SoundBubble: Finger-Bound Virtual Microphone using Headset/Glasses Beamforming

Daehwa Kim
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
daehwak@cs.cmu.edu

Chris Harrison
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
chris.harrison@cs.cmu.edu

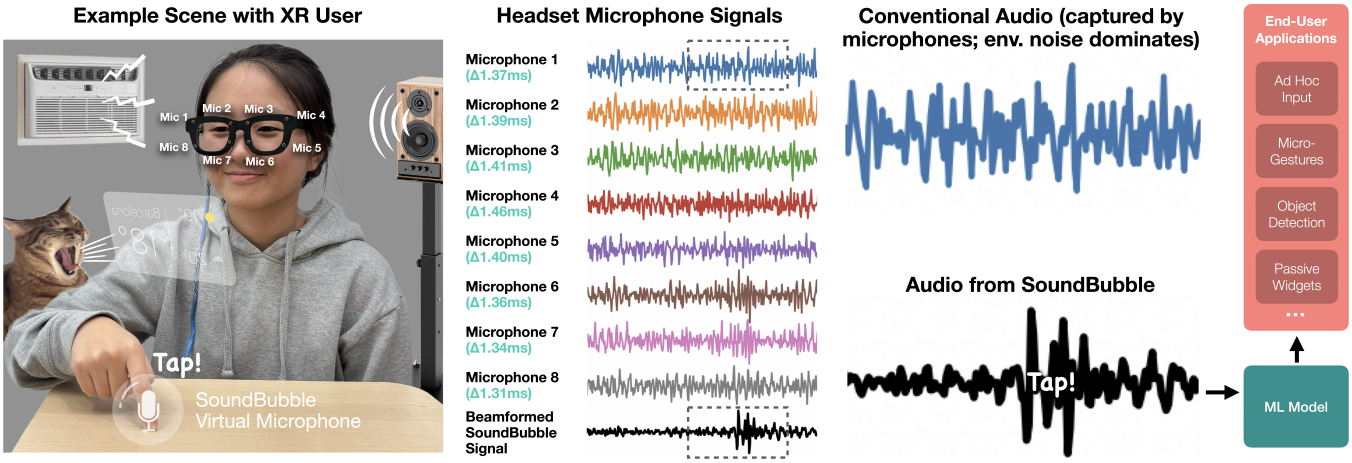

Figure 1: From left to right: Using an XR headset/glasses microphone array, SoundBubble uses beamforming to focus on sounds emanating from a user's finger (in this example scene, the sound of a tap to an ad hoc surface). The hands are already tracked in 3D space by the headset cameras, allowing SoundBubble to calculate the precise propagation delay for each microphone (cyan subheadings; real acoustic data shown). This allows a beamformer to suppress uncorrelated background noise (e.g., HVAC, music, pets), while capturing the tap signal. The resulting sound is then passed to a machine learning model that can power a variety of potential end-user applications that we discuss in the paper.

Abstract

Hands are the chief appendage with which we manipulate the world around us, creating sounds as they go. As such, they are a rich source of information that computers can leverage for input and context sensing. Indeed, many prior works in HCI have explored this idea by instrumenting users' hands with a microphone, often integrated into a ring, wristband, or watch. In this work, we explore an alternative bare-hands approach — by using a microphone array integrated into a user's headset/glasses, we can use beamforming to create a virtual microphone that tracks with the user's fingers in 3D space. We show this method can capture even the subtle noise of a finger translating across surfaces, including skin-to-skin contact for micro-gestures, as well as passive widget interactions.

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/2026/04
https://doi.org/10.1145/3772318.3791589

CCS Concepts

• **Human-centered computing** → **Sound-based input / output**.

Keywords

Acoustic sensing, activity recognition, touch input, beamforming, VR, XR, AR, glasses, headset.

ACM Reference Format:
Daehwa Kim and Chris Harrison. 2026. SoundBubble: Finger-Bound Virtual Microphone using Headset/Glasses Beamforming. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3772318.3791589

1 Introduction

Hands are our main method for interacting with the physical world, producing a variety of sounds as they grasp, tap, and manipulate objects and mechanisms. This "noise" carries valuable information about user actions and the environment. By capturing these acoustic cues, computers and future AI assistants could infer user input, intent, activity, and context, enabling new interactions and reducing interactive viscosity [76]. In response, considerable prior work

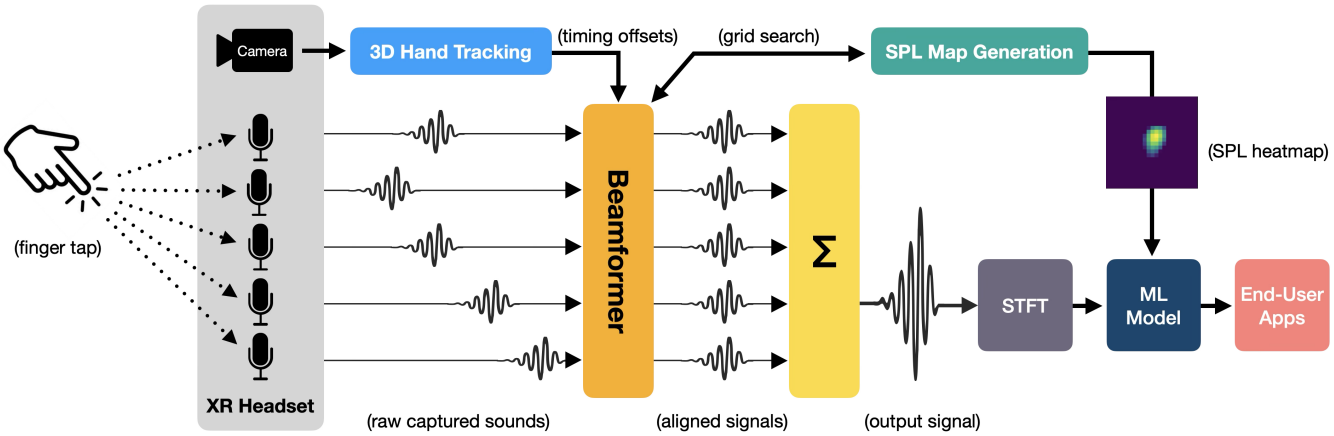

Figure 2: Left to right: A finger taps a surface, creating a sound. The sound propagates outward, arriving at five headset microphones at slightly different times. Simultaneously, the XR headset/glasses camera tracks the 3D finger position, allowing the varying propagation delays to be nullified and the signals summed into a new superior audio stream. Our beamformer also creates a sound pressure level (SPL) map of the region around the finger. The audio and SPL map are fed into an ML model, which detects tap events.

in Human-Computer Interaction (HCI) has explored this concept, generally by equipping users with vibro-acoustic sensors worn on the hand (e.g., nail- [75, 84], ring- [16, 21, 25, 53, 56], and watch-like [41, 46, 49, 68, 101] devices). Of course, many users do not wear such accessories, and if they do wear a smartwatch, it is most often worn on the non-dominant hand. Further, the small size of these devices generally implies small batteries requiring frequent recharging, especially if they are continuously sampling audio.

In this work, we propose and evaluate a new bare-hands approach for capturing hand-generated sounds called *SoundBubble* (Figure 1). In essence, this is a virtual microphone that we can place right at the user's fingertips, where sounds of interest are generated. To achieve this, we require XR headsets and glasses with integrated microphone arrays (which already exist; see Section 3.4). We use passive acoustic beamforming to receive and recover signals, guided by vision-based 3D hand tracking. Such acoustic beamforming has the added benefit of attenuating extraneous environmental noise, giving us a signal-to-noise ratio (SNR) superior to that of a single microphone (e.g., in a smartwatch), which faces the (sometimes impossible) challenge of separating noises generated by the user's hand from background noise in the environment. Our approach inherently isolates *all* finger-generated sounds, and it does so algorithmically and deterministically, without having to rely on a sound separation deep learning model *pre-trained* on a *subset* of classes (e.g., [30, 66, 78]), making our approach considerably more generalizable.

To showcase the technical feasibility of SoundBubble, we created four, functional, proof-of-concept example applications: ad hoc touch input, micro-gestures, held object activity recognition, and passive widget input. We test three of these applications in a user study, and compare directly to a model using only a single microphone's data. As we will discuss in greater detail, a single microphone cannot readily separate background from desired signals, and thus suffers from nearly triple the number of false positive events. We also use our data to simulate different device form factors, ranging in size between today's headsets and future XR glasses. Overall, we believe SoundBubble opens intriguing new interactions in XR, and in some cases, alleviates prior work from the need to instrument users' arms and hands with sensors.

2 Background & Related Work

First, we provide a brief primer on acoustic beamforming. We then review human-computer interaction uses of acoustic sensing (and vibro-acoustic sensing more broadly) using wearable devices. We conclude with the closest body of related work — array microphone beamforming systems for human-computer interactions.

2.1 Background

We refer interested readers to Van Trees [93], Johnson and Dudgeon [36], and Van Veen and Buckley [94] for an excellent introduction to beamforming, which is applicable to all types of energetic wave phenomena, including sound, vibration, and RF. We provide a very brief primer here.

In the acoustics domain, the "speed of sound" is the most important constant. In air, sounds propagate at approximately 343 m/s. This means that many microphones distributed in space (i.e., an "array") will receive sound signals at slightly different timing offsets (other than in the very specific case of a planar wavefront that is perfectly normal to a planar array). If one simply sums these received signals, they will most often destructively interfere with one another, as the signal's peaks and troughs will misalign. However, if one knows the origin source of a sound in 3D space relative to the microphone array, the precise timing offsets (also called the "time of flight") for each microphone could be computed and corrected for, such that the target signals constructively interfere. The left half of Figure 2 illustrates this process (and real-world signals are shown in Figure 1). Importantly, extraneous noises in the environment, with varying and unknown origins, generally destructively

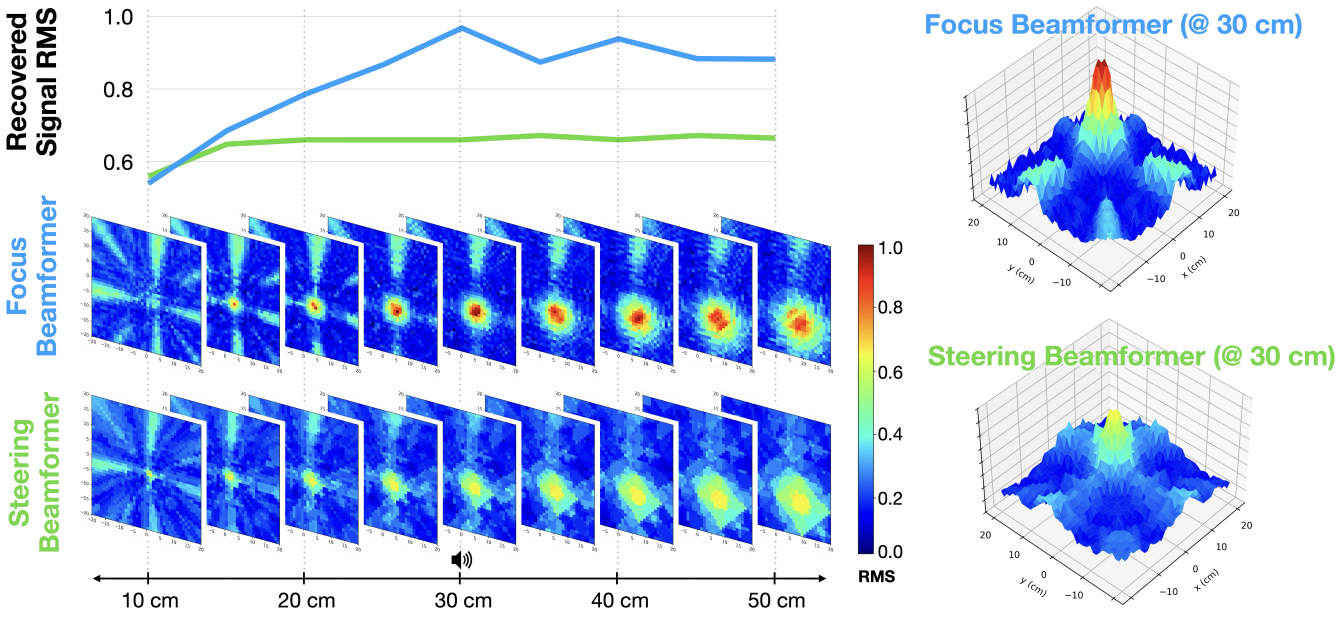

Figure 3: Simulated spatial response map (normalized RMS acoustic pressure) comparing two different beamforming strategies. In this example, a virtual sound source (but real-world recording of white noise) is placed 30 cm away, and 5 cm downwards and to the right. Note that the focal beamformer (focused on the sound source) achieves peak RMS at the sound source, while the steering beamformer (steered towards the sound source) offers roughly flat performance along its entire sensing vector (which is less desirable). To the right, we plot the response maps at 30 cm for both beamformers.

interfere, giving the target sound source a boost in SNR. This is the basis for the simple but effective delay-sum beamformer [7], which has been in use since the early 1900s.

In an open space, sounds generally propagate outwards, omnidirectionally, forming a spherical wavefront. When the sensing array is distant from the source, the small signal "patch" received, though spherical, can be approximated as a plane. This assumption is widely used by most beamforming algorithms, which can then utilize a "steering vector" that operates like a directional microphone, sensitive to sounds incoming from a specific direction. Our interactions, on the other hand, are in the near field, proximate to the array — most often less than 60 cm. At these smaller distances, the received wavefront has a spherical geometry, and so we must calculate *unique* timing offsets for *each* microphone in the array (as opposed to a single, fixed offset, as is done with a traditional steering vector). This has the benefit of making our array *focus* on a specific 3D *point* in space, as opposed to *steered* to listen along a 3D *vector*. Note that although we name our technique SoundBubble, the acoustic foci are not spherical, but rather an elongated lobe as seen in Figure 3. While our approach is sensitive to sounds extending beyond the focal point, its peak sensitivity is at the focal point, and then diminishes beyond this — a superior behavior to a standard steering vector beamformer.

In mathematical form, let $P_{ij} \in \mathbb{R}^3$ denote the position of microphone (i, j) and $P_{\text{finger}} \in \mathbb{R}^3$ the focal point (i.e., index fingertip). The propagation distance D_{ij} from the focal point to microphone (i, j), the corresponding discrete-time delay Δ_{ij} (in samples) with c_s denoting the speed of sound and f_s the audio sampling rate, and the beamformed signal $S_{\text{bf}}[n]$ are

$$D_{ij} = \|P_{ij} - P_{\text{finger}}\|, \quad \Delta_{ij} = \text{round}\left(\frac{D_{ij}}{c_s} f_s\right), \quad S_{\text{bf}}[n] = \sum_{i,j} S_{ij}[n - \Delta_{ij}]$$

2.2 Worn Vibro-Acoustic Sensing in HCI

Hands are our primary means of interacting with the physical world, and these interactions often produce distinctive mechanical vibrations, which includes vibrations that propagate through the air (i.e., acoustic). These vibro-acoustic signals can be captured and leveraged by computers for human-computer interactions. Prior work has explored this idea in contexts such as augmented object interaction [29, 77, 80], ad hoc surface touch input [6, 19, 28, 33, 58, 83, 97], environment and activity recognition [108, 109], etc. However, all of these systems require instrumenting the environment (surfaces and/or objects) with vibration and acoustic sensors.

More practical and scalable, and closer to the present work, is to equip the user with a worn sensor, allowing for capabilities that move with the user. Researchers have explored various wearable form factors for capturing interaction sounds, including nail-mounted [75, 84], ring-based [16, 21, 25, 53, 56], and wrist-worn devices [41, 46, 48, 49, 68, 101]. To maximize signal-to-noise ratio (SNR), sensors are typically placed close to the sound source. IMUs embedded near the hand, particularly on the fingernail, have been used to detect discrete taps [75] and continuous gestures [84]. These designs reveal an inherent proximity vs. accuracy tradeoff:

when the sensor is positioned farther from the fingertip [84], performance degrades. Ring-like devices have also leveraged IMUs to detect taps [16, 21] and gestures involving the hand, surfaces, or everyday objects [53]. On the wrist, systems have incorporated contact microphones along with other sensors to support input techniques such as ad hoc touch [41] and thumb-to-finger microinteractions [101]. Broader hand activities can also be recognized using wrist-borne accelerometers [48] or microphones in smartwatches [35, 68]. Finally, there are optical approaches for sensing vibro-acoustic signals, such as laser speckle sensing [86] used for ad hoc touch detection.

The aforementioned worn systems all sensed signals generated by external events, but it is also possible for a device to be active, generating its own vibrations and sounds for sensing. For instance, a speaker can emit inaudible acoustic chirps, which are then captured by a microphone after reflecting off the environment. This active sensing approach is commonly used for pose estimation tasks, including face, body, and hand tracking, as well as inferring inherent activities. Recent work has demonstrated this technique in devices such as rings [105], wristbands [32, 50], smart glasses [4, 51], earbuds [107], and mobile phones [18].

Unlike most of the systems above, SoundBubble does not require special instrumentation of the hands, and could be tightly integrated into future headsets/glasses (in fact, several current XR headsets and glasses already contain the requisite sensing hardware). Additionally, our system passively listens for signals (i.e., does not emit an active signal), but yet offers relatively high SNR through the use of beamforming. We note that other systems relying on conventional mono/stereo microphones face the significant challenge of detecting relevant signals in the presence of background noise. In our example applications and evaluation, our target signals were 100× quieter than a noisy background (see Section 5.3). Finally, IMU-driven systems generally have to worry less about background noise, but have to be worn on or near the hand to capture relevant signals. SoundBubble offers a useful mix of properties from each approach: bare-hand operation and good background noise suppression.

2.3 Mid-Air and Bare-Hand Sensing

A substantial body of work examines *around-device* interaction using cameras. Examples include ring- or finger-mounted cameras for on-body and near-surface gestures [8, 38], mobile-device–centered sensing of hand motion [23], and head-mounted cameras for mid-air manipulation. These approaches leverage visual motion cues and can capture rich trajectories, but often struggle with fine contact events (e.g., hover vs. touch) or interactions occluded by the user's own hands.

Acoustic techniques have also been used for mid-air sensing, frequently in an *active* configuration where a speaker emits a known signal and microphones measure reflections, Doppler shifts, or echo profiles. Examples include ultrasonic time-of-flight systems for 3D finger tracking [50, 52], Doppler-based gesture input [5, 23, 89], swept frequency ultrasonic reflections [47], chirp-based channel–perturbation methods for near-surface thumb gestures [31], and wrist- or ring-mounted active systems that infer proximity and motion [51, 74].

Of note, few of these prior works have utilized acoustic beamforming (more discussion in the next section), and none of these prior works have utilized vision-guided acoustic beamforming. We show that this powerful new approach can support multiple use modalities, whereas most previous approaches had to be specialized. Furthermore, accuracies our system achieves surpass most of these systems, despite not being designed for a specific interaction task.

2.4 Beamforming / Array Microphones in HCI

Acoustic beamforming provides two key capabilities: sound source localization and separation. In HCI, beamforming with multiple microphones has been effectively used for spatially augmented input via source localization. One example is smart speakers that can infer the direction of a user's voice [3, 103]. Beamforming has also enabled near-device gestural interactions on mobile phones [13, 42, 71, 97] and smartwatches [20, 26, 106], where it is used to track finger location with taps, flicks, or finger-worn speakers. Another line of work has explored desk-mounted speaker and microphone arrays to recognize users' facial expressions [17]. Finally, sound source visualization has been explored in XR headsets to enhance human cognition, supporting both accessibility applications [22] and synesthetic experiences [45].

The second key capability, sound source separation, is widely applied in consumer microphone systems such as conference room devices and AV systems, where background noise is suppressed to enhance the voice of the active speaker [2, 12]. These systems are typically non-wearable. A few studies have explored wearable implementations, where microphone arrays are integrated into headphones [9, 10, 30] or neckties [102] to enable amplification of the wearer's voice.

Most closely related to our work is SoundCraft [26], which employs four-microphone beamforming on a smartwatch to support interactions such as ad hoc touch input and tool-based interactions performed near the watch, use cases we also explore and prototype. We note that while SoundCraft can localize and separate sounds, it does not know if they are being generated by the user — the system knows the angle of arrival, but not the distance. For this reason, SoundCraft considers longer-range interactions, like detecting if a user is standing in front of a 3D printer, whereas SoundBubble focuses on sounds emanating from within a small zone around the user's hand. Moreover, our SoundBubbles are entirely virtual — we can instantiate them anywhere we please, for instance, one for each finger — offering significant flexibility in placement vs. integration into a physical smartwatch.

2.5 Comparison to Vision Approaches

We believe SoundBubble exemplifies the power of vision-acoustic sensor fusion. In Section 4, we discuss four example uses of our system. As summarized in Figure 4, these are challenging to implement with vision or audio alone.

For example, the challenge of detecting if a finger is closely hovering above a surface versus actually physically touching using computer vision alone has been well established in the literature [27, 70, 81, 87, 98], whether it be with RGB cameras or depth cameras (from recent work: "While passive surfaces offer numerous benefits for interaction in mixed reality, reliably detecting touch

Sensing Approach	Ad Hoc Touch Detection & Micro-Gestures	Held-Object Activity Recognition	Passive Object/Widget interaction
Vision-only	**Easy** to track finger location, **feasible** to disambiguate hover vs. touch.	**Easy** to identify held object, **harder** to recognize the state of object.	**Feasible** to detect finger-to-widget contact, **hard** to tell when widget is pressed.
Audio-only	**Hard** to track finger direction or location, **feasible** to detect touch events; **struggles** with bg noise.	**Feasible** to identify object and state, **hard** to detect whether it's held or nearby; **struggles** with bg noise.	**Easy** to detect a trigger, **hard** to identify which widget was triggered; **struggles** with bg noise.
Vision-Audio Fusion	**Easy** to track finger location, **feasible** to detect touch events, but **struggles** with background noise.	**Easy** to identify object, **feasible** to detect state, but **struggles** with bg noise and if many objects are running.	**Easy** to track fingers and detect widget triggers, but **struggles** with bg noise and if several fingers are resting on widgets.
Vision-Beamformed Audio Fusion	**Easy** to track finger location, **easy** to detect touch events, even in the presence of background noise.	**Easy** to identify object, **easy** to detect state, even in the presence of bg noise and if many objects are running.	**Easy** to track fingers and detect widget triggers, and **easy** to tell which widget is triggered, including with bg noise.

Figure 4: High-level comparison of four different approaches across three example interactive uses. Yellow-to-green scale indicates innate difficulty in supporting the use case. For ad hoc touch detection, imagine a finger hovering just above a surface. For held object activity recognition, consider the case of a user holding and running a power drill in a noisy workshop. For passive widget interaction, imagine the case of a user holding a rapidly-prototyped, laser-cut game controller.

input solely from headmounted cameras has been a long-standing challenge." [88] and "The problem with depth cameras, even today [2025], is noisy signal — the difference between a finger touching vs. slightly hovering above a surface is hard to distinguish." [69]). At the same time, although acoustic information has been utilized in detecting touch events [28, 34, 97], acoustic information alone is rarely enough to accurately *track* the touch. However, by leveraging both visual and acoustic data, SoundBubble can readily detect *and* track touch events on ad hoc surfaces.

Likewise, our second example use case is skin-on-skin micro-gestures, which produce a distinctive sound, but which are very hard to segment with vision alone due to the same hover vs. touch disambiguation problem, and are near-impossible to directionally disambiguate with audio alone (but which is easily done in the visual domain). SoundBubble implements such micro-gesture tracking as one of its four use cases, and we further note that our technique can also detect and track pen and pencil input, even when the tip of the implement is occluded (see Figure 6B and Video Figure 1:51). In all the latter cases, these are tasks computer vision alone would struggle to track, whereas fusion with a complementary acoustic signal makes this tractable.

Another of our example uses, passive widget interaction (e.g., clicking a button on a toy or 3D printed object), offers another point of discussion (see buttons in Figure 9, second row, and Video Figure 2:58-3:05). With the travel on most buttons being just a few mm or less, the visual change is subtle to detect with computer vision. This is made extra challenging if the object is in motion and being handled by the user, and if the observing cameras are in, e.g., an XR headset. Thus, a vision-only approach would be challenging to make robust, as would an audio-only approach (how does the model know what object the user is handling, or what button is being pressed?). However, again, vision-acoustic fusion makes this recognition task tractable, so much so that we implemented and evaluated this task as one of four example use cases.

Finally, we note that many of the signals discussed above are quiet — the clicking of a button, the sound of a finger dragging against a surface, the sound of a pen translating on paper, the rubbing together of two fingers — and that a traditional acoustic stream (i.e., mono or stereo sound) would be insufficient for detection in environments with typical levels of ambient noise. However, with acoustic beamforming, informed by vision-derived information (finger position in the case of SoundBubble), we can substantially improve acoustic SNR such that we can recover these subtle sounds (please refer to the Video Figure for a back-to-back comparison of conventional vs. beamformed audio). In our evaluation, we challenge our system by including background noise 100 × louder than our target signal (see Section 5.3); non-beamforming methods would struggle to extract such a signal from background.

Thus, in summary, we believe computer vision + beamformed acoustic sensor fusion is a new and powerful modality that has unique strengths in long-standing HCI use cases that have struggled in utilizing vision or audio signals alone (see also Section 4).

3 Implementation

We now detail the main hardware and software components of SoundBubble. Figure 2 provides an overview of our pipeline. Please also refer to the Video Figure.

3.1 Open Source

To facilitate replication, we have open-sourced our software stack at https://github.com/FIGLAB/SoundBubble. This also permits detailed exploration of our technical implementation, the small details of which are not appropriate to elaborate in a paper format. At this same URL, we provide links to commercially-available hardware and a "getting started" guide.

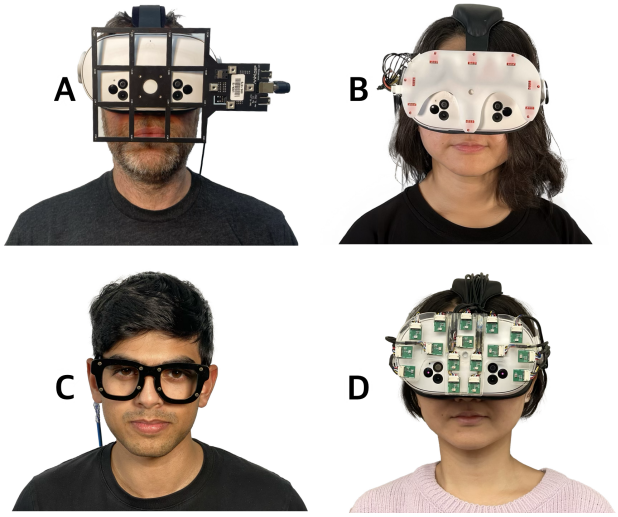

Figure 5: Over the course of development, we created several headset and glasses prototypes. These form factors are evaluated in our later ablation study (Section 6.3).

3.2 Hardware Platform

For development, we used a Meta Quest 3S as an exemplary XR headset, but we note that SoundBubble is platform agnostic. We also mocked up an XR glasses device, seen in Figure 5C, that is more similar to Snap's Spectacles [85] and Meta's Orion AR Glasses [63]. For ease of prototyping, our microphone arrays send data back to a MacBook Pro (2024) laptop over USB for processing, and this is where our software implementation runs. We note, however, that our implementation is not especially heavyweight and could run on a mobile-class CPU like that in a Quest 3S. In fact, there are several DSP chips that offer beamforming capabilities (e.g., [91, 99]), suggesting SoundBubble could even be moved to an embedded co-processor in the future (removing the processing and memory I/O burden from the application processor).

3.3 Hand Tracking

In order to target our acoustic beamformer, we must know where to focus in 3D space. For this, we use the Quest 3S' computer-vision-based hand tracker, and we attach a SoundBubble virtual microphone to the 3D position of the index fingertip keypoint. Note that we can attach virtual microphones anywhere in the field of view of the XR headset; the index finger serves as a useful demonstration of our technique. To interface with the Quest 3S, we wrote a basic Unity application that streams index finger position over WiFi to our software running on a laptop. For debugging, we stream back our beamforming results to the headset, visualized in the AR view.

3.4 Microphone Array

Our development started with commercial off-the-shelf microphone arrays, including the ReSpeaker Mic Array v2.0 (8 channels @ 44.1 kHz) [14] and MiniDSP UMA-16 v2 (16 channels @ 48 kHz) [67] (Figure 5A). To achieve a more compact form factor, we created three custom microphone arrays. Our first custom design was built using 8 ADMP401 MEMS microphones (Figure 5B) connected to an 8-channel Behringer UMC1820 USB Audio Interface offering 96 kHz sample rate. Using the same interface, we also built an 8-channel version modeled after the Meta Orion AR Glasses (Figure 5C) using miniature electret microphones (p.n. Fielect 4015P-40DB). To increase microphone count, we also built an array with 16 MSM261S4030HO MEMS microphones [61] read by an XMOS xCORE DSP [100] communicating over USB with a sample rate of 24 kHz (Figure 5D). For the latter three prototypes, we used laser-cut acrylic with slots for microphones to ensure planar alignment and accurate positioning.

For clarity, we re-emphasize our approach is entirely passive — our microphones listen, but we do not emit any signal. Likewise, the user's bare hands are tracked by the XR headset's cameras. Thus, SoundBubble represents a rare sensor fusion — a vision-guided acoustic beamforming approach — with output that could not be achieved with either sensor modality alone.

We note that microphone arrays built from commodity MEMS components are relatively inexpensive, small, and power-efficient. For example, the widely available InvenSense ICS-41350 MEMS microphone costs less than $0.75 in volume, measures just 3.5×2.65×0.98 mm, and consumes 185 μA in its always-on mode (and power could be reduced further by only turning on the microphones when the user's hands are raised into the scene or near an object, depending on the mode). As a result, microphone arrays have already been integrated into small, battery-powered, worn devices, such as Meta's Aria glasses (7-mic array), Meta's Ray-Ban Display Glasses (6-mic array), and Apple's Vision Pro (6-mic array). It may even be possible that SoundBubble could be enabled on these existing devices with a software update.

3.5 Calibration

Anytime a microphone array was affixed to the Quest 3S headset, we had to perform a one-time calibration to determine the relative 6DOF position/orientation of the array with respect to the headset. For this, we used a 30 cm aluminum dowel that snugly fit into a known calibration hole in each array. We then placed our finger onto the tip of this dowel, and read the position of the fingertip in 3D space with respect to the headset using the Quest's 3D hand tracking (averaged over many trials).

The Quest's vision-based hand tracker offers ~1 cm accuracy [24], and so we developed a second, more precise calibration process using this initial course estimate. Specifically, we render a small translucent virtual sphere in the AR scene at the initial coarse estimate. We then position the bottom speaker port of an iPhone at this location and emit white noise (in a sound booth, so that external noises do not contribute). We capture a few seconds of audio using the attached microphone array, and then perform an extensive 6DOF grid search (±1 cm spatially in 2 mm increments, and ±1° rotationally in 0.2° increments), calculating new simulated beamforming results, looking for maximum dB (i.e., highest correlated signal). We save the winning 6DOF offset.

3.6 Beamforming

Using the live-streamed multichannel microphone data and the live-streamed 3D finger position (from the Quest API, which uses

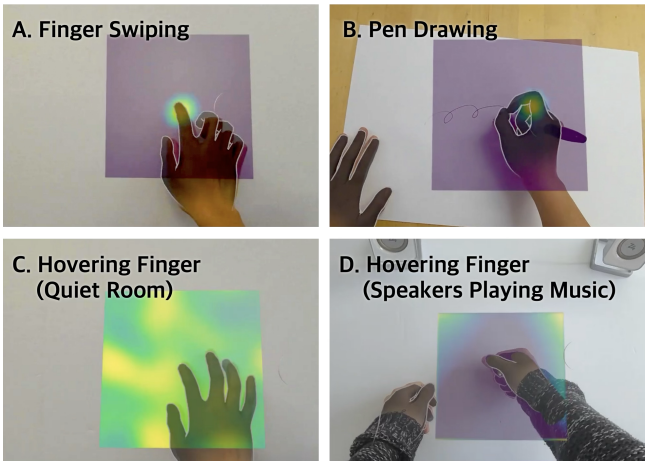

Figure 6: The sound pressure level (SPL) map for a finger swiping (A), drawing with a pen (B), ambient background noise (C), and music playing (D). In C and D, the finger is hovering above the surface and not producing any sounds, thus the SPL map does not show any sound produced at the finger (and instead shows other random correlations). Examples seen here are debug output as seen through the Quest 3S headset (i.e., SPL map overlaid onto the XR scene). Please see the Video Figure for more SPL map examples, including those with external noise sources.

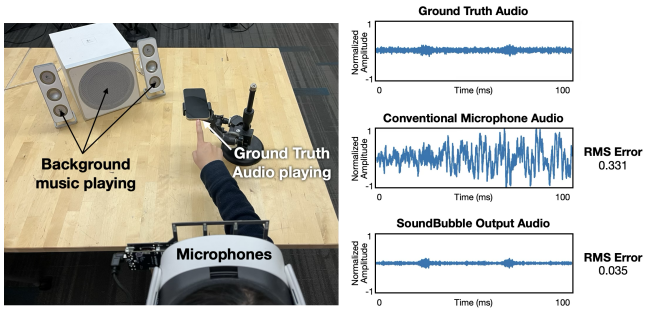

Figure 7: Left half: setup overview. A speaker plays background audio (music). Centered on the table, a smartphone plays a ground truth audio file. A user hovers their finger near the phone's speakers. Top-right: Waveform of the ground truth audio. Middle-right: Audio captured by a conventional microphone. Bottom-right: Audio output of SoundBubble. The RMS error of the signal compared to the ground truth is shown to the right. Note that SoundBubble is able to largely recover the original signal, despite the presence of noise, whereas a conventional microphone is dominated by background noise, obscuring the target signal.

Meta XR Core SDK [65]), we perform two distinct beamforming operations. First, we apply a 6kHz high-pass filter to the multi-channel audio. We then calculate the propagation delays given the relative distance between the finger and each microphone. We shift the incoming audio streams correspondingly, and produce a new, summed output audio stream (which is dominated by the target sound). Figure 2 illustrates this delay-sum process, and Figure 1 shows the effect with real-world data. Finally, the beamformed audio stream is converted into a frequency domain representation using a short-time Fourier transform (STFT), with a window size of 512 samples (256 sample overlap), which is passed as features into our machine learning model (Section 3.8).

We emphasize that the goal of this work was not to advance beamforming (but rather to be a new use of it), and we are agnostic to the beamformer method as long as it supports our application needs. That said, we did experiment with more advanced acoustic beamforming algorithms, but found several drawbacks that ultimately led us to rely on the classic delay-sum algorithm for our proof-of-concept implementation. Adaptive beamformers, such as the commonly used Minimum Variance Distortionless Response (MVDR/Capon) [36, 93], dynamically adjust weights to minimize output power while preserving gain in the desired direction. This can yield higher resolution and stronger interference suppression than delay-sum, but it requires accurate covariance estimation and is sensitive to errors from multipath (i.e., echoes). Subspace-based methods, such as MUSIC [36, 93], offer excellent direction-of-arrival estimation, but are not well suited for signal recovery, which is the focus of our work. Additionally, these more sophisticated approaches are computationally expensive, have more signal assumptions, and often require a narrow frequency of interest (or be run many times to be more broadband), all of which limit their generalizability. These reasons made delay-sum a practical choice for a proof-of-concept implementation.

In a similar vein, we also explored deep learning sound source separation and neural beamforming approaches [66, 78, 96]. However, these techniques target an adjacent problem domain to ours, requiring an *acoustic class of interest*, for example, human speech. This is less suitable for our purpose, since we want to explicitly focus on *all* sounds being generated at the hand, irrespective of the acoustic class. Lastly, these methods also require pre-training on substantial amounts of domain-specific data, which our method does not. That said, we do employ deep learning for prediction in our individual example applications (Section 4).

3.7 Sound Pressure Level Map

In addition to the STFT of the audio stream, we also create a 2D sound pressure level (SPL) map (also sometimes called an "acoustic map") [15, 79]. This provides a useful spatial characterization of the source sound. If the sound is highly localized (for example, in the case of a finger swiping across a table and the finger itself is responsible for generating noise), the map will feature a tight blob (Figure 6A). If the user is drawing while holding a pen, the sound is generated at the pen tip (Figure 6B). And finally, if there is no sound coming from the user's hand, other distributed patterns are formed; examples of ambient background noise and loud music are provided in Figure 6C and 6D. To see more examples of our SPL map output, including when substantial external noise is present, please refer to our Video Figure.

To create the SPL map, we perform a small grid search around the finger point ±100 mm horizontally and vertically with an 8 mm step size, creating a 26×26 matrix of values. At each point in the grid, we beamform the audio signals, which results in a single synthesized waveform, from which we can compute the mean sound pressure (dB). We repeat this process for all points in the grid, producing a 2D map of sound pressure. To aid in visualization, we find the maximum observed value in the grid, and set this to be max and -3 dB of this value to be minimum, with any lower values clipped. Our pipeline produces these SPL maps at 10.8 FPS, and we run our machine learning at this framerate.

3.8 Machine Learning

As depicted in Figure 2, our model takes in two inputs: the STFT and SPL map (described above). Each modality is independently encoded by a dedicated convolutional branch, composed of two convolutional layers with ReLU activations and 2×2 max pooling. These encoders produce 64-channel feature maps per modality, which are then flattened and passed through separate classifier heads to help modality-specific auxiliary outputs. For combined modality classification, the flattened features from both encoders are also concatenated into a joint latent vector, which is fed into a two-layer fully connected classifier with ReLU activation and a dropout rate of 0.5. The model is trained using a weighted sum of this main classification loss and the auxiliary losses from each modality, with the auxiliary losses weighted by a factor of 0.3.

We adapt our model slightly depending on the application use case (described in the next section). Specifically, we use different length STFTs and tune loss functions. For example, ad hoc touch is formulated as a binary classification problem, and we use binary cross-entropy loss with logits. Alternatively, our held object activity recognition demo has multiple activity classes, and we use standard cross-entropy loss.

4 Example Uses

We believe SoundBubble's finger-attached virtual microphones can be used to power a wide variety of interesting applications. In this section, we highlight four diverse use cases previously explored in the HCI literature, but with physical microphones located close to the point of interaction. We believe these use cases can now be enabled with users wearing headsets and future XR glasses without using any accessory devices.

4.1 Ad Hoc Touch

Wearers of today's XR headsets and future XR glasses could leverage surfaces in the world for ad hoc touch input. For instance, virtual interfaces could snap onto physical surfaces when available, allowing users to interact with tactile feedback, which has been shown to offer ergonomic and accuracy benefits [11, 55]. Prior work has enabled this input modality using vibro-acoustic sensing methods, e.g., wrist or finger-worn microphones, VPUs, or IMUs [21, 41, 60, 82, 84]. As illustrated in Figure 8 and shown in the Video Figure, we can achieve a similar interaction without any instrumentation of the user's hands using SoundBubble. Both taps (wide-band impulse events) and swipes (which create high-frequency sounds) create detectable noises. We evaluate this use case in our User Study.

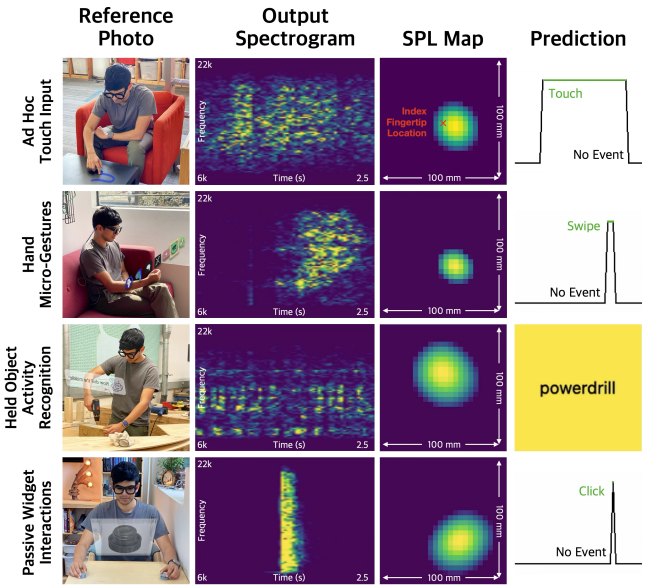

Figure 8: Four example applications are demonstrated. A beamformed audio spectrogram, SPL map, and prediction result were captured during each interaction for illustration.

4.2 Hand Micro-Gestures

Hand micro-gestures have been proposed for subtle and convenient input into XR systems [40, 53, 54, 57, 92, 95]. For example, Meta's neural wristband [62, 64] has demonstrated finger-to-finger gestures, such as directional swipes for navigation, for use with its upcoming Orion AR glasses. Taking inspiration from this work, we built a demonstration of micro-gesture input using acoustic information captured by SoundBubble. As already shown in AO-Finger [101], even subtle finger-to-finger translations create an almost white-noise-like signal, which SoundBubble easily isolates from the headset (whereas AO-Finger used a wrist-worn stethoscope). To resolve swipe direction, we rely on finger tracking provided by the Quest 3S. We did not evaluate this use case, as it was very similar to our other input condition, but it can be seen in Figure 8 and the Video Figure.

4.3 Passive Widget Interactions

We also explored using SoundBubble to detect interactions with passive, low-cost objects, potentially created through rapid prototyping. An exemplary prior work in this space is Lumello [80], which demonstrated entirely 3D printed objects integrated with 3D printed mechanisms (e.g., buttons, sliders) that produced sounds when actuated. These sounds were detected with a wired Piezo microphone affixed to the object. Although the object was entirely passive (just plastic), it could be made digitally interactive through this technique. Stane [72] is similar, but uses patterned surface features instead of moving mechanisms to capture user input. When a finger slides over these patterned regions, distinct acoustic signals are generated, which are captured by a contact microphone. A final example in the literature is Acoustic Barcodes [29], passive acrylic

tiles laser-etched with structured binary patterns. When a finger or object swipes over an Acoustic Barcode, the patterned ridges produce a series of impulses that encode digital information, which can be read and used to trigger interactive functionality. Similar to the aforementioned work, a contact microphone is affixed to the host surface.

We again take inspiration from this prior work and believe SoundBubble could enable similar use cases, but without the need to instrument objects or surfaces with microphones, and instead be self-contained in the user's headset. As one demonstration of this use case, we curated a set of six low-cost and passive mechanisms. We 3D printed three mechanisms: a shutter button [104], a turn knob [44], and a slider (using a tine mechanism inspired by Lumello [80]). To this set, we added three commercial off-the-shelf mechanisms: a dog training clicker, a metal dome button (p.n. F12450), and a momentary tactile switch (p.n. Panasonic EVQ11U). In a proper implementation, we would rely on computer vision from the headset to detect the held object (or e.g., scan a QR code) and load the appropriate widget interactors. SoundBubble would be used to detect the moment of actuation, which is challenging with computer vision alone (e.g., the travel on most buttons is just a few mm). We evaluate the efficacy of SoundBubble detecting passive widget interactions (Figure 9, middle row) in our User Study.

4.4 Held Object Activity Recognition

Context-sensitive computing systems have long been championed as a way to make interfaces more responsive to their users' needs and tasks [46, 48, 49, 68]. The most practical are worn systems that can operate anywhere their users go. Sensing approaches include wrist-worn accelerometers or microphones, sensing the vibrations or sounds emitted by held active objects [26, 46, 49]. Passive object use and various hand-centric actions can also be inferred by sound and motion, as shown in [48, 68].

As a final example use case for SoundBubble, we listen in on sounds produced by tools and other objects held and operated in the user's hand. We can either 1) utilize computer vision to detect the object class and then use acoustic information to infer when the user is operating the object, or 2) attempt to identify the held object based on its operation sounds (e.g., like ViBand [49]). As proof of concept, we built a working demonstration of the latter capability. We also evaluate the recognition accuracy of eight example handheld objects (Figure 9, bottom row) using SoundBubble in our User Study.

5 User Study

We selected three of our example applications to serve as tasks for an evaluation with users: Ad hoc touch interaction, passive widget interaction, and held object activity classification.

Of important note, in all three study tasks, we trained a model *before* the experiment using author data (40.5, 73.0, and 52.0 minutes worth of data for ad hoc touch, held object activity recognition, and passive widget interactions, respectively). The same hardware as the study was used, and the authors varied their hand and head pose naturally during tasks to add variety to the dataset. We note this training set is not particularly large or exhaustive, but it was sufficient to demonstrate compelling accuracies with unseen uses.

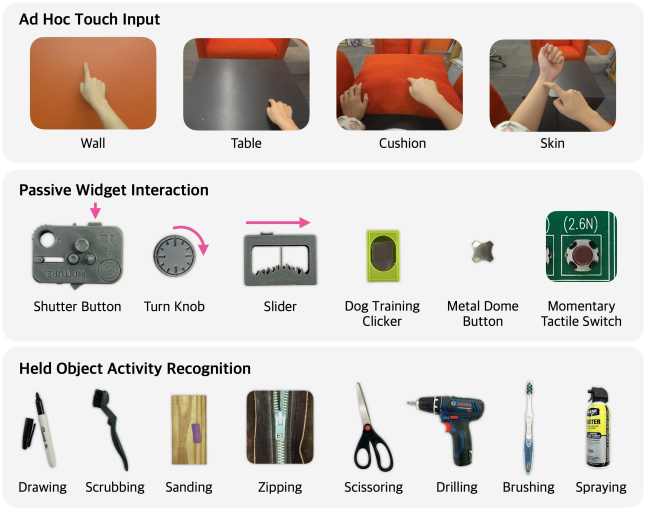

Figure 9: Our user study included three tasks: Ad hoc touch interaction (with four test surfaces), passive widget interaction (with six widgets), and held object activity recognition (with eight objects).

Crucially, training a model before the user study allowed us to assess *live classification accuracy*, without any post-hoc optimization of results. We believe that this method best conveys "out-of-the-box" accuracy that users would experience in practice. The only part of our study with post hoc analysis is an ablation study.

5.1 Apparatus

In order to run a post hoc ablation study — simulating different microphone numbers and arrangements, including glasses-like form factors — we chose to use our prototype Quest 3S fitted with a 16-channel UMA-16 v2 microphone array (Figure 5A). The headset displayed a basic study application that provided visual task prompts to participants to facilitate data collection. This interface was controlled by an experimenter using a study control interface running on a laptop. This laptop interface also allowed the experimenter to advance the study, which provided trial segmentation for later error analysis.

5.2 Participants

We recruited 10 participants (6 male, 4 female self-identified) for our one-hour-long study, which paid $20 in compensation. After completing consent forms, participants were given a brief orientation, and then fitted with the headset. Participants moved around during the study, and were seated or standing for different tasks. We describe task-specific procedures below.

5.3 Background Sound Conditions

The study was conducted in a typical open office with typical background noise (HVAC, occasional chatter among occupants, a coffee machine, typing noises, etc.). We periodically measured the volume of this ambient noise using an Extech 407730 sound level meter, and found a mean of 54.5 dB (SD=10.3).

For added challenge, we also include a noisy background condition. For this, we used a Logitech Z4i speaker system (two monitor speakers and a subwoofer) roughly 2 m away from our study area, playing a superimposition of a busy cafe recording [73] and a random song from Spotify's "Happy Hits" playlist. At 2 m, this raised the mean sound level to 69.1 dB at the array (SD=3.6).

As a third and final background sound condition, we had participants speak while performing our various study tasks. This user-generated sound is a useful complement to the previous sounds coming from the environment. Additionally, the proximity of the user's mouth to the microphone array meant the signal was loud — a mean sound level of 70.5 dB (SD=5.8) at the array.

Of critical note, the sound level of our target sounds is very small compared to these noisy backgrounds. For example, the sound of a finger running across a painted wall measured at the array (40 cm away) is just 49.2 dB — roughly 20 dB less than the background sound level (i.e., 100× quieter).

5.4 Study Task 1: Ad Hoc Touch Input

For our ad hoc touch input task, we selected four exemplary input surfaces: a wood table, a painted wall, a fabric armchair, and the user's forearm (Figure 9). For the painted wall, the participant was standing, and for the other three surface conditions, the user was seated on a sofa. On each input surface, users performed seven gestures, repeated three times each, in a random order: left swipe, right swipe, up swipe, down swipe, clockwise circle, counterclockwise circle, and figure-eight. This procedure was repeated three times for each input surface: once in the typical sound condition, once in the noisy sound condition, and once with the participant speaking. Thus, in total, our 10 participants generated 2520 input trials (4 input surfaces × 7 gestures × 3 repeats × 3 background sound conditions × 10 participants), totaling 148.3 minutes of data.

5.5 Study Task 2: Passive Widget Interactions

In this study task, we curated a small set of exemplary passive mechanical widgets (Figure 9). We 3D printed three mechanisms: a press button [104], a turn knob [44], and a slider (using a tine mechanism [37], inspired by [80]). To this widget set, we added three commercial off-the-shelf mechanisms: a dog training clicker, a metal dome button (p.n. F12450), and a momentary tactile switch (p.n. Panasonic EVQ11U). These widgets were placed on a table in front of participants and labeled with numbers for ease of understanding. Before starting the experiment, users were allowed to try each mechanism to get a feel for the task. In the study, the experiment interface requested each mechanism one at a time, in a random order, eight times each. As with the previous two study tasks, this procedure was run for typical, noisy, and user-speaking background sound conditions. Thus, in total, our 10 participants generated 1440 widget interaction trials (6 widgets × 8 repeats × 3 background sound conditions × 10 participants), totaling 87.5 minutes of data.

5.6 Study Task 3: Held Object Activity Recognition

In this study task, participants were asked to pick up and use 8 exemplary objects (Figure 9). We selected 5 handheld passive items with activities (drawing, scrubbing, sanding, zipping, cutting) and 3 tasks involving active items (drilling, tooth-brushing, and spraying). The experiment interface requested the user to pick up and operate each object three times each, in a random order. As with the previous study task, this procedure was run for typical, noisy, and user-speaking background sound conditions. Instead of binary activations, this model outputs an object classification (i.e., one of nine possible classes: eight object classes, plus a null/background class). In total, our 10 participants generated 720 object use trials (8 objects × 3 repeats × 3 background sound conditions × 10 participants), totaling 86.2 minutes of data.

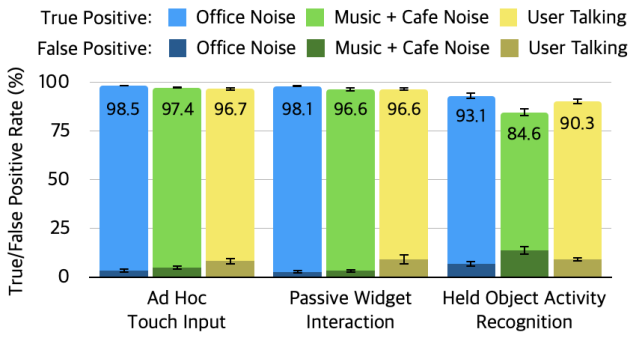

Figure 10: Per-task accuracy results broken out by background sound condition. Error bars are the standard error.

6 Results & Discussion

6.1 Accuracy Across Background Sound Conditions

Figure 10 provides an accuracy breakdown across background sound conditions. We can see that the noisy backgrounds — music + cafe noise (mean 69.1 dB) and user talking (mean 70.5 dB) — had a small, but consistent negative effect on true positive rate: -2.9%. As one might expect in a noisy environment, false positives increased (by 3.7%). A key finding is that this modest reduction in performance in adverse acoustic conditions is strong evidence of the efficacy of our beamforming approach and overall pipeline. We note that single-element systems (e.g., an omnidirectional microphone in a smartwatch) would fail in such conditions, where the signal is dominated by the much louder background noise (see Section 5.3 for more details).

6.2 Accuracy Across Tasks

Across our three study task categories, SoundBubble demonstrates a 94.7% true positive rate and a 6.8% false positive rate overall. Figure 10 provides per-task accuracies, further broken out by background sound condition.

6.2.1 Ad Hoc Touch Input Accuracy. Our ad hoc touch input task had a true positive rate of 97.5% and a false positive rate of 5.5%. The detection rates are consistently high across three different background sound conditions: 98.5% in the office, 97.4% in music and cafe noise, and 96.7% with the user talking. The false positive rate follows a similar trend: 3.5% in office noise, 4.9% in music + cafe

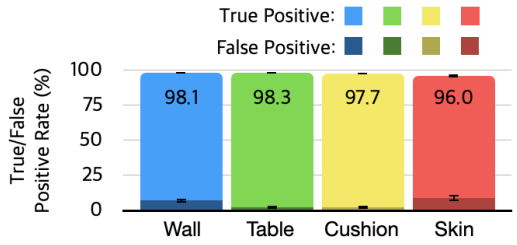

Figure 11: Ad hoc touch input results broken out by interacting surface. Error bars are standard error.

noise, and 8.2% in the user talking conditions. Figure 11 provides a breakdown by interacting surface. We see a true positive rate of 98.1% on the painted wall, 98.3% on the wood table, 97.7% on the fabric cushion, and 96.0% on the user's skin. The false positive rates were: wall 7.5%, table 2.6%, cushion 2.7%, and skin 9.3%. In the case of skin input, the increased false positives were generally caused by the model misdetecting single swipes as multiple swipes, which likely could be ameliorated with some filtering or hysteresis mechanism. Finally, we note that SoundBubble will not work on very smooth surfaces, like glass, which do not produce much sound when a finger translates across the surface.

6.2.2 Passive Widget Interaction Accuracy. In our passive widget interaction task, we see a 97.1% true positive rate and a 5.1% false positive rate (Figure 10). Across the background sound conditions, the true positive rates were 98.1%, 96.6%, 96.6% in office noise, music + cafe noises, user talking conditions, respectively. False positive rates were 2.8% in the office, 3.2% with music + cafe noise, and 9.2% with the user talking.

6.2.3 Held Object Activity Recognition Accuracy. In this task, if the model correctly detects and identifies the held object activity, this is considered a true positive event. If the model mis-triggers (no true event), triggers but misclassifies the held object activity, or outputs more than one class while an object is held, we consider these false positive events. With this evaluation criterion, we found an 89.4% true positive rate and 9.9% false positive rate. Broken out by background sound condition, true positive rates were 93.1%, 84.6%, 90.3%, and the false positive rates were 6.8%, 13.8%, and 9.3% in office noise, music + cafe noise, and user talking conditions, respectively. Figure 12 provides the confusion matrix result from the study.

6.2.4 Discussion of Task Results. Overall, we believe these task results convey a strong validation of SoundBubble as a potential sensing approach for hand-centric input in XR. Across our three study tasks and three background sound conditions, SoundBubble achieves a 94.7% true positive rate with a 6.8% false positive rate, even when target interaction sounds are up to 20 dB (100× in acoustic intensity) quieter than the background (Section 5.3). This provides strong evidence that vision-guided beamforming can reliably recover subtle hand-generated sounds that would be inaccessible to single-microphone approaches in realistic, noisy environments.

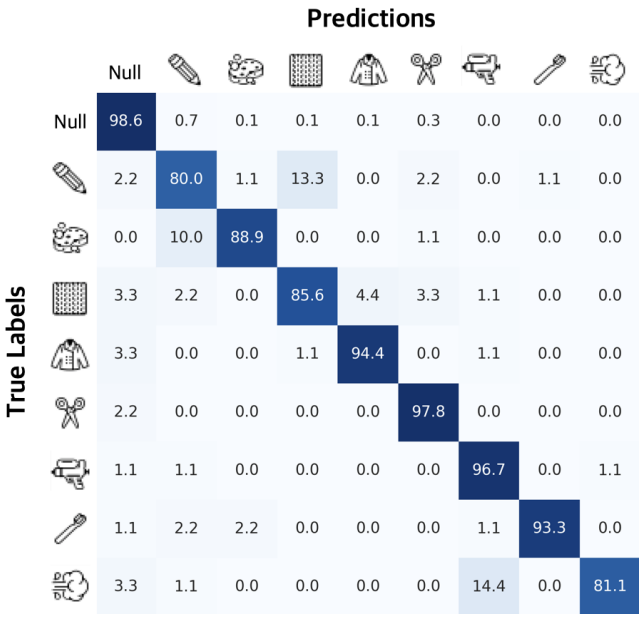

Figure 12: Results from our object recognition study task; confusion matrix for our nine test classes (Null, ✎: drawing, 🫧: scrubbing, ▦: sanding, 🧥: zipping, ✂: cutting, 🔫: drilling, 🖌: brushing, 🌫: spraying).

The task breakdown suggests where SoundBubble is most impactful. Ad hoc touch input and passive widget interactions both reach ~97% true positive rates, indicating that XR systems could leverage everyday surfaces and low-cost passive mechanisms without instrumenting the hands or objects. Held object activity recognition is more challenging (89.4% true positive), but still sufficient to support many context- and activity-aware experiences.

We note that false positives are higher in some conditions — notably for skin swipes and widget interactions when the user is talking — where the proximity of the mouth to the array can perturb the SPL map, and where a single extended gesture may be fragmented into multiple detections. These issues may be mitigated with temporal smoothing, hysteresis, or simple speech-aware gating. These issues point toward future work on improved temporal post-processing, better handling of concurrent speech, and refined spatial models for object use.

6.3 Microphone Geometry Ablation Study

Our study apparatus was purposefully over-provisioned with microphones so that we could run a post hoc ablation study, simulating different microphone array sizes and densities, but more importantly, different device form factors. We considered three general categories: 1) large microphone arrays not constrained to a headset (i.e., ideal but impractical); 2) arrays with microphones constrained to the physical bounds of a Quest 3S; and 3) microphones constrained to the smaller form of glasses-like devices. We also computed results for a baseline single-microphone device without beamforming, discussed in the next section. These microphone arrangements are illustrated in Figure 13.

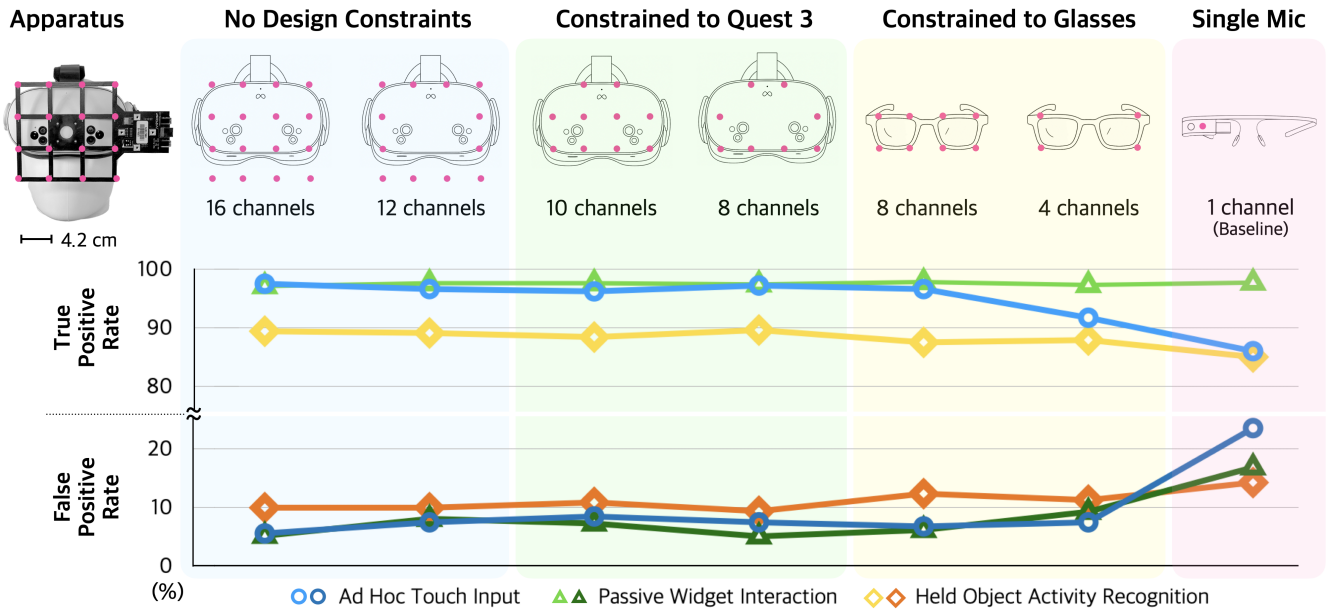

Figure 13: We used our study data to post hoc simulate the performance of different microphone arrays, varying geometry and element count. Our capture apparatus is shown on the far left. Our first two designs (blue background tint) have no design constraints — for instance, microphones are included that exist outside the enclosure of the Quest 3S. Next, we consider two designs where the microphones are constrained to the physical bounds of the Quest 3S headset (green background tint), which is more realistic. Finally, we consider a glasses form factor (yellow background tint), modeled on Meta's Orion AR glasses. We also computed results for a baseline, single-microphone device without beamforming (pink background tint). Below each arrangement, we report true positive and false positive rates.

To this end, we trained six additional models per task condition, each corresponding to a different microphone-channel configuration. These models were trained on the same dataset used for the pre-trained model in the user study (i.e., author data; see Section 5 for details), but with specific channels omitted during preprocessing. Each model was then evaluated on the data collected from our 10 user study participants. For the 1-channel baseline model, as there is no beamforming in this configuration, we keep the same network architecture for the spectrogram encoder and classifier, excluding only the components that process the SPL map. The devices and their corresponding accuracy results are shown in Figure 13.

The main insight is apparent — the smaller the array and the fewer microphones it contains, the weaker the accuracy. Accuracy is fairly flat until we reach our two 8-channel devices (true positive rate drops from 94.7% with our 16-channel device, to an average of 94.4% for our two 8-channel devices; false positives grow from 6.8% to 7.8%). At these high accuracies, the comparatively small differences do constitute significant increases in error. The first device where we see a substantial drop in accuracy is the 4-channel glasses (92.3% true positive rate, 9.3% false positive rate), where beamforming performance is significantly impacted, which has a cascading effect on accuracy. Interestingly, passive widget interaction has a fairly consistent true positive rate across all of our simulated devices. We believe this is due to the fact that impulsive "click" events are strong enough to manifest in the audio signal, irrespective of the benefits of beamforming.

6.4 Comparison to Single Microphone

Figure 13 also plots the results for a single microphone device, which offers a useful baseline with which to quantify the benefits of SoundBubble over more typical acoustic sensing with a single omnidirectional microphone. While a single microphone performs well on our passive widget interaction task (see hypothesis in the previous section), there are substantive drops in our ad hoc touch input and held object activity tasks. True positive rates drop by 10.6% and 2.5%, respectively for the two tasks, compared to the 8-channel glasses device. False positive rates grow by 16.8%, 10.7%, and 1.9% (compared to the 8-channel glasses device) for the ad hoc touch input, passive widget interaction, and held object activity recognition tasks, respectively.

7 Limitations & Future Work

First and foremost, while SoundBubble provides a significant boost in SNR, it by no means offers perfect target sound isolation. Like all acoustic systems, noisy environments will continue to be challenging, though SoundBubble may offer one avenue to improve robustness.

Our ablation study found increased accuracy with more microphones, but this is at odds with making compact glasses-like XR headsets, where every cubic millimeter and gram is precious. Fortunately, MEMS microphones now come in extremely small packages, so integration of microphone arrays is possible. Multiple consumer devices are already available with integrated microphone

arrays, such as the Meta Ray-Ban Display Glasses and Apple Vision Pro (see also discussion in 3.4). In this work, we only considered planar arrays, but microphones could also run down the arms of headsets/glasses, creating a three-dimensional array. This is the arrangement used in commercially-available devices such as the Meta Aria glasses, which feature a seven-microphone array. In general, the further the microphones can be moved apart, the better the performance of the array.

Staying on the topic of hardware, we note that power consumption would be another challenge. In XR glasses, batteries must be small. While MEMS microphones are very efficient (<1.0 mW), transmitting and processing high-frequency, multi-channel audio data may be expensive. While we run our prototype software on a laptop CPU, a commercial product would almost certainly rely on an efficient DSP chip, many of which offer beamforming functionality (e.g., Cadence Tensilica HiFi 3z DSP [90], Knowles IA8201 audio edge processor [43]). For instance, Apple's Vision Pro is advertised as having a "six-mic array with directional beamforming" [1], suggesting existing internal capabilities.

As noted earlier, more advanced acoustic beamforming techniques presented trade-offs that limited their suitability. Adaptive approaches such as MVDR/Capon [36, 93] require accurate covariance estimation and incur higher computational cost. Subspace-based methods like MUSIC [36, 93] excel at direction-of-arrival estimation, but are less appropriate for signal recovery. Both families of methods tend to assume narrowband operation or require repeated computation for broadband signals. Deep learning–based source separation and neural beamforming methods [66, 78, 96] were also not adopted, as they target class-based separation (e.g., human speech) rather than spatially constrained recovery and require large-scale pre-training. We reiterate our focus was not to advance beamforming itself, but to demonstrate a new use of it, and our approach is agnostic to beamformer choice so long as it supports our application needs. Delay-sum, although a classic method, was sufficient for our explorations.

Finally, we believe there are other interesting use cases to explore in future work. The ability to place virtual microphones into 3D scenes using a headset (and future glasses) could be coupled with other interaction modalities beyond finger tracking, including finger pointing and ray casting [39] and gaze tracking [59]. This could power, for instance, new forms of audio augmented reality, such as social-, environment-, and task-centric audio modes.

8 Conclusion

We introduced SoundBubble, a technique that virtually positions a finger-bound microphone via acoustic beamforming from an XR headset, enabling robust vibro-acoustic-based interactions. With SoundBubble, we showed that a single centralized device (i.e., an XR headset) can unify a wide range of user applications that previously required a disparate set of hardware form factors. We evaluated our system in three popular user applications and under three challenging noise conditions, including loud music, cafe noise, and user talking. The results showed that SoundBubble maintains consistently high accuracy compared to non-array microphone configurations used in prior work. In addition, we explored various microphone array geometries and showcased a diverse set of device form factors.

References

[1] 2025. Apple Vision Pro — Technical Specifications. https://www.apple.com/apple-vision-pro/specs/. Accessed: 2025-12-01.

[2] AV Access. 2024. AnyCo M1 Add-on Microphone with Echo Cancellation & Noise Suppression. https://www.amazon.com/AV-Access-AnyCo-Cancellation-Suppression/dp/B0DD3WS9FW.

[3] Karan Ahuja, Andy Kong, Mayank Goel, and Chris Harrison. 2020. Direction-of-Voice (DoV) Estimation for Intuitive Speech Interaction with Smart Devices Ecosystems. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 1121–1131. https://doi.org/10.1145/3379337.3415588

[4] Takashi Amesaka, Hiroki Watanabe, Masanori Sugimoto, and Buntarou Shizuki. 2022. Gesture Recognition Method Using Acoustic Sensing on Usual Garment. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 2, Article 41 (July 2022), 27 pages. https://doi.org/10.1145/3534579

[5] Md Tanvir Islam Aumi, Sidhant Gupta, Mayank Goel, Eric Larson, and Shwetak Patel. 2013. DopLink: using the doppler effect for multi-device interaction. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Zurich, Switzerland) *(UbiComp '13)*. Association for Computing Machinery, New York, NY, USA, 583–586. https://doi.org/10.1145/2493432.2493515

[6] Andreas Braun, Stefan Krepp, and Arjan Kuijper. 2015. Acoustic tracking of hand activities on surfaces. In *Proceedings of the 2nd International Workshop on Sensor-Based Activity Recognition and Interaction* (Rostock, Germany) *(iWOAR '15)*. Association for Computing Machinery, New York, NY, USA, Article 9, 5 pages. https://doi.org/10.1145/2790044.2790052

[7] Karl Ferdinand Braun. 1909. Electrical Oscillations and Wireless Telegraphy. Nobel Lecture, NobelPrize.org.

[8] Liwei Chan, Yi-Ling Chen, Chi-Hao Hsieh, Rong-Hao Liang, and Bing-Yu Chen. 2015. CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 549–556. https://doi.org/10.1145/2807442.2807450

[9] Ishan Chatterjee, Maruchi Kim, Vivek Jayaram, Shyamnath Gollakota, Ira Kemelmacher, Shwetak Patel, and Steven M. Seitz. 2022. ClearBuds: wireless binaural earbuds for learning-based speech enhancement. In *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services* (Portland, Oregon) *(MobiSys '22)*. Association for Computing Machinery, New York, NY, USA, 384–396. https://doi.org/10.1145/3498361.3538933

[10] Tuochao Chen, Malek Itani, Sefik Emre Eskimez, Takuya Yoshioka, and Shyamnath Gollakota. 2024. Hearable devices with sound bubbles. *Nature Electronics* 7, 11 (Nov. 2024), 1047–1058. https://doi.org/10.1038/s41928-024-01276-z

[11] Yi Fei Cheng, Tiffany Luong, Andreas Rene Fender, Paul Streli, and Christian Holz. 2022. ComforTable User Interfaces: Surfaces Reduce Input Error, Time, and Exertion for Tabletop and Mid-air User Interfaces. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 150–159. https://doi.org/10.1109/ISMAR55827.2022.00029

[12] ClearOne. 2025. Beamforming Microphone Arrays. https://www.clearone.com/products/audio-conferencing/microphones/beamforming-microphone-arrays.

[13] Artem Dementyev, Dimitri Kanevsky, Samuel Yang, Mathieu Parvaix, Chiong Lai, and Alex Olwal. 2023. LiveLocalizer: Augmenting Mobile Speech-to-Text with Microphone Arrays, Optimized Localization and Beamforming. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23 Adjunct)*. Association for Computing Machinery, New York, NY, USA, Article 75, 3 pages. https://doi.org/10.1145/3586182.3615789

[14] Digikey. 2025. *ReSpeaker Mic Array v2.0.* https://mm.digikey.com/Volume0/opasdata/d220001/medias/docus/862/107990053_Br.pdf

[15] Harvey Fletcher and W. A. Munson. 1933. Loudness, Its Definition, Measurement and Calculation. *Bell System Technical Journal* 12, 4 (1933), 377–430. https://doi.org/10.1002/j.1538-7305.1933.tb00403.x

[16] Masaaki Fukumoto and Yasuhito Suenaga. 1994. "FingeRing" a full-time wearable interface. In *Conference companion on Human factors in computing systems*. 81–82.

[17] Yang Gao, Yincheng Jin, Seokmin Choi, Jiyang Li, Junjie Pan, Lin Shu, Chi Zhou, and Zhanpeng Jin. 2022. SonicFace: Tracking Facial Expressions Using a Commodity Microphone Array. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 156 (Dec. 2022), 33 pages. https://doi.org/10.1145/3494988

[18] Yang Gao, Yincheng Jin, Jiyang Li, Seokmin Choi, and Zhanpeng Jin. 2020. EchoWhisper: Exploring an Acoustic-based Silent Speech Interface for Smartphone Users. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 3, Article 80 (Sept. 2020), 27 pages. https://doi.org/10.1145/3411830

[19] Mayank Goel, Brendan Lee, Md. Tanvir Islam Aumi, Shwetak Patel, Gaetano Borriello, Stacie Hibino, and Bo Begole. 2014. SurfaceLink: using inertial and acoustic sensing to enable multi-device interaction on a surface. In *Proceedings*

of the SIGCHI Conference on Human Factors in Computing Systems (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1387–1396. https://doi.org/10.1145/2556288.2557120

[20] Jun Gong, Aakar Gupta, and Hrvoje Benko. 2020. Acustico: Surface Tap Detection and Localization using Wrist-based Acoustic TDOA Sensing. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 406–419. https://doi.org/10.1145/3379337.3415901

[21] Yizheng Gu, Chun Yu, Zhipeng Li, Weiqi Li, Shuchang Xu, Xiaoying Wei, and Yuanchun Shi. 2019. Accurate and Low-Latency Sensing of Touch Contact on Any Surface with Finger-Worn IMU Sensor. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1059–1070. https://doi.org/10.1145/3332165.3347947

[22] Ru Guo, Yiru Yang, Johnson Kuang, Xue Bin, Dhruv Jain, Steven Goodman, Leah Findlater, and Jon Froehlich. 2020. HoloSound: Combining Speech and Sound Identification for Deaf or Hard of Hearing Users on a Head-mounted Display. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (Virtual Event, Greece) *(ASSETS '20)*. Association for Computing Machinery, New York, NY, USA, Article 71, 4 pages. https://doi.org/10.1145/3373625.3418031

[23] Sidhant Gupta, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. SoundWave: using the doppler effect to sense gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 1911–1914. https://doi.org/10.1145/2207676.2208331

[24] Shangchen Han, Po-chen Wu, Yubo Zhang, Beibei Liu, Linguang Zhang, Zheng Wang, Weiguang Si, Peizhao Zhang, Yujun Cai, Tomas Hodan, et al. 2022. UmeTrack: Unified multi-view end-to-end hand tracking for VR. In *SIGGRAPH Asia 2022 conference papers*. 1–9.

[25] Teng Han, David Ahlström, Xing-Dong Yang, Ahmad Byagowi, and Pourang Irani. 2016. Exploring Design Factors for Transforming Passive Vibration Signals into Smartwear Interactions. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction* (Gothenburg, Sweden) *(NordiCHI '16)*. Association for Computing Machinery, New York, NY, USA, Article 35, 10 pages. https://doi.org/10.1145/2971485.2971558

[26] Teng Han, Khalad Hasan, Keisuke Nakamura, Randy Gomez, and Pourang Irani. 2017. SoundCraft: Enabling Spatial Interactions on Smartwatches using Hand Generated Acoustics. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 579–591. https://doi.org/10.1145/3126594.3126612

[27] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 441–450. https://doi.org/10.1145/2047196.2047255

[28] Chris Harrison and Scott E. Hudson. 2008. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. Association for Computing Machinery, New York, NY, USA, 205–208. https://doi.org/10.1145/1449715.1449747

[29] Chris Harrison, Robert Xiao, and Scott Hudson. 2012. Acoustic barcodes: passive, durable and inexpensive notched identification tags. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 563–568. https://doi.org/10.1145/2380116.2380187

[30] Guilin Hu, Malek Itani, Tuochao Chen, and Shyamnath Gollakota. 2025. Proactive Hearing Assistants that Isolate Egocentric Conversations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 25366–25383. https://doi.org/10.18653/v1/2025.emnlp-main.1289

[31] Tianrui Hu, Taizhou Chen, and Kening Zhu. 2025. AirThumb: Supporting Mid-air Thumb Gestures with Built-in Sensors on Commodity Smartphones. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 66, 8 pages. https://doi.org/10.1145/3706599.3721219

[32] Yasha Iravantchi, Mayank Goel, and Chris Harrison. 2019. BeamBand: Hand Gesture Sensing with Ultrasonic Beamforming. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3290605.3300245

[33] Yasha Iravantchi, Yi Zhao, Kenrick Kin, and Alanson P. Sample. 2023. SAWSense: Using Surface Acoustic Waves for Surface-bound Event Recognition. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 422, 18 pages. https://doi.org/10.1145/3544548.3580991

[34] Hiroshi Ishii, Craig Wisneski, Julian Orbanes, Ben Chun, and Joe Paradiso. 1999. PingPongPlus: design of an athletic-tangible interface for computer-supported cooperative play. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) *(CHI '99)*. Association for Computing Machinery, New York, NY, USA, 394–401. https://doi.org/10.1145/302979.303115

[35] Dhruv Jain, Hung Ngo, Pratyush Patel, Steven Goodman, Leah Findlater, and Jon Froehlich. 2020. SoundWatch: Exploring Smartwatch-based Deep Learning Approaches to Support Sound Awareness for Deaf and Hard of Hearing Users. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (Virtual Event, Greece) *(ASSETS '20)*. Association for Computing Machinery, New York, NY, USA, Article 30, 13 pages. https://doi.org/10.1145/3373625.3416991

[36] Don H. Johnson and Dan E. Dudgeon. 1992. *Array Signal Processing: Concepts and Techniques*. Simon & Schuster, Inc., USA.

[37] kida. 2024. Musical Fidget. https://makerworld.com/en/models/762990-musical-fidget.

[38] David Kim, Otmar Hilliges, Shahram Izadi, Alex D. Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 167–176. https://doi.org/10.1145/2380116.2380139

[39] Daehwa Kim, Vimal Mollyn, and Chris Harrison. 2023. WorldPoint: Finger Pointing as a Rapid and Natural Trigger for In-the-Wild Mobile Interactions. *Proc. ACM Hum.-Comput. Interact.* 7, ISS, Article 442 (Nov. 2023), 19 pages. https://doi.org/10.1145/3626478

[40] Daehwa Kim, Keunwoo Park, and Geehyuk Lee. 2021. AtaTouch: Robust Finger Pinch Detection for a VR Controller Using RF Return Loss. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 11, 9 pages. https://doi.org/10.1145/3411764.3445442

[41] Daehwa Kim, Eric Whitmire, Roger Boldu, Wolf Kienzle, and Hrvoje Benko. 2024. SoundScroll: Robust Finger Slide Detection Using Friction Sound and Wrist-Worn Microphones. In *Proceedings of the 2024 ACM International Symposium on Wearable Computers* (Melbourne VIC, Australia) *(ISWC '24)*. Association for Computing Machinery, New York, NY, USA, 63–70. https://doi.org/10.1145/3675095.3676614

[42] Ellington Kirby, Seoyoon Park, Yan Wang, and Yingying Chen. 2016. HearHere: smartphone based audio localization using time difference of arrival: demo. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking* (New York City, New York) *(MobiCom '16)*. Association for Computing Machinery, New York, NY, USA, 509–510. https://doi.org/10.1145/2973750.2985625

[43] LLC Knowles Electronics. 2019. IA8201 Product Brief. https://www.knowles.com/docs/default-source/default-document-library/knowles-ia8201-product-brief-final9d761b731dff6ddbb37cff0000940c19.pdf?sfvrsn=4

[44] Kool Fingers. 2023. Fratchets. https://koolfingers.com/fratchets/.

[45] Ahmet Köse, Aleksei Tepljakov, and Sergei Astapov. 2017. Real-time localization and visualization of a sound source for virtual reality applications. In *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. 1–6. https://doi.org/10.23919/SOFTCOM.2017.8115577

[46] Gierad Laput, Karan Ahuja, Mayank Goel, and Chris Harrison. 2018. Ubicoustics: Plug-and-Play Acoustic Activity Recognition. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 213–224. https://doi.org/10.1145/3242587.3242609

[47] Gierad Laput, Xiang 'Anthony' Chen, and Chris Harrison. 2016. SweepSense: Ad Hoc Configuration Sensing Using Reflected Swept-Frequency Ultrasonics. In *Proceedings of the 21st International Conference on Intelligent User Interfaces* (Sonoma, California, USA) *(IUI '16)*. Association for Computing Machinery, New York, NY, USA, 332–335. https://doi.org/10.1145/2856767.2856812

[48] Gierad Laput and Chris Harrison. 2019. Sensing Fine-Grained Hand Activity with Smartwatches. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300568

[49] Gierad Laput, Robert Xiao, and Chris Harrison. 2016. ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 321–333. https://doi.org/10.1145/2984511.2984582

[50] Chi-Jung Lee, Ruidong Zhang, Devansh Agarwal, Tianhong Catherine Yu, Vipin Gunda, Oliver Lopez, James Kim, Sicheng Yin, Boao Dong, Ke Li, Mose Sakashita, Francois Guimbretiere, and Cheng Zhang. 2024. EchoWrist: Continuous Hand Pose Tracking and Hand-Object Interaction Recognition Using Low-Power Active Acoustic Sensing On a Wristband. In *Proceedings of the 2024 CHI Conference*

on Human Factors in Computing Systems (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 403, 21 pages. https://doi.org/10.1145/3613904.3642910

[51] Ke Li, Ruidong Zhang, Siyuan Chen, Boao Chen, Mose Sakashita, Francois Guimbretiere, and Cheng Zhang. 2024. EyeEcho: Continuous and Low-power Facial Expression Tracking on Glasses. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 319, 24 pages. https://doi.org/10.1145/3613904.3642613

[52] Ke Li, Ruidong Zhang, Bo Liang, François Guimbretière, and Cheng Zhang. 2022. EarIO: A Low-power Acoustic Sensing Earable for Continuously Tracking Detailed Facial Movements. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 2, Article 62 (July 2022), 24 pages. https://doi.org/10.1145/3534621

[53] Chen Liang, Chun Yu, Yue Qin, Yuntao Wang, and Yuanchun Shi. 2021. DualRing: Enabling Subtle and Expressive Hand Interaction with Dual IMU Rings. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 3, Article 115 (Sept. 2021), 27 pages. https://doi.org/10.1145/3478114

[54] Jaime Lien, Nicholas Gillian, M. Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: ubiquitous gesture sensing with millimeter wave radar. *ACM Trans. Graph.* 35, 4, Article 142 (July 2016), 19 pages. https://doi.org/10.1145/2897824.2925953

[55] Robert W. Lindeman, John L. Sibert, and James K. Hahn. 1999. Towards usable VR: an empirical study of user interfaces for immersive virtual environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) *(CHI '99)*. Association for Computing Machinery, New York, NY, USA, 64–71. https://doi.org/10.1145/302979.302995

[56] Guanhong Liu, Yizheng Gu, Yiwen Yin, Chun Yu, Yuntao Wang, Haipeng Mi, and Yuanchun Shi. 2020. Keep the Phone in Your Pocket: Enabling Smartphone Operation with an IMU Ring for Visually Impaired People. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 2, Article 58 (June 2020), 23 pages. https://doi.org/10.1145/3397308

[57] Christian Loclair, Sean Gustafson, and Patrick Baudisch. 2010. PinchWatch: a wearable device for one-handed microinteractions *(MobileHCI '10)*. Association for Computing Machinery, New York, NY, USA.

[58] Pedro Lopes, Ricardo Jota, and Joaquim A. Jorge. 2011. Augmenting touch interaction through acoustic sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces* (Kobe, Japan) *(ITS '11)*. Association for Computing Machinery, New York, NY, USA, 53–56. https://doi.org/10.1145/2076354.2076364

[59] Sven Mayer, Gierad Laput, and Chris Harrison. 2020. Enhancing Mobile Voice Assistants with WorldGaze. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3313831.3376479

[60] Manuel Meier, Paul Streli, Andreas Fender, and Christian Holz. 2021. TapID: Rapid Touch Interaction in Virtual Reality using Wearable Sensing. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 519–528. https://doi.org/10.1109/VR50410.2021.00076

[61] MEMSensing Microsystems Co. Ltd. 2015. *MSM261S4030H0 I²S Digital Output MEMS Microphone Data Sheet* (version 1.1 ed.). https://dl.sipeed.com Document No: DS-035.

[62] Meta. 2021. Wrist-based Interaction for the Next Computing Platform. https://tech.facebook.com/reality-labs/2021/3/inside-facebook-reality-labs-wrist-based-interaction-for-the-next-computing-platform/

[63] Meta. 2024. Orion Augmented Reality Glasses. https://about.fb.com/news/2024/09/introducing-orion-our-first-true-augmented-reality-glasses/

[64] Meta. 2025. Human-Computer Input via a Wrist-Based sEMG Wearable. https://www.meta.com/blog/surface-emg-wrist-white-paper-reality-labs/

[65] Meta Developers. 2025. *OVRHand Class (Unity API Reference)*. https://developers.meta.com/horizon/reference/unity/v78/class_o_v_r_hand Accessed: 2026-01-22.

[66] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. 2021. An Overview of Deep-Learning-Based Audio-Visual Speech Enhancement and Separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 1368–1396. https://doi.org/10.1109/TASLP.2021.3066303

[67] MiniDSP. 2025. MiniDSP UMA-16 v2 Microphone Array. https://www.minidsp.com/products/usb-audio-interface/uma-16-microphone-array.

[68] Vimal Mollyn, Karan Ahuja, Dhruv Verma, Chris Harrison, and Mayank Goel. 2022. SAMoSA: Sensing Activities with Motion and Subsampled Audio. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3, Article 132 (Sept. 2022), 19 pages. https://doi.org/10.1145/3550284

[69] Vimal Mollyn, Nathan DeVrio, and Chris Harrison. 2025. EclipseTouch: Touch Segmentation on Ad Hoc Surfaces using Worn Infrared Shadow Casting. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 195, 13 pages. https://doi.org/10.1145/3746059.3747743

[70] Vimal Mollyn and Chris Harrison. 2024. EgoTouch: On-Body Touch Input Using AR/VR Headset Cameras. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 69, 11 pages. https://doi.org/10.1145/3654777.3676455

[71] Pius Kavuma Basajjabaka Mugagga and Simon Winberg. 2015. Sound source localisation on Android smartphones: A first step to using smartphones as auditory sensors for training A.I systems with Big Data. In *AFRICON 2015*. 1–5. https://doi.org/10.1109/AFRCON.2015.7331970

[72] Roderick Murray-Smith, John Williamson, Stephen Hughes, Torben Quaade, and Steven Strachan. 2008. Rub the stane. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems* (Florence, Italy) *(CHI EA '08)*. Association for Computing Machinery, New York, NY, USA, 2355–2360. https://doi.org/10.1145/1358628.1358683

[73] myNoise. 2018. *Restaurant Ambience: 10 Hours of Busy Coffee Shop Background Noise*. https://youtu.be/h2zkV-l_TbY

[74] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. FingerIO: Using Active Sonar for Fine-Grained Finger Tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1515–1525. https://doi.org/10.1145/2858036.2858580

[75] Ju Young Oh, Jun Lee, Joong Ho Lee, and Ji Hyung Park. 2017. AnywhereTouch: Finger Tracking Method on Arbitrary Surface Using Nailed-Mounted IMU for Mobile HMD. In *HCI International 2017 – Posters' Extended Abstracts*, Constantine Stephanidis (Ed.). Springer International Publishing, Cham, 185–191.

[76] Dan Olsen. 2008. Interactive viscosity. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. Association for Computing Machinery, New York, NY, USA, 1–2. https://doi.org/10.1145/1449715.1449717

[77] Makoto Ono, Buntarou Shizuki, and Jiro Tanaka. 2013. Touch & activate: adding interactivity to existing objects using active acoustic sensing. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 31–40. https://doi.org/10.1145/2501988.2501989

[78] Hendrik Purwins, Bo Li, Tuomas Virtanen, Jan Schlüter, Shuo-Yiin Chang, and Tara Sainath. 2019. Deep Learning for Audio Signal Processing. *IEEE Journal of Selected Topics in Signal Processing* 13, 2 (2019), 206–219. https://doi.org/10.1109/JSTSP.2019.2908700

[79] John William Strutt Baron Rayleigh. 1896. *The Theory of Sound*. Vol. 2. Macmillan.

[80] Valkyrie Savage, Andrew Head, Björn Hartmann, Dan B. Goldman, Gautham Mysore, and Wilmot Li. 2015. Lamello: Passive Acoustic Sensing for Tangible Input Components. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1277–1280. https://doi.org/10.1145/2702123.2702207

[81] Vivian Shen, James Spann, and Chris Harrison. 2021. FarOut Touch: Extending the Range of ad hoc Touch Sensing with Depth Cameras. In *Proceedings of the 2021 ACM Symposium on Spatial User Interaction* (Virtual Event, USA) *(SUI '21)*. Association for Computing Machinery, New York, NY, USA, Article 5, 12 pages. https://doi.org/10.1145/3485279.3485281

[82] Xiyuan Shen, Chun Yu, Xutong Wang, Chen Liang, Haozhan Chen, and Yuanchun Shi. 2024. MouseRing: Always-available Touchpad Interaction with IMU Rings. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 412, 19 pages. https://doi.org/10.1145/3613904.3642225

[83] Yilei Shi, Haimo Zhang, Jiashuo Cao, and Suranga Nanayakkara. 2020. VersaTouch: A Versatile Plug-and-Play System that Enables Touch Interactions on Everyday Passive Surfaces. In *Proceedings of the Augmented Humans International Conference* (Kaiserslautern, Germany) *(AHs '20)*. Association for Computing Machinery, New York, NY, USA, Article 26, 12 pages. https://doi.org/10.1145/3384657.3384778

[84] Yilei Shi, Haimo Zhang, Kaixing Zhao, Jiashuo Cao, Mengmeng Sun, and Suranga Nanayakkara. 2020. Ready, Steady, Touch! Sensing Physical Contact with a Finger-Mounted IMU. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 2, Article 59 (June 2020), 25 pages. https://doi.org/10.1145/3397309

[85] Snap Inc. 2025. Spectacles. https://www.spectacles.com/

[86] Paul Streli, Jiaxi Jiang, Juliete Rossie, and Christian Holz. 2023. Structured Light Speckle: Joint Ego-Centric Depth Estimation and Low-Latency Contact Detection via Remote Vibrometry. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 26, 12 pages. https://doi.org/10.1145/3586183.3606749

[87] Paul Streli, Jiaxi Jiang, Juliete Rossie, and Christian Holz. 2023. Structured Light Speckle: Joint Ego-Centric Depth Estimation and Low-Latency Contact Detection via Remote Vibrometry. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 26, 12 pages. https://doi.org/10.1145/3586183.3606749

[88] Paul Streli, Mark Richardson, Fadi Botros, Shugao Ma, Robert Wang, and Christian Holz. 2024. TouchInsight: Uncertainty-aware Rapid Touch and Text Input for Mixed Reality from Egocentric Vision. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–16.

[89] Shunta Suzuki, Takashi Amesaka, Hiroki Watanabe, Buntarou Shizuki, and Yuta Sugiura. 2024. EarHover: Mid-Air Gesture Recognition for Hearables Using Sound Leakage Signals. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 129, 13 pages. https://doi.org/10.1145/3654777.3676367

[90] Cadence Design Systems. [n. d.]. Tensilica HiFi 3z DSP. https://www.cadence.com/en_US/home/tools/silicon-solutions/compute-ip/hifi-dsps/hifi-3z.html.

[91] Texas Instruments. 2022. *TLV320ADC5140 Low-Power, Quad-Channel, Digital Microphone Audio ADC With Enhanced Digital Microphone Support.* https://www.ti.com/lit/ds/symlink/tlv320adc5140.pdf Rev. C.

[92] Hsin-Ruey Tsai, Cheng-Yuan Wu, Lee-Ting Huang, and Yi-Ping Hung. 2016. ThumbRing: private interactions using one-handed thumb motion input on finger segments. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct* (Florence, Italy) *(MobileHCI '16)*. Association for Computing Machinery, New York, NY, USA, 791–798. https://doi.org/10.1145/2957265.2961859

[93] Harry L Van Trees. 2002. *Optimum Array Processing.* John Wiley & Sons.

[94] B.D. Van Veen and K.M. Buckley. 1988. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine* 5, 2 (1988), 4–24. https://doi.org/10.1109/53.665

[95] Anandghan Waghmare, Roger Boldu, Eric Whitmire, and Wolf Kienzle. 2023. OptiRing: Low-Resolution Optical Sensing for Subtle Thumb-to-Index Micro-Interactions. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction* (Sydney, NSW, Australia) *(SUI '23)*. Association for Computing Machinery, New York, NY, USA, Article 8, 13 pages. https://doi.org/10.1145/3607822.3614538

[96] DeLiang Wang and Jitong Chen. 2018. Supervised Speech Separation Based on Deep Learning: An Overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26, 10 (2018), 1702–1726. https://doi.org/10.1109/TASLP.2018.2842159

[97] Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. 2014. Toffee: enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services* (Toronto, ON, Canada) *(MobileHCI '14)*. Association for Computing Machinery, New York, NY, USA, 67–76. https://doi.org/10.1145/2628363.2628383

[98] Robert Xiao, Julia Schwarz, Nick Throm, Andrew D. Wilson, and Hrvoje Benko. 2018. MRTouch: Adding Touch Input to Head-Mounted Mixed Reality. *IEEE Transactions on Visualization and Computer Graphics* 24, 4 (2018), 1653–1660. https://doi.org/10.1109/TVCG.2018.2794222

[99] XMOS. 2021. *VocalFusion XVF3510 Voice Processor User Guide.* https://www.xmos.com/download/XVF3510-User-Guide(4.2).pdf

[100] XMOS Ltd. 2023. *Mic Array Library.* https://www.xmos.com/documentation/XM-014926-PC/pdf/mic_array_programming_guide.pdf

[101] Chenhan Xu, Bing Zhou, Gurunandan Krishnan, and Shree Nayar. 2023. AO-Finger: Hands-free Fine-grained Finger Gesture Recognition via Acoustic-Optic Sensor Fusing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 306, 14 pages. https://doi.org/10.1145/3544548.3581264

[102] Yong Xu, Mingjiang Yang, Yanxin Yan, and Jianfeng Chen. 2004. Wearable microphone array as user interface. In *Proceedings of the Fifth Conference on Australasian User Interface - Volume 28* (Dunedin, New Zealand) *(AUIC '04)*. Australian Computer Society, Inc., AUS, 123–126.

[103] Jackie (Junrui) Yang, Gaurab Banerjee, Vishesh Gupta, Monica S. Lam, and James A. Landay. 2020. Soundr: Head Position and Orientation Prediction Using a Microphone Array. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376427

[104] YEZAO. 2025. Twelve-in-One Fidget Toy Collection. https://makerworld.com/en/models/856029-twelve-in-one-fidget-toy-collection

[105] Tianhong Catherine Yu, Guilin Hu, Ruidong Zhang, Hyunchul Lim, Saif Mahmud, Chi-Jung Lee, Ke Li, Devansh Agarwal, Shuyang Nie, Jinseok Oh, François Guimbretière, and Cheng Zhang. 2024. Ring-a-Pose: A Ring for Continuous Hand Pose Tracking. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 4, Article 189 (Nov. 2024), 30 pages. https://doi.org/10.1145/3699741

[106] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Sumeet Jain, Yiming Pu, Sinan Hersek, Kent Lyons, Kenneth A. Cunefare, Omer T. Inan, and Gregory D. Abowd. 2017. SoundTrak: Continuous 3D Tracking of a Finger Using Active Acoustics. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 2, Article 30 (June 2017), 25 pages. https://doi.org/10.1145/3090095

[107] Shijia Zhang, Taiting Lu, Hao Zhou, Yilin Liu, Runze Liu, and Mahanth Gowda. 2023. I Am an Earphone and I Can Hear My User's Face: Facial Landmark Tracking Using Smart Earphones. *ACM Trans. Internet Things* 5, 1, Article 1 (Dec. 2023), 29 pages. https://doi.org/10.1145/3614438

[108] Yang Zhang, Gierad Laput, and Chris Harrison. 2018. Vibrosight: Long-Range Vibrometry for Smart Environment Sensing. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 225–236. https://doi.org/10.1145/3242587.3242608

[109] Yang Zhang, Sven Mayer, Jesse T. Gonzalez, and Chris Harrison. 2021. Vibrosight++: City-Scale Sensing Using Existing Retroreflective Signs and Markers. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 410, 14 pages. https://doi.org/10.1145/3411764.3445054