SketchSynth: a browser-based sketching interface for sound control

Sebastian Löbbers
Centre for Digital Music
Queen Mary University of London
London, United Kingdom
s.lobbers@qmul.ac.uk

György Fazekas
Centre for Digital Music
Queen Mary University of London
London, United Kingdom
g.fazekas@qmul.ac.uk

ABSTRACT

SketchSynth is an interface that allows users to create mappings between synthesised sound and a graphical sketch input based on human cross-modal perception. The project is rooted in the authors’ research which collected 2692 sound-sketches from 178 participants representing their associations with various sounds. The interface extracts sketch features in real-time that were shown to correlate with sound characteristics and can be mapped to synthesis and audio effect parameters via Open Sound Control (OSC). This modular approach allows for an easy integration into an existing workflow and can be tailored to individual preferences. The interface can be accessed online through a web-browser on a computer, laptop, smartphone or tablet and does not require specialised hard- or software. We demonstrate SketchSynth with an iPad for sketch input to control synthesis and audio effect parameters in the Ableton Live digital audio workstation (DAW). A MIDI controller is used to play notes and trigger pre-recorded accompaniment. This work serves as an example of how perceptual research can help create strong, meaningful gesture-to-sound mappings.

Author Keywords
cross-modal perception, sound sketching, gesture-to-sound mapping

CCS Concepts
• Applied computing → Sound and music computing; Psychology;
• Human-centered computing → User interface programming;

1. INTRODUCTION

How can a digital music performer or producer control sound in a way that does not require the sequential adjustment of parameters or scrolling through sample or preset libraries? Within the NIME community, this question is addressed by a variety of works that incorporate gesture-to-sound mappings including Auraglyph [17], which harnesses handwriting recognition algorithms on modern touch devices; Soundpainting [2] which uses a Microsoft Kinect for recognising high-level gestures from full-body movement; Handmate [13], a browser-based hand-tracking controller; and the Gestural Sound Toolkit [1], an accessible toolkit for designers. These projects either fix their mappings to a specific use-case or leave it to the user to create mappings from largely generic features. SketchSynth extracts visual features from a sketch input relevant to human cross-modal perception of sound. We argue that this approach can help create meaningful mappings more easily, while maintaining a degree of creative freedom. The interface runs entirely in the browser and can be accessed online on a computer, laptop, tablet or smartphone. Features are made available to digital synthesisers and audio effects via Open Sound Control (OSC). Through this modular design SketchSynth can be integrated into the digital music setup of a user who can rely on their experience with sketching for a familiar form of interaction. Additionally, it provides a strong audio-visual connection that can help communicate a digital music performance to an audience. Compared to sketch-to-sound systems like SonicDraw [3], our work focuses on controlling the quality or timbre of a sound, rather than pitch or temporal structure.

In this demo paper, we shortly introduce the research into sound-shape associations and sketch recognition that informs this project in Sections 1.1 and 1.2. Section 2 gives an overview of the interface and extracted features. Section 3 demonstrates how SketchSynth could be used with a concrete example using a synthesiser and delay audio effect in Ableton Live. Section 4 considers future improvement and application.

1.1 Sound-shape associations

Sound-shape associations are a subset of cross-modal associations that describe how people link stimuli from the visual and auditory modality. Spence [19] and Salgado et
Our research suggests that figurative representations are more frequently used for familiar sounds like acoustic instruments and depict real-life objects or scenes. Abstract representations are more closely linked to sound-shape associations and common for synthesised textures.

al. [18] give a comprehensive overview of the topic, showing that, while influenced by personal factors, similarities in associations exist between people. The bouba/kiki effect, illustrated in Figure 1, is an example of a sound-shape association that was observed across different cultures and demographics [2]. Perceptual research commonly asks participants to match existing stimuli, but recent work investigated similarities in sketched responses to music and sound [11, 18, 5]. Knees and Andersen [9] first proposed sketch input for the retrieval of sound in an exploratory study. Expanding on this idea, we conducted several studies on sound-sketching prior to the development of SketchSynth. In our first study [14] we identified two high-level sound-sketch categories, figurative and abstract, which are described in Figure 2. Focusing on abstract representations of synthesised sounds, the second study [15] found several correlations between sketch and auditory features between participants. However, results also suggested that similarities are greater for representations created to semantic prompts like Draw a noisy sound. For our third study [16], we trained a deep classifier to distinguish between noisy and calm sketches and mapped them to a semantically annotated synthesiser dataset. Participants were then asked to rate the sounds that were returned to their sketch input. While the classifier correctly categorised the majority of sketches, participant ratings of proposed sounds were mixed. Figure 3 shows sketches collected in this last study.

1.2 Sketch recognition

Digital sketches are typically represented as a sequence of points that are connected to create a rasterised sketch image. Information about the number of strokes, sketch length, position and size can be extracted directly from this data structure. Increasingly popular are deep learning approaches for sketch recognition. Convolutional neural networks (CNNs) were shown to outperform traditional methods on rasterised sketches from the MNIST dataset [12, 4]; seminal work by Ha and Eck [7] introduced the Quick, Draw! dataset and the Sketch-RNN architecture for sketch classification and generation. Research by Engeln et al. [6] suggests that deep learning can be used to retrieve sound samples from sketches.

2. DEVELOPMENT

This section describes the setup illustrated in Figures 4 and 5. The SketchSynth interface can be accessed online at https://sketchsynth.com.

2.1 Interface design

The interface was developed alongside our perceptual research in a series of three online design studies that included between 10 and 15 participants. In reaction to feedback collected through surveys and click-stream data, we stretched the canvas to the browser window to maximise sketch space and used Catmull-Rom splines for smoother sketching. A sketch fades out over time, which allows for continuous sketching, and limits its maximal length. The latter was shown to encourage abstract sketches that are more closely linked to sound-shape associations, as shown in Figure 2.

Figure 4: Setup using an iPad for sketch input that sends OSC messages to a Max4Live patch to manipulate synthesised sound in Ableton Live for a demonstration that can be viewed at https://youtu.be/4Yzv2rgsjOE

Figure 5: Flowchart of SketchSynth pipeline.

https://p5js.org/reference/#/p5/curveVertex
Table 1: Sketch features extracted by the SketchSynth interface and their auditory correspondences based on perceptual research that largely informed the mapping in this demonstration. The first two rows describe a sketch as a whole while the remainders focus on specific aspects of it.

Table 1:

<table>
<thead>
<tr>
<th>Sketch feature</th>
<th>Method</th>
<th>Correspondence</th>
<th>Our mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>noisiness</td>
<td>deep classifier trained on noisy/calm semantic sound-sketches</td>
<td>timbral texture [15]</td>
<td>wavetable position to interpolate between a sine and triangle wave; creating a “buzzier” sound for higher values</td>
</tr>
<tr>
<td>thinness</td>
<td>deep classifier trained on thin/thick semantic sound-sketches</td>
<td>timbral mass [15]</td>
<td>high-pass filter cut-off frequency; a higher value removes low frequencies</td>
</tr>
<tr>
<td>width/height of bounding box</td>
<td>difference between sketch points with min/max x/y position</td>
<td>loudness, perceived size of sound source [19]</td>
<td>width → wavetable unison detune; creating a wider spatial image for higher values. height → sub-bass level; creating a fuller sound for higher values</td>
</tr>
<tr>
<td>center x-position</td>
<td>half-way distance between sketch points with min/max x-position</td>
<td>position of sound source</td>
<td>left/right panning</td>
</tr>
<tr>
<td>number of strokes</td>
<td>length of stroke array in sketch data structure</td>
<td>granularity of sonic texture [14]</td>
<td>delay feedback; creating a granular texture for higher values</td>
</tr>
<tr>
<td>length of sketch</td>
<td>sum of sketch points in all strokes</td>
<td>roughness/hardness              [15]</td>
<td>volume; input is clamped to reach the maximum volume after 10 sketch points. This is a pragmatic mapping tailored to the performance that does not strictly follow perceptual correspondences.</td>
</tr>
<tr>
<td>sketching speed</td>
<td>euclidean distance between the last 3 three sketch points divided by time passed between them (0 if not currently sketching)</td>
<td>roughness/hardness              [15]</td>
<td>phaser effect rate; creating a jittery sound for higher values</td>
</tr>
</tbody>
</table>

2.2 Sketch analysis and mapping suggestions

SketchSynth provides features that describe a sketch on a macro level to define the general “nature” of a sound, and on a micro level to allow for more nuanced manipulation. This was achieved through the following methods:

- **macro**: deep-learning classification that captures the overall appearance of a sketch and corresponds to perceptual categories of a sound (noisy or calm, thin or thick).
- **micro**: algorithmically extracted features that describe specific aspects of a sketch, like position or structure which were shown to correlate with sound characteristics.

Table 1 lists all features in detail, including their cross-modal correspondences and the mapping used in our demonstration. These should be considered perceptually-informed suggestions; a user has the flexibility to select all or a subset of features for their mapping in which they might adjust sketch-to-sound connections to meet individual preferences or requirements. In addition to the listed features, we include a work-in-progress real-time corner point extractor using ShortStraw. While it did not prove reliable enough for this demonstration, it has the potential for strong sound-shape mappings in future development.

2.3 Implementation

The sketch interface including feature extraction runs entirely in the browser. The JavaScript libraries React and p5.js were used for the graphical user interface, and the deep classifiers were implemented in TensorFlow.js. The osc-js library was used for sending OSC messages via the WebSocket protocol. We provide a Max4Live patch that connects SketchSynth to Ableton Live running a Node.js Websocket server in a node.script Max object. The code is available on GitHub [https://github.com/SFRL/sketch-synth](https://github.com/SFRL/sketch-synth).

3. DEMONSTRATION

Figure 4 shows the setup of our demonstration including a linked video. We used the Serum wavetable synthesizer and a native delay plugin in Ableton Live for sound generation. A MIDI controller was used to play notes while sketching on an iPad. The demonstration first explores the performance, it also creates a “lag” that can make it difficult to facilitate immediate changes.

- The interface is sufficiently reactive and reliable to be used for sound manipulation in real-time.
- Our mapping resulted in a strong and sensible audio-visual connection.
- Because a sketch fades out, the extracted features might change even when not sketching. While it introduces an interesting element of anticipation during the performance, it also creates a “lag” that can make it difficult to facilitate immediate changes.

- Sketching Speed provided the most immediate control over sound. Introducing similar features could counterweigh the “lag” mentioned above. The experimental corner point extractor mentioned in Section 2.2 could be a suitable candidate.

References:

https://xferrecords.com/products/serum
• Sketching and playing a MIDI keyboard simultaneously might prove difficult for a user who is not trained to use their hands independently. A different setup could use a recorded MIDI track or include two performers.

4. CONCLUSION AND FUTURE WORK
This paper demonstrates how perceptual research can drive the development of interfaces for sonic interaction. SketchSynth provides features for meaningful shape-to-sound mappings through an accessible, lightweight implementation that only relies on hard- and software available to an average digital music practitioner. Our demonstration only serves as a proof-of-concept and evaluation with potential users is needed to assess how SketchSynth could be integrated into an existing practice. While our demonstration leans towards a performance context, it can also be imagined as an engaging tool for collecting perceptual sketch data. Future work will refine and extend feature extraction and consider implementing a synthesis model with the AudioWorklet[1] interface to make SketchSynth explorable entirely in the browser while continuing to provide an option for OSC communication.

5. ACKNOWLEDGMENTS
EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology (EP/L01632X/1).

6. ETHICAL STANDARDS
All studies that contributed to the development of SketchSynth were approved by the Queen Mary University of London Ethics Committee (Reference numbers QMREC2341 for [14], QMERC20.478 and QMERC20.009 for [15] and QMERC20.594 for [19]). All participants gave informed consent. No monetary compensation was offered except for Prolific consent. No monetary compensation was offered except for QMERC20.594 for [16]). All participants gave informed consent. No monetary compensation was offered except for QMERC20.478 and QMERC20.009 for [15] and consent. No monetary compensation was offered except for QMERC20.594 for [19]).

7. REFERENCES


