SCORE-INFORMED VOICE SEPARATION FOR PIANO RECORDINGS

Sebastian Ewert
Computer Science III, University of Bonn
ewerts@iai.uni-bonn.de

Meinard Müller
Saarland University and MPI Informatik
meinard@mpi-inf.mpg.de

ABSTRACT

The decomposition of a monaural audio recording into musically meaningful sound sources or voices constitutes a fundamental problem in music information retrieval. In this paper, we consider the task of separating a monaural piano recording into two sound sources (or voices) that correspond to the left hand and the right hand. Since in this scenario the two sources share many physical properties, sound separation approaches identifying sources based on their spectral envelope are hardly applicable. Instead, we propose a score-informed approach, where explicit note events specified by the score are used to parameterize the spectrogram of a given piano recording. This parameterization then allows for constructing two spectrograms considering only the notes of the left hand and the right hand, respectively. Finally, inversion of the two spectrograms yields the separation result. First experiments show that our approach, which involves high-resolution music synchronization and parametric modeling techniques, yields good results for real-world non-synthetic piano recordings.

1. INTRODUCTION

In recent years, techniques for the separation of musically meaningful sound sources from monaural music recordings have been applied to support many tasks in music information retrieval. For example, by extracting the singing voice, the bassline, or drum and instrument tracks, significant improvements have been reported for tasks such as instrument recognition [7], melody estimation [1], harmonic analysis [10], or instrument equalization [9]. For the separation, most approaches exploit specific spectral or temporal characteristics of the respective sound sources, for example the broadband energy distribution of percussive elements [10] or the spectral properties unique to the human vocal tract [1].

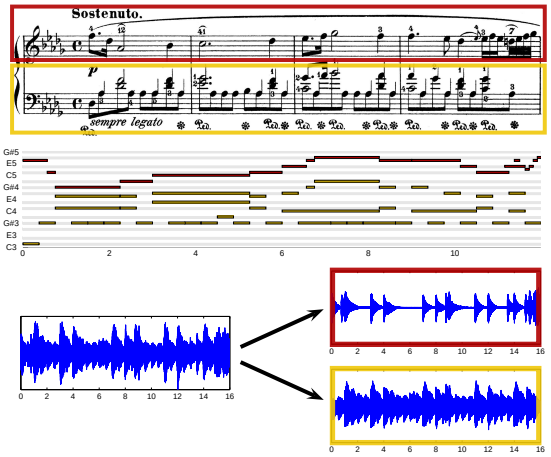

Figure 1. Decomposition of a piano recording into two sound sources corresponding to the left and right hand as specified by a musical score. Shown are the first four measures of Chopin Op. 28 No. 15.

In this paper, we present an automated approach for the decomposition of a monaural piano recording into sound sources corresponding to the left and the right hand as specified by a score, see Figure 1. Played on the same instrument and often being interleaved, the two sources share many spectral properties. As a consequence, techniques that rely on statistical differences between the sound sources are not directly applicable. To make the separation process feasible, we exploit the fact that a musical score is available for many pieces. We then use the explicitly given note events of the score to approximate the spectrogram of the given piano recording using a parametric model. Characterizing which part of the spectrogram belongs to a given note event, the model is then employed to decompose the spectrogram into parts related to the left hand and to the right hand. As an application, our goal is to extend the idea of an instrument equalizer as presented in [9] to a voice equalizer that can not only emphasize or attenuate whole instrument tracks but also individual voices or even single notes played by the same instrument. While we restrict the task in this paper to the left/right hand scenario, our approach is sufficiently general to isolate any kind of voice (or group of notes) that is specified by a given score.

So far, score-informed sound separation has received

only little attention in the literature. In [11], the authors replace the pitch estimation step of a sound separation system for stereo recordings with pitch information provided by an aligned MIDI file. In [6], a score-informed system for the elimination of the solo instrument from polyphonic audio recordings is presented. For the description of the spectral envelope of an instrument, the approach relies on pretrained information from a monophonic instrument database. In [4], score information is used as prior information in a separation system based on probabilistic latent component analysis (PLCA). This approach is in [8] compared to a score-informed approach based on parametric atoms. In [9], a score-informed system for the extraction of individual instrument tracks is proposed. To counterbalance their harmonic and inharmonic submodels, the authors have to incorporate complex regulation terms into their approach. Furthermore, the authors presuppose that, for each audio recording, a perfectly aligned MIDI file is available, which is not a realistic assumption.

In this paper, our main contribution is to extend the idea of an instrument equalizer to a voice equalizer that does not rely on statistical properties of the sound sources. As a further contribution, we do no presuppose the existence of prealigned MIDI files. Instead, we revert to high-resolution music synchronization techniques [3] to automatically align an audio recording to a corresponding musical score. Using the aligned score as an initialization, we follow the parametric model paradigm [2, 6, 7, 9] to obtain a note-wise parameterization of the spectrogram. As another contribution we show how separation masks that allow for a construction of voice-specific spectrograms can be derived from our model. Finally, applying a Griffin-Lim based inversion [5] to the separated spectrograms yields the final separation result.

The remainder of this paper is organized as follows. In Section 2, we introduce our parametric spectrogram model. Then, in Section 3, we describe how our model is employed to decompose a piano recording into two voices that correspond to the left hand and the right hand. In Section 4, we report on our systematic experiments using real-world as well as synthetic piano recordings. Conclusions and prospects on future work are given in Section 5. Further related work is discussed in the respective sections.

2. PARAMETRIC MODEL

To describe an audio recording of a piece of music using a parametric model, one has to consider many musical and acoustical aspects [7, 9]. For example, parameters are required to encode the pitch as well as the onset position and duration of note events. Further parameters might encode tuning aspects, the timbre of specific instruments, or amplitude progressions. In this section, we describe our model and show how its parameters can be estimated by an iterative method.

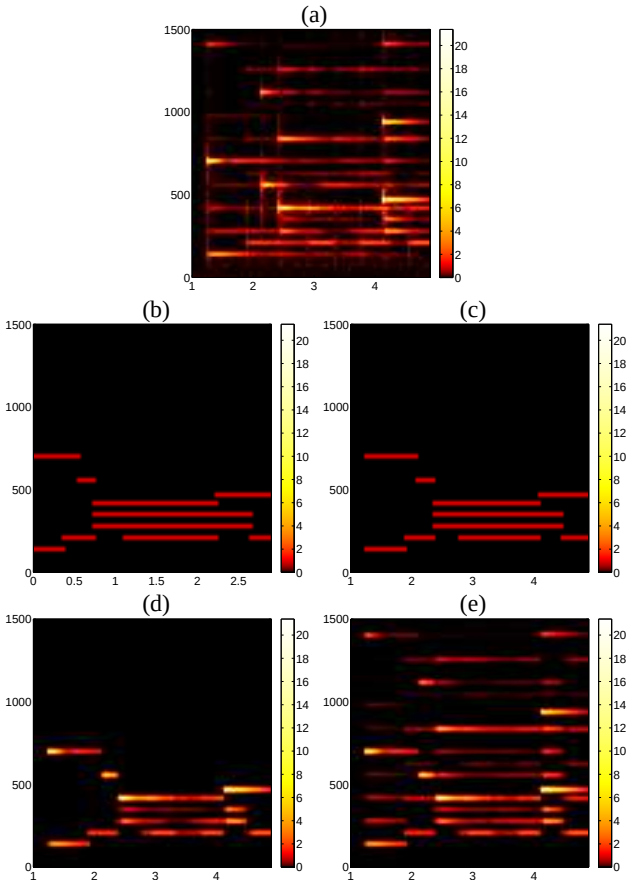

Figure 2. Illustration of the first iteration of our parameter estimation procedure continuing the example shown in Figure 1 (shown section corresponds to the first measure). **(a):** Audio spectrogram Y to be approximated. **(b)-(e)** Model spectrogram Y_λ after certain parameters are estimated. **(b):** Parameter $\mathcal{S}$ is initialized with MIDI note events. **(c):** Note events in $\mathcal{S}$ are synchronized with the audio recording. **(d):** Activity α and tuning parameter τ are estimated. **(e):** Partials' energy distribution parameter γ is estimated.

2.1 Parametric Spectrogram Model

Let $X \in \mathbb{C}^{K \times N}$ denote the spectrogram and $Y = |X|$ the magnitude spectrogram of a given music recording. Furthermore, let $\mathcal{S} := \{\mu_s \mid s \in [1:S]\}$ denote a set of note events as specified by a MIDI file representing a musical score. Here, each note event is modelled as a triple $\mu_s = (p_s, t_s, d_s)$, with p_s encoding the MIDI pitch, t_s the onset position and d_s the duration of the note event. Our strategy is to approximate Y by means of a model spectrogram $Y_\lambda^{\mathcal{S}}$, where λ denotes a set of free parameters representing acoustical properties of the note events. Based on the note event set $\mathcal{S}$, the model spectrogram $Y_\lambda^{\mathcal{S}}$ will be constructed as a superposition of note-event spectrograms Y_λ^s, $s \in [1:S]$. More precisely, we define $Y_\lambda^{\mathcal{S}}$ at frequency bin $k \in [1:K]$ and time frame $n \in [1:N]$ as

$$Y_\lambda^{\mathcal{S}}(k,n) := \sum_{\mu_s \in \mathcal{S}} Y_\lambda^s(k,n), \tag{1}$$

where each Y_λ^s denotes the part of $Y_\lambda^{\mathcal{S}}$ that is attributed to μ_s. Each Y_λ^s consists of a component describing the amplitude or activity over time and a component describing the spectral envelope of a note event. More precisely, we define

$$Y_\lambda^s(k,n) := \alpha_s(n) \cdot \varphi_{\tau,\gamma}(\omega_k, p_s), \tag{2}$$

where ω_k denotes the frequency in Hertz associated with the k-th frequency bin. Furthermore, $\alpha_s \in \mathbb{R}_{\geq 0}^N$ encodes the activity of the s-th note event. Here, we set $\alpha_s(n) := 0$, if the time position associated with frame n lies in $\mathbb{R} \setminus [t_s, t_s + d_s]$. The spectral envelope associated with a note event is described using a function $\varphi_{\tau,\gamma} : \mathbb{R} \times [1:P] \to \mathbb{R}_{\geq 0}$, where $[1:P]$ with $P = 127$ denotes the set of MIDI pitches. More precisely, to describe the frequency and energy distribution of the first L partials of a specific note event with MIDI pitch $p \in [1:P]$, the function $\varphi_{\tau,\gamma}$ depends on a parameter $\tau \in [-0.5, 0.5]^P$ related to the tuning and a parameter $\gamma \in [0,1]^{L \times P}$ related to the energy distribution over the L partials. We define for a frequency ω given in Hertz the envelope function

$$\varphi_{\tau,\gamma}(\omega, p) := \sum_{\ell \in [1:L]} \gamma_{\ell,p} \cdot \kappa(\omega - \ell \cdot f(p + \tau_p)), \tag{3}$$

where the function $\kappa : \mathbb{R} \to \mathbb{R}_{\geq 0}$ is a suitably chosen Gaussian centered at zero, which is used to describe the shape of a partial in frequency direction, see Figure 3. Furthermore, $f : \mathbb{R} \to \mathbb{R}_{\geq 0}$ defined by $f(p) := 2^{(p-69)/12} \cdot 440$ maps the pitch to the frequency scale. To account for non-standard tunings, we use the parameter τ_p to shift the fundamental frequency upwards or downwards by up to half a semitone. Finally, $\lambda := (\alpha, \tau, \gamma)$ denotes the set of free parameters with $\alpha := \{\alpha_s \mid s \in [1:S]\}$. The number of free parameters is kept low since the parameters τ and γ only depend on the pitch but not on the individual note events given by $\mathcal{S}$. Here, a low number allows for an efficient parameter estimation process as described below. Furthermore, sharing the parameters across the note events prevents model overfitting.

Now, finding a meaningful parameterization of Y can be formulated as the following optimization task:

$$\lambda^* = \operatorname*{argmin}_{\lambda} \|Y - Y_\lambda^{\mathcal{S}}\|_F, \tag{4}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. In the following, we illustrate the individual steps in our parameter estimation procedure in Figure 2, where a given audio spectrogram (Figure 2a) is approximated by our model (Figure 2b-2e).

2.2 Initialization and Adaption of Note Timing Parameters

To initialize our model, we exploit the available MIDI information represented by $\mathcal{S}$. For the s-th note event $\mu_s = (p_s, t_s, d_s)$, we set $\alpha_s(n) := 1$ if the time position associated with frame n lies in $[t_s, t_s + d_s]$ and $\alpha_s(n) := 0$ otherwise. Furthermore, we set $\tau_p := 0$, $\gamma_{1,p} := 1$ and $\gamma_{\ell,p} := 0$ for $p \in [1:P], \ell \in [2:L]$. An example model spectrogram $Y_\lambda^{\mathcal{S}}$ after the initialization is given in Figure 2b.

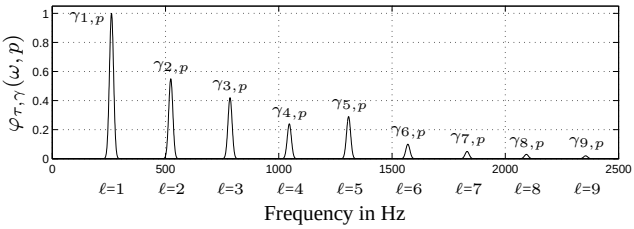

Figure 3. Illustration of the spectral envelope function $\varphi_{\tau,\gamma}(\omega, p)$ for $p = 60$ (middle C), $\tau = 0$ and some example values for parameters γ.

Next, we need to adapt and refine the model parameters to approximate the given audio spectrogram as accurately as possible. This parameter adaption is simplified when the MIDI file is assumed to be perfectly aligned to the audio recording as in [9]. However, in most practical scenarios such a MIDI file is not available. Therefore, in our approach, we employ a high resolution music synchronization approach as described in [3] to adapt the onset positions of the note events set $\mathcal{S}$. Based on Dynamic Time Warping (DTW) and chroma features, the approach also incorporates onset-based features to yield a high alignment accuracy. Using the resulting alignment, we determine for each note event the corresponding position in the audio recording and update the onset positions and durations in $\mathcal{S}$ accordingly. After the synchronization, the note event set $\mathcal{S}$ remains unchanged during all further parameter estimation steps. Figure 2c shows an example model spectrogram after the synchronization step.

2.3 Estimation of Model Parameters

To estimate the parameters in λ, we look for (α, τ, γ) that minimize the function $d(\alpha, \tau, \gamma) := \|Y - Y_{(\alpha,\tau,\gamma)}^{\mathcal{S}}\|_F$, thus minimizing the distance between the audio and the model spectrogram. Additionally, we need to consider range constraints for the parameters. For example, τ is required to be an element of $[-0.5, 0.5]^P$. To approximatively solve this constraint optimization problem, we employ a slightly modified version of approach exerted in [2]. In summary, this method works iteratively by fixing two parameters and by minimizing d with regard to the third one using a trust region based interior-points approach. For example, to get a better estimate for α, we fix τ and γ and minimize $d(\cdot, \tau, \gamma)$. This process is repeated until convergence similar to the well-known expectation-maximization algorithm. Figures 2d and 2e illustrate the first iteration of our parameter estimation. Here, Figure 2d shows the model spectrogram $Y_\lambda^{\mathcal{S}}$ after the estimation of the tuning parameter τ and the activity param-

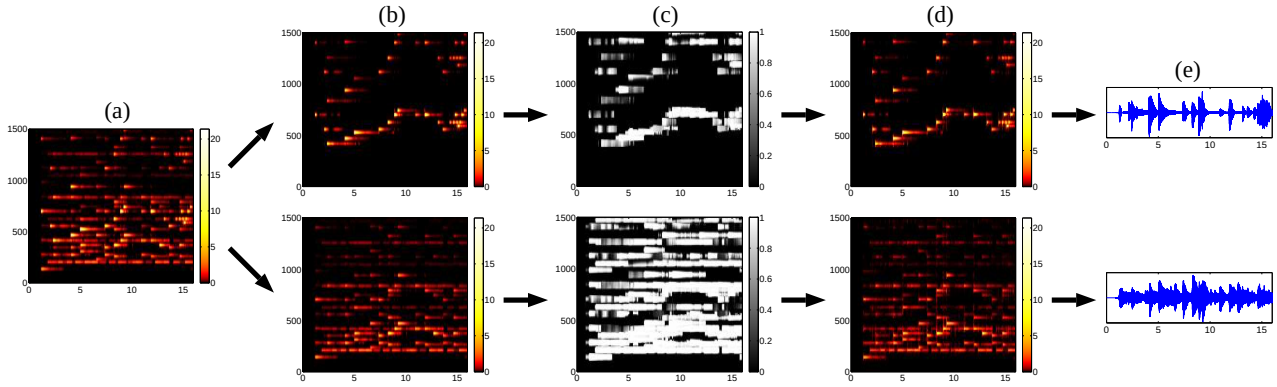

Figure 4. Illustration of our voice separation process continuing the example shown in Figure 1. **(a)** Model spectrogram $Y_\lambda^{\mathcal{S}}$ after the parameter estimation. **(b)** Derived model spectrograms $Y_\lambda^{\mathcal{L}}$ and $Y_\lambda^{\mathcal{R}}$ corresponding to the notes of the left and the right hand. **(c)** Separation masks $M^{\mathcal{L}}$ and $M^{\mathcal{R}}$. **(d)** Estimated magnitude spectrograms $\hat{Y}^{\mathcal{L}}$ and $\hat{Y}^{\mathcal{R}}$. **(e)** Reconstructed audio signals $\hat{x}^{\mathcal{L}}$ and $\hat{x}^{\mathcal{R}}$.

eter α. Figure 2e shows $Y_\lambda^{\mathcal{S}}$ after the estimation of the partials' energy distribution parameter γ.

3. VOICE SEPARATION

After the parameter estimation, $Y_\lambda^{\mathcal{S}}$ yields a note-wise parametric approximation of Y. In a next step, we employ information derived from the model to decompose the original audio spectrogram into separate channels or voices. To this end, we exploit that $Y_\lambda^{\mathcal{S}}$ is a compound of note-event spectrograms Y_λ^s. With $\mathcal{T} \subset \mathcal{S}$, we define $Y_\lambda^{\mathcal{T}}$ as

$$Y_\lambda^{\mathcal{T}}(k,n) := \sum_{\mu_s \in \mathcal{T}} Y_\lambda^s(k,n). \tag{5}$$

Then $Y_\lambda^{\mathcal{T}}$ approximates the part of Y that can be attributed to the note events in $\mathcal{T}$. One way to yield an audible separation result could be to apply a spectrogram inversion directly to $Y_\lambda^{\mathcal{T}}$. However, to yield an overall robust approximation result our model does not attempt to capture every possible spectral nuance in Y. Therefore, an audio recording deduced directly from $Y_\lambda^{\mathcal{T}}$ would miss these nuances and would consequently sound rather unnatural. Instead, we revert to the original spectrogram again and use $Y_\lambda^{\mathcal{T}}$ only to extract suitable parts of Y. To this end, we derive a *separation mask* $M^{\mathcal{T}} \in [0,1]^{K \times N}$ from the model which encodes how strongly each entry in Y should be attributed to $\mathcal{T}$. More precisely, we define

$$M^{\mathcal{T}} := \frac{Y_\lambda^{\mathcal{T}}}{Y_\lambda^{\mathcal{S}} + \varepsilon}, \tag{6}$$

where the division is understood entrywise. The small constant $\varepsilon > 0$ is used to avoid a potential division by zero. Furthermore, ε prevents that relatively small values in $Y_\lambda^{\mathcal{T}}$ lead to large masking values, which would not be justified by the model. For our experiments, we set $\varepsilon = 10^{-2}$.

For the separation, we apply $M^{\mathcal{T}}$ to a magnitude spectrogram via

$$\hat{Y}^{\mathcal{T}} := M^{\mathcal{T}} \circ Y, \tag{7}$$

where $\circ$ denotes entrywise multiplication (Hadamard product). The resulting $\hat{Y}^{\mathcal{T}}$ is referred to as *estimated magnitude spectrogram*. Here, using a mask for the separation allows for preserving most spectral nuances of the original audio. In a final step, we apply a spectrogram inversion to yield an audible separation result. Here, a commonly used approach is to combine $\hat{Y}^{\mathcal{T}}$ with the phase information of the original spectrogram X in a first step. Then, an inverse FFT in combination with an overlap-add technique is applied to the resulting spectrogram [7]. However, this usually leads to clicking and ringing artifacts in the resulting audio recording. Therefore, we apply a spectrogram inversion approach originally proposed by Griffin and Lim in [5]. The method attenuates the inversion artifacts by iteratively modifying the original phase information. The resulting $\hat{x}^{\mathcal{T}}$ constitutes our final separation result referred to as *reconstructed audio signal (relative to $\mathcal{T}$)*.

Next, we transfer these techniques to our left/right hand scenario. Each step of the full separation process is illustrated by Figure 4. Firstly, we assume that the score is partitioned into $\mathcal{S} = \mathcal{L} \dot{\cup} \mathcal{R}$, where $\mathcal{L}$ corresponds to the note events of the left hand and $\mathcal{R}$ to the note events of the right hand. Starting with the model spectrogram $Y_\lambda^{\mathcal{S}}$ (Figure 4a) we derive the model spectrograms $Y_\lambda^{\mathcal{L}}$ and $Y_\lambda^{\mathcal{R}}$ using Eqn. (5) (Figure 4b) and then the two masks $M^{\mathcal{L}}$ and $M^{\mathcal{R}}$ using Eqn. (6) (Figure 4c). Applying the two masks to the original audio spectrogram Y, we obtain the estimated magnitude spectrograms $\hat{Y}^{\mathcal{L}}$ and $\hat{Y}^{\mathcal{R}}$ (Figure 4d). Finally, applying the Griffin-Lim based spectrogram inversion yields the reconstructed audio signals $\hat{x}^{\mathcal{L}}$ and $\hat{x}^{\mathcal{R}}$ (Figure 4e).

4. EXPERIMENTS

In this section, we report on systematically conducted experiments to illustrate the potential of our method. To this end, we created a database consisting of seven representative pieces from the Western classical music repertoire, see Table 1. Using only freely available audio and score data al-

Composer	Piece	MIDI	Audio 1	Audio 2	Identifier
Bach	BWV875-01	MUT	Synthetic	SMD	'Bach875'
Beethoven	Op031No2-01	MUT	Synthetic	SMD	'Beet31No2'
Beethoven	Op111-01	MUT	Synthetic	EA	'BeetOp111'
Chopin	Op028-01	MUT	Synthetic	SMD	'Chop28-01'
Chopin	Op028-04	MUT	Synthetic	SMD	'Chop28-04'
Chopin	Op028-15	MUT	Synthetic	SMD	'Chop28-15'
Chopin	Op064No1	MUT	Synthetic	EA	'Chop64No1'
Chopin	Op066	MUT	Synthetic	SMD	'Chop66'

Table 1. Pieces and audio recordings (with identifier) used in our experiments.

lows for a straightforward replication of our experiments. Here, we used uninterpreted score-like MIDI files from the Mutopia Project [1] (MUT), high-quality audio recordings from the Saarland Music Database [2] (SMD) as well as digitized versions of historical gramophone and vinyl recordings from the European Archive [3] (EA).

In a first step, we indicate the quality of our approach quantitatively using synthetic audio data. To this end, we used the Mutopia MIDI files to create two additional MIDI files for each piece using only the notes of the left and the right hand, respectively. Using a wave table synthesizer, we then generated audio recordings from these MIDI files which are used as ground truth separation results in the following experiment. We denote the corresponding magnitude spectrograms by $Y^{\mathcal{L}}$ and $Y^{\mathcal{R}}$, respectively. For our evaluation we use a quality measure based on the signal-to-noise ratio (SNR) [4]. More precisely, to compare a reference magnitude spectrogram $Y_R \in \mathbb{R}_{\geq 0}^{K \times N}$ to an approximation $Y_A \in \mathbb{R}_{\geq 0}^{K \times N}$ we define

$$\mathrm{SNR}(Y_R, Y_A) := 10 \cdot \log_{10} \frac{\sum_{k,n} Y_R(k,n)^2}{\sum_{k,n} (Y_R(k,n) - Y_A(k,n))^2}.$$

The second and third column of Table 2 show SNR values for all pieces, where the ground truth is compared to the estimated spectrogram for the left and the right hand. For example, the left hand SNR for 'Chop28-15' is 17.79 whereas the right hand SNR is 13.35. The reason the SNR being higher for the left hand than for the right hand is that the left hand is already dominating the mixture in terms of overall loudness. Therefore, the left hand segregation is per se easier compared the the right hand segregation. To indicate which hand is dominating in a recording, we additionally give SNR values comparing the ground truth magnitude spectrograms $Y^{\mathcal{L}}$ and $Y^{\mathcal{R}}$ to the mixture magnitude spectrogram Y, see column six and seven of Table 2. For example for 'Chop28-15', $\mathrm{SNR}(Y^{\mathcal{L}}, Y) = 3.48$ is much higher compared to $\mathrm{SNR}(Y^{\mathcal{R}}, Y) = -2.47$ thus revealing the left hand dominance.

[1] http://www.mutopiaproject.org

[2] http://www.mpi-inf.mpg.de/resources/SMD/

[3] http://www.europarchive.org

[4] Even though SNR values are often not perceptually meaningful, they at least give some tendencies on the quality of separation results.

Identifier	SNR $(Y^{\mathcal{L}},\hat{Y}^{\mathcal{L}})$	SNR $(Y^{\mathcal{R}},\hat{Y}^{\mathcal{R}})$	SNR $(Y^{\mathcal{L}},\hat{Y}^{\mathcal{L}})$	SNR $(Y^{\mathcal{R}},\hat{Y}^{\mathcal{R}})$	SNR $(Y^{\mathcal{L}},Y)$	SNR $(Y^{\mathcal{R}},Y)$
	prealigned		distorted			
Bach875	11.24	12.97	11.17	12.89	-1.99	3.03
Beet31No2	12.65	10.38	12.47	10.23	1.24	-0.09
BeetOp111	13.21	12.26	12.92	11.99	0.16	0.97
Chop28-01	10.52	13.96	10.43	13.84	-3.38	4.48
Chop28-04	17.63	10.48	17.58	10.45	8.65	-7.55
Chop28-15	17.79	13.35	17.56	13.18	3.48	-2.47
Chop64No1	12.93	11.86	12.60	11.55	-0.06	1.31
Chop66	11.61	11.17	11.46	11.03	-0.41	2.01
Average	**13.45**	**12.05**	**13.27**	**11.90**	**0.96**	**0.21**

Table 2. Experimental results using ground truth data consisting of synthesized versions of the pieces in our database.

Using synthetic data, the audio recordings are already perfectly aligned to the MIDI files. To further evaluate the influence of the music synchronization step, we randomly distorted the MIDI files by splitting them into 20 segments of equal length and by stretching or compressing each segment by a random factor within an allowed distortion range (in our experiments we used a range of $\pm 50\%$). The results for these distorted MIDI files are given in column four and five of Table 2. Here, the left hand SNR for 'Chop28-15' decreases only moderately from 17.79 (prealigned MIDI) to 17.56 (distorted MIDI), and from 13.35 to 13.18 for the right hand. Similarly, the average SNR also decreases moderately from 13.45 to 13.27 for the left hand and from 12.05 to 11.90 for the right hand, which indicates that our synchronization works robustly in these cases. The situation in real world scenarios becomes more difficult, since here the note events of the given MIDI may not correspond one-to-one to the played note events of a specific recording. An example will be discussed in the next paragraph, see also Figure 5.

As mentioned before, signal-to-noise ratios and similar measures cannot capture the perceptual separation quality. Therefore, to give a realistic and perceptually meaningful impression of the separation quality, we additionally provide a website [5] with audible separation results as well as visualizations illustrating the intermediate steps in our procedure. Here, we only used real, non-synthetic audio recordings from the SMD and EA databases to illustrate the performance of our approach in real world scenarios. Listening to these examples does not only allow to quickly get an intuition of the method's properties but also to efficiently locate and analyze local artifacts and separation errors. For example, Figure 5 illustrates the separation process for 'BeetOp111' using an interpretation by Egon Petri (European Archive). As a historical recording, the spectrogram of this recording (Figure 5c) is rather noisy and reveals some artifacts typical for vinyl recordings such as rumbling and cranking glitches. Despite these artifacts, our model approximates the audio spectrogram well (w.r.t. to the euclidean norm) in most areas (Figure 5d). Also the resulting

[5] http://www.mpi-inf.mpg.de/resources/MIR/2011-ISMIR-VoiceSeparation/

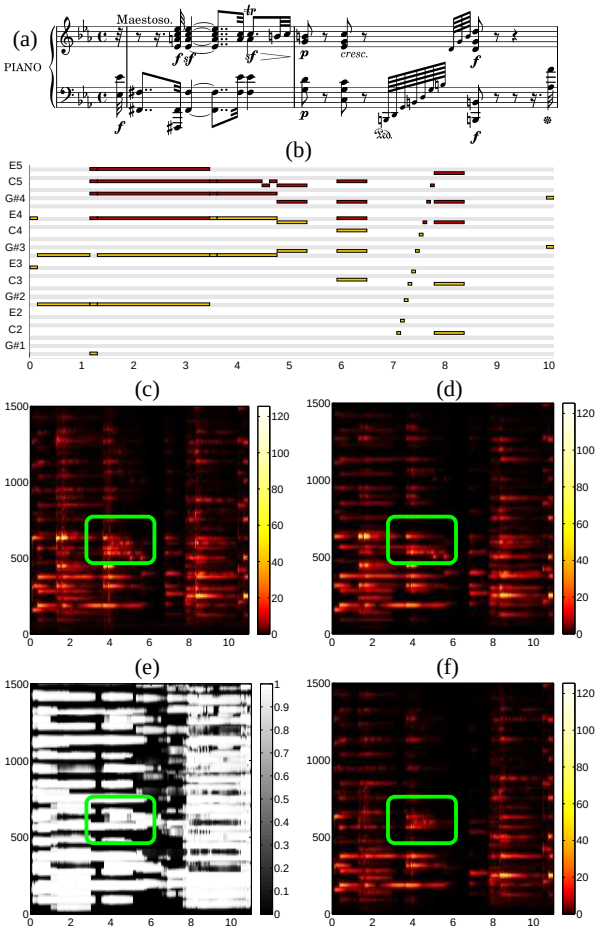

Figure 5. Illustration of the separation process for 'BeetOp111'. **(a):** Score corresponding to the first two measures. **(b):** MIDI representation (Mutopia Project). **(c):** Spectrogram of an interpretation by Petri (European Archive). **(d):** Model spectrogram after parameter estimation. **(e):** Separation mask $M^{\mathcal{L}}$. **(f):** Estimated magnitude spectrogram $\hat{Y}^{\mathcal{L}}$. The area corresponding to the fundamental frequency of the trills in measure one is indicated using a green rectangle.

separation results are plausible, with one local exception. Listening to the separation results reveals that the trills towards the end of the first measure were assigned to the left instead of the right hand. Investigating the underlying reasons shows that the trills are not correctly reflected by the given MIDI file (Figure 5b). As a consequence, our score-informed approach cannot model this spectrogram area correctly as can be observed in the marked areas in Figures 5c and 5d. Applying the resulting separation mask (Figure 5e) to the original spectrogram leads to the trills being misassigned to the left hand in the estimated magnitude spectrogram as shown in Figure 5f.

5. CONCLUSIONS

In this paper, we presented a novel method for the decomposition of a monaural audio recording into musically meaningful voices. Here, our goal was to extend the idea of an instrument equalizer to a voice equalizer which does not rely on statistical properties of the sound sources and which is able to emphasize or attenuate even single notes played by the same instrument. Instead of relying on prealigned MIDI files, our score-informed approach directly addresses alignment issues using high-resolution music synchronization techniques thus allowing for an adoption in real world scenarios. Initial experiments showed good results using synthetic as well as real audio recordings. In the future, we plan to extend our approach with an onset model while avoiding the drawbacks discussed in [9].

Acknowledgement. This work has been supported by the German Research Foundation (DFG CL 64/6-1) and the Cluster of Excellence on Multimodal Computing and Interaction at Saarland University.

6. REFERENCES

[1] J.-L. Durrieu, G. Richard, B. David, and C. Févotte. Source/filter model for unsupervised main melody extraction from polyphonic audio signals. *IEEE Transactions on Audio, Speech and Language Processing*, 18(3):564–575, 2010.

[2] S. Ewert and M. Müller. Estimating note intensities in music recordings. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 385–388, Prague, Czech Republic, 2011.

[3] S. Ewert, M. Müller, and P. Grosche. High resolution audio synchronization using chroma onset features. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1869–1872, Taipei, Taiwan, 2009.

[4] J. Ganseman, P. Scheunders, G. J. Mysore, and J. S. Abel. Source separation by score synthesis. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 462–465, New York, USA, 2010.

[5] D. W. Griffin and J. S. Lim. Signal estimation from modified short-time Fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(2):236–243, 1984.

[6] Y. Han and C. Raphael. Desoloing monaural audio using mixture models. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 145–148, Vienna, Austria, 2007.

[7] T. Heittola, A. Klapuri, and T. Virtanen. Musical instrument recognition in polyphonic audio using source-filter model for sound separation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 327–332, Kobe, Japan, 2009.

[8] R. Hennequin, B. David, and R. Badeau. Score informed audio source separation using a parametric model of non-negative spectrogram. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 45–48, Prague, Czech Republic, 2011.

[9] K. Itoyama, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno. Instrument equalizer for query-by-example retrieval: Improving sound source separation based on integrated harmonic and inharmonic models. In *Proceedings of the International Conference for Music Information Retrieval (ISMIR)*, pages 133–138, Philadelphia, USA, 2008.

[10] Y. Ueda, Y. Uchiyama, T. Nishimoto, N. Ono, and S. Sagayama. HMM-based approach for automatic chord detection using refined acoustic features. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5518–5521, Dallas, USA, 2010.

[11] J. Woodruff, B. Pardo, and R. B. Dannenberg. Remixing stereo music with score-informed source separation. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 314–319, 2006.