The Sound Sketchpad: Expressively Combining Large and Diverse Audio Collections

Citation: Singh, Nikhil. 2021. "The Sound Sketchpad: Expressively Combining Large and Diverse Audio Collections."

As Published: https://doi.org/10.1145/3397481.3450688

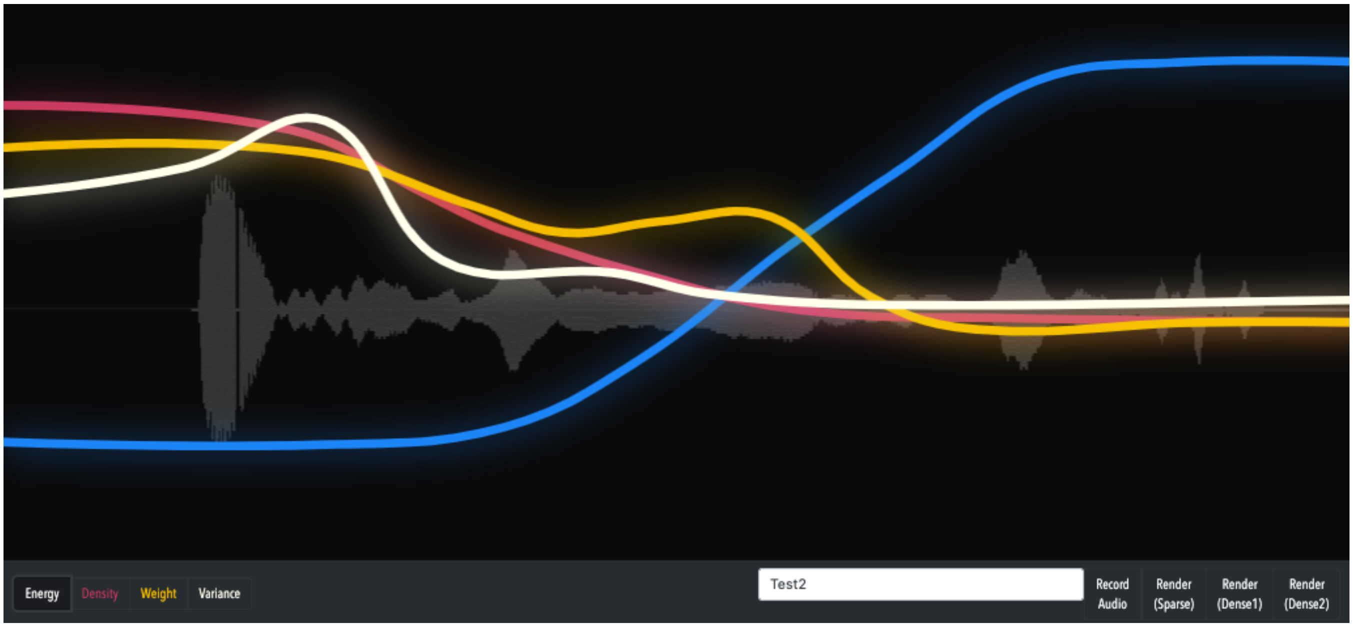
Publisher: ACM|26th International Conference on Intelligent User Interfaces

Persistent URL: https://hdl.handle.net/1721.1/145987

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

The Sound Sketchpad: Expressively Combining Large and Diverse Audio Collections

Nikhil Singh*
nsingh1@mit.edu
MIT Media Lab
Cambridge, Massachusetts



Test2

Figure 1: Prototype interface for the Sound Sketchpad. The grey waveform represents the input audio sketch, which the user supplies (for example, through vocalization) as a "template" for the resulting composition. Each colored contour represents a parameter that can be varied over the course of the sketch to shape the qualities of the output.

ABSTRACT

Software tools for media production have largely been adapted from physical media paradigms, offering blank canvases upon which to import, combine, and process content. In music production, this increasingly involves meticulous manual assembly of audio clips often carefully curated from diverse sources. As collections of audio content scale upwards in sample size, diversity, and number, creative projects require exponentially more time, effort, and attention to effectively shape them. New tools must find new ways to contend with this abundance of content. We propose the Sound Sketchpad, an algorithm-in-the-loop audio-graphical system and interface for combining sounds from a database into new music. It allows a user to sketch broad musical ideas by making sound, and then interactively modify and refine the resulting composition by drawing visual paths. We discuss the design, implementation, and advantages of this approach.

IUI '21, April 14–17, 2021, College Station, TX, USA
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8017-1/21/04.
https://doi.org/10.1145/3397481.3450688

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; • **Applied computing** → **Sound and music computing**; *Media arts*; • **Information systems** → *Users and interactive retrieval.*

KEYWORDS

music, sound, audio, media production, composition, creativity support, expressive tools

ACM Reference Format:
Nikhil Singh. 2021. The Sound Sketchpad: Expressively Combining Large and Diverse Audio Collections. In *26th International Conference on Intelligent User Interfaces (IUI '21), April 14–17, 2021, College Station, TX, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3397481.3450688

1 INTRODUCTION

1.1 Motivation

The composer Edgard Varèse once asked the question "what is music but organized noises?" [29] The proliferation of broadcast and mechanical reproduction technologies in the mid-twentieth century has given rise musical styles in which sounds—not notes—form the essential units from which music is built. Much music has explored the potential of such "recombinant" media. [13, 21] This is perhaps most notable in hip hop; beginning in the 1980s, hip hop DJs excerpted, reused, and combined segments of other recordings to produce new forms. [18]

In the age of big data, rich media and specifically recorded sound abounds in both volume and variety. While this growing wealth of content could offer more creative possibilities, audio software tools for composing with them lack support for spontaneity, agility, and flexibility needed in proportion. Instead, they rely on intensive and manual processes whose outcomes are difficult to adjust. Combining many sounds together to create new compositions requires intensive collection, observation, organization, and assembly. This project addresses the assembly part; rather than retrieving sounds from a file-system manually and placing them into a Digital Audio Workstation (DAW) for assembly, the user encodes their intent, regarding the outcome, in the form of bi-modal sketches, and these are resolved into easily adjustable recombinant compositions.

1.2 Contributions

This project supports a new media production workflow for interactively and iteratively creating recombinant musical compositions. This workflow is declarative and flexible. Users demonstrate a compositional idea, prescribe parametric control, and can quickly and easily create variations and develop compositions, in contrast to standard methods and interfaces (especially DAWs) that require extensive experience and laborious construction. This approach points to a path forward for media production tools to benefit from our growing expanse of material, while retaining expressive control.

2 BACKGROUND AND RELATED WORK

2.1 Graphical Sketching for Sound and Music

Goldschmidt describes sketching as a form of "interactive imagery", a process of visual reasoning through which form develops. [11] A number of computer-aided musical sketching interfaces have operated directly with sound, as far back as the first half of the twentieth century. [2, 16, 24]. In these devices, sketches act as control signals for synthesizers. Other systems apply sketching gestures as interfaces to collections of recorded sound, as ways to impose structure onto them. Examples of such projects include CataRT [27], earGram [5], the Infinite Drum Machine [17], the recent commercial product XO [15], and Constellation [28], in which drawing imposes structure by connecting sounds linearly and forming visual paths, creating persistent audio sketches.

Some drawing-based music composition systems instead opt for parametric forms of drawing. SonicExplorer [1] uses space, color, and gesture to explore a multidimensional space of audio parameters. Hyperscore [9] assigns parameters to drawn freehand contours, interpreting them as statements and elaborations of predetermined motifs in the symbolic domain. In DAWs, parametric notation is commonplace in the form of automation [4]. Based on adjusting hardware controls in real-time, digital automation takes on a new role, of interactively sculpting sound. In this paper, we explore the intersection of the generative, parametric systems with the structure-imposing audio systems.

2.2 Sonic "Sketching": Designing Sound with Provisional Audio Representations

Some systems have used vocalizations to parameterize sound synthesis engines. One project building on this idea is SkAT-VG [25], which aims to combine vocal sketching with gestural articulation to produce a bi-modal interface for creative sound design. Another way that audio examples are used for sound design is in Corpus-Based Concatenative Synthesis (CBCS) [26] systems. For example, AudioGuide [12] combines sound segmentation and source separation with a matching pursuit algorithm to mimic target audio with combinations of database sound segments.

Computer-aided orchestration systems need to represent a target with a combination of sound units, and often formulate this as a combinatorial optimization problem. Orchidée [8], for example, uses a multiobjective constrained optimization process. In our case, sketches reflect ideas rather than ground truth to be precisely reconstructed, and so we use a greedy algorithm and a simpler optimization approach combined with graphical control.

3 DESIGN PRINCIPLES

Our objectives are for the system to be:

1. **Flexible**: Olsen [20] describes flexibility as facilitating "rapid design changes that can then be evaluated by users." This means that our computational methods must be able to perform quickly. We absorb also the related goals of expressive leverage and match, by reducing the input space from detailed manual media import and assembly to guiding sketching across two modalities.
2. **Combinative**: Boden [6] describes the goal of combinatorial creativity as combining concepts to create novel ones. In our limited case of composition, we treat sounds as carefully curated basic building blocks to combine into new compositions. This is in contrast to generative models which mimic corpora, producing outputs from the input distribution, and to most corpus-based compositional tools which decompose sounds into smaller units for later linear concatenation.
3. **Extensible**: We consider two forms of extensibility, in the space of sounds and controls respectively. For sounds, the system should be able to handle large databases elegantly, and support their growth. For controls, new parameters should be able to be introduced to support a diversity of user practices, styles, and goals.
4. **Learnable**: We identify learnability in two ways. The first is that users should be able to learn the system's behavior, and leverage this experience creatively. The second is that any users, even those without experience in related tools, should be able to express with it in a way that supports any sounds they may be interested in. We can summarize this

by considering Resnick and Silverman's model of low floors, high ceilings, and wide walls [23], albeit within the space of sound-based musical compositions.

4 METHOD AND SOFTWARE IMPLEMENTATION

As noted, the Sound Sketchpad combines two input modalities. The first one is sonic: users can specify a sonic target, whether vocalized, performed on an instrument, or otherwise. This target specifies an initial "sound world", establishing a general context and trajectory for the collage. The second is graphical; users can draw patterns that take on roles as parameters, guiding specific aspects of the composition.

4.1 Data Preprocessing

When an audio sketch is supplied, we first extract a number of time-varying audio features, depending on the Essentia [7] library to do so. For information on common audio features and those referred to later in this work, we refer the reader to excellent available audio feature reviews [19, 22], and Essentia's own documentation [1], for details.

The process of assembling compositions can be quite intensive. This requires an initial selection of, ideally < 100, pertinent sounds from the larger database. This additional limitation helps to keep the interaction agile, as well as focuses the qualities of the compositions around the sketch audio. The system relies on either pre-selected sound collections, from the full database, or it automatically selects a number of sketch-relevant sounds. This is currently primarily done by predominant pitch, which we compute as the median of estimated fundamental frequencies with greater-than-mean estimation confidence.

We then transform these into options. We first retrieve a multi-feature matrix F_s for each sound, where $F_s \in \mathbb{R}^{k \times n}$ ($k = N_{features}, n = N_{frames}$), and produce multiple versions of each where $len(sound) < len(sketch)$. We time-shift feature matrices in increments of $q = \frac{len(sketch)}{10}$, and zero-pad to $k \times n$, maintaining a corresponding list of sound sources and offsets. This avoids sketch pre-segmentation and a time-offset variable, allowing multiple (time-shifted) instances of a sound and simplifying the algorithms needed for the arrangement process.

The system also works to estimate each feature's relevance to the given sketch. We do this categorically, as is enumerated below. Given the importance scores S, we apply the softmax function so that $\sum_{x \in S} = 1$, and then assign the resulting weights to the relevant features.

- **Pitch**: P = the portion ($\in [0, 1]$) of filtered pitch values that are non-zero.
- **Harmony**: Based on the Shannon entropy H of the time-averaged chroma [3] vector K: $(1 - \frac{H(K)}{3})(1 - P)$.
- **Timbre**: $\left(\sigma_C^2(\frac{max(C) - min(C)}{2})^2 + \frac{\mu_S}{3 \times 10^7}\right)(1 - P)$ where C denotes the spectral centroid, σ_C^2 denotes its variance, and μ_S represents the mean spectral spread.

4.2 Graphical Controls

We specify and implement two different forms of control parameters. Feature controls influence the template-matching process. These are implemented as a dictionary that maps the parameter name to a set containing one or more audio feature names, which the parameter contours respectively replace. The second type is processor controls, which apply additional post-processing to the output audio. These are implemented as a dictionary mapping a parameter name to a function of the audio signal and the control parameter contour which returns the processed audio signal. This format makes it trivial to introduce new expressive parameters as desired, based on context or additional experimentation. Currently implemented parameters are:

1. **Density**: The textural complexity of the output, with larger values favoring more timbrally diffuse and dissonant combinations of sounds (spectral spread and dissonance).
2. **Variance**: The amount of spectral change as a function of time (spectral flux).
3. **Weight**: The output spectrum's center of mass over time (spectral centroid).
4. **Energy**: Gain envelope for the assembled audio, giving it dynamic shape over time ($f(x, l) = x \circ l$).

4.3 Assembling Compositions

The first method, which results in collages with a relatively sparse texture, is a simple greedy algorithm. In each iteration, it seeks to find and add one sound option to the arrangement that most helps it more closely match the template, if any. Or, we seek the sound from options L, with template T, template-scaling constant c (defaults to 10), number of features k, arrangement A, and feature-weight w:

$$m = argmin \left\{ \sum_{i=0}^{k} || \, cT_i - (A + X_i) \, ||_2 w_i : X \in L \right\} \tag{1}$$

The second method results in sound collages with a much denser texture, with a relatively large number of sound instances. It is also generally slower, and should ideally be used with a small, carefully curated collection of sounds. For this method, we formulate a continuous optimization problem using simulated annealing [14], a probabilistic method for optimization. The design variable s here reflects amplitude per sound $\in [0, 1]$. We also introduce ρ or density. In assembling audio after optimization, sounds assigned an amplitude below $1 - \rho$ are not included, so this parameter acts as a kind of filter. As a preprocessing step, we initialize s with the sparse method's selections ($\subseteq \{0, 1\}$) plus a small amount of noise. This method tries to minimize, with limited iterations, the KL-divergence between template features and combined features:

$$\sum_{i=0}^{k} D_{KL} \left(\sum_{j=0}^{N} s_j L_{ji} \; || \; cT_i \right) w_i \quad s.t. \quad 0 \leq s \leq 1 \tag{2}$$

4.4 User Interface

The interface is implemented in React.js. User-drawn contours are smoothed and rendered with the Konva library. The screen shows the sketch and audio sketch waveform, with line selection

[1] https://essentia.upf.edu/documentation.html

and rendering functionality at the bottom. A prototype version showing the lines, sketch waveform, and controls can be seen in fig. 1. Sketched contours are interpolated and scaled, and sent along with the audio sketch to the backend application. A modal view handles additional functionality, including preview and download for the assembled composition and original sketch audio.

5 EVALUATION AND PROJECT DISCUSSION

We examine the proposed workflow in comparison to existing systems for sound-based composition with regard to each of our previously-cited design goals, to offer insight into its advantages.

Flexibility. As noted, traditional DAWs based on the tape machine paradigm provide support for media file import, recording, editing, and mixing. Users compose by combining these basic functions to assemble new pieces of music. The Sketchpad, by automating the assembly and providing descriptive inputs, diverts the focus of the user from the *how* to the *what*. Additionally, with the Sketchpad, iteration is trivial, and requires only changes in easily drawn contours.

Combination and Scale. Our goal is to increase the availability of large sound collections to creative media production *in situ*. The underlying sound database supports this; new sounds and collections can be added, and they remain available to future work. In addition, this expanding pool of resources is applied directly, yielding swift mechanisms for combination both in sequence and in layers. This improves upon DAW patterns and existing sound databases, such as Freesound [10], which largely require manual search, downloading, importing, and editing before their contents can then be composed with. It also offers advantages over CBCS approaches, in that it is scalable, supports both linear and vertical combinations, allows interactive refinement, and benefits from a growing ecosystem of content.

Extensibility. Our parameter control framework easily admits new controls, that can be designed as desired to map onto features or apply audio post-processing. The former builds on target-based control in CBCS and the latter follows from DAW parametric automation. This combination supports both control over the assembly process and the output aesthetics respectively, and allows arbitrary extension, limited only by possible feature mappings and signal processing. Notably, discovering new parameters is not a simple process, and so our initial implementation provides a few to begin with.

Learnability. The bi-modal interface allows simple, expressive inputs and facilitates expression by new populations, specifically those without training or experience in music composition or media production. Additionally, because we don't depend on learned black box models, the processes are more interpretable in the sense that a user might become accustomed to the system's behavior, and discover new ways to maneuver its constraints into controllable output and, ultimately, new interesting output forms.

These qualities together enable the system to provide creative access to large amounts of material, while the interface makes this access easy to learn, experiment with, and extend for new users and musical applications.

6 LIMITATIONS AND FUTURE WORK

This interface and its methods, by expecting pre-selection of sound collections from the database, are able to operate relatively quickly to generate new compositions from inputs. The process of curating independent sound units and organizing them into collections is still very cumbersome, however, and ongoing work aims to address these particular tasks and connect them to the Sketchpad. Additionally, automated selection methods are currently somewhat inflexible; future work might consider more situation-specific search criteria.

Another important challenge is effectively handling stylistic information and variation. The Sketchpad aims to account for overall and gradual variations in texture, pitch, and envelope, as are typical in soundscape composition and related styles. Future work may extend these techniques to consider important structural elements of other recombinant musical styles, such as precise rhythm, melody, and harmony, and imagine how new directions might be possible for these by empowering creators with new scales of content and powerful tools for production with them.

7 CONCLUSION

In this work, we examined the design, implementation, and possible applications of a new interface and tool for sound-based music composition with large, diverse audio collections. In doing so, it is hoped that the problems, principles, and techniques explored can motivate more work in designing creative media production tools suitable for the age of big data, that thrive on scale and diversity and support emerging forms of expression.

REFERENCES

[1] Alexander Travis Adams, Berto Gonzalez, and Celine Latulipe. 2014. SonicExplorer: Fluid Exploration of Audio Parameters. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 237–246. https://doi.org/10.1145/2556288.2557206

[2] Irina Aldoshina and Ekaterina Davidenkova. 2016. The History of Electro-Musical Instruments in Russia in the First Half of the Twentieth Century. (01 2016).

[3] M. A. Bartsch and G. H. Wakefield. 2001. To Catch a Chorus: Using Chroma-Based Representations for Audio Thumbnailing. In *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*. 15–18.

[4] David Bawiec. 2018. What Is Mix Automation? Everything You've Been Too Afraid to Ask. [Online]. Available from: https://www.izotope.com/en/learn/what-is-mix-automation.html.

[5] Gilberto Bernardes, Carlos Guedes, and Bruce Pennycook. 2013. EarGram: An Application for Interactive Exploration of Concatenative Sound Synthesis in Pure Data. In *From Sounds to Music and Emotions*, Mitsuko Aramaki, Mathieu Barthet, Richard Kronland-Martinet, and Sølvi Ystad (Eds.). 110–129.

[6] Margaret Boden. 01 Jan. 2009. *Chapter Thirteen. Creativity: How Does It Work?* Brill, Leiden, The Netherlands, 235 – 250. https://doi.org/10.1163/ej.9789004174443.i-348.74

[7] Dmitry Bogdanov, Nicolas Wack, Emilia Gómez, Sankalp Gulati, Perfecto Herrera, Oscar Mayor, Gerard Roma, Justin Salamon, José R. Zapata, and Xavier Serra. 2013. Essentia: An Audio Analysis Library for Music Information Retrieval. In *ISMIR*.

[8] Grégoire Carpentier, Gérard Assayag, and Emmanuel Saint-James. 2010. Solving the Musical Orchestration Problem Using Multiobjective Constrained Optimization with a Genetic Local Search Approach. *Journal of Heuristics* 16, 5 (Oct. 2010), 681–714. https://doi.org/10.1007/s10732-009-9113-7

[9] Morwaread M. Farbood, Egon Pasztor, and Kevin Jennings. 2004. Hyperscore: A Graphical Sketchpad for Novice Composers. *IEEE Comput. Graph. Appl.* 24, 1 (Jan. 2004), 50–54. https://doi.org/10.1109/MCG.2004.1255809

[10] Frederic Font, Gerard Roma, and Xavier Serra. 2013. Freesound Technical Demo. In *Proceedings of the 21st ACM International Conference on Multimedia* (Barcelona, Spain) *(MM ’13)*. Association for Computing Machinery, New York, NY, USA, 411–412. https://doi.org/10.1145/2502081.2502245

[11] Gabriela Goldschmidt. 1991. The Dialectics of Sketching. *Creativity Research Journal* 4, 2 (1991), 123–143. https://doi.org/10.1080/10400419109534381

[12] Benjamin Hackbarth, Norbert Schnell, and Diemo Schwarz. 2010. AudioGuide: A Framework for Creative Exploration of Concatenative Sound Synthesis. (2010).

[13] Paul Harkins. 2016. Microsampling: From Akufen’s Microhouse to Todd Edwards and the Sound of UK Garage. In *Musical Rhythm in the Age of Digital Reproduction*. 177–194.

[14] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. 1983. Optimization by Simulated Annealing. *science* 220, 4598 (1983), 671–680.

[15] Malte Kobel. 2019. The Drum Machine’s Ear: XLN Audio’s Drum Sequencer XO and Algorithmic Listening. *Sound Studies* 5, 2 (2019), 201–204. https://doi.org/10.1080/20551940.2019.1661163

[16] Peter Manning. 2012. The Oramics Machine: From Vision to Reality. *Organised Sound* 17, 2 (2012), 137–147. https://doi.org/10.1017/S1355771812000064

[17] Kyle McDonald, Manny Tan, and Yotam Mann. 2017. The Infinite Drum Machine. [Online]. Available from: https://experiments.withgoogle.com/drum-machine.

[18] Kembrew McLeod. 2002. How Copyright Law Changed Hip Hop: An Interview with Public Enemy’s Chuck D and Hank Shocklee. *Stay Free Magazine* 20 (2002).

[19] Dalibor Mitrović, Matthias Zeppelzauer, and Christian Breiteneder. 2010. Features for Content-Based Audio Retrieval. In *Advances in computers*. Vol. 78. Elsevier, 71–150.

[20] Dan R. Olsen. 2007. Evaluating User Interface Systems Research. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology* (Newport, Rhode Island, USA) *(UIST ’07)*. Association for Computing Machinery, New York, NY, USA, 251–258. https://doi.org/10.1145/1294211.1294256

[21] John Oswald. 1985. Plunderphonics, or Audio Piracy as a Compositional Prerogative. In *Wired Society Electro-Acoustic Conference, Toronto*.

[22] Geoffroy Peeters. 2004. A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project. *CUIDADO IST Project Report* 54, 0 (2004), 1–25.

[23] Mitchel Resnick and Brian Silverman. 2005. Some Reflections on Designing Construction Kits for Kids. In *Proceedings of the 2005 conference on Interaction Design and Children*. 117–122.

[24] Emily Robertson. 2010. *"It Looks Like Sound!": Drawing a History of "Animated Music" in the Early Twentieth Century*. Master’s thesis. University of Maryland, College Park.

[25] Davide Rocchesso, Guillaume Lemaitre, Patrick Susini, Sten Ternström, and Patrick Boussard. 2015. Sketching Sound with Voice and Gesture. *Interactions* 22, 1 (Jan. 2015), 38–41. https://doi.org/10.1145/2685501

[26] D. Schwarz. 2007. Corpus-Based Concatenative Synthesis. *IEEE Signal Processing Magazine* 24, 2 (2007), 92–104.

[27] Diemo Schwarz, Grégory Beller, Bruno Verbrugghe, and Sam Britton. 2006. Real-Time Corpus-Based Concatenative Synthesis with CataRT. In *9th International Conference on Digital Audio Effects (DAFx)*. 279–282. https://hal.archives-ouvertes.fr/hal-01161358

[28] Akito van Troyer. 2013. Constellation: A Tool for Creative Dialog Between Audience and Composer. In *Proceedings of the International Symposium on Computer Music Multidisciplinary Research*. 534–541.

[29] Edgard Varèse and Chou Wen-chung. 1966. The Liberation of Sound. *Perspectives of New Music* 5, 1 (1966), 11–19. http://www.jstor.org/stable/832385