SEPARATION OF IMPULSIVE ACOUSTICAL EVENTS

Xue Wen, Yuan-Yuan Shi, Bin She

HCI Lab, Samsung Advanced Institute of Technology
{x.wen, yy.shi, bin.she}@samsung.com

ABSTRACT

Impulsive acoustical events, or impacts, compose a big family of everyday sounds. Detection and separation of these sounds is an important task of current computational auditory scene analysis. In this paper we propose a method along with an online architecture for detecting and separating acoustical impacts, which is able to find out each and every impulsive acoustical event in a continuous data flow. An energy density function on a time-frequency span is given as the result of separating each impact from the background and other overlapping impacts. Onsets are used for finding events. A prediction-based method is introduced for separating events that overlap both in time and in frequency. The method does not rely on spectral peak tracks or harmonic properties, thus is applicable to a broad class of sounds.

1. INTRODUCTION

Impulsive acoustical events, or impacts, compose a big family of everyday sounds. Understanding these sounds is an important task of current computational auditory scene analysis (CASA). This paper deals with the analysis of acoustical impacts. Given an acoustical waveform, our goal is to answer three questions: 1) whether there are any impacts present, 2) when and in which band, and 3) how their energy is distributed. By answering these questions, we separate each and every existing impact out of the continuous data flow into the form of its time-frequency (*T-F*) representation. Similar topics have been studied in the literature [1-3]. Some of these systems rely on pre-detected spectral peak tracks to form auditory elements, thus are best for harmonic or quasi-harmonic sounds, such as music and speech. Impacts on the other hand, may be harmonic or not. However, impacts have their own special properties that enable effective methods for the separation purpose.

In common sense, we say that a sound is impulsive if the stimulus on the sound source lasts a very short time, therefore may be approximated by an impulse signal. The resulted sound thus contains two stages, a very short transient stage followed by an unforced vibration stage. In the transient stage, the energy reaches the top in a very short instant, forming an onset. In the unforced vibration stage, the sound attenuates in a pattern determined by the source object only. We reduce the analysis task to two problems: onset detection and event forming. The onset detector finds each existing impact and triggers the event former; the latter then tells how this impact progresses. These two parts are integrated in a filterbank based online parallel architecture, which is to be described in details.

Section 2 describes the system structure. Sections 3 and 4 explain the onset detector and event former. Results are given in section 5. Section 6 concludes our discussion.

2. THE SYSTEM

Our system takes as input a waveform audio stream, and outputs a T-F representation for each detected impact. This representation includes a time-frequency span S and a positive energy density function $E(t, f)$ defined on S. Here a time-frequency span is defined as a subset of the 2D space $T \times F$, where T denotes time and F denotes frequency. In our problem both T and F are discrete and all events are of limited band and finite duration, so S is always finite (except the zero event, see section 4).

The whole system comprises a universal controller (*UC*) and a bunch of working units (*scouts*). UC advances at a constant interval and serves as a synchronizer that drives each scout at proper instants, which allows the choice of individual step length and starting phase for every scout. The scouts operate in a parallel manner. Each scout is associated with a subband, working with its local data (e.g. band-limited instantaneous power, etc.) provided by UC. A scout is expected to master whatever happens in its band. It records statistical and structural properties of local data, detects local onsets (called *onset components*), and tracks local events (called *event components*). The UC then combines the results from the scouts and provides necessary feedbacks to direct them. UC and scouts communicate each other through messages. After each progress step of UC, UC and scouts process their messages iteratively, until all messages are handled.

Event components (*ECs*) are the basic units in the system. An EC is the restriction of an event on a subband

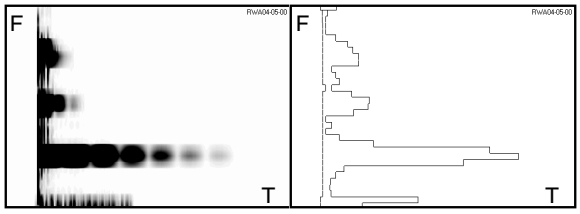

Figure 1. Spectrogram (left) and T-F span of an impact

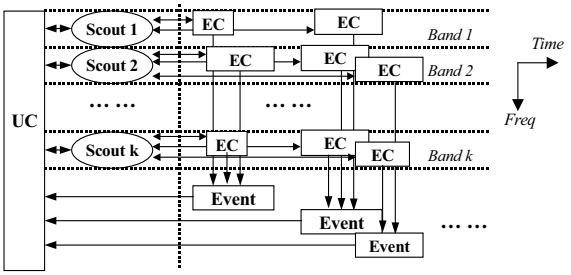

Figure 2. Working modules of the system

at frequency f_k associated with a scout *k*. The T-F span of the EC is restricted to $T \times \{f_k\}$. An EC is formed when the scout believes a new event has happened which causes power increase in its band. For impacts, since the stimulus is zero after the transient, an EC does not likely to make further contribution once its power has fallen below a floor level. Hence we assume that the T-F span of an EC of an impact takes the form of $\{(t, f_k) \mid t_i \le t \le t_j\}$.

ECs are grouped into events according to the common-onset cue, i.e. components of the same event tend to start at the same time [4]. Therefore an event is composed of a bunch of ECs starting at almost the same time, each lasting a finite duration. Figure 1 shows the spectrogram and the T-F span of a single impact. By careful arrangement of working procedures, our system is able to separate impacts from continuous data, which is to be explained in the following sections.

Figure 2 describes the relationship among the four elements in our system, namely the universal controller, scouts, event components and events.

3. ONSET DETECTION

Onsets play a key role in finding impacts in sounds. Each impact features an onset at the beginning. By detecting all the onsets, we locate all possible beginnings of impacts. Methods for onset detection have been studied by many [5-7]. Filterbank based approaches are favored for their accuracy and robustness. We use a similar method in our system that detects onset components in each subband. Very close onset components are combined into a global onset by UC according to the common-onset cue.

For detection of onset components, a scout applies an exponential onset filter to its local logarithmic amplitude

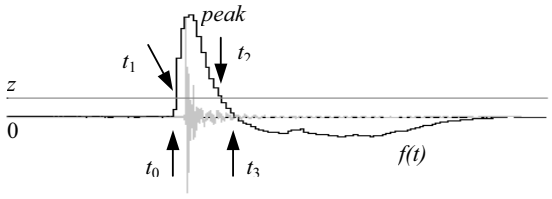

Figure 3. Endpoints of an onset component

track. The impulse response of this filter is given by

$$h(t) = Ae^{t/T_1} - Be^{t/T_2}$$
$$where \quad A = 1 - e^{-1/T_1}, B = 1 - e^{-1/T_2}, T_1 < T_2. \qquad (1)$$

The filter emphasizes fast changes in its input signal and suppresses slow variations. Notice that $h(t)$ has zero mean. Figure 3 shows the output of the onset filter $f(t)$ in the vicinity of an onset component. At each positive peak of $f(t)$, the scout examines the peak value and some statistical records of $f(t)$ to decide if the peak indicates an onset component.

For grouping onset components, we locate 4 critical points on each detected local onset (Fig.3): t_0 and t_3 at the zeroes of $f(t)$ closest to the main peak, as well as t_1 and t_2 at the zeroes of $f(t) - z$ closest to the main peak, where z is some small value above zero. We call the interval (t_0, t_3) outer domain of the onset component, and interval (t_1, t_2) inner domain of it. We think that two onset components are "very close" if and only if the inner domain of either one overlaps the outer domain of the other. Very close onset components are combined into groups. A group accepts as its new member any onset component that is very close to one of its already existing members. In the end when a group stops growing, it defines a *global onset*, which is useful for global onset notifications, to be explained in section 4.1.

4. EVENT TRACKING

After finding the onset of an event, a couple of ECs are created to track the behavior of the event, each in one subband. In each band no more than one EC may exist for the same event. An EC is controlled by the scout of its band. One scout may manage several living ECs at the same time, allowing events to have overlapping T-F spans.

4.1 Birth of event components

The assumption that every event starts from an onset, along with the common-onset cue, implies that every EC starts from an onset. This happens in two ways, onset-triggered EC or notice-triggered EC. An onset-triggered EC starts from an onset component. It is created by a scout when the latter detects the onset component. Notice-triggered ECs are related to the fact that an event does not

always make distinct onsets in all of its subbands. When a global onset is defined, the UC sends a notification to each scout that has not declared onset components during the onset. The notified scout then checks to see if this onset affects its subband. If it does, the scout creates a notice-triggered EC. After this step, all ECs associated with the onset are known. These ECs compose an *event* associated with the given onset.

Each scout utilizes a local predictor for detecting notice-triggered ECs. The local predictor predicts future data or certain properties of future data following current pattern. For our current purpose, recent local data before the global onset is used to predict the signal power during the onset. If the observed data shows an increase in power stronger than expected, a notice-triggered EC is declared.

4.2 Tracking event components

For each scout, the global task of event separation is restricted within its own subband, thus reduced to two problems. One is finding the end of each EC in its subband, giving the restricted T-F span; the other is estimating the power of each EC at each instant during its life, giving the restricted energy density function. The solution to the first problem is straightforward: an EC dies when its power drops below a floor level. An EC's instantaneous power is derived from the local data of its scout, e.g. the total power in the subband. When there is only one living EC in the band, its power equals the total power. When there are multiple living ECs from several events, the scout must distribute the total power among them. This is what the second problem deals with.

Like the scout, each EC has a local predictor of its own, which estimates its future power following current pattern. When a new EC is created, it's given a *masking* status, indicating it dominates its subband. All other living ECs in the same subband are *masked*. The instantaneous power of a masked EC is estimated by its local predictor. These predicted values are then subtracted from the total band-limited power. The residue gives an estimation of the instantaneous power of the masking EC. Usually at the beginning of a new EC, the residues are much larger than the predicted powers, thus giving a fairly reliable estimation. If at any time the power of the masking EC falls below one of the masked ECs', the masking EC loses its masking status and becomes masked, meanwhile the strongest masked EC gets hold of the masking status. The local predictor of an EC may be trained or updated only when the EC is masking and dominates the total power.

A masked EC dies when its predicted power falls below a local floor level. A masking EC dies when its estimated power falls below the same local floor level. An event dies when all of its living components are masked and the sum of their powers falls below a global floor level. Some restrictions are imposed on the predictors to ensure that all events end up properly.

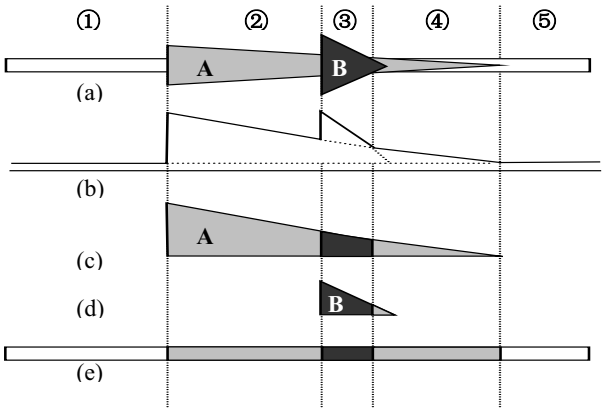

Figure 4. Tracking event components

4.3 Zero event

We define a special event called *zero* to represent the general background. The T-F span of the zero event is the whole $T \times F$ space, i.e. the zero event is born with the system, spans all frequencies, and never dies. The zero event has one and only one component (zero EC) in every subband. Operations with the zero event are treated almost the same way as acoustical events. A zero EC is responsible for monitoring the local signal when it's the only living EC in its subband, keeping records of statistical measures of the acoustical background. The zero event and its components provide global and local floor levels. A zero EC always stands on the local floor level. A zero EC is masked whenever there is another EC in its subband. Otherwise it is the masking EC.

The local predictor of a zero EC often takes the form of a constant $y(t) = y(t_0)$. To ensure that the zero event is able to follow the variations of the background, the zero ECs are allowed to directly utilize local data for updating their local predictors even if they're masked. However, the update rate of a zero EC in this way is kept very slow, ensuring a stable estimation of the acoustical background. When a zero EC gets masking status, the update rate is much faster, and its local predictor will converge to the current background very quickly. This is especially preferred when the "general background" includes loud but slow varying sounds which do not have onsets.

Figure 4 illustrates what EC tracking is all about. Suppose there are two ECs A and B, B overlapping the mid part of A (Fig.4a). Fig.4b depicts the local data (in solid curve). Since 2 onset components are found, the scout analyzes this piece of data into 3 ECs, including the zero EC (Fig. 4c – 4e). A is masked by B in section 3 and masks B in section 4. The zero EC is masked by A in sections 2 and 4, and by B in section3. When masked, an EC goes on by prediction (dotted curves in Fig.4b).

4.4 Event forming

We have seen from section 4.1 that an event already knows all its components in its early stage. These ECs together form the event. Its T-F span is the union of all its components' T-F spans. Its energy density at point (t, f_k) is the instantaneous power of its component in subband k at time t.

Now we have separated every event with an onset, the only thing left is to verify whether it is an impact. We simply check the track of its logarithmic power to see if it conforms to some common pattern of impacts [8]. For example, we use a function $y(t) = 1 - t^{\lambda}$ to approximate the normalized logarithmic power. An ideal impact has $\lambda = 1$. For most impacts the exponent λ is smaller than 3 and the approximation error is not too large.

5. EXAMPLES

Since the representation of each event is given as an energy density function in a T-F span, we use the spectrogram the illustrate the results of our separator.

An example of artificially mixed sounds is given in Figure 5. Two very clean samples, a sound of beating china and another of hand clapping, are added to a background of Gaussian white noise. The original spectrogram is given in Fig.5a. The separated zero event (noise), beating china event and hand clapping event are given in 5b through 5d respectively.

An example of door knocks with rattles is given in Figure 6. Fig.6a shows the incoming signal. Six events are visible from Fig.6a, including two knocks and 4 rattles. This sound is decomposed into 7 normal events (Fig.6b-6h) and the zero event (Fig.6i). Events in 6d, 6f, 6g and 6h are those rattles seen in 6a, and the event in 6e is just another impact too weak to be visible here. The rattles are successfully removed from the door knocks, while "stealing" some low frequency energy that does not belong to them.

6. CONCLUSION

In this paper we have described a system for separating impulsive acoustical events from other sounds. Onsets are used for finding impacts, and a prediction based method is developed for distributing energy among event components. Our current system relies only on a few very common assumptions for impact detection, thus is capable to deal with the general class of impacts. We have also proposed an online architecture for implementation of stream processing, featuring supervised cooperation of multiple working units. Such a system structure offers the flexibility to incorporate advanced knowledge or filtering techniques into current implementation, which is the key to further enhance our system in future.

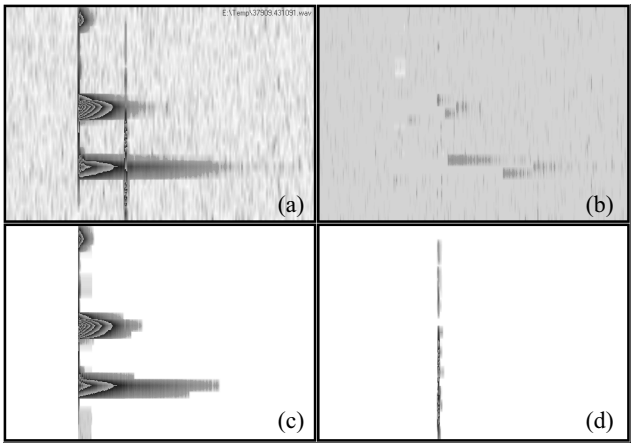

Figure 5. Beating china, hand clap, and white noise

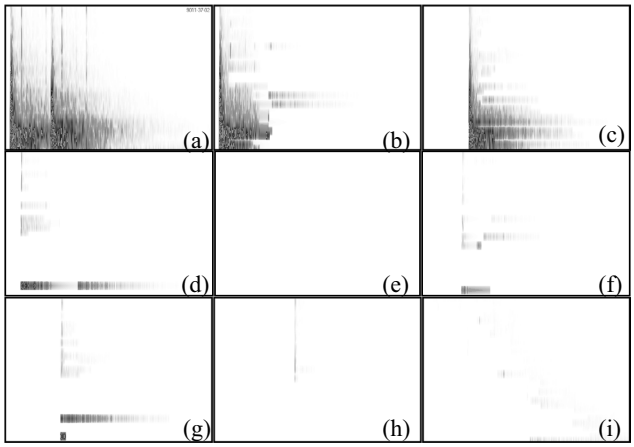

Figure 6. Door knocks with rattles

7. REFERENCES

[1] G. J. Brown and M. Cooke, "Computational auditory scene analysis", Computer Speech and Language 8, p.297-336, 1994.

[2] D. Ellis, "Prediction-driven computational auditory scene analysis", PhD thesis, MIT, 1996.

[3] D. Godsmark and G. J. Brown, "A blackboard architecture for computational auditory scene analysis", Speech Communication 27, p.351-366, 1999.

[4] A. S. Bregman, *Auditory Scene Analysis*, MIT Press, 1990.

[5] L. S. Smith, "Onset-based sound segmentation", in D. S. Touretzky, M. C. Mozer and M. E. Hasselmo, editors, Advances in Neural Information Processing Systems 8, p.729-735, MIT Press 1996.

[6] A. Klapuri, "Sound onset detection by applying psycho-acoustic knowledge", in IEEE ICASSP 1999.

[7] C. Duxbury, M. Sandler and M. Davies, "A hybrid approach to musical note onset detection", 5th Int. Conference on Digital Audio Effects, 2002.

[8] K. Hiyane and J. Iio, "Non-speech sound recognition with microphone array", International Workshop on Hands-free Speech Communication, 2001.