Progress towards automatic music transcription

Anssi Klapuri and Tuomas Virtanen*

November 6, 2006

1 Introduction

Written musical notation describes music in a symbolic form that is suitable for performing a piece using the available musical instruments. Traditionally, musical notation indicates the pitch, target instrument, timing, and duration of each sound to be played. The aim of music transcription either by humans or a machine is to infer these musical parameters given only the acoustic recording of a performance. In terms of data representations, this can be seen as transforming an audio signal into a MIDI[1] file. Signals of particular interest here are *polyphonic* music signals where several sounds are playing simultaneously.

Automatic recovery of the musical notation of an audio signal allows modifying, rearranging, and processing music at a high abstraction level and then resynthesizing it again. Structured audio coding is another important application: For example, a MIDI-like representation is extremely compact yet retains the characteristics of a piece of music to an important degree. Other uses of music transcription comprise information retrieval, musicological analysis of improvised and ethnic music, and interactive music systems which generate an accompaniment to the singing or playing of a soloist.

Attempts to automatically transcribe polyphonic music have been reported over a time-scale of about thirty years, starting from the work of Moorer in 1970s [50].

State-of-the-art transcription systems are still clearly behind human musicians in accuracy and flexibility, but considerable progress has been made during the last ten years and it is the purpose of this chapter to survey selected literature representing these advances. The main emphasis will be on signal processing methods for resolving mixtures of pitched musical instrument sounds. Methods wil be discussed for both estimating the fundamental frequencies (F0s) of concurrent sounds and for separating component sounds from a mixture signal. Other subtopics of music transcription, in particular beat tracking, percussion

*The authors are with Institute of Signal Processing, Tampere University of Technology, Korkeakoulunkatu 1, 33720 Tampere, Finland. E-mail: Anssi.Klapuri@tut.fi, Tuomas.Virtanen@tut.fi

[1] Musical instrument digital interface (MIDI) is a standard for exchanging performance data and parameters between electronic musical devices.

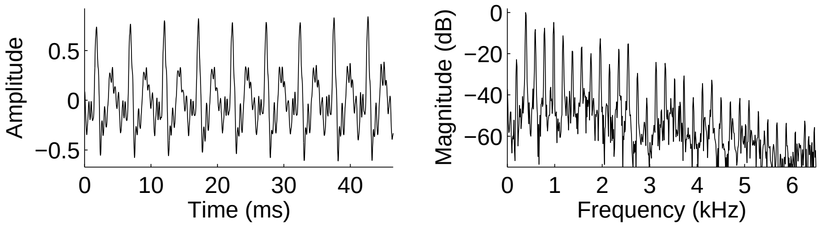

Figure 1: A harmonic sound in the time and frequency domains. The example represents a violin sound with fundamental frequency 196 Hz and fundamental period 5.1 ms.

transcription, musical instrument classification, and music structure analysis, will not be discussed here but an overview can be found in [37].

This chapter is organized as follows. Section 2 will introduce musical sounds and the basic principles of F0 estimation. Section 3 will discuss multiple-F0 estimators that follow principles of human hearing and pitch perception. Section 4 will describe sound separation methods that are based on the sinusoidal representation of music signals. Section 5 will discuss statistical inferences from parametric signal models. Section 6 describes unsupervised learning methods which make no assumptions about the nature of the sound sources and are thus suitable for percussion transcription, too. Concluding remarks are made in Section 7.

2 Musical sounds and F0 estimation

In the majority of Western music, melody and harmony are communicated by *harmonic* sounds. These are sounds that are nearly periodic in the time domain and show a regular spacing between the significant spectral components in the frequency domain (Fig. 1). An ideal harmonic sound consists of frequency components at integer multiples of its F0. In the case of plucked and struck string instruments, however, the partial frequencies are not in exact integral ratios but obey the formula $f_j = jF\sqrt{1+\beta(j^2-1)}$, where F is the fundamental frequency, j is the partial index, and β is an inharmonicity factor, due to the stiffness of real strings [20]. Despite this imperfection, the general structure of the spectrum is similar to that in Fig. 1.

Some of the methods discussed later in this chapter assume harmonic sounds as input. This limitation is not very severe in Western music where harmonic sounds are produced by most instrument families, namely by bowed and plucked string instruments, brass and reed instruments, flutes, pipe organs, and the human voice. Outside this category are mallet percussion instruments (e.g., vibraphone, xylophone, and marimba) whose waveforms are nearly periodic but whose spectra are definitely inharmonic.

A large number of different methods are available for monophonic F0 esti-

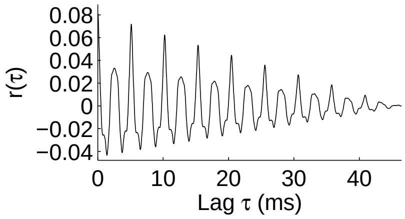

Figure 2: Autocorrelation function calculated within a 46 ms time frame for the violin waveform of Fig. 1.

mation. For speech signals, comparative evaluations of these can be found in [56, 26, 13]. Here, only the main ideas of different algorithms are introduced so as to provide a background for the analysis of polyphonic music signals described in detail in subsequent sections.

Algorithms that measure the periodicity of a time-domain signal have been among the most frequently used F0 estimators (see [64, 14]). As noted in [13], quite accurate F0 estimation can be achieved simply by an appropriate normalization of the autocorrelation function (ACF) $r(\tau)$. Then, the F0 can be computed as the inverse of the lag τ that corresponds to the maximum of $r(\tau)$ within a predefined lag range. The ACF of a signal $x(n)$ within a K-length analysis frame is given by

$$r(\tau) = \frac{1}{K} \sum_{k=0}^{K-1-\tau} x(k)x(k+\tau). \tag{1}$$

Figure 2 illustrates the ACF for the violin waveform shown in Fig. 1.

An indirect way of measuring time-domain periodicity is to match a harmonic pattern to the signal in the frequency domain. According to the Fourier theorem, a periodic signal with period τ can be represented with a series of sinusoidal components at frequencies j/τ, where j is a positive integer. As an example, Doval and Rodet [16] performed maximum-likelihood spectral pattern matching to find the F0 which best explained the observed frequency partials. Other strategies for spectral pattern matching have been proposed by Brown [6] and Maher and Beauchamp [43]. The time-domain ACF calculation is closely related to these methods since, according to the Wiener-Khintchine theorem [25, p.334], the ACF of a signal $x(n)$ is the inverse Fourier transform (IDFT) of its power spectrum $|X(k)|^2$. Writing out the IDFT we get

$$r(\tau) = \frac{1}{K} \sum_{k=0}^{K-1} \cos\left(\frac{2\pi\tau k}{K}\right) |X(k)|^2, \tag{2}$$

where K is the size of the analysis frame. It is easy to see that Eq. (2) emphasizes frequency components at harmonic spectral locations (integer multiples of $k = K/\tau$) because $r(\tau)$ is maximized when the cosine maxima line up with the frequencies of the harmonics (see Fig. 4).

Another class of F0 estimators measures the periodicity of the Fourier spectrum of a sound [40]. These methods are based on the observation that harmonic sounds have a quasi-periodic magnitude spectrum, the period of which is the F0. In its simplest form, the autocorrelation function $\tilde{r}(m)$ of a K-length magnitude spectrum is calculated as

$$\tilde{r}(m) = \frac{2}{K} \sum_{k=0}^{K/2-m-1} |X(k)||X(k+m)|. \tag{3}$$

The above formula bases F0 calculations on a fundamentally different type of information than the time-domain ACF: Here, any two frequency components with a certain spectral interval support the corresponding F0. An interesting difference between this method and the time-domain ACF is that measuring the periodicity of the time-domain signal is prone to F0 halving since the signal is periodic at twice the fundamental period too, whereas the methods that measure the periodicity of the magnitude spectrum are prone to F0 doubling since the spectrum is periodic at twice the F0 rate, too. There are ways to combine these complementary approaches to improve the overall result, as will be discussed in the next section.

3 Auditory model based multiple-F0 analysis

The human auditory system is very efficient at analyzing sound mixtures. It is therefore reasonable to learn from its function as much as possible, especially since the peripheral parts of hearing are relatively well known, and precise auditory models exist which are able to approximate the signal in the auditory nerve [49]. This enables the computation of a data representation similar to that used by the central auditory system.

3.1 Model of the auditory periphery

Computational models of the peripheral auditory system comprise two main parts which can be summarized as follows:

1) An acoustic input signal is passed through a bank of bandpass filters (aka *channels*) which represents the frequency selectivity of the inner ear. Typically, about 100 filters are used with center frequencies f_c uniformly distributed on a critical-band scale,

$$f_c = 229(10^{\xi/21.4} - 1), \tag{4}$$

   where ξ is the critical band number. Usually *gammatone* filters are used, with the bandwidths b_c of the filters obeying $b_c = 0.108 f_c + 24.7\text{Hz}$.

2) The signal at each band is processed to simulate the transform characteristics of *hair cells* that produce neural impulses to the auditory nerve [27]. In signal processing terms, this involves compression, half-wave rectification, and lowpass filtering.

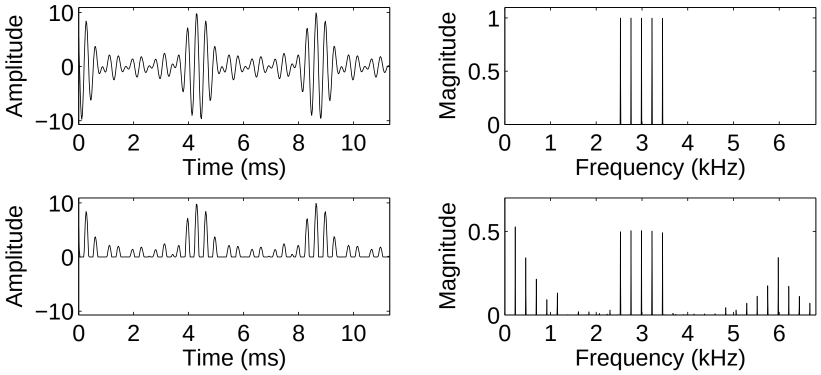

Figure 3: Upper panels show a signal consisting of the overtone partials 11–15 of a sound with F0 230 Hz in the time and the frequency domains. Lower panels illustrate the signal after half-wave rectification.

These seemingly very simple steps account for some important properties in pitch perception. In particular, the half-wave rectification operation in subbands allows a synthesis of the time-domain and frequency-domain periodicity analysis mechanisms discussed in Sect. 2. The half-wave rectification (HWR) is defined as

$$\mathrm{HWR}(x) = \max(x, 0) = \frac{1}{2}(|x| + x). \tag{5}$$

Fig. 3 illustrates the HWR operation for a subband signal which consists of five overtones of a harmonic sound. What is important is that the rectification generates spectral components that correspond to the frequency intervals between the input partials. The components below 1 kHz in the figure represent the amplitude envelope of the input signal. Any signal that consists of more than one frequency component exhibits periodic fluctuations, beating, in its time-domain amplitude envelope. That is, the partials alternatingly amplify and cancel out each other, depending on their phase. The rate of the beating caused by each pair of two frequency components depends on their frequency difference and, for a harmonic sound, the frequency interval corresponding to the F0 dominates.

After the rectification, periodicity analysis of the subband signals using, for instance, the ACF, would make use of both time and frequency domain periodicity by matching a harmonic pattern to both the input partials and the beating partials, which leads to more reliable F0 analysis [39]. Balance between the two types of information can be determined by applying a lowpass filter which partly suppresses the original passband at higher subbands but leaves the spectrum of the amplitude envelope intact.

3.2 Pitch perception models

The above-described two steps produce a simulation of the signal that travels in the different fibers ("channels") of the auditory nerve. Pitch perception models try to explain how this signal is further processed to derive a stationary percept of pitch. The processing mechanisms in the brain are not accurately known, but the prevailing view is that periodicity analysis of some form takes place for the signals within each auditory channel and the results are then combined across channels [11]. Meddis and Hewitt [48] implemented these two steps as follows:

3) ACF estimates $r_c(\tau)$ are computed within channels.

4) The ACFs are summed across channels to obtain a summary autocorrelation function $s(\tau) = \sum_c r_c(\tau)$. The maximum of $s(\tau)$ within a pre-defined lag range is used to predict the perceived pitch.

Together with the above-mentioned peripheral processing stages, this became known as the "unitary model" of pitch perception, and the authors showed that it can reproduce many important phenomena in pitch perception, such as missing fundamental, repetition pitch, and pitch shift of equally spaced inharmonic components [48].

Some music transcription systems have applied an auditory model to compute an intermediate data representation that is then used by a higher-level inference procedure. Martin [46] proposed a system for transcribing piano performances of four-voice Bach chorales. His system used the log-lag correlogram model of Ellis [17] (similar to the unitary model) as a front-end to an inference architecture where knowledge about sound production was integrated with rules governing tonal music. More recently, Marolt [45] used adaptive oscillators and neural networks to detect notes at the output of the peripheral hearing model described above.

3.3 Extensions of the unitary pitch model

The pitch perception model outlined above is not sufficient as such for accurate multiple-F0 estimation. It suffers from certain shortcomings, and the following described methods can be seen as attempts to alleviate these problems. Most importantly, the model accounts only for a single simultaneous pitch. Several pitches in a mixture signal cannot be detected simply by picking several local maxima in the summary ACF. De Cheveigné and Kawahara [12] addressed this problem by proposing a system where pitch estimation was followed by the cancellation of the detected sound in order to reveal the other sounds in the mixture. The cancellation was performed either by subband selection or by performing within-band cancellation filtering, and the estimation step was then iteratively repeated for the residual signal.

Tolonen and Karjalainen [65] addressed the computational complexity of the unitary model by first prewhitening the input signal and then dividing it into only two subbands, below and above 1 kHz. A "generalized ACF" was

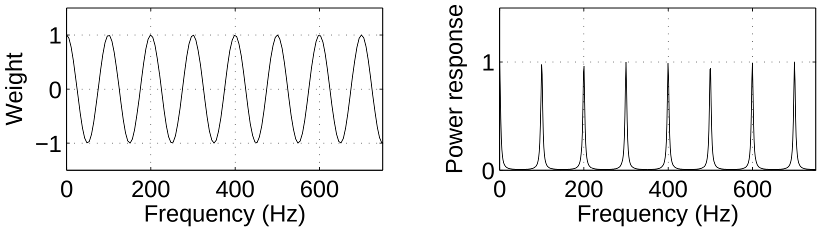

Figure 4: The left panel shows the weights $\cos(2\pi\tau k/K)$ of ACF calculation in (2), when the lag τ is 10 ms (corresponding to 100 Hz). The right panel shows the power response of a comb filter with a feedback lag of 10 ms.

then computed for the lower-channel signal and for the amplitude envelope of the higher-channel signal, and the two ACFs were summed. Despite the drastic reduction in computation compared to the original unitary model, many important characteristics of the model were preserved. Extension to multiple-F0 estimation was achieved by cancelling subharmonics in the summary ACF, by clipping the summary ACF to positive values, time-scaling it to twice its length, and by subtracting the result from the original clipped summary ACF. This cancellation operation was repeated for scaling factors up to about five. From the resulting enhanced summary autocorrelation function, all F0s were picked without iterative estimation and cancellation. The method is relatively easy to implement and produces good results when the component F0s are clearly below 1 kHz.

The generally weak robustness of the unitary model in polyphonic music was addressed by Klapuri [39]. He used a conventional peripheral hearing model (see Sect. 3.1) but replaced the ACF by a periodicity analysis mechanism where a bank of comb filters was simulated in the frequency domain. Figure 4 illustrates the power response of a comb filter for a time lag candidate τ and, for comparison, the corresponding weights $\cos(2\pi\tau k/K)$ of the ACF calculation in (2). Giving less emphasis to the spectrum between the harmonic partials of a F0 candidate alleviates the interference of concurrent sounds. Estimation of multiple F0s was achieved by cancelling each detected sound from the mixture and repeating the estimation for the residual, similarly to [12]. Quite accurate multiple-F0 estimation was achieved in the pitch range 60–2100 Hz.

4 Sound separation using sinusoidal modeling

Sound separation refers to the task of estimating the signal produced by an individual sound source from a mixture. Multiple F0 estimation and sound separation are closely related: an algorithm which achieves sound separation facilitates multiple F0 estimation, and vice versa. Even when the F0s of musical tones have already been found, additional information obtained using sound separation is useful for classifying the tones to sources. For example, similar

F0s and spectra in adjacent frames can be concatenated to form notes, and notes with different F0s can be grouped to obtain the whole passage of an instrument.

An efficient initial decomposition for the sounds produced by musical instruments is afforded by the sinusoids plus noise model, which represents the signal as a sum of deterministic and stochastic parts, or, as a sum of a set of sinusoids plus a noise residual [58], [4]. The model assumes that sinusoidal components are produced by vibrating systems which produce harmonically related frequencies, i.e., that their frequencies are integer multiples of one or more fundamental frequencies. The residual contains the energy produced by the excitation mechanisms and other components which are not a result of periodic vibration. The deterministic part of the model, which is called the sinusoidal model, has been used widely in audio signal processing, for example in speech coding by McAulay and Quatieri [47]. In musical signal processing it became known by the work of Smith and Serra [61].

The usefulness of the sinusoidal model in sound source separation and automatic music transcription stems from its physical plausibility. Since all sinusoids correspond to the vibrating modes of the sources in a mixture, it is plausible to map the sinusoids into the individual sound sources and to estimate higher-level information such as notes played by each source. Sound source separation algorithms which apply the sinusoidal model can be roughly divided into three categories: 1) methods which first estimate sinusoids and then group them into sound sources, 2) methods which first estimate F0s of the sources and then estimate sinusoids using partial frequencies predicted by the F0s, and 3) methods, which jointly estimate the number of sources, their F0s, and parameters of the sinusoids. The general signal model is discussed in Section 4.1, and the three separation approaches are discussed in Sections 4.2, 4.3, and Section 5, respectively.

4.1 Signal model

The parameters of natural sound sources are usually slowly-varying; therefore, the signal is analyzed in short segments or frames. In general, the parameters are assumed to be fixed during each frame, although continuous time-varying parameters are used in some systems. The sinusoidal model for one frame of a music signal can be written as

$$x(n) = \sum_{j=1}^{J} a_j \cos(2\pi f_j n + \theta_j) + e(n), \tag{6}$$

where the J sinusoids represent the harmonic partials of all sources, n is the time index, a_j, f_j and θ_j are the amplitude, frequency, and phase of the j^{th} sinusoid, respectively, and $e(n)$ is the residual.

Basic algorithms for the estimation of sinusoids from music signals have been reviewed by Rodet [57] and by Serra [58], and a theorethical framework for the parameter estimation is discussed by Kay [35]. Usually the estimation is done in the frequency domain. The frequencies of sinusoids can be estimated by picking the most prominent peaks from the magnitude spectrum. Many useful practical details for estimating the peaks are given in [3]. Also, matching pursuit algorithms have been used, which use a dictionary of time-domain elements to decompose the signal. For harmonic sounds, the use of harmonic atoms [24] provides a decomposition which is a good basis for the analysis of music signals.

4.2 Grouping sinusoids to sources

Psychoacoustically motivated methods have been among the most widely used in sound source separation. The cognitive ability of humans to perceive and recognize individual sound sources in mixture signals is called called *auditory scene analysis* [5]. Computational models of this function typically consist of two main stages where an incoming signal is first decomposed into its elementary time-frequency components and these are then organized to their respective sound sources.

Bregman pointed out a number of measurable acoustic "cues" which promote the grouping of time-frequency components to a common sound source as perceived by human listeners [5]. Among these are: proximity in time-frequency, harmonic frequency relationships, synchronous changes in the frequency or amplitude of the components, and spatial proximity (i.e., the same direction of arrival). In the computational modeling of auditory scene analysis, the most widely used cues are the proximity in time and harmonic frequency relationships; usually the signals are analyzed in short frames, and they are assumed to be harmonic or close to harmonic.

Kashino and Tanaka [34] implemented a subset of Bregman's cues for the purpose of sound source separation in music signals. Using the sinusoidal model as a decomposition, the authors viewed the source separation task as two grouping problems where sinusoids were first clustered to sound events which were then grouped to particular sound sources (instruments). Harmonic mistuning and onset asynchrony were used as the cues to initialize new sound events, and sinusoids were then grouped to these. Grouping the sound events to their respective sources was achieved by using timbre models and an "old-plus-new" heuristic. The latter means that a complex sound is interpreted as a combination of "old sounds" as much as possible, and the remainder is perceived as a "new sound" With this principle, automatic tone modeling was achieved without prestored templates. Evaluation results were shown for polyphonies up to three simultaneous sounds and for several different instruments. In a subsequent paper, the authors implemented the system in the framework of a Bayesian probability network and integrated musical knowledge to it [33].

The two-way mismatch procedure proposed by Maher and Beauchamp implements grouping based on harmonic frequency relationships [43]. Their method can be used to find the most likely F0s of harmonic sounds within one frame, given the frequency and amplitude estimates of prominent sinusoids. The fundamental frequencies are chosen so that the mismatch error between the estimated sinusoids and the partials predicted from trial F0 values is minimized. The name two-way mismatch stems from a procedure in which each estimated sinusoid is matched to the closest predicted partial, and each predicted partial is matched to the closest estimated sinusoid, and the total mismatch is measured as the average discrepancy in all the matches. The exact procedure and mismatch function is explained in [43]. The method requires a search for possible F0s within a designated range, and the number of sources has to be set manually. The algorithm is relatively straightforward to implement and has been used for the separation of duet signals [44] and to analyze the melody line in a polyphonic recording [53].

Sterian [62] implemented perceptual grouping rules as a set of likelihood functions, each of which evaluated the likelihood of the observed sinusoids given a hypothesized grouping of the sinusoids to note candidates. Distinct likelihood functions were defined to take into account onset and offset timing, harmonicity, low partial support, partial gap, and partial density (see the reference for the definitions of the latter concepts). The product of all the likelihood functions was used as a criterion for optimal grouping.

Godsmark and Brown [21] applied an auditory model to estimate dominant time-frequency components at different bands. These were mapped to sound sources by performing grouping according to onset and offset synchrony, temporal and frequency proximity, harmonicity, and common frequency movement.

4.3 Sound separation given the fundamental frequencies

When the number of concurrent sounds is high, the robustness of the grouping methods presented in the previous section decreases mainly because overlapping partials (partials whose frequencies are very close or the same) are difficult to resolve at the grouping stage. Estimation of the partials can be done more robustly if some higher-level information is available, such as the F0s of the sources, which can usually be estimated more accurately than individual sinusoids. This section deals with the estimation of sinusoids assuming that the F0s are estimated in advance using some other method.

Usually sinusoidal parameters are estimated by minimizing the reconstruction error between the sinusoidal model and the observed signal. Since the sinusoidal model is nonlinear with respect to frequencies, in general, a global solution for optimal parameters cannot be guaranteed. However, estimated F0s can be used to generate rough predictions of the partial frequencies, and iterative procedures which alternately estimate amplitudes and phases while frequencies are fixed and then update the frequencies, repeating until convergence is achieved, can be used. Most authors have used the energy of the residual to measure the reconstruction error, which leads to least-squares estimation. A general procedure, using a nonlinear least-squares method has been described in [63].

For fixed frequencies, amplitudes and phases can be estimated using the standard least-squares approach [36] which usually produces good results, even when the frequencies of the partials are slightly different. However, when the frequencies of two or more partials are equal, the exact amplitudes and phases of these partials cannot be resolved based on spectral peaks. This phenomenon is very common in musical signals, since F0s are often in harmonic relationships. However, an estimate of the amplitudes can be obtained by interpolating from adjacent frames or partials. The shape of a typical instrument spectrum is slowly-varying with respect to time and frequency, so that, in general, interpolation of amplitudes produces tolerable results. The phases are perceptually less important, so they can be easily interpolated to produce smooth transitions between frames.

Quatieri and Danisewicz [55] used adjacent frame interpolation for separation of speech voices. The method is not as useful for musical signals since F0s are often in harmonic relationship for relatively long durations. Maher [44] suggested using interpolation from adjacent partials. However, a problem arises if several adjacent partials overlap. The method proposed by Virtanen and Klapuri [70] performs interpolation implicitly, since it forces smooth amplitude spectra of sources by means of linear overtone series models.

Using the estimated amplitudes and phases, frequencies can be updated, for example, by using an algorithm proposed by Depalle and Hélie [15]. Each partial's frequency can be separately estimated, or frequency ratios can be fixed, so that only the fundamental frequency is updated, as done by Quatieri and Danisewicz [55] and by Virtanen and Klapuri [69].

If an F0 estimate fails, the signal $x(n)$ must contain components for which no sinusoids have been assigned in the model given by Eq. (6). The additional components may affect the estimate, since both the target and the interfering sounds are assumed to have a harmonic spectral structure. The effect of harmonic interference can be decreased, for example, by post-processing the estimated parameters using a perceptually motivated spectral smoothing method proposed by Klapuri [38].

5 Statistical inference within parametric signal models

By making certain assumptions about the component sounds in a music signal, the multiple-F0 estimation problem can be viewed as estimating the parameters of a chosen signal model. In this section, statistical methods to this end are discussed.

5.1 Frequency-domain models

Let us first consider a model for the short-term power spectrum of a signal. Here it is useful to apply a logarithmic frequency scale and to measure frequency in cent units. One cent is 1/100 of a semitone in equal temperament and there are 1200 cents in an octave. The relationship between cents and frequencies in Hertz is given by

$$z = 1200 \log_2(f_{\mathrm{Hz}}/440) + 6900, \tag{7}$$

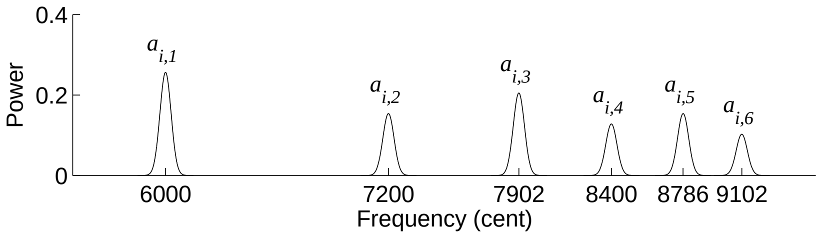

Figure 5: Illustration of the parametric model for spectral energy distribution in Eq. (8). Here ω_i is 6000 cents (260 Hz).

where the reference frequency 440 Hz is fixed to 6900 cents.

To enable statistical modeling, the power spectrum $\Psi(z)$ of a signal is considered to be a distribution of small units of energy along the frequency axis. In terms of an underlying random variable ζ, a realization $\zeta = z$ is thought to assign a unit of spectral energy to cents-frequency z. Consider the following parametric model for the spectral energy distribution proposed by Goto [22]:

$$p(\zeta = z|\theta) = \sum_{i=1}^{I} \alpha_i \sum_{j=1}^{J_i} a_{i,j} \mathrm{G}(z; \omega_i + 1200 \log_2(j), \sigma), \tag{8}$$

where $p(\zeta = z|\theta)$ represents the probability of observing energy at cents-frequency z given the model parameters θ. The function $\mathrm{G}(z; z_0, \sigma)$ is a Gaussian distribution with mean z_0 and standard deviation σ. The above formula assumes a mixture of I harmonic sounds where ω_i is the fundamental frequency of sound i in cents, α_i is the gain of sound i, and $a_{i,j}$ are the powers of the partials j of sound i. The shorthand notation θ represents all the parameters, $\theta = \{\omega_i, \alpha_i, a_{i,j}, I, J_i\}$.

Figure 5 illustrates the above model. Intuitively, each of the component sounds is modeled as a mixture of Gaussian distributions centered at integer multiples of F0. The variance around each partial models spreading of spectral energy due to time-domain windowing, variations in the F0, and the inharmonicity phenomenon mentioned in Sect. 2. A nice feature of the model is that, due to the logarithmic frequency scale applied, the actual variance is larger for the higher-order partials although σ is common for all.

Goto [23] extended the above model by introducing one additional dimension, so that multiple adaptive tone models were estimated for each F0. In practice, the weights became two-dimensional ($\alpha_{i,m}$) and the adaptive tone models three-dimensional ($a_{i,m,j}$), where m denotes the model index. This was done in order to allow various kinds of harmonic structures to appear at each F0 and to enable the use of several alternative tone model priors (spectral templates) side-by-side.

The problem to be solved, then, is to infer the model parameters θ given an observed short-time spectrum and possibly some prior information about the parameter distributions. Goto derived a computationally feasible expectation-maximization (EM) algorithm which iteratively updates the tone models and their weights, leading to maximum a posteriori parameter estimates. He avoided the explicit estimation of the F0 parameters in Eq. (8) by distributing a large number of F0s uniformly on the log-frequency scale and by estimating only the weights for each F0. The algorithm requires only a few iterations to converge and is suitable for real-time implementation. Goto used the method to detect the melody and the bass lines in real-world CD recordings. Temporal continuity of the estimates was considered by framewise tracking of the F0 weights within a multiple-agent architecture. Although the overall method in [23] is quite complex, the core EM algorithm is straightforward to implement and particularly suitable for detecting one (predominant) F0 in polyphonic signals.

Kameoka [32] applied the signal model (8) and performed maximum-likelihood estimation of the F0s ω_i, weights α_i, and tone model shapes $a_{i,j}$ using another EM algorithm. In addition, he used an information theoretical criterion to estimate the number of concurrent sounds I. Kameoka proposed a two-stage estimation procedure where F0 values were first constrained within a one-octave range and equal amplitudes were used for all harmonics, then the true F0s were found among the harmonics and subharmonics of each candidate and the spectral shape of the sounds was estimated. Promising results were reported for polyphonic music transcription and for pitch estimation of two simultaneous speakers.

5.2 Time-domain models

In time-domain models, the main difference compared to frequency-domain models is that the phases of partials have to be taken into account. Walmsley [71] performed parameter estimation in a Bayesian framework using a reformulated sinusoidal model for the time-domain signal:

$$y(t) = \sum_{i=1}^{I} \gamma_i \sum_{j=1}^{J_i} [a_{i,j} \cos(j\omega_i t) + b_{i,j} \sin(j\omega_i t)] + e(t), \tag{9}$$

where γ_i is a binary indicator variable switching the sound i on or off, ω_i is the F0 of sound i, and $a_{i,j}$, $b_{i,j}$ together encode the amplitude and phase of individual partials. The term $e(t)$ is a residual noise component. Similarly to Eq. (8), the above model assumes a mixture of harmonic signals, but a significant difference is that here the phases of the partials are also taken into account and estimated.

Another novelty in Walmsley's method was that the parameters were estimated jointly across a number of adjacent frames to increase robustness against transient events. A joint posterior distribution for all the parameters (given an observed signal segment) was defined which took into account the modeling error $e(t)$, prior distributions of the parameters, and the dependencies of the parameters on longer-term hyperparameters that modeled frequency variation over time. Optimal parameter estimates were produced by locating regions of high probability in the joint posterior distribution. Whereas this method proved intractable for analytical optimization, Walmsley was able to use Markov chain Monte Carlo (MCMC) methods to generate samples from the posterior. A transition kernel was proposed which consisted of heuristic rules for the fast exploration of the parameter space. For example, sounds were switched on and off, F0 values were switched between their harmonics and subharmonics, and the residual was analyzed to detect additional notes.

Davy and Godsill [10] extended Walmsley's model to accommodate time-varying amplitudes, non-ideal harmonicity, and non-white residual noise. They also improved the MCMC algorithm and reconsidered the prior structure. The resulting system was reported to work robustly for polyphonies up to three simultaneous sounds.

Another interesting time-domain model has been proposed by Cemgil [9] who placed emphasis on explicitly modeling the sound generation process. He modeled musical sounds as a sum of harmonically related and damped "noisy" sinusoids drawn from a stochastic process. The damping factors were tied to a global factor which was assumed to be trained in advance. The sound generators were controlled by a collection of binary indicator variables $\gamma_{i,t}$, a "piano roll", which represented the activity of different notes i as a function of time t. Here we use a matrix $[\mathbf{\Gamma}]_{i,t} = \gamma_{i,t}$ to denote the entire piano roll.

The transcription problem was then viewed as the task of finding a piano-roll $\mathbf{\Gamma}^*$ which maximizes the posterior probability $p(\mathbf{\Gamma}|\mathbf{y})$ given a time-domain signal $\mathbf{y}$. Since this optimization task was analytically intractable, Cemgil et al. developed inference procedures for certain special cases. Chords were identified by a greedy algorithm which started from an initial configuration and iteratively added or removed single notes until the probability of the mixture converged. Piano-roll inference was achieved by analyzing the signal in short time windows and by assuming that only one note may switch on or off within a window.

6 Unsupervised learning techniques

Recently, information theoretical methods such as independent component analysis (ICA) have been successfully used to solve blind source separation problems in several application areas [30]. ICA is based on the assumption of statistical independence of sources, and in certain conditions it enables the blind estimation of sources from mixed observations. The term blind means that there is no prior knowledge of the sources. Basic ICA cannot be directly used to separate one-channel time-domain signals, but this becomes possible with a suitable signal representation, as described in Section 6.1. In addition to ICA, other unsupervised learning algorithms such as non-negative matrix factorization (NMF) [41] and sparse coding have been used in the analysis and separation of monaural (single-channel) music signals. Unlike ICA, NMF and sparse coding do not aim at statistical independence but just at representing the observed data efficiently. In the case of music signals, this usually results in the separation of sources, at least to some degree. The basic principles of ICA, NMF, and sparse coding are described in Sections 6.2, 6.3, and 6.4, respectively.

6.1 Signal model for monaural signals

Basic ICA and many other blind source separation algorithms require that the number of sensors is greater than or equal to the number of sources. In multichannel sound separation this means that there should be at least as many microphones as there are sources. However, automatic music transcription usually aims at finding notes in monaural or stereo signals, for which basic ICA methods may not be adequate. The most common approach to overcome this limitation is to represent the input signal in the time-frequency domain, for example by using the short-time Fourier transform (STFT). Basic generative models of ICA, NMF, and sparse coding are linear: For each analysis frame m, the short-time spectrum vector $\mathbf{x}_m$ can be expressed as a weighted sum of basis spectra $\mathbf{b}_j$. The model is not necessarily noise-free. Thus, with a residual term $\mathbf{e}_m$ it can be written as

$$\mathbf{x}_m = \sum_{j=1}^{J} g_{j,m}\mathbf{b}_j + \mathbf{e}_m, \tag{10}$$

where J is the number of basis spectra and $g_{j,m}$ is the gain of the j^{th} basis spectrum for the m^{th} frame. For M frames the model can be written in a matrix form as

$$\mathbf{X} = \mathbf{B}\mathbf{G} + \mathbf{E}, \tag{11}$$

where $\mathbf{X} = \left[\mathbf{x}_1, \ldots \mathbf{x}_M\right]$ is the spectrogram matrix, $\mathbf{B} = \left[\mathbf{b}_1, \ldots, \mathbf{b}_J\right]$ is the basis matrix, $[\mathbf{G}]_{j,m}$ is the gain matrix, and $\mathbf{E} = \left[\mathbf{e}_1, \ldots, \mathbf{e}_M\right]$ is the residual matrix. Usually $\mathbf{X}$ is given as the input and $\mathbf{B}$, $\mathbf{G}$, and $\mathbf{E}$ are unknown. Each $\mathbf{b}_j$ corresponds to a short-time spectrum, which has a time-varying gain $g_{j,m}$. The term *component* is used to refer to one basis spectrum and its time-varying gain. Each sound source is usually modeled as a sum of one or more components. Multiple components per source are used since each pitch value of a harmonic source corresponds to a different spectrum. Also, even one note of an instrument may not have a stationary spectrum over time, so that multiple components may be needed. However, the model is flexible in the sense that it is suitable for representing both harmonic and percussive sounds. A simple two-note example is illustrated in Figure 6 where the components were estimated using the non-negative matrix factorization algorithm described in [60] and [41].

The phase spectra of natural sound sources are very unpredictable. Therefore, the phases are often discarded and the estimation is done using the magnitude or power spectra. Linear superposition of time-domain signals does not imply linear superposition of the magnitude or power spectra. However, a linear superposition of incoherent time domain signals can be approximated as a linear superposition of short-time power spectra.

In most systems, a discrete Fourier transform of fixed window size is used, but, in general, estimation algorithms allow the use of any time-frequency representation. For example, short-time signal processing has been used without an explicit frequency transform by Abdallah [1] and Jang and Lee [31]. It

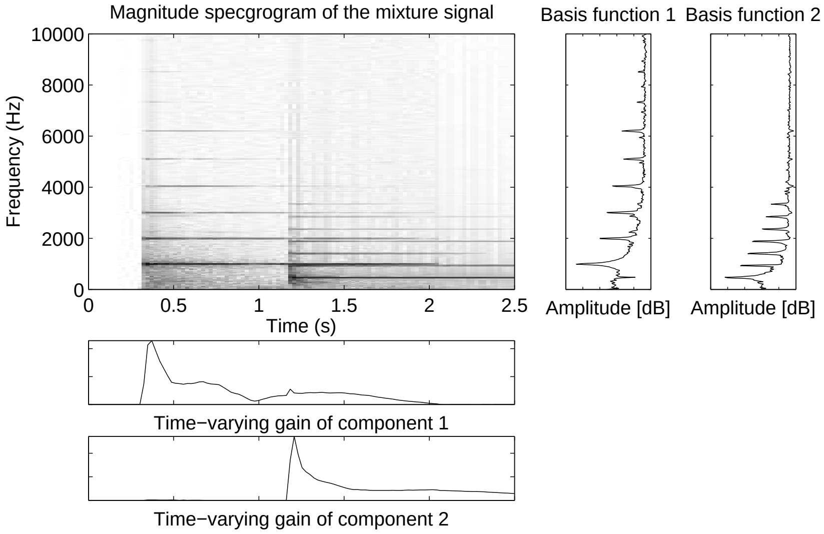

Figure 6: A mixture signal consisting of two piano notes (B5 and A#4, setting on at times $t = 0.3$ and $t = 1.2$, respectively) and two separarated components. Both components have a fixed spectrum with a time-varying gain.

turns out that their systems learned basis functions from time-domain music and speech signals which were very similar to those used in wavelet and STFT analysis.

There are several alternative criteria for the estimation of the unknown parameters, the basic principles of which are discussed in the next three sections. In many cases the number of basis functions, J, is unknown, and principle component analysis (PCA) can be used to estimate it. However, in the case of monaural sound source separation, the problem of estimating the number of sources J has not received much attention yet.

Once the short-time spectrum of the input signal is separated into components, the components are further analyzed to obtain musically important information. This includes, for example, the onset and offset times and F0s of each component. Ideally, a component is active when its gain $g_{j,m}$ is non-zero, but in practice, activity detection has to be made using a threshold which is larger than zero. There are some alternative possibilities for the estimation of the F0. For example, prominant peaks can be located from the spectrum and the two-way mismatch procedure [43] can be used, or the fundamental period can be estimated from an autocorrelation estimate. Sometimes a component may represent more than one pitch. This happens especially when notes occur consistently simultaneously, as in the case of chords.

6.2 Independent subspace analysis

In monaural sound source separation, the term independent subspace analysis (ISA) has been used to denote the ICA of the spectrogram of a time-domain signal [8]. In ISA, either the time-varying gains or the spectra of the components are assumed to be statistically independent. There are several criteria and algorithms for obtaining the statistical independency, but they all have a common mathematical background [42]. The criteria include, for example, nongaussianity and negentropy, which are usually measured using high-order cumulants such as kurtosis [30]. As a preprocessing for ICA, the observation matrix is usually whitened and the dimensions are reduced by PCA. The core of ICA is the estimation of an unmixing matrix. Independent components are obtained by multiplying the whitened observation matrix by the estimate of the unmixing matrix. There are several ICA algorithms, some of which are freely available, for example FastICA [29] and JADE [7].

ISA has been used in several automatic music transcription and sound source separation systems, for example by Casey and Westner in general audio source separation [8] and by FitzGerald in percussion transcription [18].

6.3 Non-negative matrix factorization

If short-time power spectra are used as observations, each component has a fixed power spectrum with a time-varying gain. It is reasonable to restrict these to be entry-wise non-negative, so that the components are purely additive and the spectra do not have negative values. A problem with ISA is that the standard ICA algorithms do not allow non-negativity restrictions, and in practice the algorithms produce also negative values. This problem has been addressed e.g. by Plumbley and Oja [54], whose non-negative PCA algorithm solves the problem with some limitations.

Non-negative matrix factorization (NMF) has been successfully used in unsupervised learning with the non-negativity restrictions [41]. When $\mathbf{X} \approx \mathbf{BG}$, the non-negativity of $\mathbf{B}$ and $\mathbf{G}$ alone seems to be a sufficient condition for the blind estimation of sources in many cases. In the analysis of music signals NMF has been used, for example, by Smaragdis and Brown [60].

Lee and Seung proposed two cost functions and estimation algorithms for obtaining $\mathbf{X} \approx \mathbf{BG}$ [41]. The cost functions are the square of the Euclidean distance, or, the Frobenius norm of the error between the observation matrix $\mathbf{X}$ and the model $\mathbf{BG}$, given by

$$||\mathbf{X} - \mathbf{BG}||_F^2 = \sum_{k,m} ([\mathbf{X}]_{k,m} - [\mathbf{BG}]_{k,m})^2 \tag{12}$$

and divergence D, defined as

$$D(\mathbf{X}||\mathbf{BG}) = \sum_{k,m} d([\mathbf{X}]_{k,m}, [\mathbf{BG}]_{k,m}) \tag{13}$$

where the function d is given by

$$d(p,q) = p\log(\frac{p}{q}) - p + q. \tag{14}$$

Both cost functions are lower-bounded by zero, which is obtained only when $\mathbf{X} = \mathbf{BG}$. The estimation algorithms presented in [41] initialize $\mathbf{B}$ and $\mathbf{G}$ with random values, and then update them iteratively, so that the value of the cost funtion is non-increasing at each update.

Since the NMF aims only at representing the observed spectrogram with non-negative components, it does not guarantee the separation of sources. Especially when the sources are always present simultaneously, the algorithm tends to represent then with a single component. However, it has turned out that the factorization of a magnitude spectrogram by minimizing the divergence of Eq. (13) produces better separation results than ISA in most cases.

6.4 Sparse coding

A technique called sparse coding has been successfully used to model the early stages of vision [51]. The term sparse refers to a signal model in which the data are represented in terms of a small number of active elements chosen out of a larger set. The sparseness restriction is usually placed for the gains $[\mathbf{G}]_{j,m}$ in Eq. (11). Sparseness of $\mathbf{G}$ means that the probability of an element of $\mathbf{G}$ being zero is high, so that only a few components are active at a time, and each component is active only in a small number of frames. For musical sources this is usually a valid assumption, since each component typically corresponds to a single pitch value or to a percussive source.

A sparse representation is obtained by minimizing a cost function which is the sum of a reconstruction error term and a term which incurs a penalty on the non-zero elements of $\mathbf{G}$. An example of such a cost function c is given by

$$c(\mathbf{B},\mathbf{G}) = ||\mathbf{X} - \mathbf{BG}||_F^2 + \lambda \sum_{j,m} f([\mathbf{G}]_{j,m}). \tag{15}$$

The function f is used to penalize non-zero entries of $\mathbf{G}$ and the scalar $\lambda \geq 0$ is used to balance the reconstruction error cost and the sparseness cost. For example, Olshausen and Field [51] used $f(x) = \log(1+x^2)$, and Hoyer [28] used $f(x) = |x|$. This approach requires that the scale of either $\mathbf{B}$ or $\mathbf{G}$ is fixed, for example to unity variance.

As in NMF, the parameters are usually solved using iterative algorithms. Hoyer [28] proposed a non-negative sparse coding algorithm by combining NMF and sparse coding. For musical signal analysis, sparse coding has been used for example by Abdallah and Plumbley [1, 2] to analyze pitched sounds and by Virtanen [67] to transcribe drums from synthesized MIDI signals.

It is not clear whether the explicit assumption of sparseness ($\lambda > 0$) really increases the quality of the separation. However, with non-negativity and sparseness constraints one has to use a projected gradient descent optimization algorithm which is inefficient compared to the multiplicative update rules used in NMF.

6.5 Discussion

Manual music transcription requires a lot of prior knowledge and training. It is not known whether automatic transcription of music is possible without prior information of the sources, for example the knowledge that they are harmonic. In some simple cases it is possible to estimate the components without prior information, but this may not always be the case. Some attempts to utilize prior information have been made. Usually they are based on supervised learning since it is difficult to constrain harmonic basis functions in the model (10). Vincent and Rodet proposed a polyphonic transcription system based on ISA and hidden Markov models which were trained using monophonic material [66]. FitzGerald [18] used spectral templates of instruments as an initialization of his prior subspace analysis based drum transcription system. The time-varying gains obtained for the components were further processed using ICA. All these systems use a limited instrument set. In general, all the possible instruments cannot be trained in advance but some kind of model adaptation is needed.

The linear model (10) is not well suited for separating singing voice signals, since different phonemes have different spectra. Also, other instruments with a strongly varying spectral shape are problematic. Researchers have tried to overcome the limitations of the model by using a more complex model which includes a two-dimensional time-frequency basis function for each component instead of a static spectrum. Initial experiments with this kind of approach have been presented by Smaragdis [59] and Virtanen [68]. The model can also be used to represent time-varying fundamental frequencies [19].

The dynamic range of music signals is wide, and low-intensity observations may be perceptually important. The power spectrum domain is problematic in the sense that it causes separation algorithms to concentrate on high-intensity observations, thus failing to separate low-energy sources. This has been addressed by Vincent and Rodet [66] who used a specific algorithm in the log-power spectral domain and by Virtanen [68] who used perceptually motivated weights to mimic human loudness perception.

Automatic music transcription and sound source separation using unsupervised learning techniques is currently an active research topic, and none of the proposed methods is clearly better than others. The existing algorithms are most successful in cases where instrument sets and polyphony are limited. For example, FitzGerald [18] reported good results for an algorithm extended from ISA, and Paulus and Virtanen [52] successfully used NMF for the transcription of drum patterns consisting of bass, snare and hi-hat drums.

7 Summary and conclusions

As the preceding sections have shown, music transcription can be performed using very different kinds of methods and assumptions. The underlying assumptions made for unsupervised learning methods are completely different than those made for signal-model-based statistical inference methods, yet both approaches can yield meaningful results. At the present time, none of the described main approaches stands out as clearly the most promising. Instead, one of the decisive factors determining the popularity of different methods is their conceptual simplicity vs. performance. A problem with many transcription systems is that they are very complex entities. However, there are methods in all the main categories that are quite straightforward to implement and lead to good analysis results. Among these are, for example, the auditorily-oriented method of Tolonen and Karjalainen [65], the sinusoidal-modeling method of Maher and Beauchamp [43], the expectation-maximization algorithm of Goto [23] for detecting the most predominant F0 in polyphonic signals, and the non-negative matrix factorization algorithm of Smaragdis and Brown [60].

Research is being carried out to combine the advantages of different approaches, and in many cases this has produced the most successful results. Also, it should be remembered that this chapter has focused primarily on acoustic signal analysis methods without addressing the use of musicological information or the larger-scale structure of music signals.

References

[1] S. A. Abdallah. *Towards Music Perception by Redundancy Reduction and Unsupervised Learning in Probabilistic Models.* PhD thesis, Department of Electronic Engineering, King's College London, 2002.

[2] S. A. Abdallah and M. D. Plumbley. An independent component analysis approach to automati c music transcription. In *Audio Engineering Society 114th Convention*, Amsterdam, Netherlands, March 2003.

[3] M. Abe and J. O. Smith. Design criteria for simple sinusoidal parameter estimation based on quadratic interpolation of FFT magnitude peaks. In *Audio Engineering Society 117th Convention*, San Francisco, USA, 2004.

[4] X. Amatriain, J. Bonada, A. Loscos, and Xavier Serra. Spectral processing. In U. Zölzer, editor, *DAFX - Digital Audio Effects.* John Wiley & Sons, 2002.

[5] A.S. Bregman. *Auditory scene analysis.* MIT Press, Cambridge, USA, 1990.

[6] J. C. Brown. Musical fundamental frequency tracking using a pattern recognition method. *Journal of the Acoustical Society of America*, 92(3):1394–1402, 1992.

[7] J.-F. Cardoso. High-order contrasts for independent component analysis. *Neural Computation*, 11(1), 1999.

[8] M. A. Casey and A. Westner. Separation of mixed audio sources by independent subspace analysis. In *International Computer Music Conference*, Berlin, Germany, 2000.

[9] A.T. Cemgil, B. Kappen, and D. Barber. A generative model for music transcription. *IEEE Transactions on Speech and Audio Processing*, 13(6), 2005.

[10] M. Davy and S. Godsill. Bayesian harmonic models for musical signal analysis. In *Seventh Valencia International meeting Bayesian statistics 7*, Tenerife, Spain, June 2002.

[11] A. de Cheveigné. Pitch perception models. In C. J. Plack, A. J. Oxenham, R. R. Fay, and Popper A. N., editors, *Pitch*. Springer, New York, 2005.

[12] A. de Cheveigné and H. Kawahara. Multiple period estimation and pitch perception model. *Speech Communication*, 27:175–185, 1999.

[13] A. de Cheveigné and H. Kawahara. Comparative evaluation of F0 estimation algorithms. In *7th European Conf. Speech Communication and Technology*, Aalborg, Denmark, 2001.

[14] A. de Cheveigné and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

[15] Ph. Depalle and T. Hélie. Extraction of spectral peak parameters using a short-time fourier transform modeling and no sidelobe windows. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Palz, USA, 1997.

[16] B. Doval and X. Rodet. Estimation of fundamental frequency of musical sound signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3657–3660, Toronto, Canada, 1991.

[17] D. P. W. Ellis. *Prediction-driven computational auditory scene analysis*. PhD thesis, Massachusetts Institute of Technology, 1996.

[18] Derry FitzGerald. *Automatic Drum Transcription and Source Separation*. PhD thesis, Dublin Institute of Technology, 2004.

[19] Derry FitzGerald, Matt Cranitch, and Eugene Coyle. Generalised prior subspace analysis for polyphonic pitch transcription. In *International Conference on Digital Audio Effects*, Madrid, Spain, 2005.

[20] N.H. Fletcher and T.D. Rossing. *The Physics of Musical Instruments*. Springer, Berlin, Germany, 2nd edition, 1998.

[21] D. Godsmark and G. J. Brown. A blackboard architecture for computational auditory scene analysis. *Speech Communication*, 27(3):351–366, 1999.

[22] M. Goto. A robust predominant-F0 estimation method for real-time detection of melody and bass linjes in cd recordings. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, June 2000.

[23] Masataka Goto. A real-time music scene description system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. *Speech Communication*, 43(4):311–329, 2004.

[24] R. Gribonval and E. Bacry. Harmonic decomposition of audio signals with matching pursuit. *IEEE Transactions on Signal Processing*, 51(1):101–111, 2003.

[25] W. M. Hartmann. *Signals, sound, and sensation.* Springer, New York, 1998.

[26] W. J. Hess. Pitch and voicing determination. In S. Furui and M. M. Sondhi, editors, *Advances in speech signal processing*, pages 3–48. Marcel Dekker, New York, 1991.

[27] M. J. Hewitt and R. Meddis. An evaluation of eight computer models of mammalian inner hair-cell function. *Journal of the Acoustical Society of America*, 90(2):904–917, 1991.

[28] P. Hoyer. Non-negative sparse coding. In *IEEE Workshop on Networks for Signal Processing XII*, Martigny, Switzerland, 2002.

[29] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

[30] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* John Wiley & Sons, 2001.

[31] G.-J. Jang and T.-W. Lee. A maximum likelihood approach to single channel source separation. *Journal of Machine Learning Research*, 23:1365 – 1392, 2003.

[32] H. Kameoka, T. Nishimoto, and S. Sagayama. Separation of harmonic structures based on tied gaussian mixture model and information criterion for concurrent sounds. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, 2004.

[33] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka. Organisation of hierarchical perceptual sounds: Music scene analysis with autonomous processing modules and a quantitative information integration mechanism. In *International Joint Conference on Artificial Intelligence*, pages 158–164, Montreal, Quebec, 1995.

[34] K. Kashino and H. Tanaka. A sound source separation system with the ability of automatic tone modeling. In *International Computer Music Conference*, pages 248–255, Hong Kong, China, 1993.

[35] M. Kay. *Modern Spectral Estimation.* Prentice Hall, 1988.

[36] Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory.* Prentice Hall, 1993.

[37] A. Klapuri and M. Davy, editors. *Signal Processing Methods for Music Transcription.* Springer, New York, 2006.

[38] A. P. Klapuri. Multiple fundamental frequency estimation based on harmonicity and spectral smoothness. *IEEE Transactions on Speech and Audio Processing*, 11(6):804–815, 2003.

[39] A. P. Klapuri. A perceptually motivated multiple-F0 estimation method for polyphonic music signals. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Palz, USA, 2005.

[40] M. Lahat, R. Niederjohn, and D.A. Krubsack. Spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 6:741–750, June 1987.

[41] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Neural Information Processing Systems*, pages 556–562, Denver, USA, 2001.

[42] T.-W. Lee, M. Girolami, A. J. Bell, and T. J. Sejnowski. A unifying information-theoretic framework for independent component analysis. *Computers and Mathematics with Applications*, 31(11), 2000.

[43] R.C. Maher and J.W. Beauchamp. Fundamental frequency estimation of musical signals using a two-way mismatch procedure. *Journal of the Acoustical Society of America*, 95(4):2254–2263, 1994.

[44] Robert C. Maher. Evaluation of a method for separating digitized duet signals. *Journal of the Acoustical Society of America*, 38(12), 1990.

[45] M. Marolt. SONIC: transcription of polyphonic piano music with neural mnetworks. In *MOSART Workshop on Current Research Directions in Computer Music*, Barcelona, Spain, November 2001.

[46] K. D. Martin. Automatic transcription of simple polyphonic music: robust front end processing. Technical Report 399, MIT Media Laboratory Perceptual Computing Section, 1996.

[47] R. J. McAulay and T. F. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Speech and Audio Processing*, 34(4), 1986.

[48] R. Meddis and M. J. Hewitt. Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification. *Journal of the Acoustical Society of America*, 89(6):2866–2882, 1991.

[49] B. C. J. Moore, editor. *Hearing—Handbook of Perception and Cognition.* Academic Press, San Diego, California, 2nd edition, 1995.

[50] J. A. Moorer. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer.* PhD thesis, Dept. of Music, Stanford University, 1975. Distributed as Dept. of Music report No. STAN-M-3.

[51] B. A. Olshausen and D. F. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.

[52] Jouni Paulus and Tuomas Virtanen. Drum transcription with non-negative spectrogram factorisation. In *European Signal Processing Conference*, Turkey, 2005.

[53] G. Peterschmitt, E. Gómez, and P. Herrera. Pitch-based solo location. In *Proceedings of MOSART Workshop on Current Research Directions in Computer Music*, Barcelona, Spain, 2001.

[54] M. D. Plumbley and E. Oja. A 'non-negative PCA' algorithm for independent component analysis. *IEEE Transactions on Neural Networks*, 15(1):66–67, 2004.

[55] Thomas F. Quatieri and Ronald G. Danisewicz. An approach to co-channel talker interference suppression using a sinusoidal model for speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(1), 1990.

[56] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal. A comparative performance study of several pitch detection algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(5):399–418, 1976.

[57] X. Rodet. Musical sound signal analysis/synthesis: Sinusoidal+residual and elementary waveform models. In *IEEE Time-Frequency and Time-Scale Workshop*, IEEE Time-Frequency and Time-Scale Workshop, 1997.

[58] X. Serra. Musical sound modeling with sinusoids plus noise. In C. Roads, S. Pope, A. Picialli, and G. De Poli, editors, *Musical Signal Processing.* Swets & Zeitlinger, 1997.

[59] Paris Smaragdis. Discovering auditory objects through non-negativity constraints. In *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing*, Jeju, Korea, 2004.

[60] Paris Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Palz, USA, 2003.

[61] J.O. Smith and X. Serra. Parshl: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation. In *International Computer Music Conference*, Urbana, USA, 1987.

[62] A. D. Sterian. *Model-based segmentation of time-frequency images for musical transcription*. PhD thesis, University of Michigan, 1999.

[63] Petre Stoica and Randolph L. Moses. *Introduction to Spectral Analysis*. Prentice Hall, 1997.

[64] D. Talkin. A robust algorithm for pitch tracking. In W. B. Kleijn and K. K. Paliwal, editors, *Speech Coding and Synthesis*, pages 495–517. Elsevier Academic Press, Amsterdam, 1995.

[65] T. Tolonen and M. Karjalainen. A computationally efficient multipitch analysis model. *IEEE Transactions on Speech and Audio Processing*, 8(6):708–716, 2000.

[66] E. Vincent and X. Rodet. Music transcription with ISA and HMM. In *the 5th International Symposium on Independent Component Analysis and Blind Signal Separation*, 2004.

[67] Tuomas Virtanen. Sound source separation using sparse coding with temporal continuity objective. In *International Computer Music Conference*, Singapore, 2003.

[68] Tuomas Virtanen. Separation of sound sources by convolutive sparse coding. In *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing*, Jeju, Korea, 2004.

[69] Tuomas Virtanen and Anssi Klapuri. Separation of harmonic sounds using multipitch analysis and iterative parameter estimation. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Palz, USA, 2001.

[70] Tuomas Virtanen and Anssi Klapuri. Separation of harmonic sounds using linear models for the overtone series. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Orlando, USA, 2002.

[71] P.J. Walmsley. *Signal Separation of Musical Instruments. Simulation-based methods for musical signal decomposition and transcription*. PhD thesis, Department of Engineering, University of Cambridge, September 2000.