X-MICKS – INTERACTIVE CONTENT BASED REAL-TIME AUDIO PROCESSING

Norbert Schnell, Diemo Schwarz, Remy Müller

Real Time Applications
IRCAM, Paris, France
{Norbert.Schnell|Diemo.Schwarz|Remy.Muller}@ircam.fr

ABSTRACT

In this article we present the real-time audio plug-in *X-Micks*, an audio processing application allowing for remixing and hybridization of two beat-synchronized audio streams, which provides user interaction based on the extraction and visual rendering of information from the two real-time audio streams. In the current version, the plug-in uses the beat grid information provided by the plug-in host and a real-time estimation of energy in chosen frequency bands to construct an interactive matrix representation allowing for intuitive and efficient user interaction based on familiar representations such as the sonogram and the step sequencer. After trying to formulate the constitutional qualities of a rising new generation of audio processing tools of which we claim *X-Micks* being an exemplary specimen, the article gives an overview over the application's interface, functionalities and implementaion.

1. INTRODUCTION

With the availability of robust real-time analysis/resynthesis methods and powerful music information retrieval methods a new generation of interactive real-time processing tools becomes possible. Two additional encouraging factors for these development have to be mentioned, the availability of exchange file formats for temporal music information retrieval data and – most important for the development of all successive generations of interactive real-time processing tools – the availability of fast machines with efficient real-time graphic display providing reactive user interaction.

This new generation of tools goes beyond the friendly cohabitation of real-time display of audio parameters with graphical user interface elements for the manipulation of parameters of real-time audio processing algorithms, inherited from conventional mixing consoles and other audio devices. We permit ourselves to baptize this new generation of tools "Interactive Real-Time Content Based Audio Processing".

The *X-Micks* application presented in this article is a modest implementation of such a tool satisfying the criteria one could list to define its genre:

- real-time rendering of the interaction interface according to the audio content
- robustness and intuitiveness of the chosen representation in terms of its pertinence for the given content
- integration of offline analysis with real-time analysis and (re-)synthesis

The first point can be seen as the constitutional kernel of this list, while the second defines the criteria of "pertinence" inviting for creativity as well as for further refinement of the definition and discussions concerning the pertinent qualities of different classes of audio content such as speech, environmental sounds and music as well as different styles of music as known from from other research topics in music technology and information retrieval as an interdisciplinary field.

Even if the third point can be considered as optional in terms of supporting our definition, it represents an important challenge for the development of future applications fully taking advantage of the increasing availability of musically pertinent, although inherently non real-time, music information retrieval algorithms within the context of real-time audio processing tools.

2. THE *MA-TRICKS*

The *X-Micks* interface mainly consists a $12 \times B$ matrix representing the distribution of energy in twelve frequency bands and B beats (or *sub*-beats – typically 16th notes) of one bar of music. For music based on a $4/4$ meter, a 12×16 matrix is displayed.

The representation can be seen as a reduced sonogram obtained by coarse discretisation of the time axis to (sub-)beats and the frequency axis to perceptive frequency bands and was first developed for a simple application called *Ma-Tricks*.

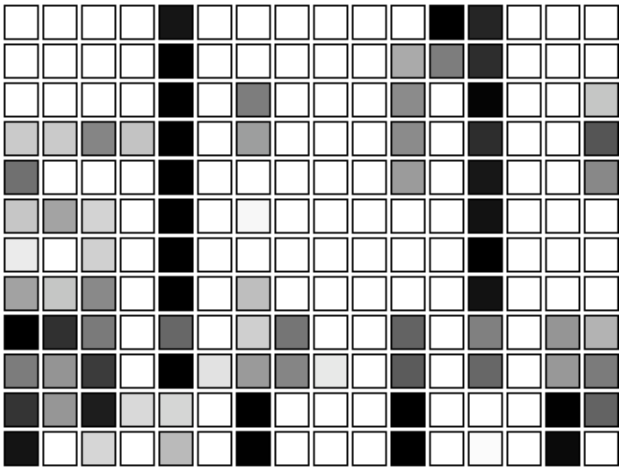

Figure 1: *Representation of a reduced sonogram within the Ma-Tricks application.*

Figure 1 shows a screenshot of the *Ma-Tricks* sonogram of one bar – sixteen sixteenth beats – of the chorus of the song *"Die Another Day"* by Madonna. One can easily distinguish the bass drum and snare drum beats, without listening to the music. Watching the representation while listening to the music permits to easily associate further perceived events in the auditory stream such as further

percussion instruments, but also sung words, or specific elements of the orchestration to regions in the time-frequency plane represented by the matrix squares. For example on the 11th and 12th sub-beat – just before the second snare drum off-beat on the 13th – one can clearly distinguish in the upper frequencies a column of 4 squares followed by 2, belonging to the word *"yes"* pronounced by the singer.

The *Ma-Tricks* display inherits properties from two familiar representations: the sonogram and the drum step sequencer. Both representations are united to a novel intuitive representation easy to calculate from an audio stream under the condition that tempo and/or beats are known or extracted from the audio.

2.1. Perceptive frequency bands

Since the works of Eberhard Zwicker [1] various approaches have been proposed to partition the bandwidth of human hearing into perceptually equal frequency bands from different points of view. Recently real-time audio processing applications of filter banks modeling the human auditory system have been proposed to the electronic music community by Pressnitzer and Gnansia [2].

The *Bark* scale [3] proposes a scale of 24 *critical bands of hearing* up to 15.5 kHz. Using the Bark scale is was easy to partition the frequency axis of the matrix display to 12 perceptually pertinent frequency bands each two Bark large. The number 12 seemed to be a good choice in terms of providing a good visual divisibility for the matrix display. An efficient implementation has been obtained by summing the corresponding frequency bins of the energy spectrum obtained by a short-time Fourier transform (SFFT). Here the FFT size has to be adjusted to permit the representation of the lowest frequency band by at least one bin and also a good approximation of the other frequency bands by the bin frequencies.

2.2. Beat-aligned analysis

Evidently the *Ma-Tricks* representation most pertinently applies to music with a rhythmic structure characterized by the recurrence of a limited number of beat patterns such as today's dance music. To obtain the representation, the SFFT of the 12 frequency bands has to be well synchronised to the beat of the audio stream. The energy regarding each frequency band has to be integrated over one sub-beat. If taking into account tempos between 60 and 200 bpm with a subdivision of 4 sub-beats per beat, a sub-beat has a period between 75 and 250 milliseconds corresponding to 3307.5 and 11025 samples at a sampling rate of 44.1kHz.

Two possibilities have been considered to obtain the given time-frequency matrix using SFFT computation:

- adapting FFT size and hop size to the beat
- using a constant FFT and hop size adding successive FFT frames belonging to the same beat

The advantage of the first is the perfect adaptation of the processing to the perceived rhythm and onsets. For the current implementation of the application, the second possibility has been adopted for its simplicity with success. The hop size has been chosen rather low (128 samples) in order to provide a sufficient temporal resolution.

A bar by bar display of the beat pattern represented by the matrix synchronised to the audio stream in real-time requires either a prior (non real-time) analysis of the reduced sonogram or the delaying of the audio stream by the duration of one bar. The real-time

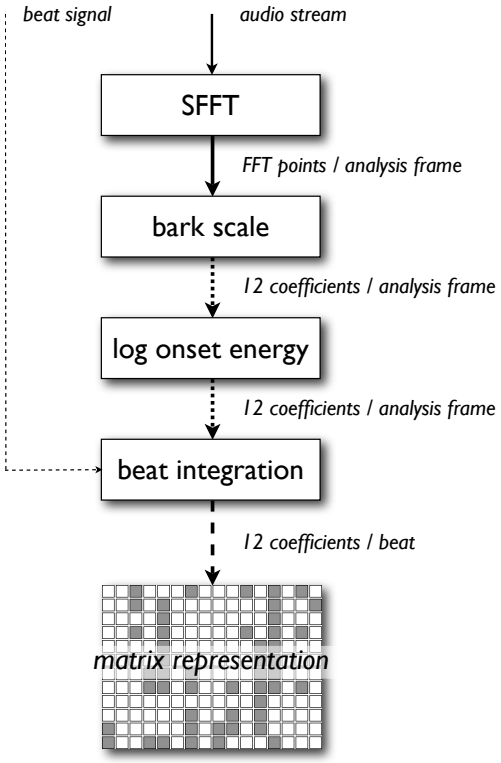

Figure 2: *Data-flow diagram of the X-Micks/Ma-Tricks analysis stage.*

display chosen for the *X-Micks* application continuously shows the analysis result of the last (sub-)beat in a column of the matrix, while the column representing the current (sub-)beat is hidden by a cursor advancing synchronously with the timing of the music. A memory effect can be added to the display reinforcing the visual presence of recurrent elements of the beat patterns of successive bars.

In the context of the described work the beat analysis and tracking of the audio stream is considered calculated outside of framework of the described application. The processing requires a simple control signal (or event) giving a count for each sub-beat of a bar. Generally, audio plug-in standards such as VST[1], AudioUnits[2] or RTAS[3], provide information concerning the signature, tempo and the exact onset time of beats from which the required sub-beat synchronous signal can be easily derived.

Originally the meter and beat information of plug-ins was used in the context of composition environments mixing music representations with a metric structure and audio processing. Here it allows audio effects such as a multi-delay or a chorus to be synchronised to the meter of a synthesised accompaniment. A newer generation of tools allows beat synchronous processing by extracting the tempo and onsets in real-time from an audio stream and generating the beat related plug-in information on the fly or by a previous off-line analysis of the processed audio file. The actual availability of beat information within a particular audio applica-

[1] Steinberg's *Virtual Studio Technology* plug-in standard.
[2] Apple's audio plug-in standard for Mac OS X.
[3] Digidesign's plug-in format developed for ProTools.

tion depends on the implementation of the plug-in host.

Figure 2 shows an overview over the described analysis stage.

3. THE *X-MICKS* APPLICATION

The *X-Micks* application uses the described matrix representation to interact with a real-time audio processing algorithm filtering and mixing to beat synchronised (stereo) audio streams to a hybrid audio stream. Also the beat-synchronisation of the two incoming audio streams is considered to be external to the application. In the case of a plug-in implementation the beat signal and the audio streams have to be handled synchronised to the *X-Micks* plug-in. This is the case for the envisaged integration of an *X-Micks* VST plug-in into the host application *Traktor*[4].

3.1. The graphical user interface

Each of the two matrices of the *X-Micks* interface is associated to one of the input audio streams. The matrices display each the reduced sonogram of one bar of the beat synchronised audio streams in real-time. Figure 3 shows the prototype interface of the *X-Micks* applications.

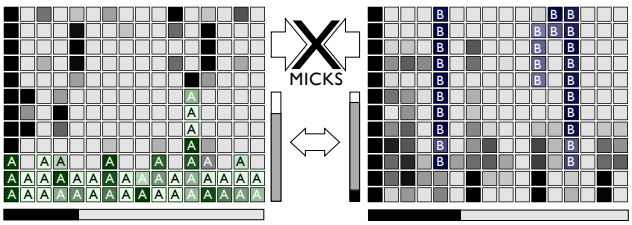

Figure 3: *The prototype interface of the X-Micks application (version 3) showing the real-time analysis of two beat-synchronized songs, the selected time-frequency regions with the beat cursor (on the first beat) and the other user interface elements.*

The user can interact with both matrix displays by clicking to the squares of the matrix toggling the state between unselected (grey) and selected (coloured and additionally marked *A* and *B* in figure 3). With the default settings, the selected matrix regions are audible, while the unselected are not. This way the user can reduce each of audio streams to certain beats and frequency bands. Short cuts are provided to control entire lines – corresponding to particular frequency bands – and entire columns – corresponding to particular beats. The matrices can be tied so the interaction with one matrix automatically also modifies the other in a way that one matrix always represents the inverse selection of the other.

An additional short cut permits to select all matrix squares belonging to the same energy range in the column around a selected square and to the same frequency bands in other beats. The tolerance of the energy range around the value of the selected square can be dynamically adjusted by dragging the mouse after clicking, so that the user can select easily the time-frequency regions of distinguishable recurrent components of the audio stream such as bass drum, snare drum or hi-hat. This interaction depends directly on the analysed energy so that one could speak here of a simple case of *content based interaction*.

[4] *Traktor DJ Studio* is a trademark of Native Instruments Software Synthesis Gmbh – see http://www.nativeinstruments.de/

The range sliders between the matrix displays can be used to adjust the actual level for the selected (upper part of the slider) and unselected (lower part of the slider) time-frequency regions. The slider in the middle left of figure 3 shows that the selected regions of the audio stream represented by the left matrix is slightly lowered while the unselected regions are completely suppressed. In the contrary, the unselected regions of the audio stream represented by the right matrix are not completely supressed as shown by the slider on the right.

3.2. Beat-synchronous filtering

The actual sound processing to create the final hybrid audio stream out of the two incoming audio streams is sufficiently described as *time-variant beat-synchronous filtering*. As well the filtering can be relatively easily implemented based on SFFT. Figure 4 shows an overview over the involved data-flow including the user interaction.

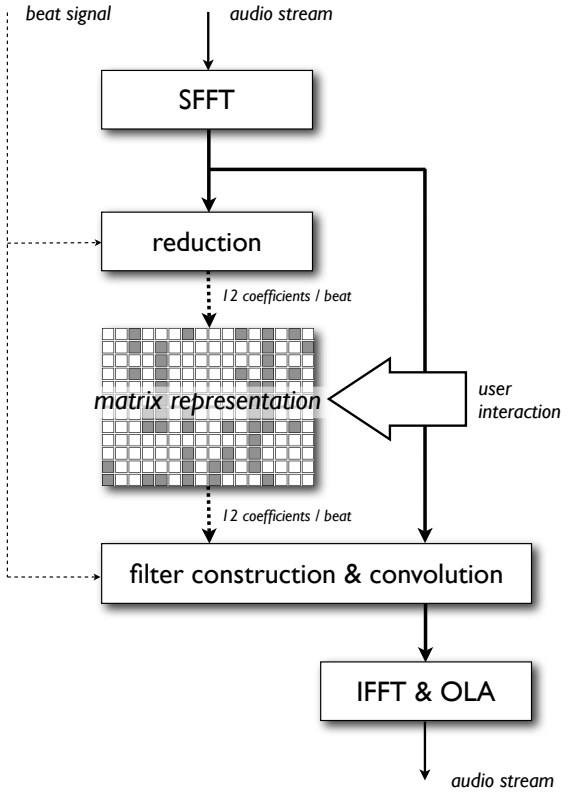

Figure 4: *Data-flow diagram of the beat-synchronous processing of the X-Micks application.*

A straight forward implementation derives the reduced sonogram representation from the same SFFT spectra of the input streams which are used for the convolution and re-synthesis (by IFFT and overlap-add) after the superposition of the spectra from both streams to a single output steam. One can see the overall algorithm as a simple real-time analysis/synthesis process which provides an intermediate representation allowing for intuitive user interaction. In the case of using beat synchronous SFFT the algorithm implements a beat-synchronous overlap-add (*BSOLA*)

algorithm as proposed by Peeters for a different context [4]. In analogy to *frequency-domain PSOLA*, here one could speak of *FD BSOLA*.

3.3. Prototyping and implementation environment

The implementation of the developed *X-Micks* application is entirely based on the *Gabor* [5] library developed at IRCAM with the *FTM* [6] extension for the Max/MSP [7] environment[5]. The matrix representations as well as the SFFT processing are implemented using the FTM *fmat* class and the related functionalities allowing very rapid and efficient prototyping within Max/MSP. The plug-in version has been developed using the pluggo[6] extension for Max/MSP.

4. CONCLUSIONS AND FUTURE WORK

X-Micks is a prototype application of interactive real-time content based audio processing. Even if *X-Micks* features rather basic signal processing it perfectly embodies a novel paradigm of audio processing. We made a tentative to define this novel genre of interactive audio processing applications in a way that is meant to be inspiring for further developments.

For the *X-Micks* application itself we imagine multiple future enhancements. One is related to one of the ideas being at the origin of this work, which can be called *interactive source separation*. The act of choosing time frequency regions – beats and frequency bands – belonging to a certain instrument or sound source while listening to the filtered result can be seen as an interactive approach to source separation. One can easily imagine to enhance this approach by additional features and more sophisticated audio processing for example introducing source separation [8] and segmentation algorithms. A preliminary off-line analysis (or even decomposition) of the audio stream is acceptable for many applications.

Further efforts have to go in the further simplification and refinement of the graphical user interface and interaction to enhance the performability of the application.

We see a potential in the described matrix representation for the description and indexing of rhythmic patterns and segmented audio loops. When coding the energy of the matrix with 2-bit (four steps) each line of a 12×16 matrix can be represented by a 32-bit word. The 12 frequency bands maybe can be further reduced to 3 or 4 for this application.

5. ACKNOWLEDGEMENTS

This work was carried out in the framework of the the European project *Semantic HIFI* and is the result of a fruitful collaboration with the company *Native Instruments*. Thanks to Florian Plenge and Egbert Juergens from NI for their encouraging support and helpful comments.

The ideas for the *X-Micks* application are inspired by the work of Andrea Cera and initially came up in a discussion with Andrea, Nicola Donin and Samuel Goldszmidt, with whom the mentioned *Ma-Tricks* application has been realized.

Further inspirations for this project are coming from the work and discussions with Geoffroy Peeters. Finally we'd like to thank our colleagues Riccardo Borghesi and Frederic Bevilacqua for courageously having endured unbearable hours and weeks of testing the *X-Micks* plug-in and its preceding prototypes with the same short loops extracted from pop songs.

6. REFERENCES

[1] E. Zwicker, G. Flottorp, and S. S. Stevens, "Critical bandwidth in loudness summation," *J. Acoust. Soc. Am.*, vol. 29, pp. 548–557, 1957.

[2] D. Pressnitzer and D. Gnansia, "Real-time auditory models," in *Proc. Int. Comp. Music Conf. (ICMC'05),* Barcelona, Spain, 2005, pp. 295–298.

[3] J. O. Smith III and J. Abel, "Bark and ERB bilinear transform," *IEEE Trans. Speech and Audio Proc.*, vol. 7, no. 6, pp. 697–708, Nov. 1999.

[4] A. La Burthe and G. Peeters, "Résumé sonore," in *Internship in the framework of the master ATIAM – final report IRCAM/INPG Grenoble*, Paris, France, 2002.

[5] N. Schnell and D. Schwarz, "GABOR, multi-representation real-time analysis/synthesis," in *Proc. Int. Conf. on Digital Audio Effects (DAFx-05),* Madrid, Spain, 2005, pp. 122–126.

[6] N. Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Müller, "FTM – complex data structures for Max," in *Proc. Int. Comp. Music Conf. (ICMC'05),* Barcelona, Spain, 2005, pp. 9–12.

[7] M. Puckette, "Combining event and signal processing in the MAX graphical programming environment," *Computer Music J.*, vol. 15, no. 3, pp. 68–77, 1991.

[8] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech and Language Proc.*, vol. 14, no. 4, pp. 1462–1469, July 2006.

[5] http://www.cycling74.com/products/maxmsp
[6] http://www.cycling74.com/products/pluggo