RECOGNITION OF ACOUSTIC NOISE MIXTURES BY COMBINED BOTTOM-UP AND TOP-DOWN PROCESSING

Jukka Sillanpää, Anssi Klapuri, Jarno Seppänen, Tuomas Virtanen
Signal Processing Laboratory, Tampere University of Technology,
P.O.Box 553, FIN-33101 Tampere, FINLAND
Tel: +358 3 3652124; fax: +358 3 3653857
e-mail: {s157529,klap,jams,tuomasv}@cs.tut.fi

ABSTRACT

In this paper, a system is described for the recognition of mixtures of noise sources in acoustic input signals. The problem is approached by utilizing both bottom-up signal analysis and top-down predictions of higher-level models. The developments are made using musical signals as test material. Validation experiments are presented both for self-generated sound mixtures and for real musical recordings.

1 INTRODUCTION

Recognition of acoustic noise mixtures is viewed here as the detection and broad classification of noisy sound sources in acoustic mixture signals. Applications of this are numerous, including acoustic surveillance, speech processing in a noisy background, acoustic database queries, noise pollution controlling, and hearing aids. However, performing the task faces several fundamental problems. Resolving a set of stochastic signals from their linear mixture is a complicated and usually ambiguous search problem. Besides that, modeling sound sources in a way that defines their characteristics yet retains flexibility for inside-class variations is difficult.

Automatic noise recognition (ANR) has been an area of active research during the last few years. Recognition of isolated noise sources has obtained good results. Some promising techniques include e.g. statistical signal processing [1,2] and hidden Markov models combined with linear prediction coding [3]. Recognition of mixtures of noise signals is significantly more difficult. Couvreur et al. have taken the approach to decompose noise mixtures into a set of stationary Gaussian processes, using minimum description length (MDL) criterion to optimize the solution in information theoretic sense, but paying no attention to the temporal structure of the signals [4,5]. Automatic learning of an unknown number of source models from their linear mixtures has been addressed by Linares et al. in [6].

In this paper, we approach the problem by including top-down processing algorithms. *Bottom-up* techniques are characterized by bottom-up flow of information: observations from an acoustic waveform are combined to meaningful features and passed to higher levels for interpretation. *Top-down* processing utilizes internal, high-level models of the acoustic environment and prior knowledge of the properties and dependencies of the objects in it [8].

The developments to be described are made using musical signals as an experiment material. Music is a nice area for developing ANR methods, since it comprises a wide
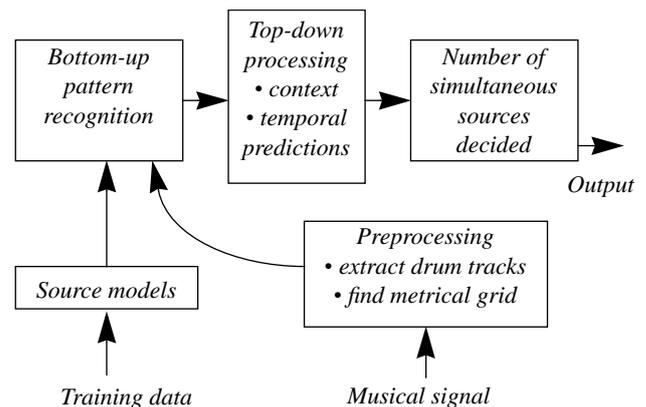

Figure 1. System overview.

range of well defined and varying classes of sound sources, on which a lot of test material is available. Secondly, music provides a rich collection of top-down processing rules in the theory of music and composition. The concrete mechanisms to be described include bottom-up pattern recognition, and top-down temporal predictions and utilization of context. Validation experiments were made both for self-generated sound mixtures and for real musical recordings.

2 SYSTEM OVERVIEW

The overall system comprises the blocks illustrated in Figure 1. An incoming musical signal is first processed by a sinusoids plus noise model, which is used to extract the drum tracks, i.e., noise residual from the input signal [9]. This underlies the assumption that drum sounds remain in the residual signal, and others not. Although not strictly correct, this assumption is useful in practice. This is followed by a so-called metrical grid estimator, which finds the time quantization step and produces temporal anchor points over the signal. Recognition of drum instruments takes place for the noise residual at each onset (beginning of a sound), at the points of the metrical grid.

Bottom-up signal analysis is based on a pattern recognition algorithm which uses time-frequency energy spectrograms as features. Frequency resolution is according to Bark scale critical bands, time resolution is logarithmic, gradually decreasing the further we get from the onset of the sound. Source models are compiled by extracting the spectrograms from a collection of training samples, and using a clustering algorithm to yield a limited number of models per class.

The source probabilities from bottom-up signal analysis are further manipulated by top-down error correcting algo-

rithms, which utilize temporal predictions and the context. Finally, the number of simultaneous sources at each onset is decided using an information theoretic criterion which compromises between number of sources and remaining error.

3 PREPROCESSING

3.1 Sinusoidal Modeling

Drum tracks of a musical signal are extracted by a sinusoids plus noise spectrum model, described by Tuomas Virtanen in [10]. It estimates the harmonic parts of the signal and subtracts them in the time domain to obtain the noise residual.

The idea of sinusoidal modeling is to use sinusoids with time-varying frequencies and amplitudes to represent the harmonic components of an audio signal. To obtain these sinusoids from the original signal, its short-time spectra is analyzed by taking discrete Fourier transform of the windowed signal to locate the prominent peaks in the amplitude spectra. The prominent spectral peaks are interpreted as sinusoidal waves at detected frequency in analysis window, and the amplitudes and phases of the sinusoids can be obtained from the complex spectrum.

Once amplitude and frequency of each peak are estimated at each frame, the detected peaks are tracked together in interframe trajectories. A peak continuation algorithm tries to find the most suitable peak in the next frame, which has the frequency and amplitude close to the existing trajectory in current frame. The result of the peak continuation algorithm is a set of sinusoidal trajectories with time-varying frequencies and amplitudes [9].

From detected trajectories, a series of sinusoids can be synthesized which reproduce the instantaneous phase and amplitude of the partials of original signal. To avoid clicks at the frame boundaries, the parameters are smoothly interpolated from frame to frame. An instantaneous amplitude can be easily obtained using linear interpolation. Frequencies and phases are tied together (frequency is the phase derivative), so we use cubic interpolation function for them.

Residual is obtained by subtracting the synthesized sines from the original signal in time domain.

3.2 Metrical grid estimation

Metrical grid estimation is viewed here as the task of finding the shortest and perceptually lowest-level pulse in music, which is called the *metrical grid*. The other, temporally longer pulse sensations are its integer multiples.

The used metrical grid estimation has been described by Jarno Seppänen in [10]. The algorithm is based on interonset intervals and greatest common divisors. Input data consists of onset times that are calculated from the acoustic signal using the algorithm described in [11], and the output is the metrical grid. Drum detection and recognition takes place at each onset point at the metrical grid.

4 BOTTOM-UP PATTERN RECOGNITION

4.1 Feature extraction

Through psychoacoustic experiments, it has been found that for stationary, noise-like signals, the ear is not sensitive to variations of energy within each Bark band. Between 20 Hz and 22 kHz there are 25 Bark critical bands. Therefore, we need only to know the short-time energy at each Bark band [9]. These are defined as

$$z = \left\lfloor 13 arc\tan(0.76f) + 3.5 arc\tan\left(\frac{f}{7.5}\right)^2 \right\rfloor, \qquad (1)$$

where the frequency f is in kHz.. As in detection of sinusoids, the residual is windowed and spectrum is obtained using Fourier transform. Then short-time energy within each Bark band is calculated frame by frame.

Time resolution is made logarithmic by sampling Bark band energies at time instants

$$t_k = 10\exp(0.36 \cdot k) - 10 \ \text{ms}, \qquad (2)$$

where $k=0,1,2,...$, used value for $r=0.36$, and the reference point is at sound onset. This gradually decreasing rate of sampling gives more emphasis near the beginning of the sound, and provides robustness for sound with varying durations. Feature extraction is done at each onset, and for each training drum sample.

4.2 Source modeling

Although a basic drum set comprises only a small number of different instrument classes, there is a lot of inside-class variation. Drum dimensions, different physical constructions, and playing techniques make large variations in the produced tonal colour. In a spectral pattern recognition approach, it is unimaginable to use a single average model to represent each class. Instead, several models per class are needed to include the different subclasses.

The models were constructed by forming the described Bark-frequency and log-time resolution spectrograms from each training sample. These were encoded into a feature vector, where the overall energy in each frame was put together with the normalized energies at each Bark band, expressed in decibel scale. The number of these feature vectors was reduced into four cluster centers per class using fuzzy c-means clustering. The results were reviewed by listening to the resynthesized models.

4.3 Pattern matching

For each onset in the input musical signal, the models of all classes are matched to the measured features of the mixture signal. To achieve an appropriate matching, we use fitting that minimizes the weighted least square error for class k

$$E_k = \sum_i \{M_k(i) \cdot [Y(i) - M_k(i)]^2\}, \qquad (3)$$

where $Y(i)$ is the feature vector of the mixture signal, $M_k(i)$ is the feature vector of class k, and i goes through the values of the vectors. Weighted RMS-error fitting is used, because it is able to weight errors at the bands where the model is strong. This is crucial in sound mixtures, where the presence of interfering sounds must not affect the pattern matching of a model which occupies different frequency bands.

The overall energy of each matched and scaled model is denoted by W_k. Together with the matching error, this is
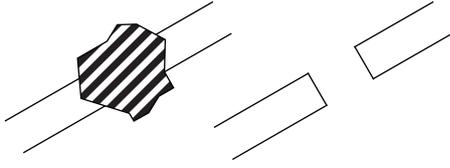
Figure 2. Examples on visual restoration.

used to calculate simple measures of goodness for models as $G_k = W_k - 0.5E_k$. The values are then normalized and scaled to yield bottom-up probabilites of sounds sources.

5 TOP-DOWN PROCESSING

Top-down processing is a vital part of human perception and auditory scene analysis [8]. Especially in music, there are simple rules that can solve otherwise ambiguous situations.

5.1 Temporal predictions

Temporal prediction produces an expectancy that a source that appears two times with time interval t_0 will occur again after another t_0 interval. This proved to be a compact and general way of modeling musical rhythms, but also applies to some environmental sounds, for example, to the sound of steps, engines and clocks.

The temporally predicted probability of a drum sound to appear at frame k is calculated as follows. We use $P_{bu}(s,k)$ to denote the bottom-up probability of a source s in time frame k. First, the locally most prominent periodicity for sound s is found by calculating the product of bottom-up probabilities of the sound s in surrounding every h^{th} frames

$$P_{pred}(s, k_0) = \frac{max}{h} \left\{ \prod_{k=-K}^{K} \{1 - w(k) \cdot [1 - P_{bu}(s, k_0 + hk)]\} \right\}$$

where $K=3$, and used windowing $w(k)=[3,7,10,0,10,7,3]/10$. In addition to finding the locally most prominent periodicity in $P_{bu}(s,k)$, this calculates the probability of the prediction.

Two additional terms must be included to calculate the effective probability of a temporal prediction. The effective total prediction probability is calculated as

$$P_{eff}(s_0, k) = P_{pred}(s_0, k) \cdot U(s_0, k), \qquad (4)$$

where

$$U(s_0, k) = \sum_{s, s \neq s_0} [P_{bu}(s, k) \cdot S_{s, s_0}] \qquad (5)$$

where S_{s, s_0} is a value reflecting the similarity of the sounds s and s_0, and thus the ability of sound s to mask s_0. Thus $U(s,k)$ in Eq. (4) represents the integral probability of other simultaneous sounds in frame k to *mask* sound s_0. Thus prediction probability for s_0 is dependent on the likelihood of it being masked. There is no sense in predicting a sound to a frame where it can be observed to be non-existent.

The need for the term $U(s,k)$ is motivated by an example from visual prediction in Fig. 2. A so-called visual restoration creates illusory continuity in the case that there is a mask which prevents from seeing a part of an object (left). On the other hand, if the break is clearly seen and there is no mask (right), no illusory continuity is created. In the same

manner, predicting s_0 at frame k is done in relation to the probability that there is a good explanation why it has not been detected in bottom-up analysis.

5.2 Usage of *a priori* probabilities

Drum classes have enough inside-class variation to make the tonal spaces of different classes overlap. This appears in the analysis of real signals so that e.g. a snare or a bass drums is recognized as tom-tom in some time frames. A pragmatic solution to this problem is to use *a priori* probabilites. For example, the expectance of a snare drum in popular music is higher than that of tom-tom. The latter often appears in a sequence of subsequent sounds, in which case the temporal prediction takes care of raising it above the detection level.

Context was used to alter the *a priori* probabilities. This was done in the sense that sources that have already been detected in a signal are given probabilistic priority over the sources that never appeared in the analyzed signal.

6 DECIDING THE NUMBER OF SIMULTANEOUS SOURCES

The matched and scaled source models are linearly subtracted from the mixture spectrum. This is done in an iterative manner, one-by-one, in descending probability order. Although stochastic signals cannot be exactly subtracted, this can be done approximately for power spectra. The criterion used to stop the iteration and use the selected N models to represent the mixture signal is done according to

$$L(N) = -\log\left(\frac{\sum_i M_N(i)Y(i)}{\sum_i Y(i)Y(i)} \right) + \alpha N, \qquad (6)$$

where $M_N(i)$ is the *sum* of the models of the selected and scaled source models $M_1,...,M_N$, and the value of α is trained so that $L(N)$ reaches its minimum for N for which the iteration should be stopped. This measure is derived from the minimum description length (MDL) principle, as proposed in [7], and was found to be a good model for the detection of the number of signals, after value α is found.

7 SIMULATION EXPERIMENTS

Drum samples for simulation experiments were collected from Roland R-8 mk II drum machine, Roland XP30 synthesizer, and from McGill Univesity Master Samples collection. The included instruments and source classes were bass drums, snares, tom-toms, open and close hi-hats, rides, and crash cymbals.

In the first experiment, bottom-up recognition of random mixtures of drum sounds was tried. A fact immediately realized was that bottom-up source recognition is very difficult for sound mixtures. This is due to the inside-class variations, and to the differences in the relative levels of simultaneous sounds. Difficulties arise when two sounds occupy common frequency bands, and one is significantly louder than the other. Recognition of isolated sounds works with few errors, recognition of two sounds is much more difficult, and for the case of three simultaneous sounds, confusion becomes a
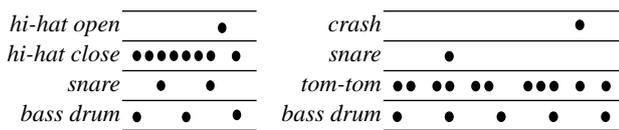
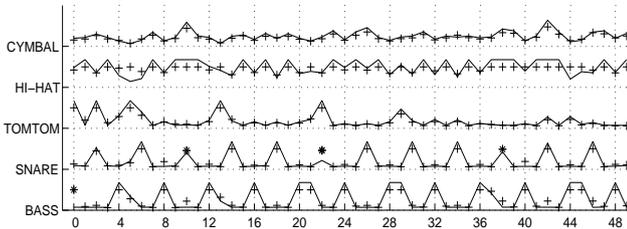**Figure 3.** Two examples of the generated drum patterns.

Figure 4. Effect of temporal predictions on source probabilities. *Line*: probabilities of different drums as a function of frame number after bottom-up processing. *Crosses*: probabilities when temporal predictions have been included.

commonplace, although Eq. 6 model generally works well.

In the second experiment, rhythmic patterns were generated by defining different patterns of drum combinations, as illustrated in Fig. 3. For each occurrence of a drum, it was independently randomized from among all the samples available from that class. A general remark was that the temporal prediction is able to restore a large number of masked sounds that remain undetected in bottom-up analysis. An example case for the leftside pattern of Fig. 3 is presented in Fig. 4. Examples of recognition errors that have been corrected by predictions are for the bass drum in frame 0 and for the snare drum in frames 10,22,38. Repeating hi-hat pattern is revealed in several frames where it was masked.

In the third experiment, drums were recognized in ten second excerpts taken from popular music recordings. As observed through listening, sinusoids plus noise modeling does rather good job in extracting the drum tracks from polyphonic music. Although some harmonic portions remain in the residual – enough to hear out most of the melody, drum sounds clearly dominate the residual. On the other hand, the crude assumption that drum sounds consist of noise was practical enough. For some drums, especially tom-toms, approximately half of the energy is periodic and thus modeled by sinusoids. However, enough spectral energy is left to the noise residual that recognition can take place.

Some more detection errors took place for real musical performances that for generated signals. The following remarks were made. First, it seems that the relative level of a signal segment must be taken into account in recognition, since the spectral content as such does not suffice to discriminate between e.g. snare and hi-hat sounds in strong interference. Second, automatic source modeling from the analyzed signals should be attempted, since in real signals, the very same acoustic sources keep repeating in different combinations. A fixed set of models could not discriminate between e.g. snare and bass drum sounds in certain signals where the difference could only be observed by comparing

these two to each other. In spite of these reservations, recognition of drum sounds for real musical recordings was possible and meaningful with the presented methods.

8 CONCLUSIONS AND FUTURE WORK

Although the current results are only preliminary, some conclusions can be made. First, spectral pattern recognition as such is not sufficient for robust bottom-up recognition of sound mixtures. Although drum sounds are excited in a transient-like manner, some time-alignment using e.g. hidden Markov modeling is needed to compensate for echos and instrumental variations. Second, the obtained results highlight the importance of top-down processing. In spite of that, the foundation of signal analysis is in reliable low-level observations, and without it no further processing will yield meaningful results. The more top-down expectationas are utilized, the more unique details are dropped, but the more robust is the overall result. Third, it was shown that musical rhythms can be well modeled by compact and generic rules.

REFERENCES

[1] Dufournet, Jouenne. (1997). "MADRAS, an intelligent assistant for noise recognition". Internoise 97, Budapest, August 1997.

[2] El-Maleh, Samouelian, Kabal. (1999). "Frame-level noise classification in mobile environments". In Proceedings of the International Conference of Acoustics, Speech and Signal Processing, ICASSP 99.

[3] Gaunard, Mubikangiey, Couvreur, Fontaine. (1998). "Automatic Classification of Environmental Noise Events by Hidden Markov Models". ICASSP 98.

[4] Couvreur, Bresler. (1996) "Dictionary-Based Decomposition of Linear Mixtures of Gaussian Processes". In proceedings of ICASSP 96.

[5] Couvreur, Bresler. (1999) "Classification of mixtures of acoustic noise signals". In Proceedigs of 8th IEEE Workshop on Signal Processing (DSP '98), Utah, Aug. 1998.

[6] Linares, Nocera, Meloni. (1997) "Mixed acoustic events classification using ICA and subspace classifier". In Proceedings of ICASSP 97.

[7] Wax, Ziskind. (1989). "Detection of the Number of Coherent Signals by the MDL Principle". *IEEE Trans. on Acoust., Speech, and Signal Processing,* Vol. 37, No 8, Aug. 1989.

[8] Ellis. (1996). "Prediction-driven computational auditory scene analysis". PhD thesis, Massachusetts Institute of Technology, 1996.

[9] Levine. (1998). "Audio representations for data compression and compressed domain processing". PhD thesis, Stanford University, 1998.

[10] Klapuri (editor). "Contributions to technical seminar on content analysis of music and audio", Tampere International Center for Signal Processing, TICSP Series #7, April 2000.

[11] Klapuri. "Sound Onset Detection by Applying Psychoacoustic Knowledge". IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 1999.