TAPESTREA: A New Way to Design Sound

[Extended Abstract]

Ananya Misra
Princeton University
35 Olden St.
Princeton, NJ 08540, USA
amisra@cs.princeton.edu

Ge Wang
Stanford University
660 Lomita Dr.
Stanford, CA 94305, USA
ge@ccrma.stanford.edu

Perry R. Cook
Princeton University
35 Olden St.
Princeton, NJ 08540, USA
prc@cs.princeton.edu

ABSTRACT

TAPESTREA is a sound design and composition framework that facilitates the creation of new sound from existing digital audio recordings, through interactive analysis, transformation and re-synthesis. During analysis, sound templates of different types are extracted using a variety of techniques. Each extracted template is transformed and synthesized independently, allowing specialized transformations on each template based on its type. The user interacts with TAPESTREA via a set of graphical interfaces that offer parametric control over every stage of analysis, transformation and re-synthesis. Synthesis is further controlled through ChucK scripts. These combined techniques form a workbench for completely transforming a sound scene, dynamically generating soundscapes, or creating musical tapestries by weaving together transformed elements from different recordings. Thus, TAPESTREA introduces a new paradigm for sound design, composition and sonic sculpting tasks.

Categories and Subject Descriptors

D.0 [**Software**]: General

General Terms

Design, Experimentation

Keywords

Sound design, composition, audio, multimedia, signal processing, real-time, open source software

1. INTRODUCTION

A sound designer or artist manipulating existing digital audio for purposes such as sound scene creation or musical composition often encounters difficulties because the existing sounds are not exactly as desired. A single recording is unlikely to have all desired sounds; unwanted sounds may overlap the wanted parts; the wanted part may not have the exact desired frequency, length, or other quality. Tools for transforming sounds are constrained either in the range of sounds to which they apply or in the manipulation paradigms and variety of results they offer. TAPESTREA aims to surpass these limitations by presenting a unified framework for creating new sound from any combination of existing audio, with expressive freedom in selecting both *what* to re-use and *how* to re-use it.

Given one or more recordings, TAPESTREA provides well-defined means to:

- Identify points of interest in a sound and extract them into re-usable *templates*
- Transform individual templates independently of the background and/or other events
- Continually re-synthesize background sound textures
- Controllably place event templates over backgrounds, using a novel graphical user interface and/or scripts written in the ChucK [18] audio programming language

In this way, it provides a new way to completely transform a sound scene and to compose and design sound by combining elements from different recordings [12, 13].

2. RELATED WORK

Currently available sound editors for working with existing sounds range from inexpensive to professional. Free or inexpensive commercially available software such as Audacity and GoldWave perform simple audio production tasks. Midline audio editing systems, including Peak, Logic, and Cubase, are geared toward music production, while high-end hardware/software systems such as Pro Tools are geared toward commercial sound production. Most of these products support Virtual Studio Technology (VST) plug-ins that perform synthesis algorithms and apply audio effects. However, none of them provides one real-time, integrated analysis-transformation-synthesis workspace.

Existing tools also include specialized software and libraries that implement particular audio analysis and synthesis algorithms. The CLAM library [1], Loris [7], SPEAR [9] and AudioSculpt [4] enable spectral analysis and re-synthesis. However, they do not offer parametric control in combining different types of transformed elements from multiple source sounds (see Figure 1). TAPESTREA brings together a variety of techniques for effectively capturing and transforming different types of sounds. Techniques used include sinusoidal modeling and spectral modeling synthesis (SMS) [11, 14, 16], transient detection [17, 3], and sound texture synthesis [6, 2, 19, 8].

3. TAPESTREA

TAPESTREA manipulates sound in several phases (see Figure 2). In the analysis phase, desired parts of a recording

MM'09, October 19–24, 2009, Beijing, China.
ACM 978-1-60558-608-3/09/10.

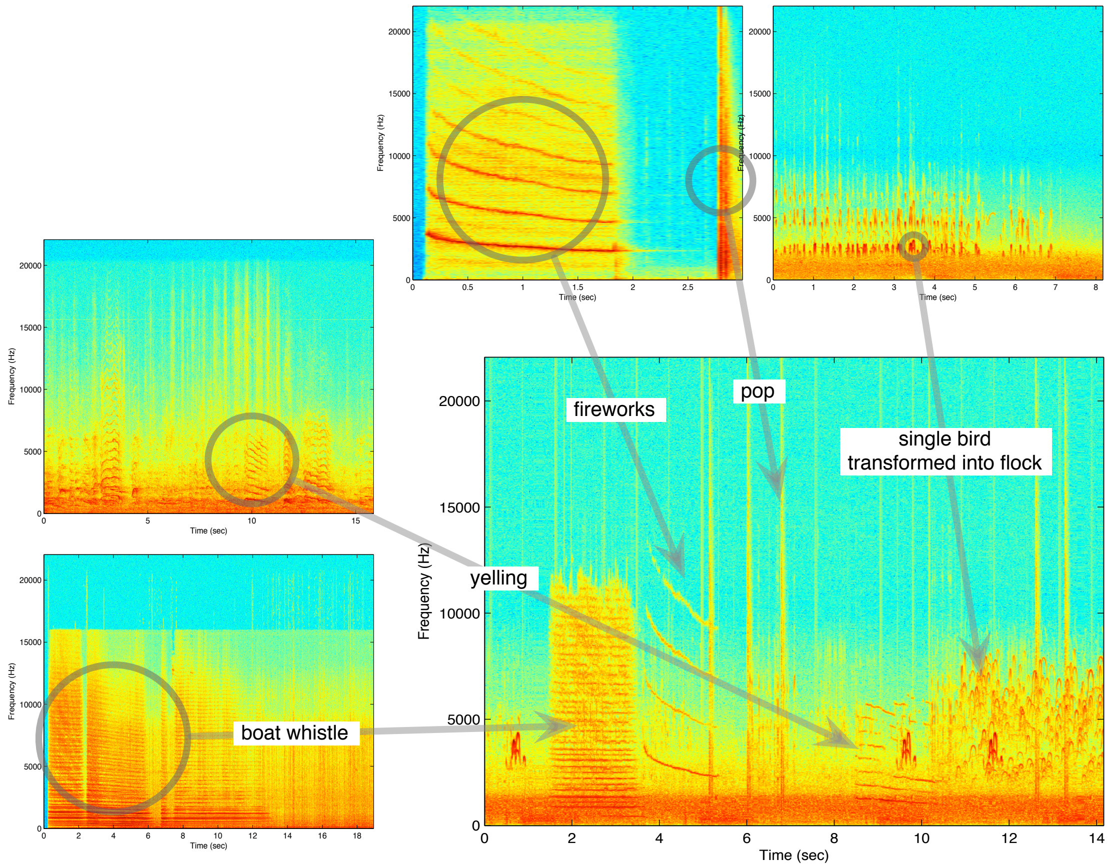

Figure 1: The TAPESTREA concept: Selected elements from multiple source sounds are extracted, transformed, and combined to create a new sound scene.

are interactively extracted into templates of different types. Pitched components are saved as *sinusoidal* templates using SMS [11, 14, 16]. Brief noisy events are captured as *transient* templates, using signal processing methods for transient and onset detection [17, 3]. The *background* din is extracted by removing sinusoidal components in the frequency domain [16], and replacing transient segments in the time domain with a brief sound texture synthesized by wavelet tree learning [6]. A region in time and frequency can be captured as a *raw spectrogram* template, using Fast Fourier Transform (FFT) filtering.

Each template can then be transformed independently, including special low-level structural changes to sinusoidal templates [10]. In the synthesis phase, extreme time and frequency transformations are available for sinusoidal templates, which are re-synthesized using additive synthesis. Transient and raw spectrogram templates may also undergo time and frequency transformations via a phase vocoder [5]. Background din may be continuously and parametrically generated using wavelet tree learning [6]. Special *synthesis* templates such as *loops*, *timelines* and *mixed bags* enable the flexible combination of these transformed sounds to produce a new composite soundscape or composition. A set of graphical user interfaces offers interactive parametric control over each phase of analysis, transformation and resynthesis. ChucK *scripting* provides additional control over the synthesis by allowing the simultaneous, precise manipulation of many parameters; it also provides an interface for control from external devices or user-defined GUI elements.

The TAPESTREA software is open-source, cross-platform, and freely available at `http://taps.cs.princeton.edu/`.

4. APPLICATIONS

Figure 1 represents the TAPESTREA sound design concept through the creation of an example sound scene. Selected parts of several different recordings are extracted, individually transformed, and put together to create a new sound scene that combines elements of existing scenes. This leads to a new sound design paradigm applicable to audio generation for games, virtual reality, video and other entertainment. The same techniques may also create musical compositions related to *musique concrète* [15] and other genres, in which parts of the finished composition are arbitrarily similar to or unrecognizable from the corresponding source sounds. Thus, applications of TAPESTREA include:

- Synthesis, sound design, composition: Creating sound scenes for entertainment, virtual reality, and musical compositions;
- Analysis, information extraction: Parametrically extracting a part of a recording, possibly gaining a better understanding of the sound through the analysis techniques and parameters that best capture it;

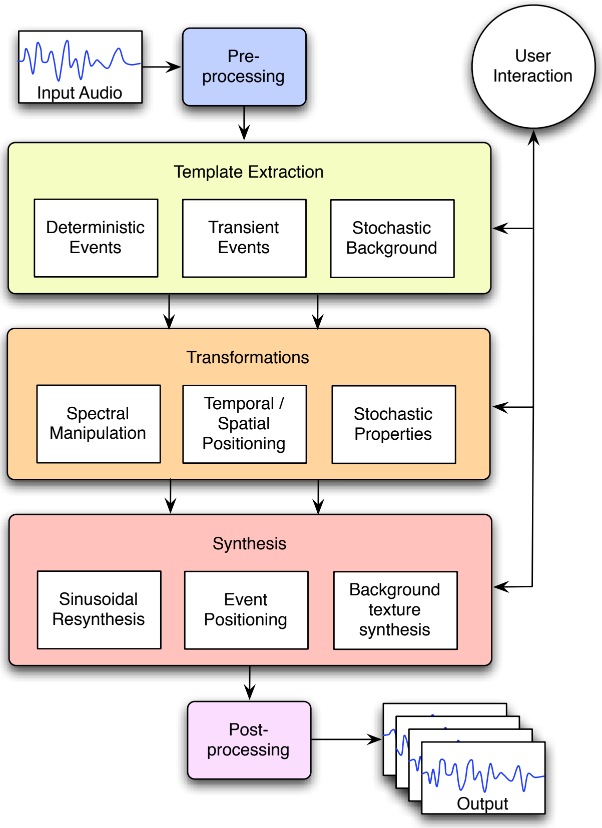

Figure 2: TAPESTREA sound design pipeline.

- Pedagogy: Enhancing understanding of digital audio, signal processing, and computer music concepts via the interactive analysis, synthesis, and audiovisual display.

5. CONCLUSIONS AND FUTURE WORK

TAPESTREA provides a new way to design sound through highly flexible manipulation of existing recordings. The tools and interface together provide great freedom in selecting which parts of a sound to re-use and how to transform and combine selected elements. The interactive coupling of analysis and synthesis techniques and paradigms make TAPESTREA stronger than the sum of its parts, leading to a powerful tool for sound design, composition and pedagogy. Areas for further work include continuing to integrate user feedback to improve the system, as well as expanding it to provide even more powerful scripting, intelligent parameter suggestions via machine learning, and a meaningfully searchable database of extracted templates.

6. REFERENCES

[1] X. Amatriain and P. Arumi. Developing cross-platform audio and music applications with the CLAM framework. In *Proceedings of the International Computer Music Conference*, 2005.

[2] M. Athineos and D. P. W. Ellis. Sound texture modeling with linear prediction in both time and frequency domains. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 5, pages 648–651, 2003.

[3] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 13(5), 2005.

[4] N. Bogaards, A. Robel, and X. Rodet. Sound analysis and processing with AudioSculpt 2. In *Proceedings of the International Computer Music Conference*, 2004.

[5] M. B. Dolson. The phase vocoder: A tutorial. *Computer Music Journal*, 10(4):14–27, 1986.

[6] S. Dubnov, Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, and M. Werman. Synthesizing sound textures through wavelet tree learning. *IEEE Computer Graphics and Applications*, 22(4), 2002.

[7] K. Fitz, L. Haken, S. Lefvert, and M. O'Donnel. Sound morphing using Loris and the reassigned bandwidth-enhanced additive sound model: Practice and applications. In *Proceedings of the International Computer Music Conference*, 2002.

[8] M. Fröjd and A. Horner. Fast sound texture synthesis using overlap-add. In *Proceedings of the International Computer Music Conference*, 2007.

[9] M. Klingbeil. Software for spectral analysis, editing, and synthesis. In *Proceedings of the International Computer Music Conference*, 2005.

[10] T. Lieber, A. Misra, and P. R. Cook. Freedom in TAPESTREA! Voice-aware track manipulations. In *Proceedings of the International Computer Music Conference*, 2008.

[11] R. McAulay and T. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(4):744–754, 1986.

[12] A. Misra, P. R. Cook, and G. Wang. A new paradigm for sound design. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, 2006.

[13] A. Misra, G. Wang, and P. R. Cook. Musical tapestries: Re-composing natural sounds. *Journal of New Music Research*, 36(4), 2007.

[14] T. Quatieri and R. McAulay. Speech transformations based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(6):1449–1464, 1986.

[15] P. Schaeffer. *À la Recherche d'une Musique Concrète*. Seuil, Paris, 1952.

[16] X. Serra. *A System for Sound Analysis / Transformation / Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Stanford University, 1989.

[17] T. S. Verma and T. H. Meng. An analysis/synthesis tool for transient signals that allows a flexible sines+transients+noise model for audio. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 12–15, 1998.

[18] G. Wang and P. R. Cook. ChucK: A concurrent, on-the-fly, audio programming language. In *Proceedings of the International Computer Music Conference*, 2003.

[19] X. Zhu and L. Wyse. Sound texture modeling and time-frequency LPC. In *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, 2004.