GrainTrain: A Hand-drawn Multi-touch Interface for Granular Synthesis

Anıl Çamcı
Department of Performing Arts Technology
University of Michigan
acamci@umich.edu

ABSTRACT

We describe an innovative multi-touch performance tool for real-time granular synthesis based on hand-drawn waveform paths. *GrainTrain* is a cross-platform web application that can run on both desktop and mobile computers, including tablets and phones. In this paper, we first offer an analysis of existing granular synthesis tools from an interaction stand-point, and outline a taxonomy of common interaction paradigms used in their designs. We then delineate the implementation of *GrainTrain*, and its unique approach to controlling real-time granular synthesis. We describe practical scenarios in which *GrainTrain* enables new performance possibilities. Finally, we discuss the results of a user study, and provide reports from expert users who evaluated *GrainTrain*.

Author Keywords

Granular synthesis; hand-drawn; multi-touch; cross-platform; interaction design

CCS Concepts

•Human-centered computing → Web-based interaction; Sound-based input / output; •Applied computing → *Sound and music computing;*

1. INTRODUCTION

Granular synthesis as an audio production technique has a robust history dating back to the 1950s [6]. With the introduction of real-time granular synthesis systems in the 1980s [10], this technique made its way into performance practices. Today, there are numerous software and hardware tools for granular synthesis that adopt different computational and interactive approaches.

In this paper, we first offer an interaction taxonomy of real-time granular synthesizers to contextualize our work. By analyzing 20 granular synthesizers, we elaborate 3 common paradigms of interaction design across these instruments. We discuss the properties of these paradigms, and outline their implications in terms of musical style and expression.

We then introduce, *GrainTrain*, an innovative implementation of real-time granular synthesis based on hand-drawn waveform paths. Designed as a cross-platform web application, *GrainTrain* opens up interesting performance possibilities. In addition to mouse interaction on desktop computers, it supports multi-touch interaction, which has been applied to granular synthesis relatively recently with the introduction of mobile devices that utilize capacitive touch screens.

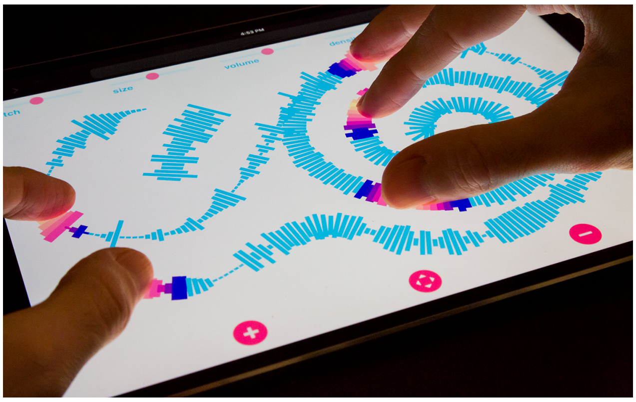

Figure 1: Image of *GrainTrain* in use on a tablet computer. The performer is using 5 fingers to interact with 4 waveform paths that are hand-drawn in various shapes and spatial configurations.

As its primary contribution, *GrainTrain* enables users to create custom hand-drawn interfaces for granular synthesis. We discuss the implications of this approach in terms of multi-touch interaction ergonomics and performance techniques. We then offer the results of a user study, and an analysis of the feedback gathered from expert users.

2. RELATED WORK

In one of the earliest applications of granular synthesis, the composer Iannis Xenakis spliced together short pieces of tape, and played them back at high speeds to create a dense cloud of sound grains [7]. With advances in digital computing, it became possible to program a computer to output a granulated version of an audio file based on pre-determined parameters. Since then, numerous real-time applications of granular synthesis that adopt different interfaces and processing techniques have been developed. Although these applications rely on a similar principle of synthesis based on the micro-organization of sound elements, they can implement vastly different approaches to how the user controls this process. The mouse being the predominant input device on modern computers, most granular synthesis applications are designed around point, click and drag interactions.

On the other hand, the widespread adoption of multi-touch screens in consumer-grade mobile devices has ushered

NIME'18, June 3-6, 2018, Blacksburg, Virginia, USA.

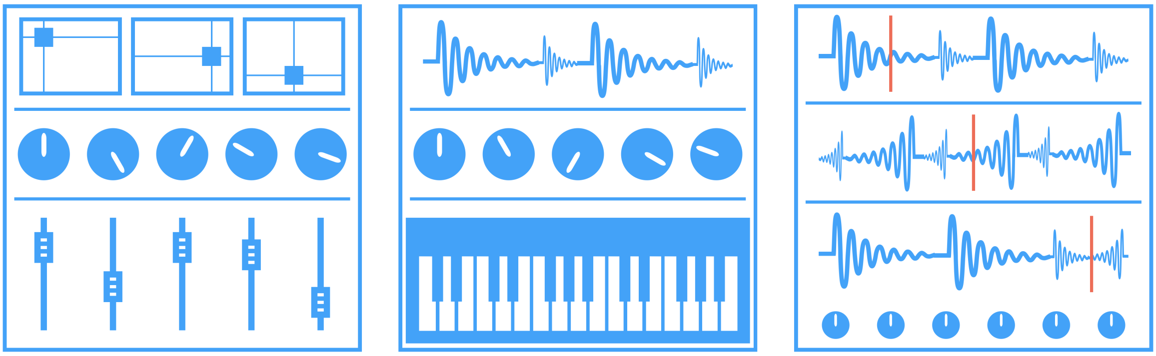

Figure 2: Interface archetypes for the three interaction paradigms for granular synthesis: parameter control (left), keyboard performance (middle), waveform scrubbing (right).

in a flurry of new musical applications, including those of granular synthesis. Some of these applications emulate the mouse-based desktop interfaces, where touch actions essentially serve as individual mouse cursors. However, there are also applications that rely on the unique possibilities of multi-touch interaction, which we will discuss in the following sections.

In addition to software UIs, other interaction modalities are used for granular synthesis as well. *The PebbleBox*, for instance, relies on tactile interactions, where the sounds of a user playing with pebbles in a box are analyzed in real-time to extract parameters for the granulation of arbitrary sounds [5]. *ShakeStick* is a physical interface in the form of a wooden stick equipped with an accelerometer; by waving the stick, the user can control various granulation parameters that are mapped to the pitch, yaw and roll values of the stick [13]. In *Node Kara*, a depth camera is used to track human movements in 3D space, which are then used to drive various parameters of a granular synthesis engine [3].

3. AN INTERACTION TAXONOMY OF GRANULAR SYNTHESIZERS

Despite the great variety of UIs implemented in granular synthesis applications, common threads across these can be identified. We reviewed 20 real-time granular synthesizers that have been released over the past 18 years. These include SoundGrain 6 (2017), Generative 2 (2015), Borderlands Granular 2 (2015), iDensity 2 (2015), The Mangle (2015), Granulator II (2013), GrainProc (2012), MegaCurtis (2012), SAMPLR (2012), HourGlass (2012), Grain Science (2011), Granite (2011), SampleWiz (2011), SampleToy (2010), Narrativas Sonoras II (2010), MetaSynth 5 (2009), CataRT (2007), Partikel (2007), Emission Control (2004), and Granulab (2000). Among this list are desktop applications that are designed for mouse or keyboard input, and mobile applications that are designed for touch input. Moreover, some of these applications can be controlled externally via MIDI or OSC.

All of these applications offer creative interfaces that support unique forms of musical expression. Examining these interfaces, we elaborate three main interaction paradigms for real-time granular synthesizers: parameter control, keyboard performance, and waveform scrubbing. In Fig.2, we offer illustrations of archetypal interfaces for these 3 categories. Although we define these interaction paradigms separately, many granular synthesizers combine these approaches with different emphases. In Fig. 3, we provide few examples of how existing applications can be situated within this taxonomy.

3.1 Parameter Control

Control of granulation parameters is intrinsic to most granular synthesizers; common granulation parameters include window size and type, density, pitch, and amplitude. Some of the applications we reviewed prioritizes the performance of changes in these parameters. In such applications, knobs, sliders, and X-Y controllers take up most of the interface. Examples of these applications include *Emission Control* [7] and Partikkel [1]. *GrainProc* similarly emphasizes slider-based control of parameters, but with the intent of enabling the use of toes to manipulate the output of an instrument, which the performer is playing with their hands [8].

These interfaces commonly facilitate the detailed manipulation of continuous grain streams. Although a slider for either explicit or probabilistic control of the playhead position can also be found in these interfaces, this slider often occupies the same level of interaction hierarchy as those of other parameters.

The focus on the manipulation of continuous granulation make these interfaces especially powerful for exploring gradually evolving qualities of microsound. This form of interaction renders this UI paradigm particularly suitable for generating textural sounds. However, more sparse or gestural sounds can also be achieved with specific parameter combinations and by using sound sources that display such qualities.

3.2 Keyboard performance

Applications under this category utilize an interface that is based on the keyboard instrument model. The interface often includes a virtual piano keyboard, or can be controlled externally with a midi keyboard. Pressing a key triggers a grain stream that is transposed accordingly. Knobs and sliders in these UIs serve a more ancillary role.

In these applications, granular synthesis serves a similar function as the oscillator of a subtractive synthesizer, rather than as a means to explore the emergent qualities of microsound processing. From a stylistic point of view, these interfaces are often used for playing pad-like sounds that consist of chordal combinations of grain clouds. While these can be better suited for performing textural sounds, the user can also perform more gestural sounds by playing phrases with shorter notes.

For instance, in *MegaCurtis*, the user can move a playhead on a waveform to determine the point of granulation. The user then plays a virtual keyboard, which takes up most of the UI, to synthesize granular streams that are transposed according to the keys pressed. Similarly, in *SamopleWiz*, the user can either play a virtual keyboard or scrub over a piano-roll to granulate a file at different transpositions.

3.3 Waveform Scrubbing

In these applications, mouse or touch interactions are used to move a playhead across the waveform of an audio file. By scrubbing through the file, the user can change the point at which the grain windowing is applied. In this mode of interaction, the emphasis is put on the temporal exploration of an audio file by scanning it for parts that respond to granulation in different ways. Scrubbing movements can often lead to gestural sounds due to changes in spectral and transient characteristics of a sound over time. Sustained or textural sounds can also be achieved by maintaining the position of the playhead.

For instance, in *Borderlands Granular* for iOS, the user creates visual granulation nodes, under which they can place multiple waveforms. By controlling the parameters of a node, the user can change how it interacts with the waveforms that it coincides with. Waveforms can also be resized and rotated the to alter the range of granulation. The developer of this application describes his approach as putting the emphasis on "gestural interaction over knobs and sliders" [2].

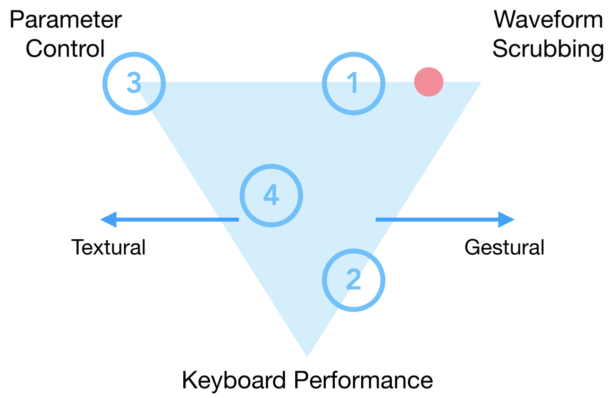

Figure 3: 4 examples of existing applications situated within the interaction taxonomy discussed here. 1: Borderlands Granular emphasizes waveform scrubbing while also offering a separate interface state for the control of grain parameters. 2: MegaCurtis interface heavily relies on a virtual piano keyboard but the user can also move the playhead on the waveform to change the point of granulations triggered with the keyboard. 3: Emission Control gives the user a complex combination of sliders and X-Y controllers for the fine manipulation of a grain cloud, 4: Granulator 2 plug-in for Ableton Live offers an interface consisting of multiple knobs and sliders. The grains can be set to loop or be triggered with a MIDI keyboard. Although the plug-in does not offer a direct waveform-scrubbing interaction, the playhead position can be controlled with a knob. The pink dot represents *GrainTrain*.

4. GRAINTRAIN

GrainTrain is a novel application of granular synthesis based on multi-touch interaction. Within the context of the interaction taxonomy described above, *GrainTrain* offers an innovative approach to the waveform-scrubbing paradigm with hand-drawn waveform paths as seen in Fig. 4. This allows the user to create custom interfaces that open up unique expressive possibilities, some of which are described later in this paper, and demonstrated in our video abstract. [1]

[1] https://vimeo.com/graintrain/video

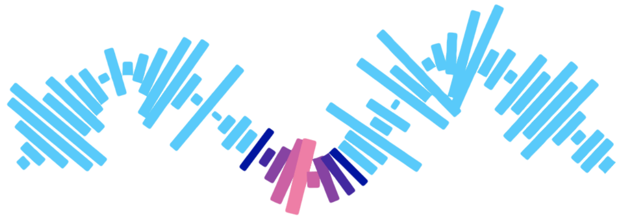

Figure 4: An interactive waveform in *GrainTrain*. The point of interaction is highlighted with gradual changes in bar color and thickness. The extent of these changes is based on the spread parameter.

4.1 System Design

GrainTrain is designed for the web browser using the WebAudio API, the WebGL library Three.js, HTML5 and CSS. *GrainTrain* therefore runs on multiple hardware platforms (i.e. desktop and mobile computers) and operating systems (e.g., iOS and Android). It adopts a fully client-side operation, which does not require a network connection after the system has been loaded. The user can load any audio file from their local file system without a need for uploading files to a remote server. On mobile devices that do not offer a user-accessible file system, cloud storage services such as iCloud or Google Drive can be used for loading files into *GrainTrain*. The native file input UI of the operating system is used for file selection on all platforms.

4.2 Interface

On start-up, *GrainTrain* offers a simple interface with 3 buttons and 5 sliders. In a sense, *GrainTrain* launches with a lack of a UI, which the user can gradually construct and customize during the course of a performance as seen in Fig. 1. The 3 buttons at the bottom of the screen are used for adding, moving, and deleting sound files. Upon pressing the add button, the user is prompted to select a sound file from local or cloud storage. After selecting a file, the user can draw an arbitrary path anywhere on the screen. Once the drawing is complete, the waveform of the selected file is drawn on this path. After pressing the move button, the user can relocate any of the waveforms already drawn on the screen. Finally, the delete button is used for removing existing waveforms. A similar mapping of an audio file to a hand-drawn path is used in *Different Strokes*, where the drawing speed determines the speed of the file playback [14].

Waveforms in *GrainTrain* are made up of bars that represent momentary amplitudes in the audio file at regular intervals as seen in Fig. 4. Each waveform, regardless of its path length, represents the entirety of the file. The number of bars in a waveform depends on the length of the path drawn. This implies that a longer path for the same file will offer a higher temporal resolution for interaction. When a mouse or touch action collides with any of the bars in a waveform, the file gets granulated at the corresponding position. The range around this point from which grains are selected is represented with gradually changing colors.

The 5 sliders at the top of the screen gives the user control over size, amplitude, pitch, density, and spread of the grains. The spread parameter controls the range from which grains are selected around the point of interaction on a waveform. Additionally, touch force is mapped to the density parameter on touchscreen devices. Although we position *GrainTrain* within the waveform-scrubbing paradigm outlined earlier, we believe that these five fundamental parameters of granulation unlock a range of sonic possibilities, and are therefore integrated into our UI albeit with minimal footprint. Unlike most parameter-control applications, where the user manipulates a continuous stream of grains, a stream in *GrainTrain* is generated only when the user scrubs a waveform.

4.3 Ergonomics of Touch Interaction

Over the past decade, touch interfaces have become a standard in mobile computing. While most of our interactions with touchscreen devices still rely on single-point controls similar to those performed with a mouse, modern phones and tablets also implement multi-touch interactions [12]. However, multi-touch interactions with 2D surfaces bring about ergonomic restrictions [4]. This is primarily why most multi-touch interactions rely on either the duplication of single-point interactions (e.g., two-finger swipes and taps), or the imitation of interactions with physical objects (e.g., pinch and spread gestures).

Most granular synthesizers within the waveform-scrubbing paradigm utilize a 1-dimensional timeline, where the playhead is used to traverse an audio file by moving a cursor horizontally on this timeline. Some synthesizers also map the vertical position of the cursor to various parameters, such as pitch or amplitude. A mouse interaction is perfectly sufficient to perform these one or two-dimensional actions. With touch surfaces, concurrent control of multiple timelines or parameters becomes possible. However, multiple horizontal timelines can pose ergonomic problems due to variations in how different fingers can be extended [11]. A primary goal in designing *GrainTrain* was to enable the creation of non-linear waveform trajectories that can better suit the anatomy of a user's hand. This way, the user can more easily perform simultaneous granulations of multiple audio files with multi-touch gestures.

4.4 Synthesis Engine

GrainTrain's synthesis engine is based on a windowed playback of asynchronous sample buffers. The hop size between windows is determined by the grain size and density parameters. Each mouse or touch interaction with a waveform instantiates a new voice, which is maintained until the interaction is completed. The user can create concurrent streams of granulation through multi-touch or by interacting with overlapping portions of two or more waveforms as seen in Fig 5a.

The user's interaction with a waveform is quantized to the bars that make up a waveform. With each interaction, the bar index is used to determine the position in the audio buffer to be sampled. This position is updated as the user moves their mouse or finger from one bar to another. Grains are picked from a user-controllable range around this position in a randomized fashion, which alleviates repetition artifacts that might arise due to the quantization.

5. PERFORMANCE TECHNIQUES

GrainTrain enables a combination of new and existing interaction techniques that expand the expressive possibilities of granular synthesis on mobile and desktop computers. Here, we outline a few of the performance techniques that are based on hand-drawn waveform paths and multi-touch interaction.

5.1 Superimposition of multiple waveforms

An arbitrary number of waveforms can be drawn on top of each other. When the user interacts with a point where two or more waveforms intersect, grains from all audio files that correspond to these waveforms will be picked. *Borderlands Granular* facilitates a similar interaction via overlapping audio files with different orientations [2].

Even on a desktop browser that offers a single point of interaction, unique expressions can be achieved by superimposing multiple waveforms as seen in Fig. 5a. At the top, three different waves are intersected. At the bottom, a single file is drawn twice from left to right and right to left. This allows the user to mix forward and reverse playbacks of a file. When applied on a multi-touch device, this technique greatly expands the number of concurrent granulations, and opens up distinct mixing possibilities.

5.2 Forking interaction paths

Drawing partially overlapping curved paths for different sound files enables an interesting gesture, where using a single swipe on a forking path will cross-fade between multiple audio files. In, Fig. 5b, the user starts granulating one file but switches over to another waveform with a single gesture as indicated by the dashed line.

5.3 Discontinuous scrubbing

Waveform scrubbing on a linear timeline often results in phrases that are continuous regardless of the direction of scrubbing. On a curved waveform, such as the one seen in Fig. 5c, the user can perform a continuous scrubbing action on the dashed line that would result in discontinuous granulations that would have required discrete interactions on a linear timeline.

5.4 Short-touch interactions

In addition to continuous mouse or touch gestures, the user can also play the waveforms with brief, trigger-like interactions. Using an ergonomically shaped arc as seen in Fig. 5d, the user can perform gestures that resemble playing trills on a piano.

5.5 Indeterminacy through complex paths

By drawing complex curves as seen in Fig. 5e, the user can obfuscate the relationship between the temporal progression of a sound and the waveform that represents it. This creates a degree of interaction indeterminacy that leads to improvisational possibilities. Especially when the drawing takes up a large portion of the screen, it approximates the effect of the audio being randomly scattered over the UI. This allows the user to perform arbitrary gestures that will result in unpredictable combinations of grains but within the confines of a single audio file.

5.6 Intra-mixing

Intra-mixing (i.e. the mixing of an audio file with itself) is enabled by drawing paths that loop in winding or circular shapes. In Fig. 5f, the user has drawn several circles in a loop. When the user interacts with any point on the drawing, grains from various parts of the waveform that correspond to that point are selected, effectively mixing discrete times in the same audio file. When used with multi-touch interaction, this extracts complex temporal structures from a single audio file.

5.7 Temporal resolution through path length

The user can control the temporal resolution at which they interact with a file by altering the length of the path the waveform is drawn onto. In Fig. 5g, the same audio file has been mapped to two drawings. While the top spiral allows the user to explore the file in finer temporal detail, the shorter line facilitates the picking of grains from a wider temporal range. In Fig. 5d, the four paths underneath the arc similarly allows the user to trigger grains from the entire span of an audio file. Resizable waveforms in *Borderlands Granular* enables a similar technique.

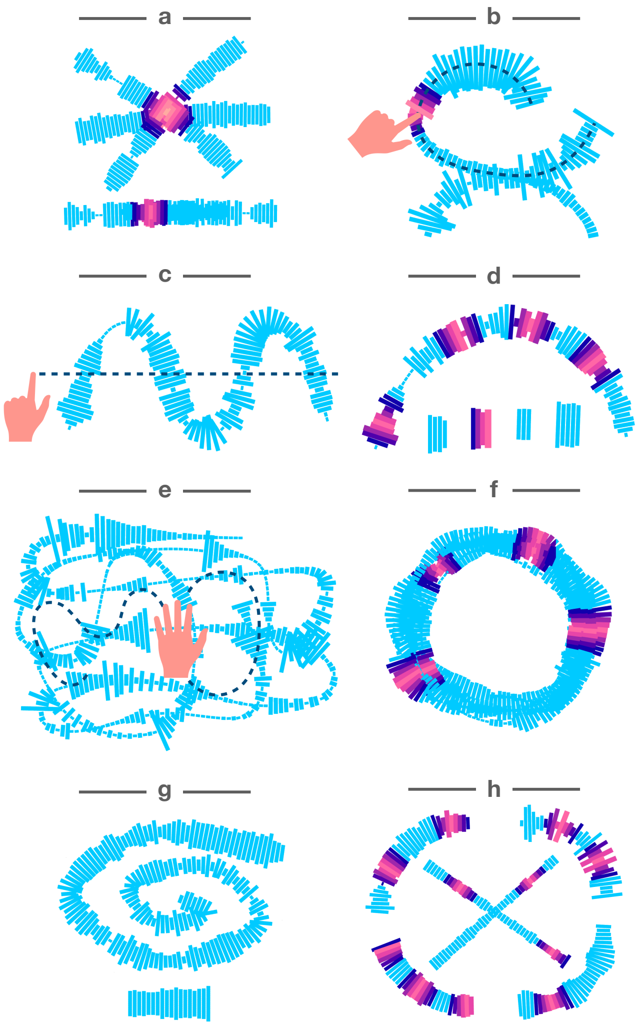

Figure 5: Illustrations of various performance techniques enabled by *GrainTrain*'s hand-drawn waveform paths, and multi-touch interactions.

5.8 Multi-user interaction

While the multi-touch capabilities of a device inherently imply that multiple users can interact with it simultaneously, a UI that is geared towards a single user often obstructs multi-user operation. The hand-drawn UI elements in *GrainTrain* offers the flexibility to create custom layouts that can facilitate multi-user interactions. In Fig. 5h, four users have demarcated their corners on a tablet using curved waveforms. Additionally, two diagonal waveforms serve as shared UI elements between pairs of users.

5.9 Sustain

Due to the way *GrainTrain*'s synthesis engine implements the relationship between user interaction and grain generation, vibrato-like gestures result in the triggering of multiple grain streams while finger position is maintained. Especially when the spread parameter is set to minimum, this gesture prolongs playback in a way that is similar to how the same gesture sustains the sound of a stringed instrument.

6. EVALUATION

To evaluate the usability and the expressive capabilities of *GrainTrain*, we conducted a study with novice users, and reached out to expert users for feedback.

6.1 User Study

To evaluate certain use cases of *GrainTrain*, and its effectiveness in facilitating the use of waveform scrubbing for granular synthesis, we conducted a study with 10 users. 3 of the participants described themselves as having no prior knowledge of granular synthesis. Each study was performed on a 10.9" iPad Pro, and took approximately 20 minutes.

6.1.1 Method

Each user was given a 1-minute tutorial on the operation of *GrainTrain*. They were then asked to use the application in the following scenarios: 1) a single linear waveform, 2) 2 linear waveforms on two separate rows, 3) a single arched waveform, 4) 2 arched waveforms side by side. In structured interviews following each scenario, the users were asked to report their general impressions, the maximum number of fingers from a single hand that felt comfortable to use, and how two-handed performance fared on a tablet screen. Finally, they were asked to create a free-form interface, play with it, and verbally describe their overall experience.

6.1.2 Results and Discussion

With a single linear waveform, users unanimously reported 3 as the maximum number of fingers that felt comfortable to perform with. 7 users reported that 4 fingers were possible to place on a linear timeline but uncomfortable to scrub with. While a few of the users attempted placing 5 fingers on the linear timeline, none reported this as a viable interaction method. We observed that all users were able to perform naturally with one or two fingers, and most users comfortably maintained 3 fingers on a single linear waveform as they scrubbed through it. With a fourth finger placed, although keeping them in position was possible, users struggled to keep the little finger in contact with the waveform when scrubbing.

With two linear waveforms spaced apart in rows, users tried scrubbing with two hands. While the reports on the number of fingers that can be placed comfortably on a waveform remained similar to that from the first scenario, some users also attempted cross-placement of fingers (e.g., the thumb of the top hand touching the bottom waveform). The most problematic aspect in this scenario was reported as hands getting in the way of each other when moved in opposite directions. Users reported that being able to further separate the waveforms vertically made it more comfortable to perform this action, implying that even with more traditional waveforms, some flexibility in the UI is preferred.

With a single arched waveform, most users immediately placed all 5 fingers on the waveform. Furthermore, they reported that scrubbing with 4 or 5 fingers were comfortable. Being able to draw arcs that fit their hand size was found to significantly improve their ability scrub with multiple fingers. Another common gesture was playing the arc like a piano with brief touch events.

With two arched waveforms, most users were able to place all 10 fingers comfortably, with the exception of 2 users whose hands were too large to fit on the tablet surface. These users instead used two arcs crossing each other roughly were the thumbs would be placed. One of these users performed a zipper-like finger interlocking as two hands moved in opposite directions on the arcs.

During the free-form design task, all users came up with distinct interfaces, implying that *GrainTrain* can support a variety of approaches to musical expression. All of the users drew abstract non-linear paths using 2 or more sounds without any supervision. Only 3 users interacted with the grain-parameter sliders. Almost all of the users experimented with paths that intersect in various ways. Once a user settled on a design, their interaction with it ranged from brief playful explorations to extended performance-like session. Furthermore, given that these were all first-time users, some of whom did not have previous experience with granular synthesis, we find the complexity of their designs particularly encouraging in terms of *GrainTrain*'s potential to afford a low-barrier and a high ceiling for usability, which is deemed a desirable trait for interfaces that support creativity [9]. All users reported having enjoyed playing with *GrainTrain*.

6.2 Expert Feedback

Additionally, we reached out to 5 expert users, who described themselves as having used real-time granular synthesis in performance contexts, or offered instruction on the topic. These users were briefed on the functionality and the design of *GrainTrain*. They were then asked to evaluate it in their own time, and document their experiences with screenshots, recordings, and written notes. Further feedback was gathered via follow-up interviews.

One of the experts described that they used *GrainTrain* to alter the way they physically interacted with their tablet device. By drawing waveforms to the left and right edges of the screen, they were able to hold the device like an accordion with the screen facing outwards, and perform with short-touch interactions. This user also mentioned experimenting with representational drawings, and that it was interesting to think about such visual metaphors as UI elements.

Another expert mentioned that they preferred drawing wavy lines as these functioned as space-filling curves that increased the extent of a waveform within the confines of a tablet screen. They mentioned that while complex shapes came with a mental cost, wavy lines allowed them to explore sounds easily and intuitively.

Another expert argued that being able to scan complex shapes for interesting sound combinations which they could then riff on was the most engaging affordance of the UI. They described that exploring how various shapes and sounds wound up together often lent itself to powerful contrasts between gestural and textural sounds. Similarly, another expert described that creating random scribbles and playing them with multiple fingers were at times more expressive than playing with a predictable visual pattern such as a line.

Another expert mentioned that the ability to fill up the screen with many interface elements of different shapes and sizes (e.g., regions, spirals, buttons) was inspiring from a performance stand point. They've also expressed that moving waveforms in space to have them gradually overlap each other proved to be an interesting way to think spatially about the temporal progression of a performance.

7. FUTURE WORK AND CONCLUSION

In the short term, we aim to address user requests such as automatic dynamics management, waveform duplication, scene saving and loading, and separate color schemes for individual waveforms. We also plan to map the saturation of each bar's color to the spectral centroid of the audio it represents to create another channel of visual feedback. A feature which we believe will enhance the live performance capabilities of our system is the ability to swap the file that is attached to a path. Additionally, we are investigating UI schemes for implementing per-waveform control over grain parameters. As we expand the features of our UI, we fully intend to maintain its low barrier of entry for novice users.

Since *GrainTrain* is a cross-platform web application, online collaboration is a natural next step in our development. With this feature, users will be able to access a common instance of *GrainTrain* remotely to mutually interact with the same set of waveforms.

In this paper, we presented an interaction taxonomy for granular synthesizers that can support the analysis of existing applications as well as the design of new ones. With *GrainTrain*, we introduced a new approach to the waveform-scrubbing paradigm with interactive waveforms that are drawn onto custom paths created by the user. We identified some of the performance techniques that this approach enables. Furthermore, the feedback we gathered from users indicated the potential of *GrainTrain* in facilitating other new and interesting expressive possibilities for granular synthesis.

8. REFERENCES

[1] Ø. Brandtsegg, S. Saue, and T. Johansen. Particle synthesis–a unified model for granular synthesis. In *Proceedings of Linux Audio Conference*, 2011.

[2] C. Carlson and G. Wang. Borderlands: An audiovisual interface for granular synthesis. In *Proceedings of International Conference on New Interfaces for Musical Expression (NIME)*, 2012.

[3] A. Çamcı and A. Forbes. Node kara: An audiovisual mixed reality installation. In *IEEE VR Workshop on Mixed Reality Art*, pages 1–4, 2016.

[4] E. Hoggan, M. Nacenta, P. O. Kristensson, J. Williamson, A. Oulasvirta, and A. Lehtiö. Multi-touch pinch gestures: Performance and ergonomics. In *Proceedings of ACM International Conference on Interactive Tabletops and Surfaces*, pages 219–222, 2013.

[5] S. O'Modhrain and G. Essl. Pebblebox and crumblebag: Tactile interfaces for granular synthesis. In *Proceedings of International Conference on New Interfaces for Musical Expression (NIME)*, pages 74–79, 2004.

[6] C. Roads. *Microsound*. MIT press, 2004.

[7] C. Roads. Grains, forms and formalization. In S. Kanach, editor, *Xenakis Matters*. Pendragon Press, 2012.

[8] M. Sanganeria and K. Werner. Grainproc: A real-time granular synthesis interface for live performance. In *Proceedings of International Conference on New Interfaces for Musical Expression (NIME)*, pages 223–226, 2013.

[9] B. Shneiderman. Creativity support tools: Accelerating discovery and innovation. *Communications of the ACM*, 50(12):20–32, 2007.

[10] B. Truax. Real-time granular synthesis with a digital signal processor. *Computer Music Journal*, 12(2):14–26, 1988.

[11] W. Westerman. *Hand tracking, Finger Identification, and Chordic Manipulation on a Multi-touch Surface*. PhD thesis, University of Delaware, 1999.

[12] W. Westerman, J. G. Elias, and A. Hedge. Multi-touch: A new tactile 2-d gesture interface for human-computer interaction. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 45(6):632–636, 2001.

[13] A. Wilson and A. Hindle. The shake stick. Technical report, PeerJ PrePrints, 2015.

[14] M. Zadel and G. P. Scavone. Recent developments in the different strokes environment. In *Proceedings of International Computer Music Conference (ICMC)*, 2008.