-versus-all classification approaches. Only holistic systems can decide whether a sounding frequency is the real fundamental frequency or a harmonic overtone of another note.

Furthermore our system generalizes very well over a wide range of various pianos, resulting in transcription results previously only achieved by systems tuned specifically for a single instrument.

7. REFERENCES

[1] S. Dixon, "On the computer recognition of solo piano music," in *Proceedings of the Australasian Computer Music Conference*, 2000.

[2] J.P. Bello, L. Daudet, and M.B. Sandler, "Automatic piano transcription using frequency and time-domain information," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 6, November 2006.

[3] C. Raphael, "Automatic transcription of piano music," in *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, 2002.

[4] M.P. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005.

[5] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, vol. 6, 2004.

[6] G.E. Poliner and D.P.W. Ellis, "A discriminative model for polyphonic piano transcription," *EURASIP J. Appl. Signal Process.*, January 2007.

[7] C.G.v.d. Boogaart and R. Lienhart, "Note onset detection for the transcription of polyphonic piano music," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2009)*, July 2009.

[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computing*, vol. 9, no. 8, 1997.

[9] F. Eyben, S. Böck, B. Schuller, and A. Graves, "Universal onset detection with bidirectional long short-term memory neural networks," in *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR 2010)*, 2010.

[10] S. Böck and M. Schedl, "Enhanced Beat Tracking with Context-Aware Neural Networks," in *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx-11)*, Paris, France, September 2011.

[11] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, August 2010.

[12] S. Handel, *Listening: an introduction to the perception of auditory events*, MIT Press, 1989.