SoundWatch: Exploring Smartwatch-based Deep Learning Approaches to Support Sound Awareness for Deaf and Hard of Hearing Users

Dhruv Jain
Computer Science and Engineering, University of Washington, Seattle, WA, USA
djain@cs.uw.edu

Hung Ngo
Computer Science and Engineering, University of Washington, Seattle, WA, USA
hvn297@cs.uw.edu

Pratyush Patel
Computer Science and Engineering, University of Washington, Seattle, WA, USA
patelp1@cs.uw.edu

Steven Goodman
Human Centered Design and Engineering, University of Washington, Seattle, WA, USA
smgoodmn@uw.edu

Leah Findlater
Human Centered Design and Engineering, University of Washington, Seattle, WA, USA
leahkf@uw.edu

Jon Froehlich
Computer Science and Engineering, University of Washington, Seattle, WA, USA
jonf@cs.uw.edu

Figure 1: *SoundWatch* uses a deep-CNN based sound classifier to classify and provide feedback about environmental sounds on a smartwatch in *real-time*. Images show different use cases of the app and one of the four architectures we built (*watch+phone*).

ABSTRACT

Smartwatches have the potential to provide glanceable, always-available sound feedback to people who are deaf or hard of hearing. In this paper, we present a performance evaluation of four low-resource deep learning sound classification models: *MobileNet*, *Inception*, *ResNet-lite*, and *VGG-lite* across four device architectures: *watch-only, watch+phone, watch+phone+cloud,* and *watch+cloud*. While direct comparison with prior work is challenging, our results show that the best model, VGG-lite, performed similar to the state of the art for non-portable devices with an average accuracy of 81.2% (*SD*=5.8%) across 20 sound classes and 97.6% (*SD*=1.7%) across the three highest-priority sounds. For device architectures, we found that the *watch+phone* architecture provided the best balance between CPU, memory, network usage, and classification latency. Based on these experimental results, we built and conducted a qualitative lab evaluation of a smartwatch-based sound awareness app, called *SoundWatch* (Figure 1), with eight DHH participants. Qualitative findings show support for our sound awareness app but also uncover issues with misclassifications, latency, and privacy concerns. We close by offering design considerations for future wearable sound awareness technology.

ASSETS '20, October 26–28, 2020, Virtual Event, Greece

ACM ISBN 978-1-4503-7103-2/20/10…$15.00
https://doi.org/10.1145/3373625.3416991

CCS CONCEPTS

• **Human-centered computing** → Accessibility; Accessibility technologies.

KEYWORDS

Accessibility, Deaf, hard of hearing, sound awareness, smartwatch, wearable, deep learning, CNN, sound classification

ACM Reference Format:
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 *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '20), October 26–28, 2020, Virtual Event, Greece.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3373625.3416991

1 INTRODUCTION

Smartwatches have the potential to provide glanceable, always-available sound feedback to people who are deaf or hard of hearing (DHH) in multiple contexts [4, 8, 31]. A recent survey with 201 DHH participants [4] showed that, compared to smartphones and head-mounted displays, a smartwatch is the most preferred device for non-speech sound awareness. Reasons included improved privacy, social acceptability, and integrated support for both visual and haptic feedback. Most prior work in wearable sound awareness, however, has focused on smartphones [3, 32, 40], head-mounted displays [9, 13, 18], and custom wearable devices [22, 34] that provide limited information (*e.g.,* loudness) through a single modality (*e.g.,* vision). A few Wizard-of-Oz studies have explored using visual and vibrational feedback on smartwatches for sound awareness [8, 31, 32]; however, the evaluations of the prototypes were preliminary. One exception includes Goodman *et al.* [8], who conducted a Wizard-of-Oz evaluation of smartwatch-based designs, gathering user reactions in different audio contexts (a student lounge, a bus stop, and a cafe). However, this work was intentionally formative with no functioning implementations.

Furthermore, recent deep-learning research has investigated multi-class sound classification models, including for DHH users [21, 40]. For example, Jain *et al.* [21] used deep convolutional neural networks to classify sounds in the homes of DHH users, achieving an overall accuracy of 85.9%. While accurate, these cloud or laptop-based models utilize a high memory and processing power and are unsuitable for low-resource portable devices.

Building on the above research, in this paper we present two smartwatch-based studies. First, we quantitatively examine four state-of-the-art low-resource deep learning models for sound classifications: *MobileNet* [15], *Inception* [41], *ResNet-lite* [42], and a quantized version of *HomeSound* [21], which we call *VGG-lite*, across four device architectures: *watch-only, watch+phone, watch+phone+cloud,* and *watch+cloud.* These approaches were intentionally selected to examine tradeoffs in computational and network requirements, power efficiency, data privacy, and latency. While direct comparison to prior work is challenging, our experiments show that the best classification model (VGG-lite) performed similarly to the state of the art for non-portable devices while requiring substantially less memory (~1/3rd). We also observed a strict accuracy-latency trade-off: the most accurate model was also the slowest (*avg. accuracy*=81.2%, *SD*=5.8%; *avg. latency*=3397ms, *SD*=42ms). Finally, we found that the two phone-based architectures (*watch+phone* and *watch+phone+cloud*) outperformed the watch-centric designs (*watch-only, watch+cloud*) in terms of CPU, memory, battery usage, and end-to-end latency.

To complement these quantitative experiments, we built and conducted a qualitative lab evaluation of a smartwatch-based sound awareness app, called *SoundWatch* (Figure 1), with eight DHH participants. SoundWatch incorporates the best performing classification model from our system experiments (VGG-lite) and, for the purposes of evaluation, can be switched between all four device architectures. During the 90-min study session, participants used our prototype in three locations on a university campus (a home-like lounge, an office, and outdoors) and took part in a semi-structured interview about their experiences, their views regarding accuracy-latency tradeoffs and privacy, and ideas and concerns for future wearable sound awareness technology. We found that all participants generally appreciated SoundWatch across all three contexts, reaffirming past sound awareness work [4, 8]. However, misclassifications were concerning, especially outdoors due to background noise. For accuracy-latency tradeoffs, participants wanted minimum delay for urgent sounds (*e.g.,* car honk, fire alarms)—to take any required action—but maximum accuracy for non-urgent sounds (*e.g.,* speech, background noise) to not be unnecessarily disturbed. Finally, participants selected *watch+phone* as the most preferred architecture because of privacy concerns with the cloud, versatility (no Internet connection required), and speed (*watch+phone* classified faster than *watch* only).

In summary, our work contributes: (1) a comparison of four deep learning models for sound classification on mobile devices, including accuracy-latency results, (2) a new smartwatch-based sound identification system, called SoundWatch, with support for four different device architectures, and (3) qualitative insights from in-situ evaluation with eight DHH users, including reactions to our designs, architecture preferences, and ideas for future implementations.

2 RELATED WORK

We contextualize our work within sound awareness needs and tools of DHH people as well as prior sound classification research.

2.1 Sound Awareness Needs

Prior formative studies have investigated the sounds, audio characteristics, and feedback modalities desired by DHH users. In terms of sounds of interest, two large-scale surveys by Findlater *et al.* [4] and Bragg *et al.* [3] with 201 and 87 participants respectively showed that DHH people prefer urgent and safety-related sounds (*e.g.,* alarms, sirens) followed by appliance alerts (*e.g.,* microwave beep, kettle whistle) and sounds about the presence of people (*e.g.,* door knock, name calls). These preferences may be modulated by cultural factors: participants who prefer oral communication were more interested in some sounds (*e.g.,* phone ring, conversations) than those who prefer sign language [3, 4].

In addition to desired sounds, prior work has shown DHH users desire certain *characteristics* of sound (*e.g.,* identity, location, time of occurrence) more than others (*e.g.,* loudness, duration, pitch) [8, 27]. However, the utility of these characteristics may vary by physical location. For example, at home, awareness of a sound's identity and location may be sufficient [20, 21], but directional indicators may be more important when mobile [32]. Besides location, Findlater *et*

al. [4] showed that social context (*e.g.,* friends vs. strangers) could influence the use of the sound awareness tool and thus customization (*e.g.,* using sound filtering) is essential. Informed by this work, Goodman *et al.* [8] explored using smartwatch designs in different locations (*e.g.,* bus stop, coffee shop), including the sound filtering options (*e.g.,* using identity, direction, or loudness).

In terms of feedback modalities, several studies recommend combining visual and vibrational information for sound awareness [8, 31, 32]; a smartwatch can provide both. To help users consume sound feedback information, past work recommends using vibration to notify about sound occurrence and a visual display for showing additional information [3, 20]—which we also explore—although a recent study also showed promise in using vibration patterns (*tactons*) to convey richer feedback (*e.g.,* direction) [8]. In the same study, participants valued the role of smartwatch as a glanceable, private, and portable display that can be used in multiple contexts.

We build on the above studies by examining the use of working smartwatch prototypes in three contexts, revealing qualitative reactions, system design suggestions, and location-based customization options.

2.2 Sound Awareness Technologies

Early research in sound awareness studied vibrotactile wrist-worn solutions, mainly to aid speech therapy by conveying voice tone [45] or frequency [44]; that work is complementary to our non-speech sound awareness. Researchers have also tried methods to completely substitute hearing with tactile sensation using more larger, more obtrusive form factors such as waist-mounted [36] or neck-worn [7], but this has shown little promise.

More recent work has examined stationary displays for sound awareness [14, 27, 28, 43], such as on desktops [14, 27]. Though useful for their specific applications, these solutions are not conducive to multiple contexts. Towards portable solutions, Bragg *et al.* [6] and Sicong *et al.* [31] used smartphones to recognize and display sound identity (*e.g.,* phone ringing, sirens). However, they evaluated their app in a single context (office [3], a deaf school [40]) and focused on user interface rather than system performance—both are critical to user experience, especially given the constraints of low-resource devices [11, 23].

Besides smartphones, wearable solutions such as head-mounted displays [9, 13, 18] and wrist-worn devices [22, 34] have been examined. For example, Gorman [9] and Kaneko *et al.* [22] displayed the direction of sound sources using a head-mounted display and a custom wrist-worn device, respectively. We explore smartwatches to provide sound identity, the most desired sound property by DHH users [3, 19, 27]. While not specifically focused on smartwatches, Jain *et al.* [21] examined smartwatches as complementary alerting devices to smarthome displays deployed that sensed and processed sound information locally and broadcasted it to the smartwatches; we examine a self-contained smartwatch solution for multiple contexts.

In summary, while prior work has explored sound awareness tools for DHH people, including on portable devices [10, 18, 22, 34], this work has not yet built and evaluated a working smartwatch-based system—a gap which we address in our work.

2.3 Sound Classification Research

Early efforts in classifying sounds relied on hand-crafted features such as zero-crossing rate, frame power, and pitch [33, 37, 38]. Though they performed reasonably well on clean sound files with a small number of classes, these features fail to account for acoustic variations in the field (*e.g.,* background noise) [26]. More recently, machine learning based classification has shown promise for specific field tasks such as gunshot detection [5] or intruder alert systems [2]. Specifically for DHH users, Bragg *et al.* [3] explored a preliminary GMM-based sound detection algorithm to classify two sounds (alarm clock, door knock) in an office setting. For more broad use cases, deep learning-based solutions have been investigated [21, 40]. For example, Sicong *et al.* [40] explored a lightweight CNN-based architecture on smartphones to classify nine sounds preferred by DHH users (*e.g.,* fire alarm, doorbell) in a school setting. Jain *et al.* [21] used deep convolutional neural networks running on a tablet to classify sounds in the homes of DHH users, achieving an overall accuracy of 85.9%. We closely follow the latter approach in our work but use embedded devices (phone, watch) and perform evaluations in varying contexts (home, work, outdoors). We also train and evaluate four low-resource deep learning models and possible watch-based architectures, as well as collect user preferences.

3 THE SOUNDWATCH SYSTEM

SoundWatch is an Android-based app designed for commercially available smartwatches to provide glanceable, always-available, and private sound feedback in multiple contexts. Building from previous work [8, 21], SoundWatch informs users about three key sound properties: *sound identity*, *loudness*, and *time of occurrence* through customizable sound alerts using visual and vibrational feedback (Figures 1 and 3). We use a deep learning-based sound classification engine (running on either the watch or on the paired phone or cloud) to continually sense and process sound events in real-time. Below, we describe our sound classification engine, our privacy-preserving sound sensing pipeline, system architectures, and implementation. The SoundWatch system is open sourced on GitHub: https://github.com/makeabilitylab/SoundWatch.

3.1 Sound Classification Engine

To create a robust, real-time sound classification engine, we followed an approach similar to *HomeSound* [21], which uses transfer learning to adapt a deep CNN-based image classification model (VGG) for sound classification. We downloaded four recently released (in Jan 2020 [46]) TensorFlow-based image-classification models for small devices: *MobileNet*, 3.4MB [15], *Inception*, 41MB [41], *ResNet-lite*, 178.3MB [42], and a quantized version of model used in HomeSound [21], which we call *VGG-lite*, 281.8MB. Since the size of four models differ considerably, we hypothesized that they would offer different tradeoffs in terms of accuracy and latency.

To perform transfer learning, similar to Jain *et al.* [21], we used a large corpus of sound effect libraries—each of which provide a collection of high-quality, pre-labeled sounds. We downloaded 20 common sounds preferred by DHH people (*e.g.,* dog bark, door knock, speech) [3, 20] from six libraries—BBC [47], Freesound [6], Network Sound [48], UPC [49], TUT [30] and TAU [1]. All sound

Table 1: The sounds and categories used to train our sound classification models

All sounds (*N*=20)	Fire/smoke alarm, Alarm clock, Door knock, Doorbell, Door-in-use, Microwave, Washer/dryer, Phone ringing, Speech, Laughing, Dog bark, Cat meow, Baby crying, Vehicle running, Car horn, Siren, Bird chirp, Water running, Hammering, Drilling
High priority (*N*=3)	Fire/smoke alarm, Alarm clock, Door knock
Medium priority (*N*=10)	Fire/smoke alarm, Alarm clock, Door knock, Doorbell, Microwave, Washer/dryer, Phone ringing, Car horn, Siren, Water running
Home context (*N*=11)	Fire/smoke alarm, Alarm clock, Door knock, Doorbell, Door-in-use, Microwave, Washer/dryer, Speech, Dog bark, Cat meow, Baby crying
Office context (*N*=6)	Fire/smoke alarm, Door knock, Door-in-use, Phone ringing, Speech, Laughing
Outdoor context (*N*=9)	Dog bark, Cat meow, Vehicle running, Car horn, Siren, Bird chirp, Water running, Hammering, Drilling

clips were converted to a single format (16KHz, 16-bit, mono) and silences greater than one second were removed, which resulted in 35.6 hours of recordings. We then divided the sounds into three categories based on prior work [3, 27]: high priority (containing the 3 most desired sounds by DHH people), medium-priority sounds (10 sounds), and all sounds (20 sounds) (see Table 1). We used the method in Hershey *et al.* [12] to compute the *log mel-spectrogram* features in each category, which were then fed to the four models, generating three models of each architecture (12 in total).

3.2 Sound Sensing Pipeline

For always-listening apps, privacy is a key concern. While SoundWatch relies on a live microphone, we designed our sensing pipeline to protect user privacy. The system processes the sound locally on the watch or phone and, in the case of the cloud-based architectures, only uploads non-reconstructable mel-spectrogram features. While the uploaded features can be used to identify the kind of activity a user is engaged in (*e.g.,* speaking, cooking), conversational information is not retrievable. For signal processing, we take a sliding window approach: the watch samples the microphone at 16KHz and segments data into 1-second buffers (16,000 samples), which are fed to the sound classification engine. To extract loudness, we compute the average amplitude in the window. All sounds at or above 50% confidence and 45dB loudness are notified to the user, the others are ignored.

3.3 System Architectures

We implemented four device architectures for SoundWatch: *watch-only*, *watch+phone*, *watch+cloud*, and *watch+phone+cloud* (Figure 2). Because the sound classification engine (computing features and predicting sound) is resource intensive, the latter three architectures use a more powerful device (phone or cloud) for classification. For only the cloud-based architectures, to protect user privacy, non-reconstructable sound features are computed before being sent to the cloud—that is, on the watch (*watch+cloud*) or on the phone (*watch+phone+cloud*). We use Bluetooth Low Energy (BLE) for watch-phone communication and WiFi or a cellular network for watch-cloud or phone-cloud communication.

3.4 User Interface

To increase glanceability, we designed the SoundWatch app as a push notification; when a classified sound event occurs, the watch displays a notification along with a vibration alert. The display includes sound identity, classification confidence, loudness, and

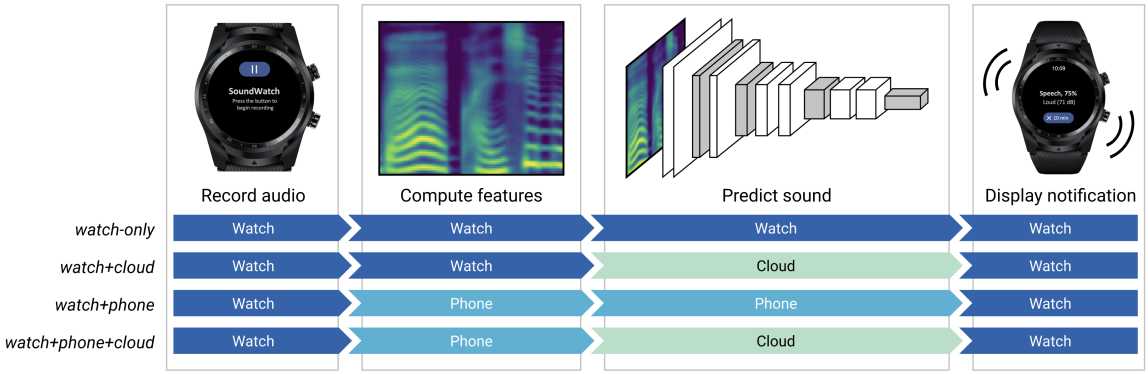

Figure 2: A diagram of the four SoundWatch architectures and a breakdown of their sensing pipelines. Block widths are for illustration only and are not indicative of actual computation time.

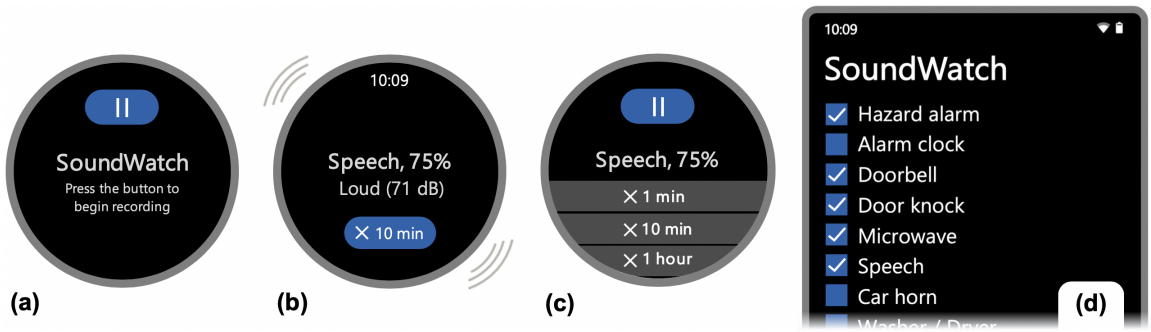

Figure 3: The SoundWatch user interface showing the opening screen with a button to begin recording the audio for classification (a), the notification screen with a "10-min" mute button (b), and the main app screen with more mute options (c). (d) shows a partial view of the paired phone app to customize the list of enabled sounds.

time of occurrence (Figure 3). Importantly, each user can mute an alerted sound by clicking on the "10-min" mute button, or by clicking on the "open" button and selecting from a scroll list of mute options (1 min, 5 min, 10 min, 1 hour, 1 day, or forever). Additionally, the user can select which sounds to receive alerts for by using the paired phone app, which displays a customization menu (Figure 3d). While future versions should run as an always-available service in Android, currently the app must be explicitly opened on the watch to run (Figure 3a). Once the app is opened, it continuously runs in the background.

4 SYSTEM EVALUATION

To assess the performance of our SoundWatch system, we perform two sets of evaluations: (1) a comparison of the four state-of-the-art sound classification models for embedded devices and (2) a comparison of the four architectures: *watch-only*, *watch+phone*, *watch+cloud*, and *watch+phone+cloud*. For all experiments, we used the *Android Ticwatch Pro* watch (4×1.2GHz, 1GB RAM) [50] and the *Honor 7x* Android phone (8×2GHz, 3GB RAM) [51]. For emulating the cloud, we used an Intel i7 desktop running Windows 10.

4.1 Model Comparison

To determine how different models perform on the watch and phone, we trained and evaluated the classification accuracy and speed of our four model architectures. To compare with prior approaches in sound classification, we also evaluated the full-VGG model (281.8MB) on a non-portable device. Below we detail the experiments and results.

4.1.1 Accuracy. To calculate the "*in-the-wild*" inference accuracy [52] of the models, we collected our own 'naturalistic' sound dataset similar to *HomeSound* [21]. We recorded 20 sound classes from nine locations (three homes, three offices, three outdoors) using the same hardware as SoundWatch: the TicWatch Pro with a built-in microphone. For each sound class, we recorded three 10-second samples at three distances (5, 10, and 15 feet). We attempted to produce sounds naturally (*e.g.,* using a microwave or opening the door). For certain difficult-to-produce sounds—like a fire alarm—we played snippets of predefined videos on a laptop or phone with external speakers (54 total videos were used). In total, we collected 540 recordings (~1.5 hours).

Before testing our model, we divided our recordings into the three categories (all sounds, high priority, medium priority) similar to our training set (Table 1). For the medium and high priority testsets, 20% of the sound data that we added was from excluded categories that our models should ignore (called the "unknown" class). For example, 20% of the high priority testset included recordings from outside of the three high priority sound classes (fire/smoke alarm, alarm clock, door knock).

For this experiment, we classified data in each category using the models. The results are shown in Figure 4. Overall, VGG-lite performed best (*avg. inference accuracy*=81.2%, *SD*=5.8%) followed by ResNet-lite (65.1%, *SD*=10.7%), Inception (38.3%, *SD*=17.1%) and MobileNet (26.5%, *SD*=12.3%); a *post hoc* one-way repeated measures ANOVA on all sounds yielded a significant effect of models on the accuracy ($F_{3,2156} = 683.9$, $p < .001$). As expected, the inference accuracy increased as the number sounds decreased from all (20 sounds) to medium (10 sounds) and high priority (3 sounds). For example, if we only classify the three highest-priority sounds, our average accuracies increase from 81.2% (*SD*=5.8%) to 97.6% (*SD*=1.7%) for VGG-lite and from 65.1% (*SD*=10.7%) to 78.1% (*SD*=11.9%) for ResNet-lite. Finally, in analyzing performance as a function of location context, home and office outperformed outdoors for all models. With VGG-lite, for example, average accuracies were 88.6% (*SD*=3.1%) for *home*, 86.4% (SD=4.3%) for *office*, and 71.2% (*SD*=8.2%) for *outdoors*. A *post hoc* inspection revealed that outdoor sound recordings may have incurred interference due to the background noise.

To further assess model performance, we computed a confusion matrix for medium-priority sounds, which helps highlight inter-class errors (Figure 5). While per-class accuracies varied across models, *microwave*, *door knock*, and *washer/dryer* were consistently the best performing classes with VGG-lite achieving average accuracies of 100% (*SD*=0), 100% (*SD*=0), and 96.3% (*SD*=2.3%) respectively. The worst performing classes were more model dependent but generally included *alarm clock*, *phone ring*, and *siren* with VGG-lite achieving 77.8% (*SD*=8.2%), 81.5% (*SD*=4.4%), and 88.9% (*SD*=3.8%) respectively. For these poorer performing classes, understandable mix-ups occurred—for example, alarm clocks and phones rings, which are similar sounding, were commonly confused.

	MobileNet	Inception	ResNet-lite	VGG-lite
All sounds	26.5 (12.3)	38.3 (17.1)	65.1 (10.7)	81.2 (5.8)
Med. Priority	41.8 (11.6)	59.0 (20.8)	78.1 (11.9)	89.6 (8.7)
High Priority	63.0 (8.5)	82.9 (7.6)	91.1 (3.4)	97.6 (1.7)
Home	30.5 (6.4)	41.7 (13.9)	71.3 (6.4)	88.6 (3.1)
Office	31.2 (8.9)	43.9 (11.7)	70.5 (6.4)	86.4 (4.3)
Outdoors	20.1 (14.7)	32.0 (23.2)	56.3 (15.3)	71.2 (8.2)

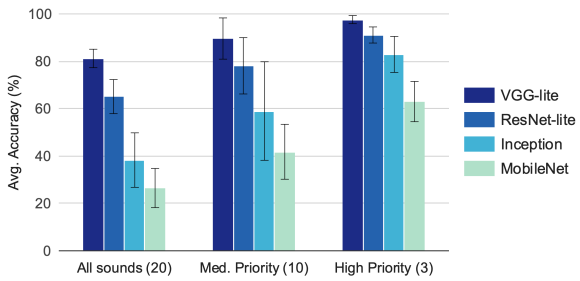

Figure 4: Average accuracy (and SD) of the four models for three sound categories and three contexts. Error bars in the graph show 95% confidence intervals.

4.1.2 Latency. In addition to accuracy, the *speed* with which a model performs classifications is crucial to achieving a real-time sound identification system. To evaluate model latency, we measured the time required to classify sounds from the input features on both the watch and the phone. We wrote a script to loop through the sound recordings in our dataset for three hours (1080 sounds) and measured the time taken for each classification. Understandably, the latency increased with the model size: the smallest model, MobileNet, performed the fastest on both devices (*avg. latency* on watch: 256ms, *SD*=17ms; phone: 52ms, *SD*=8ms), followed by Inception (*avg. latency* on watch: 466ms, *SD*=15ms; phone: 94ms, *SD*=4ms), and ResNet-lite (*avg. latency* on watch: 1615ms, *SD*=30ms; phone: 292ms, *SD*=13ms). VGG-lite, the largest model, was the slowest (*avg. latency* on watch: 3397ms, *SD*=42ms; phone: 610ms, *SD*=15ms).

In summary, for phone and watch models, we observed a strict accuracy-latency tradeoff—for example, the most accurate model VGG-lite (*avg. accuracy*=81.2%, *SD*=5.8%) was the slowest (avg. latency on watch: 3397ms, *SD*=42ms). Further, the models MobileNet and Inception performed too poorly for practical use (avg. accuracy < 40%). ResNet-lite was in the middle (*avg. accuracy*=65.1%, *SD*=10.7%; avg. latency on watch: 1615ms, *SD*=30ms).

4.1.3 Cloud model (VGG-16). To attempt comparison with past work, we also evaluated the performance of the full VGG model [25] on the cloud. On average, the inference accuracy (84.4%, *SD*=5.5%) was only slightly better than our best mobile-optimized model (VGG-lite, *avg.*=81.2%, *SD*=5.8%). This result is promising because our VGG-lite model is more than three times smaller than VGG (281.8MB *vs.* 845.5MB). However, the full model on the cloud performed much faster (*avg. latency*=80ms, *SD*=5ms) than our models on phone or watch.

4.2 Architecture Evaluation

Besides model evaluation, we also compared the performance of four different architecture designs for the SoundWatch system: *watch-only*, *watch+phone*, *watch+cloud*, and *watch+phone+cloud*. These architectures differ in terms of where classification computations occur, battery usage, classification speed, network requirements, and privacy—which impacts both technical performance and usability.

For our architecture evaluation, we used the most accurate model on the watch and phone: VGG-lite; the cloud used the full VGG model. Informed by prior work [11, 23, 29], we measured CPU, memory, and network usage, end-to-end latency, and battery consumption. For the test, we used a script running on a laptop that looped through the sound recordings for three hours to generate sufficient sound samples (1080). For the battery experiment only, the script ran until the watch battery reached 30% or less (*i.e.,* just

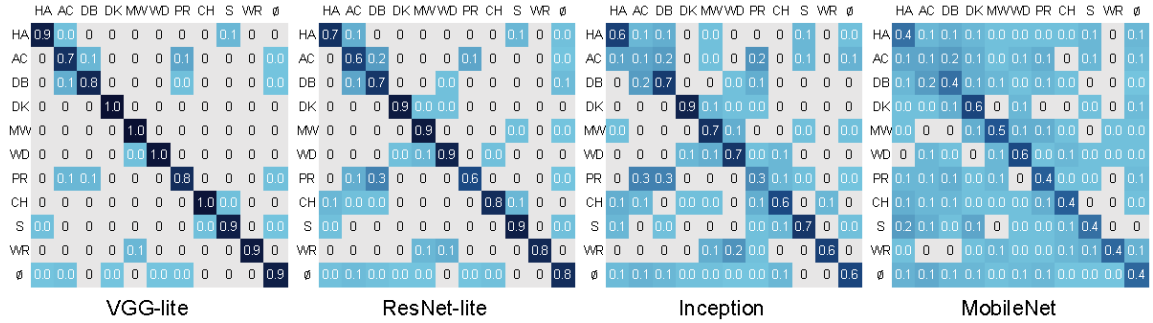

Figure 5: Confusion matrices for the four models when classifying 10 sounds in the medium-priority category. Darker blue indicates higher accuracy. *HA*=Hazard Alarm, *AC*=Alarm Clock, *DB*=Doorbell, *DK*=Door Knock, *MW*=Microwave, *WD*=Washer/Dryer, *PR*=Phone Ringing, *CH*=Car Horn, S=*Siren*, *WR*=Water Running, *Ø*=Unknown.

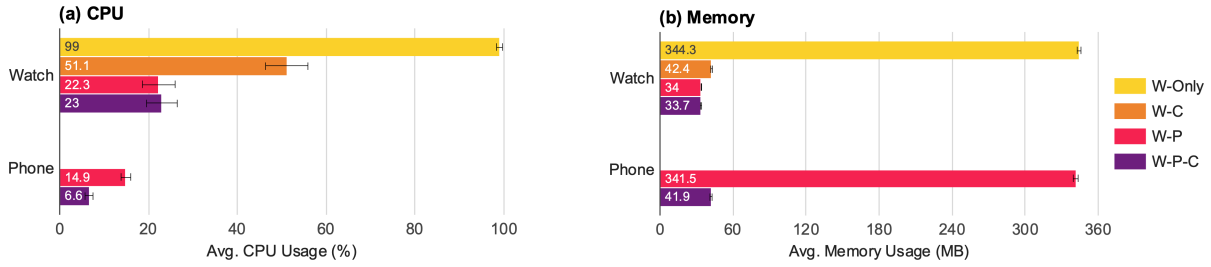

Figure 6: Average CPU (a) and memory (b) usage of the four architectures (using the VGG-lite model). Error bars show 95% confidence intervals.

above the 25% trigger for low-power mode), a common evaluation approach (*e.g.,* see [29]).

To determine CPU, memory, and network usage, we used *Android Profiler* [53], a commonly used tool in the literature [16]. For the battery, we used *Battery Historian* [54]. Finally, to determine end-to-end latency, we measured the elapsed time (in milliseconds) between the start of the sound recording window to when the notification is shown. Below, we detail the results.

4.2.1 CPU Utilization. Minimizing CPU utilization is crucial to maximizing the smartwatch's battery performance and lowering the impact on other running apps. Our results for CPU usage on the watch and phone are shown in Figure 6a. As expected, the watch's CPU utilization was lowest when classifications were performed on the phone (*watch+phone*; *avg.*=22.3%, *SD*=11.5%, *max*=42.3%) or in the cloud (*watch+phone+cloud*; *avg.*=23.0%, *SD*=10.8%, *max*=39.8%). Here, the watch is used only for *recording* sounds, *transmitting* data via Bluetooth, and *displaying* sound feedback. For *watch+cloud*, the watch is computing the sound features and communicating directly with the cloud via WiFi for classification, which resulted in significantly higher CPU utilization (*avg.*=51.1%, *SD*=14.9%, *max*=76.1%). Finally, if the entire classification model runs on the watch directly, the CPU utilization is nearly maxed out (*avg.*=99.0%, *SD*=2.1%, *max*=100%) and is thus not practical for real-world use.

4.2.2 Memory usage. A smartwatch app must also be memory efficient. We found that the memory usage was primarily dependent on where the model (281.8MB) was running, hence, *watch-only* and *watch+phone* consumed the highest memory on the watch (*avg.*=344.3MB, *SD*=2.3MB, *max*=346.1MB) and phone (*avg.*=341.5MB, *SD*=3.0MB, *max*=344.1MB) respectively (Figure 6b). This indicates that running a large model like VGG-lite on the watch could exceed the memory capacity of some modern watches (*e.g.,* [55]). The other app processes (*e.g.,* UI, computing features, network) required less than 50MB of memory.

4.2.3 Network usage. Having a low network requirement increases the portability of an app, especially for low-signal areas. Additionally, some users may feel uncomfortable with their data being uploaded to the cloud, even with privacy and security measures in place. For our cloud-based architectures, we found minimal network consumption: for *watch+cloud,* the average was 486.8B/s (*SD*=0.5B/s, *max*=487.6B/s) and for *watch+phone+cloud,* it was 486.5B/s (*SD*=0.5B/s, *max*=487.2B/s); both are negligible compared to the network bandwidth of modern IoT devices. The non-cloud architectures used no network bandwidth as they perform all classifications locally on the device(s): either the *watch* or *watch+phone*.

4.2.4 Battery consumption. A fully mobile app needs to be energy efficient. We measured the battery drain from full charge until 30% (Figure 7). First considering the watch-based architectures, the *watch-only* architecture used a large amount of battery: 30% at 3.3 hours, a 6.3x increase over the baseline (without our app). Within the remaining three architectures, both *watch+phone* (30% at 15.2 hours, 1.4x over baseline) and *watch+phone+cloud* (30% at 16.1 hours, 1.3x over baseline) were more efficient than *watch+cloud* (30% at 12.5 hours, 1.7x over baseline), because the latter used WiFi which is less energy efficient than BLE [39].

Similar trends were observed on the phone; however, running the model on the phone (*watch+phone*) was still tolerable (1.3x over baseline) as compared to the watch (6.3x over baseline). In summary, we expect that the watch-only architecture would be impractical for daily use, while the other architectures are usable.

4.2.5 End-to-end latency. Finally, a real-time sound awareness feedback system needs to be performant. Figure 8 shows a computational breakdown of end-to-end latency, that is, the total time spent in obtaining a notification for a produced sound. On average, *watch+phone+cloud* performed the fastest (*avg. latency*=1.8s, *SD*=0.2s). This was followed by *watch+phone* (*avg.*=2.2s, *SD*=0.1s), which needed more time for running the model on the phone (vs. cloud), and *watch+cloud* (*avg.*=2.4s, *SD*=0.0s) which required more time to compute features on the watch (vs. phone in *watch+phone+cloud*). As expected, *watch-only* was significantly slower (*avg.*=5.9s, *SD*=0.1s) and is thus, currently unusable (though future smartwatch generations will be more capable). In summary, except for watch-only, all architectures had a latency of ~2s; we evaluate whether this is acceptable in the user study.

4.2.6 Summary. Overall, we found that *watch+phone* and *watch+phone+cloud* outperformed the *watch+cloud* architecture for all system parameters. Additionally, the *watch-only* architecture was impractical for real-life use due to high CPU, memory, and battery usage, as well as a large end-to-end latency. Within the phone-based architectures, the *watch+phone+cloud* performed better than the *watch+phone*.

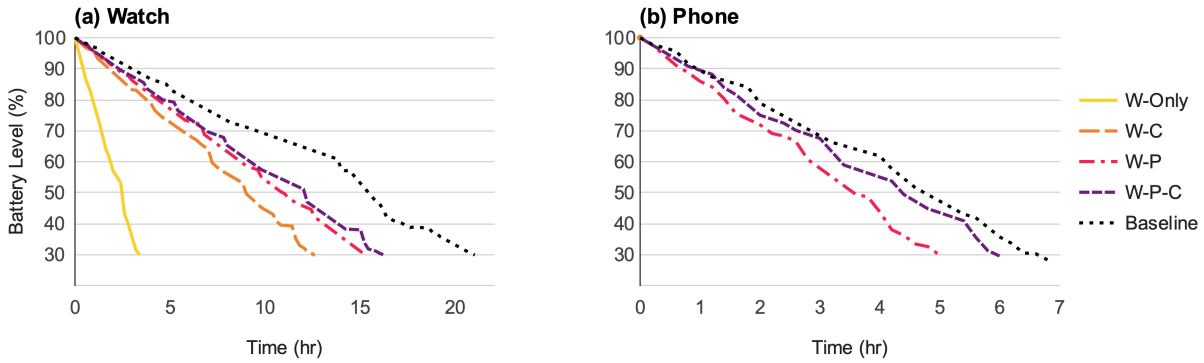

Figure 7: Battery level over time for watch (a) and phone (b) for the four architectures. *W-only*=watch only, *W-C*=watch+cloud, *W-P*=watch+phone, *W-P-C*=watch+phone+cloud. Baseline represents the case without the SoundWatch app running

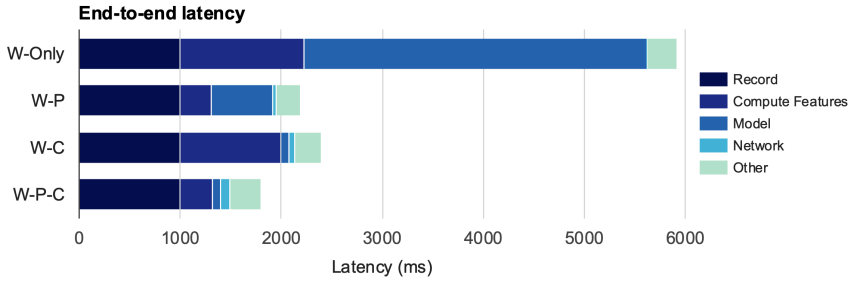

Figure 8: Breakdown of end-to-end latency for the four architectures.

5 USER STUDY

To gather qualitative feedback on our system results and general reactions to smartwatch-based sound awareness in multiple contexts, we performed a lab and campus walkthrough evaluation of our SoundWatch app with eight DHH participants. SoundWatch is designed to support all four device architectures and can be switched between them; however, based on our system experiments above, we used the best performing architecture (*watch+phone*) and model (VGG-lite) for the user study.

5.1 Participants

We recruited eight DHH participants (3 women, 3 men, 2 non-binary) using email, social media and snowball sampling (Table 2). Participants were on average 34.8 years old (*SD*=16.8, *range*=20–63). Four had profound hearing loss, three had severe, and one had moderate. Seven reported onset as congenital and one reported one year of age. Seven participants reported using hearing devices: three participants used cochlear implants, one used hearing aids, and three used both. For communication, five participants preferred sign language, and three preferred to speak verbally. All reported fluency with reading English (5/5 on rating scale, 5 is best). Participants received $40 as compensation.

Table 2: Demographics of the DHH participants.

ID	Age	Gender	Identity	Hearing loss	Onset age	Hearing device
P1	31	Male	hard of hearing	Moderate	Birth	Hearing aids
P2	26	Female	deaf	Profound	1 year	Cochlear implants
P3	20	Non-binary	deaf	Profound	Birth	Cochlear implants
P4	20	Female	hard of hearing	Severe	Birth	Both
P5	57	Male	deaf	Severe	Birth	Both
P6	23	Female	Deaf	Profound	Birth	Both
P7	38	Non-binary	Deaf	Severe	Birth	None
P8	63	Male	Deaf	Profound	Birth	Cochlear implants

5.2 Procedure

The in-person procedure took place on a university campus and lasted up to 90 minutes. Sessions were led by the first author who is hard of hearing and knows level-2 ASL. A real-time transcriptionist attended all sessions, and five participants opted to additionally have a sign language interpreter. Instructions and interview questions were presented visually on an iPad (see supplementary materials), while responses and follow-up discussion were spoken or translated to/from ASL. The session began with a demographic and background questionnaire, followed by a three-part protocol, the first and third of which took place in a quiet conference room:

5.2.1 Part 1: Introduction of SoundWatch prototype (5-10 mins). In the first phase, we asked about general thoughts on using smartwatches for sound awareness. Participants were then asked to wear the watch running SoundWatch. To demonstrate the app, a researcher made three example sounds (speech, door knock, and phone ring) while explaining the watch and the phone UI. Participants were also encouraged to make their own sounds such as by speaking or knocking to examine SoundWatch's behavior.

5.2.2 Part 2: Campus walk (20-25 mins). For Part 2, the researcher and the participant (with the watch and phone) visited three locations on campus in a randomized order: (1) a home-like location (lounge of our building), (2) an office-like location (a grad student office), and (3) an outdoor location (a bus stop). These locations enabled the participants to experience SoundWatch in different contexts and soundscapes. In each location, participants used the watch naturally for about five minutes (*e.g.,* by sitting on a chair in an office, or walking and conversing outdoors). In locations with insufficient sound activity (*e.g.,* if the lounge was empty on weekends), the researcher produced some sounds (*e.g.,* by running the microwave, or washing hands). Participants were also encouraged to use the sound customization options (mute on watch and checklist on phone) if they desired. Before exiting each location, participants filled a short feedback form to rate their experience on a 5-point scale and document any open-ended comments.

5.2.3 Part 3: Post-trial interview (45-50 mins). After completing the three locations, participants returned to the lab for Part 3. Here, we conducted a semi-structured interview inquiring about the participant's overall experience and perceptions of SoundWatch across the different locations, reactions to the UI, privacy concerns, and future design ideas. We then transitioned to asking about specific technical considerations, including accuracy-latency tradeoffs and the four possible SoundWatch architectures. For accuracy-latency, we explained the concept and then asked about their expectations for minimum accuracy, maximum delay, and whether their perspectives changed based on sound type (*e.g.,* urgent *vs.* non-urgent sounds) or context (*e.g.,* home, office). To help discuss the four SoundWatch architectures—and to more easily allow our participants to understand and track differences—we prepared a chart (see supplementary materials) enumerating key characteristics such as battery or network usage and a *HIGH*, *MEDIUM*, or *LOW* rating based on our system experiment findings. Finally, we asked participants to rate the "ease-of-use" of each architecture (high, med, or low) by weighing factors such as the Internet requirement, number of devices to carry (*e.g.,* 1 for *watch-only vs.* 2 for *watch+phone*), and the size of visual display (*e.g.,* small for watch *vs.* medium for phone), and to specify reasons for their rating.

5.3 Data Analysis

The interview transcripts and the in-situ form responses were analyzed using an iterative coding approach [7]. To begin, we randomly selected three out of eight transcripts; two researchers independently read these transcripts and identified a small set of potential codes. These codes were used to develop a mutually agreeable initial codebook to apply holistically to the data. The two researchers then used a copy of the codebook to independently code the three transcripts, while simultaneously refining their own codebook (adding, merging or deleting codes). After this step, the researchers met again to discuss and refine the codebook, and resolve any disagreements on the code assignments. The final codebook contained a two-level hierarchy (12 level-1 codes, 41 level 2- codes), of which the level-1 codes form the high-level themes. This codebook was then used to independently code the remaining five transcripts. For this last step, interrater agreement between the two coders, measured using Krippendorff's alpha [24], was on average 0.79 (*SD*=0.14, *range*=0.62-1) and the raw agreement 93.8% (*SD*=6.1%, *range*=84.4%-100). Again, the conflicting code assignments were resolved through consensus.

5.4 Findings

We detail experience with SoundWatch during the campus walk as well as comments on model accuracy-latency, different system architectures, and the user interface. Quotes are drawn verbatim from the post-trial interview transcripts and in-situ form responses.

5.4.1 Campus walk experience. For the campus walk with SoundWatch, we describe the participants' thoughts on the overall usefulness of the prototype and the variation with contexts. All participants found the watch generally useful in all three contexts (a home-like lounge, office, and outdoors) to help with the everyday activities. For example,

> "My wife and I tend to leave the water running all the time so this app could be beneficial and save on water bills. It was helpful to know when the microwave beeps instead of having to stare at the time [display]." (P6)

> "This is very useful for desk type work situations. I can use the watch to help alert me if someone is knocking the door, or coming into the room from behind me." (P7)

However, participants (8/8) also noticed problems with SoundWatch, the most notable being delay and misclassifications; the latter were higher in outdoor contexts than in others. For example,

> "Delay might be a problem. When a person came into a room, that person surprised me before the watch notified me [about door-in-use]" (P5)

> "It doesn't feel refined enough with outside sounds and background noises. The app is perfect for quiet settings such as home and outdoor activities (e.g., hiking in the woods). [While outdoors,] some sounds

were misinterpreted, such as cars were recognized as water running" (P3)

In-situ feedback form ratings reflect these comments, with average usefulness for lounge (4.8/5, *SD*=0.4) and office (4.6/5, *SD*=0.5) being higher than for outdoors (3.5/5, *SD*=0.5). Even with a low usefulness rating, all participants wanted to use SoundWatch for outdoor settings, mentioning that they can use context to supplement the inaccurate feedback (5/8):

> "Sure there were some errors outdoors, but it tells me sounds are happening that I might need to be aware of, so I can [visually] check my environment..." (P7)

Besides usefulness, the app usage also changed with location. As expected, all participants chose to enable different sounds for each location; the obvious common choices were fire/smoke alarm, microwave, water running, and speech for lounge; door knock, door-in-use, and speech for office; and bird chirp, car horn, and vehicle running for outdoors, as determined from the system logs. The total number of enabled sounds were also different for each location (avg. 8.3 for lounge, *SD*=1.2; 7.5 for office, *SD*=1.5; and 4.2 for outdoors, *SD*=2.6)—for outdoors specifically, 5/8 participants speculated that the app accuracy may decrease with background noise, and thus deselected all un-important sounds to compensate. For example,

> "I deselected 'Speech' for outside because I didn't want to know someone was talking outside. It's noisy. [...] I only selected 'car honk', 'vehicle running' and 'siren' [as] they are the bare minimum I need. It seemed to work well then." (P2)

5.4.2 Model Accuracy-Latency Comparison. Because deep learning-based sound recognition will likely have some error and latency, we asked participants about the maximum tolerable delay and the minimum required accuracy for a future smartwatch app. The most common general preference was a maximum delay of "five seconds" (5/8) and a minimum accuracy of 80% (6/8); however, this choice was additionally modulated by the type of sound. Specifically, for the urgent sounds (*e.g.,* fire alarms or car horn), participants wanted the minimum possible delay (but would tolerate inaccuracy) to get quick information for any required action. For example,

> "because I'll at least know something is happening around me and I can use my eyes to look around and see if a car is honking at me..." (P2)

> "If an important sound is not occurring, I would just be disturbed for a moment, that's all [...] But, if it's an alarm and if this [watch] misses it, that is a problem." (P1)

In contrast, for non-urgent sounds (*e.g.,* speech, laughing) more accuracy was preferred because repeated errors could be annoying (7/8). For example,

> "I don't care about speech much, so if there is a conversation, well fine, doesn't matter if I know about it 1-2 second later or 5 seconds later, does it? But if it makes mistakes and I have to get up and check who is speaking every time it makes a mistake, it can be really frustrating" (P5)

Finally, if a sound is a medium priority for the participants (*e.g.,* microwave for P3), participants wanted a balance, that is, a moderate amount of delay is tolerable for moderate accuracy (7/8).

Besides variation with sound type, we asked if the accuracy-latency preference would change with the context of use (home vs. office vs. outdoors). In general, similar to the type of sound preferences, participants erred towards having less delay in more urgent contexts and vice versa. For the home, participants (8/8) wanted high accuracy (more delay is acceptable) because, for example:

> "I already know most of what is going on around my home. And when I am at home, I am generally more relaxed [so] delay is more acceptable. But, I would not want to be annoyed by errors in my off time." (P8)

For the office, participants (6/8) felt they would tolerate a moderate level of accuracy with the advantage of having less delay, because "something may be needing my attention but it's likely not a safety concern" (P8). Finally, preferences for outdoors were split: four participants wanted less delay overall with outdoor sounds, but the other four participants did not settle for a single response, saying that the tradeoff would depend on the urgency of the sound outdoors, for example:

> "if it's just a vehicle running on the road while I am walking on the sidewalk, then I would want it to only tell if it's sure that it's a vehicle running, but if a car is honking say if it behind me, I would want to know immediately." (P2)

5.4.3 Architecture Comparison. By saliently introducing the performance metrics (*e.g.,* battery usage) and usage requirements (*e.g.,* Internet connection for cloud), we gathered qualitative preferences for the four possible SoundWatch architectures: *watch-only*, *watch+phone*, *watch+cloud*, and *watch+phone+cloud* during the interview.

In general, *watch+phone* was the most preferred architecture for all participants, because, compared to *watch-only*, it is faster, requires less battery, and has more visual state available for customization. In addition, compared to cloud-based designs, *watch+phone* is more private and self-contained (does not need Internet).

However, five participants wanted the option to be able to customize the architecture on the go, mentioning that in outdoor settings, they would instead prefer to use *watch+phone+cloud* because of additional advantages of speed and accuracy. This is because in the outdoor context, data privacy was less of a concern for them. For example, P6 said:

> "Whenever the Internet is available, I prefer cloud for outdoors instead of home/office because of possible data breach at home/office [...] Accuracy problems could be more [outdoors] due to background noise and [thus] I prefer to use cloud if [the] internet is available."

Watch+cloud was preferred by two participants only for cases where it is hard to carry a phone, such as in a "gym or running outdoors" (P1); others did not share this concern as they reported always carrying the phone—for example: "I can't really imagine a situation where I would have my watch and not my phone." (P4).

Finally, *watch-only* was not preferred for any situation because of a large battery drain, and a small input area (*e.g.,* for customization).

5.4.4 User Interface Suggestions. Overall, participants appreciated the minimalistic app design, including the information conveyed (identity, loudness, and time) (8/8) and the customization options (mute button, checklist on phone) (7/8). When asked about future improvements, participants suggested three. First, they wanted the app to indicate the urgency of sounds—for example, using vibration patterns or visual colors (*e.g.,* one pattern/color for high priority sounds, and another for low priority sounds). Second, to increase utility, participants suggested to explore showing multiple overlapping sounds (5/8), the most urgent sound (3/8), or the loudest sound (2/8) instead of the most probable sound as in our design. P4 also said that conveying multiple "possible" sounds could help her compensate for inaccuracy:

> "You could give suggestions for what else sound could be when it's not able to recognize. For example, [...] if it is not able to tell between a microwave and a dishwasher, it could say "microwave or dishwasher", or at least give me an indication of how it sounds like, you know like a fan or something, so I can see and tell, oh yeah, the dishwasher is running."

Finally, two participants (P3, P8) wanted the direction of sound source for outdoor context:

> "I need to know if the vehicle is running or honking behind me or on the side of me. If it's on the side on the road, then I don't have to do anything. If it's behind me, I will move away." (P8)

When asked whether they would need direction for home or office as well, they replied no, stating that context awareness is higher for those locations (2/2):

> "No, not needed for these contexts [home and office]. I know the locations of where the sound [source] could be, if it shows "microwave", it's in the kitchen. If it's "speech", I know where [my spouse] is." (P3)

6 DISCUSSION

Our work reaffirms DHH users' needs and user interface preferences for smartwatch-based sound awareness [8, 32] but also: (1) implements and empirically compares state-of-the-art deep learning approaches for sound classification on smartwatches, (2) contributes a new smartwatch-based sound identification system with support for multiple device architectures, and (3) highlights DHH users' reactions to accuracy-latency tradeoffs, classification architectures, and potential concerns. Here, we reflect on further implications and limitations of our work.

6.1 Utility of smartwatch-based sound classification

How well does a smartwatch-based sound classification tool need to perform to provide value? As both our systems evaluation and user study reveal, this is a complex question that requires further study. While improving overall accuracy, reducing latency, and supporting a broad range of sound classes is clearly important, participants felt that *urgent* sounds should be prioritized. Thus, we wonder, would an initial sound awareness app that supports three to ten urgent sounds be useful? More work is needed here. One way to explore this question would be by releasing SoundWatch—or a similar app—to the public with multiple customization options, then studying actual usage and soliciting feedback. However, this approach also introduces ethical and safety concerns. Automatic sound classification will never be 100% accurate. High accuracy on a limited set of sounds could (incorrectly) gain the user's trust, and the app's failure to recognize a safety sound (*e.g.,* a fire alarm) even once could be dangerous. In general, a key finding of our research and of other recent work [8, 32] is that users desire *customization* (*e.g.,* which sounds to classify, notification options, sound priorities) and *transparency* (*e.g.,* classification confidence) with sound awareness tools.

6.2 Towards improving accuracy

Our user study suggests a need to further improve system accuracy or at least explore other ways to mitigate misclassification costs. One possibility, as our participants suggested, is to explore showing multiple "possible" sounds instead of the most probable sound—just as text autocomplete shows n-best words. Another possibility is to sequentially cascade two models (*e.g.,* see [35]), using the faster model to classify a small set of urgent sounds and to employ the slower model for lower-confidence classifications and less-urgent sounds. End-user customization should also be examined. While installing the app, each user could select the desired sounds and the required accuracies, and the app could dynamically fine-tune the model (*e.g.,* by using a weighted average accuracy metric based on the sound urgency). Finally, as proposed by Bragg *et al.* [3], researchers should explore end-user interactive training of the model. Here, guided by the app, participants could record sounds of interest to either improve existing sound classes or to add new ones. Of course, this training may be tedious and difficult if the sound itself is inaccessible to the DHH user.

6.3 Privacy implications

Our participants' showed concern for cloud-based classification architectures: they valued their own "sound" privacy and of others around them. However, uploading and storing data on the cloud has benefits. These datasets can be used for improving the classification model. Indeed, modern sound architectures on IoT devices (*e.g.,* Alexa, Siri) use the cloud for exchanging valuable data. A key difference to our approach is that these devices only transmit after listening to a trigger word. Thus, what are the implications for future always-on, always-listening sound awareness devices? We see three. First, the users should have control of their sound data. Indeed, P3 corroborated this:

> "I can see myself potentially using the watch + phone + cloud, if I can, [...] open my laptop and [select/deselect] what [sound data] gets uploaded and who gets to see what. Otherwise I fear that [someone] may misuse something that I don't want them to."

This data upload can also be customized based on context (*e.g.,* the office might have more private conversations than outdoors). Second, future apps will need clear privacy policies such as GDPR [56] or CCPA [57] that outline how and where the data is stored

and what guarantees the users have. Finally, users should always have access to their data and to potentially delete it, in entirety, from the cloud.

6.4 Future smartwatch applications

In contrast to past wearable sound awareness solutions [9, 18, 22], we used commercially available smartwatches, a mainstream popular device that is more socially acceptable than HMDs [9, 18] or custom hardware-based [22, 34] solutions. A recent survey with 201 DHH participants [4] showed that smartwatch-based sound awareness was preferred over smartphones as well. So, what are other compelling applications of a smartwatch for DHH users? Full speech transcription, a highly preferred feature by DHH users [4, 17] is difficult to accommodate on the small watch screen, but future deep learning work could explore highlighting important keywords or summarizing the conversation topics. Sound localization is also highly desired [3, 8] and could be explored by coupling the watch with a small external microphone array or designing a custom watch with multiple microphones. But, how best to combine different features (*e.g.,* topic summarization, direction, identity) on the watch is an open question. Goodman *et al.* [8] recently investigated different designs for combining sound identity, direction, and loudness, however, this study was formative with a focus on user interface design. Future work should explore the system design of showing multiple features with classification confidence—a challenging problem given the smartwatch's low-resource constraints.

6.5 Limitations

Our lab study included a 20-min out-of-lab component intended to help participants think about and experience SoundWatch across "real-world" contexts. While useful as an initial, exploratory study, important pragmatic issues could not be investigated such as user perception of battery life, integration of the watch into daily life, and long-term usage patterns. Future work should perform a deployment study and compare results with our lab findings.

Moreover, our model accuracy results, though performed on real-life recordings of 20 sounds, do not accurately reflect real-world use as other sounds beyond those 20 may also occur. Our tests, however, were enough for our goal to compare the models and contextualize the user study findings. A more accurate experiment would include a *post hoc* analysis of sound data collected from longitudinal watch use.

Finally, we considered our DHH participants (who identify as deaf, Deaf or hard of hearing) as a homogenous group while reporting user study findings. Indeed, past work [3, 4] shows that these groups, despite their cultural differences, have synergetic access needs and preferences. Recruiting cross-culturally allowed us to explore solutions for a diversity of users. Nevertheless, future work should examine how preferences may vary with culture and hearing levels.

7 CONCLUSION

In this paper, we performed a quantitative examination of modern deep learning-based sound classification models and architectures as well as a lab exploration of a novel smartwatch sound awareness app with eight DHH participants. We found that our best classification model performed similar to the state of the art for non-portable devices while requiring a substantially less memory (~1/3rd), and that the phone-based architectures outperformed the watch-centric designs in terms of CPU, memory, battery usage, and end-to-end latency. Qualitative findings from the user study contextualize our system experiment results, and also uncover ideas, concerns, and design suggestions for future wearable sound awareness technology.

ACKNOWLEDGMENTS

We thank Emma McDonnell for reviewing paper drafts and Ana Liu for helping recruit participants. This work was supported by National Science Foundation Grant no: IIS-1763199.

REFERENCES

[1] Sharath Adavanne, Archontis Politis, and Tuomas Virtanen. 2019. TAU Moving Sound Events 2019 - Ambisonic, Anechoic, Synthetic IR and Moving Source Dataset.

[2] Rosa Ma Alsina-Pagès, Joan Navarro, Francesc Al\'\ias, and Marcos Hervás. 2017. homesound: Real-time audio event detection based on high performance computing for behaviour and surveillance remote monitoring. *Sensors* 17, 4: 854.

[3] Danielle Bragg, Nicholas Huynh, and Richard E. Ladner. 2016. A Personalizable Mobile Sound Detector App Design for Deaf and Hard-of-Hearing Users. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, 3–13.

[4] Leah Findlater, Bonnie Chinh, Dhruv Jain, Jon Froehlich, Raja Kushalnagar, and Angela Carey Lin. 2019. Deaf and Hard-of-hearing Individuals' Preferences for Wearable and Mobile Sound Awareness Technologies. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*., 1–13.

[5] Pasquale Foggia, Nicolai Petkov, Alessia Saggese, Nicola Strisciuglio, and Mario Vento. 2015. Reliable detection of audio events in highly noisy environments. *Pattern Recognition Letters* 65: 22–28.

[6] Eduardo Fonseca, Jordi Pons Puig, Xavier Favory, Frederic Font Corbera, Dmitry Bogdanov, Andres Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. 2017. Freesound datasets: a platform for the creation of open audio datasets. In *Hu X, Cunningham SJ, Turnbull D, Duan Z, editors. Proceedings of the 18th ISMIR Conference; 2017 oct 23-27; Suzhou, China.[Canada]: International Society for Music Information Retrieval; 2017. p. 486-93.*

[7] Karyn L. Galvin, Jan Ginis, Robert S. Cowan, Peter J. Blamey, and Graeme M. Clark. 2001. A Comparison of a New Prototype Tickle Talker^{TM} with the Tactaid 7. *Australian and New Zealand Journal of Audiology* 23, 1: 18–36.

[8] Steven Goodman, Susanne Kirchner, Rose Guttman, Dhruv Jain, Jon Froehlich, and Leah Findlater. Evaluating Smartwatch-based Sound Feedback for Deaf and Hard-of-hearing Users Across Contexts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1–13.

[9] Benjamin M Gorman. 2014. VisAural: a wearable sound-localisation device for people with impaired hearing. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility*, 337–338. https://doi.org/10.1145/2661334.2661410

[10] Benjamin M Gorman and David R Flatla. 2014. VisAural: A Tool for Converting Audible Signals into Visual Cues.

[11] Seungyeop Han, Haichen Shen, Matthai Philipose, Sharad Agarwal, Alec Wolman, and Arvind Krishnamurthy. 2016. Mcdnn: An approximation-based execution framework for deep stream processing under resource constraints. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, 123–136.

[12] Shawn Hershey, Sourish Chaudhuri, Daniel P W Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, and others. 2017. CNN architectures for large-scale audio classification. In*2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 131–135.

[13] Eric G Hintz, Michael D Jones, M Jeannette Lawler, Nathan Bench, and Fred Mangrubang. 2015. Adoption of ASL classifiers as delivered by head-mounted displays in a planetarium show. *Journal of Astronomy & Earth Sciences Education (JAESE)*2, 1: 1–16.

[14] F Wai-ling Ho-Ching, Jennifer Mankoff, and James A Landay. 2003. Can You See What I Hear?: The Design and Evaluation of a Peripheral Sound Display for the Deaf. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03), 161–168. https://doi.org/10.1145/642611.642641

[15] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

[16] Shahidul Islam, William G Buttlar, Roberto G Aldunate, and William R Vavrik. 2014. Measurement of pavement roughness using android-based smartphone application. *Transportation Research Record* 2457, 1: 30–38.

[17] Dhruv Jain, Bonnie Chinh, Leah Findlater, Raja Kushalnagar, and Jon Froehlich. 2018. Exploring Augmented Reality Approaches to Real-Time Captioning: A Preliminary Autoethnographic Study. In *Proceedings of the 2018 ACM Conference Companion Publication on Designing Interactive Systems*, 7–11.

[18] Dhruv Jain, Leah Findlater, Christian Volger, Dmitry Zotkin, Ramani Duraiswami, and Jon Froehlich. 2015. Head-Mounted Display Visualizations to Support Sound Awareness for the Deaf and Hard of Hearing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 241–250.

[19] Dhruv Jain, Rachel Franz, Leah Findlater, Jackson Cannon, Raja Kushalnagar, and Jon Froehlich. 2018. Towards Accessible Conversations in a Mobile Context for People who are Deaf and Hard of Hearing. In *Proceedings of the International Conference on Computers and Accessibility (ASSETS)*, 12 pages.

[20] Dhruv Jain, Angela Carey Lin, Marcus Amalachandran, Aileen Zeng, Rose Guttman, Leah Findlater, and Jon Froehlich. 2019. Exploring Sound Awareness in the Home for People who are Deaf or Hard of Hearing. In *SIGCHI Conference on Human Factors in Computing Systems (CHI). In Submission.*

[21] Dhruv Jain, Kelly Mack, Akli Amrous, Matt Wright, Steven Goodman, Leah Findlater, and Jon E Froehlich. 2020. HomeSound: An Iterative Field Deployment of an In-Home Sound Awareness System for Deaf or Hard of Hearing Users. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (CHI '20), 1–12. https://doi.org/10.1145/3313831.3376758

[22] Y Kaneko, Inho Chung, and K Suzuki. 2013. Light-Emitting Device for Supporting Auditory Awareness of Hearing-Impaired People during Group Conversations. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, 3567–3572. https://doi.org/10.1109/SMC.2013.608

[23] Yiping Kang, Johann Hauswald, Cao Gao, Austin Rovinski, Trevor Mudge, Jason Mars, and Lingjia Tang. 2017. Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. *ACM SIGARCH Computer Architecture News* 45, 1: 615–629.

[24] Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications.

[25] Gierad Laput, Karan Ahuja, Mayank Goel, and Chris Harrison. 2018. Ubicoustics: Plug-and-play acoustic activity recognition. In *The 31st Annual ACM Symposium on User Interface Software and Technology*, 213–224.

[26] Lie Lu, Hong-Jiang Zhang, and Hao Jiang. 2002. Content analysis for audio classification and segmentation. *IEEE Transactions on speech and audio processing* 10, 7: 504–516.

[27] Tara Matthews, Janette Fong, F. Wai-Ling Ho-Ching, and Jennifer Mankoff. 2006. Evaluating non-speech sound visualizations for the deaf. *Behaviour & Information Technology* 25, 4: 333–351.

[28] Tara Matthews, Janette Fong, and Jennifer Mankoff. 2005. Visualizing non-speech sounds for the deaf. In *Proceedings of the 7th international ACM SIGACCESS conference on Computers and Accessibility - Assets '05*, 52–59. https://doi.org/10.1145/1090785.1090797

[29] Amrita Mazumdar, Brandon Haynes, Magda Balazinska, Luis Ceze, Alvin Cheung, and Mark Oskin. 2019. Perceptual Compression for Video Storage and Processing Systems. In *Proceedings of the ACM Symposium on Cloud Computing*, 179–192.

[30] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2016. TUT Sound events 2016. https://doi.org/10.5281/zenodo.45759

[31] Matthias Mielke and Rainer Brück. 2015. A Pilot Study about the Smartwatch as Assistive Device for Deaf People. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, 301–302.

[32] Matthias Mielke and Rainer Brueck. 2015. Design and evaluation of a smartphone application for non-speech sound awareness for people with hearing loss. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, 5008–5011.

[33] Silvia Pfeiffer, Stephan Fischer, and Wolfgang Effelsberg. 1997. Automatic audio content analysis. In *Proceedings of the fourth ACM international conference on Multimedia*, 21–30.

[34] A J Phillips, A R D Thornton, S Worsfold, A Downie, and J Milligan. 1994. Experience of using vibrotactile aids with the profoundly deafened. *European journal of disorders of communication* 29, 1: 17–26.

[35] Dhrubojyoti Roy, Sangeeta Srivastava, Aditya Kusupati, Pranshu Jain, Manik Varma, and Anish Arora. 2019. One size does not fit all: Multi-scale, cascaded RNNs for radar classification. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 1–10.

[36] Frank A Saunders, William A Hill, and Barbara Franklin. 1981. A wearable tactile sensory aid for profoundly deaf children. *Journal of Medical Systems* 5, 4: 265–270.

[37] John Saunders. 1996. Real-time discrimination of broadcast speech/music. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, 993–996.

[38] Eric Scheirer and Malcolm Slaney. 1997. Construction and evaluation of a robust multifeature speech/music discriminator. In *1997 IEEE international conference on acoustics, speech, and signal processing*, 1331–1334.

[39] Khurram Shahzad and Bengt Oelmann. 2014. A comparative study of in-sensor processing vs. raw data transmission using ZigBee, BLE and Wi-Fi for data intensive monitoring applications. In *2014 11th International Symposium on Wireless Communications Systems (ISWCS)*, 519–524.

[40] Liu Sicong, Zhou Zimu, Du Junzhao, Shangguan Longfei, Jun Han, and Xin Wang. 2017. UbiEar: Bringing Location-independent Sound Awareness to the Hard-of-hearing People with Smartphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2: 17.

[41] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*.

[42] Sasha Targ, Diogo Almeida, and Kevin Lyman. 2016. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*.

[43] Martin Tomitsch and Thomas Grechenig. 2007. Design Implications for a Ubiquitous Ambient Sound Display for the Deaf. In *Conference & Workshop on Assistive Technologies for People with Vision & Hearing Impairments Assistive Technology for All Ages (CVHI 2007)*.

[44] Eddy Yeung, Arthur Boothroyd, and Cecil Redmond. 1988. A Wearable Multichannel Tactile Display of Voice Fundamental Frequency. *Ear and Hearing* 9, 6: 342–350.

[45] Hanfeng Yuan, Charlotte M. Reed, and Nathaniel I. Durlach. 2005. Tactual display of consonant voicing as a supplement to lipreading. *The Journal of the Acoustical Society of America* 118, 2: 1003. https://doi.org/10.1121/1.1945787

[46] Hosted models | TensorFlow Lite. Retrieved May 5, 2020 from https://www.tensorflow.org/lite/guide/hosted_models

[47] BBC Sound Effects. Retrieved September 18, 2019 from http://bbcsfx.acropolis.org.uk/

[48] Network Sound Effects Library. Retrieved September 15, 2019 from https://www.sound-ideas.com/Product/199/Network-Sound-Effects-Library

[49] UPC-TALP dataset. Retrieved September 18, 2019 from http://www.talp.upc.edu/content/upc-talp-database-isolated-meeting-room-acoustic-events

[50] TicWatch Pro - Mobvoi. Retrieved May 5, 2020 from https://www.mobvoi.com/us/pages/ticwatchpro

[51] Honor 7X - Huawei. Retrieved May 5, 2020 from https://www.gsmarena.com/honor_7x-8880.php

[52] AI Inference: Applying Deep Neural Network Training. Retrieved May 5, 2020 from https://mitxpc.com/pages/ai-inference-applying-deep-neural-network-training

[53] Measure app performance with Android Profiler | Android Developers. Retrieved May 5, 2020 from https://developer.android.com/studio/profile/android-profiler

[54] Profile battery usage with Batterystats and Battery Historian. Retrieved May 5, 2020 from https://developer.android.com/topic/performance/power/setup-battery-historian

[55] Mobvoi TicWatch E2. Retrieved September 15, 2019 from https://www.mobvoi.com/us/pages/ticwatche2

[56] General Data Protection Regulation (GDPR) – Official Legal Text. Retrieved July 21, 2020 from https://gdpr-info.eu/

[57] California Consumer Privacy Act (CCPA) | State of California - Department of Justice - Office of the Attorney General. Retrieved July 21, 2020 from https://oag.ca.gov/privacy/ccpa