Blowhole: Blowing-Activated Tags for Interactive 3D-Printed Models

Carlos Tejada*   Osamu Fujimoto†   Zhiyuan Li‡   Daniel Ashbrook§

Rochester Institute of Technology

Figure 1: Example Blowhole applications. The elephant (a) has six holes (one on the head, one on the back, one on each foot) which, when blown into, activate text-to-speech with different elephant facts. Each bar in the bar chart (b) triggers a readout of the quantity, and can be moved to represent different data. The sperm whale, dolphin, and orca (c) each produce a different tone to trigger videos of those animals. The cell model's blowholes interact with a quiz program to test knowledge of its components (d).

ABSTRACT

Interactive 3D models have the potential to enhance accessibility and education, but can be complex and time-consuming to produce. We present *Blowhole*, a technique for embedding blowing-activated tags into 3D-printed models to add interactivity. Requiring no special printing techniques, components, or assembly and working on consumer-level 3D printers, Blowhole adds acoustically resonant cavities to the interior of a model with unobtrusive openings at the surface of the object. A gentle blow into a hole produces a unique sound that identifies the hole, allowing a computer to provide associated content. We describe the theory behind Blowhole, characterize the performance of different cavity parameters, and describe our implementation, including easy-to-use software to automatically embed blowholes into preexisting models. We illustrate Blowhole's potential with multiple working examples.

Index Terms: Human-centered computing—Visualization—Visualization techniques—Treemaps; Human-centered computing—Visualization—Visualization design and evaluation methods

1 INTRODUCTION

Adding interactivity to 3D-printed objects in an end-user-friendly way is a difficult undertaking. While doing so can enable a wealth of educational and accessibility applications (e.g., Figure 1), adding interactivity can require extra components [7, 17], special printing techniques [2, 14, 19], or large features that can disrupt the object's surface [4,13,18,22] or make it larger [11]. While such solutions can offer rich interaction possibilities such as buttons, touch sensitivity, and grasp sensing, many useful applications can be enabled by simply "tagging" 3D models, allowing a system to know what part of a model a user is interested in and respond with information.

---

*cet1318@rit.edu
†oaf7862@rit.edu
‡zl7904@rit.edu
§daniel.ashbrook@rit.edu

In this paper, we present *Blowhole*, a system for unobtrusively tagging 3D models or parts of models. Inspired by previous work in acoustic sensing, our system creates embedded, resonant cavities that a user gently blows into (Figure 1). Different-sized cavities create sounds at different pitches that can be recognized by a computer. The main focus of Blowhole is to enable 3D-printing novices to create interactive objects which can be printed and immediately used with no during-print intervention, post-processing, or training necessary. The contributions of our work are as follows:

- a technique for embedding resonant cavities—printable on commodity 3D printers—into existing 3D models, which produce distinct pitches when blown into;

- characterization of the range of frequencies that can be generated and recognized by our system;

- a design environment which enables non-expert users to indicate areas which they wish annotated, and automatically generates and embeds appropriate cavities;

- and a user-independent system which recognizes the sounds of the user blowing into the cavities and plays back the associated annotations.

2 RELATED WORK

Adding interactivity to 3D-printed (and other) objects has long been a research topic in HCI. The main sensing approaches to date include embedding electronics, non-acoustic external sensing, and both active and passive acoustic sensing.

2.1 Embedded Electronics

Embedding electronics in 3D-printed objects can enable directly sensing the user. For example, adding an internal camera [17] or accelerometers [7] can allow sensing of movable elements. Other researchers have used multi-material 3D printers to add capacitive touch pathways directly to flat-bottomed prints which are placed on a touchscreen to enable sensing [3, 19, 25]. Electronics have also been used to measure the deformation of flexible objects, using air pressure sensors [23], piezoresistive wires [2], or conductive fabric [14]. These methods offer precise sensing of user input at the

cost of special parts, materials, printers, and post-printing assembly. In contrast, Blowhole works with consumer-level 3D printers and works immediately with no assembly; the trade-off is less-rich input.

2.2 External Sensing

Some work has investigated adding touch detection to 3D-printed objects via external sensors. A six-degree-of freedom force sensor [8] achieved high touch resolution, but required recalibration upon object movement; computer vision approaches [15, 21] avoid this issue, but suffer from occlusion. Blowhole works in any position, does not require line-of-sight, and needs no recalibration when moved.

2.3 Acoustic Sensing

Many systems use sound as a method to add sensing capabilities to objects. Active sensing approaches such as Touch & Activate [12] or Acoustruments [10] use a speaker to create sound which is modified by user action on an object. These methods require less expertise and effort from users than embedded approaches, but rely on custom electronics or complex printing. Blowhole requires no components beyond the 3D-printed model itself and is printable in a single piece on a standard consumer-level 3D printer.

Passive acoustic sensing approaches require the user to generate sound themselves in conjunction with an object; many previous approaches, however, require large interaction surfaces. Acoustic Barcodes uses the sound made by a fingernail running over ridges to encode IDs [4], occupying at minimum 7×22 mm, while Lamello [18] and Tickers & Talker [22] use elements which make distinct sounds when struck or strummed, requiring 4×50 mm (Lamello) or 11.5×17 mm (Tickers & Talker) interaction elements. Blowhole hides the majority of its structure below the surface, with only a 5 mm diameter hole required. The holes lead to cavities which require *volume*, but our design software automates the selection of cavity size to fit within the available model space.

Most similar to Blowhole are approaches that use the sound of air to add interactivity. Acoustic Voxels embeds multiple cavities into an object to change the absorption of different frequencies [11], but requires multiple 2cm-cube structures to encode information, necessitating large objects. Squeezapulse [5] identifies which of several tubes in an object air is flowing through; however, the corrugated tubes must be manually fabricated via molding and inserted post-fabrication along with an embedded microphone. Printone uses mathematical simulation to allow users to design custom 3D-printable wind instruments [26], but is optimized for musical applications and requires extensive user input. Whoosh [16] adds small predefined cavities to a watch, allowing hands-free control via blowing at the watch face. Blowhole extends this previous work, adding multiple blow-activated tags to imbue arbitrary 3D-printed objects with interactivity. Blowhole achieves high accuracy with up to nine tags per object—more than most other passive acoustic approaches in the literature.

3 Blowhole

Blowhole is based on the property of acoustic resonance; a familiar example is the sound created when blowing across the mouth of a bottle. Blowhole embeds cavities into 3D models, with tubular openings to the surface. Varying the volumes of the cavities and lengths of the tubes produces varying frequencies in response to gentle blowing into the holes, with the object held 5–10 cm away from the mouth. Our system recognizes the characteristic sound of each hole, linking the blow sound to an action associated with the hole's location on the model. Our design tool allows a user to select the placement of holes on arbitrary 3D models and associate actions with each hole; the software then optimizes blowhole size and placement, providing a printer-ready file.

The cavities used in Blowhole must satisfy several criteria: they must support sufficient variation in parameters to produce a range of frequencies when blown into; they must be sufficiently small to embed into models small enough to hold and manipulate; they should present a consistent hole appearance to the user; and they should be printable at any orientation and without support material on a consumer-grade printer.

This last criterion poses the strongest limitation due to the limited ability of FDM[1] 3D printers to print with *overhangs*—angles greater than 45° from gravity—and *bridging*—printing material with nothing underneath it as a support. We experimented with a number of shapes. Simple tubes (as used, for example, in Whoosh [16]) printed well at any orientation but quickly became too large to embed in smaller objects while supporting a range of frequencies. We tested several variations on shorter tubes connecting to larger cavities, which preserve a standard opening size but allow the production of a greater range of frequencies. We experimented with cavities shaped like spheres, cylinders, cylinders with cones on tops, cubes, and cubes with pyramids. The shape resulting in the best combination of clear sound and multi-orientation printability was a sphere with a tube connecting to the surface of the model. In the next section, we detail the theory behind the resonant properties of this structure.

3.1 Cavity Resonator Theory

Blowhole operates on the principle of acoustic resonance, where particular frequencies are amplified or attenuated due to the physical properties of a cavity. Blowhole uses spherical cavities inside a 3D-printed model with straight pipes opening onto the surface; the resonant frequency of a cavity depends on the area and length of the opening and the volume of the cavity, and is classically modeled using the Helmholtz resonance equation [6]:

$$f = \frac{cd_t}{\pi} \sqrt{\frac{3}{8(L_t + .75d_t)d_s^3}} \qquad (1)$$

with c the speed of sound, d_s the diameter of the spherical cavity and d_t and L_t the diameter and length, respectively, of the tube connecting the cavity to the surface of the object. Figure 2 illustrates these parameters of Blowhole cavities.
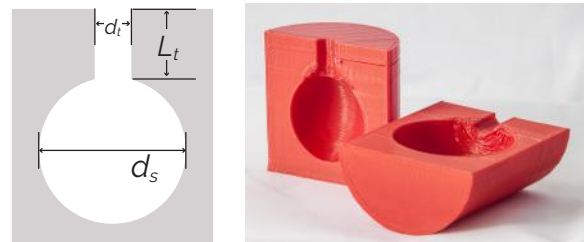

Figure 2: Left: an ideal spherical Helmholtz resonator, with tube diameter d_t, tube length L_t, and sphere diameter d_s. Right: cross-sections of two Blowhole test objects, showing the resonator structures.

3.2 Blowhole Characterization

In order for Blowhole to be of the most practical use, we want to understand how many different cavities we can fit inside a given object. As can be seen from Equation 1, we can vary three parameters—d_t, L_t or d_s—to change the resonant frequency of a Blowhole cavity. As the tube is the only user-facing element of Blowhole, its appearance should be consistent, with the size of the opening large enough to easily blow into, but not so large as to interfere with the features

---

[1]Fused Deposition Modeling—the most-common consumer printer technology

Figure 3: A subset of our test cylinders with varying cavity volumes and tube lengths.

of the printed model. After some initial experimentation, we set d_t to 5 mm, leaving L_t and d_s as the available parameters to manipulate. Multiple combinations of these can produce the same predicted frequency; for example, $L_t = 2.5$ mm and $d_s = 35.3$ mm produce a prediction of 1000 Hz, as do $L_t = 5$ mm and $d_s = 28$ mm.

To understand the practical limits on the frequencies we could detect and differentiate between, we produced a large number of test objects (Figure 3) using consumer FDM printers (Qidi Technology X-One, LulzBot Taz 4, and LulzBot Taz Mini). Wanting to understand the practical limits on the frequencies we could detect and differentiate between, we produced 48 objects with cavities and tubes of different sizes. Holding d_t at 5 mm, we manipulated d_s from 8–40 mm in steps of 4 mm and tested L_t at 2.5, 3.5, 5, 7.5, 8.5 and 10 mm. These configurations gives us a frequency space ranging from 500 Hz, ($d_t = 5$ mm, $L_t = 10$ mm and $d_s = 40$ mm) to 5900 Hz ($d_t = 5$ mm, $L_t = 2.5$ mm and $d_s = 8$ mm).

We asked ten people to blow into each cylinder between one and four times, recording the data via a laptop computer's built-in microphone at a 44,100 Hz sampling rate. We extracted the fundamental frequencies of each blow using Welch's method [27]. Comparing the fundamental frequencies to the values predicted by the Helmholtz equation (Equation 1), we find deviation, sometimes significant. However, the deviation is not constant, but presents as noise, with two main patterns: frequencies under 1000 Hz are much noisier than those higher; and longer tubes exhibit more noise than shorter. Figure 4 illustrates the spectrogram from one user for a series of blows into different cavity sizes.

This behavior may be explained by several factors. First, the Helmholtz equation is a theoretical model known to be inexact for varying properties of cavity geometry [1, 20], assumes a reflective, smooth surface, and as L_t and d_s approach 2–5% of the resonant wavelength, the model begins to break down [20]. As shown in Figure 2, our 3D-printed models are not smooth. As the cavity size increases, the top of the sphere approaches horizontal, which can cause drooping or stringing. These features of 3D prints may affect the resonance. To validate print features as a potential cause, we printed eight duplicate test objects on a resin-based printer (the FormLabs Form 2), resulting in hard, smooth objects. Blows into these objects resulted in frequencies much closer to the predicted values. However, our goal is to maintain broad accessibility of our technique, so while noting the potential for resin printers to produce better results, we continue to describe our results using FDM printers.

To validate the printability of our cavities, we tested multiple cavity sizes, from 5 mm to 60 mm in diameter. While all objects printed correctly, the smallest cylinders resulted in frequencies highly variable over the course of a single blow, and the 60 mm cavity failed to produce any strong harmonic at all.

We also tested the consistency of sound at different angular positions of the tube opening, from 0°(straight down) to 180°(straight up) in 22.5°increments with d_s as 16 mm and L_t as 5 mm. Although different orientations revealed different (minor) printing artifacts such as slight stringing and tube opening shape inconsistency, the results were consistent, with a mean deviation of under 240 Hz from the Helmholtz-predicted value.

We tested Blowhole with multiple printers: a LulzBot Taz 4, a LulzBot Taz Mini, two Qidi X-One v2 printers, and a Form Labs Form 2 resin-based SLA printer. All printed successfully; inspecting the spectrograms, we found little variance amongst the FDM prints, and that the SLA prints produce dominant frequencies on average 100 Hz closer to the Helmholtz-predicted frequency than the FDM prints and with less variation over the signal.

4 SYSTEM IMPLEMENTATION

As a system, Blowhole consists of three parts: the design software to modify existing 3D models to add blowholes; the physical printed-out models with resonant cavities and holes embedded; and the software that recognizes the sound of the user blowing into a cavity and performs an action. All pertinent code and designs can be found online [2].

4.1 Design Software

Our design software is built on top of Autodesk Meshmixer (Figure 5, left) using its Python API for scripting remote command execution. To add Blowhole tags to a model, a user simply imports an existing model and then clicks on the model to specify tag locations and desired actions. Currently supported actions include opening URLs, launching files such as images and movies, and reading text via text-to-speech. After the user indicates all of their desired blowhole positions, the software determines the best set of cavity sizes to embed in the model. The naïve approach would be to simply place the largest possible available cavity in a location when the user selects it. However, this approach quickly fails; for example, if the user wished to place a blowhole in each eye of the elephant in Figure 1a, the first click would fill the elephant's head with a cavity and the second click would fail to find an acceptable cavity size. The opposite approach—choosing the smallest available size first—suffers from similar issues.

Instead, our search algorithm allows the user to add all requested positions first, then attempts to optimize cavity placement by finding a set of L_t and d_s that will fit all requested blowhole locations without cavities colliding (we can optionally fix L_t to a single value). The algorithm is based on depth-first search with backtracking. We represent the solution space—mapping a set of available cavities to desired blowhole positions—as a tree, with the root representing the original model, internal nodes as intermediate steps towards solutions, and leaves as final solutions. Each step of the algorithm takes as input a node, a list of candidate locations, and a list of unused cavity sizes. It tests each of the available cavities in the next candidate location until it finds one that fits within the model at that location. It then removes the candidate location and the cavity size from their respective lists and passes the newly modified model as a node to run the algorithm again. If, for a given node, no cavity can be found that fits within the candidate location, the tree is pruned at that point and the algorithm returns to the parent node to try the next child. When no candidate locations remain, we can output the current node's model as the solution; if candidate locations do remain but there are no more cavity sizes, we inform the user that we cannot find a solution. The search process takes under a minute.

The final result is a set of location/cavity pairs, which we then use to construct the model for printing (Figure 5, right). Once the cavities are placed, the software writes out a configuration file

---

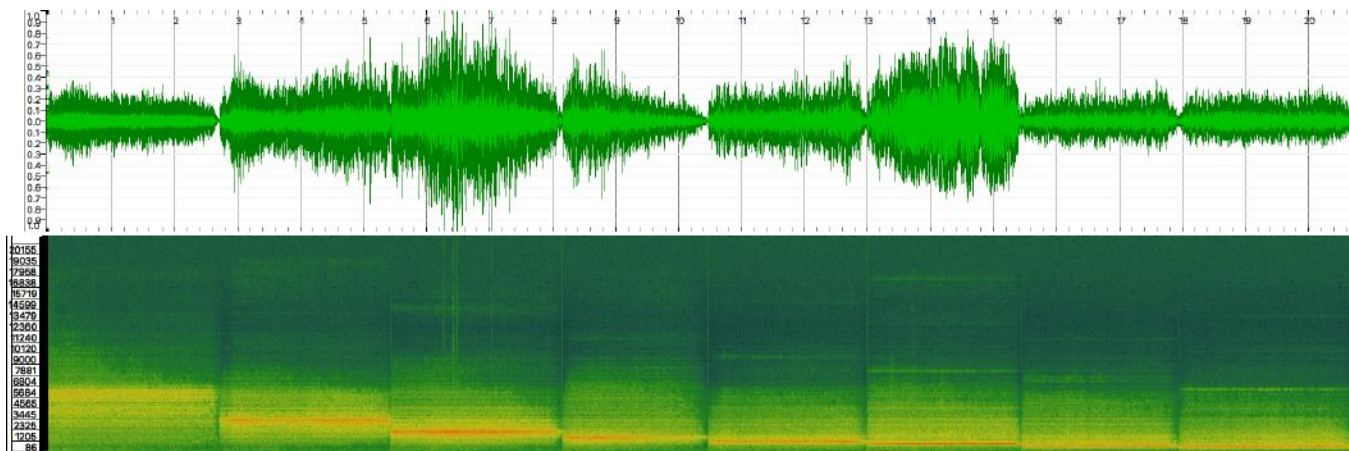[2] `https://github.com/fetlab/blowhole-gi18`

Figure 4: Waveform (top) and spectrogram (bottom) of blows into holes with a tube length L_t of 2.5 mm, with the cavity diameter d_s varying in steps of 2 mm from 4 mm on the left to 18 mm on the right.

linking the cavity parameters L_t and d_s to the specified action. The final model may be exported to a STL file for 3D printing on a commodity printer.

In informal experiments with Blowhole-enabled objects, we found that locating the holes by feel alone could be challenging; on complex models like the elephant (Figure 1a) the hole gets "lost" in the model's geometry. Because one use scenario for Blowhole objects is as an aid for people with visual impairments, we added a "ring" feature. This simply adds a short (3 print layers, or about 0.64 mm high) ring extending 1 mm around the hole (Figure 7). In informal tests, we found that the ring is easily distinguishable by touch alone, allowing the hole to be easily located without vision. In order to avoid changing the sound produced, when we add the ring we shorten the inner end of the tube by the height of the ring, thereby maintaining the same L_t.

4.2 Blowhole Objects

Our software places blowholes into existing models, therefore models that are 3D-printable will remain so with the addition of cavities and openings. Because the cavities are spherical, and most hobbyist-level 3D printers can print up to 45° of overhang, the models can be produced on most printers with no modification; importantly, no support material is necessary inside the cavities or tubes.

Once a Blowhole-enabled object has been printed, some minor cleanup may be required: with larger spherical cavities, the top of the sphere becomes nearly horizontal, and the printer may produce some "3D printer spaghetti" (a small amount is visible in Figure 2) that can slightly muffle the sound. A simple solution is to simply insert a drill bit of the appropriate size and twist it by hand to quickly remove the strands.

4.3 Blow Sound Recognition

The last component of our system recognizes the sounds produced by the user blowing into the blowholes, producing the resonant frequency characteristic to cavity/tube combinations (Equation 1), allowing us to link the sound to the particular location the user is interacting with. Our software is implemented in Python running on a laptop, but is simple enough to run on phones and smartwatches as well.

To identify the resonant frequency, we window the 44,100 Hz incoming audio signal in 0.1s non-overlapping segments. We compute the RMS value of each and look for .5s worth of contiguous windows that exceed an empirically determined threshold. We apply Welch's method to extract the power spectrum of the signal [27], and

use the strongest frequency as the resonance. We then take the set of cavity/tube (d_s/L_T) combinations available and match the resonant frequency to the Helmholtz-predicted frequencies to determine which hole the user is interacting with. Once a blow is classified, the system executes the action referenced in the configuration file produced by the design software.

Our main implementation is on a laptop computer, using its built-in microphone. We also tested with a LG-R Android smartwatch which transmits audio data to the same recognition pipeline. Our software runs in Python and uses the scikit-learn library for recognition.

4.4 Performance Testing

To validate our recognition procedure, we collected a total of 830 blow segments from ten participants as described in section 3.2, with L_t of 2.5, 5, and 10 mm, and d_s varying from 8–40 mm in 4 mm increments. We divided the data according to the tube length and evaluated our recognition procedure both overall and on a per-user basis. Figure 6 presents a cross-validated evaluation of our classification system, where we can see that the reliability decreases as the diameter of the sphere increases.

We found that as we include larger spheres and larger tubes, the recognition accuracy decreases. The best performance/versatility tradeoff occurs at L_t of 2.5 mm and six spheres from 8–28 mm, which yields an overall 98% accuracy. Adding a 32 mm sphere decreases accuracy to 90%, and further spheres continue to decrease accuracy.

5 Examples

To illustrate the potential of Blowhole, we present several possible applications. Each was built with our software and works with our recognition algorithm.

Cell Model. We adapted an existing model of an animal cell[3] to add Blowhole tags to the different parts of the cell (Figure 1d). When the tags are activated, the listening computer application launches the Wikipedia page for the associated cell component.

Globe. Similar to the example in Tickers and Talker [22], we tagged the continents on a 3d-printed globe[4] (Figure 5c). When a user blows into the associated hole, our software speaks the name of the labeled continent using text-to-speech.

---

[3] http://www.thingiverse.com/thing:689381
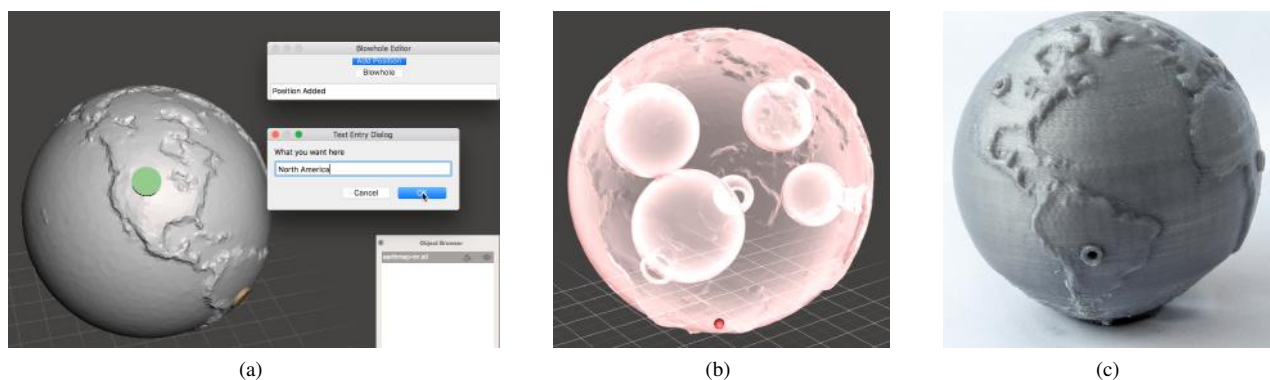[4] http://www.thingiverse.com/thing:17336

Figure 5: (a,b): Detail view of our Blowhole design software, based on Meshmixer. (a) shows the software with the user inserting blowholes: clicking a point on the model results in a placeholder (green dot) and a dialog box where the user can specify the action to be taken upon blowing. (b) shows the interior of the model, illustrating the different-sized cavities the software inserted. (c) shows the final 3D-printed object with blowholes embedded.
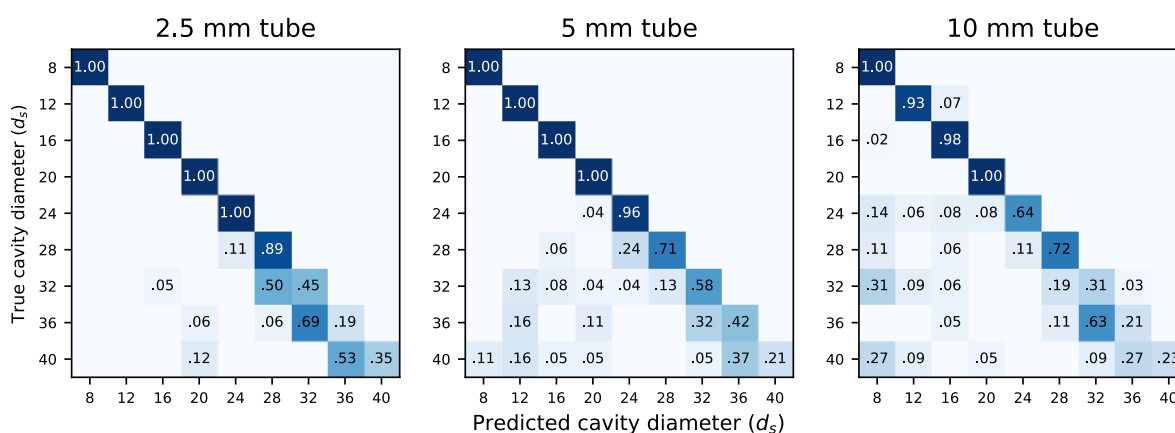

Figure 6: Confusion matrices across all test participants for the three tested tube lengths with sphere diameters from 8–40 mm.
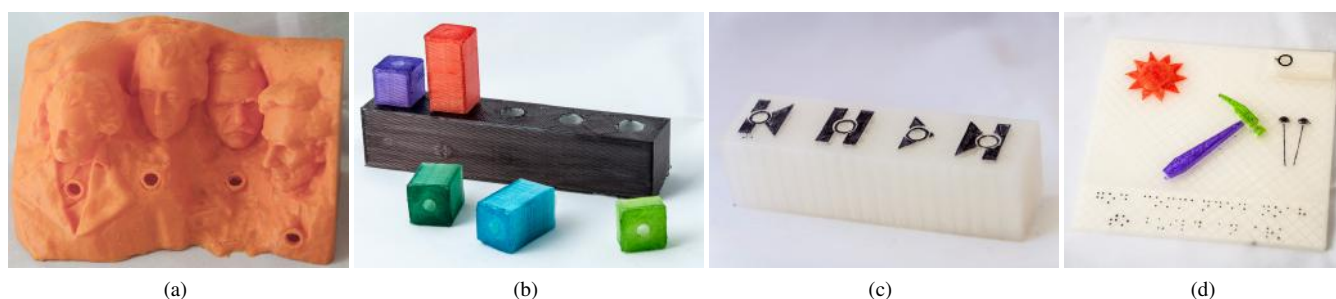

Figure 7: Example Blowhole-augmented objects: (a) a model of Mt. Rushmore with each president tagged; (b) the bar graph from Figure 1b in a different arrangement; (c) a box that controls a music player by blowing; and (d) a 3D-printed tactile picture book for blind or low-vision children with a blowhole (upper right) which triggers text-to-speech of the Braille text.

Interactive Animals. We printed three different cetaceans: a dolphin[5], a whale[6], and an orca[7], and adapted the position of the cavity to the location of the animal's blowhole (Figure 1c). When the user blows, the application plays a video about that animal.

Music Controller. A "music box" with raised controls (Figure 7c)

allows a user to control the flow of music by blowing. Each "button" has a different blowhole underneath it. Our segmentation algorithm described earlier is robust to background sound and in initial testing, its performance was not affected by the sound of the music playing.

Augmented Tactile Book. Previous research [9] has investigated 3D-printed tactile picture books for blind children. Some examples of these books have Braille text[8]. We created a set of custom rect-

---

[5] http://www.thingiverse.com/thing:1121803
[6] http://www.thingiverse.com/thing:232247
[7] http://www.thingiverse.com/thing:665571

[8] https://tactilepicturebooks.org/

angular resonators, thinner than our standard Blowhole spherical resonators, to add to each page of a 3D-printed book (Figure 7d). When the user blows into the hole, the computer reads aloud the text written in Braille on the page.

Reconfigurable Bar Chart. Figure 1b shows a Blowhole-enabled physical visualization [24]. Taking advantage of the Helmholtz property that varying L_t varies the frequency, the base of the bar chart contains cavities with identical d_s values. Each bar, being a different height, has a different L_t; when a bar is plugged into the chart, the tone produced is due to the size of the bar. This characteristic enables reconfiguring the bar chart, illustrating data in different orders while maintaining the labels on the individual bars. Placing the resonating cavities in the base rather than the bars allows the bars to maintain smaller cross-sections but still produce audible tones.

6 DISCUSSION AND FUTURE WORK

Our goal with Blowhole was to create a system to enable non-experts to design and 3D print interactive objects, with the particular aim of simplicity, avoiding interventions during printing, post-print processing, and complex training processes. Blowhole is usable for simple cases; with up to six cavities, the system achieves a high user-independent performance of 98%. As illustrated by our examples in Figures 1 and 7, six cavities are sufficient for many applications, and is a larger number of tags per object than have been demonstrated by other passive acoustic-based systems [4, 5, 10, 11, 16, 22].

While Blowhole is successful, there is room for future improvement. We are interested in further characterizing the behavior of the cavities with different print settings; for example, we have encountered some tentative evidence that the type and amount of infill the printer uses to fill the solid parts of the model may have an effect on the sound. Refinements to the cavity shapes may also have an effect, for example by shaping the top of the spheres to avoid greater than 45°overhang in order to prevent the "stringing" effect. Additionally, we are interested in exploring the user experience while using this tool. We believe that tools like Blowhole can be particularly useful for educational settings, specially for children, due to its playfulness, as well as to enable new ways low vision users can interact with their technology. We will carry out user studies in the future to assess the experience of these two populations while using Blowhole.

7 CONCLUSION

We presented Blowhole, a system for adding acoustic "tags" to 3D-printed models via embedded cavities which resonate at characteristic frequencies when a user blows into them. Our system enables high performance for up to nine different blowholes, provides simple point-and-click design software, and Blowhole-enabled models are ready to use immediately post-printing with no assembly or external components required. We detailed the theory behind Blowhole's operation, presented our characterization of its performance, and demonstrated high user-dependent and -independent recognition rates. We demonstrated Blowhole's potential through multiple examples, including educational models, 3D-printed book pages, and a reconfigurable physical visualization.

REFERENCES

[1] M. Alster. Improved calculation of resonant frequencies of Helmholtz resonators. *Journal of Sound and Vibration*, 24(1):63–85, Sept. 1972.

[2] M. Bächer, B. Hepp, F. Pece, P. G. Kry, B. Bickel, B. Thomaszewski, and O. Hilliges. DefSense: Computational Design of Customized Deformable Input Devices. In *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 3806–3816. ACM Press, New York, New York, USA, 2016.

[3] T. Götzelmann. *LucentMaps: 3D Printed Audiovisual Tactile Maps for Blind and Visually Impaired People*. 3D Printed Audiovisual Tactile Maps for Blind and Visually Impaired People. ACM, New York, New York, USA, Oct. 2016.

[4] C. Harrison, R. Xiao, and S. E. Hudson. Acoustic Barcodes: Passive, Durable and Inexpensive Notched Identification Tags. In *UIST '12: Proceedings of the 25th annual ACM symposium on User interface software and technology*, pp. 563–567. ACM Press, New York, New York, USA, 2012.

[5] L. He, G. Laput, E. Brockmeyer, and J. E. Froehlich. SqueezaPulse: Adding Interactive Input to Fabricated Objects Using Corrugated Tubes and Air Pulses. In *International Conference on Tangible, Embedded, and Embodied Interaction*, pp. 341–350. ACM, New York, New York, USA, Mar. 2017.

[6] H. L. F. Helmholtz. *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. Longmans, Green, and Co., London, United Kingdom, 2 ed., 1885.

[7] J. Hook, P. Wright, T. Nappey, S. Hodges, and P. Olivier. Making 3D printed objects interactive using wireless accelerometers. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, pp. 1435–1440. ACM, Apr. 2014.

[8] C. Hudin, S. Panëels, and S. Strachan. INTACT: Instant Interaction with 3D Printed Objects. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 2719–2725. ACM, New York, New York, USA, May 2016.

[9] J. Kim and T. Yeh. Toward 3D-Printed Movable Tactile Pictures for Children with Visual Impairments. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 2815–2824. ACM, New York, New York, USA, Apr. 2015.

[10] G. Laput, E. Brockmeyer, S. E. Hudson, and C. Harrison. Acoustruments: Passive, Acoustically-Driven, Interactive Controls for Handheld Devices. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 2161–2170. ACM, New York, New York, USA, Apr. 2015.

[11] D. Li, D. I. W. Levin, W. Matusik, and C. Zheng. Acoustic voxels: computational optimization of modular acoustic filters. *ACM Transactions on Graphics (TOG)*, 35(4):88–12, July 2016.

[12] M. Ono, B. Shizuki, and J. Tanaka. Touch & activate: adding interactivity to existing objects using active acoustic sensing. In *UIST '13 Proceedings of the 26th annual ACM symposium on User interface software and technology*, pp. 31–40. ACM, New York, New York, USA, Oct. 2013.

[13] J. Ou, G. Dublon, C.-Y. Cheng, F. Heibeck, K. Willis, and H. Ishii. Cillia: 3D Printed Micro-Pillar Structures for Surface Texture, Actuation and Sensing. In *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5753–5764. ACM, New York, New York, USA, May 2016.

[14] H. Peng, J. Mankoff, S. E. Hudson, and J. McCann. A Layered Fabric 3D Printer for Soft Interactive Objects. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 1789–1798. ACM Press, New York, New York, USA, 2015.

[15] A. Reichinger, A. Fuhrmann, S. Maierhofer, and W. Purgathofer. Gesture-Based Interactive Audio Guide on Tactile Reliefs. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 91–100. ACM, New York, New York, USA, Oct. 2016.

[16] G. Reyes, D. Zhang, S. Ghosh, P. Shah, J. Wu, A. Parnami, B. Bercik, T. Starner, G. D. Abowd, and W. K. Edwards. Whoosh: non-voice acoustics for low-cost, hands-free, and rapid input on smartwatches. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, pp. 120–127. ACM, New York, New York, USA, Sept. 2016.

[17] V. Savage, C. Chang, and B. Hartmann. Sauron: embedded single-camera sensing of printed physical user interfaces. In *UIST '13 Proceedings of the 26th annual ACM symposium on User interface software and technology*, pp. 447–456. New York, New York, USA, Oct. 2013.

[18] V. Savage, A. Head, B. Hartmann, D. B. Goldman, G. Mysore, and W. Li. Lamello: Passive Acoustic Sensing for Tangible Input Components. In *CHI '15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 1277–1280. ACM Press, New York, New York, USA, 2015.

[19] M. Schmitz, M. Khalilbeigi, M. Balwierz, R. Lissermann, M. Mühlhäuser, and J. Steimle. Capricate: A Fabrication Pipeline

to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *UIST '15: Proceedings of the 28th annual ACM symposium on User interface software and technology*, pp. 253–258. ACM Press, New York, New York, USA, Jan. 2015.

[20] A. Selamet, N. S. Dicky, and J. M. Novak. Theoretical, Computational and Experimental Investigation of Helmholtz Resonators With Fixed Volume: Lumped Versus Distributed Analysis. *Journal of Sound and Vibration*, 187(2):358–367, Oct. 1995.

[21] L. Shi, R. McLachlan, Y. Zhao, and S. Azenkot. Magic Touch: Interacting with 3D Printed Graphics. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 329–330. ACM, New York, New York, USA, Oct. 2016.

[22] L. Shi, I. Zelzer, C. Feng, and S. Azenkot. Tickers and Talker: An Accessible Labeling Toolkit for 3D Printed Models. In *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 4896–4907. ACM, New York, New York, USA, May 2016.

[23] R. Slyper and J. Hodgins. Prototyping robot appearance, movement, and interactions using flexible 3D printing and air pressure sensors. In *2012 RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pp. 6–11. IEEE, 2012.

[24] S. Swaminathan, C. Shi, Y. Jansen, P. Dragicevic, L. Oehlberg, and J.-D. Fekete. Supporting the design and fabrication of physical visualizations. In *CHI'14: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3845–3854. ACM Press, New York, New York, USA, 2014.

[25] B. Taylor, A. K. Dey, D. Siewiorek, and A. Smailagic. Customizable 3D Printed Tactile Maps as Interactive Overlays. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 71–79. ACM, New York, New York, USA, Oct. 2016.

[26] N. Umetani, A. Panotopoulou, R. Schmidt, and E. Whiting. Printone: interactive resonance simulation for free-form print-wind instrument design. *ACM Transactions on Graphics (TOG)*, 35(6):184–14, Nov. 2016.

[27] P. Welch. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2):70–73, June 1967.