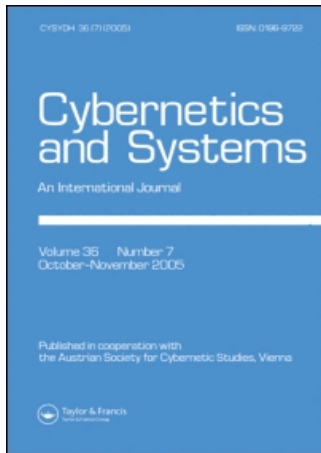


This article was downloaded by:[EPSCoR Science Information Group (ESIG) Dekker Titles only Consortium]
On: 3 August 2007
Access Details: [subscription number 777703943]
Publisher: Taylor & Francis
Informa Ltd Registered in England and Wales Registered Number: 1072954
Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



Cybernetics and Systems An International Journal

Publication details, including instructions for authors and subscription information:
<http://www.informaworld.com/smpp/title~content=t713722751>

Automatic Music Transcription and Audio Source Separation

M. D. Plumbley; S. A. Abdallah; J. P. Bello; M. E. Davies; G. Monti; M. B. Sandler

Online Publication Date: 01 September 2002

To cite this Article: Plumbley, M. D., Abdallah, S. A., Bello, J. P., Davies, M. E., Monti, G. and Sandler, M. B. (2002) 'Automatic Music Transcription and Audio Source Separation', Cybernetics and Systems, 33:6, 603 - 627

To link to this article: DOI: 10.1080/01969720290040777

URL: <http://dx.doi.org/10.1080/01969720290040777>

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: <http://www.informaworld.com/terms-and-conditions-of-access.pdf>

This article maybe used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

© Taylor and Francis 2007



AUTOMATIC MUSIC TRANSCRIPTION AND AUDIO SOURCE SEPARATION

**M. D. PLUMBLEY
S. A. ABDALLAH
J. P. BELLO
M. E. DAVIES
G. MONTI
M. B. SANDLER**

Department of Electronic Engineering, Queen Mary,
University of London, London, United Kingdom

In this article, we give an overview of a range of approaches to the analysis and separation of musical audio. In particular, we consider the problems of automatic music transcription and audio source separation, which are of particular interest to our group. Monophonic music transcription, where a single note is present at one time, can be tackled using an autocorrelation-based method. For polyphonic music transcription, with several notes at any time, other approaches can be used, such as a blackboard model or a multiple-cause/sparse coding method. The latter is based on ideas and methods related to independent component analysis (ICA), a method for sound source separation.

Over the last decade or so, and particularly since the publication of Bregman's seminal book *Auditory Scene Analysis* (Bregman 1990), there has been an increasing interest in the problem of computational auditory

S. Abdallah is supported by an EPSRC Postgraduate Studentship. J. P. Bello and G. Monti are supported by the OMRAS project, jointly funded by JISC (UK) and NSF (USA). The music sequences used to generate Figure 6(a) and (c) are used by permission from the Classical Piano Midi Page <http://www.piano-midi.de>, Copyright by Bernd Krueger.

Address correspondence to Dr. M. D. Plumbley, Department of Electronic Engineering, Queen Mary, University of London, Mile End Road, London E1 4NS, United Kingdom. E-mail: mark.plumbley@elec.qmul.ac.uk

scene analysis (CASA): how to design computer-based models that can analyze an auditory scene. Imagine you are standing on a busy street among a crowd of people. You can hear traffic noise, footsteps of people nearby, the bleeping of a pedestrian crossing, your mobile phone ringing, and colleagues behind you having a conversation. Despite all these different sound sources, you have a pretty good idea of what is going on around you. It is more than just a mess of overlapping noise, and if you try hard you can concentrate on one of these sources if it is important to you (such as the conversation behind you).

This has proved to be a very difficult problem. It requires both separation of many sound sources and analysis of the content of these sources. However, a few authors have begun to tackle this problem in recent years, with some success (see, e.g., Ellis 1996).

One particular aspect of auditory scene analysis of interest to our group is *automatic music transcription*. Here, the sound sources are one or more instruments playing a piece of music, and we wish to analyze this to identify the instruments that are playing, and when and for how long each note is played. From this analysis we should then be able to produce a written musical score that shows notes and the duration of each on a written conventional music notation (for conventional western music, at least). In principle, this musical score could then be used to recreate the musical piece that was played.

We are also interested in the ability to separate sound sources based on their different locations in an auditory scene. This capability, known as *blind source separation* (Bell and Sejnowski 1995), can be useful on its own if we wish to eliminate some sound sources and concentrate on others. It also offers the potential to be combined with automatic music transcription systems in the future to improve transcription performance based on the location of instruments as well as the different sounds they make.

AUTOMATIC MUSIC TRANSCRIPTION

As we mentioned earlier, the aim of automatic music transcription is to analyze an audio signal to discover the instruments and notes that are playing, and so be able to produce a written transcription of the piece of music. Some pieces of music are *monophonic*, having only one instrument playing only one note at a time, such as a trumpet solo. On the

other hand, most pieces of music are *polyphonic*, having one or more instruments playing several notes at once, such as a piece for piano or full orchestra. (The use of the term *monophonic* here should be distinguished from a *mono*—as opposed to *stereo*—recording, which may be polyphonic music, but has been recorded with just a single audio channel.)

While the general auditory scene analysis is something we would expect most human listeners to have reasonable success at, this is not the case for the automatic music transcription problem. Most people, even musicians, do not have the ability to name the pitch of a note heard in isolation (so-called *perfect pitch*). Furthermore, only after special training are people able to perform musical transcription, and then they typically need to listen to the piece repeatedly, writing out one musical line (voice) at a time. Even the best human music transcribers are typically limited to pieces with less than five notes at once. For general listeners, the identity of individual notes does not seem to be important: rather it is the overall musical quality conveyed by the combinations of notes (chords). The composer may also deliberately be using the features and limitations of human hearing to create special effects, such as a single instrument alternating between high and low notes to give the perception of two separate musical lines (streams).

Thus the automatic music transcription problem is somewhat artificial: it is not something that we would expect an average human to be able to do easily. Nevertheless, there is much that we can learn from the human auditory system, and some of the approaches that have been used are based on, or inspired by, either what is known about its operation, or what is known about its ability to learn. We shall draw a distinction between two types of approach in the descriptions that follow: *knowledge-based* models and *learning* models.

In the knowledge-based approach, we try to use information we have about the physics of the musical generation process and about the human auditory system in order to build our analysis model. For monophonic music transcription, for example, we can use our knowledge of the physics of vibrating objects such as the repetition of near-identical sound pressure waveforms in the auditory signals. For polyphonic music transcription, we might use time-frequency analysis techniques inspired by the cochlea in the human ear, followed by a knowledge integration system allowing us to piece together partial evidence into our proposed transcription.

In the learning approach, we may still use knowledge about the music generation process (sometimes called prior knowledge), but we also have adjustable parameters in our model which we allow to adapt (or learn) on the basis of the data that we are given. This approach is therefore sometimes called a data-driven approach, since we can be considered to be extracting knowledge from the audio signal (data) that is input into the model. For polyphonic music transcription, we might use this approach to learn the “shapes” (frequency content) of the various notes and instruments, at the same time as the “amounts” (volumes) of each note at each time.

Monophonic Transcription

While monophonic (single-note) music transcription, sometimes called *pitch tracking*, might seem like just a special case of polyphonic (many-note) music transcription, it is an important case that is worth treating separately. From the practical point of view, the knowledge that only one note is being played at any one time can help make the transcription process both simpler and more robust. In addition, there are many potential applications for monophonic transcription, from recognizing a hummed tune to helping a trumpet player to write down the notes they have just played.

One method that has been used for monophonic transcription is *autocorrelation* (Brown and Zhang 1991). Our approach is based on an initial autocorrelation analysis, together with detection of note onsets and approximation to the nearest musical instrument digital interface (MIDI) note (Figure 1).

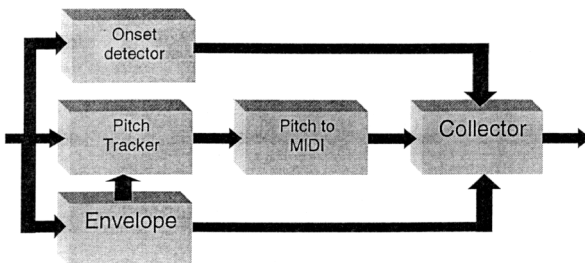


Figure 1. Block diagram of monophonic transcription system.

The sampled audio signal is initially divided up into frames of N samples (in our case $N = 4,096$) and the autocorrelation function can be calculated as:

$$r_{xx}(n) = \frac{1}{N} \sum_{t=0}^{N-t-1} x(t)x(t+n)$$

In practice we calculate a similar function more efficiently via the fast Fourier transform (FFT). The autocorrelation measures the similarity between shifted versions of the time waveform. The delay corresponding to the highest peak in the autocorrelation gives the period of the waveform, thus giving us the pitch of the signal (Figure 2). For higher frequencies, where the duration is only a few samples and many waveforms feature in each frame, more accurate results can be obtained by performing a frequency transform of the frame (Bello, Monti and Sandler 2000).

A complete transcription system needs more than just a running estimate of the current pitch. The estimate needs to be converted into a representation consisting of a succession of notes and durations. To this end, we use an onset detector to determine the start of new notes. The onset detector filters the audio signal to emphasize its high frequency content (which is normally present in the transients at the start of a new note). Relative differences in this energy produce a sharp peak at note

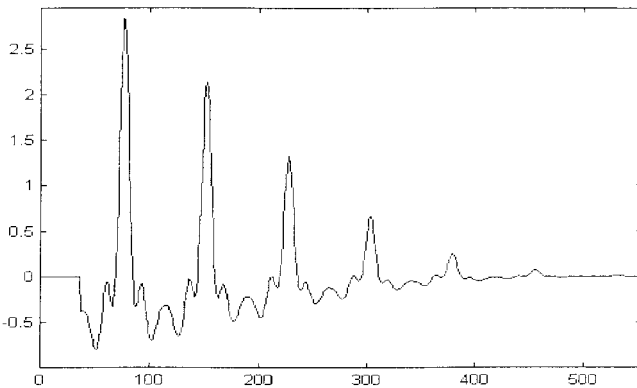


Figure 2. Output from the autocorrelation stage of the monophonic transcription system, revealing a pitch period of about 76 samples. The output corresponding to small delays has been set to zero to suppress the large initial peak at the zero-shift $\tau = 0$ position.

onset transients. Together with the pitch information and overall envelope (volume) this is fed into a collector. The collector uses prior knowledge about musical notes to produce a sequence of notes and durations. For example, notes must have a minimum duration of 100 ms: apparent notes shorter than this will be smoothed out using a memory inside the collector. Because the system is to produce a MIDI sequence, pitches will be assigned to the nearest MIDI note. The end of notes also have special handling: they are determined either by the onset of a new note (at most one note is allowed at a time), or by the envelope falling below an audibility threshold (interpreted as no note playing). The pitch at the end of a note is held flat if it shows a tendency to drift, which is common in forced-vibration instruments.

Example results (Figure 3) show the transcription of a jazz solo trumpet piece. In fact, part of the character of jazz music is the “bending” of pitch away from a normal note pitch, and this is clearly visible in the figure. Thus the transcription, while successfully transcribing the notes into a standard format, will have lost something of the original rendition.

Polyphonic Transcription 1: Blackboard System

Transcription of polyphonic music introduces a number of new complexities that are not present in the monophonic version of this problem. Because we have more than one possible note at once, we can no longer be sure that there will be a single delay at which the whole waveform will repeat, so the autocorrelation approach is no longer suitable. Instead, we base our approaches on an initial time-frequency analysis of the signal, such as that from an FFT.

In order to transcribe the audio signal correctly, we must identify which frequency components are due to each individual note. Each note played will typically generate frequency components at multiples of its fundamental frequency: these frequency components are called *harmonics*. The first harmonic is the fundamental frequency. If we can correctly identify the harmonics that are due to each note, we will have gone some way to solving the transcription problem. However, this is not easy for a number of reasons. For example, the frequency range of the harmonics can be very wide, so some frequency ranges can contain harmonics from two or more different notes.

A more awkward problem is that some frequency components may be caused by harmonics of more than one note. In fact, composers

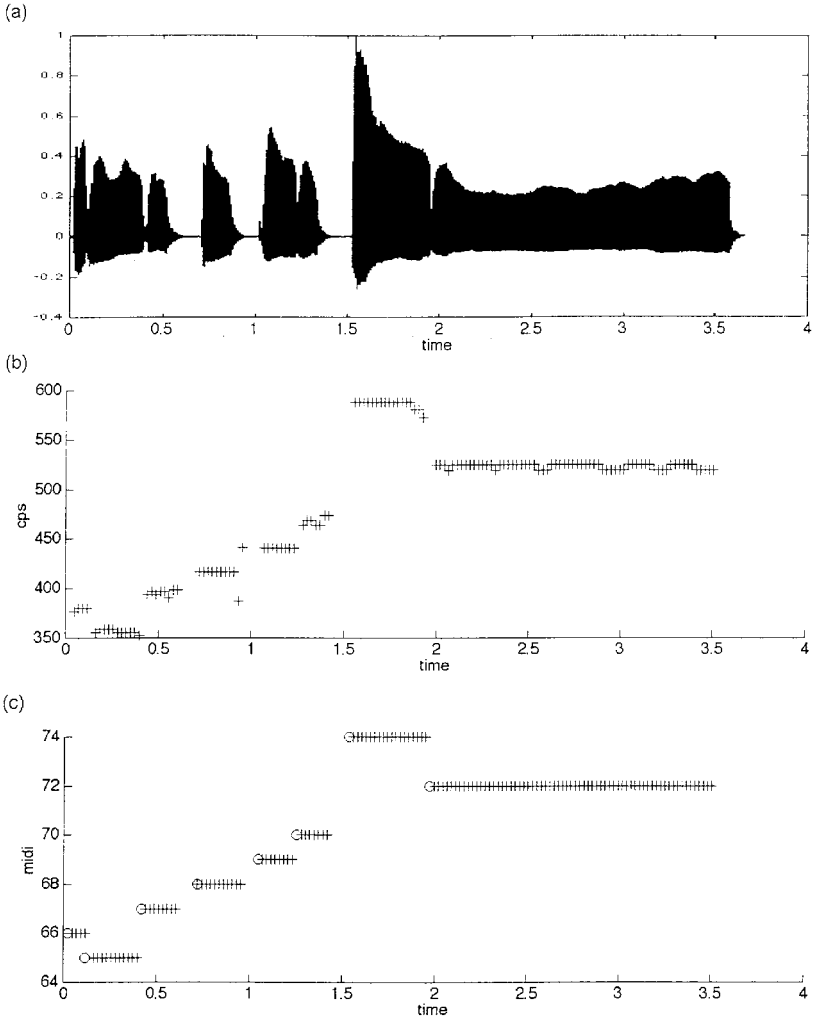


Figure 3. Results of the monophonic transcription process, showing (a) the original trumpet audio waveform (b) the output from the pitch tracker, and (c) the conversion to the nearest MIDI notes, indicated by start time (O) and duration (+).

often use chords containing notes that have a simple ratio between their fundamental frequencies, such as 2:1 (an *octave*), 3:2 (a *fifth*), or 4:3 (a *third*), because these typically produce a sound which is pleasing to the human ear. If one note of 100 Hz is played at the same time as another

one octave up, at 200 Hz, then every harmonic of the upper note will correspond to the even harmonics of the lower note. For example, the component at 400 Hz will be due to both the second harmonic of the note at 200 Hz and the fourth harmonic of the note at 100 Hz. In fact, because the 200 Hz note produces no other frequency components than might be produced by a 100 Hz note, we will have to use some other information (such as the energy expected in each harmonic for a particular instrument) to infer the presence of the 200 Hz note.

To integrate these sources of information, and to make inferences about the likely notes present, we adopted the use of a blackboard system as used by, for example, Mellinger (1991) and Martin (1996). The outline of the blackboard system we used (Bello and Sandler 2000) is shown in Figure 4. The first stage is a signal processing stage, which performs a time-frequency analysis of the audio signal. There are many possible approaches here, including the well-known FFT, where we transform short frames of audio into a representation of the amplitude and phase of the frequency content of the signal within each frame. Alternative time-frequency analysis techniques can be used, such as the multiresolution Fourier transform (MFT), wavelets, and “ear models” built to emulate the human auditory system. These typically permit finer time resolution (and consequently coarser frequency resolution) at higher frequencies, where the normal FFT has the same resolution in time and frequency for all frequency bands, and may help to distinguish note onsets or other fast changes (Pielemeier, Wakefield, and Simoni 1996).

The blackboard system proper is a knowledge-based inference engine that can incorporate information from a variety of sources to produce hypotheses about (in our case) the notes present in the audio signal. The blackboard is a hypothesis database upon which initial input observations are first written. There is also a set of expert agents, or *knowledge sources* (KSs) which are able to make inferences about some of the

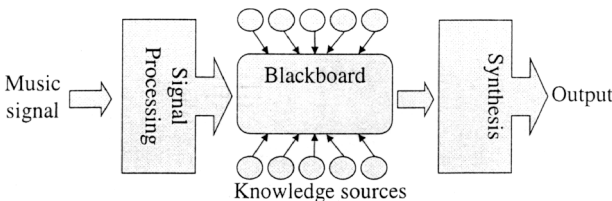


Figure 4. Outline of polyphonic transcription system based on a blackboard model.

hypotheses that may appear on the blackboard. When a KS recognizes a set of hypotheses present on the blackboard, it may fire, writing a new hypothesis onto the blackboard. This new hypothesis may in turn be used by another KS to make a further KS hypothesis. A scheduler determines the order in which KSs are allowed to fire.

As a simple example, suppose that the signal processing stage had determined that frequency components (*tracks*) at 100 Hz, 200 Hz, 400 Hz, and 500 Hz were present. Suppose also that there is one KS that knows that the presence of frequency tracks at f and $2f$ is partial evidence for a note at f , and another KS that knows that tracks at $4f$ and $5f$ is also evidence for a note at f . Suppose also that a third KS knows that two or more hypotheses of partial evidence for a note at f is good evidence for a note at f . Then the first two KSs are able to fire, each generating a partial hypothesis of a note at 100 Hz. The final KS can then integrate these two partial hypotheses into a good hypothesis of a note at 100 Hz, even though its third harmonic (300 Hz) was missing, perhaps due to noise.

Note that the first KS can also generate a partial hypothesis of a note at 200 Hz (because frequency tracks at 200 Hz and 400 Hz are present), but this is not supported by further evidence so is likely to be rejected in this simple example. Depending on the design of the blackboard system, hypotheses may be used up on firing, which can reduce confusion, but may prevent correct operation where a single frequency track is caused by more than one underlying note.

The initial blackboard system we used had three levels of hypotheses: frequency tracks, partials, and notes, with KSs designed for western piano music. We also experimented with additional KSs based on Elman networks and time-delay neural networks (TDNNs) for detection of chords from polyphonic notes, and prediction of melodic lines (Figure 5).

An example of the analysis is shown in Figure 6, where a graphical transcription is shown for part of two piano solo pieces. For the Liszt *Etude*, with two to three notes present at any time, the transcription (Figure 6b) is very close to the original (Figure 6a). The Mussorgski *Promenade* is a "richer" piece, with four to six notes present at any time. While most of the notes here are recognized in some form, the duration of many notes in the transcription (Figure 6d) is shorter than the original (Figure 6c), particularly for the higher frequency notes. Further investigation indicates that these notes decay away relatively quickly, and we have found that it is difficult to set an appropriate threshold that will retain these notes while suppressing noise.

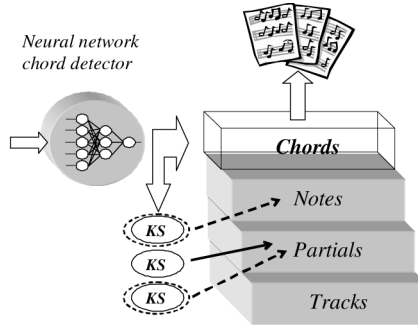


Figure 5. Hypothesis levels within the blackboard system.

Polyphonic Transcription 2: Multiple-Cause Model

One of the issues to be addressed in polyphonic music transcription is the issue of timbre modeling. If we knew what the shapes of the notes were like, in the frequency domain, we could find a combination of notes that exactly matched the frequency content of the input. However, this is somewhat of a chicken-and-egg problem: until we know what the note shapes are for a particular piece, it seems that we cannot find the notes, and if we do not know the notes, we cannot find the note shapes. One approach would be to obtain initial samples of all instruments and notes in isolation (Kashino et al. 1998), but this may not always be possible: perhaps if a single recording of unusual instruments was all that was available. In this case, one possible solution to this is to use a type of neural network called the *multiple-cause model* (Saund 1995).

The multiple-cause model (Figure 7) searches for representations of the underlying causes of the input data, together with amounts of each cause, which take account of the input data as closely as possible. This model is designed to cope with input data that is composed of several causes active at the same time. In contrast to many neural networks, this network does not operate in a simple feed-forward manner; rather, the encoding layer and connections are adjusted until the encoding forms a good reconstruction of the observed data.

The multiple-cause model was originally proposed for analyzing images into constituent causes. A classic example is the bars problem, where each image is composed of a number of horizontal and vertical black bars on a white background. Thus an image is caused by several

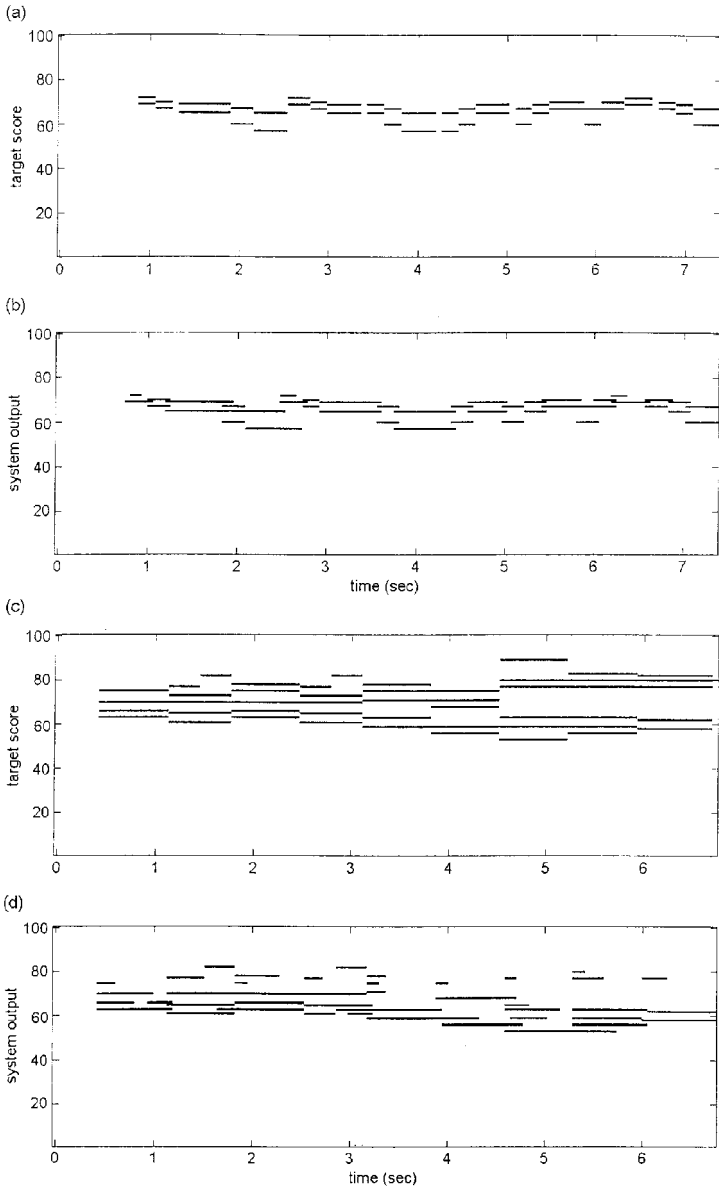


Figure 6. Results of polyphonic music transcription using the blackboard system: (a) extract of Liszt: *Etude No. 5 aus Grandes Etudes de Paganini*, (b) transcription of (a), (c) extract of Mussorgsk: *Promenade - Ballett der Kuchlein in ihren Eierschalen*, and (d) transcription of (c). This rendition of the extracts is copyrighted by Bernd Krueger.

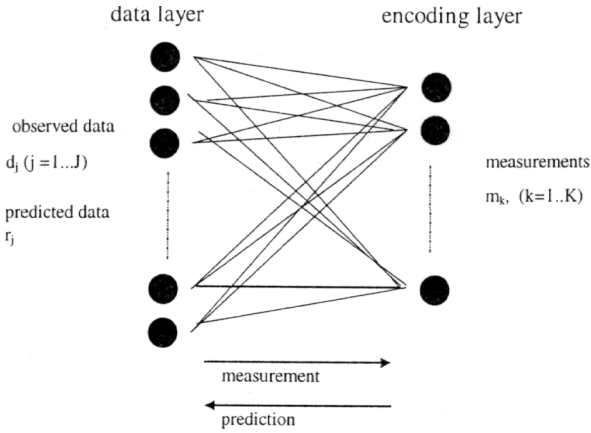


Figure 7. Multiple-cause model.

bars, maybe overlapping, and each pixel in the input image may be caused by one or more bars being present in the image. The multiple-cause model is trained by minimizing an error function, such as negative log likelihood or mean squared error, by adjusting both the underlying measurements of each cause and the patterns due to each cause.

The multiple-cause model is underdetermined, due to interdependence between the measurements of each cause and the underlying patterns. Depending on the problem, some form of constraints may need to be imposed (Charles and Fyfe 1998; Harpur and Prager 1995). For a musical example, we would expect positive amounts of each underlying note. On the assumption that notes and instruments were independent sources, each would then contribute an approximately positive amount to the received spectrum.

Artificial Spectra. As an initial step, we next applied the multiple-cause model to artificial spectra produced from synthesized sounds (Klingseisen and Plumbley 2000). We first tested the ability to separate spectra from different notes of the same instrument. We used linear mixtures of spectra, downsampled to 30 bins, of a synthesized clarinet playing one of 8 notes (G_3 , C_4 , A_3 , D_4 , F_4 , G_4 , A_4 , E_4), with a probability of 0.4 of each spectrum appearing in each mixed pattern. We repeated this for notes from a synthesized violin and also for an alto recorder: in each case presentation of about 800 training patterns were needed for successful learning. We also explored separation of synthesized spectra of different

instruments playing the same note, with similar results, although it appeared to take longer to learn than the experiments with different notes on the same instrument. This is probably due to the similarity between the patterns representing the same note, particularly the alignment of the fundamental and first few harmonics into the same bins in each pattern.

Combining these two approaches, patterns composed from the spectra of three instruments (clarinet, oboe, and trumpet) playing each of three notes were used for the training patterns. The spectra used to form the training patterns were downsampled into log-scaled bins, with a relative scaling of $2^{1/12}$ (equivalent to one semitone) between the bins. With synthesized instruments, different notes on the same instrument would then appear simple as shifted along this log-frequency scale, with 1 bin shift equivalent to 1 semitone. A multiple-cause model learned the basic pattern spectra after about 700 presentations of the training patterns. After this, we were able to post-process the resulting patterns to identify which patterns were relative shifts of each other. This correctly identified that three instruments were used, with relative semitone shifts of (0, +2, +4), (0, +1, +5), and (0, +2, +3).

Real Sounds. We constructed an audio signal composed of a linear time-domain addition of pulses of notes played on different instruments from the University of Iowa musical instrument samples web page (<http://theremin.music.uiowa.edu/~web/sound/>). This time the audio signals (rather than the spectra) were added, to be closer to the situation in real audio mixing. The multiple-cause model found most of the nine underlying patterns after 300 presentations of the set of training patterns, equivalent to 20 hours learning using Matlab on a 350 MHz Pentium II (Klingseisen and Plumbley 2000).

In Figure 8 we see that the sounds have been separated such that the input sound is represented by a small number of measurement units, with other units off. This indicates that the output units have found a *sparse coding* (Field 1994), that is, a coding with many units off, even without penalty terms that have been used to encourage sparse output distributions (Harpur and Prager 1996; Olshausen and Field 1996).

We can see that there are some instruments that are not completely separated: in particular, it seems that separation of flute and clarinet is difficult. In this example, flute E_{4b} (input 8) and clarinet G_{4b} (input 2) are poorly separated, with parts of each instrument found in the

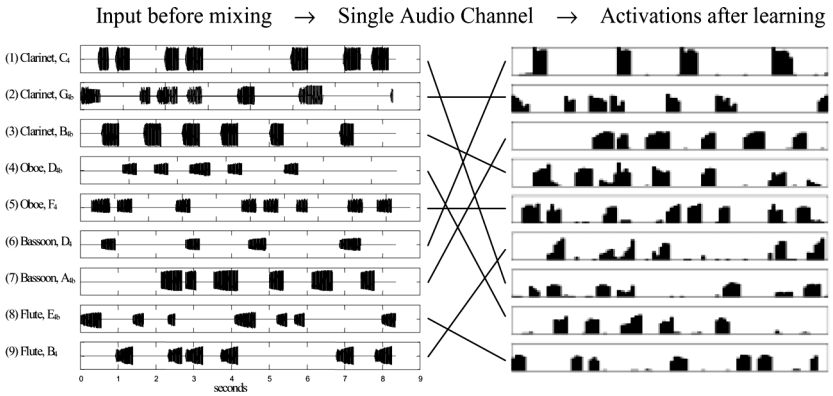


Figure 8. Results of multiple-cause model separation of audio sounds.

corresponding outputs—the label on the right-hand side indicates the closest instrument/note. Also, some of the flute B₄ (input 9) is still mixed with the clarinet B_{4b} (input 3), with the attack phase of the flute being picked up by the clarinet output. This difficulty may be due to the relatively pure waveforms, and therefore dominant fundamentals, that these instruments have, although more investigation is needed to confirm this.

While this multiple-cause model does not yet give us a quality of analysis that we would need for accurate transcription, the concepts it embodies do crop up again in the approach we describe at the end of this article. However, before we go on to describe this, we will discuss some of the concepts and technology in *blind source separation* that we have used elsewhere, and that will form part of our final polyphonic transcription method.

BLIND SOURCE SEPARATION AND INDEPENDENT COMPONENT ANALYSIS

So far we have concentrated on the transcription problem of analyzing a musical audio signal, normally a single channel, to reveal the underlying notes that caused the signal that was observed. Another task we are investigating is the problem of separating out the sounds from several sources, when more than one observed signal (microphone) is available. This is sometimes called the *cocktail party problem*, after the apparent

ability of two-eared human listeners to separate the various conversations going on in a cocktail party, and concentrate on just one. This problem of blind source separation (blind because the mixing process is not assumed to be known) has recently been the subject of much research, particularly since the paper of Bell and Sejnowski (1995) brought this problem to the attention of the neural networks community.

In the simplest form of this problem (Figure 9), we suppose that we have n observations containing different instantaneous linear mixtures of n original sources, and also that the original sources are independent from each other. In this case we can use *independent component analysis* (ICA) to tackle this problem. Mathematically, we express this as $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$ where \mathbf{x} and \mathbf{s} are n -dimensional real vectors, and \mathbf{A} is a nonsingular mixing matrix. The probability density of the source components s_i are assumed to be independent, so that $p(\mathbf{s}) = \prod_i p(s_i)$.

A typical ICA method proceeds in two stages (Figure 10). First, covariances in the observations $\mathbf{x}(t)$ are removed by *whitening*, that is, linearly transforming the observations to give $\mathbf{z}(t) = \mathbf{Q}\mathbf{x}(t)$ where \mathbf{Q} is a linear matrix and \mathbf{z} has identity covariance matrix, and is therefore *decorrelated*. Second, the axes are rotated until the outputs become truly independent. Here this second stage relies on higher-order statistics, such as fourth order.

The ICA procedure has a number of limitations. For example, it cannot separate sources with more than one Gaussian (normally distributed) source. These will be decorrelated, but the original source directions will remain unknown. Another is that the estimated sources obtained are subjected to an unknown scaling and perturbation. We cannot tell what the amplitudes (variances) of the original sources were

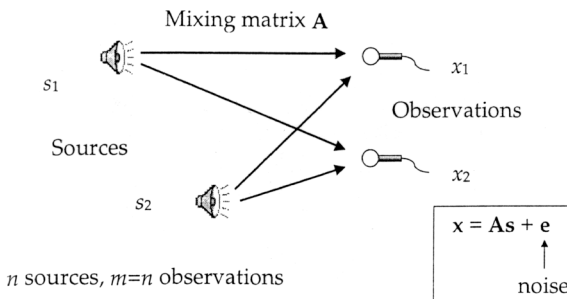


Figure 9. The simple blind source separation problem.

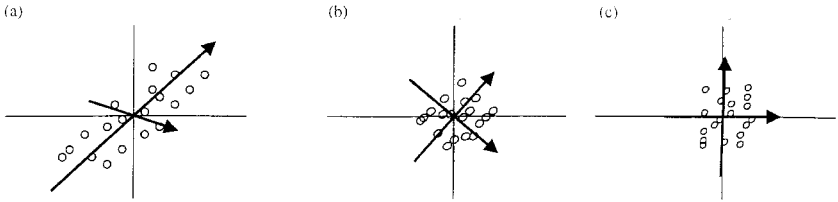


Figure 10. A typical ICA procedure (a) takes the original data, (b) performs a whitening stage to remove covariances, and finally (c) rotates the data axes to produce the independent outputs.

(including knowing whether our estimate is an inverted version of the source), nor can we tell in which order they originally appeared. In many cases, this latter issue, the so-called *permutation problem*, does not really matter, but it can be important if we are considering several linked problems where the relative order is critical.

Convolutional Mixing

In realistic auditory environments, room acoustics can introduce delays, echoes, and reverberation, leading to convolutional mixing of the sources on their way to the microphones (Figure 11). We can either view this mixing in the time domain, as $\mathbf{x}(t) = \sum_{\tau} \mathbf{A}(\tau) \mathbf{s}(t - \tau)$ or in the frequency

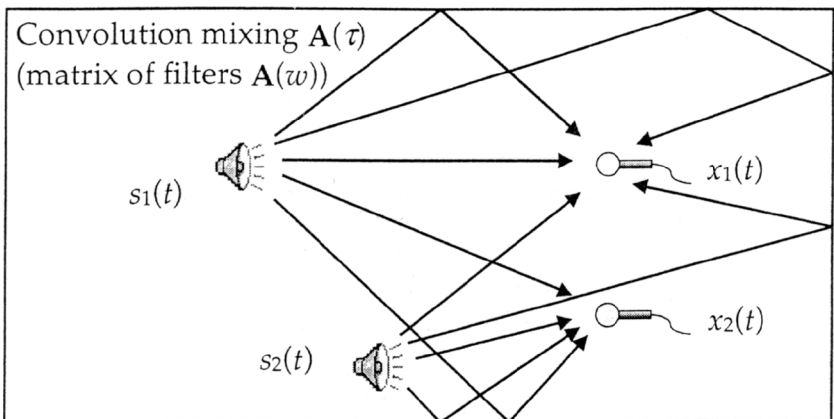


Figure 11. Convolutional mixing: the observations are composed of various delayed versions of the original sources, which we can consider to be filtered versions of the original sources.

domain as $\mathbf{X}(\omega) = \mathbf{F}(\omega)\mathbf{S}(\omega)$ where $\mathbf{F}(\omega)$ is the Fourier transform of $\mathbf{A}(\tau)$, and \mathbf{X} (resp. \mathbf{S}) is the Fourier transform of \mathbf{x} (resp. \mathbf{s}). Thus in the frequency domain, we can consider to be many parallel source separation problems, for each different value of ω .

Several authors have proposed algorithms to tackle this problem (Cichocki, Amari, and Cao 1996; Lee, Bell, and Lambert 1997; Smaragdis 1997). Many of these approaches can be viewed as attempting to perform ICA on different frequency bands, with some coupling between the frequency components. The coupling is necessary to overcome the permutation problem that we mentioned earlier. If we simply attempted to separate each frequency band independently, we may get, for example, frequency band ω_a of source 1 presented on output 1, but frequency band ω_v of the same source being presented instead on output 2.

Our approach to this problem (Davies 2001) relies on the tendency for the amplitudes of the various frequency bands of a particular source to vary together. In a Bayesian modeling framework, we introduce an extra scale parameter b_k for each source k , which scales all frequency components of that source, and is assumed to vary slowly. In the different frequency bands we then use a *likelihood ratio jump test* to try out the possible permutations, and choose the permutation with maximum likelihood. We compared our algorithm with the Smaragdis (1997) algorithm, this time on speech sources. For simple room acoustics, we found that either approach worked well. However, for room acoustics with non-trivial echo, for example at 5–10 ms, we found that the Smaragdis algorithm failed to separate the sources, while our likelihood ratio jump test was still successful (Figure 12). We also found that it was possible to obtain a similar result on simpler acoustics by randomizing the initial mixing matrices. Thus it may be that the Smaragdis algorithm (which uses a gradient descent method) may be trapped in local minima, while the likelihood ratio jump test is able to jump out of these if required.

TRANSCRIPTION FROM ICA AND SPARSE CODING

In the previous section, we considered how to separate mixtures of original source signals, where different amounts of each source signal arrive in each microphone due to an unknown linear mixing process, and where we have N microphones to separate N sources. But consider again the transcription problem, where notes of unknown timbre (spectral shape) are mixed together to give the audio signal we listen to. If we consider our

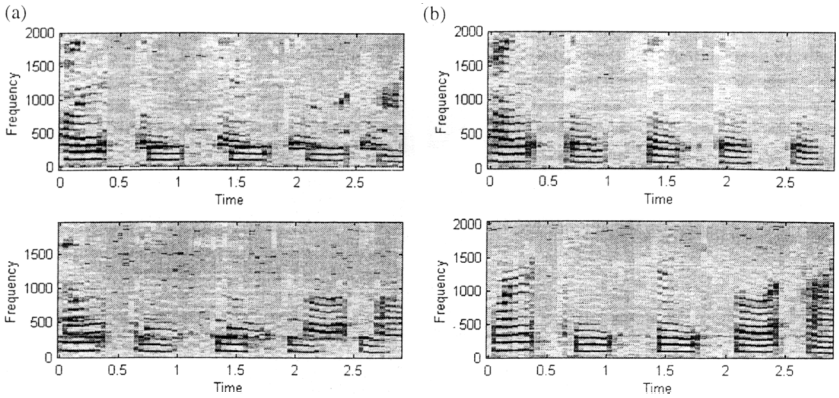


Figure 12. Separation results for mixtures of speech (“A-B-C-D-E” mixed with “1-2-3-4-5”) with nontrivial echo for (a) the Smaragdis (1997) algorithm and (b) the likelihood ratio jump test. We can see that the Smaragdis algorithm appears to have extracted the low frequency components from one source, together with the high frequency components of the other, and vice-versa.

N observations to be different frequency components of the signal, we should be able to use an ICA-like approach to separate out the note activities as if they were different sources. Here the mixing matrix, or *basis*, that we find will be the spectral shape of each note, which should all be different (even if some are just a frequency-scaled version of others).

One other principle will help us. For most musical pieces, even if they are polyphonic, only a few of the many possible notes are being played at any one time. If we wish to describe the volume, or amplitude, of all notes at any particular time, most of these will be zero. This leads to the concept of *sparse coding* (Field 1994), where each item is coded using a representation where only a few values are nonzero. We came across this in our discussion of the multiple-cause model, and sparse coding has also been suggested as suitable for representing natural images (Olshausen and Field 1996).

This sparse coding approach can help to overcome some of the limitations associated with standard ICA. For example, it is possible to separate more sources than the observations available (the *overcomplete basis* case), and it is capable of handling noise on the observations (Lewicki and Sejnowski 2000). Thus it may be able to separate an audio signal into more possible notes than the frequency components available (see also Bofill and Zibulevski 2000). The sparse coding method is a

Bayesian, probabilistic method. The “mostly zero” expectation is modeled as a prior probability for note activation which is sharply peaked around zero. Otherwise, the structure of the sparse coding model is similar to the multiple-cause model described earlier. It attempts to find a set of sparse and independent notes, together with their spectral shapes, which will reconstruct (or regenerate) the observed signal. This is therefore sometimes referred to as a *generative model* or *analysis-by-synthesis* approach.

We tested our algorithm on real sounds generated by a MIDI synthesizer and then resampled. We used Bach: *Partita in A minor for keyboard* (BWV827), which was chosen because it mainly consists of two or three independent lines, with a few block chords. We used magnitude spectra rather than power spectra as input to the model, so that the noise is approximately Gaussian, as assumed in our model (Abdallah and Plumbley 2001). The model learned 55 basic spectra, 49 of which were note spectra, and others appeared to represent transitions (e.g. note onsets) rather than notes. When the output activations of these basis vectors are plotted, they show a good “piano-roll” representation for the music contained in the input spectrogram (Figure 13).

The note shapes obtained can clearly be seen in Figure 13c. Due to the permutation problem, the sparse-coding method cannot automatically identify which note shape corresponds to which note pitch. In this example, the notes were previously reordered by eye, but we anticipate that it should be possible to construct automatic reordering methods in future.

To check that the analysis was reasonable, we performed a simple resynthesis by feeding the identified notes into a MIDI piano synthesizer. After some trial-and-error adjustments, the result was a passable rendition of the original piece. A few notes can be heard to be missing here and there, and timing jitters are present that made the resynthesis sound a little like a slightly nervous piano student, although these are more likely to be created by different relative alignment of frame boundaries with note onset times.

DISCUSSION

Analysis and separation of musical audio is still in relative infancy at present, compared with, for example, automatic speech recognition. However, we have tried to give an indication of the variety of techniques

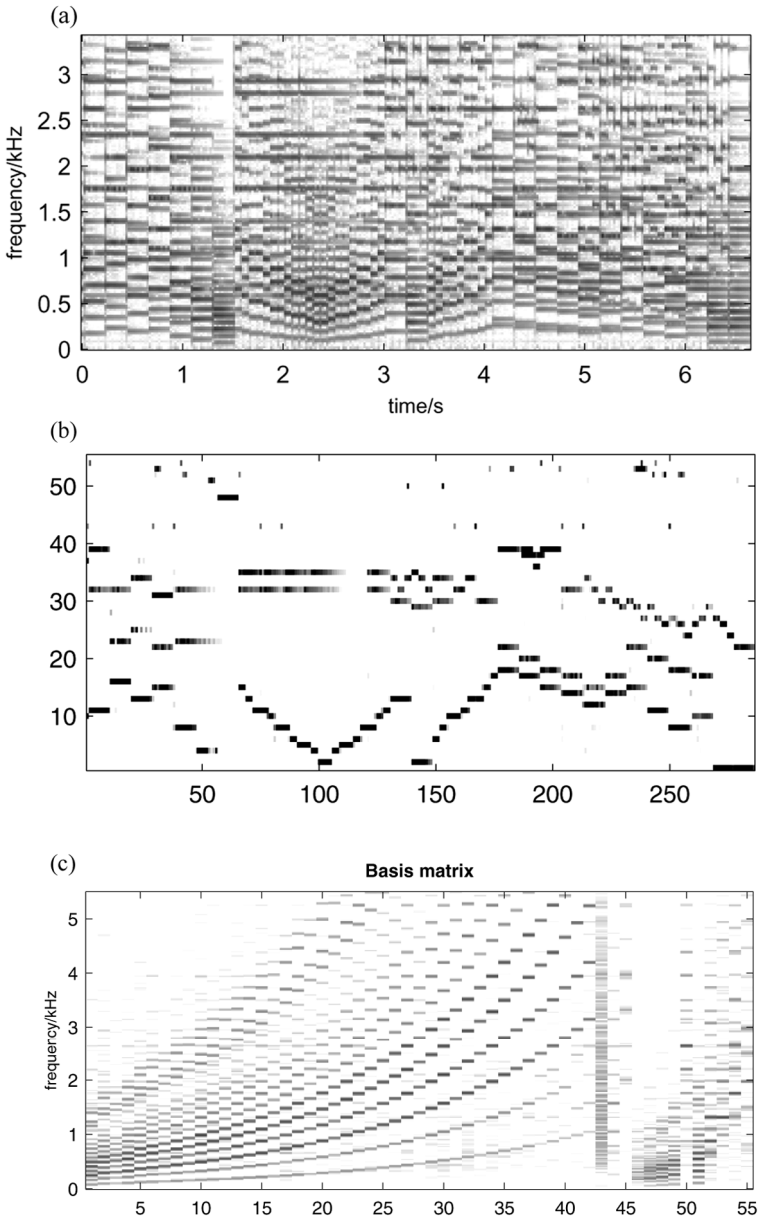


Figure 13. Polyphonic transcription of extract of Bach: *Partita*, showing (a) input spectrogram and (b) output note activations, which are visibly sparser than the input. The note shapes (c), which had to be ordered manually, clearly show the harmonic structure, except for the notes above 43 which probably represent transitions.

being used to tackle this problem, from one particular perspective. Other approaches to music analysis and transcription include the Bayesian approach (Kashino et al. 1998; Walmsley, Godsill, and Rayner 1999), estimation of multiple sinusoids using fast block-based algorithms (Macleod and Tsatsoulis 1999), the number theoretic approach (Klapuri 1998), the use of Kalman filters (Sterian 1999), or comodulation and dynamical clustering (Scheirer 2000). We feel that this problem is a very interesting and fruitful area for future research.

In blackboard systems, knowledge sources can be quite complex, and may rely on many sets of expert knowledge. For example, Godsmark and Brown (1999) describe a blackboard model which integrates evidence from several grouping principles, such as temporal and frequency proximity, common onset and offset times, common frequency modulation, and similarity of timbre (the frequency shape of each note). Each principle is handled by a special hypothesis formation expert, with the parameters of the KSSs set to be consistent with known psychophysical measurements.

There are a number of issues that are outside the scope of this article, although they are likely to be important for the analysis of musical audio (see also Byrd and Crawford 2001). We have, for example, largely ignored the temporal structure of the musical signals so far, concentrating to a large extent on the frequency content of the current audio signal. However, music includes many types of time structure, from small, fast variations of pitch (e.g. vibrato), through rhythmic beat structure, to the structuring of verses, and these are likely to be an important part of any practical music transcription system.

Another issue that is likely to complicate practical music transcription systems is that the timbre of a note (its frequency content) changes over time. For example, in a plucked-string instrument such as the harpsichord, when the string is first plucked there is a wide range of harmonics present in the sound, and a bright sound is heard. However, the high frequency harmonics decay relatively quickly, as they lose their energy, so the note becomes darker, leading to the concept of *timbre tracks* traced out by a note as its sound changes (Godsmark and Brown 1999). Our current sparse coding method relies on the timbre remaining relatively constant, so this issue will need tackling before we move on to more complex instruments. In passing it is worth mentioning that monophonic music transcription does not need information about the timbre of notes to analyze the pitch, so it is likely to be more robust against changing timbre than many polyphonic methods.

There are also instruments that do not have a fundamental simple harmonic structure, such as percussion instruments like drums or cymbals, and the polyphonic analysis methods we have outlined in this article would not be suitable for these. Even apparently straightforward instruments, such as the piano, may be more complex than appears at first. Due to the different modes of vibration of the strings within the piano, some harmonics occur at slightly noninteger multiples of the fundamental frequency. In effect, their phase drifts (i.e., they drift “out of sync”) relative to the fundamental as time proceeds (Daudet, personal communication, 2001).

Although we have concentrated here on the concept of automatic music transcription for writing down a printed score, one potential application for automatic music transcription is in the area of audio coding for compression. Currently most audio compression methods, such as the currently popular MP3 format (MPEG-1 Layer 3), attempt to reduce the number of bits used to transmit certain frequency bands to match their audibility. An alternative approach would be to analyze the audio to uncover the “objects” that gave rise to the audio sound, and send a coded description of those objects instead of the audio itself. The recent MPEG-4 structured audio (MP4-SA) standard (Vercoe, Gardner, and Scheiret 1998) would offer a possible mechanism to transmit these audio objects, by sending an orchestra description, describing what instruments (sound sources) are available, followed by a score, specifying how these should be played. Currently these MP4-SA descriptions are authored directly (e.g., from a human playing at a keyboard), but if high quality polyphonic music transcription is possible, these structured descriptions could be created automatically from audio signals. With the continual need for bandwidth to store and deliver music over the Internet, including delivery to mobile devices, it may be that this object-based coding application turns out to be more important than the production of musical scores.

CONCLUSIONS

We have given an overview of a range of approaches to analysis and separation of musical audio used within our group. In particular, we have investigated the problems of automatic music transcription and audio source separation. Monophonic music transcription, where a single note is present at one time, can be tackled using an autocorrelation-based method. For polyphonic music transcription, with several notes at any

time, other approaches can be used, such as a blackboard model or multiple-cause/sparse coding method. The latter is based on ideas and methods related to ICA, a method for sound source separation.

We feel that the research area of musical audio analysis and separation is a very interesting and fruitful area for future research, and may lead to improved methods for audio coding and compression as well as tools to assist musicologists.

REFERENCES

- Abdallah, S. A., and M. D. Plumbley. 2001. *Sparse coding of music signals*. Submitted for publication.
- Bell, A. J., and T. J. Sejnowski. 1995. An information maximization approach to blind separation and blind deconvolution. *Neural Computation*: 1129–1159.
- Bello, J. P., G. Monti, M. B. Sandler. 2000. Techniques for automatic music transcription. In *Proceedings of the International Symposium on Music Information Retrieval (MUSIC IR 2000)*. 23–25 October, Plymouth, MA.
- Bello, J. P., and M. B. Sandler. 2000. Blackboard system and top-down processing for the transcription of simple polyphonic music. In *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00)*. 7–9 December, Verona, Italy.
- Bofill, P., and M. Zibulevski. 2000. Blind separation of more sources than mixtures using sparsity of their short-term Fourier transform. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation*. 19–22 June, Helsinki.
- Bello, J. P., G. Monti, and M. B. Sandler. 2000. Blackboard system and top-down processing for the transcription of simple polyphonic music. In *Proceedings of the COST G-6 Conference on digital Audio Effects (DAFX-00)*, 7–9 December, Verona, Italy.
- Bofill, P., and M. Zibulevski. 2000. Blind separation of more sources than mixtures using sparsity of their short-term Fourier transform. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation*, 19–22 June, Helsinki.
- Bregman, A. 1990. *Auditory scene analysis*. Cambridge, MA: MIT Press.
- Brown, J. C., and B. Zhang. 1991. Musical frequency tracking using the methods of conventional and ‘narrowed’ autocorrelation. *Journal of the Acoustical Society of America* 89:2346–2354.
- Byrd, D., and T. Crawford. 2001. Problems of music information retrieval in the real world. *Information Processing and Management*. Submitted for Publication.

- Charles, D., and C. Fyfe. 1998. Modelling multiple-cause structure using rectification constraints. *Network: Computation in Neural Systems* 9:167–182.
- Cichocki, A., S. Amari, and J. Cao. 1996. Blind separation of delayed and convolved signals with self-adaptive learning rate. In *Proceedings of the 1996 International Symposium on Nonlinear Theory and Its Applications (NOLTA '96)*, 7–9 October, Kochi, Japan.
- Davies, M. E. 2001. Audio source separation. In *Mathematics of signal processing V*, edited by J. G. McWhirter and I. K. Proudler. In press.
- Ellis, D. P. W. 1996. *Prediction-driven computational auditory scene analysis*. Ph.D. diss., Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.
- Field, D. J. 1994. What is the goal of sensory coding? *Neural Computation* 6:559–601.
- Godsmark, D., and G. J. Brown. 1999. A blackboard architecture for computational auditory scene analysis. *Speech Communication* 27:351–366.
- Harpur, G. F., and R. W. Prager. 1995. *Techniques for low entropy coding*. CUED/F-INFENG/TR. 197, Engineering Department, Cambridge University, Cambridge, UK.
- Harpur, G. F., and R. W. Prager. 1996. Development of low entropy coding in a recurrent network. *Network: Computation in Neural Systems* 7:277–284.
- Kashino, K., K. Nakadai, T. Kinoshita, and H. Tanaka. 1998. Application of the Bayesian probability network to music scene analysis. In *Computational auditory scene analysis*, edited by D. F. Rosenthal and H. Okuno, 115–137. Mahwah, NJ: Lawrence Erlbaum.
- Klapuri, A. 1998. Automatic transcription of music. M. Sc. thesis, Department of Information Technology, Tampere University of Technology.
- Klingseisen, J., and M. D. Plumbley. 2000. Towards musical instrument separation using multiple-cause neural networks. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation, (ICA'2000)*, 19–22 June, Helsinki.
- Lee, T. W., A. J. Bell, and R. H. Lambert. 1997. Blind separation of delayed and convolved sources. edited by M. C. Mozer, M. I. Jordan, and T. Petsche. In *Advances in Neural Information Processing Systems 9*, 758–764. Cambridge, MA: MIT Press.
- Lewicki, M. S., and T. J. Sejnowski. 2000. Learning overcomplete representations. *Neural Computation* 12:337–365.
- Macleod, M. D., and C. I. Tsatsoulis. 1999. Approaches to music analysis using the sinusoid plus residual model. Presented at the *Cambridge Music Processing Colloquium*, 30 September, Cambridge, UK.
- Martin, K. 1996. A blackboard system for automatic transcription of simple polyphonic music. Technical Report No. 385, Perceptual Computing Section, Massachusetts Institute of Technology Media Laboratory.

- Mellinger, D. K. 1991. Event formation and separation in musical sound. Ph.D. diss., Center for Computer Research in Music and Acoustics, Stanford University.
- Olshausen, B. A., and D. J. Field. 1996. Natural image statistics and efficient coding. *Network: Computation in Neural Systems* 7:333–339.
- Pielemeier, W. J., G. H. Wakefield, and M. H. Simoni. 1996. Time-frequency analysis of musical signals. *Proceedings of the IEEE: Special Issue on Time-Frequency Analysis* 84:1216–1230.
- Saund, E. 1995. A multiple cause mixture model for unsupervised learning. *Neural Computation* 7:51–71.
- Scheirer, E. D. 2000. Music-listening systems. Ph.D. diss. Massachusetts Institute of Technology Media Laboratory.
- Smaragdis, P. 1997. Information theoretic approaches to source separation. M. Sc. thesis, Media Arts and Sciences Department, Massachusetts Institute of Technology.
- Sterian, A. D. 1999. Model-based segmentation of time-frequency image for musical transcription. Ph.D. diss., Department of Electrical Engineering, University of Michigan.
- Vercoe, B. L., W. G. Gardner, and E. D. Scheirer. 1998. Structured audio: Creation, transmission, and rendering of parametric sound representations. *Proceedings of the IEEE* 86:922–940.
- Walmsley, P. J., S. J. Godsill, and P. J. W. Rayner. 1999. Polyphonic pitch tracking using joint Bayesian estimation of multiple frame parameters. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 17–20 October, New Paltz.