Lamello: Passive Acoustic Sensing for Tangible Input Components

Valkyrie Savage*†, Andrew Head†, Björn Hartmann†, Dan B Goldman*, Gautham Mysore*, Wilmot Li*
* Adobe Research, † UC Berkeley EECS
{valkyrie,andrewhead,bjoern}@eecs.berkeley.edu, {dgoldman,gmysore,wilmotli}@adobe.com

ABSTRACT
We describe *Lamello*, an approach for creating tangible input components that recognize user interaction via passive acoustic sensing. Lamello employs comb-like structures with varying-length tines at interaction points (e.g., along slider paths). Moving a component generates tine strikes; a real-time audio processing pipeline analyzes the resultant sounds and emits high-level interaction events. Our main contributions are in the co-design of the tine structures, information encoding schemes, and audio analysis. We demonstrate 3D printed Lamello-powered buttons, sliders, and dials.

Author Keywords
3D Printing; Sound; Tangible Interaction

ACM Classification Keywords
H.5.2 User Interfaces: Prototyping

INTRODUCTION
Tangible input components have advantages over "flat" interfaces like touch screens. They are critical for eyes-free interaction and muscle memory, and can improve task speed and precision [9]. Such devices typically comprise integrated assemblies of electronics, enclosures, and microcontroller code.

Recently, researchers have begun to explore acoustically sensing interactions – such as scratching – with digitally-fabricated objects that encode information in surface textures [7, 11]. In this paper, we extend this line of work from sensing surface features to sensing interactions with tangible components that users can push, slide, and turn.

Our technique, Lamello, integrates algorithmically-generated tine structures into movable components to create passive tangible inputs (Figure 1). Manipulating these inputs creates sounds which can be captured using an inexpensive contact microphone and interpreted using real-time audio signal processing. Lamello predicts the fundamental frequency of each tine based on its geometry: thus, recognition does not require training examples. The decoded high-level events can then be forwarded to interactive applications. The name "Lamello" is derived from the Lamellophone family of instruments, which create sound through vibrating tongues of varying lengths.

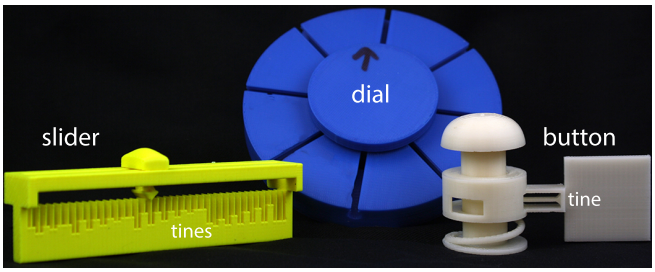

Figure 1. Passive tangible inputs that can be sensed acoustically.

Recognizing mechanically-generated sound for input has important limitations – only movement generates sound, so steady state cannot be sensed – but also appealing characteristics: Components can be fabricated from a single material (e.g., 3D printed ABS plastic), and "wiring" only requires attaching a microphone. In this paper, we provide design and fabrication guidelines, and demonstrate several components that use the Lamello approach. Our evaluation shows that training-free recognition is possible, though our recognizer only has useful accuracy for a subset of tested tines.

RELATED WORK
Lamello builds on prior work in interactive device design and acoustic sensing. Recent work proposes a variety of approaches for creating interactive objects via digital fabrication: Printed Optics [16] uses printed lightpipes in optically-sensed input components; Sauron [13] modifies hollow 3D models to enable an embedded camera to sense interactions. These systems generate geometry designed to work with specific recognition techniques. We adopt a similar strategy but focus on acoustic sensing for enabling interactivity.

Passive acoustic sensing has been used for detecting mid-air gestures (SoundWave [3]), enabling touch sensing on the forearm and palm (Skinput [6]), distinguishing between different objects striking a touch surface (TapSense [5]), and recognizing stroke gestures on surfaces [4]. Active swept-frequency ultrasound sensing, which uses controlled audio output to infer interactions, can detect touch and grasp gestures on rigid objects [12]. The most relevant previous systems are Acoustic Barcodes [7] and Stane [11], which encode

CHI 2015, April 18–23, 2015, Seoul, Republic of Korea.
ACM 978-1-4503-3145-6/15/04.
http://dx.doi.org/10.1145/2702123.2702207

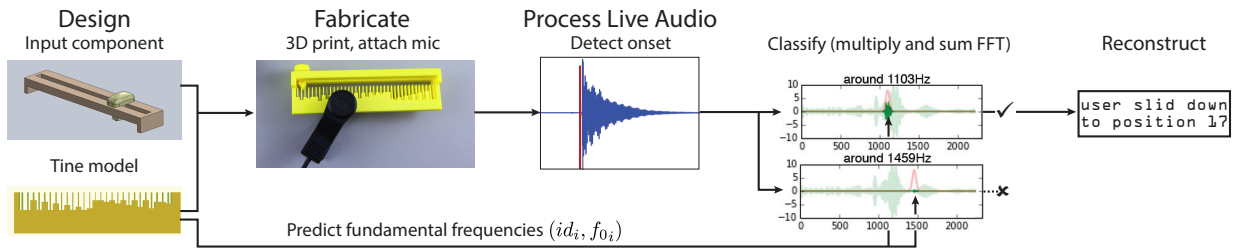

Figure 2. Lamello reuses information between the design, fabrication, and audio processing steps to allow training-free passive acoustic sensing.

information in fabricated objects that emit specific, detectable sounds during interaction. We extend their contributions: Our 3D tines encode additional information in their vibration frequencies, and we enable a wider variety of interactions than stroking or scratching. Designing specific vibration frequencies for objects has also been explored in procedurally generated metallophones [14]. Other approaches to providing passive tangible input include IR markers [15], capacitive signatures [1], and magnetic fields [8]. Audio sensing has the benefit that it does not require direct contact or line-of-sight.

LAMELLO: AUDIO GENERATION AND PROCESSING

There are two main challenges in developing a passive acoustic sensing technique that supports a variety of input controls: designing the physical mechanisms for generating sounds and developing recognition algorithms that can interpret those sounds in the intended manner (Figure 2).

Generating sounds with tines

To generate sounds, we embed tine structures in input components (Figure 1). Our tines are rectangular beams, attached at their base to the component and free to deflect at their top. Interacting with a component causes tine plucks; these vibrate the body of the component and are captured by a contact microphone. Tines can be arranged in configurations supporting different interactions (e.g., sliding, rotating, pressing).

Acoustic uniqueness: Different tines should generate unique, distinguishable sounds: we achieve this by systematically varying tine geometry. We model a vibrating tine as an ideal cantilevered beam of uniform density in free vibration [10]:

$$f_0 = \frac{1.875^2 \sqrt{\frac{E\frac{bh^3}{12}}{\rho(bh)L^4}}}{2\pi} \text{Hz}$$

Fundamental frequency (f_0) is governed by several variables: tine breadth (b), height (h), and length (L), as well as material properties (density ρ, Young's Modulus E). Our designs keep b and h constant, varying L to achieve different frequencies.

Our prototypes are 3D printed, resulting in non-uniform material deposition. To test the applicability of our model, we compared predicted and observed f_0 for several tines printed on two uPrint SE Plus FDM printers using Stratasys ABSplus-P430 thermoplastic. We find an appropriate material parameter by minimizing the error between observations and measurements. Fitted E values ranged from 9500 to 15500 based on print orientation and particular printer. The remaining error $\mu = 69.0Hz$ ($\sigma = 112.5Hz$) shows our model usefully applies to printed tines. Estimation of f_0 can be further improved by measuring post-print with calipers.

Figure 3. We experimented with two different encoding mechanisms for sliders: linearly increasing sequences (left) and de Bruijn (right). de Bruijn sequences allow classification of fewer tine lengths, but require more consecutive tine recognitions to determine position and direction.

Physical and fabrication constraints: A tine which is too thin can break or hit adjacent tines when struck, reducing recognition accuracy. A too-thick tine requires greater striking force, interfering with a user's experience. In our experience, ABS tines need $1{:}25 < thickness : length < 1{:}2$ to have appropriate stiffness for classification. Tine performance also depends on print orientation: on fused-deposition modeling machines, a tine is most reliable with its length laid in the printer's XY plane, as it is thus filled by a continuous extrusion. When printed in Z, tines can break as layers separate. Filleting can mitigate this problem, but may reduce the accuracy of the vibration model in predicting f_0. We fabricated tines with f_0 between $400Hz$ ($4mm$ x $50mm$ x $6mm$) and $4000Hz$ ($7.25mm$ x $6.0mm$ x $1.2mm$). Minimum tine size is determined by printer limitations: our printers have Z resolution $0.254mm$ and minimum XY feature size $1.194mm$.

Alternative fabrication techniques: Other printing or fabrication processes may not be orientation dependent. We have laser cut tines from Polyoxymethylene (Delrin) sheets, integrating these tine strips into 3D printed components. Tine sizes are similar, as laser cutting caused heat deflection in smaller feature sizes. Smaller tine sizes and higher frequencies may be achievable using different fabrication processes, e.g., injection molding or MEMS micromachining.

Encoding information: We use unique f_0s to differentiate buttons and directions on a D-pad. For position sensing, f_0 can increase across the range of motion (Figure 3 left). If more distinctions are needed than can be reliably recognized by varying f_0, we create de Bruijn patterns [2] (Figure 3 right). A de Bruijn sequence $D(k, n)$ is one which, given an alphabet size k and a subsequence length n, contains each subsequence exactly once: we can uniquely infer sequence position from n recognitions. This requires fewer f_0s, but more contiguous tine recognitions to determine user input.

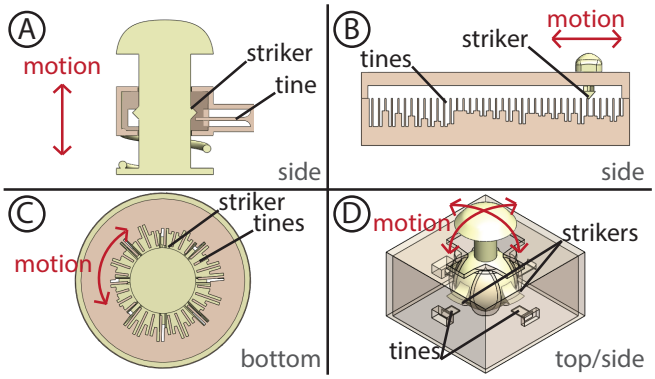

Figure 4. The Lamello technique can be used to sense a variety of physical motions, including up/down on a button, back/forward on a slider, and rotation on a dial or scroll wheel. Four tines used together can sense up, down, left, and right on a direction pad.

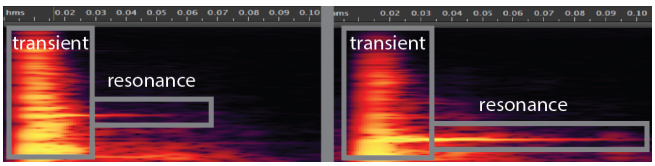

Figure 5. Two typical tine strikes (100ms): high-frequency (left) and low-frequency (right). We mark transients and resonance. Note the higher frequency has less energy (darker colors) and faster decay (shorter).

Integration of tines into larger components

We augmented several traditional input components: buttons, sliders, dials, and joysticks. Each has a "striker" attached to the user-facing "handle" (Figure 4). These strikers overlap with tine ends by $0.25 - 1mm$, balancing clear signal generation with easy interaction. Through testing, we determined that a triangular striker profile works best. The button has a rib around its shaft that strikes a tine when a user depresses it. The slider has a wiper that overlaps with the tops of tines (tines have different lengths, but are top-aligned). The dial works similarly, arranged radially rather than linearly. The D-pad derives from the button: a striker strikes a tine on the base as the user moves the handle up, down, left, or right.

Audio processing pipeline

The audio signal of a tine strike is characterized by an initial transient—a short high energy sound across a wide range of frequencies—followed by free vibration with a local long-decay energy peak at the tine's resonant frequency (Figure 5). Conceptually, our recognizer detects a transient, finds the dominant resonant frequency after the transient passes, and compares it to predicted tine frequencies.

Our audio processing pipeline, written in Python, uses basic frequency-domain features for classification. We sample our contact microphone at $16000Hz$. Our frames are 2048 samples ($128ms$), and our hop length (offset between successive, overlapping frames) is 800 samples ($50ms$), for a frame overlap of 61%. Analyzing a frame takes $5ms$, plus additional latency incurred by sound hardware. In addition to the real-time audio stream, our recognition algorithm also takes an ordered list of (id_i, f_{0i}) tuples describing the tine ordering and fundamental frequencies of a component as input.

For each frame, we first determine if a tine strike is present using a standard onset detector (with an empirically-determined amplitude threshold). Once an onset is detected, we wait 2 frame hops for the transient response to pass. We classify the subsequent frame (computation time: $5ms$). Our best-case onset-to-classification latency is therefore $2 * 50ms + 5ms = 105ms$. In practice, we have seen latency of $107.3ms$ (σ=$9.67ms$). Our sound card and driver introduce latency as they collect and report blocks: one could reduce overhead with optimized sound drivers and sample block sizes.

To classify, we compute a Fast Fourier Transform on the window, then normalize FFT bin values to represent fractions of overall audio energy. For each tine id_i, we generate a new metric: the dot product of the scaled FFT and a Gaussian centered at the bin for f_{0i}, representing the fraction of audio energy e_i in the neighborhood of f_{0i}. To account for lower energy at higher frequencies, we use a scaling factor proportional to the frequency and a σ for the Gaussians empirically determined per component, giving an adjusted list of e_{iadj}.

Mapping a recognized tine identity $id_R = argmax(e_{iadj})$ to user actions is straightforward. For buttons and joysticks, id_R maps directly to a discrete input (press, up, down, left, or right). Similarly, for dials and sliders that encode position with linearly increasing tine lengths, id_R maps to a unique position. For dials and sliders that use a de Bruijn sequence $D(k, n)$, we use each sequence of n recognized tines to determine the corresponding position within the sequence.

EVALUATION

To determine whether Lamello tines can be classified reliably and in real-time, we performed an accuracy test and built a real-time demonstration using Lamello controls.

We recorded 10 strikes for each tine on a printed dial and slider, and a laser-cut Delrin slider. Printed components were actuated with their strikers; Delrin tines were hand-plucked to determine effects of different striking methods. We classified each strike and report classification accuracy in Table 1.

We achieve promising accuracy (precision = 93%, recall = 90%) with a four-tine slider (predicted frequencies 924, 1103, 1340, and $1662Hz$). This suggests that useful interaction with Lamello is within reach. However, precision and recall rates are much lower on a set of 7 tines with frequencies above $2kHz$: the recognizer fails to classify higher f_0, which have lower energies and shorter decays (Figure 6).

To explore the utility of our technique, we used a Lamello slider and mouse emulation to control a Pong game. The achieved latency was sufficient for simple gameplay; however, Lamello may be better suited to provide input for applications such as volume or lighting control, where some latency and occasional misclassified events are acceptable.

Control	P_{4tines}	R_{4tines}	P_{7tines}	R_{7tines}
FDM Slider	93%	90%	49%	56%
FDM Dial	90%	85%	63%	54%
Plucked Delrin	98%	97%	72%	73%

Table 1. Recognition precision and recall of our printed input components using model geometry-predicted frequencies.

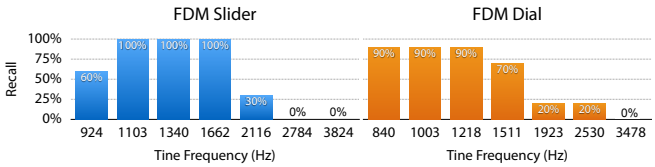

Figure 6. Per-tine recall for striking Lamello slider and dial tines. We observe high recall for low-frequency subsets of tines.

DISCUSSION AND FUTURE WORK

Initial experiments with Lamello are encouraging: components augmented with tines are easy to print and use, and tines produce unique, predictable frequencies. However, classification accuracy still needs improvement, and may require a new approach for $f_0 > 2kHz$. We identified several sources of errors to address in future work:

Striker mechanism: Finger-plucked tines having higher recognition rates than striker-actuated tines suggests that striker-created noise contributes to misclassifications.

Signal attenuation: While microphones placed at opposite ends of a printed slider produce similar overall accuracy, tines are more correctly classified by the closer microphone. Though we could not directly correlate microphone distance and signal RMS energy, this suggests minimizing distance between microphone and tines may improve accuracy.

Resonance and harmonics: Struck tines exhibit an energy peak at the predicted f_0, but their frequency spectrum is considerably more complex due to harmonics, component resonance, and other unmodeled material effects (e.g., the layered construction of 3D prints). Competing with non-fundamental vibrations is most problematic for short tines, which have lower energy. Future work can also probe optimal frequency distributions to avoid overlap between tine harmonics.

Beyond recognition rates, the Lamello approach can only detect position *changes*—it cannot sense static configurations, nor can we currently distinguish between the two directions in which a tine can be struck. In the future, we envision a library of parametric models added to larger tangible devices: we plan to leverage existing microphones in smartphones and tablets for sensing. This would open applications beyond prototyping (e.g., custom controllers for tablet games), but will require more sophisticated signal processing to filter out environmental noise and to deinterlace multiple input components manipulated simultaneously.

In conclusion, we have demonstrated the use of mechanically generated sound to support interactions beyond strokes [7] and scratches [11]. We hope our work opens a larger space of Lamellophonic input devices.

ACKNOWLEDGEMENTS

This work is supported in part by the NSF under grant DGE 1106400 and a Sloan Fellowship.

REFERENCES

1. Chan, L., Müller, S., Roudaut, A., and Baudisch, P. CapStones and ZebraWidgets: Sensing stacks of building blocks, dials and sliders on capacitive touch screens. In *CHI '12*, 2189–2192.
2. de Bruijn, N. G. A combinatorial problem. In *Koninklijke Nederlandse Akademie v. Wetenschappen 49*, Indagationes Mathematicae (1946), 758–764.
3. Gupta, S., Morris, D., Patel, S., and Tan, D. SoundWave: Using the Doppler effect to sense gestures. In *CHI '12*, 1911–1914.
4. Harrison, C., and Hudson, S. E. Scratch input: Creating large, inexpensive, unpowered and mobile finger input surfaces. In *UIST '08*, 205–208.
5. Harrison, C., Schwarz, J., and Hudson, S. E. TapSense: Enhancing finger interaction on touch surfaces. In *UIST '11*, 627–636.
6. Harrison, C., Tan, D., and Morris, D. Skinput: Appropriating the body as an input surface. In *CHI '10*, 453–462.
7. Harrison, C., Xiao, R., and Hudson, S. Acoustic barcodes: Passive, durable and inexpensive notched identification tags. In *UIST '12*, 563–568.
8. Hwang, S., Ahn, M., and Wohn, K.-y. MagGetz: Customizable passive tangible controllers on and around conventional mobile devices. In *UIST '13*, 411–416.
9. Klemmer, S. R., Hartmann, B., and Takayama, L. How bodies matter: Five themes for interaction design. In *DIS '06*, 140–149.
10. Meirovitch, L. *Analytical Methods in Vibration*. The MacMillan Company, 1967.
11. Murray-Smith, R., Williamson, J., Hughes, S., and Quaade, T. Stane: Synthesized surfaces for tactile input. In *CHI '08*, 1299–1302.
12. Ono, M., Shizuki, B., and Tanaka, J. Touch & Activate: Adding interactivity to existing objects using active acoustic sensing. In *UIST '13*, 31–40.
13. Savage, V., Chang, C., and Hartmann, B. Sauron: Embedded single-camera sensing of printed physical user interfaces. In *UIST '13*, 447–456.
14. Umetani, N., Mitani, J., and Igarashi, T. Designing custom-made metallophone with concurrent eigenanalysis. In *NIME '10*, 26–30.
15. Weiss, M., Wagner, J., Jansen, Y., Jennings, R., Khoshabeh, R., Hollan, J. D., and Borchers, J. SLAP Widgets: Bridging the gap between virtual and physical controls on tabletops. In *CHI '09*, 481–490.
16. Willis, K. D. D., Brockmeyer, E., Hudson, S., and Poupyrev, I. Printed optics: 3D printing of embedded optical elements for interactive devices. In *UIST '12*, 589–598.