

POLYPHONIC PITCH DETECTION WITH CONVOLUTIONAL RECURRENT NEURAL NETWORKS

Carl Thomé

DoReMIR Music Research AB
carl.thome@doremir.com

Sven Ahlbäck

DoReMIR Music Research AB
sven.ahlback@doremir.com

ABSTRACT

Recent directions in automatic speech recognition (ASR) research have shown that applying deep learning models from image recognition challenges in computer vision is beneficial. As automatic music transcription (AMT) is superficially similar to ASR, in the sense that methods often rely on transforming spectrograms to symbolic sequences of events (e.g. words or notes), deep learning should benefit AMT as well. In this work, we outline an online polyphonic pitch detection system that streams audio to MIDI by ConvLSTMs. Our system achieves state-of-the-art results on the 2007 MIREX multi-F0 development set, with an f-measure of 83% on the bassoon, clarinet, flute, horn and oboe ensemble recording without requiring any musical language modelling or assumptions of instrument timbre.

1. INTRODUCTION

Deep learning has dramatically improved the state-of-the-art in speech recognition [20] and it seems likely therefore to be a promising framework for music information retrieval (MIR) tasks like automatic music transcription (AMT), as AMT shares many similarities with ASR in the sense that methods often rely on estimating sequences of events from input sequences with fixed sampling rates such as spectrograms (Figure 2).

[17] showed that deep neural networks can learn to transform spectrograms to piano rolls (Figure 1) with no explicit assumptions about musical structure or instrument timbre, and particularly convolutional layers seem beneficial to avoid overfitting. [27] used skip-connected convolutional networks as described in [13] to guess active fundamentals around predetermined onsets (and then joined notes by always extending offsets to be performed as legato) with an average onset-only f-measure of 83% on the MIREX 2016 solo piano note tracking task.

However, as music is sequential it should be beneficial to promote temporal dependencies in the model architecture and previous studies have used recurrent neural networks (RNNs) to do so [5] with average onset-only

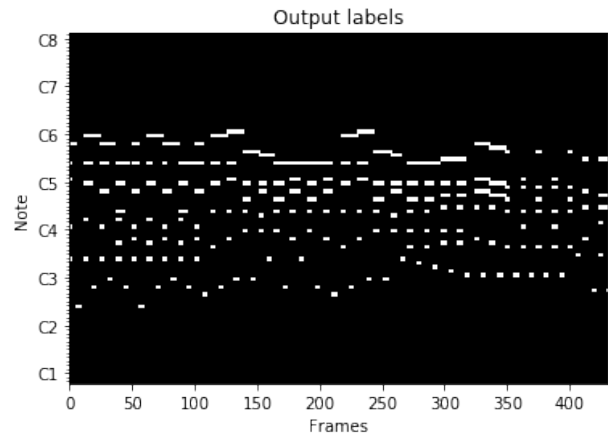


Figure 1. Example of an output piano roll with MIDI notes drawn over time frames.

f-measures of 92% for piano recordings. However, a problem with RNNs is that the recurrence is fully-connected [26], meaning deeper networks will overfit, and essentially memorize timbre and/or melodies as spatial structure is discarded. To circumvent overfitting [5] used semitone filterbanks, but we believe that distinguishing pitch in polyphonic signals requires a high frequency resolution such that spectral peaks don't overlap. Essentially the spectrogram has to be invertible for us to believe that a model gets a fair shot at learning.

This work's main goal is to propose a combination of convolutional neural networks and recurrent neural networks to perform polyphonic pitch detection without requiring musical language modelling or assumptions of instrument timbre. This approach is inspired by the nature of human pitch perception, which occurs on a sensory level and can be regarded fundamentally a psychoacoustical phenomenon

2. RELATED WORK

Many variants of combining convolutional neural networks with recurrent neural networks have been tried [1, 7], often by stacking networks such that the output of a convolutional model is directly fed into a recurrent model, and trained jointly. This is straightforward but still discards spatial structure in the recurrent part of the network. Instead, [28] proposed replacing the recurrent connection with convolutions (ConvLSTM), and applied a seq2seq



variant to precipitation nowcasting¹. ConvLSTMs were later applied to spectrograms in ASR [29], although they pool over time before the RNN as their output is only a few symbols per sample (words in sentences) compared to the output resolution of piano rolls (hundreds of frames in order to put notes at accurate onsets). Despite needing high time resolution, we choose to work with piano rolls to avoid having an encoder and decoder, and to benefit from forcing models to choose a MIDI note number regardless of tuning and inharmonicity, although future work will look into attention seq2seq models where the output is a list of pitch events instead of piano rolls that needs to be peak picked from.

Additionally, [29] used rectified linear units (ReLUs) [23] and batch normalization [16] which means they might have been hindered by trouble described in [9]. They settled on only normalizing input connections and not the recurrence, as proposed by [1]. For seq2seq this might not pose a problem, but for polyphonic pitch detection it's important to normalize feature maps over time because the output is essentially a direct translation of the input with a recoloring of the spectrogram into a binary piano roll.

3. PROBLEM FORMULATION

MIREX makes a distinction between estimating active fundamental frequencies with fixed sampling rate, called "framewise evaluation", and estimating note events with determined start and stop times, called "note tracking". Both tasks are evaluated separately. In this submission we're concerned with determining note events on the MIDI scale, instead of tracking pitch contours, although the latter will be investigated in future work for vibrato tracking, note-level instrument recognition and overlapping polyphonicity in ensemble recordings (not representable in piano rolls).

For both note tracking and framewise multi-f0 estimation, two things make the problem hard:

1. When multiple notes are played together all tones are mixed together in the same audio signal, so there is more noise and it can be hard to distinguish fundamentals from partials, particularly when partials overlap to create false fundamentals.
2. Human perception of pitch is complicated. Pitch often corresponds to the fundamental frequency of a sound wave, or the presence of a harmonic series, but not always. Timbre is also important for pitch perception [24].

In other words, for AMT it can be better to have a pitch detector that adds non-existing frequencies, because humans generally perceive pitch categorically as stable note events, e.g. we hear notes in melodies and unconsciously modify the sound to fit musical expectations.

¹ Loosely speaking, precipitation nowcasting is the problem of predicting if it will rain from satellite images.

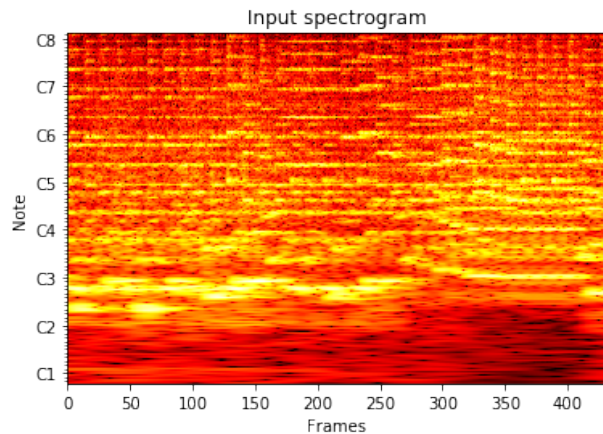


Figure 2. Example of an input spectrogram where the audio has been constant-Q transformed. Note that only log-scaled power is shown as predicting phase is still an open question in MIR research.

4. METHOD

4.1 Preprocessing

We use the real-time variant of a non-stationary Gabor transform where Q is constant called sliCQ [15] (Figure 2), but base our implementation on the time-domain implementation of the constant-Q transform (CQT) in [22]. The reasoning for relying on CQT is that it promotes pitch invariance compared to an ordinary short-time Fourier transform (STFT) [3], which is important for convolutional neural networks that share weights over the frequency axis. As the presence of partials is also important for pitch perception we stack multiple spectrograms in the depth dimension as in [4], although we only use harmonics 1, 2, 3 and 4. All spectrograms are log-scaled power spectrograms so convolution kernels can suppress noise as well as detecting harmonics (which are often a lot louder).

4.2 Model

The model is a deep neural network that transforms spectrograms (Figure 2) to piano rolls (Figure 1). We treat the occurrence of a new note onset as orthogonal to the presence of pitch, so we have two separate output channels, one for articulation and one for sustain (Section 4.3). The sustain channel is zero everywhere except during note events where it is set to ones (i.e. a typical piano roll representation). The articulation channel has the additional constraint that it is only one at the start of a note event (i.e. the first-order derivative of the sustain channel, without negative values).

The network performs a succession of convolution and average pooling operations in frequency to remove inharmonicity and timbre. The following recurrent layers have fully-connected matrix multiplies replaced with convolutions to preserve spatial structure in each feature map. We then upsample over time and apply additional convolutions to account for spectrogram frame overlap. Scaled exponential linear units are used as activation functions

and scaled hyperbolic tangents are used in the recurrence [8, 19]. Layer normalization [2] is used over time, which unlike [9] has the benefit of not requiring too many parameters. As gating we use long short-term memory cells [14]. We have also experimented with exchanging the ConvLSTM with a ConvGRU. The GRU [6] has been shown in [12] to work as effectively as the LSTM [14] but it is our belief that uncoupled forget gates are important for mapping spectrograms to piano rolls as songs consist of different sections that can change suddenly, so we keep all LSTM gates with uncoupled forget gates [10], and also add peepholes [11] for the same reason. Skip-connections [13] are added around each recurrent, pooling and upsampling layer to encourage later layers to make adjustments early in the training, and were preferred over highway networks because gradient flow is improved without extra parameters. We also believe that the residual style is particularly suited for suppressing overtones. The final layer is a network-in-network (i.e. 1x1 convolution) with sigmoid outputs as in [17, 27].

Training was done with Adam [18] and truncated back-propagation through time with RNN state passthrough across minibatches. Parameters were initialized with orthogonal initialization [25].

4.3 Postprocessing

Finally, we peak pick the two-channel activation matrix to convert the framewise piano roll to a list of note events. Per note, we step through each time frame and place an onset at positions where the articulation channel is above a set threshold, and then include all frames onward until the sustain channel is under another fixed threshold, at which point we output an offset. If a new articulation is found during an active note event we simply fragment it by outputting additional offsets and onsets.

5. RESULTS

To be announced as MIREX 2017 is reported.

6. CONCLUSIONS AND FUTURE WORK

Our polyphonic pitch detection system gets state-of-the-art results on the 2007 MIREX multi-F0 development set, with an f-measure of 83% on the bassoon, clarinet, flute, horn and oboe ensemble recording without requiring any musical language modelling or assumptions of instrument timbre.

A minor weakness of our approach is that if we want to extend our method with vibrato detection or ADSR envelopes, we would need to produce such annotations first. Therefore, for future work we are looking into complementing the deep learning approach with traditional signal processing techniques such as partial tracking [21] although we strongly believe in a data-driven, end-to-end approach that avoids hard-coded heuristics.

Still, the primary weakness of our approach is the rigidity of piano rolls. There's no natural support for same-note polyphonicity and piano rolls contain a lot of redun-

dancy. If we were able to supplant them with note events directly (e.g. seq2seq) that would go a long way [29] but it is not straightforward for AMT as there are invariances not covered in an events list because of the polyphonicity. For example, it shouldn't matter which order simultaneous note onsets are listed in (e.g. a chord). Data augmentation to represent invariances should be tried, but might inflate training times. Ultimately, we believe the major bottleneck for further progress is the piano roll representation.

7. REFERENCES

- [1] Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. *arXiv preprint arXiv:1512.02595*, 2015.
- [2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- [3] Emmanouil Benetos and Simon Dixon. A shift-invariant latent variable model for automatic music transcription. *Computer Music Journal*, 36(4):81–94, 2012.
- [4] Rachel M Bittner, Brian McFee, Justin Salamon, Peter Li, and Juan P Bello. Deep salience representations for f0 estimation in polyphonic music. In *18th International Society for Music Information Retrieval Conference, ISMIR*, 2017.
- [5] Sebastian Böck and Markus Schedl. Polyphonic piano note transcription with recurrent neural networks. In *Acoustics, speech and signal processing (ICASSP), 2012 IEEE international conference on*, pages 121–124. IEEE, 2012.
- [6] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [7] Keunwoo Choi, George Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. *arXiv preprint arXiv:1609.04243*, 2016.
- [8] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.
- [9] Tim Cooijmans, Nicolas Ballas, César Laurent, Çağlar Gülçehre, and Aaron Courville. Recurrent batch normalization. *arXiv preprint arXiv:1603.09025*, 2016.

- [10] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.
- [11] Felix A Gers, Nicol N Schraudolph, and Jürgen Schmidhuber. Learning precise timing with lstm recurrent networks. *Journal of machine learning research*, 3(Aug):115–143, 2002.
- [12] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 2016.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [15] Nicki Holighaus, Monika Dörfler, Gino Angelo Velasco, and Thomas Grill. A framework for invertible, real-time constant-q transforms. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(4):775–785, 2013.
- [16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [17] Rainer Kelz, Matthias Dorfer, Filip Korzeniowski, Sebastian Böck, Andreas Arzt, and Gerhard Widmer. On the potential of simple framewise approaches to piano transcription. *arXiv preprint arXiv:1612.05153*, 2016.
- [18] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [19] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *arXiv preprint arXiv:1706.02515*, 2017.
- [20] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [21] Robert McAulay and Thomas Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(4):744–754, 1986.
- [22] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. 2015.
- [23] 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)*, pages 807–814, 2010.
- [24] Andrew J Oxenham. Pitch perception. *Journal of Neuroscience*, 32(39):13335–13338, 2012.
- [25] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.
- [26] Siddharth Sigtia, Emmanouil Benetos, and Simon Dixon. An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5):927–939, 2016.
- [27] Daylin Troxel. Music transcription with a convolutional neural network 2016. Technical report.
- [28] SHI Xingjian, Zhouong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, pages 802–810, 2015.
- [29] Yu Zhang, William Chan, and Navdeep Jaitly. Very deep convolutional networks for end-to-end speech recognition. *arXiv preprint arXiv:1610.03022*, 2016.