TRAP: TRAnsient Presence detection exploiting Continuous Brightness Estimation (CoBE)

G. Presti[1], D.A. Mauro[2], and G. Haus[1]
[1]Laboratorio di Informatica Musicale (LIM), Dipartimento di Informatica (DI), Universitá degli Studi di Milano
Via Comelico 39, 20135 Milan, Italy
giorgio.presti@unimi.it
[2]Iuav University of Venice, Department of Architercture and Arts
Dorsoduro 2196, 30123 Venice, Italy
dmauro@iuav.it

ABSTRACT

A descriptor of features' modulation, useful in classification tasks and real time analysis, is proposed. This descriptor is computed in the time domain, ensuring fast computation speed and optimal temporal resolution.

In this work we take into account amplitude envelope as inspected feature, so the outcome of this process can be useful to gain information about the input' energy modulation and can be exploited to detect transients presence in audio segments.

The proposed algorithm relays on an adaptation of *Continuous Brightness Estimation* (CoBE).

1. INTRODUCTION

In the context of Music Information Retrieval (MIR) a naive approach for tracking the amount and nature of modulations can be achieved measuring the standard deviation inside a window of the underlying modulated feature, but this method can only quantify the amount of the modulation, with no information about the shape or frequency. For example, the envelope of a rhythmic pattern may have the same standard deviation of a sustained signal with lot of amplitude modulation, while the pitch of an arpeggio may have the same standard deviation of the pitch of a frequency modulated tone.

A possible alternative to this tracking technique might be to estimate the high frequency content of the time series of the feature under consideration. In such a way the outcome is a measure that only depends on the shape, frequency, and amount of the modulation (i.e. how *rich*, *crispy*, or *jagged* is the feature).

Techniques which can promptly respond to this needs providing good approximations with fast computing time are welcome in real-time applications or when analysing very large datasets. This approach is then motivated by its implementation in the temporal domain, low computational cost and parametrizable temporal resolution.

In this paper we present a case study where we approach this issues using CoBE [1] as main algorithm to measure the presence of transients in audio segments. The rationale for this technique came from trying to automatically classify sonification examples (to appear in [2]), where a feature useful to distinguish between continuous sounds and discrete events can be exploited.

2. THE COBE BEHAVIOUR

CoBE can be interpreted as the ratio of high frequencies in a signal. It is computed comparing the energy of a filtered version of the input with the original one. This approach matches in some way the definitions of Brightness given by [3], [4], [5] and [6] but instead of being computed in frequency domain, it is computed in the time domain, enabling some interesting properties, besides performance improvements. For example, exploiting the inverse transfer function of the magnitude of the filter used, it is possible to infer the frequency of a sine wave having the same CoBE value of the input signal, namely the *Equivalent Brightness Frequency* (EBF), shown in Eqn. 1 (for further details see [1]).

$$f = (\frac{f_s}{\pi}) \arcsin(\frac{B}{2}) \tag{1}$$

Where f_s is the sampling frequency and B is the CoBE Brightness value.

2.1 Implementation

With respect to the implementation previously described in [1], a slightly different implementation is proposed here and detailed in Fig. 1. It presents some advantages in terms of stability and it is more readable while preserving the same output. The source code written in Matlab language is the following:

```
function [B,EBF] = CoBE(X,fs,EnvFun,
    varargin)
        % Amplitude envelope E
        E   = EnvFun(X, varargin{:});
        % Filtered version dX
        dX  = diff( [0; X] );
        % dX amplitude envelope Ed
        Ed  = EnvFun(dX, varargin{:});
```

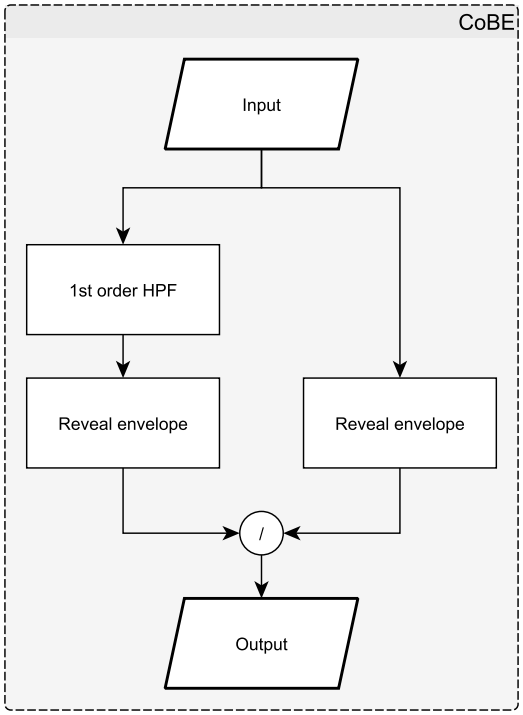

Figure 1. Diagram of the CoBE algorithm, intended as the ratio of high frequencies that constitutes the signal.

```
        % Brightness as Ed / E
        B    = Ed./E;
        % Equivalent Brightness Freq.
    EBF = (fs/pi).*asin(B./2);
end;
```

In contrast to the *High Frequency Content* feature described in [7], the behaviour of CoBE is independent from the signal level, and for monophonic sine waves it is also independent from signal filtering [1]. However, as can be clearly pointed out from both Fig. 1 and source code, it strongly depends on the envelope follower algorithm.

2.2 Behaviour with different envelope followers

Four different envelope followers has been analysed:

- VU-meter style follower, with zero attack and slow release (*Vu*);
- RMS of a moving window, lowpass filtered in order to remove residual ripples (*RMS*);
- Local maxima inside a moving window (*Max*);
- Classic Rectify and Filter approach (*RF*) [2].

As can be seen in Fig. 2, the very-low frequency band may cause issues with the first three algorithms, while high frequencies may be tracked incorrectly by *Vu* and, less significantly, by *Max* and *RF*. In Fig. 3 it is possible to notice

[1] In such a case filtering can be considered as a simple *delay and scale* function

[2] Used in MIRToolbox and implemented according to [8].

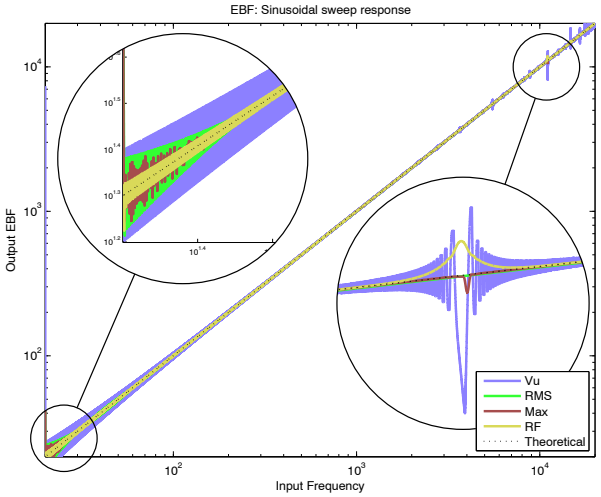

Figure 2. EBF Sweep response using different envelope followers. Differences in the top and bottom ends of the spectrum are magnified.

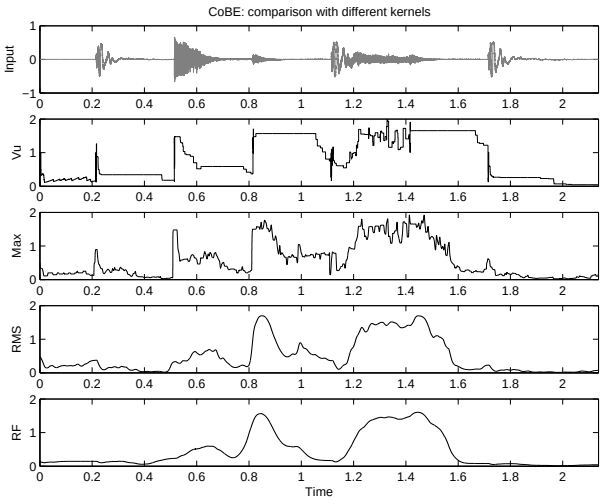

Figure 3. CoBE of a drum sample estimated with different envelope followers.

that the different followers are characterized by an increasing level of smoothness, with *Vu* which behaves in a peculiar way, holding previous CoBE values during release phase. This behaviour may be useful for percussive sounds analysis or transient detection, since it holds the brightness of the attack phase and ignores the release phase. Nevertheless, for all other purposes, the use of *Vu* as envelope follower for CoBE is discouraged. As regards *Max*, it shows a very sharp function, which may be ideal for some application, but non for general purpose. The same can be stated for *RF* for its extreme smoothness.

In conclusion, *RMS* and *RF* seems to be the best choices for general purpose CoBE. In this context *RMS* is used, since it is easy to implement both in analogue and digital domain and, most important, it is related to a physical property (the *effective value* [3]) which applies to any signal.

3. TRANSIENT PRESENCE DETECTION BY ENERGY-ENVELOPE BRIGHTNESS (TRAP)

The main idea is to measure the brightness of the amplitude envelope using the CoBE algorithm. Applying CoBE to the signal envelope, instead of the signal itself, should reveal that continuous amplitude envelopes (where sound is likely to be a smooth modulation of features) will produce a low CoBE value, while crispy amplitude envelopes (corresponding to strong amplitude modulations or numerous transients) will present a high CoBE value. Two examples of TRAP signal behaviour are shown in Fig. 4 and Fig. 5.

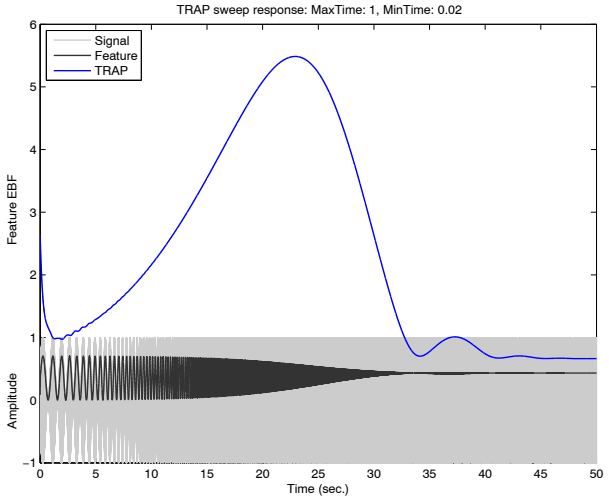

Figure 4. TRAP signal for an input created using a low frequency sweep as modulation for a 1kHz sine wave.

3.1 Implementation and tuning

We choose *RMS* also as the follower that will produce the main envelope signal, basically for the same reason we choose *RMS* as follower inside CoBE. To avoid confusion we will refer to the signal envelope as the *feature*, and to

[3] RMS. The value of the direct current that would produce the same power dissipation in a resistive load.

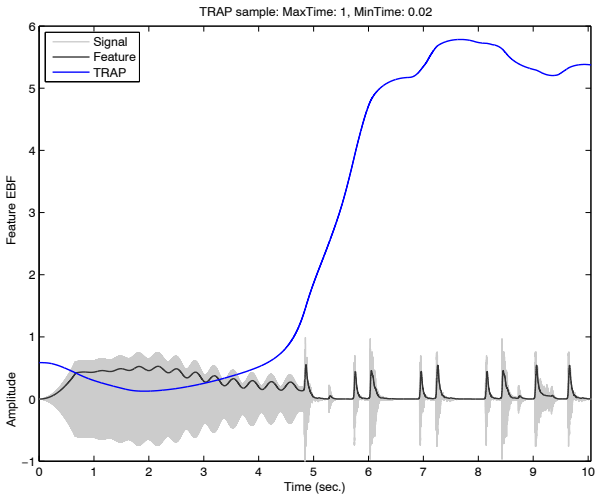

Figure 5. Output of the TRAP algorithm. The input file has been created to show different behaviours.

the algorithm used inside CoBE as the *kernel*. The window size of the feature follower (called *minTime* in the code) basically defines what is the minimum distance between sound events or, in other words, how smooth has to be the envelope that will be fed to the CoBE algorithm. This feature is then down-sampled to spare computing power and then fed into CoBE. The window size of the kernel (called *maxTime*) defines the overall smoothness of the output: smaller windows accentuates short therm variations of the envelope, while larger windows will generate smooth outputs. Finally, to make the algorithm independent from sampling rate, we consider EBF instead of the mere CoBE value.

The following code is an example of how this can be implemented in Matlab, the algorithm is also represented in Fig. 6.

```
function G =TRAP(X,maxTime,minTime,fs)
        % Kernel and Feature functions
        Feature = @RMSEnvelope;
        Kernel  = @RMSEnvelope;
        % Time to samples conversion
        minTime = floor(minTime*sr);
        k = 100; sre = fs/k;
        maxTime = floor(maxTime*sre);
        % Feature extraction
        E = Feature(X,minTime);
        E = downsample(E,k);
        % Feature EBF extraction
        [~,G] = CoBE(E,fs,Kernel,
            maxTime);
end
```

Lowering *minTime* too much makes the algorithm fitting the waveform instead of the envelope, thus introducing noise. This noise increase considerably the envelope brightness and EBF. On the other hand, higher values of *minTime* may ends in ignoring transients or short burst of signal. We empirically found that a value between 0.0125 and 0.0250 seconds may be suitable for most situation. Best results were

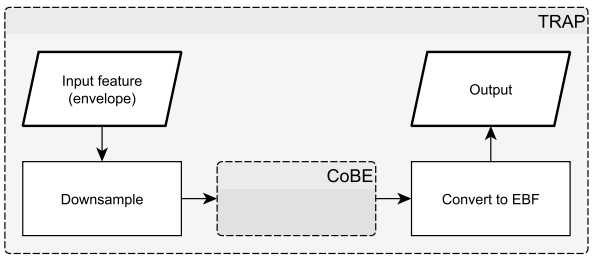

Figure 6. Diagram of the TRAP algorithm.

obtained with $minTime = 0.02$. This window size can detect variations up to 50 Hz, while higher frequencies will be smoothed out and considered as a continue envelope.

Please note that only those results obtained with the same *minTime* are fully comparable, for this reason we suggest to set $minTime = 0.02$ as conventional starting point [4]. For comparison different values of *minTime* are shown in Fig. 7.

For what concerns *maxTime*, high values average out the whole signal, while low values (*maxTime*<0.5) fit the signal more precisely, magnifying the sharp amplitude modulations of the signal. In this case, a default value of 1 second may fit most of the scenarios. The behaviour obtained with different *maxTime* values is shown in Fig. 8.

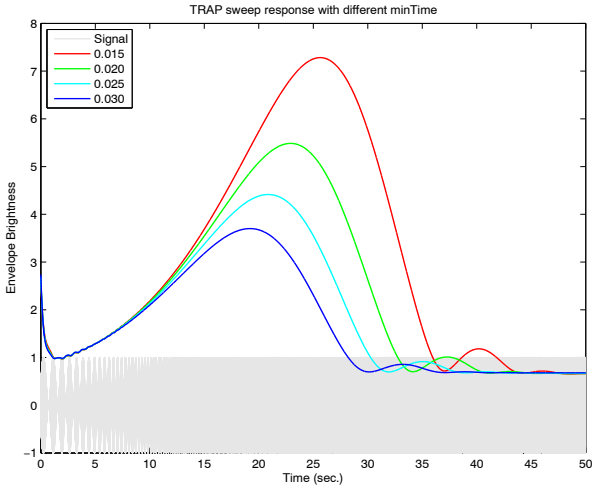

Figure 7. Same signal of Fig. 4 analysed with different *minTime* values.

4. EVALUATION AND TESTING

To inspect the information redundancy carried by the TRAP signal, correlation analysis with other features is performed.

Since the richness of the envelope may depends on the presence of transients, we took into account spectral descriptors normally used in onset detection tasks [5] ([9], [10], [11]), besides common time domain energy descriptors (listed below).

[4] This value is the same default value provided by MIRToolbox as time constant for *mirenvelope*.

[5] Please note that even if correlated onsets and transients are not exactly the same.

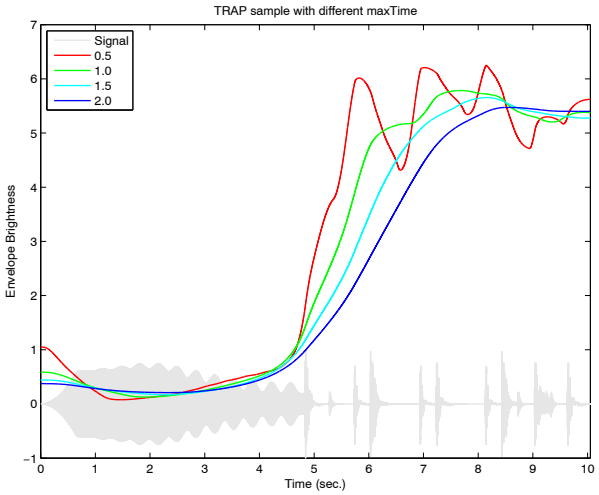

Figure 8. Same signal of Fig. 5 analysed with different *maxTime* values.

Chosen features can be grouped in two main categories: *Monodimensional time-varying* features, each represented by a single time series, and *general descriptors*, where each feature is represented with a scalar value. Time series are then collapsed to scalar values by taking the median value and interquartile range (IQR); as pointed out by [12]. Those measures are more stable and resilient to silence segments and outliers than mean and standard deviation.

Chosen features are shown in Table. 1.

Type	**Name**	**Reference**
General	Pulse clarity	[13]
	Event density	[3]
	Low energy	[3]
	Modulation frequency	[12]
	Modulation amount	[12]
Time varying	TRAP	
	CoBE EBF	[1]
	RMS	
	Peak	
	Crest factor	
	Attack leap	[3]
	Spectral Flux	[3]
	Centroid	[3]
	Flatness	[3]
	Hi-Frequency Content (HFC)	[7], [15]

Table 1. Extracted features

The sound samples are divided into 5 groups:

- 10 monophonic instruments taken from the MUMS database [16], characterized by a pizzicato or percussive excitation;
- 10 monophonic bowed or wind instruments taken from the MUMS database;

- 10 segments of orchestral music;
- 10 segments of POP music taken randomly within POP sub-genres;
- 10 voice recordings containing various examples (singing and spoken, males and females).

Samples from the MUMS database are made of single notes interleaved with silence. This files were manually edited to make silence between notes constant to 100 ms. To obtain a robust correlation analysis we decided to use Spearman rank correlation, instead of the typical linear Pearson correlation as proposed in [12]. For the feature extraction we used MIRToolbox [3] and TimbreToolbox [12]. Results are shown in Table 2 and Table 3. Fig. 9 shows a dendrogram built using *1 - ABS(correlation)* as distance to try to reveal a hierarchy of the extracted features.

Feature	**Correlation**	p**-value**
Modulation amount	0,76	<0,05
Event density	0,60	<0,05
Centroid (IQR)	0,55	<0,05
CoBE EBF (IQR)	0,54	<0,05
TRAP (IQR)	0,52	<0,05
Flatness (IQR)	0,50	<0,05
Attack leap (med)	0,45	<0,05
Flatness (med)	0,45	<0,05
HFC (med)	-0,43	<0,05
Spectral flux (med)	0,41	<0,05
Centroid (med)	0,37	<0,05
Peak (IQR)	0,34	<0,05
Crest factor (IQR)	0,34	<0,05
RMS (IQR)	0,33	<0,05
Low energy	0,31	<0,05
Decay	-0,31	<0,05

Table 2. TRAP median, correlation with other features and p-value (sorted by decreasing absolute correlation, only significative values are reported.)

Feature	**Correlation**	p**-value**
Modulation amount	0,55	<0,05
TRAP (med)	0,52	<0,05
Low energy	0,40	<0,05
Flatness (IQR)	0,40	<0,05
HFC (med)	-0,38	<0,05
HFC (IQR)	-0,34	<0,05
Centroid (IQR)	0,32	<0,05
RMS (med)	-0,29	<0,05

Table 3. TRAP IQR correlation with other features and p-value (sorted by decreasing absolute correlation, only significative values are reported.)

As shown by the dendrogram in Fig. 9, the "distance" between TRAP and other time varying features is low thus implying that it provides different information. Table 2 and Table 3 show that correlation, when present, is significant, in particular the features that seems to be more related to

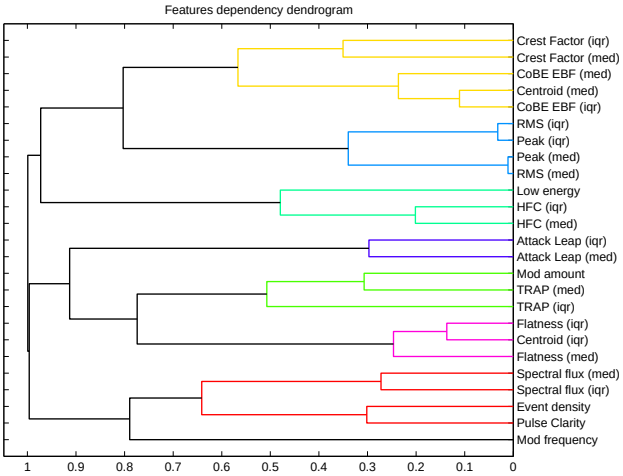

Figure 9. The dendrogram extracted from the correlation data shows the hierarchy of the investigated features.

TRAP are: *Energy Modulation Amount*, *Event Density* and *Centroid IQR*.

Energy Modulation Amount and *Event Density* are exactly the features we expected to see as the most correlated, since they affect the energy envelope (the former more explicitly than the latter). Also the *Spectral Centroid interquartile range*, with other spectrum-dependent interquartiles, are correlated with TRAP median. This can be explained by the fact that changes in timbre may correspond to different sound events and variations in the energy envelope, and during transients variations of spectral features are commonly found.

Time consumption analysis has been made comparing TRAP computing time with some of the most correlated features: *Energy Modulation Amount*, *Event Density*, *Flatness*, *Centroid* and *Low Energy*.

The results are shown in Fig. 10 and Table 4 and prove the implementation to be useful in terms of computational time, especially in the case of *Event Density* and *Energy Modulation Amount*.

Feature	**Processing time**	**Ratio**
Event density	48,89 ms	5,88
Energy Modulation	22,05 ms	2,65
Flatness	15,74 ms	1,89
Centroid	14,26 ms	1,72
TRAP	08,31 ms	1,00 (ref)

Table 4. Median time necessary to compute one second of audio and ratio with TRAP time. Data computed from those in Fig. 10

Finally, in Fig. 11, we scattered the sound samples to show the distribution of TRAP.

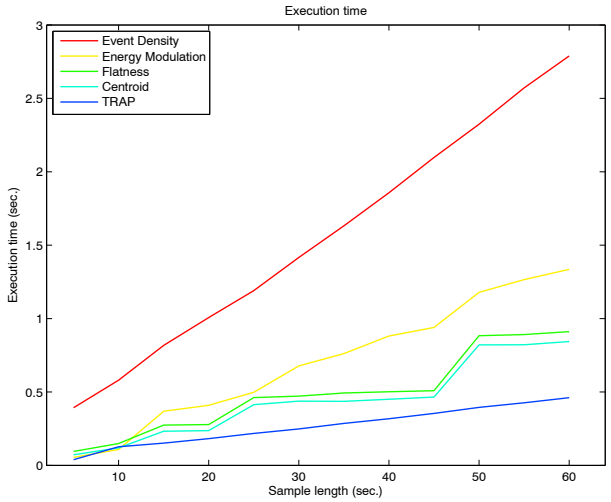

Figure 10. Average execution time for audio samples of different length. The test run on a common laptop computer with an Intel I5 processor with a clock frequency of 1.7 GHz

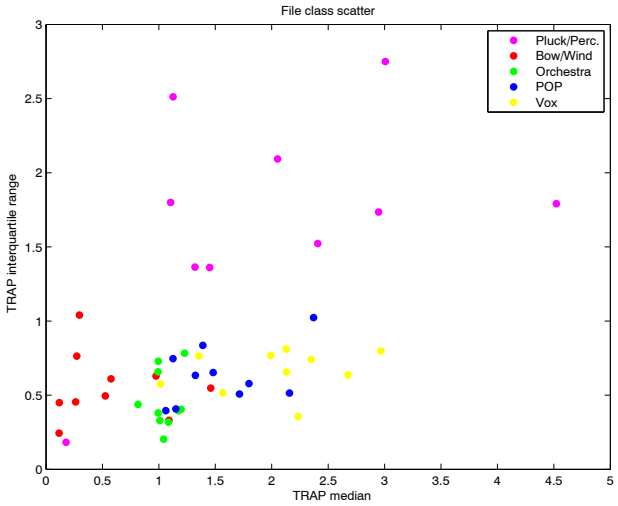

Figure 11. TRAP median and IQR used to plot the dataset.

5. CONCLUSIONS & FUTURE WORKS

A descriptor for features' shape has been proposed. In particular this method has been applied to energy envelope and has been proved as an indicator for transient presence and energy modulations.

TRAP has been used to distinguish between continuous signals and discrete acoustic events in [2]. With appropriate thresholding, it is useful to describe the presence of transient in segments of sounds.

It might also serve to create automatic dynamics processors that change their behaviour according to the content of the signal. Another possibility is to apply this very same method not to energy envelope but to other features (e.g. the pitch contour).

In order to better explain results an experimental set-up for testing perceptual correlations is advised: simple signals (amplitude modulated noise/sine waves) clustered by this feature and by humans can be compared. Finally, to overcome the possible limitation of the envelope follower method as presented in Section 2.2 a comparison of different approach can be taken into consideration.

6. REFERENCES

[1] G. Presti and D. Mauro, "Continuous brightness estimation (cobe): Implementation and its possible applications," in *10th International Symposium on Computer Music Multidisciplinary Research (CMMR)*. Laboratoire de Mécanique et d'Acoustique, 2013, pp. 967–974.

[2] L. A. Ludovico and G. Presti, "The sonification space: a reference system for sonification tasks," *Accepted for Journal on Human Computer Studies: special issue on Data sonification and sound design in interactive systems*, 2015.

[3] O. Lartillot and P. Toiviainen, "Mir in matlab (ii): A toolbox for musical feature extraction from audio," in *Proceedings of the 8th International Conference on Music Information Retrieval*, Vienna, Austria, September 23-27 2007, pp. 127–130.

[4] P. N. Juslin, "Cue utilization in communication of emotion in music performance: relating performance to perception." *Journal of Experimental Psychology: Human perception and performance*, vol. 26, no. 6, p. 1797, 2000.

[5] E. Schubert, J. Wolfe, and A. Tarnopolsky, "Spectral centroid and timbre in complex, multiple instrumental textures," in *Proceedings of the international conference on music perception and cognition, North Western University, Illinois*, 2004, pp. 112–116.

[6] P. Laukka, P. Juslin, and R. Bresin, "A dimensional approach to vocal expression of emotion," *Cognition & Emotion*, vol. 19, no. 5, pp. 633–653, 2005.

[7] P. Masri and A. Bateman, "Improved modelling of attack transients in music analysis-resynthesis," in *Pro-*

ceedings of the International Computer Music Conference. Citeseer, 1996, pp. 100–103.

[8] A. P. Klapuri, A. J. Eronen, and J. T. Astola, "Analysis of the meter of acoustic musical signals," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 1, pp. 342–355, 2006.

[9] S. Dixon, "Onset detection revisited," in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-06)*. Citeseer, 2006, pp. 133–137.

[10] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 1035–1047, 2005.

[11] N. Collins, "A comparison of sound onset detection algorithms with emphasis on psychoacoustically motivated detection functions," in *Audio Engineering Society Convention 118*. Audio Engineering Society, 2005.

[12] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," *The Journal of the Acoustical Society of America*, vol. 130, no. 5, pp. 2902–2916, 2011.

[13] O. Lartillot, T. Eerola, P. Toiviainen, and J. Fornari, "Multi-feature modeling of pulse clarity: Design, validation and optimization." in *ISMIR*. Citeseer, 2008, pp. 521–526.

[14] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," 2004.

[15] K. Jensen and T. H. Andersen, "Real-time beat estimationusing feature extraction," in *Computer Music Modeling and Retrieval*. Springer, 2004, pp. 13–22.

[16] F. J. Opolko and J. Wapnick, *MUMS: McGill University Master Samples*. McGill University, Faculty of Music, 1989.