Landschaften – Visualization, Control and Processing of Sounds in 3D Spaces

Savannah Agger, Jean Bresson, Thibaut Carpentier

Landschaften – Visualization, Control and Processing of Sounds in 3D Spaces

Savannah Agger
University of Birmingham, UK
savannahagger@yahoo.se

Jean Bresson
UMR STMS: IRCAM/CNRS/UPMC
Paris, France
jean.bresson@ircam.fr

Thibaut Carpentier
UMR STMS: IRCAM/CNRS/UPMC
Paris, France
thibaut.carpentier@ircam.fr

ABSTRACT

This paper explores some techniques developed in the creation of the multichannel acousmatic piece Landschaften *and the use of OpenMusic tools for the dynamic manipulation of 3D audio description data. The compositional idea derives from sculpting sound objects as 3D models in space, used to control sound synthesis and spatialization, and to connect the spectral and diffusion spaces.*

1. INTRODUCTION

Compositional techniques like retrograde and inversion have existed for hundreds of years. Finding compositionally meaningful generalizations of these techniques to generate variations of the musical material involved in electroacoustic works is a challenging problem, which goes beyond generating variations of melodic/harmonic materials, to variations of the inner structure of sound objects.

This paper explores some techniques used in the creation of Savannah Agger's multichannel acousmatic piece *Landschaften*. [1] The compositional processes were realized using prototypes developed in the OpenMusic (OM) computer-aided composition and visual programming environment [1, 2]. [2] These developments include new tools for the visualization and processing of sound description data in 3D, and applications of signal processing and spatialization.

2. COMPOSITIONAL BACKGROUND AND MOTIVATION

The composer's main approach to this project is based on the imagination that one could sculpt sound objects as 3D models in "space": space as in "*spectral space*" [3], and as in *spatial sound space* [4, 5, 6].

Sound spatialization in compositional contexts most often implies positioning/projecting/moving sound elements in a space, typically relying on point-source trajectory paradigms. The suggested approach here is concerned with accessing the sound object/theme from all possible angles, and listening to this object as if it was a space in itself. The compositional work therefore explores the idea of sound considered as space, and composition as a landscape, travelling through the different dimensions of the sound and its inner structures.

Xenakis [7] discussed how "sonic events [...] made out of thousands of isolated sounds [collision of hail or rain with hard surfaces, or the song of cicadas in a summer field]" could be seen "as totality, [as] a new sonic event." In the real world, one can indeed listen to a "sonic event" as described by Xenakis from different spots or angles, record it, and get closer or further away from the constituent sounds out of which this sonic event is made. By changing one's listening position relative to the sound objects and to the sonic event as a whole, several attributes can be affected, such as the loudness of the individual sounds, their micro-temporal relations, or the global sound spectrum. What if the computer could also perform this change of viewpoint/listening perspective at a micro, macro and meta level? What kind of methods and manipulations would be useful to achieve interesting "variations of the theme" with the inner sound structure?

In the following sections we will describe how the idea of representation and control of the spectral and spatial sound spaces is implemented as a compositional tool. In order to achieve this goal, we process through:

- Visualization, mapping, and transformation of the spectral features of a sound in a 3D space. Re-synthesis of the transformed spectra.
- Application of filters (pre-spatialization) to the transformed and re-synthesized elements, decomposing the signals into multiple frequency bands.
- Spatialization over a multichannel loudspeaker system, where gestures in space are controlled by the inner properties of sounds.

3. SPECTRAL REPRESENTATION AND PROCESSING OF SOUNDS IN THE 3D SPACE

Representing additive sound analyses in 3D – Manipulating representations in the 3D space – Re-synthesizing sounds.

The approach taken for the 3D representation of sound starts with a decomposition of the signal into partial-tracks using a standard additive analysis technique [8]. Each partial is represented as a sequence of *time* × *frequency* × *amplitude* points. Based on an interactive visual representation of the 3D space containing all these sequences of points (connected and displayed as 3D lines), we apply basic geometric transformations (rotations, translations) to the structure as a whole and then convert back the transformed points into suitable material for re-synthesis.

[1] *Landschaften* (2017) will be premiered at the Seoul International Computer Music Festival 2017. The main material for the piece was produced during a residency at IRCAM in 2016-2017.

[2] The tools and examples presented in this paper are part of a new prototype implementation of OpenMusic, described in [2].

3.1 Implementation with partials

The OpenMusic patch in Figure 1 shows an example of the 3D visualization and processing of a spectral sound representation. The basic material is derived from a sound file using the *partial-tracking* function of the OM-PM2 library. Partials' time, frequency and amplitude values are extracted from a file in the SDIF format [9] containing the analysis data produced by the PM2 sound processing kernel. These values are converted into 3D curves, input and represented in the *3D-viewer* object developed for this project.

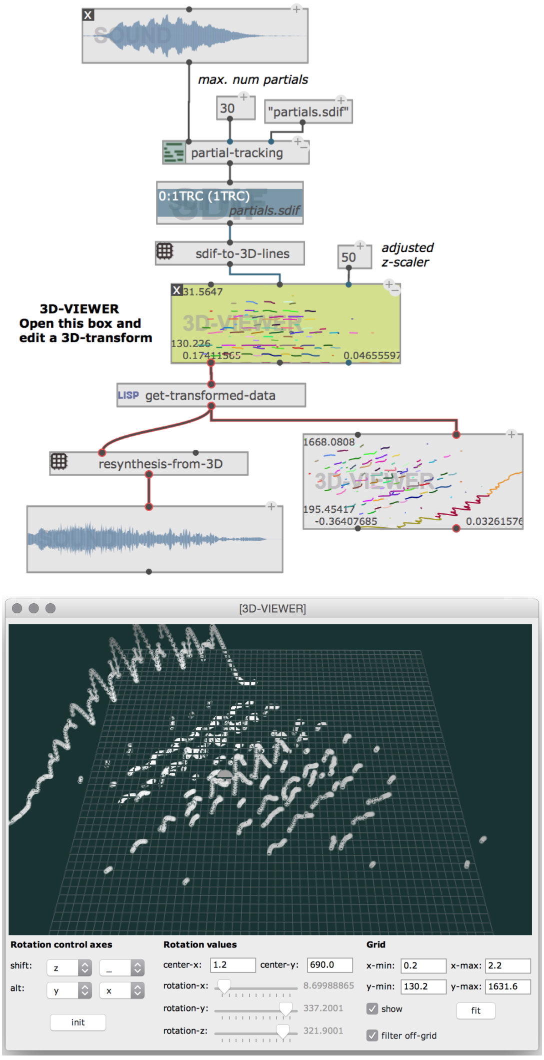


Figure 1. Analysis, 3D-manipulation and resynthesis of partials. Top: OpenMusic patch. Bottom: Editor of the *3D-viewer* object.

The *3D-viewer* enables interactive navigation in the 3D space, and the control of a number of spatial transformations (mostly, rotations) applied to the 3D objects. 3D-modified data is then extracted out of the viewer (see *get-transformed-data* on Figure 1) and further processed to be converted back into consistent *time* $\times$ *frequency* $\times$ *amplitude* data (i.e. a new set of partials), stored as a new SDIF file reused by OM-PM2 to perform additive synthesis.

The reactive functionality in OM [10] (materialized by red-highlighted patch cords in Figure 1) is used to automatically update the exported data and trigger the additive synthesis computation upon specific actions performed by the composer in the *3D-viewer* interface.

The grid laid out in the viewer's background sets a referential for abstract data visualization. It can also serve as a "filter" to restrain the 3D-rotated data within a certain time/frequency range for extraction and re-synthesis.

3.2 Implementation with grains/point-controllers

As an alternative approach to the same idea, sound textures can be represented using clouds of points or "grains" in a 3D space. Here the material extracted from *partial-tracking* analyses of sounds is reduced and converted into MIDI-like events on a *piano-roll* (described by a single onset, pitch and velocity, optionally a duration, see Figure 2). These events can also be turned into simple 3D segments and represented in the *3D-viewer* interface (see Figure 3-a). After 3D modification in the viewer, the 3D segments are projected back on the *time* $\times$ *pitch* $\times$ *velocity* axes as a transformed *piano-roll* (Figure 3-b), and can be played with an external sampler or any MIDI-compatible application.

This strategy opens up to using, for example, granular synthesis and other sample-based techniques to render sonic variations from an original sound. It also suggests connecting this approach to further MIDI- or score-oriented processing and instrumental music composition.

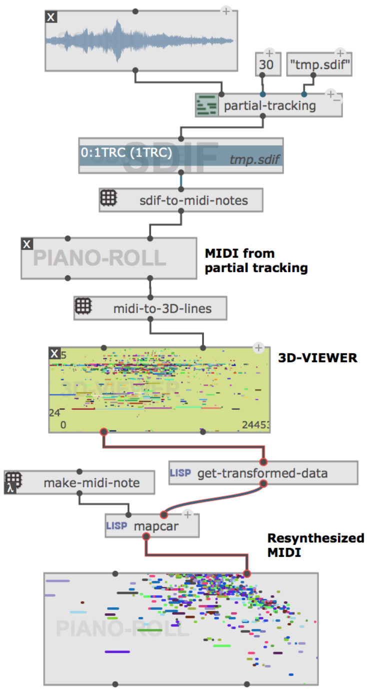
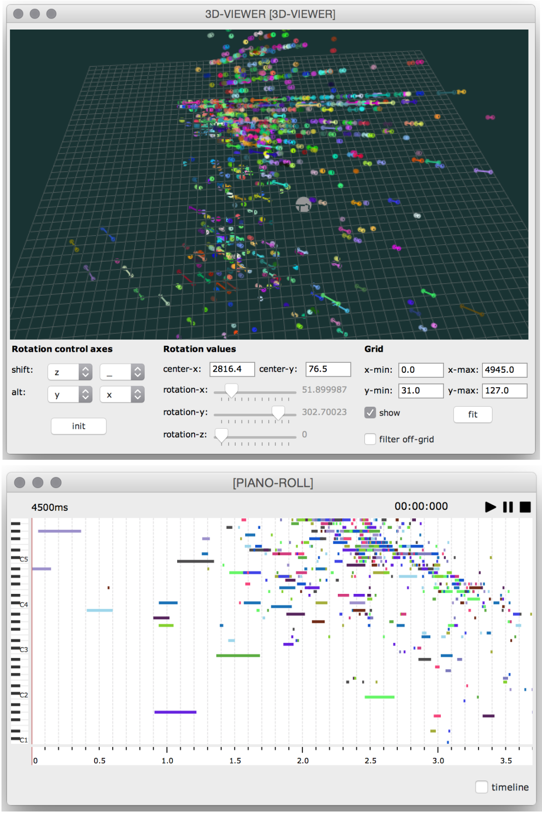


Figure 2. MIDI conversion, 3D-transformation, and output of the sound analysis data.

4. SPATIAL SOUND PRE-PROCESSING AND RENDERING

Programming filter banks – Spatializing sources – Controlling the temporal evolution of filtering and spatialization processes.

One objective of the sound spatialization for *Landschaften* was to distribute the sonic elements in the concert space according to their spectral content. Most signal processing and spatialization operations are done using DSP tools



(a)

(b)

Figure 3. 3D-transformation and rendering of MIDI-formatted data (a) in the *3D-viewer* editor and (b) converted back into a MIDI piano-roll.

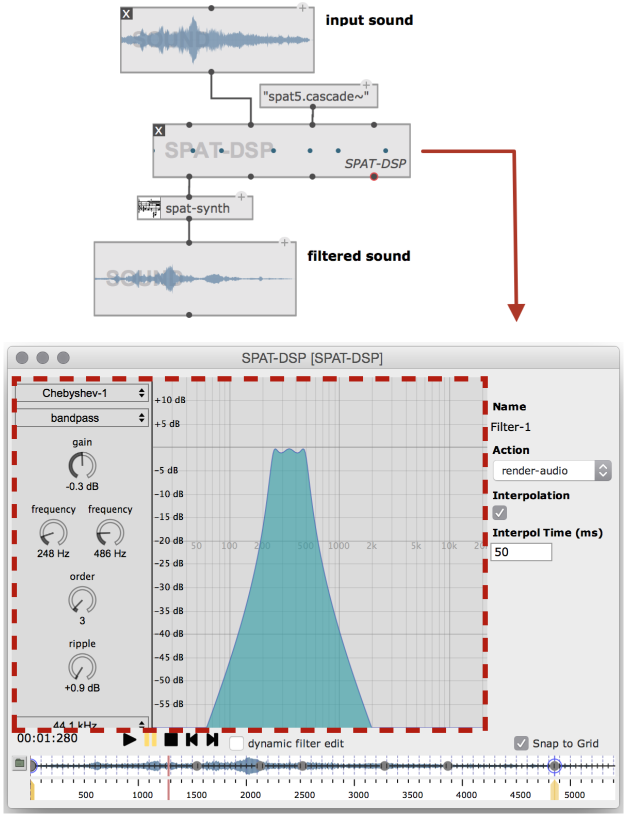


Figure 4. External DSP controllers in OpenMusic: a cascade of second-order (IIR) filters. The editor of the DSP object displays an embedded GUI module for filter design, controlling the state of the DSP engine over a timeline. The dashed frame highlights the borders of the GUI imported from the SPAT library.

derived from the SPAT library [11]. The OM framework embeds them in a composition-oriented setup, enabling the algorithmic programming and instantiation of banks of DSP processors, and the control of their evolution over time.

4.1 Filtering

A preliminary band-filtering stage acts as pre-processing, determining both frequency ranges for the sounds and regions for their spatial projection. This pre-processing is applied to sounds generated from the *3D-viewer*, and to other, unaltered sounds.

A set of new DSP objects created in OM link to GUI and DSP components from a C library, and integrate them in visual programs and graphical editors as controllers for the real-time or offline processing of audio buffers (see Figure 4).

These objects store sequences of time-stamped OSC *bundles* containing messages to describe the successive states of a digital signal processor and its controller (see the timeline at the bottom of the editor on Figure 4, depicting the different states of the filter). The main GUI (a view imported from the external library) is used to set the state of the DSP controller at different time-points. At rendering time, its display and the corresponding DSP control messages can be interpolated to perform smooth transitions from one specified state to another.

DSP parameters and their evolution in time can also be generated or processed in OM visual programs. In *Landschaften*, manually-defined states for the DSP objects are extrapolated to algorithmically derive parameter sets and instantiate series of other DSP objects aimed at processing a collection of sounds. For instance, the OSC bundles and message values controlling the band-filters can be processed and transformed, creating multiple replicas of a filter topology with shifted cut-off frequencies (see Figure 5).

4.2 Spatialization

Sounds obtained and transformed through the previous processes are spatialized following 3D trajectories precisely determined according to their spectral range (generally determined by the previous stage of band-filtering, and mapped to the elevation of the sounds in the 3D space), or to other time-varying sound descriptors (also extracted from the signals prior to synthesis using different OM libraries). These techniques, letting the spectral content and other characteristics of the sound control movements in the physical space, derive from the same compositional objective to approach the sound as a spectral body in space, rather than an object projected in a volume.

The SPAT-SCENE object [12] is used as the main controller for sound spatialization (see Figure 6). This DSP object provides spatial authoring and rendering functionalities applied to audio sources and specified using hand-drawn and/or algorithmically generated trajectories. As is the case for other DSP processors, spatialization can be performed in real-time (in the SPAT-SCENE editor or in another rendering/playback context), or offline in order to produce bounced multi-channel audio files. Finally, SPAT-SCENE permits composers to easily interchange spatialization tech-

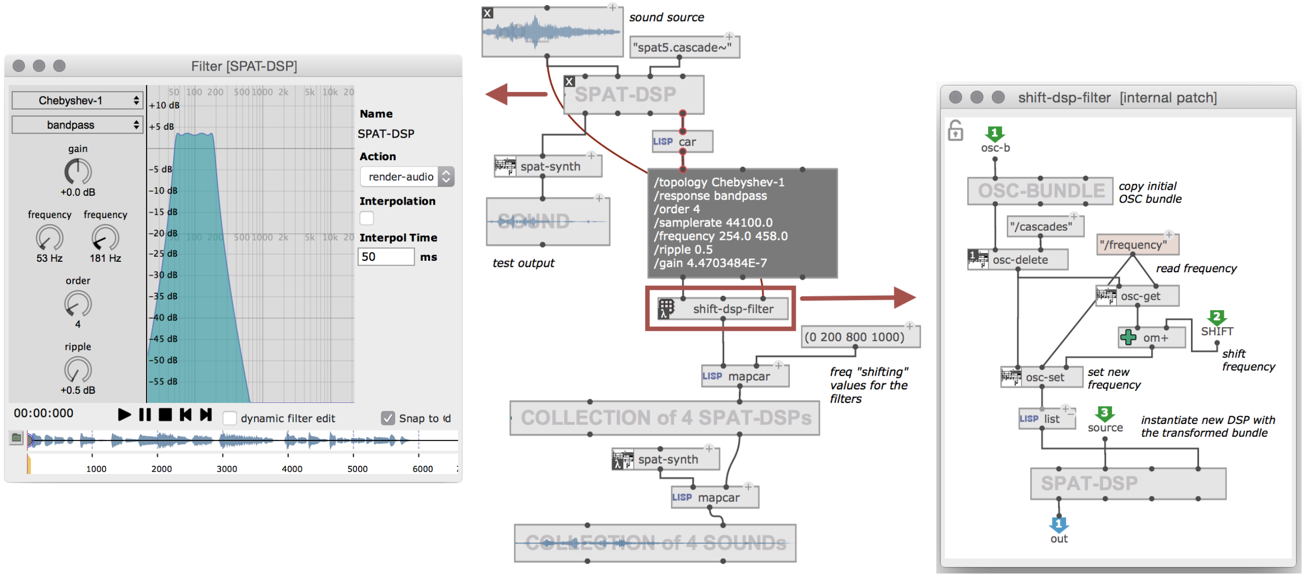


Figure 5. Algorithmic generation of DSP objects for the batch processing of sound sources. The initial filter is set manually in the SPAT-DSP editor at the left. The *shift-dsp-filter* patch at the right, applied iteratively with different frequency-shift values, performs transformations on the original OSC message values in order to modify the filter cut-off frequencies, and instantiates new DSP objects.

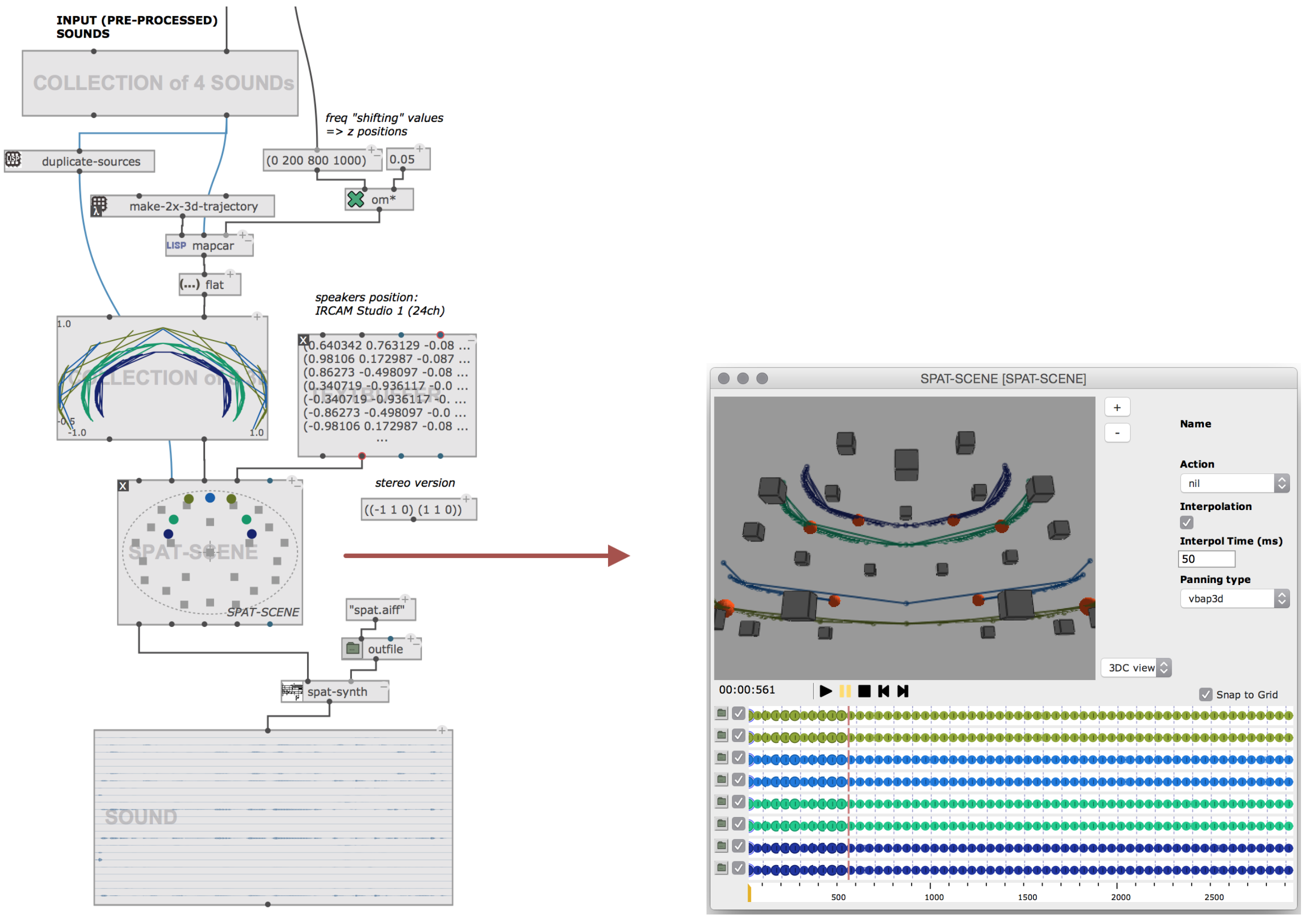


Figure 6. Generation of a spatial scene from pre-processed sound materials and synthesis of a multi-channel audio file. The sub-patch *make-2x-3D-trajectory*, whose content is hidden from the figure, generates for each source two symmetrical trajectories computed from descriptor analyses of the sound. The SPAT-SCENE editor visible at the right provides an animated and interactive display of the scene. The editable timelines in the lower-part of the window represent the sequence of control metadata for each source trajectory.

niques and the configuration of speakers for the display and rendering of spatial scenes: a practical necessity in such compositional work, requiring to constantly switch between headphone-monitoring, home studio, studio rehearsal and concert-hall configuration.

5. PREVIOUS AND RELATED WORKS

Visual methods for the manipulation of sounds have been explored in the past, starting with early graphical interfaces such as Xenakis' UPIC system [13], and through more recent software packages such as METASYNTH [3]. OPENSOUNDEDIT [14] proposed 3D visualization and manipulation of sound descriptions (also based on SDIF-formatted data), coupled with real-time audio rendering features. SDIFEDIT [15] was another project aiming at this same direction using 3D visualization, and at the integration of sound description data manipulation in compositional processes. The integration of graphical sound visualization in the visual programming workflow of the OM computer-aided compositional environment positions our work in the close follow-up of this project, while interactive 3D-transformations of the data introduce more intuitive dimensions in the process, enabling for instance the simulation of listening angles (or "viewpoints"), filtering and other manipulations on the material prior to re-synthesis.

The sound spatialization part of the presented work — in particular the "spectral-oriented" spatial ditribution of the sound sources — can also be compared to existing works: see for instance [16] or [17] for examples of similar approaches. In this context as well, our system is in line with previous projects focusing on the compositional control of sound spatialiation carried out in OM [18], and brings in a tight and powerful integration of real-time DSP technology in the complex, timed musical structures computed from the algorithmic processes implemented in the computer-aided composition environment.

6. CONCLUSION

We presented computer-aided composition tools and processes designed for the production of *Landschaften*, a multichannel acousmatic piece by Savannah Agger. From a compositional point of view this work permitted the manipulation and auditory exploration of sounds as three dimensional spectral bodies in space.

3D manipulations enable the production of a broad variety of sounds, from simple to very complex transformations where one can only perceptually localize some traits from the original input. The tools herein presented allow this material to be to grasped in both an intuitive and algorithmic manner. The OM visual programming framework makes it easy to specify additional constraints to the input, the transformations and the output. It also enables further combination of these processes with numerous other compositional techniques and tools/objects that are available in the computer-aided composition environment.

A constraint of the proposed approach is that graphical operations may become cumbersome and computationally inefficient when applied to very large datasets — for instance noise-based sounds, which analysis using the additive method usually results in a high number of pseudo-partials, are not really suited for this approach. In this perspective, future developments could be oriented towards the exploration and use of different analysis and synthesis techniques, to represent sounds with new descriptors mapped to 3D-transformation, processing and spatialization parameters.

[3] http://www.uisoftware.com/MetaSynth/

Acknowledgments

This project and artistic residency were carried out in the context of the EFFICACe research project ANR-13-JS02-0004 funded by the French National Research Agency.

7. REFERENCES

[1] J. Bresson, C. Agon, and G. Assayag, "OpenMusic. Visual Programming Environment for Music Composition, Analysis and Research," in *ACM MultiMedia – OpenSource Software Competition*, Scottsdale, USA, 2011.

[2] J. Bresson, J. Garcia, D. Bouche, T. Carpentier, and D. Schwarz, "Next-generation Computer-aided Composition Environment: A New Implementation of OpenMusic," in *Proceedings of the International Computer Music Conference (ICMC'17)*, Shanghai, China, 2017.

[3] D. Smalley, "Spectromorphology: Explaining Sound-shapes," *Organised Sound*, vol. 2, no. 2, pp. 107–126, 1997.

[4] P. Khosravi, "Spectral Spatiality in the Acousmatic Listening Experience," Ph.D. dissertation, City University, London, 2012.

[5] A. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound. Cambridge*. MIT Press, 1994.

[6] G. Kendall, "Spatial perception in Multichannel Audio for Electroacoustic Music," *Organised Sound*, vol. 15, no. 3, pp. 228–238, 2010.

[7] I. Xenakis, *Formalized Music: Thought and Mathematics in Composition*. Indiana University Press, 1971.

[8] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 4, pp. 744–754, 1986.

[9] M. Wright, A. Chaudhary, A. Freed, S. Khouri, and D. Wessel, "Audio Applications of the Sound Description Interchange Format Standard," in *Audio Engineering Society Convention 107*, New York, USA, 1999.

[10] J. Bresson and J.-L. Giavitto, "A Reactive Extension of the OpenMusic Visual Programming Language," *Journal of Visual Languages and Computing*, vol. 25, no. 4, pp. 363–375, 2014.

[11] T. Carpentier, M. Noisternig, and O. Warusfel, "Twenty Years of Ircam Spat: Looking Back, Looking Forward," in *Proceedings of the International Computer Music Conference (ICMC'15)*, Denton, USA, 2015.

[12] J. Garcia, T. Carpentier, and J. Bresson, "Interactive-Compositional Authoring of Sound Spatialization," *Journal of New Music Research*, vol. 46, no. 1, pp. 74–86, 2017.

[13] G. Marino, M.-H. Serra, and J.-M. Raczinski, "The UPIC system: Origins and innovations," *Perspectives of New Music*, vol. 31, no. 1, pp. 258–269, 1993.

[14] A. Chaudhary, A. Freed, and L. A. Rowe, "Open-SoundEdit: An Interactive Visualization and Editing Framework for Timbral Resources," in *Proceedings of the International Computer Music Conference (ICMC'98)*, Ann Arbor, USA, 1998.

[15] J. Bresson and C. Agon, "SDIF Sound Description Data Representation and Manipulation in Computer-Assisted Composition," in *Proceedings of the International Computer Music Conference (ICMC'04)*, Miami, USA, 2004.

[16] D. Topper, M. Burtner, and S. Serafin, "Spatio-Operational Spectral (S.O.S) Synthesis," in *Proceedings of the International Conference on Digital Audio Effects (DAFx'02)*, Hamburg, Germany, 2002.

[17] D. Kim-Boyle, "Spectral Spatialization – An Overview," in *Proceedings of the International Computer Music Conference (ICMC'08)*, Belfast, Ireland, 2008.

[18] M. Schumacher and J. Bresson, "Spatial Sound Synthesis in Computer-Aided Composition," *Organised Sound*, vol. 15, no. 3, pp. 271–289, 2010.