DOLPHIN: A Framework for the Design and Perceptual Evaluation of Ultrasound Mid-Air Haptic Stimuli

Lendy Mulot
lendy.mulot@ens-rennes.fr
Univ Rennes, ENS Rennes
Rennes, France

Guillaume Gicquel
guillaume.gicquel@irisa.fr
CNRS, Univ Rennes, Inria, IRISA
Rennes, France

Quentin Zanini
quentin.zanini@ens-rennes.fr
Univ Rennes, ENS Rennes
Rennes, France

William Frier
william.frier@ultraleap.com
Ultraleap
Bristol, United Kingdom

Maud Marchal
maud.marchal@irisa.fr
Univ Rennes, INSA, IRISA, Inria, CNRS
Rennes, France

Claudio Pacchierotti
claudio.pacchierotti@irisa.fr
CNRS, Univ Rennes, Inria, IRISA
Rennes, France

Thomas Howard
thomas.howard@irisa.fr
CNRS, Univ Rennes, Inria, IRISA
Rennes, France

ABSTRACT

Ultrasound mid-air haptic interfaces can display highly reconfigurable vibrotactile shapes in mid-air for human-computer interaction (HCI) applications. The choice of stimulus shape, spatial, temporal and modulation parameters yields a complex design space, yet relatively little is known about the impact of these design choices on perceived stimulus properties. We define the combination of a spatial discretization of an abstract shape and a set of rules for the temporal display order and intensity modulation of the resulting points as a *sampling strategy*. We developed DOLPHIN, an open-source framework to aid in designing mid-air stimuli. DOLPHIN allows the study of the impact of rendering parameters on perceived stimulus properties. This platform-agnostic framework standardizes stimulus descriptions as a step toward more replicability and easier communication in the field. It enables reproduction of stimuli between perceptual experiments and ensures stimuli used in applications correspond to those evaluated in prior perceptual studies. We validated DOLPHIN's usability by conducting a user study assessing the impact of sampling strategy design on curvature discrimination for dynamic mid-air haptic stimuli. The Weber fractions for just-noticeable differences (JNDs) in curvature were found to range between 1 and 1.4, yet no significant effect of the number of spatial sampling points on curvature discrimination was found. This result shows a practical use-case for DOLPHIN and provides insight into rendering mid-air haptic curvature.

SAP '21, September 16–17, 2021, Virtual Event, France

https://doi.org/10.1145/3474451.3476232

CCS CONCEPTS

• **Software and its engineering → Software design engineering**; • **Human-centered computing → Haptic devices**.

KEYWORDS

Mid-air haptics, Stimulus design, Perceptual evaluation

ACM Reference Format:
Lendy Mulot, Guillaume Gicquel, Quentin Zanini, William Frier, Maud Marchal, Claudio Pacchierotti, and Thomas Howard. 2021. DOLPHIN: A Framework for the Design and Perceptual Evaluation of Ultrasound Mid-Air Haptic Stimuli. In *ACM Symposium on Applied Perception 2021 (SAP '21), September 16–17, 2021, Virtual Event, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474451.3476232

1 INTRODUCTION

Ultrasound mid-air haptic (UMH) interfaces are a popular and novel paradigm for tactile rendering, allowing easy reconfiguration of stimuli without the need for any device held or worn by the user [Rakkolainen et al. 2020]. The high spatial and temporal reconfigurability these devices provide comes at the cost of a steep learning curve for stimulus designers, with many parameters coming into play, such as stimulus design (static, dynamic, passive or active), modulation method, as well as spatial and temporal sampling strategies. Indeed, stimuli can either be designed as static, where the user perceives the whole shape at once, or dynamic, where the user perceives spatial motion of the focal point or shape as a function of time [Hajas et al. 2020]. A stimulus can also be rendered passively, i.e. independently of the user's interaction with it, or actively, by updating the displayed portion of the stimulus based on user tracking [Dzidek et al. 2018]. The interfaces' transducers emit waves at frequencies which are not perceptible by human touch (usually 40 kHz [Carter et al. 2013; Iwamoto et al. 2008] or 70 kHz [Ito et al. 2016]) and thus require some form of modulation for tactile rendering. Three forms of modulation are employed to this end. Amplitude modulation (AM) [Iwamoto et al. 2008] consists in varying the focal point's intensity at a rate perceptible by the user. Spatio-temporal modulation (STM) [Kappus and Long 2018]

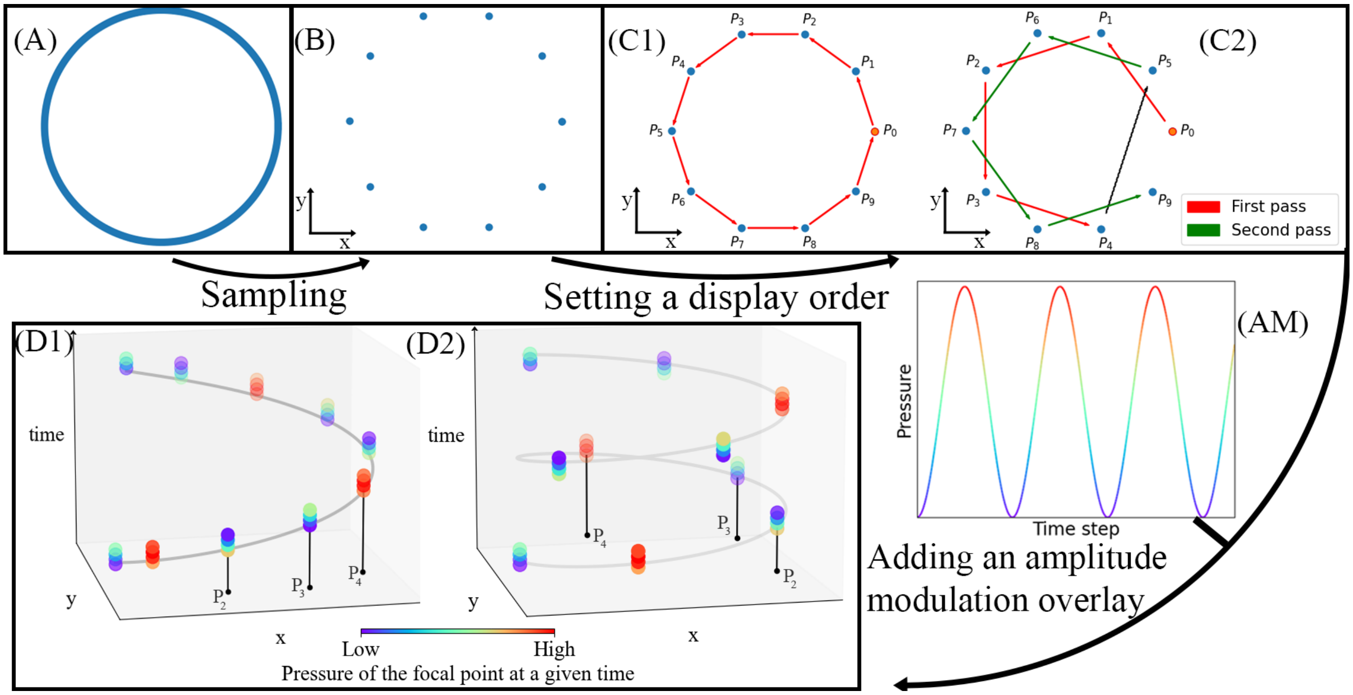


Figure 1: The three steps for applying a sampling strategy to a shape. The process begins with (A) an abstract shape, here a circle. The shape is then (B) spatially sampled. (C1) and (C2) show examples of possible display orders of the sampled points. Finally, an intensity modulation overlay (AM) is applied. (D1) and (D2) show the application of (AM) to the display orders respectively shown in (C1) and (C2). The vertical axis corresponds to time. Points are color-coded with respect to their intensity at a given display time step, with weak intensities shown as blue and high intensities as red. Dwell time is a third of the AM period.

uses a focal point rendered at full intensity and rapidly moved along a path so as to circle back to its original position at a rate corresponding to the desired draw frequency. Lateral modulation (LM) [Takahashi et al. 2018] is similar to STM, with the focal point moving back and forth along a path. Compared to AM, both STM and LM produce stronger sensations, and allow drawing of tactile shapes with many more points. AM and STM can also be combined (e.g. [Hajas et al. 2020; Howard et al. 2019]). Finally, a given abstract shape can be displayed in different ways, depending on how it is spatially sampled, and how the resulting display points are temporally rendered. We define a *sampling strategy* as a spatial sampling of an abstract geometric shape combined with rules dictating the temporal display order of the points and an optional overlay of amplitude modulation with respect to time [Frier et al. 2019]. For any given shape, different sampling strategies will induce different sensations. Fig. 1 illustrates the process of turning an abstract shape into its mid-air tactile representation using sampling strategies. Little is known about the perceptual implications of the design choices described above. Furthermore, the immense design space for UMH stimuli complicates their study, as well as the replication of studies.

In this paper, we present DOLPHIN, an open-source framework which standardizes and facilitates the design and evaluation of UMH rendering of geometric shapes. Currently, DOLPHIN focuses on the design and study of passive 2D stimuli, whether they are static or dynamic. This framework deals only with rendering of UMH stimuli but it can be used in conjunction with frameworks such as PsychoPy to create multimodal experiments. The core of the framework is a stimulus designer tool that lets the user easily design complex stimuli using customizable geometric shapes and sampling strategies. The tool also enables simulation of the resulting pressure distributions on the skin and stimulus rendering using an UMH interface. Currently, only Ultraleap interfaces are fully supported but the hardware-agnostic design of the framework will allow us to add support for more devices in the future and lets any user implement the required functions to add support for their own. Stimuli can then be saved and played back in other applications or in perceptual experiments using the reader component of the framework. Fig. 2 sums up the framework's components.

This paper presents the following contributions:

- A framework aiming to make the design of ultrasound stimuli accessible to everyone, without the need for strong programming knowledge and experience with UMH.
- A general and standardized approach for stimulus design and corresponding data representations for stimuli and sampling strategies. This ensures reusability of stimuli between applications and facilitates reproduction of experiments.
- A validation of our framework's functionality through a proof of concept experiment, showing the application of DOLPHIN to the study of curvature discrimination for dynamic UMH stimuli.

2 RELATED WORK

UMH can be used in a variety of HCI applications such as virtual [Martinez et al. 2018; Matsubayashi et al. 2019] and augmented [Dzidek et al. 2018; Romanus et al. 2019] reality, digital signage [Georgiou et al. 2019] and gestural interfaces [Large et al. 2019]. This implies rendering of stimuli with different geometric and temporal properties, which in turn requires understanding the user's ability to discriminate between these properties. Our framework aims to allow easy configuration of perception evaluation experiments to study and help design mid-air tactile feedback.

The need for accessible stimulus design tools is not unique to UMH. Many open-source and commercial stimulus design tools exist for contact haptics. For example, Syntacts [Pezent et al. 2020] allows hardware-agnostic vibrotactile stimulus creation from audio, while Apple's Core Haptics[1] also provides a complete API to design haptic stimuli for their mobile devices. However, the design needs for contact vibrotactile devices are not the same as for UMH, since UMH stimulus design has to deal with spatial aspects in addition to the temporal and amplitude aspects of the vibrotactile stimuli. To the best of the authors' knowledge, no open-source framework currently exists to design and evaluate UMH stimuli. Some UMH stimulus design software currently exists, such as the Ultraleap Sensation Editor[2], which provides a variety of pattern presets, some customization options, and the possibility to export configured stimuli. However, it is designed as a demo and rapid prototyping tool and thus does not provide in depth control over sampling strategies or any simple way of interfacing with 3rd-party psychophysical study or rendering applications. With such a tool, certain aspects of stimulus design are hidden to the user, which can cause misunderstanding of the nature of the stimuli for non-experts aiming to perform psychophysics work on UMH stimuli. An important goal of DOLPHIN is thus to standardize the stimulus design workflow and to explicitly show the user all design parameters, to ensure full control over the nature of the stimuli. The most notable 3rd-party evaluation tools for UMH are open-source acoustic field simulation tools (Ultraino[3], HandyBeam[4]), which have been widely used to estimate device and rendering performances during the design phase (e.g. [Carter et al. 2013; Hasegawa and Shinoda 2013; Hoshi 2011; Inoue and Shinoda 2014]). These can be considered complementary to the current framework, allowing for simulation and visualization of the physical effects of rendering choices. Open-source tools for psychophysics experiments also exist, the most notable being PsychoPy [Peirce et al. 2019], which we chose to complement DOLPHIN during our evaluation (see Sec. 5).

Perception studies on UMH stimuli are an active area of research with multiple objectives: (A) understanding the technology's performances and limits for rendering with respect to human perception, (B) optimizing and improving the quality of rendering, (C) validating novel rendering and interaction techniques and (D) investigating fundamental questions relating to human haptic perception. The majority of these studies focus on passively displayed UMH stimuli, which is why at its core, DOLPHIN focuses on the design of such stimuli.

(A) Understanding UMH device performances. Studies in this field focus on determining stimulus detection and discrimination thresholds in relation to stimulus rendering parameters. Several studies have investigated the relationship between peak focal point pressure and stimulus detection for AM [Hoshi 2011; Howard et al. 2019; Iwamoto et al. 2008; Takahashi et al. 2019], LM [Takahashi et al. 2019] and STM [Howard et al. 2019] stimuli. Stimulus detection was also investigated as a function of other parameters such as stimulus duration [Gil et al. 2018; Hoshi et al. 2009a], AM frequency [Carter et al. 2013; Hasegawa and Shinoda 2018; Raza et al. 2019], location on the skin [Chongyang et al. 2019; Gil et al. 2018; Mizutani et al. 2019; Takahashi et al. 2018] and with respect to the array [Raza et al. 2019]. Perception studies have been conducted to assess discrimination of stimuli intensities [Marchal et al. 2020], duration [Hoshi et al. 2009a], locations [Vo and Brewster 2015], and number [Alexander et al. 2011; Carter et al. 2013]. Work has also been conducted on discriminating properties of geometric primitives. Carter et al. [Carter et al. 2013] performed an initial study to determine the 2-point threshold for AM stimuli at different modulation frequencies. Inoue et al. [Inoue et al. 2015] investigated the JND for angles between pairs of lines displayed in 3D using multiple AM points, as well as the JND for line position in 3D. Finally, multiple studies have focused on the perception of dynamic stimuli. Hoshi et al. [Hoshi et al. 2009b, 2010] studied the identification and discrimination of movement directions for linear, circular, spiral, and random AM focal point motions. Using Ultraleap's Sensation Editor, Rutten et al. assessed the identification of both static and dynamic AM focal point [Rutten et al. 2019], and later the discrimination between intensities and velocities of dynamic AM mid-air haptic patterns [Rutten et al. 2020].

(B) Optimizing rendering parameters. UMH interfaces have the drawback of achieving limited sensation intensity, delivering comparably weaker stimuli than contact vibrotactile actuators. Some work based on perception studies has been conducted with the objective of optimizing sampling strategy parameters to improve perceived stimulus intensity. Korres et al. [Korres et al. 2017] studied the perceptual implications of sampling strategy design for sequential displays of AM focal points along circular shapes. Similarly, for STM shapes, Frier et al. [Frier et al. 2018, 2019] studied how the draw rate, number of sampling points and dwell times at each point influenced users' subjective perception of sensation intensity for circles of different sizes.

(C) Validating novel approaches to rendering and interaction. UMH interfaces have spawned a multitude of novel tactile rendering and interaction techniques, for tactile user interfaces [Ito et al. 2019; Rümelin et al. 2017], mid-air tactile icons [Rocchesso et al. 2019; Yoshimoto et al. 2019] as well as virtual and augmented reality. The effectiveness of these techniques needed to be evaluated through perceptual experiments. In the context of mid-air interaction with 2D shapes, multiple experiments on shape identification and perceived shape properties were conducted using both static and dynamic, AM and STM and hybrid stimuli [Ablart et al. 2019; Hajas et al. 2020; Hoshi 2012] In the latter study, the authors show that shape recognition rates are far higher when displayed with a dynamic tactile pointer than when displayed as a static STM stimulus. The possibility for displaying stimuli in a 3D volume above the

[1] https://developer.apple.com/documentation/corehaptics
[2] https://developer.ultrahaptics.com/downloads/sensation-editor/
[3] https://github.com/asiermarzo/Ultraino
[4] https://github.com/ultraleap/HandyBeam

array has naturally led to perception studies on rendering algorithms for interaction with virtual 3D shapes using AM ([Inoue and Shinoda 2014; Long et al. 2014]) , STM ([Martinez et al. 2019; Matsubayashi et al. 2019]) or hybrid approached ([Howard et al. 2019]). There is currently no work on displaying volumetric 3D UMH stimuli. Interactions with volumetric objects are rendered through projections into 2D planes (Long et al. [Long et al. 2014], Matsubayashi et al. [Matsubayashi et al. 2019]). This is explained by the fact that UMH rendering is limited by device energy output, and therefore stimulus designers usually focus on concentrating energy at the surface of a user's skin, resulting in a 2D rendering problem. This observation motivated our focus on the design of 2D geometric stimuli within DOLPHIN.

Geometric UMH stimuli with different sampling strategies also find applications beyond rendering of purely geometric properties. For example, Marchal et al. [Marchal et al. 2020] showed that intensity variations in STM shapes can be used to render sensations of virtual object stiffness and Sakiyama et al.

(D) Investigating fundamental questions pertaining to human haptic perception. The fact that UMH interfaces provide contactless and highly-reconfigurable vibrotactile stimuli makes them interesting candidates for investigating haptic perception of stimuli with complex spatio-temporal properties. Reardon et al. [Reardon et al. 2019] studied how wave propagation at the surface of the skin generated by single-pass STM patterns affects tactile perception of motion. Perquin et al. [Perquin et al. 2021] used dynamic STM patterns to study discrimination of motion direction and find evidence of directional bias in the process. Furthermore, the dynamic reconfiguration of stimuli allows for the investigation of perceptual effects resulting from this. For example, Pittera et al. [Pittera et al. 2019] investigated illusory tactile movement between two hands depending on focal point frequency, stimulus onset asynchrony and direction of movement for pairs of fixed focal points as well as moving focal points. Morisaki et al.

The studies discussed here highlight the wide variety of stimulus parameters used in rendering and studied as experimental variables, which is behind the motivation for designing a tool that enables stimulus design and provides control over each of these parameters through a unified framework and workflow. To the best of our knowledge, all psychophysics studies involving mid-air haptic stimuli have used either custom stimuli written "from scratch" or they have restricted themselves to the use of a limited set of standard stimuli available thanks to, e.g., commercial stimulus generation tools such as the Ultraleap Sensation Editor (e.g. [Rutten et al. 2020, 2019]). All of these works could have benefited from a framework to facilitate design and later reuse of stimuli. While for the most part, the cited studies provide sufficient information to reproduce the stimuli used, standardized design tools, software representations of stimuli and perceptual evaluation tools would ease the process of reproducing and expanding upon their results.

3 DESCRIPTION OF THE STIMULUS DESIGN

An UMH stimulus has two main aspects. The first aspect is spatial, and concerns the stimulus' geometry as well as its position in space, as shown in Fig. 1-(A). The shape is represented as a parametric function $f : [0, 1]^k \to \mathbb{R}^3$. The number of parameters k may differ depending on the shape. A time parameter can be added for time dependant shapes such as a moving line. The second aspect is encompassed under the term of *sampling strategy*, i.e. the shape's spatial sampling and temporal evolution of the stimulus, illustrated in Fig. 1. We thereby define a set of N sampled points $P^s = \{q_i, 0 \leq i < N\}$ $s.t.$ $\forall i, \exists t_{1,i}, ..., t_{k,i}, q_i = f(t_{1,i}, ..., t_{k,i})$ as a point on the shape. The temporal evolution in turn encompasses the order and timing with which the sampled points on the shape are displayed, as well as the evolution of the focal point's amplitude with respect to time. We can formally define the sampling order as a finite series $P^{s,o} = ((P_j, T_j))_j$ where $P_j \in P^s$ for all j and T_j is the dwell time on point P_j. By adding the amplitude overlay and sampling rate information, we obtain a configuration for each time step $C_k = (P_{j_k}, I_k)$ where $0 \leq j_k < N$ and $I_k = \frac{p(P_j)}{p_{peak}(P_j)}$ represents the pressure to peak pressure ratio for the device at the position P_j, with $p : \mathbb{R}^3 \to \mathbb{R}^+$.

A sampling strategy can thus be represented as the transformation $f \mapsto (C_k)_k$, given a sampling rate and peak pressure. This means that a same sampling strategy can be applied to different shapes. A sampling strategy transforms a possibly continuous shape (Fig. 1-(A)) into a discrete and finite representation (Fig. 1-(B)). Since the same set of spatially sampled points P^s can be displayed in different orders and with different timings, the sampling strategy also provides rules to define both these aspects (Fig. 1-(C1, C2)). This step leads to the creation of $P^{s,o}$. In addition to this, the sampling strategy must provide information about the peak pressure of the focal point at each time step (Fig. 1-(AM)). For instance, a simple STM approach would generate the maximum possible pressure at each time step while an AM approach generates a pressure varying as a (usually sinusoidal) function of time. This data, combined with the previous one finally leads to the computation of the different configurations C_k at each time step (Fig. 1-(D1, D2)).

4 THE DOLPHIN FRAMEWORK

We developed an open-source framework[5] to help users create haptic stimuli by customizing both shape and sampling strategy, and to use them for perception evaluations. A stimulus design tool provides a visual representation of the stimulus being created along with physical information about generated pressure distributions. Users can use DOLPHIN to directly test the stimulus and export it for later reuse in the framework's design tool or in a third-party application using DOLPHIN's reader for the ultrasound interface. Fig. 2 shows DOLPHIN's components and possible use cases.

4.1 Design Tool

The design tool architecture can be split into four parts. The "geometry" component handles the representation of mathematical shapes. The "sampling strategy" component implements different user-customized ways of displaying a shape. An "import and export" component handles saving and loading of shapes and strategies for reuse within the framework or by external applications executing the framework's Reader. Finally, the "user interface" links everything together. The tool also uses the Emitter library presented in

[5] URL: https://gitlab.com/h-reality/dolphin

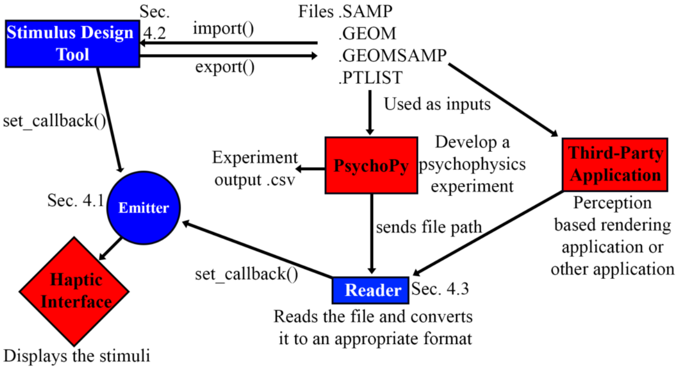


Figure 2: DOLPHIN's architecture. Rectangles represent programs, the diamond represents a device and the circle represents a class. Blue elements are part of the framework while red ones are not. Numbers next to the framework elements indicate the section presenting the element. Arrows and their corresponding text show how the different parts are linked. Export from the design tool is possible in four formats corresponding to the shape (.GEOM), the sampling strategy (.SAMP), or both (.GEOMSAMP, .PTLIST) with the generic .PTLIST format intended for use in external tools. Stimulus files can be used in experiments or third-party applications thanks to a reader program which sends data to the emitter for rendering.

Section 4.2 to display shapes with according sampling strategies on a hardware interface. The stimulus design tool is developed in C++ and uses the Qt framework[6].

4.1.1 Geometry. This component handles the mathematical representation of shapes to be displayed to the user. Every shape class inherits from the `Geometry` virtual class. The tool currently handles circles, ellipses, squares, rectangles, triangles, segments, point clouds and arcs. Users can create new shapes by creating sub-classes of `Geometry`, adding support for their configuration in the user interface and expanding the import and export operations to deal with the new shape.

Each shape can be customized using parameters specific to the shape, such as e.g. the radius and center position for a circle.

These shapes allow the on-screen display of the stimulus to the user and the computation of its spatial sampling as per the selected sampling strategy. To do so, each shape implements an `evaluate_at` method defining the parametric representation f of the shape. While the 2D plane in which the shapes are defined could theoretically be tilted, the software currently only supports a horizontal plane parallel to the device because empirical tests show a degradation of the tactile sensations for tilt angles greater than 25 degrees. More work is required to determine the causes of this issue and how to overcome it before integrating the tilt option in the tool. The tool currently only displays shapes defined as continuous curves or point clouds, and will be extended to surface and volume rendering in the future.

4.1.2 Sampling Strategies. The `SamplingStrategy` virtual class presents the main characteristics needed for a strategy. Combined with a Geometry it is used to create a list of 2D or 3D points corresponding to the sampled points, along with a list representing the intensity at each point. The following four strategies are currently available.

`StepSampling` corresponds to a uniform division of the shape into n points with a step parameter k. It leads to the sequential display of point 1, then point $k + 1$, $2k + 1$ and so on, modulo n. For instance with $k = 2$ and an odd number of points n, we first display the point with even indices and then display the ones with odd indices.

`CustomSampling` enables the user to create a non-uniform sampling. It is configured using a series of `(p, k)` instructions, meaning that the next k points will be at a distance of p times the distance corresponding to a uniform sampling. The value of p can be negative meaning that the focal point will travel in the opposite direction. This strategy can also be merged with `StepSampling` by setting up the `skip_step` parameter.

`RandomSampling` corresponds to a random sampling of n points. It is mainly intended to be used as a reference to compare with the performances of other strategies.

`CustomTimeStepSampling` is similar to the `CustomSampling` except that the user control focuses on temporal sampling. It is also parameterized by a series of `(p, k)` instructions, meaning that the focal point will spend p time steps on each of the next k points, leading to a dwell time of $T = p.s$ with s being the sampling rate. While dwelling on one point, the amplitude is modulated following a sine curve with configurable frequency. This strategy can be used to make the user feel a tactile pointer moving along the shape [Hajas et al. 2020].

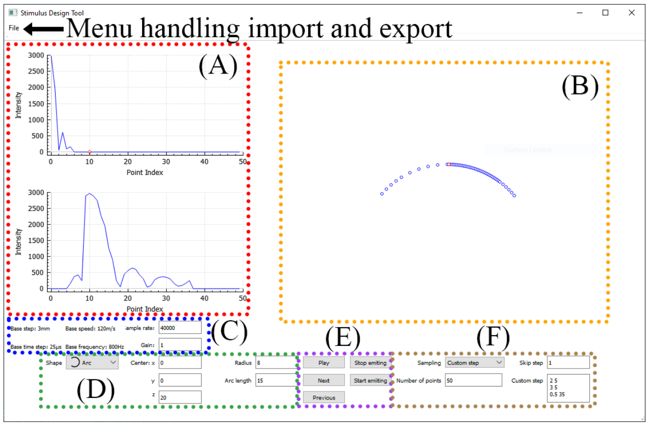
Each of the spatial sampling strategies can be complemented with additional AM information. Currently two methods are available. The first one sets the intensity of each point to 100%, corresponding to a classical STM approach, while the other one follows a sine curve with configurable frequency, which can be useful for displaying dynamic stimuli. The tool can also be used to make pure AM stimuli by setting a long dwell time on each point along with an AM overlay.

4.1.3 Import and Export. The previous two parts presented how stimuli are internally represented. Users may want to use such stimuli for other purposes unrelated to the design tool, such as experiments, thus requiring a way to export stimuli to a convenient file format. It is also important to provide the possibility of importing stimuli to enable design over several sessions or the reuse of pre-made shapes or strategies.

The design tool can independently create geometry or sampling strategy files. These are custom comma-separated variable (CSV) files, starting with the name of the geometry or strategy, followed by the different parameters required to recreate the object. A second line serves as a comment letting users know what each element represents, ensuring external compatibility of the file format.

Users can also store a complete stimulus, corresponding to a combination of shape and strategy. In this case, two files are created. The first one is a concatenation of the shape and strategy files described above, and is intended for reuse in the design tool. The second one is a CSV-like file starting with the number of time steps

[6] https://www.qt.io/



Base step: 3mm
Base speed: 120m/s
ample rate: 40000
Base time step: 25µs
Base frequency: 800Hz
Gain: 1
Shape: Arc
Center: x 0
y 0
z 20
Radius 8
Arc length 15
Sampling: Custom step
Skip step 1
Number of points 50
Custom step 2 5
3 5
0.5 35

Figure 3: Stimulus Designer user interface. (A) Graphs of the peak pressure as a function of position indices along the shape at a given time step. The top graph shows the pressure distribution with x=0 corresponding to the current point of emission, the bottom graph shows the same pressure distribution but with x=0 fixed at a reference point on the shape (red dot in the top graph) (B) Visual representation of the spatial sampling. (C) Basic information concerning the interface used (sampling rate, peak pressure output). (D) Shape dimensions and position in space (values in cm). (E) Animation and emission. Animation triggers a slowed down visual display of the sampling strategy, moving the red dot in (B) along the path corresponding to the strategy and updating the pressure graphs in (A). (F) Sampling strategy parameters.

along with the hardware sample rate and the maximum intensity of the device used to create the file, followed by a list of `x y z i` corresponding to the position and intensity of the point at each time step. The position is given in the system coordinate associated with the direct frame of reference centered on the device with the z axis perpendicular to the array plane, pointing outward. This format is voluntarily simple, allowing users to easily create external readers to translate the file and send information to an emitter.

In terms of performance, the export time for a .PTLIST file is proportional to the number of time steps, saving about 60000 time steps per second, running on a single thread on an Intel Core i7-10750H CPU. Importing in the design tool is almost instant as the creation of geometry and sampling strategy objects do not require heavy computations. Since these operations are done in the design tool, the time does not impact any rendering activity.

4.1.4 User Interface. The user interface (see Fig. 3) links everything together. On the right is a preview of the spatially sampled shape. On the left side are two graphs corresponding to the pressure distributions as a function of spatial sampling point indices at each timestep. These graphs are obtained using Ultraleap's HandyBeam simulation tool[7]. The bottom part allows the user to set stimulus parameters, with the shape parameters on the left and sampling parameters on the right. Preview animation and emission options are located in the middle. The user interface also provides basic information about modulation frequencies, spatial and temporal steps and focal point speed. The top menu provides access to import and export operations.

Given that the software is open-source, users can freely use individual components of the stimulus design tool in third-party applications or scripts.

4.2 Emitter

As we aim to make the software usable with different UMH interfaces, an abstraction of the actual device is required. Currently only Ultraleap's STRATOS Explore interface is supported as it is the device used during initial development of DOLPHIN. Having a plateform-agnostic representation of an emitter is the next logical step toward more universality in the stimulus use and representation. The representation is quite simple as each emitter just needs to implement a `start` and `stop` option along with an operation to set a callback and getters for some device characteristics such as the hardware sample rate and the maximum achievable pressure. Thus nothing in the framework is specific to a particular device. We chose to go with a callback approach in order to better separate the different operations. The design tool runs parallel to the callback which sends the appropriate data to the device displaying the focal points. Any user may add support for a new interface by creating a sub class of `Emitter` and implementing its methods.

If the emitter's sample rate is different from the one used to create the stimuli, an upsampling or downsampling strategy needs to be applied to adapt the stimuli to the new device while conserving the initial temporal display and modulation characteristics. Resampling presents some limitations that the user must be aware of, thus each saved file contains enough information to recreate an accurate version of the initial stimulus. Users can also include the *ImportExport* class in their program along with the file representation initially made for the design tool to avoid resampling the intensity. This recomputes the intensities for the correct time steps.

4.3 Interfacing for Perception Evaluation

DOLPHIN provides an interface for perception evaluation tools such as PsychoPy [Peirce et al. 2019] thanks to the stimulus file format. PsychoPy is a tool for conducting studies on perception which lets the experiment designer trigger audio, visual and haptic stimulus display hardware, deal with precise timing, ask the user custom questions and record responses and multiple objective metrics. While DOLPHIN is not a multimodal tool, it can be used in the study of multimodal perception as Psychopy can trigger and synchronize different types of stimuli. The tool automatically generates data files logging responses, information regarding stimuli and experiment progress for subsequent analyses.

A user designing an experiment begins by designing the individual stimuli required using DOLPHIN's stimulus design tool. PsychoPy is then provided with the paths to the stimuli, allowing it to trigger the appropriate stimulus during an experiment. To do so, PsychoPy executes the reader component of DOLPHIN with the appropriate file path. The reader in turn uses the emitter library and optionally libraries from the design tool to recreate the stimulus and send it to the haptic interface.

[7] https://github.com/ultraleap/HandyBeam

In terms of performances, loading a .PTLIST file with the reader program took about 11 ms per 1000 time steps on an Intel Core i7-10750H CPU. While this does not affect stimulus display quality or interactivity of the program, it can be an issue for stimuli with large numbers of time steps required to be played in sequence without pauses between them. A workaround for this issue is to load all stimuli to be displayed in sequence prior to the experiment. Whenever possible, the user should use the *ImportExport* class to import a .GEOMSAMP file instead of parsing the .PTLIST file, as it only takes about 0.3 ms to recreate the sampling objects and generate the same 1000 time steps on the same hardware.

5 PROOF OF CONCEPT

In order to validate DOLPHIN's usefulness, we used it to design and conduct a user study on the effect of sampling strategy parameters on curvature discrimination for dynamic tactile pointers [Hajas et al. 2020]. 19 volunteers (2 f., 16 m., 1 non-binary, mean age ± std = 23 ± 3.4), all right-handed) participated in the experiment. Stimuli were generated with DOLPHIN's design tool, then fed into the experiment designed with PsychoPy [Peirce et al. 2019], which in turn used the framework's reader component. All stimuli files and experiment scripts are available in DOLPHIN's repository[8] to allow replication of this experiment.

5.1 Materials and Methods

5.1.1 Stimuli. The stimuli were dynamic tactile pointers [Hajas et al. 2020] displaying arcs of four possible curvature radii across the width of the subject's hand for a fixed duration of two seconds. The stimuli were designed using a `CustomTimeStepSampling` strategy. The four different ratios of curvature radii to hand width were equal to $\frac{1}{\pi}$ (semi-circle), $\frac{3}{2\pi}$, $\frac{5}{2\pi}$, ∞ (straight line). For the computations, the radius to hand width ratio for the straight line was set to 1.99 which is the smallest ratio ensuring the height difference at the edge of the hand remains smaller than the radius of a focal point. We evaluated five strategies with $n = 10, 50, 200, 400, and 800$ points respectively, resulting in dwell times on each point of $t = \frac{2}{n}$ seconds. This corresponds to a single rule $\frac{80000}{n}$ n as the Ultraleap STRATOS Explore's hardware sample rate is 40 kHz. The static amplitude modulation frequency was set to 200 Hz as in [Hajas et al. 2020]. Each stimulus was displayed 10 cm above the array and the tactile pointer always moved from the thumb side towards the pinkie. DOLPHIN allowed us to generate the required stimuli with a simple script, and a similar result could be achieved without any programming by manually exporting from the design tool.All the generated .PTLIST files were then sent to the reader programm of DOLPHIN by PsychoPy as needed.

5.1.2 Protocol. Fig. 4 shows the experimental setup. Users positioned their hand horizontally over the box containing the array, parallel to the device plane, with the palm centered on the stimulus presentation location. Two test stimuli were first presented, to allow the user to get used to the feel of the tactile stimuli and enable them to correctly center their hand position. The users wore noise-cancelling headphones playing pink noise to hide the device's operating noise.

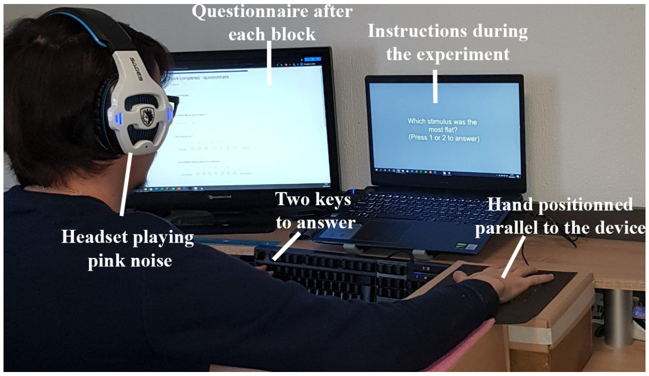

Questionnaire after each block
Instructions during the experiment
Which stimulus was the most flat? (Press 1 or 2 to answer)
Two keys to answer
Hand positionned parallel to the device
Headset playing pink noise

Figure 4: Experiment set-up: Subjects sit facing the computer with their dominant hand centered 10cm above the array, using their other hand to answer the experimental questions. A headset plays pink noise to cover device noise.

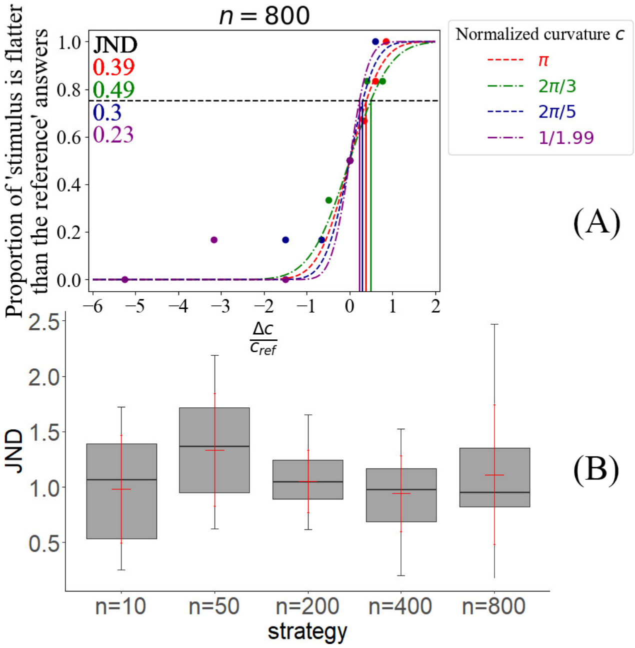


Figure 5: Experiment results. (A) Cumulative Gaussian curves fit to the results for one user for the $n = 800$ points strategy, with one curve fit per reference stimulus. The proportions of "stimulus flatter than reference" answers from the user are plot against the Weber fraction in curvature. For a stimulus with radius r_i, the curvature c_i is defined as $\frac{1}{r_i}$. JND estimates for this user and strategy are reported on the left. (B) Box plot of the mean JNDs obtained for each strategy, using the average of the four JNDs corresponding to the different reference stimuli for each user and strategy. Red lines represent the mean and one standard deviation.

The experiment was divided into five blocks corresponding to the five studied sampling strategies. Each of the five blocks followed

[8] https://gitlab.com/h-reality/dolphin

a two-alternatives forced choice protocol, where pairs of stimuli with different curvatures were presented with a 1.5 s break between them, after which users had to indicate which felt flattest (i.e. closest to a straight line). Users responded with their non-dominant hand, using two keys mapped to the two stimuli. Stimulus pairs were repeated three times per block, yielding thirty-six trials per block. The order of the pairs within blocks as well as the order of blocks were randomized. Subjects filled out a short questionnaire about the difficulty of the task and the strategies they used when responding after each block.

5.2 Results

For each user and each strategy, we plotted the proportion of 'stimulus is flatter than the reference' answers against the relative difference in curvature for each of the four references, as shown in Fig. 5-A. For a reference stimulus with radius r_1 and test stimulus with radius r_2, the x value corresponding to the pair was $\frac{c_2-c_1}{c_1}$, where $c_i = \frac{1}{r_i}$ corresponds to the curvature of an arc with radius r_i. We added the hypothetical point at (0, 0.5) corresponding to a fully random answer when both stimuli are identical. We then fitted the set of observations with a cumulative gaussian centered on 0 and used the curves to calculate the 75% JND estimate in curvature for this user, strategy and reference. The obtained JND is thus expressed as the Weber fraction in curvature. The data for 4 of the 19 participants was omitted from the analyses because of convergence issues preventing calculation of the JNDs. Among the 300 remaining JND estimates (15 participants × 5 strategies × 4 curvatures), 35 could not be computed.

Outlier JND values (values greater than $Q_{75\%} + 1.5IQR$ where $Q_{75\%}$ is the 75% quartile and IQR is the interquartile) were removed, yielding mean subject JND distributions for each stategy (see Fig. 5-B). These distributions were assumed to be normal (Shapiro-Wilk test did not reject H0 with $p > 0.05$ for all five distributions). We therefore performed an ANOVA which revealed no significant difference between strategies in terms of mean resulting JND in curvature (F= 1.676, p = 0.165). We conclude that when designing dynamic tactile pointers, the number of points used for the sampling strategy does not have a significant impact on users' ability to discriminate arc curvatures. The mean JNDs obtained were 0.98, 0.98, 1.33, 1.05 and 1.11 for the strategies with $n = 10, 50, 200, 400$ *and* 800 points respectively. These results mean that designers have more freedom when designing such stimuli as the strategy will likely not adversely impact stimulus discrimination.

6 INTEROPERABILITY

While our work is built around the Ultraleap STRATOS Explore interface because of its commercial availability (see Sec. 4.2), DOLPHIN is designed to be platform-agnostic. We aim to make it compatible with other arrays (e.g. Ultraino [Marzo et al. 2017], or the AUTD [Iwamoto et al. 2008]). Two blocks have to be carefully designed in order to create a platform-agnostic framework. While the geometries, user interface and sampling strategies are not inherently dependant on the device, the emitter object is. The export format also has to contain enough data so that any stimulus can be recreated accurately enough using any device. There is no strict assurance that the stimulus will feel exactly the same on different platforms since differences in computation of the acoustic field may introduce perceptual discrepancies, but DOLPHIN aims to provide the highest reproduction fidelity possible given this limitation using the information from the exported files and possible resampling. Any device owner can create a sub-instance of the `Emitter` class to make their device compatible with DOLPHIN. The user then implements the callback function which runs parallel to the design tool, sending appropriate data to the device to display the focal point. All the required information can be recovered from the shape and sampling strategy, regardless of whether the phases and amplitude for each transducer are computed by the device upon receiving the data or beforehand, within the callback function, and independently of the computation model used for this. This makes the emitter model essentially platform-agnostic. The exported file formats are made platform-agnostic by adding the necessary information to either recreate the shape or strategy object, or to resample the stimulus accordingly while warning the user about potential differences in the stimuli if the hardware characteristics do not match. The coordinate system used by each device may also differ from the one used by the framework. The emitter class thus implements a coordinate change. An inherent limitation to interoperability comes from the fact that different devices can have different sized workspaces, meaning that devices with a smaller workspace may not be capable of displaying stimuli designed on devices with a larger workspace.

7 CONCLUSION, LIMITATIONS AND FUTURE WORK

This work presented a new open-source framework, DOLPHIN, enabling easy mid-air ultrasound tactile stimulus design for rendering and experimental evaluation. It was shown to be functional by conducting a psychophysics experiment on curvature discrimination. We hope this framework will ease research on mid-air haptic stimuli and improve study replicability.

Short term future work will focus on minor features for the design tool such as support for new shapes, new sampling strategies, and other hardware platforms, along with a detailed evaluation of cross-platforms compatibility. We will also aim to include support for displaying multiple focal points. We intend to interface the tool with simulations of the acoustic fields and spatio-temporal behavior of skin indentation. This should provide designers with more information on how to achieve specific effects and help explain perception results with a physical model.

We then want to extend the shape workspace and allow active rendering, which is currently not supported. Though we will be adding more features, the core workflow and framework design will remain identical and we will ensure all current features remain fully functional.

Finally, we plan to continue to conduct psychophysics research using this tool to study the impact of other sampling parameters on subjects' perception of stimulus properties.

ACKNOWLEDGMENTS

This project has received funding from the European Union's Horizon 2020 programme under grant agreement No 801413; project "H-Reality".

REFERENCES

Damien Ablart, William Frier, Hannah Limerick, Orestis Georgiou, and Marianna Obrist. 2019. Using Ultrasonic Mid-air Haptic Patterns in Multi-Modal User Experiences. In *2019 IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*. IEEE, 1–6.

Jason Alexander, Mark T Marshall, and Sriram Subramanian. 2011. Adding haptic feedback to mobile tv. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*. 1975–1980.

Tom Carter, Sue Seah, Ben Long, Bruce Drinkwater, and Sriram Subramanian. 2013. UltraHaptics: Multi-point mid-air haptic feedback for touch surfaces. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 505–514. https://doi.org/10.1145/2501988.2502018

SUN Chongyang, NAI Weizhi, and SUN Xiaoying. 2019. Tactile sensitivity in ultrasonic haptics: Do different parts of hand and different rendering methods have an impact on perceptual threshold? *Virtual Reality & Intelligent Hardware* 1, 3 (2019), 265–275.

Brygida Dzidek, William Frier, Adam Harwood, and Richard Hayden. 2018. Design and evaluation of mid-air haptic interactions in an augmented reality environment. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 489–499.

William Frier, Damien Ablart, Jamie Chilles, Benjamin Long, Marcello Giordano, Marianna Obrist, and Sriram Subramanian. 2018. Using spatiotemporal modulation to draw tactile patterns in mid-air. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 270–281.

William Frier, Dario Pittera, Damien Ablart, Marianna Obrist, and Sriram Subramanian. 2019. Sampling Strategy for Ultrasonic Mid-Air Haptics. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI 2019)*. ACM, 1–11. https://doi.org/10.1145/3290605.3300351

Orestis Georgiou, Hannah Limerick, Loïc Corenthy, Mark Perry, Mykola Maksymenko, Sam Frish, Jörg Müller, Myroslav Bachynskyi, and Jin Ryong Kim. 2019. Mid-Air Haptic Interfaces for Interactive Digital Signage and Kiosks. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–9.

Hyunjae Gil, Hyungki Son, Jin Ryong Kim, and Ian Oakley. 2018. Whiskers: Exploring the use of ultrasonic haptic cues on the face. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

Daniel Hajas, Dario Pittera, Antony Nasce, Orestis Georgiou, and Marianna Obrist. 2020. Mid-Air Haptic Rendering of 2D Geometric Shapes with a Dynamic Tactile Pointer. *IEEE Transactions on Haptics* 13, 4 (Oct.-Dec. 2020), 806–817. https://doi.org/10.1109/TOH.2020.2966445

Keisuke Hasegawa and Hiroyuki Shinoda. 2013. A method for distribution control of aerial ultrasound radiation pressure for remote vibrotactile display. In *The SICE Annual Conference 2013*. IEEE, 223–228.

Keisuke Hasegawa and Hiroyuki Shinoda. 2018. Aerial vibrotactile display based on multiunit ultrasound phased array. *IEEE transactions on haptics* 11, 3 (2018), 367–377.

Takayuki Hoshi. 2011. Midair input-output interface consisting of depth camera and noncontact tactile display. In *SICE Annual Conference 2011*. IEEE, 608–609.

Takayuki Hoshi. 2012. Handwriting transmission system using noncontact tactile display. In *2012 IEEE Haptics Symposium (HAPTICS)*. IEEE, 399–401.

Takayuki Hoshi, Daisu Abe, and Hiroyuki Shinoda. 2009a. Adding tactile reaction to hologram. In *RO-MAN 2009-The 18th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 7–11.

Takayuki Hoshi, Takayuki Iwamoto, and Hiroyuki Shinoda. 2009b. Non-contact tactile sensation synthesized by ultrasound transducers. In *World Haptics 2009-Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. IEEE, 256–260.

Takayuki Hoshi, Masafumi Takahashi, Takayuki Iwamoto, and Hiroyuki Shinoda. 2010. Noncontact tactile display based on radiation pressure of airborne ultrasound. *IEEE Transactions on Haptics* 3, 3 (2010), 155–165.

Thomas Howard, Gerard Gallagher, Anatole Lécuyer, Claudio Pacchierotti, and Maud Marchal. 2019. Investigating the recognition of local shapes using mid-air ultrasound haptics. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, 503–508.

Seki Inoue, Yasutoshi Makino, and Hiroyuki Shinoda. 2015. Active touch perception produced by airborne ultrasonic haptic hologram. In *2015 IEEE World Haptics Conference (WHC)*. IEEE, 362–367.

Seki Inoue and Hiroyuki Shinoda. 2014. A pinchable aerial virtual sphere by acoustic ultrasound stationary wave. In *2014 IEEE Haptics Symposium (HAPTICS)*. IEEE, 89–92.

Mitsuru Ito, Yuji Kokumai, and Hiroyuki Shinoda. 2019. Midair click of dual-layer haptic button. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, 349–352.

Mitsuru Ito, Daisuke Wakuda, Seki Inoue, Yasutoshi Makino, and Hiroyuki Shinoda. 2016. High spatial resolution midair tactile display using 70 kHz ultrasound. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 57–67.

Takayuki Iwamoto, Mari Tatezono, and Hiroyuki Shinoda. 2008. Non-contact method for producing tactile sensation using airborne ultrasound. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 504–513.

Brian Kappus and Ben Long. 2018. Spatiotemporal Modulation for Mid-Air Haptic Feedback from an Ultrasonic Phased Array. *The Journal of the Acoustical Society of America* 143 (March 2018), 1836–1836. https://doi.org/10.1121/1.5036027

Georgios Korres, Tamas Aujeszky, and Mohamad Eid. 2017. Characterizing tactile rendering parameters for ultrasound based stimulation. In *2017 IEEE World Haptics Conference (WHC)*. IEEE, 293–298.

David R Large, Kyle Harrington, Gary Burnett, and Orestis Georgiou. 2019. Feel the noise: Mid-air ultrasound haptics as a novel human-vehicle interaction paradigm. *Applied ergonomics* 81 (2019), 102909.

Benjamin Long, Sue Ann Seah, Tom Carter, and Sriram Subramanian. 2014. Rendering volumetric haptic shapes in mid-air using ultrasound. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–10.

Maud Marchal, Gerard Gallagher, Anatole Lécuyer, and Claudio Pacchierotti. 2020. *Can Stiffness Sensations Be Rendered in Virtual Reality Using Mid-air Ultrasound Haptic Technologies?* Springer, 297–306. https://doi.org/10.1007/978-3-030-58147-3_33

Jonatan Martinez, Daniel Griffiths, Valerio Biscione, Orestis Georgiou, and Tom Carter. 2018. Touchless haptic feedback for supernatural VR experiences. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 629–630.

Jonatan Martinez, Adam Harwood, Hannah Limerick, Rory Clark, and Orestis Georgiou. 2019. Mid-Air Haptic Algorithms for Rendering 3D Shapes. In *2019 IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*. IEEE, 1–6.

Asier Marzo, Tom Corkett, and Bruce W Drinkwater. 2017. Ultraino: An open phased-array system for narrowband airborne ultrasound transmission. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 65, 1 (2017), 102–111.

Atsushi Matsubayashi, Yasutoshi Makino, and Hiroyuki Shinoda. 2019. Direct finger manipulation of 3d object image with ultrasound haptic feedback. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–11.

Saya Mizutani, Masahiro Fujiwara, Yasutoshi Makino, and Hiroyuki Shinoda. 2019. Thresholds of Haptic and Auditory Perception in Midair Facial Stimulation. In *2019 IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*. IEEE, 1–6.

Jonathan Peirce, Jeremy R Gray, Sol Simpson, Michael MacAskill, Richard Höchenberger, Hiroyuki Sogo, Erik Kastman, and Jonas Lindeløv Kristoffer. 2019. PsychoPy2: Experiments in behavior made easy. *Behavior Research Methods* 51, 1 (2019), 195–203. https://doi.org/10.3758/s13428-018-01193-y

Marlou N Perquin, Mason Taylor, Jarred Lorusso, and James Kolasinski. 2021. Directional biases in whole hand motion perception revealed by mid-air tactile stimulation. *Cortex* (2021).

Evan Pezent, Brandon Cambio, and Marcia K. O'Malley. 2020. Syntacts: Open-Source Software and Hardware for Audio-Controlled Haptics. *IEEE Transactions on Haptics* 14, 1 (2020), 225–233. https://doi.org/10.1109/TOH.2020.3002696

Dario Pittera, Damien Ablart, and Marianna Obrist. 2019. Creating an illusion of movement between the hands using mid-air touch. *IEEE transactions on haptics* 12, 4 (2019), 615–623.

Ismo Rakkolainen, Euan Freeman, Antti Sand, Roope Raisamo, and Stephen Brewster. 2020. A Survey of Mid-Air Ultrasound Haptics and Its Applications. *IEEE Transactions on Haptics* (2020).

Ahsan Raza, Waseem Hassan, Tatyana Ogay, Inwook Hwang, and Seokhee Jeon. 2019. Perceptually correct haptic rendering in mid-air using ultrasound phased array. *IEEE Transactions on Industrial Electronics* 67, 1 (2019), 736–745.

Gregory Reardon, Yitian Shao, Bharat Dandu, William Frier, Ben Long, Orestis Georgiou, and Yon Visell. 2019. Cutaneous wave propagation shapes tactile motion: Evidence from air-coupled ultrasound. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, 628–633.

Davide Rocchesso, Francesco Saverio Cannizzaro, Giovanni Capizzi, and Francesco Landolina. 2019. Accessing and selecting menu items by in-air touch. In *Proceedings of the 13th Biannual Conference of the Italian SIGCHI Chapter: Designing the next interaction*. 1–9.

Ted Romanus, Sam Frish, Mykola Maksymenko, William Frier, Loïc Corenthy, and Orestis Georgiou. 2019. Mid-air haptic bio-holograms in mixed reality. In *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 348–352.

Sonja Rümelin, Thomas Gabler, and Jesper Bellenbaum. 2017. Clicks are in the air: How to support the interaction with floating objects through ultrasonic feedback. In *Proceedings of the 9th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. 103–108.

Isa Rutten, William Frier, and David Geerts. 2020. Discriminating Between Intensities and Velocities of Mid-Air Haptic Patterns. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 78–86.

Isa Rutten, William Frier, Lawrence Van den Bogaert, and David Geerts. 2019. Invisible touch: How identifiable are mid-air haptic shapes?. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–6.

Ryoko Takahashi, Hasegawa, and Hiroyuki Shinoda. 2018. Lateral Modulation of Midair Ultrasound Focus for Intensified Vibrotactile Stimuli. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 276–288.

Ryoko Takahashi, Keisuke Hasegawa, and Hiroyuki Shinoda. 2019. Tactile stimulation by repetitive lateral movement of midair ultrasound focus. *IEEE transactions on* *haptics* 13, 2 (2019), 334–342.

Dong-Bach Vo and Stephen A Brewster. 2015. Touching the invisible: Localizing ultrasonic haptic cues. In *2015 IEEE World Haptics Conference (WHC)*. IEEE, 368–373.

Azuma Yoshimoto, Keisuke Hasegawa, Yasutoshi Makino, and Hiroyuki Shinoda. 2019. Midair haptic pursuit. *IEEE transactions on haptics* 12, 4 (2019), 652–657.