PianoTranscriber – A Note-based approach for Multipitch Tracking

Peter Steiner, Simon Stone, Peter Birkholz

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

Abstract: In this paper we propose a new analysis-by-synthesis algorithm for multipitch tracking. It uses spectral base components extracted from isolated piano notes to model a spectrum. The advantage compared to other signal-based algorithms is the application of musical knowledge for the identification process. The *PianoTranscriber* uses a limited set of the 88 musical notes of the standard piano. The extracted time series for all possible notes are used for a resynthesis step. Original and resynthesized signals are compared using an adapted onset-detection function. The *PianoTranscriber* was parametrized, optimized and evaluated using subsets of the publicly available MAPS-database. For the evaluation, the *PianoTranscriber* was compared to the state-of-the-art algorithm SONIC. The results show that the *PianoTranscriber* outperformed SONIC using base components from the analyzed piano and achieved similar results using base components from other pianos.

1 Introduction

Transcription by listening to music is difficult, because the ability to identify the correct f_0 of a musical note (the so called "absolute pitch") is required. According to [1], only 1 out of 10 000 people in North America and Europe have this ability, which makes an automatic system very desirable. Thus, in recent years many approaches for Multi-Pitch-Tracking (the identification of the individual notes played at any given time) have been proposed that can be grouped into signal-based approaches [2, 3], matrix factorization [4], and neural-network-based approaches [5, 6]. Many signal-based approaches such as [2] use an auditory filterbank that mimicks the human perception of music to obtain a mid-level representation. In this mid-level-representation or directly in the unscaled magnitude spectrum [3], the most pronounced frequency component is estimated and used to model an artificial musical note of the perceived pitch. This note is iteratively subtracted from the entire signal representation. Approaches based on matrix factorization use linear basis transforms to decompose a complex music signal into its basic components, which are supposed to represent notes. Approaches based on neural networks use the time-domain or several mid-level representations as input for the neural networks, which are responsible for the classification step. In the signal-based approaches, no spectral information from recorded isolated notes has been used so far for the note identification. However, incorporating this information could be used to improve the results of signal-based approaches for Multi-Pitch-Tracking and may create a useful foundation for further tasks in Automatic Music Transcription (AMT). This work therefore proposes an algorithm called *PianoTranscriber* that incorporates musical knowlege by using base components extracted from real isolated notes of a standard piano.

2 Signal model and main idea

Like many signal-based approaches for multipitch tracking such as [2, 3], we assume a musical note model as the superposition of multiple sinusodials with harmonic relations to the fundamental frequency f_0. All components have an individual amplitude and phase. In this paper, the MIDI pitch, which assigns an integer value to every musical note, is used to describe notes. The relationship between f_0 and MIDI-pitch p is given in equation (1), where the chamber tone A with $f_0 = 440\,\mathrm{Hz}$ is assigned to the MIDI pitch 69. A standard piano with 88 notes has a MIDI-pitch-range between 21 and 108.

$$f_0\,(p) = 2^{\frac{p-69}{12}} \cdot 440\,\mathrm{Hz} \tag{1}$$

The FOURIER transform assumes that a signal is the sum of single sinusodials with individual amplitudes and phases. Similar to this assumption, the main idea of the *PianoTranscriber* is to assume that the combination of more than one note is the sum of single isolated notes. Thus, the *PianoTranscriber* takes advantage of the limited set of musical notes and use base components obtained from one reference piano to model the entire signal frame by frame with these components.

2.1 Overview of the *PianoTranscriber* algorithm

The *PianoTranscriber* compares an input signal frame by frame with note components, which are extracted from the public available MAPS database [7] that is further used for determining several parameters. The MAPS database will be described in Section 3.1 in detail. The method is similar to the FOURIER-transform, which compares an input signal with pure sinusodial components. In Figure 1, the outline of the *PianoTranscriber* is visualized. An input signal $x_{\text{orig}}\,[k]$ is windowed using a gaussian window of length 93 ms and an overlap of 83 ms and converted into a sequence of frame vectors $\vec{x}\,[k]$ with the time index k. Next, the short-term magnitude spectrum is calculated for every frame vector to obtain a sequence of spectra $\vec{X}\,[n]$ with the frequency index n. Using the highest cosine similarity between all base components and the input spectrum, the pre-dominant note is identified. The corresponding base component is subtracted from the entire input spectrum. Up to nine more notes are iteratively identified and subtracted to obtain a maximum number of ten likely notes. This differs from the proposed signal-based algorithms, in which an harmonic signal representation out of the pre-dominant f_0 is created and subtracted from the input spectrum. The result of the frame by frame spectral difference are binary time-series for all base components. These are stored in the matrix $\mathbf{C}\,[m]$. Every row represents the time-series of one base component with the new time index of the m-th frame. The time-series are refined using knowledge about minimum tone duration and cross-correlation in the time domain. Afterwards, the refined time series $\mathbf{C}'\,[m]$ are used to resynthesize a time signal $x_{\text{res},1}\,[k]$. An energy-based onset detection is performed in the input and resynthesized signals to compare these. In this way, erroneously detected notes can be removed from the time-series, which need to be refined once again. The final time series $\mathbf{C}'''\,[m]$ are used to resynthesize the final output time signal $x_{\text{res},2}\,[k]$.

2.2 Base components

The base components are calculated from the 88 isolated notes played by a reference piano in two different loudnesses *forte* and *mezzoforte*. Similar to the recognition process in Section 2.1, the audio signals of the individual notes are each split into frames of 93 ms with an overlap of

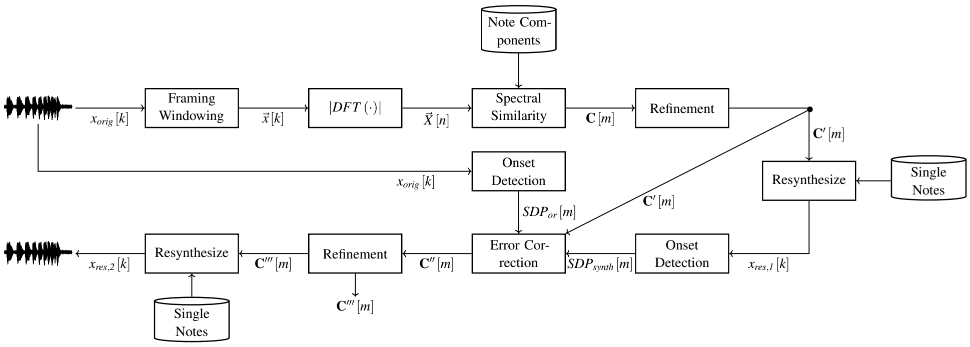

Figure 1 – The *PianoTranscriber*: An input signal $x_{\text{orig}}[k]$ is windowed and converted to a sequence of frames $\vec{x}[k]$. In the spectral domain ($\vec{X}[n]$), every frame is decomposed into its note components using the spectral difference to obtain time-series $\mathbf{C}[m]$ for every note. After refining the time-series, these are used for a resynthesis. Original and synthesized signals are compared using an energy-based onset-detection to identify further detection errors. $x_{\text{res}}[k]$ and $\mathbf{C}'''[m]$ are the final output signal and time-series.

83 ms. The FOURIER transform with a Gaussian window is calculated to obtain the magnitude spectrum for every frame. The low-energy frames close to the start and end of the signal do not contain information and are thus deleted. For each of the 5512 frequency bins between 0 Hz and 11 025 Hz, mean value and standard deviation over time are calculated from the remaining tonal frames. The mean vector $\hat{\vec{X}}[n]$ is a model spectrum for the base components and the standard deviation vector for each frequency bin provides information about the spectral flux. The *PianoTranscriber* uses in total $2 \cdot 88 = 176$ base components to compare their similarities with the input signal.

2.3 Spectral similarity for note identification

For conventional f_0-extraction from speech, autocorrelation [8], cross-correlation [9] or periodicity functions [10] are well-known methods. They all work in the lag domain, where the lags of maxima or minima are potential candidates for the true period length and one of these is finally chosen as the fundamental period based on various criteria. For multipitch tracking in music signals, one big advantage is the limited set of notes. The *PianoTranscriber* assumes that a combination of more than one note is the sum of its isolated notes. Thus, another identification approach can be used here – the spectral difference between the analyzed frame and the base components. If the difference between an analyzed frame and a base component is small, it is likely that the corresponding note was active in this frame and the similarity between the analyzed frame and the base component is high. The cosine similarity (2) is used to calculate the similarity, because it is independent of amplitudes and a correlation measure.

$$\text{sim} = \frac{\sum\limits_{n=1}^{N} \vec{X}[n]\hat{\vec{X}}[n]}{\sqrt{\sum\limits_{n=1}^{N} \vec{X}[n]^2}\sqrt{\sum\limits_{n=1}^{N} \hat{\vec{X}}[n]^2}} \tag{2}$$

Because the maximal similarity between an analysis frame and all base components needs to be determined, this results in an $\arg\max(\cdot)$-decision. Using the cosine similarity without restricting the maximum distance, leads to too many candidates for each frame. A minimum correlation of $\theta = 0.6$ was empirically found to be a good value to reduce the number of false notes without increasing the number of missing notes.

2.4 Refinement

Result of the frame based identification are binary time series for all 88 notes representing whether or not each note is present in each of the analyzed frames. Since the frame shift is only 10 ms, it is unlikely that a note would change its state (present or absent) faster than once every few frames. The MAPS database, which was used for parametrizing, contains very fast chromatic scales with a note duration of 50 ms. This would mean that one single note must appear at least in five consecutive frames. Thus we tried to find a useful minimum note duration between 30 ms and 70 ms using a parameter sweep and $t_{n,\min} = 60\,\mathrm{ms}$ was the most suitable note duration in terms of further reducing the number of missing and false notes. To take this into account, the first derivatives of all time-series are calculated. If a note is detected for the first time, the derivative is larger than zero. If it is detected for the last time, smaller than zero. If the distance between on- and offset is less than six frames, the detected note is assumed to be erroneously detected and set to zero in this interval. Too short gaps between detected notes were treated analogously.

Another step for refinement is a cross-correlation of the entire time-domain signal with the time functions of all previously detected notes. Using a note that is correctly identified, this will lead to a much larger global maximum than using an incorrectly identified note. The maxima of all detected notes will be normalized to the best global maximum. Then, a threshold for the minimum correlation ratio $\frac{M_k}{M_{\max}} = 0.08$ is defined. Every detected time series that is smaller than this ratio, is set to zero. The result are refined time-series $\mathbf{C}'[m]$.

2.5 Resynthesis and onset detection

The obtained smoothed time series $\mathbf{C}'[m]$ are used to resynthesize the signal. Assuming a perfect analysis result for every frame, it were possible to reconstruct the analyzed signal. However, in real-world scenarios it is unlikely to get a perfect frame-based result. Comparing the analyzed and resynthesized signal in this case leads to differences. Further error correction steps can minimize the difference between the analyzed and resynthesized signal.

For the resynthesis step, at first a new signal is initialized with zeros, so that original and resynthesis signal have the same length. The first derivatives of all time series $\mathbf{C}'[m]$ are used to find potential new note events. In this case, the derivative will be greater than zero. This returns the time index, where the time function of a corresponding isolated note is added. After adding all notes to the resynthesis signal, this will be normalized to a maximum amplitude of 1. Next step is an energy-based onset-detection based on (3).

$$SDP_1[m] = \sum_{n=1}^{N} H\left(|X_n[m+1]| - |X_n[m]|\right) \quad \text{with} \quad H(x) = \frac{x+|x|}{2} \tag{3}$$

$H(x)$ is a function to consider only positive energy differences, because the onset is characterized by a strong positive increase of energy. Assuming the magnitude spectrum of a periodic signal, significant spectral lines will only occur at multiples of the f_0. In between, many values are almost zero and they change randomly over time. The *PianoTranscriber* calculates the entire energy difference between two frames at first and applies $H(x)$ to this difference according to (4).

$$SDP_2[m] = H\left(\sum_{n=1}^{N} |X_n[m+1]| - \sum_{n=1}^{N} |X_n[m]|\right) \quad \text{with} \quad H(x) = \frac{x+|x|}{2} \tag{4}$$

With this small modification, the function will be only greater than zero if the entire energy increases. This modified detection function is calculated for the original ($SDP_{\text{or}}[m]$) and resynthesized ($SDP_{\text{synth}}[m]$) signal to compare them and for the further error correction using the following error function:

$$oErr[m] = SDP_{\text{or}}[m] - SDP_{\text{synth}}[m] \tag{5}$$

Two types of errors can occur – missing and additional onsets. The first is difficult to correct, because it is not yet possible to assign the missing onset to a specific note. Additional onsets can be assigned to a note by reconsidering the time series from previous steps.

If *oErr* is zero, then no error occured. If *oErr* is greater than zero and both SDP_{or} and SDP_{res}, are greater than zero, a correct onset was detected. Errors occur only if *oErr* is not equal to zero and one detection function greater than zero.If SDP_{synth} is greater than zero, an additional onset was detected, which can be matched to the time-series. There, the incorrect value is set to zero. The last step after correcting all potentially incorrect onsets, is to refine the new time-series again and to resynthesize the final output signal.

3 Evaluation

To evaluate the *PianoTranscriber*, a corpus of piano chords with increasing polyphony was analyzed. We used *PianoTranscriber* with base components obtained from the analyzed piano and from other pianos. The state-of-the-art algorithm SONIC analyzed the same corpus with standard settings, but its output of absolute frequency values was converted to MIDI pitches.

3.1 Audio material

The audio material used for parametrizing and evaluation was taken from the MAPS database [7] that contains in total nine different datasets of piano recordings with annotations of all occuring MIDI pitches within one time interval. Each dataset consists of 528 isolated notes in several playing styles, monophonic excerpts (66 repeated notes, 132 trills, 9 chromatic scales), 3498 chords in different polyphony and classical music. For parametrizing and evaluation, the dataset *AkPnBcht* was used without the loudness *piano* and classical music.

Because MIDI is a protocol for electrical musical instruments, the on- and offset times differ from real-world data. Thus, the isolated notes and chords in the dataset were manually annotated to obtain the real-world on- and offset times searching for the first and last tonal frame in the signal. To annotate monophonic excerpts, offsets were determined by observing, how long one f_0 can be measured and perceived. The audio signals are stereo recordings. The channel with the highest amplitude has been chosen to analyze mono signals.

3.2 Error measurements

Widely used error scores are taken from [11]. N_{sys} is the number of pitches reported by the system under test, N_{ref} the number of ground-truth pitches and N_{corr} the number of correctly identified pitches. E_{tot} is a score for the overall transcription quality and the sum of three components E_{subs}, E_{miss} and E_{fa}. If $N_{\text{sys}} >> N_{\text{ref}}$, E_{tot} can be greater than 1 and $N_{\text{sys}} = 0$ will lead to $E_{\text{tot}} = 1$.

$$E_{\text{tot}} = \frac{\sum\limits_{m=1}^{M} \max \big(N_{\text{ref}}[m]), N_{\text{sys}}[m]\big) - N_{\text{corr}}[m]}{\sum\limits_{m=1}^{M} N_{\text{ref}}[m]} \tag{6}$$

The substitution error E_{subs} counts the number of ground-truth pitches that are not correctly identified. It is not important, which pitch was substituted. Only the overall number of substitutions is important.

$$E_{\text{subs}} = \frac{\sum\limits_{m=1}^{M} \min \big(N_{\text{ref}}[m], N_{\text{sys}}[m]\big) - N_{\text{corr}}[m]}{\sum\limits_{m=1}^{M} N_{\text{ref}}[m]} \tag{7}$$

The missing error E_{miss} counts the number of ground-truth pitches that could not be matched with any reported pitch. Here, none of the reported pitches has to be correctly identified.

$$E_{\text{miss}} = \frac{\sum\limits_{m=1}^{M} \max \big(0, N_{\text{ref}}[m] - N_{\text{sys}}[m]\big)}{\sum\limits_{m=1}^{M} N_{\text{ref}}[m]} \tag{8}$$

The false alarm error E_{fa} counts the number of reported pitches that could not be matched with any ground-truth pitch. Again, it is not important whether any of the reported pitches is correctly identified.

$$E_{\text{fa}} = \frac{\sum\limits_{m=1}^{M} \max \big(0, N_{\text{sys}}[m]\big) - N_{\text{ref}}[m]}{\sum\limits_{m=1}^{M} N_{\text{ref}}[m]} \tag{9}$$

If the number of output tends to be much larger than the ground-truth pitches, E_{fa} can be greater than 1.0. Then, the total error will be greater than 1.0 as well.

3.3 Results

Figure 2 shows the transcription results of the PT using single notes obtained from the analyzed piano as base components. Analyzing a low polyphony leads to many false alarm errors because of over-estimating the polyphony in many frames. Analyzing a higher polyphony leads to a strongly increase of E_{miss}. This means that PT tends to recognize too few notes. This might be caused by the distance limit θ.

Further base functions from the isolated notes of other pianos in the MAPS-database were obtained and evaluated the same dataset as before to testify the ability to generalize. The results are summarized in Figure 3. The whiskers of all plots mark the standard deviation of all errors. Compared to the base functions of the analyzed piano, the error scores increased, because PT was optimized with that specific piano. The standard deviation is small for all errors. This means that the *PianoTranscriber* is basically able to generalize to other pianos.

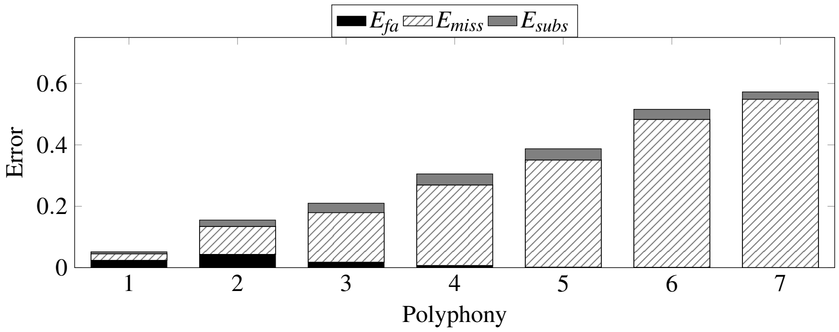

Figure 2 – Error scores for the PT using single notes obtained from the analyzed piano as base components. The total error E_{tot} is the height of each bar.

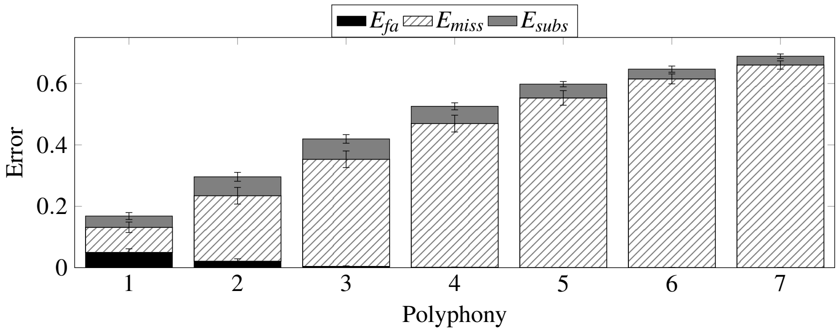

Figure 3 – Error scores for the PT using single notes obtained from other pianos as base components. The total error E_{tot} is the height of each bar. The whiskers show the standard deviation of all error scores.

Finally, in Figure 4 the reference algorithm SONIC is evaluated with standard settings. Comparing the result with PT using base functions from the analyzed piano, SONIC was outperformed up to a polyphony of 5. Afterwards, similar results were achieved. The highest error score is again E_{miss}, whereas E_{fa} had the lowest influence on the analysis result. Obviously, SONIC produces more errors caused by substitutions than the PT with arbitrary base functions. This is critical for a reliable transcription, because it will lead to incorrect notes.

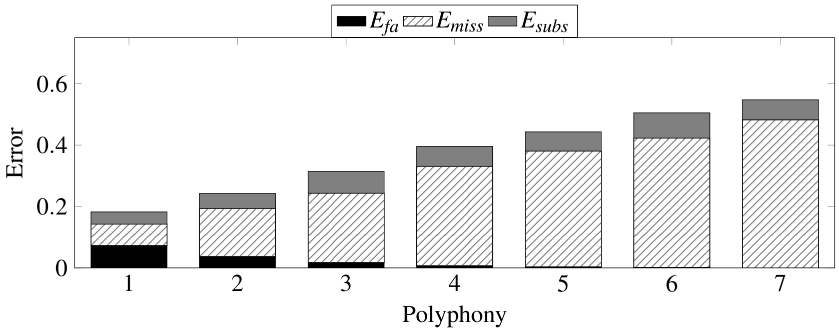

Figure 4 – Error scores for the reference algorithm SONIC. The total error E_{tot} is the height of each bar.

4 Conclusions and outlook

A new algorithm for Multi-Pitch-Tracking in music signals has been presented. Similar as the FOURIER-transform decomposes an analysis signal into its single frequency components, PT decomposed an input signal into notes and used base functions obtained from isolated notes on a standard piano. Using isolated notes from the analyzed piano lead to error scores that are comparable to the state-of-the-art. Further base functions from other pianos than the analyzed one were used to testify the generalization. This lead to an increased error rate, but the standard deviation is small, which is important for a generalization. In comparison to the state-of-the-art algorithm SONIC, PT outperformed SONIC up to a polyphony of 5 and achieved similar results for a larger polyphony using base functions from the analyzed piano. Using PT with base functions obtained from other pianos, SONIC outperformed PT. The *PianoTranscriber* is currently implemented as a MATLAB GUI and as a script for server applications.

References

[1] DEUTSCH, D., K. DOOLEY, T. HENTHORN, and B. HEAD: *Absolute pitch among students in an American music conservatory: Association with tone language fluency*. *The Journal of the Acoustical Society of America*, 125(4), pp. 2398–2403, 2009. doi:10.1121/1.3081389.

[2] KLAPURI, A. P.: *A perceptually motivated multiple-F0 estimation method*. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005*, pp. 291–294. 2005. doi:10.1109/ASPAA.2005.1540227.

[3] YEH, C., A. ROBEL, and X. RODET: *Multiple fundamental frequency estimation of polyphonic music signals*. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)., 2005.*, vol. 3, pp. iii/225–iii/228 Vol. 3. 2005. doi:10.1109/ICASSP.2005.1415687.

[4] SMARAGDIS, P. and J. C. BROWN: *Non-negative matrix factorization for polyphonic music transcription*. In *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (IEEE Cat. No.03TH8684)*, pp. 177–180. 2003. doi:10.1109/ASPAA.2003.1285860.

[5] MAROLT, M.: *A connectionist approach to automatic transcription of polyphonic piano music*. *IEEE Transactions on Multimedia*, 6(3), pp. 439–449, 2004. doi:10.1109/TMM.2004.827507.

[6] THICKSTUN, J., Z. HARCHAOUI, and S. M. KAKADE: *Learning features of music from scratch*. *ArXiv e-prints*, 2016. arXiv:1611.09827.

[7] EMIYA, V., 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*, 18(6), pp. 1643–1654, 2010. doi:10.1109/TASL.2009.2038819.

[8] BOERSMA, P.: *Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound*. In *Proceedings of the Institute of Phonetic Sciences*, vol. 17, pp. 97–110. Amsterdam, 1993.

[9] TALKIN, D.: *A robust algorithm for pitch tracking (rapt)*. *Speech coding and synthesis*, 495, pp. 497 – 518, 1995.

[10] DE CHEVEIGNÉ, A. and H. KAWAHARA: *Yin, a fundamental frequency estimator for speech and music*. *The Journal of the Acoustical Society of America*, 111(4), pp. 1917–1930, 2002. doi:10.1121/1.1458024.

[11] POLINER, G. E. and D. P. W. ELLIS: *A Discriminative Model for Polyphonic Piano Transcription*. *EURASIP Journal on Advances in Signal Processing*, 2007(1), p. 048317, 2006. doi:10.1155/2007/48317.