Instrument-Centered Music Transcription of Solo Bass Guitar Recordings

Jakob Abeßer and Gerald Schuller, *Senior Member, IEEE*

***Abstract*—This paper deals with the automatic transcription of solo bass guitar recordings with an additional estimation of playing techniques and fretboard positions used by the musician. Our goal is to first develop a system for a robust estimation of the note parameters pitch, onset, and duration (score-level parameters). As a second step, we aim to automatically detect the applied plucking and expression style as well as the fret and string positions for each note (instrument-level parameters). Our approach is to first apply a note onset detection followed by a tracking of the fundamental frequency contours based on a reassigned magnitude spectrogram. Then, we model the spectral envelope of each note and derive various timbre-related audio features. Using a support vector machine classifier, we automatically classify the instrument-level parameters for each detected note event. Our results show that the proposed system achieves accuracy values above 0.88 for the estimation of the plucking style, expression style, and string number for isolated note samples. As an additional contribution, we analyze the influence of the note duration characteristics in the classification performance. In a score-level evaluation on a novel public dataset of solo bass guitar tracks, our method outperforms three existing transcription algorithms for bass transcription in polyphonic music as well as a melody transcription algorithm for monophonic music.**

***Index Terms*—Bass transcription, electric bass guitar, fretboard position, playing technique estimation, string detection.**

I. Introduction

In THE field of Music Information Retrieval (MIR), various algorithms have been proposed for automatic music transcription tasks such as predominant melody estimation, drum transcription, multipitch estimation, or bass transcription. The transcription process involves the detection of all played notes of one or multiple musical instruments as sound events within a given audio signal. Each note event is described by *score-level parameters*, such as *onset* time, *duration*, and *pitch*.

It has been frequently observed that the main challenge for music transcription algorithms is the inherent interference of multiple instrument signals in the audio mix [1]. In this paper, we focus on the transcription of solo bass guitar tracks, which is a simplification of the general task of transcribing multi-instrumental music recordings. Solo instrument tracks can be extracted from multitrack studio recordings or by applying source separation algorithms to multi-instrumental music recordings. The ultimate goal is to integrate the proposed transcription algorithm into music learning applications, where the user's instrument signal can be recorded without any overlap with other instrument sources. We simplify the general transcription task since we want to extend the set of score-level parameters by additional *instrument-level parameters*, which allow to describe the recorded music performance in greater detail.

TABLE I
Taxonomy of Bass Guitar Playing Techniques

Plucking Style	Abbr.	Expression Style	Abbr.
Finger-Style	FS	Normal	NO
Picked	PK	Harmonics	HA
Muted	MU	Dead-Notes	DN
Slap-Thumb	ST	Vibrato	VI
Slap-Pluck	SP	Bending	BE
		Slide	SL

Five Plucking Styles and Six Expression Styles are Distinguished.

Various playing techniques exist for musical instruments like the bass guitar, which enable the musician to create a large variety of different sounds. Also, playing notes with the same pitch at different positions on the instrument neck changes the sound characteristics due to different string gauges and pickup positions relative to the string length. Here, we consider the instrument-level parameters *plucking style*, i. e., the playing technique used to excite the string vibration with the plucking hand, *expression style*, i. e., the playing technique applied to the vibrating string after the plucking, as well as the fretboard position, which is defined by the *string number* and the *fret number*. All note parameters are estimated solely from the acoustic recording without any additional cues from visual or sensory measurements such as discussed in [2].

Table I summarizes the 5 plucking styles and 6 expression styles that we consider in this paper. In short, the plucking of the string can be performed using the index and middle fingers (*finger-style*), a plastic pick (*picked*), or muted with the thumb (*muted*). Strong excitations of the string using a hammer-like movement (*slap-thumb*) or a strong plucking (*slap-pluck*) cause it to collide with the metal frets. The string can be damped fully (*dead-note*) or at integer fractions of the string length to invoke *harmonics*. Also, the note pitch can be modulated by *sliding* to a

Manuscript received February 1, 2017; revised April 18, 2017; accepted April 28, 2017. Date of publication May 8, 2017; date of current version July 12, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Matthew E. P. Davies. *(Corresponding author: Jakob Abeßer.)*

J. Abeßer is with Fraunhofer IDMT, Ilmenau 98693, Germany (e-mail: jakob.abesser@idmt.fraunhofer.de).

G. Schuller is with the Institute for Medientechnik, Tu Ilmenau, Ilmenau 98693, Germany (e-mail: shl@idmt.fraunhofer.de).

Digital Object Identifier 10.1109/TASLP.2017.2702384

target tone or by bending and releasing the string once (*bending*) or periodically (*vibrato*).

Our main contribution is an *instrument-centered transcription algorithm*, which is tailored to solo bass guitar recordings. This paper extends our previous work [3], [4] by a simplified envelope modeling procedure (see Section III-E) as well as additional evaluation experiments towards bass transcription and instrument-level parameter estimation. The remainder of this paper is structured as follows. We will first review related publications in Section II and then describe the proposed transcription algorithm in Section III. Afterwards, we will outline the experiments, which were performed to seperately evaluate the estimation of score-level and instrument-level parameters (Section IV). Finally, Section V will conclude our work.

II. Related Work

A. Score Parameter Estimation

Most music transcription algorithm are designed "from an instrument-free perspective" [5] and extract a score or piano-roll representation from the audio data without taking instrument-specific parameters into account. Existing bass transcription algorithms commonly assume that the bass line is the lowest monophonic melody line in an audio recording [6]. Due to the low fundamental frequency range of bass notes, *down-sampling* is often applied to increase the computational efficiency of the transcription algorithms and to filter out harmonic signal components from instruments playing in higher pitch registers.

Most existing bass transcription algorithms are designed for transcribing polyphonic music recordings and therefore include *source separation* strategies such as melody line removal [7] or harmonic-percussive separation [8] to isolate the bass line first. In the presented work, we focus on solo bass guitar recordings and do not require an initial source separation step. The Short-time Fourier Transformation (STFT) is most often used for *spectral estimation* in bass transcription algorithms [6], [7], [9], [10] due to its computational efficiency. However, the spectral leakage effect limits the achievable frequency resolution especially in lower frequency bands. As a consequence, the use of other spectral estimation techniques, as the instantaneous frequency (IF) spectrogram [9], [11], and the constant-Q transform [12] was proposed.

For the *spectral decomposition*, different authors propose to extract a *pitch saliency function*, which measures the likelihood of a given fundamental frequency value at a given time. Klapuri and Ryynänen compute the harmonic salience of a f_0 candidate by summing up the spectral energy at the frequency bins of the corresponding harmonic frequencies [10]. Salamon and Goméz extract a saliency function from the mid-level chromagram-based Harmonic Pitch Class Profile (HPCP) [13] but do not perform an additional note grouping step. For the *note-grouping* step, Ryynänen and Klapuri combine an acoustic model and a musicological model in [10]. In the *PreFEst* (predominant-F0 estimation) algorithm proposed by Goto in [11], the most salient peaks of a pitch saliency function are tracked over time and grouped to note events.

Apart from a lower pitch range of interest, the transcription task tackled in this paper is similar to monophonic melody transcription. Therefore, we compare the performance of the proposed algorithm with the state-of-the-art melody transcription algorithm pYIN proposed by Mauch and Dixon [14] (see Section IV-B1).

B. Instrument Parameter Estimation

Several instrument-centered transcription algorithms have been published, which are tailored towards violin [15], [16], cello [17], electric guitar [18], [19], and piano [20] recordings. These algorithms incorporate additional instrument-specific constraints such as the overall pitch range or the maximum number of strings. Concerning the estimation of *playing techniques* and *fretboard positions* from audio recordings, the main focus has been on the analysis of string instruments like the guitar and violin, which have similar sound production mechanisms as the bass guitar. Most authors analyze audio signals while video recordings from attached cameras or sensory data from capacitive sensors or movement sensors is used in [2], [21], [22]. For audio analysis, microphones, electro-magnetic, or piezo pickup systems are commonly used to convert the string vibration to an electric signal.

Various algorithms were proposed to automatically detect guitar playing techniques like *damping* [23], *vibrato* [19], [23]–[25], *bending* [19], [26], *slides* (also denoted as glissando or appogiatura) [19], [25]–[27], as well as note-transition techniques like *hammer-on* and *pull-off* [24]–[27]. Similarly, violin techniques like *bowing* are analyzed using either movement sensors [21], [22] or audio analysis [2], [28]. Barbancho *et al.* classify between seven violin playing techniques in [29].

Algorithms for estimating the *fretboard position* from monophonic guitar recordings were proposed in [30]–[34]. For instance, Barbancho *et al.* [31] compute various timbre-related spectral audio features such as the inharmonicity, relative magnitude of the harmonics, or the temporal decay factor of harmonics to classify the *string number*. Humphrey and Bello recently propose a system for guitar chord recognition and tablature creation using deep convolutional networks [35]. The reader is referred to [4] for an extensive literature review.

III. Proposed Bass Guitar Transcription Algorithm

The proposed bass guitar transcription algorithm uses two main constraints. The bass line is assumed to be *perfectly isolated* without any interference from other musical instruments such as drums or guitars as well as *monophonic*, i. e., all notes must be played one after the other without any temporal overlap. The processing steps of the algorithm are summarized in Fig. 1. Two main estimation tasks are performed. Firstly, the *score-level parameters* onset $\mathcal{O}$, duration $\mathcal{D}$, pitch $\mathcal{P}$, and intensity $\mathcal{L}$ are extracted for each detected note event. Based on a machine learning approach, the *instrument-level parameters* plucking style $\mathcal{S}_P$, expression style $\mathcal{S}_E$, string number $\mathcal{N}_S$, as well as fret number $\mathcal{N}_F$ are automatically classified. In the following sections, all processing steps of the transcription algorithm are detailed.

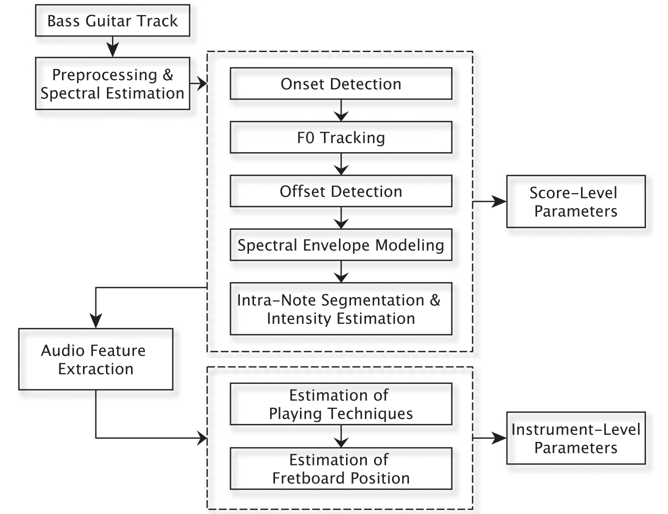
Bass Guitar Track
Preprocessing & Spectral Estimation
Onset Detection
F0 Tracking
Offset Detection
Spectral Envelope Modeling
Intra-Note Segmentation & Intensity Estimation
Score-Level Parameters
Audio Feature Extraction
Estimation of Playing Techniques
Estimation of Fretboard Position
Instrument-Level Parameters

Fig. 1. Processing flowchart of the proposed bass guitar transcription algorithm.

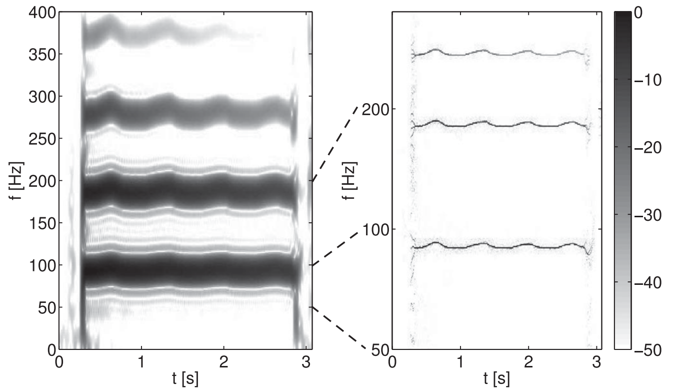

Fig. 2. STFT magnitude spectrogram (linearly-spaced frequency axis) and IF spectrogram (logarithmically-spaced frequency axis) with dB magnitude scale for a bass guitar note played with the expression style *vibrato* [4]. Correpondencies between the frequency axes are shown for 50 Hz, 100 Hz, and 200 Hz.

A. Pre-Processing & Spectral Estimation

Since the focus of the proposed bass transcription algorithm is on lower frequency ranges, we perform an initial downsampling of the signal to $f_s = 5512.5\,Hz$. Then we compute the magnitude spectrogram $M \in \mathbb{R}^{K \times N}$ based on a *Short-time Fourier Transform* (STFT) using a Hanning window, a window length (blocksize) of 512 samples, an FFT length of 4096 (using zero-padding), and a hopsize of 32 samples (K denotes the number of frequency bins and N denotes the number of frames). The resulting linearly-spaced frequency axis is defined as $f(k) = \frac{k}{N_{\text{FFT}}} f_s$ with $k \in [0, N_{\text{FFT}}/2]$. This spectrogram representation is used for the spectral envelope modeling step discussed in Section III-E.

Second, we compute a *reassigned magnitude spectrogram* [36] $M_{\text{IF}} \in \mathbb{R}^{K_{\log} \times N}$ based on the *instantaneous frequency* (IF) values $\hat{f}(k,n) \in \mathbb{R}^{K \times N}$ of the STFT spectrogram. Here, we use a logarithmically-spaced frequency axis defined as $f_{\log}(k_{\log}) = 440 \times 2^{\frac{22+k_{\log}/10-69}{12}}$ with the indices $k_{\log} \in [0, 780]$ that covers a frequency range of $f_{\log}(k_{\log}) \in [29.1, f_s/2]$.[1] We use a relatively high frequency resolution of 10 bins per semitone to capture small fundamental frequency (f_0) modulations, which are characteristic for the expression styles *bending*, *vibrato*, and *slide*. In the proposed bass guitar transcription algorithm, the IF spectrogram is used for onset detection and f_0 tracking as will be discussed in Section III-B and Section III-C.

The instantaneous frequency $\hat{f}(k,n)$ for each time-frequency bin of the STFT spectrogram is estimated from the time-derivative of the local phase in the STFT spectrogram using the method proposed by Lagrange and Marchand in [37] (Equation 17). The reassigned magnitude spectrogram M_{IF} is computed as follows: For each time-frequency bin (k,n) in the STFT spectrogram, the magnitude value $M(k,n)$ is mapped to the corresponding time-frequency bin $(k_{\log}, n)$ in the reassigned spectrogram M_{IF} such that the frequency value $f_{\log}(k_{\log})$ is closest to the instantaneous frequency value $\hat{f}(k,n)$. For each time-frequency bin $(k_{\log}, n)$ in M_{IF}, we accumulate all magnitude values from $M(k,n)$, which are mapped to that bin. Harmonic signals such as bass guitar notes show sinosoidal peaks at the fundamental frequency f_0 and the overtone frequencies. In the frequency bins around each peak, one can observe stable instantaneous frequency values. As a consequence, the aforementionend mapping procedure results in sharp magnitude peaks in M_{IF} which are well-suited for a temporal tracking.

Fig. 2 shows a bass guitar note played with the expression style *vibrato* both as STFT magnitude spectrogram (left figure) and as IF spectrogram (right figure). It can be observed in the IF spectrogram that the sharper peaks are better suited for tracking of harmonic frequency components over time.

B. Note Onset Detection

For detecting note onsets, we propose a novel *onset detection function* to measure the degree of *harmonic novelty*. We compute the 2-D discrete convolution

$$M_{\text{IF, K}} = M_K *_{2d} M_{\text{IF}} \tag{1}$$

with $M_{\text{IF,K}} \in \mathbb{R}^{K_{\log} \times N}$ between the IF spectrogram M_{IF} and the kernel matrix

$$M_K = \left[0.3, 1, 0.3\right]^T \times \left[1, 1, 1, 0, -1, -1, -1\right] \tag{2}$$

and store the convolution result as $M_{\text{IF,K}} \in \mathbb{R}^{K_{\log} \times N}$. The kernel matrix is a time-reversed (matched) filter with two properties—smoothing along the frequency axis (sinosoidal peaks of harmonic components) and detection of edges along the time axis (note onset transients).[2] We derive the harmonic novelty onset detection function $\alpha_{\text{On}} \in \mathbb{R}^N$ by computing the frame-wise maxima

$$\alpha_{\text{On}}(n) = \max_k M_{\text{IF,K}}(k,n). \tag{3}$$

[1] Here, 69 is the MIDI pitch corresponding to the frequency 440 Hz and 22 is the MIDI pitch of the lowest fundamental frequency considered here. The normalization factor 10 corresponds to the resolution of 10 bins per semitone.

[2] This approach is similar to image edge detection.

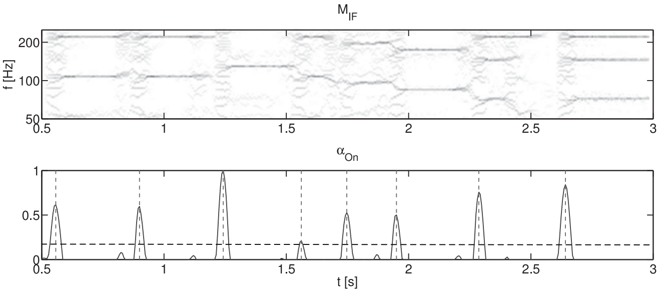

Fig. 3. The upper figure shows the IF spectrogram $M_{\text{IF}}(f,t)$ of an excerpt from a solo bass recording. The beginning of stable harmonic components (partials) indicate note events. The lower plot shows the normalized harmonic novelty onset detection function $\alpha_{\text{On}}(t)$ (solid line), the threshold (horizontal dashed line), as well as the detected note onset times (vertical dashed lines) [4].

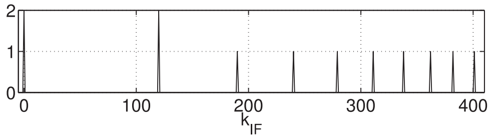

Fig. 4. Harmonic spectral template $T(k_{\log})$ based on a logarithmically-spaced frequency axis and with doubled magnitude on the first two harmonics [4].

Then, we derive note onset positions $\mathcal{O} \in \mathbb{R}^{N_{\text{Notes}}}$ from local maximum positions with a value greater than one fifth of the global maximum of α_{On}. Using the threshold value of $\tau_{\text{On}} = 0.2$, we achieved the highest F-measure in onset detection ($F = 0.95$) on a separate development data set. During initial experiments on our development data set, this approach outperformed a dynamic thresholding approach [38] based on the spectral flux onset detection function ($F = 0.91$).

As an example, Fig. 3 illustrates the IF spectrogram $M_{\text{IF}}(f,t)$ of an excerpt of a bass signal (upper plot) and the corresponding onset detection function $\alpha_{\text{On}}(t)$ (lower plot). The detected onset positions are indicated as dashed lines. It can be observed that the IF spectrogram does not capture wide-band transients well, which typically occurr during the note attack part of plucked string instruments. Nevertheless, the shape of the aforementioned kernel function leads to a robust detection of note onsets.

C. Fundamental Frequency Tracking

After detecting a note's onset time, we perform the following steps to track its fundamental frequency contour $f_0(n)$. First, we average the magnitude spectrogram M_{IF} over the first 20% of the note's distance to the following onset time (inter-onset-interval) and compute an accumulated magnitude spectrogram $M_{\text{IF,acc}} \in \mathbb{R}^{K_{\log}}$. In order to get a robust estimate of the note's fundamental frequency f_0, we compute the cross-correlation between $M_{\text{IF,acc}}$ and a *harmonic spectral template* $T \in \mathbb{R}^{K_{\log}}$, which is illustrated in Fig. 4. The template T resembles an idealized harmonic magnitude spectrogram of a tone with peaks at the harmonic frequencies

$$f_h = (h+1) f_0 \sqrt{1 + \beta (h+1)^2} \text{ with } h \in [0, 9]. \tag{4}$$

with β denoting the *inharmonicity coefficient*[3] [39].

We estimate β using a grid search along 100 equidistant grid points within $[0, 0.001]$. For each candidate $\hat{\beta}$, we compute the corresponding harmonic frequencies $\hat{f}_{h,\hat{\beta}}$ using Equation 4 and estimate the harmonic magnitudes $\hat{a}_{h,\hat{\beta}}$ using linear interpolation from $M_{\text{IF,acc}}$. We obtain β by maximizing the function $L(\hat{\beta}) = \sum_{h=0}^{9} \hat{a}_{h,\hat{\beta}}$. We found experimentally that by doubling the magnitude of the first two peaks of the template T, the pitch detection accuracy improved from $A = 0.96$ for unit peak magnitudes up to $A = 0.98$ [3]. Based on the estimated fundamental frequency value f_0, the note pitch of the i-th note is computed as $\mathcal{P}(i) = \lfloor 12 \log_2 \left(\frac{f_0}{440} \right) + 69 + 0.5 \rfloor$ using a quantization to pitch frequencies of equal temperament tuning.

In the second step, the fundamental frequency contour $f_0(n)$ is tracked over time. Starting at a start frame n_0 at 10 % of the inter-onset-interval, we perform a temporal tracking forwards and backwards in time. We use a *continuity-constraint* such that in each time frame n, only those frequency indices k_{IF} adjancent to the estimated fundamental frequency index of the preceding frame are considered as possible f_0 candidates. Among these candidates, we select the frequency index with the highest cross-correlation between the shifted harmonic spectral template and the current spectral frame in M_{IF}. We store the maximum frame-wise cross-correlation values as $C_{\max}(n)$ for the offset detection procedure explained in the next section.

D. Note Offset Detection

Plucked string instrument notes are commonly segmented into an *attack part*, which is characterized by a rapidly increasing magnitude envelope and a *decay part*, which is characterized by approximately exponentially decaying magnitude values. As discussed in Section III-B, note onset times coincide with attack transients, which are time-localized wideband events in the audio signal. In contrast, the decay of the note envelope of plucked string instruments is continuous with no clearly defined note ending except if the string is damped. We estimate the *note offset frame* $n_{\text{Off}}(i)$ of the i-th note as the first frame after the note onset frame $n_{\text{On}}(i)$, where the cross-correlation value $C_{\max}(n)$ (see Section III-C) remains below 5 % of the maximum cross-correlation value for at least four adjacent frames or a new note begins.

E. Spectral Envelope Modeling

In [3], we proposed a parametric spectral envelope modeling approach in order to estimate the harmonic magnitudes $a_h(n)$ over the duration of each note. Here, we decided to use a simpler, less computationally expensive approach here. Based on the fundamental frequency contour $f_0(n)$, we compute the harmonic frequencies $f_h(n)$ using Equation (4). Then, we estimate the frame-wise harmonic magnitudes $a_h(n)$ of the first 10 partials (including the fundamental frequency) using linear interpolation from the STFT magnitude spectrogram M.

[3] The template function is obtained by a convolution with a Hanning window of width 5.

F. Intensity Estimation & Segmentation Into Attack and Decay

We compute the *intensity* of the *i*-th note from the envelope peak log-magnitude $\mathcal{L}(i) = 20\log_{10} a\left(n_{\text{Peak}}\right)$. The peak frame n_{Peak} is the boundary frame between the note's attack and decay part and is computed as $n_{\text{Peak}} = \underset{n}{\operatorname{argmax}}\, a(n)$ using the accumulated magnitude envelopes of all partials $a(n) = \sum_{h=0}^{9} a_h(n)$.

G. Feature Extraction

This section will summarize the audio features (denoted as χ) that we extract to capture different timbre properties of bass guitar notes.

1) Harmonic Magnitude and Frequency Relationships: The first set of features describes the shape of the aggregated magnitude envelope $a(n)$. Following the two-stage envelope model (compare Section III-F), $a(n)$ is modeled as an increasing linear function over the attack part as

$$a(n) \approx \theta_1 n + \theta_0 \text{ for } n \in [n_{\text{On}}, n_{\text{Peak}}] \tag{5}$$

and as a decreasing exponential function over the decay part as

$$a(n) \approx a(n_{\text{Peak}}) e^{-\delta_1 n} \text{ for } n \in [n_{\text{Peak}}, n_{\text{Off}}]. \tag{6}$$

We use linear regression to estimate θ_1 and δ_1 and use both as features.

The second feature set describes the magnitude and frequency relationships of the partials in the note's peak frame n_{Peak}. We compute the *relative harmonic magnitudes*

$$\chi_{\text{a,rel}}(h) = a_h(n_{\text{Peak}})/a_0(n_{\text{Peak}}) \text{ for } h \in [1,9] \tag{7}$$

and the *inharmonicity coefficient* $\beta(n_{\text{Peak}})$ (compare Section III-C) as features. Using linear regression, we interpolate the harmonic magnitudes as a decaying linear function over the harmonic index h as $a_h \approx \gamma_1 h + \gamma_0$ and use γ_1 as feature to measure the spectral magnitude decay over frequency.

Based on the fundamental frequency $f_0(n_{\text{Peak}})$ and the inharmonicity coefficient β, we compute the hypothetical harmonic frequencies $\hat{f}_h(n_{\text{Peak}})$ using Equation 4. In pratice, the measured harmonic frequency values $f_h(n_{\text{Peak}})$ deviate from $\hat{f}_h(n_{\text{Peak}})$, hence, we compute the *normalized frequency deviations*

$$\chi_{\Delta,\text{f}}(h) = \frac{\hat{f}_h(n_{\text{Peak}}) - f_h(n_{\text{Peak}})}{f_h(n_{\text{Peak}})} \text{ for } h \in [1,9] \tag{8}$$

as features. Finally, we compute the statistical measures minimum, maximum, mean, median, variance, skewness, and kurtosis over the two vectors $\chi_{\text{a,rel}}$ and $\chi_{\Delta,\text{f}}$ to derive additional timbre features.

2) Aggregated Timbre Features: In order to characterize the overall spectral envelope in the attack and decay part of a note, we first compute the frame-wise features *tristimulus* and *spectral irregularity* from the harmonic magnitudes $a_h(n)$ as proposed in [40] as well as the *spectral centroid*, *spectral crest factor* (plus the first time derivative), *spectral roll-off*, *spectral slope*, and *spectral spread* from the magnitude spectrogram M as proposed in [40], [41]. These frame-wise features are aggregated separately over the duration of the attack and the decay part using the same statistical measures as for $\chi_{\text{a,rel}}$ and $\chi_{\Delta,\text{f}}$ in the previous paragraph.

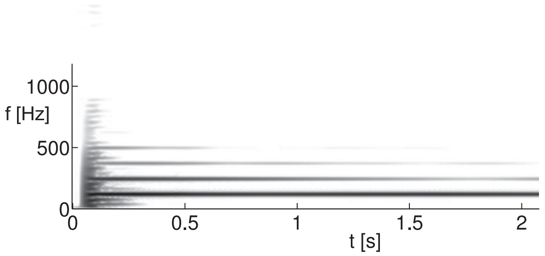

Fig. 5. STFT spectrogram M of a bass guitar note played with the *harmonics* expression style on the low E-string. It can be observed that observed fundamental frequency of $f_0 = 123.6\,Hz$ corresponds to the third harmonic of the open string fundamental frequency of 41.2 Hz [4].

3) Noisiness: Notes played with the plucking styles *slap-thumb* and *slap-pluck* as well as the expression style *dead-note* exhibit a percussive, wide-band characteristic in the attack part of the magnitude spectrogram. In order to capture this spectral characteristic, we remove the harmonic peaks from a given spectral frame M by applying a harmonic spectral template, which is tuned to the fundamental frequency $f_0(n)$ and the inharmonicity coefficient β, as filter. The remaining magnitude spectrogram is stored in M_R. Then, we compute the energy ratio between the filtered and the original magnitude spectrogram, averaged over the note attack part as feature:

$$\chi_{\text{Noisiness}} = \frac{1}{n_{\text{Peak}} - n_{\text{On}}} \sum_{n=n_{\text{On}}}^{n_{\text{Peak}}} \frac{\sum_k M_R(k,n)}{\sum_k M(k,n)}. \tag{9}$$

4) Subharmonic Energy: Fig. 5 illustrates a bass guitar note that was played on the seventh fret position on the low E string with the *harmonics* (HA) expression style. The fundamental frequency of the open E string of 41.2 Hz (and the corresponding harmonic frequencies) can be observed in the beginning of the note decay part (between 0.1 s and 0.25 s). However, due to the string damping at a third of the overall string length, only the third vibration mode and its octaves remain audible after $t = 0.25\,s$. All other vibration modes are damped. As a consequence, the perceived fundamental frequency is three times higher ($f_0 = 3 \cdot 41.2\,Hz = 123.6\,Hz$) than the fundamental frequency of the open string. The harmonics of the open E string's fundamental frequency can be interpreted as subharmonics of f_0.

We propose to compute audio features to characterize the spectral energy at *subharmonic frequency positions* as follows. We compute harmonic spectral templates as described in Section III-C that are tuned to different virtual fundamental frequencies f_0/m with the subharmonic indices $m \in [2,7]$.[4] At the same time, these spectral templates are modified in such way that they have no spectral peaks at multiples of the "real" fundamental frequency f_0. We compute a likelihood (in a loose sense) $\chi_{\text{sub,m}}$ for f_0/m by multiplying the spectrum with these

[4] The highest vibration mode played with the *harmonics* style in the evaluation dataset is $m = 7$.

modified spectral templates and computing the energy sum ratio similar to Equation 9. High likelihood values indicate the presence of subharmonic components.

If *harmonics* are played on a particular string, then the correspoding harmonic peaks are likely to be filtered out by a spectral template that is tuned to the open string fundamental frequency. Therefore, we compute *string likelihood values* using a similar approach as in Equation 9 using spectral templates tuned to the fundamental frequencies of all four bass guitar strings.[5]

5) Fundamental Frequency Modulation: The f_0 contour of notes played with the three expression styles *vibrato*, *bending*, and *slide* have characteristic shapes. While the vibrato results in a periodic f_0 modulation, notes played with bending usually show an increase and subsequent decrease in f_0 due to bending of the vibrating string and changing its length. Slide notes are characterized here by a initial period of stable pitch followed by a continuous increase or decrease in pitch.

We use the following features to characterize the f_0 contours. First, we estimate the *mean modulation frequency* from the first non-zero lag position of a local maximum in the autocorrelation function over $f_0(n)$. Also, we compute the *number of modulation quarter-periods* as features since a *slide* is usually characterized by one ascending or descending f_0 contour part per note while *bendings* include a rise and successive fall and the *vibrato* technique shows multiple modulation periods of the f_0 contour. We compute the *modulation lift* as $1200 \log_2 \left(\frac{\max f_0(n)}{\min f_0(n)} \right)$ in cent relative to the lowest f_0 value. Finally, we measure the *overall pitch progression* as the difference between the average fundamental frequency in the last 30 % and the first 30 % of the note frames, also given in cent.

H. Estimation of Plucking Style & Expression Style

Using the training set (depending on the evaluation experiment setting), we train three independent classifier models for the classification of the string number, plucking style, and expression style. The training set feature matrix is normalized to zero mean and unit variance. Then, the feature selection method *Inertia Ratio Maximization using Feature Space Projection* (IRMFSP) proposed in [42] is applied to reduce the dimensionality of the feature space to $N_D = 50$ to avoid overfitting. Then, a *Support Vector Machine* (SVM) classifier [43][6] with a *Radial Basis Function* (RBF) kernel is trained for each of the three classification tasks. Finally, during the transcription process, we estimate the *string number*, *plucking style*, and *expression style* parameters by maximizing the class probability values obtained from the trained classifiers based on the extracted feature vector.

I. Estimation of String Number & Fret Number

Depending on the expression style $\mathcal{S}_E$, three different scenarios must be distinguished. Firstly, for *dead-notes*, the fret number is considered to be irrelevant and the string number is either randomly set for single notes, or set to the string number of the closest note in the transcribed bass signal played with one of the expression styles *normal*, *vibrato*, *bending*, or *slide*, which results in an easier to play bass line. Secondly, if notes are played using the *harmonics* expression style, we estimate the string number by maximizing the aforementioned string likelihood values and mode number by maximizing the mode likelihood values as explained in Section III-G4. Most *harmonics* can be played at multiple fret positions, hence, we chose the fret position as close as possible to the previous note. For all other expression styles, the string number $\mathcal{N}_S$ is automatically classified using the classification approach explained in the previous paragraph. Based on the string number, the note pitch $\mathcal{P}$, and the pitch of the classified string $\mathcal{P}_{\text{String}}(\mathcal{N}_S)$, we compute the fret number as

$$\mathcal{N}_F = \mathcal{P} - \mathcal{P}_{\text{String}}(\mathcal{N}_S). \tag{10}$$

The MIDI pitch values associated with the standard open string tuning of four-string bass guitar are $\mathcal{P}_{\text{String}} = [28, 33, 38, 43]$.

TABLE II
EVALUATION DATASETS.

Dataset	Abbr.	Description	# Files	Dur. [min]
IDMT-SMT-BASS	ISB	Solo bass guitar note recordings	4212	176.7
IDMT-SMT-BASS-VAR-DUR	ISBVD	Solo bass guitar note sequences with variable note durations	30	16.8
IDMT-SMT-BASS-SINGLE-TRACKS	ISBST	Realistic solo bass guitar lines	17	5.9

IV. Evaluation

A. Datasets

Given the proposed instrument-centered music transcription algorithm, two main tasks need to be evaluated: the estimation of the score-level parameters, which corresponds to the classical music transcription task, and the estimation of the instrument-level parameters, which consists of three classification tasks, namely the classification of the string number, the plucking style, and the expression style. Each task requires different annotations and evaluation measures. Three datasets as listed in Table II were used in the experiments and made available online[7] as a public evaluation benchmark.

The *IDMT-SMT-BASS* (ISB) dataset contains 4734 isolated note recordings performed on three different 4-string bass guitars with various pickup settings. The included notes cover the full fret range up to the 12th fret of a 4 string bass guitar as well as all discussed plucking and expression styles listed in Table I. In the ISB dataset, all recorded notes are plucked once and then decay without additional damping due to subsequent

[5] We restrict ourselves to four strings here, since the evaluation datsets discussed in Section IV-A do not include notes played on the low B string of a five-string bass guitar.

[6] Using the LibSVM library [44].

[7] http://www.idmt.fraunhofer.de/en/business_units/m2d/research.html

note events. However, in realistic bass lines, the magnitude envelopes of notes only rarely decay without interruption. Instead, notes have different durations and are therefore are often damped during the decay part by the plucking hand. We expect this phenomenon to have significant effect on the feature extraction step discussed in Section III-G and therefore on the trained classification models.

The *IDMT-SMT-BASS-VAR-DUR* (ISBVD) dataset was recorded to address this problem. The first part of the dataset contains note sequences that were played individually on each bass guitar string with varying plucking styles and fixed expression styles. The second part contains note sequences that were played accross the strings with varying plucking styles and fixed expression style and the other way around. For both sets, only the instrument-level parameters are given as ground truth annotation. The score parameters pitch, onset, and duration were not annotated for this dataset so far.

This *IDMT-SMT-BASS-SINGLE-TRACKS* (ISBST) dataset includes 17 bass guitar recordings of realistic basslines taken from the music genres blues, funk, rock, bossa nova, forró, and hip hop. The dataset includes most combinations of plucking and expression styles and cover most fretboard positions. It includes full annotation of both score-level and instrument-level parameters.

TABLE III
FRAME-BASED EVALUATION RESULTS FOR SCORE-LEVEL EVALUATION.

Algorithm	Evaluation Measures				
	VRC	VFAR	RPA	RCA	OA
RK	0.835	0.209	0.696	0.794	0.728
SG	**0.934**	0.296	0.701	**0.820**	0.698
DDR	0.741	0.291	0.585	0.624	0.606
pYIN	0.604	**0.137**	0.507	0.54	0.593
AS (proposed)	0.890	0.427	**0.765**	0.796	**0.735**

The best performing algorithms are indicated in bold print.

TABLE IV
NOTE-BASED EVALUATION RESULTS FOR SCORE-LEVEL EVALUATION.

Algorithm	Evaluation Measures		
	R	P	F
RK	0.751	0.841	0.787
DDR	0.512	0.815	0.599
pYIN	0.584	0.881	0.685
AS (proposed)	**0.897**	**0.908**	**0.901**

The best performing algorithms are indicated in bold print.

B. Experiments

1) Performance of the Score-level Estimation (Bass Transcription): In this experiment, we evaluated the performance of the proposed bass guitar transcription algorithm against three state-of-the-art bass transcription algorithms. We used the 17 basslines from the ISBST dataset introduced in Section IV-A as test dataset. In particular, the annotated score-level parameters onset, offset, and pitch are used as ground truth reference data. The proposed transcription algorithm (denoted as **AS**) was compared against the bass transcription algorithms by Ryynänen & Klapuri [10] (**RK**), Salamon & Gómez [13] (**SG**), and Dittmar *et al.* [9] (**DDR**). In addition, we include the **pYIN** algorithm proposed by Mauch & Dixon [14] as representative melody estimation algorithm into the evaluation (compare Section II-A).[8] It must be noted that the algorithm **SG** is limited to a two-octave pitch range between the MIDI pitch values 21 and 45 (f_0 values between 27.5 Hz - 110 Hz). Also, as a melody transcription algorithm, the **pYIN** algorithm focusses on a MIDI pitch range between 33 (55 Hz) to 81 (880 Hz). Hence it is not capable to detect lower notes, which are present in the test data.

The algorithms **RK**, **DDR**, **pYIN**, and **AS** provide note-based transcription results, i. e., note onset and offset position with corresponding MIDI pitch values. The algorithm **SG** provides only frame-based estimates of the fundamental frequency and can therefore only be evaluated using the frame-based evaluation measures. We derive frame-based f_0 values from the transcription results from the algorithms **RK**, **DDR**, **pYIN**, and **AS** using the same temporal resolution of $\Delta t = 5.8$ ms as in **SG** in order to compare the results.

For the *note-based evaluation*, we compute the Recall (R), Precision (P), and F-measure (F). As proposed in [10], we consider a note to be correctly transcribed, if it can be assigned to a reference note with the same MIDI pitch and a note onset within an absolute range of 150 ms. For the *frame-based evaluation*, we compute the Voicing Recall Rate (VCR), Voicing False Alarm Rate (VFAR), Raw Pitch Accuracy (RPA), Raw Chroma Accuracy (RCA), as well as Overall Accuracy (OA) as *frame-based evaluation measures* as detailed in [45] and [46]. These standard measures are used in the Audio Melody Extraction task as part of the MIREX (Music Information Retrieval Evaluation eXchange) campaign [46].

The frame-based evaluation measures are shown in Table III. In terms of pitch estimation, the proposed algorithm **AS** outperforms the others with a Raw Pitch Accuracy of $\mathrm{RPA} = 0.765$. However, if the octave information is neglected, algorithm **SG** shows the best performance for the Raw Chroma Accuracy with $\mathrm{RCA} = 0.82$. Keeping in mind that the algorithm **SG** only considers a limited pitch range of two octaves, a better performance of **SG** for the Raw Chroma Accuracy is likely if a larger pitch range would be considered by the algorithm. In terms of the detection of voiced frames, algorithm **SG** outperforms the others with the highest Voicing Recall Rate of $\mathrm{VRC} = 0.934$. Interestingly, the **pYIN** algorithm shows the lowest Voicing False Alarm Rate.

The note-based evaluation measures are summarized in Table IV. The proposed algorithm **AS** clearly outperforms the other two algorithms **RK** and **DDR** in recall ($\mathrm{R} = 0.897$), precision ($\mathrm{P} = 0.908$), and F-measure ($\mathrm{F} = 0.901$). While **RK** and **DDR** show comparable precision values, **RK** clearly has

[8] Using the publically-available VAMP plugin implementation from http://vamp-plugins.org/, we obtain the frame-wise and note-wise transcription results from the "smoothed-pitchtrack" and "notes" parameter, respectively.

TABLE V
SUMMARIZED EVALUATION RESULTS FOR ESTIMATION OF THE INSTRUMENT-RELATED PARAMETERS PLUCKING STYLE, EXPRESSION STYLE, AND STRING NUMBER

Plucking Style								Expression Style						String Number			
Experiment 1: Cross-Validation with the ISB dataset																	
	FS	**MU**	**PK**	**SP**	**ST**		**BE**	**DN**	**HA**	**NO**	**SL**	**VI**		**E**	**A**	**D**	**G**
FS	**0.95**	0.02	0.02	0.00	0.01	**BE**	**0.92**	0.00	0.01	0.02	0.04	0.02	**E**	**0.94**	0.04	0.01	0.01
MU	0.05	**0.94**	0.00	0.00	0.01	**DN**	0.00	**0.98**	0.01	0.01	0.00	0.00	**A**	0.07	**0.83**	0.07	0.02
PK	0.01	0.00	**0.98**	0.00	0.00	**HA**	0.00	0.01	**0.96**	0.02	0.00	0.01	**D**	0.02	0.06	**0.82**	0.10
SP	0.01	0.00	0.00	**0.97**	0.01	**NO**	0.01	0.01	0.02	**0.95**	0.00	0.02	**G**	0.00	0.01	0.07	**0.92**
ST	0.03	0.00	0.01	0.02	**0.94**	**SL**	0.02	0.00	0.00	0.00	**0.98**	0.01					
						VI	0.02	0.00	0.02	0.03	0.04	**0.89**					
Experiment 2: Cross-Validation with the ISBVD dataset																	
	FS	**MU**	**PK**	**SP**	**ST**		**BE**	**DN**	**HA**	**NO**	**SL**	**VI**		**E**	**A**	**D**	**G**
FS	**0.92**	0.04	0.01	0.01	0.02	**BE**	**0.83**	0.00	0.00	0.00	0.09	0.09	**E**	**0.96**	0.04	0.00	0.00
MU	0.06	**0.92**	0.01	0.00	0.00	**DN**	0.00	**0.96**	0.04	0.00	0.00	0.00	**A**	0.05	**0.90**	0.02	0.03
PK	0.01	0.01	**0.96**	0.00	0.01	**HA**	0.00	0.04	**0.91**	0.00	0.04	0.00	**D**	0.02	0.02	**0.86**	0.11
SP	0.01	0.01	0.00	**0.63**	0.35	**NO**	0.00	0.00	0.04	**0.74**	0.22	0.00	**G**	0.00	0.00	0.05	**0.95**
ST	0.01	0.00	0.01	0.16	**0.82**	**SL**	0.17	0.04	0.00	0.30	**0.39**	0.09					
						VI	0.30	0.00	0.00	0.00	0.13	**0.57**					
Experiment 3: Training with the ISB dataset, Prediction on the ISBST dataset																	
	FS	**MU**	**PK**	**SP**	**ST**		**BE**	**DN**	**HA**	**NO**	**SL**	**VI**		**E**	**A**	**D**	**G**
FS	**0.74**	0.01	0.01	0.00	0.23	**BE**	**0.88**	0.00	0.00	0.12	0.00	0.00	**E**	**0.52**	0.37	0.03	0.08
MU	0.55	**0.07**	0.01	0.03	0.34	**DN**	0.46	**0.00**	0.14	0.04	0.32	0.04	**A**	0.08	**0.35**	0.38	0.18
PK	0.65	0.00	**0.03**	0.02	0.29	**HA**	0.19	0.12	**0.50**	0.00	0.09	0.09	**D**	0.07	0.03	**0.43**	0.46
SP	0.62	0.04	0.00	**0.24**	0.10	**NO**	0.81	0.00	0.00	**0.02**	0.15	0.01	**G**	0.04	0.02	0.27	**0.68**
ST	0.49	0.02	0.00	0.16	**0.33**	**SL**	0.90	0.00	0.00	0.05	**0.05**	0.00					
						VI	0.94	0.00	0.00	0.00	0.06	**0.00**					
Experiment 4: Training with the ISBVD dataset, Prediction on the ISBST dataset																	
	FS	**MU**	**PK**	**SP**	**ST**		**BE**	**DN**	**HA**	**NO**	**SL**	**VI**		**E**	**A**	**D**	**G**
FS	**0.93**	0.02	0.00	0.01	0.03	**BE**	**0.62**	0.00	0.00	0.12	0.12	0.12	**E**	**0.81**	0.15	0.03	0.00
MU	0.68	**0.30**	0.00	0.01	0.01	**DN**	0.00	**0.25**	0.25	0.36	0.14	0.00	**A**	0.13	**0.54**	0.33	0.00
PK	0.57	0.00	**0.31**	0.02	0.09	**HA**	0.00	0.22	**0.47**	0.12	0.19	0.00	**D**	0.00	0.13	**0.56**	0.31
SP	0.14	0.00	0.04	**0.14**	0.68	**NO**	0.03	0.06	0.00	**0.53**	0.26	0.12	**G**	0.00	0.02	0.17	**0.82**
ST	0.24	0.00	0.00	0.12	**0.64**	**SL**	0.00	0.00	0.00	0.55	**0.35**	0.10					
						VI	0.26	0.00	0.00	0.35	0.23	**0.16**					

Confusion matrices are provided for four experiments detailed in Section IV-B2 with varying training and test sets. In the confusion matrices, rows correspond to the true labels and columns correspond to the estimated class labels.

the higher recall value in the direct comparison. The **pYIN** algorithm outperforms the **DDR** algorithm and shows a particular high precision value compared to the other algorithms except **AS**.

As discussed in Section III-B and Section III-C, we used a small subset of isolated notes recorded with the same bass guitar that was used to record the test data for this experiment (although keeping both datasets strictly separated) as development set for the algorithm development. However, we believe that the obtained results can be interpreted as upper performance limit under these (idealized) conditions, which are for instance relevant in music tuition software, where the transcription algorithm can be adapted to the user's instrument sound characteristics for instance using parts of the recorded audio data as additional training data for parameter optimization.

2) Performance of the Instrument-Level Estimation: In the second experiment, we aimed to evaluate the proposed algorithm for the estimation of the three instrument-level parameters plucking style, expression style, and string number. In particular, we wanted to investigate the influence of the training and test set note duration characteristics on the algorithms's performance. As discusssed in Section IV-A, the ISB dataset includes isolated note recordings with long note decay times. In contrast, the ISBVD and ISBST datsets contain notes with varying durations, which are more likely to be observed in real-world bass lines.

We used four different combinations of training and test sets. In the first two experiments, we performed a 10-fold cross-validation using the ISB and the ISBVD dataset, respectively. We used random sampling to compensate for unequal class sizes and performed feature normalization, feature selection of the best 50 features using IRMFSP, and classification using Support Vector Machine (SVM) classifier with RBF kernel function. We computed the confusion matrix and the mean class accuracy $\overline{A}$ as evaluation measures. In the third and fourth experiment, we used the ISB and ISBVD datasets for training the models, respectively, and made predictions on the ISBST dataset, which contain the most realistic bass lines. In these experiments, we computed features from the ISBST dataset based on the ground truth transcription annotation, i. e. the correct note onset, duration, and pitch values.

Table V illustrates the confusion matrices and the mean class accuracy values obtained for each of the four combinations and each of the three parameters. The results can be summarized as follows. As expected, the accuracy values for the classification of plucking styles and expression styles are very high for isolated bass notes in experiment 1 ($\overline{\mathrm{A}} = 0.96$ and $\overline{\mathrm{A}} = 0.95$) while they are lower for variable-duration notes in experiment 2 ($\overline{\mathrm{A}} = 0.85$ and $\overline{\mathrm{A}} = 0.73$). Interestingly, the classification of the string number is slighly better in experiment 2 with $\overline{\mathrm{A}} = 0.92$ compared to experiment 1 with $\overline{\mathrm{A}} = 0.88$.[9] For all three parameters, the classification performance for testing with realistic data (experiment 3 and 4) is significantly worse compared to the cross-validation experiments 1 and 2. We assume both the variable note durations as well as the multitude of combinations of plucking and expression styles, which are not covered in the training sets (compare Section IV-A), to further complicate the classification tasks.

By comparing the results for experiments 3 and 4, we could confirm that for the real-world application use cases, a model training with variable-duration notes (experiment 4) improves the model performance (increase in accuracy of 0.18, 0.16, and 0.18 for the plucking style, expression style, and string number, respectively). Based on the confusion matrices, we observe that most misclassifications for the string number classification tasks are between neighbored strings. Human listeners show the same type of confusions as we observed in listening tests [32]. As discussed in [3], musical knowledge about typical finger positions in bass lines of particular music genres could be used in an additional error correction step to further reduce this kind of confusions. However, since the bass lines in ISBST dataset cover various music genres (compare Section IV-A), we did not follow this approach here.

For the plucking style classification (experiment 3 and 4), most confusions lead towards the finger-style class, which is the most common plucking style. Similarly, most confusions for the expression style classification lead towards the bending style (experiment 3) and the normal style (experiment 4). Often, bendings show only small deviations from the original note pitch and therefore sound very similar to normal notes.

V. Conclusion

In this paper, we proposed a novel bass transcription algorithm, which is tailored towards the instrument characteristics of the electric bass guitar. The algorithm extracts note parameters on two levels—the score-level and the instrument-level. We showed in different experiments that the proposed system performed well for the estimation of plucking style, expression style, as well as string number from isolated note recordings. In the score-level evaluation, the proposed method could outperform three state-of-the-art algorithms for bass transcription, which were originally designed for analyzing polyphonic audio content instead of solo bass recordings.

We furhermore investigated the influence of the note duration characteristics in the training data on the performance of plucking style, expression style, and string classification. We could show that the classification results improve, if the algorithm is trained with instrument recordings having similar rhythmic and durational properties as the test data. In general, we believe that the design of instrument-centered transcription algorithms could be a valid approach to break the glass ceiling for automatic music transcription algorithms, which was often reported in the Music Information Retrieval (MIR) literature in the past years. While these algorithms allow to extract a larger set of parameters from recorded instrument performances, they usually require the instrument track to be recorded in isolation or to be extracted from a musical mixture using source separation algorithms.

Acknowledgment

The authors would like to thank C. Dittmar, J. Salamon, M. Ryynänen, and A. Klapuri for providing their algorithms' transcription results and M. Mauch for publishing the pYIN VAMP plugin.

References

[1] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions." *J. Intell. Inf. Syst.*, vol. 41, no. 3, pp. 407–434, 2013.

[2] A. Perez-Carrillo and M. Wanderley, "Indirect acquisition of violin instrumental controls from audio signal with hidden Markov models," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 5, pp. 932–940, May 2015.

[3] J. Abeßer and G. Schuller, "Instrument-centered music transcription of bass guitar tracks," in *Proc. AES 53rd Conf. Semantic Audio*, London, U.K., 2014, pp. 166–175.

[4] J. Abeßer, "Automatic transcription of bass guitar tracks applied for music genre classification and sound synthesis," Ph.D. dissertation, Ilmenau Univ. of Technology, Ilmenau, Germany, 2014.

[5] A. Loscos, Y. Wang, and W. J. J. Boo, "Low level descriptors for automatic violin transcription," in *Proc. 7th Int. Conf. Music Inf. Retrieval*, Victoria, Canada, 2006, pp. 164–167.

[6] S. W. Hainsworth and M. D. Macleod, "Automatic bass line transcription from polyphonic music," in *Proc. Int. Comput. Music Conf.*, La Habana, Cuba, 2001, pp. 431–434.

[7] Y. Uchida and S. Wada, "Melody and bass line estimation method using audio feature database," in *Proc. IEEE Int. Conf. Signal Process. Commun. Comput.*, 2011, pp. 1–6.

[8] E. Tsunoo, N. Ono, and S. Sagayama, "Musical bass-line pattern clustering and its application to audio genre classification," in *Proc. 10th Int. Soc. Music Inf. Retrieval Conf.*, Kobe, Japan, 2009, pp. 219–224.

[9] C. Dittmar, K. Dressler, and K. Rosenbauer, "A toolbox for automatic transcription of polyphonic music," in *Proc. Audio Mostly Conf.*, 2007, pp. 58–65.

[10] M. P. Ryynänen and A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Comput. Music J.*, vol. 32, pp. 72–86, Apr. 2008.

[11] M. Goto, "A real-time music-scene-description system - predominant-F0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Commun.*, vol. 43, no. 4, pp. 311–329, Sep. 2004.

[12] E. Tsunoo, G. Tzanetakis, N. Ono, and S. Sagayama, "Beyond timbral statistics: Improving music classification using percussive patterns and bass lines," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 4, pp. 1003–1014, May 2011.

[13] J. Salamon and E. Gómez, "A chroma-based salience function for melody and bass line estimation from music audio signals," in *Proc. 6th Sound Music Comput. Conf.*, Porto, Portugal, 2009, pp. 23–25.

[14] M. Mauch and S. Dixon, "pYIN: A fundamental frequency estimator using probabilistic threshold distributions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Florence, Italy, 2014, pp. 659–663.

[15] L. S. Pardue, C. Harte, and A. P. McPherson , "A low-cost real-time tracking system for violin," *J. New Music Res.*, vol. 44, no. 4, pp. 305–323, 2015.

[9] We did not apply the aggregation over multiple frame-based features for the string number classification as proposed in [32] in order to allow the system to process notes with arbitrary note duration.

[16] J. Yin, Y. Wang, and D. Hsu, "Digital violin tutor: An integrated system for beginning violin learners," in *Proc. ACM Multimedia*, Singapore, 2005, pp. 976–985.

[17] Y.-L. Chen, T.-M. Wang, W.-H. Liao, and A. W. Y. Su, "Analysis and trans-synthesis of acoustic bowed-string instrument recordings—A cast study using bach cello suites," in *Proc. 14th Int. Conf. Dig. Audio Effects*, Paris, France, 2011, pp. 63–67.

[18] X. Fiss and A. Kwasinski, "Automatic real-time electric guitar audio transcription," in *Proc. IEEE Conf. Acoust., Speech, Signal Process.*, Praha, Czech Republic, 2011, pp. 373–376.

[19] C. Kehling, J. Abeßer, C. Dittmar, and G. Schuller, "Automatic tablature transcription of electric guitar recordings by estimation of score- and instrument-related parameters," in *Proc. 17th Int. Conf. Digit. Audio Effects*, Erlangen, Germany, 2014, pp. 1–8.

[20] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 6, pp. 1643–1654, Aug. 2010.

[21] E. Maestre, M. Blaauw, J. Bonada, E. Guaus, and A. Pérez, "Statistical modeling of bowing control applied to violin sound synthesis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 4, pp. 855–871, May 2010.

[22] A. P. Carrillo, J. Bonada, E. Maestre, E. Guaus, and M. Blaauw, "Performance control driven violin timbre model based on neural network," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 3, pp. 1007–1021, Mar. 2012.

[23] C. Erkut, M. Karjalainen, and M. Laurson, "Extraction of physical and expressive parameters for model-based sound synthesis of the classical guitar," in *Proc. 108th Audio Eng. Soc. Conv.*, 2000, pp. 19–22.

[24] E. Guaus, T. Ozaslan, E. Palacios, and J. L. Arcos, "A left hand gesture caption system for guitar based on capacitive sensors," in *Proc. 10th Int. Conf. New Interfaces Musical Expression*, Sydney, Australia, 2010, pp. 238–243.

[25] Y.-P. C. Chen, L. Su, and Y.-H. Yang, "Electric guitar playing technique detection in real-world recordings based on F0 sequence pattern recognition," in *Proc. 16th Int. Soc. Music Inf. Retrieval Conf.*, 2015, pp. 708–714.

[26] L. Reboursière, O. Lähdeoja, T. Drugman, S. Dupont, C. Picard-Limpens, and N. Riche, "Left and right-hand guitar playing techniques detection," in *Proc. Int. Conf. New Interfaces Muscial Expression*, Ann Arbor, MI, USA, 2012, pp. 1–4.

[27] T. H. Özaslan, E. Guaus, E. Palacios, and J. L. Arcos, "Attack based articulation analysis of nylon string guitar," in *Proc. 7th Int. Symp. Comput. Music Model. Retrieval*, Málaga, Spain, 2010, pp. 285–298.

[28] A. Krishnaswamy and J. O. Smith, "Inferring control inputs to an acoustic violin from audio spectra," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Baltimore, MD, USA, 2003, pp. 3–6.

[29] I. Barbancho, C. de la Bandera, A. M. Barbancho, and L. J. Tardón, "Transcription and expressiveness detection system for violin music," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Taipei, Taiwan, Apr. 2009, pp. 189–192.

[30] C. Traube and J. O. Smith, "Extracting the fingering and the plucking points on a guitar string from a recording," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, New Paltz, NY, USA, 2001, pp. 2–5.

[31] I. Barbancho, A. M. Barbancho, S. Sammartino, and L. J. Tardón, "Pitch and played string estimation in classic and acoustic guitars," in *Proc. 126th Audio Eng. Soc. Conv.*, Munich, Germany, 2009, pp. 1–9.

[32] J. Abeßer, "Automatic string detection for bass guitar and electric guitar," in *From Sounds to Music and Emotions*, M. Aramaki, M. Barthet, R. Kronland-Martinet, and S. Ystad, Eds., vol. 7900. London, U.K.: Springer-Verlag, 2013, pp. 333–352.

[33] I. Barbancho, S. Member, L. J. Tardón, S. Sammartino, and A. M. Barbancho, "Inharmonicity-based method for the automatic generation of guitar tablature," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 6, pp. 1857–1868, Aug. 2012.

[34] A. Perez-Carrillo, J.-L. Arcos, and M. Wanderley, *Estimation of Guitar Fingering and Plucking Controls Based on Multimodal Analysis of Motion, Audio and Musical Score*. New York, NY, USA: Springer-Verlag, 2016, pp. 71–87.

[35] E. Humphrey and J. Bello, "From music audio to chord tablature: Teaching deep convolutional networks toplay guitar," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2014, pp. 6974–6978.

[36] F. Auger and P. Flandrin, "Improving the readability of time-frequency and time-scale representations by the reassignment method," *IEEE Trans. Signal Process.*, vol. 43, no. 5, pp. 1068–1089, May 1995.

[37] M. Lagrange and S. Marchand, "Estimating the instantaneous frequency of sinusoidal components using phase-based methods," *J. Audio Eng. Soc.*, vol. 55, no. 5, pp. 385–399, 2007.

[38] S. Dixon, "Onset detection revisited," in *Proc. 9th Int. Conf. Digit. Audio Effects*, Montréal, QC, Canada, 2006, pp. 1–6.

[39] N. H. Fletcher and T. D. Rossing, *The Physics Of Musical Instruments*, 2nd ed. New York, NY, USA: Springer-Verlag, 1998.

[40] K. Jensen, "Timbre Models of Musical Sounds," Ph.D. dissertation, Univ. of Copenhagen, Copenhagen, Denmark, 1999.

[41] Geoffroy Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," IRCAM, Analysis/Synthesis Team, Paris, France, Tech. Rep., 2004.

[42] G. Peeters and X. Rodet, "Hierarchical gaussian tree with inertia ratio maximization for the classification of large musical instruments databases," in *Proc. 6th Int. Conf. Digit. Audio Effects*, London, U.K., 2003, pp. 1–6.

[43] V. N. Vapnik, *Statistical Learning Theory*, 1st ed. New York, NY, USA: Wiley, 1998.

[44] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011.

[45] G. E. Poliner, D. P. W. Ellis, A. F. Ehmann, E. Gómez, S. Streich, and B. Ong, "Melody transcription from music audio - approaches and evaluation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 4, pp. 1247–1256, May 2007.

[46] J. Salamon, E. Gómez, D. P. Ellis, and G. Richard, "Melody extraction from polyphonic music signals : Approaches, applications and challenges," *IEEE Signal Process. Mag.*, vol. 31, no. 2, pp. 118–134, Mar. 2013.

Jakob Abeßer received Dipl.Ing. degree in computer engineering and Ph.D. degree (Dr. Ing.) in media technology in 2014 from Ilmenau University of Technology, Ilmenau, Germany.

He is a Postdoctoral Researcher in the Semantic Music Technologies group at Fraunhofer IDMT. During his Ph.D. degree, he was a Visiting Researcher in the Finnish Centre of Excellence in Interdisciplinary Music Research, University of Jyväskylä, Finland in 2010. As a Research Scientist at Fraunhofer, he has experience with algorithm & software development in the fields of automatic music transcription, symbolic music analysis, machine learning, and music instrument recognition. Also, from 2012 until 2017, he worked as a Postdoctoral Researcher in the Liszt School of Music in Weimar in the Jazzomat Research Project, focussing on analyzing Jazz solo recordings using music information retrieval technologies.

Gerald Schuller received the Diploma degree in electrical engineering from the Technical University of Berlin, Berlin, Germany, in 1989, and the Ph.D. (Dr. Ing.) degree from the University of Hanover Hanover, Germany, in 1997, studied at the Massachusetts Institute of Technology Cambridge, MA, USA, in 1989/1990 and at the Georgia Institute of Technology, Atlanta, GA, USA, in 1993.

Since 2008, he has been a Full Professor in the Institute for Media Technology of the Technical University of Ilmenau, Ilmenau, Germany. He was the Head of the Audio Coding for Special Applications group, Fraunhofer Institute for Digital Media Technology, Ilmenau, Germany, since January 2002 until 2008, and is now a member of Fraunhofer IDMT. Before joining the Fraunhofer Institute, he was a member of Technical Staff at Bell Laboratories, Lucent Technologies, and Agere Systems, a Lucent Spin-off, from 1998 to 2001. There he worked in the Multimedia Communications Research Laboratory.

Dr. Schuller was an Associate Editor of the IEEE Transactions on Speech and Audio Processing from 2002 until 2006, and the IEEE Transactions on Signal Processing from 2006 to 2009, and of the IEEE Transactions on Multimedia from 2008 to 2010. He received the 2006 IEEE Best Paper Award in the Audio and Electroacoustics Area. His research interests include filter banks, audio coding, music signal processing, and 3-D audio and visual object representations.