

Evaluating Acoustic Data Transmission Schemes for Ad-Hoc Communication Between Nearby Smart Devices

FLORENTIN PUTZ, PHILIPP FORTMANN, JAN FRANK, CHRISTOPH HAUGWITZ, MARIO KUPNIK, and MATTHIAS HOLICK, Technical University of Darmstadt, Germany

Acoustic data transmission offers a compelling alternative to Bluetooth and NFC by leveraging the ubiquitous speakers and microphones in smartphones and IoT devices. However, most research in this field relies on simulations or limited on-device testing, which makes the real-world reliability of proposed schemes difficult to assess. We systematically reviewed 31 acoustic communication studies for commodity devices and found that none provided accessible source code. After contacting authors and re-implementing three promising schemes, we assembled a testbed of eight representative acoustic communication systems. Using over 11 000 smartphone transmissions in both realistic indoor environments and an anechoic chamber, we provide a systematic and repeatable methodology for evaluating the reliability and generalizability of these schemes under real-world conditions. Our results show that many existing schemes face challenges in practical usage, largely due to severe multipath propagation indoors and varying audio characteristics across device models. To support future research and foster more robust evaluations, we release our re-implementations alongside the first comprehensive dataset of real-world acoustic transmissions. Overall, our findings highlight the importance of rigorous on-device testing and underscore the need for robust design strategies to bridge the gap between simulation results and reliable IoT deployments.

CCS Concepts: • **Networks** → **Mobile ad hoc networks**; Short-range networks; Network reliability; **Network performance evaluation**; • **Human-centered computing** → **Ubiquitous and mobile devices**; **Smartphones**; • **Hardware** → Signal processing systems.

Additional Key Words and Phrases: acoustic communication; ultrasonic communication; near-ultrasonic; audio-based communication; data-over-sound; reproducibility; conceptual replication; generalizability; dataset; open data; open source; audio recordings; iot; ubiquitous computing; experimental testbed; systematic literature study; re-implementation; performance evaluation; device-to-device communication; smartphone-to-smartphone communication; short-range; ad-hoc; interoperability; spontaneous; nearby; proximity

ACM Reference Format:

Florentin Putz, Philipp Fortmann, Jan Frank, Christoph Haugwitz, Mario Kupnik, and Matthias Hollick. 2026. Evaluating Acoustic Data Transmission Schemes for Ad-Hoc Communication Between Nearby Smart Devices. *ACM Trans. Internet Things* 7, 1, Article 8 (February 2026), 31 pages. <https://doi.org/10.1145/3779439>

1 Introduction

Smartphones and Internet of Things (IoT) devices can communicate using sound waves via their built-in speakers and microphones [39]. Acoustic data transmission promises to be a practical alternative to Bluetooth and near-field communication (NFC), given the universal presence of audio I/O in devices like smartphones, smartwatches, smart speakers, and home automation systems. Over the past decade, there has been increasing interest in using acoustic communication as a wireless, short-range ad-hoc channel to facilitate ubiquitous and mobile computing use cases, such as mobile payments [68], context-aware computing [40, 41], or location-based services [45, 71]. Its location-limited nature also fits well with ubiquitous computing scenarios, where users intuitively

Authors' Contact Information: Florentin Putz, fputz@seemoo.de; Philipp Fortmann, pfortmann@seemoo.de; Jan Frank, jfrank@seemoo.de; Christoph Haugwitz, christoph.haugwitz@tu-darmstadt.de; Mario Kupnik, mario.kupnik@tu-darmstadt.de; Matthias Hollick, mhollick@seemoo.de, Technical University of Darmstadt, Darmstadt, Germany.

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Internet of Things*, <https://doi.org/10.1145/3779439>.

bring devices into proximity so they can “talk” [11], aligning with intuitive user behaviour [6, 25, 26]. The widespread availability of low-cost IoT hardware equipped with speakers and microphones has even led to commercial uptake, such as the smart-speaker manufacturer Sonos employing acoustic communication for spatial awareness and device pairing [27].

Despite the growing body of research proposing acoustic data transmission systems [4, 13, 14, 33, 38, 70], many have been tested only through simulations or under very limited real-world conditions. Real-device testing, however, is crucial to account for factors like hardware imperfections, user handling, realistic signal propagation, and background noise—factors that are notoriously difficult to model precisely. Moreover, acoustic channels differ substantially from radio channels: sound waves travel more than 87 000 times slower than electromagnetic waves, leading to delay spreads on the order of tens of milliseconds [33, 59]. Acoustic signals also suffer from highly variable ambient noise and, due to low sample rates on most smartphones and IoT devices, bandwidth is limited to at most 22 kHz, making high-throughput transmissions challenging. Existing evaluations typically focus on individual devices or scenarios, providing little insight into how well a scheme might generalize across diverse real-world settings.

At the same time, there is a distinct gap between academic research and commercial practice. Commercial products for acoustic data transmission on smart devices typically achieve only 10–100 bps [9, 13, 27] and occasionally up to 200 bps [62], prioritizing robustness and reliability in noisy environments. In contrast, researchers often report much higher throughput up to 500–10 000 bps [4, 14, 70] or beyond [69]. This discrepancy raises fundamental questions about whether these systems can maintain reliability in real-world scenarios, or whether additional constraints—such as hardware limitations, user behavior, or environmental interference—ultimately force practical systems to operate at lower data rates.

A major barrier to resolving these questions is that publications in this field often omit source code, limiting independent validation and hindering follow-up work. In this paper, we take a first step toward consolidating existing findings by investigating how well representative acoustic data transmission schemes generalize to various practical scenarios. Specifically, we address the following research questions:

RQ1: “How challenging is it to obtain or re-implement systems proposed in this field?”

RQ2: “How well do these schemes generalize to ad-hoc smartphone communication use cases?”

RQ3: “Which practical challenges affect nearby acoustic data transmission between smart devices?”

1.1 Contributions

This paper presents the first **independent evaluation of the generalizability of acoustic data communication schemes** proposed by other researchers, comparing them in realistic settings. Our contributions are as follows:

- **Systematic literature study.** We conducted a systematic review and identified 31 publications proposing acoustic data transmission systems for commercial off-the-shelf (COTS) devices (Section 4). Regrettably, none of these studies provide implementation source code.
- **Sourcing and re-implementation of representative schemes.** We reached out to the authors for software implementations. Only three authors supplied functioning code, one provided a non-functioning implementation, and the others either did not respond or had lost access. To better understand the generalizability of acoustic communication research, we additionally re-implemented three promising schemes (Section 5).
- **Evaluation of generalizability.** In total, we evaluated the practical reliability of eight acoustic communication schemes [4, 13, 14, 33, 38, 70], including two variants of the popular

open-source project `ggwave` [12] (Section 6). Our testbed focuses on diverse smartphone models as a practical and accessible subset of IoT devices, although the underlying insights hold for any device with comparable audio input and output capabilities. We expand prior work by evaluating these schemes across various IoT use cases and analyzing how factors such as device distance, model, user handling, and background noise affect communication reliability (measured by the rate of errors in transmissions).

- **Measurement dataset.** We publish our re-implementations and analysis code along with the first dataset in this field—11 900 real recordings of acoustic data transmissions—to support transparent, reproducible research [51].

1.2 Scope

This work supports the IoT research community by analyzing acoustic data transmission schemes from the perspective of researchers or developers who want to integrate this technology into smart devices for nearby data exchange. From this perspective, acoustic communication should be as seamless to adopt as Bluetooth or NFC, requiring no specialized knowledge of the underlying wireless standards. Accordingly, we evaluate representative schemes on multiple smartphone models, which serve as the most common and versatile IoT form factor. Rather than focusing on physical-layer performance or simulations, we treat the complete communication system as a black box, including all components—such as synchronization and error correction—as originally described by the authors, ensuring we analyze its practical suitability on real devices.

2 System model

This study evaluates *acoustic data transmission schemes*: software implementations that broadcast data from one device to nearby devices using sound waves. From a system developer’s perspective, each scheme acts as a black box that transmits data on one device and receives it on another. For our evaluation, we model this capability as an interface with two functions, depicted in Figure 1:

- (1) A **transmitter**, $wav = TX(b_{TX})$, which converts data bits into a WAV file for playback on a loudspeaker.
- (2) A **receiver**, $b_{RX} = RX(wav)$, which extracts data from a WAV file containing the microphone recording of the transmission.

This interface separates the digital signal processing of the modulator and demodulator from audio playback and recording. Although some schemes handle playback and recording internally, we treat them independently for clarity. In practice, a developer might not store intermediate WAV files but instead stream samples directly from a buffer. Nonetheless, our interface fundamentally reflects real-world usage and facilitates a fair comparison of each scheme under uniform playback and recording conditions.

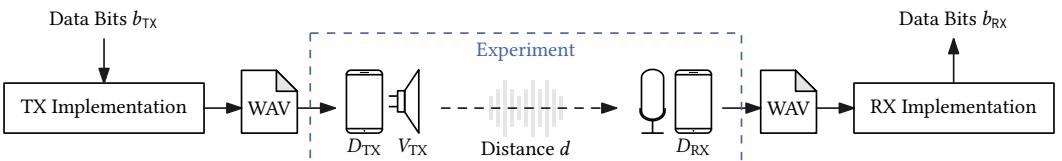


Fig. 1. **Acoustic data transmission model.** In our experiments, we play the WAV files from the TX implementation on device D_{TX} (at volume V_{TX}) and record this on device D_{RX} . The resulting WAV file gets decoded using the RX implementation.

2.1 Scenarios

We focus on the use case of data exchange between nearby smart devices, where software modifications are possible (e.g., by installing apps on smartphones) but changes to hardware or firmware are not. Our emphasis is on broadcast schemes that enable communication without requiring special arrangements with the communication partner. We divide common use cases for acoustic communication (Section 3) into three categories based on the communication distance—a key factor affecting usability. The practical difference in usability is significant, ranging from the need for devices to touch (as in NFC) to the flexibility of holding your device several meters away from your communication partner. Throughout this paper, we focus on these concrete communication scenarios, which represent most current use cases:

- (1) **Near distance** ($d \leq 10$ cm): Transmitting a public key or a small file (≈ 4096 bit) over a short distance, similar to NFC. Examples are mobile payments [8, 63], secure device pairing [31, 64, 70], and access control [8, 45].
- (2) **Medium distance** ($10 \text{ cm} < d \leq 1 \text{ m}$): Transmitting a public key fingerprint or a hash value (≈ 128 bit) over a medium distance. Examples are IoT device provisioning [27, 58], side channels for air-gapped devices [54], and control channels for personal electronic devices and hearing aids [43].
- (3) **Far distance** ($d > 1 \text{ m}$): Transmitting a short ID (≈ 16 bit) over larger distances. Examples are providing location-specific information near public signs or in shops [10], indoor IoT communication [18, 23], and enhancing media experiences, such as social television [36] and second screen services via hidden audio data in TV and radio [28].

3 Related work

In this section, we briefly review prior research on acoustic data transmission before presenting our systematic literature study in Section 4.

Although smartphone and IoT audio hardware is primarily intended for media playback and voice communication, it can also be repurposed for sensing [5, 66, 72] or even for aerial acoustic data transmission. In the latter case, speakers encode information in sound waves using frequency or amplitude variations, similar to electromagnetic communication [39], thereby enabling data exchange across a diverse group of devices, including smartphones and voice-controlled appliances, without special radio frequency (RF) hardware [11]. This communication channel is limited by the typical 0 – 22 kHz audio range (due to common sampling rates of 44.1 kHz), pronounced frequency selectivity [21], and non-linearities at higher volumes (see Section 7.3). Data can be transmitted audibly or using near-ultrasonic frequencies, and—unlike Wi-Fi or Bluetooth—no network setup or pairing is required, allowing ad-hoc broadcasts over several meters. Because it is entirely software-defined, acoustic communication can be easily deployed on existing devices and tailored to support a variety of communication needs [38].

Research in ubiquitous and mobile computing has explored acoustic data transmission for context-aware computing [40, 41] and location-limited communication between nearby devices [1, 67]. The limited range has also attracted interest for secure device interactions [23, 52], including device pairing [15, 16, 50, 53, 70] and protecting implantable medical devices [17]. User experience aspects of acoustic transmissions have been evaluated [38, 39, 42, 52], as the audibility of the data exchange is a new aspect distinguishing it from previous wireless technologies. Various commercial applications leverage acoustic data transmission for mobile payments [8, 63], context-aware applications [37], location-based services [9, 60], data transfer [62], and device pairing [31, 64]. Despite its application potential, no previous work has attempted to systematically evaluate these systems on real devices. The absence of published source code hinders replication of prior results.

4 Literature study

We conducted a systematic literature survey to gather all papers that propose new aerial acoustic software-defined communication systems, according to the following scope.

4.1 Scope

To ensure our research aligns with ubiquitous computing and everyday consumer interactions, we focused our literature survey on papers that describe practical implementations of aerial acoustic device-to-device communication specifically for COTS devices, like smartphones. We excluded studies related to other types of acoustic communication, such as underwater or industrial uses, and those that are not applicable to general-purpose data transmission systems.

4.2 Method

To identify relevant literature on acoustic data transmission schemes, we began by collecting a wide array of papers related the subject. We then refined this collection to include only those with research implementations for our target use case. Our method was guided by PRISMA [48], which is a framework for assessing large bodies of medical research in a structured manner. PRISMA provides a flowchart with three steps:

- (1) **Identification:** Define the data sources and search strategy.
- (2) **Screening:** Filter the research according to the desired scope.
- (3) **Inclusion:** Select which papers to include.

4.2.1 *Identification.* As the information source for this literature study we used the Web of Science (WoS),¹ which is a comprehensive database indexing journals and conference proceedings. We selected WoS due to its extensive, regularly updated catalog and its advanced filtering capabilities with nested logical operators, which are crucial for narrowing the results to our specific scope. Our search strategy balanced breadth and manageability. We designed and executed two mutually exclusive search queries² on May 18, 2023, yielding 631 papers. These queries targeted papers published between 2000 and 2022 in the fields of acoustics, computer science, and engineering, focusing on relevant keywords in the titles. We excluded papers related to nautical or medical fields by blocking specific keywords associated with underwater communication and ultrasonic imaging. The second query broadened the search to include a wider range of keyword combinations. We refer to the query URLs² for details and keywords.

4.2.2 *Screening.* We processed all identified papers to determine their relevance to our scope. In the first iteration, we reviewed the titles and abstracts of all 631 papers and eliminated 422 papers that were clearly out of scope—such as those focusing on speech-based communication, tracking with audio, or communication through metal barriers. In a second iteration, we examined the full text of the remaining 209 papers, classifying them based on their applicability to ubiquitous computing use cases and whether they describe a testable implementation for COTS devices. We excluded papers dependent on specialized hardware, like microphone or speaker arrays for multiple-input and multiple-output (MIMO) communication, and those embedding data in music, unless the scheme could be utilized as a general-purpose communication system.

4.2.3 *Inclusion.* In total, our literature study identified 31 papers that directly align with our study’s scope and describe a testable communications scheme (as shown in Table 1).

¹<https://clarivate.com/products/web-of-science/>, last access: 2025-03-03

²Search query 1: <https://www.webofscience.com/wos/woscc/summary/4c89e96d-5de5-4543-9167-23d73086d8e6-8a0aeac3/relevance/1>; search query 2: <https://www.webofscience.com/wos/woscc/summary/392d145e-a4ac-4aff-8995-ad2103dd116b-898dca59/relevance/1>.

Table 1. **Surveyed acoustic communication schemes**, sorted by throughput. The right table continues from the left. Data not available in the publications is marked with “?”.³

Name	Modulation	Frequencies [kHz]	Inaudible	Throughput [bps]	Name	Modulation	Frequencies [kHz]	Inaudible	Throughput [bps]
Moriyama et al. [44]	ASK	~19	✓	5	Gonçalves et al. [14]	OFDM + PSK	0 – 4	✗	600
Kim et al. [30]	16-FSK	18 – 20	✓	5	PriWhisper [70, 71]	MFSK	9 – 17	✗	1000
Ka et al. [28]	Chirp	18.5 – 19.5	✓	15	DigitalVoices [38]	ASK, (M)FSK, SS	1 – 10	✗	1280
Lee2015 [33]	Chirp	19.5 – 22	✓	16	A-NFC [35]	QPSK	16 – 21	✓	2200
Lee2020 [34]	Chirp	18.5 – 19.5	✓	16	Dhwani [45]	OFDM + PSK	6 – 7	✗	2400
Hornych et al. [20]	ASK, FSK	18 – 20	✓	33	Hush [47]	OFDM + QAM	17.5 – 21	✓	4900
TagScreen [36]	Chirp	18 – 20	✓	36	He et al. [19]	AM, FSK	0 – 22	✗	6000
Haus et al. [18]	OFDM + ASK	22 – 24	✓	64	Yamamoto et al. [69]	16-QAM	?	?	32000
Miegel et al. [43]	OOK, FSK	16 – 20	✓	84	Arentz et al. [1]	ASK	20 – 23	✓	?
Rustle [24]	custom ¹	0 – 15	✗	90	Bits-over-Sound [29]	DBPSK	?	✗	?
Nearby [13]	DSSS + MFSK	18.5 – 20	✓	95	Jeon et al. [22]	PSK	20	✓	?
Bang et al. [3]	FSK	18 – 22	✓	170	No Entry [32]	Chirp	?	?	?
Sadeq et al. [54]	FSK	18 – 19	✓	440	Sasano et al. [56]	OFDM + PSK	6.4 – 8	✗	?
HRCSS [4]	Chirp	18 – 22	✓	500	SilentInformer [57]	FSK	17 – 19	✓	?
Dolphin [65, 73]	OFDM + ASK	8 – 20	✗	500					

¹ Rustle’s custom approach is called “furtive modulation”.

5 Obtaining acoustic data transmission implementations

To independently evaluate existing acoustic communication systems and embed them in IoT testbeds, we first needed to obtain functional implementations. We applied three different methods to acquire the implementations (see Section 7.1 for a discussion of the associated challenges).

5.1 Publicly available implementations

None of the 31 papers surveyed referenced a publicly available implementation.

5.2 Author requests

We contacted every author via email as shown in Figure 2. Only six replied and just three of those responses provided a functional implementation:

- **Digital Voices**, proposed by Lopes et al. [38] in 2001, is among the earliest documented aerial acoustic communication schemes. It uses binary amplitude-shift keying (ASK) across eight carrier frequencies. The authors provided their Java source code.
- **HRCSS**, introduced by Cai et al. [4], is a chirp-based communication scheme that uses multiple chirp carriers simultaneously. The MATLAB source code provided to us lacks their bidirectional probing feature, so we only evaluate the data transmission.
- **Gonçalves et al.** [14] proposed a scheme using orthogonal frequency-division multiplexing (OFDM) and phase-shift keying (PSK) on each subcarrier, incorporating a short chirp sequence for synchronization. The authors provided us with the MATLAB source code.

A fourth response (Hush [47]) was unusable on current Android versions.³

³The authors of Hush [47] shared their Android implementation, but we encountered fatal runtime errors that hindered signal transmission and recording. These issues were likely due to changes in the Android ecosystem and incomplete code.

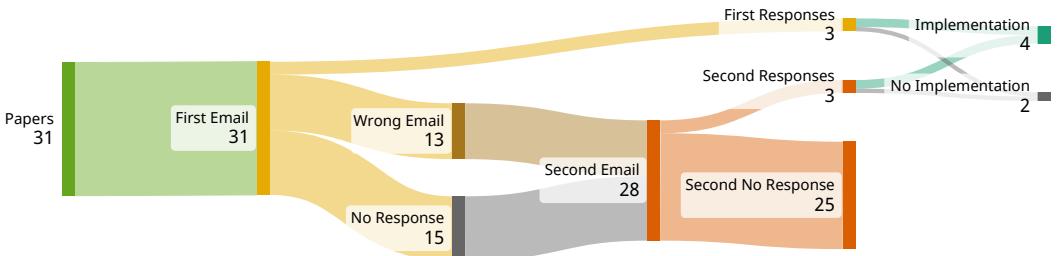


Fig. 2. **Responses from authors** when we requested implementations for 31 publications on acoustic data transmission. We contacted the first authors and all co-authors. For 13 papers, the provided email addresses were outdated, prompting us to search for current contact details manually. We followed up if there was no reply after three weeks. Of the four implementations received, we evaluated three due to runtime errors in the fourth. The other responding authors reported no longer having access to the implementations, citing reasons such as the lead author departing the university or hardware failures.

5.3 Re-implementing acoustic data transmission schemes

We re-implemented the schemes from three highly cited papers, chosen to represent our three target use cases, each with distinct throughput and maximum transmission distances:

- **Lee et al.** [33] claim to be the first to create a prototype that uses chirps in practical acoustic communication, targeting the far-distance use case. In their system, each communication frame begins with a long chirp preamble, followed by 16 individual chirps.
- **Nearby** was proposed by Getreuer et al. [13] and has been integrated into several Google products, including Google Nearby. It is tailored for the medium-distance use case and employs a combination of multiple frequency-shift keying (MFSK), direct-sequence spread spectrum (DSSS), and single-sideband modulation (SSB).
- **PriWhisper**, introduced by Zhang et al. [70], is an acoustic communication system for the nearby use case, akin to NFC. It utilizes MFSK and includes friendly jamming for physical-layer security. However, in our study, we focus solely on the communication capabilities of PriWhisper and have not implemented its security features.

We validated our re-implementations using simulations and practical experiments, ensuring that they perform similar to their original evaluation results. The system proposed by Lee et al. [33] is thoroughly detailed, enabling us to implement all aspects as described in their paper. The other two required filling in undocumented details (see [Appendix B](#)). We publish our MATLAB re-implementations as part of our replication package [51].

5.4 Summary

We ultimately obtained a subtotal of six academic implementations for state-of-the art acoustic communication systems. Additionally, we included two non-academic implementations for comparison from the open-source project `ggwave`,⁴ in both its audible (`ggwavea`) and inaudible (`ggwavei`) configurations. Thus eight schemes advance to evaluation ([Table 2](#)).

The project has not been updated since 2017 and the authors acknowledged in their correspondence that they are aware of the lack of maintenance. Despite this, we still commend Novak et al. [47] for making their source code available.

⁴Most-starred “data-over-sound” project on GitHub: <https://github.com/topics/data-over-sound> (as of 2025-03-03).

Table 2. **Evaluated acoustic communication schemes**, sorted by throughput. The first column specifies the source of the implementation: P – referenced in the paper; A – by contacting the authors; R – by re-implementing them ourselves; N – non-academic solution for comparison. In the last three columns, we classify each scheme’s a priori suitability for the three use case categories defined in Section 2.1, based on their specified throughput and maximum distance.

Via	Name	Modulation	Frequencies	Inaudible	Throughput ¹	Near	Medium	Far
R	Lee et al. [33]	Chirp BOK	19.5–22 kHz	✓	15 bps	✗	✗	✓
A	Digital Voices [38]	MFSK	1–3 kHz	✗	80 bps	✗	✓	– ²
R	Nearby [13]	DSSS + MFSK	18.5–20 kHz	✓	84 bps	✗	✓	✓
N	ggwave _a [12]	MFSK	1.8–6.3 kHz	✗	268 bps	✗	✓	– ²
N	ggwave _i [12]	MFSK	15–19.5 kHz	✓	268 bps	✗	✓	– ²
A	HRCSS [4]	OCSS	18–22 kHz	✓	500 bps	✓	✓	✓
A	Gonçalves et al. [14]	OFDM + QPSK	0.1–4 kHz	✗	600 bps	✓	✗	✗
R	PriWhisper [70]	MFSK	9–17 kHz	✗	729 bps	✓	✗	✗

¹ We list the net data rate, as this is the usable data rate after error correction from the user’s perspective.

² Maximum distance not specified, therefore no a priori assessment.

6 Evaluation

In this section, we systematically evaluate the generalizability and practical suitability of acoustic data transmission across various use cases. For instance, indoor IoT communication [18] requires reliable transmission over distances of several meters, whereas mobile payment systems [8, 63] operate effectively over just a few centimeters. When integrated into smartphone applications (e.g., Google’s Nearby system [13]), acoustic data transmission must accommodate diverse smartphone models with varying audio hardware and frequency responses. Furthermore, transmissions occur under different environmental conditions and must remain robust against various background noises. Our evaluation addresses these considerations systematically.

6.1 Method

We conducted four experiments in which we varied the distance (Section 6.3), device models (Section 6.4), noise interference (Section 6.5), and transmission environment (Section 6.6). We treated each implementation as a black box according to our system model (Figure 1), using it as designed by the authors with their specified parameters.⁵ Our goal was to evaluate each scheme as a complete communication system from a *link layer* (layer two) perspective, including framing, synchronization, and error correction.

We randomly chose the data b_{tx} for transmission. The transmitter converted this data into audio and could utilize the 0–24 kHz baseband freely. Each WAV file was normalized to a specific maximum amplitude before transmitting it from one smartphone to another. The received audio was recorded into a WAV file and processed by the receiver implementation, which demodulated the signal to output data b_{rx} . We then verified whether b_{rx} matches b_{tx} . This procedure allowed us to treat each scheme as a black box, capable of transmitting data from one device to another. Each measurement was repeated at least 20 times for each scheme [22], while the scheme from Lee et al. [33] was repeated 100 times due to its smaller transmission duration. We conducted additional measurements in case of measurement errors. Our experimental setup is shown in Figure 3 and is

⁵ Most of the original papers only present one set of PHY parameters. There are two exceptions: (1) PriWhisper [70], which varied the number of frequencies in an M-FSK scheme—we used the highest number to match their claimed throughput; and (2) DigitalVoices [38], which varied PHY parameters solely to assess sound pleasantness, which is outside our scope.

described in detail in [Appendix A](#) to provide further transparency, including full hardware and software configurations.

6.1.1 Metrics. To assess reliability, we introduce the total error rate (TER), a link-layer metric that combines elements of the bit error rate (BER) and the packet error rate (PER), allowing more direct comparisons across different schemes. We also provide the underlying raw BER data in the supplementary material. Under our system model ([Figure 1](#)), the TER measures the final error rate of received data bits *after* error correction by comparing b_{rx} with b_{tx} . If the receiver returns an error (e.g., from synchronization failures) instead of data, we treat the transmission as having a 100% TER.⁶ This classification treats such transmissions equal to transmissions where every bit was decoded incorrectly, as both instances represent completely unusable transmissions from the user’s perspective. Previous work on acoustic data transmission often reports only BER [14, 22, 57] or PER [13, 70], but each has drawbacks. BER overlooks transmissions that fail to decode, allowing a possible 0% BER even when most transmissions are undecodable, and PER treats a single bit error the same as a thousand bit errors. By combining both metrics, TER better reflects real-world reliability, accounting for individual bit errors as well as complete transmission failures.⁷

6.1.2 Parameters. We conducted four experiments in which we systematically varied the following parameters to determine their impact on reliability. These parameters are the most commonly evaluated ones in prior research using real devices [13, 14, 33, 70].

- **Distance** ([Section 6.3](#)). The transmission distance d significantly affects suitability for our designated use case categories. For instance, a scheme failing to transmit reliably over multiple meters is unsuitable for the far-distance use case. We set a default distance of $d = 50$ cm, typical for medium-distance use cases.
- **Device models** ([Section 6.4](#)). The transmitter D_{TX} and receiver D_{RX} influence transmission reliability, as the audio hardware in smart devices is not designed for data transmission. Each device has a different frequency response, particularly at near-ultrasonic frequencies [21]. We selected the Google Pixel 4a smartphone as the default transmitter D_{TX} due to its relatively low distortion at a volume setting⁸ of $V_{TX} = 19/25$, which we experimentally identified to be a good trade-off between high volume and low distortions due to amplifier and speaker non-linearities. The default receiver D_{RX} was a Samsung Galaxy S20 Ultra smartphone, chosen for its low self-noise.
- **Noise interference** ([Section 6.5](#)). Practical transmissions often occur under suboptimal conditions, such as loud ambient noise or interference from how users handle their smartphones. By default, our testing occurred in a quiet environment without external noise.
- **textbfEnvironment** ([Section 6.6](#)). The environment influences the multipath profile of the acoustic channel. By default, we evaluated the schemes in a typical indoor setting using quiet office rooms and a hallway for long-distance measurements. Because these environmental

⁶Our measurements show that error rates can exceed 50% when significant portions of the received message are missing. To correctly penalize complete failures—when no message is received—we assign 100% TER.

⁷In summary, the TER is equal to the BER unless the receiver returns an error instead of data, in which case the TER is 100%. Illustrative examples (assuming 100 data bits per frame):

- (1) Receiving 99 correct bits yields a TER of 1% (identical to the BER).
- (2) Failing to decode entirely yields a TER of 100% (where the BER is undefined).
- (3) Receiving all bits incorrectly also yields a TER of 100% (identical to the BER).
- (4) Across 50 transmissions in case (1) and 50 in case (2), the average TER is 50.5%, whereas BER is misleadingly low at 1%, and PER is misleadingly high at 100%.

⁸On smartphones, the volume can typically only be set to one of a few discrete volume indices, where the maximum index varies per device.

conditions cannot be replicated exactly, we also conducted tests in an anechoic chamber (Figure 3) without reverberation or ambient noise, offering a controlled and reproducible testing environment. Finally, we also tested a large lecture room with mechanical ventilation. The background noise levels averaged 53.60 dB SPL in the office, 60.01 dB SPL in the lecture room, and 27.65 dB SPL in the anechoic chamber (all unweighted RMS).

6.2 Preliminary testing

6.2.1 *Setup.* Before extensive smartphone-based tests, we performed a preliminary evaluation under near-optimal conditions to confirm each implementation’s functionality. Using high-quality audio equipment, each scheme was transmitted by a Neumann KH 80 DSP studio speaker and recorded by an Earthworks M23R reference microphone, positioned on-axis at 50 cm in an office room (see Section A.4.1 for details).

6.2.2 *Results.* The scheme by Lee et al., DigitalVoices, Nearby, PriWhisper, and both ggwave variants achieved a perfect 0% TER (Figure 4). The scheme from Gonçalves et al. was mostly accurate but showed an error rate of 0.8%, likely due to the scheme being designed for closer distances. However, the HRCSS implementation produced extremely high TER, which was unexpected given the high-quality equipment. We unfortunately had to conclude that this implementation is unsuitable for practical tests and therefore excluded it from further testing (see discussion in Section 7.4.3).

6.3 Impact of different distances

6.3.1 *Setup.* Next, we evaluated each scheme over distances from 5 cm to 40 m. Tests were conducted in both a quiet indoor setting (representative of everyday use) and an anechoic chamber with minimal reverberation and noise. In both environments, we measured distances up to 5 m; for distances beyond 5 m (up to 40 m), we used the hallway adjacent to the office.

6.3.2 *Results.* In the *anechoic chamber* (Figure 5a), most schemes achieved low error rates at short distances. The scheme from Lee et al. maintained a zero error rate across all tested distances. Both ggwave schemes performed nearly perfectly, with slight exceptions at 2 m and 5 m. Nearby also excelled, aside from a small outlier at 50 cm. Gonçalves et al.’s scheme kept error rates under 0.5% for distances below 10 cm, but rose to around 13% at longer distances. DigitalVoices and PriWhisper stayed below 1% at close range, then deteriorated notably beyond 20 cm.

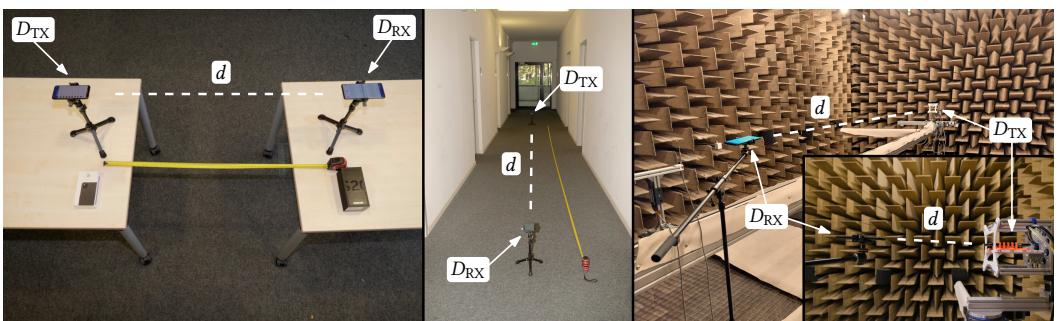


Fig. 3. **Experimental setup.** The left picture shows our office setup, with device D_{TX} transmitting to device D_{RX} at distance d . The middle picture shows our setup in the office hallway for larger distances from 5 m to 40 m. The right pictures show our setup in the anechoic chamber at two different distances.

In the *office environment* (Figure 5b), error rates increased noticeably due to stronger multipath effects and ambient noise. DigitalVoices, PriWhisper, Nearby, and Gonçalves et al.’s scheme all showed a strong distance dependency, degrading over longer ranges. At short distances up to 10 cm, however, all schemes maintained a TER under 0.5% except DigitalVoices. Although DigitalVoices performed worse at close range than in the anechoic chamber, it surprisingly improved between 50 – 500 cm, showing that the DigitalVoices receiver is not yet fully optimized. We investigated the DigitalVoices discrepancy and observed that post-processing and amplifying the recordings significantly reduced its error rates in the anechoic environment. However, we did not include post-processed results in our plots to compare the schemes as described in their publications. Nearby performed well within 1 m, but became unreliable at larger distances, including an outlier at 10 m, likely due to the transition into the hallway. This non-monotonic distance behavior is consistent with Getreuer et al.’s original Nearby evaluation [13, Fig. 13], which attributes such variability to nontrivial multipath behavior. Lee et al.’s scheme performed exceptionally across all distances, with only a minor outlier at 5 m. The inaudible ggwave variant remained effective up to 20 m, whereas the audible variant only worked up to 1 m before failing to decode.

6.4 Impact of different device models

6.4.1 Setup. This experiment assessed how well acoustic data transmission schemes perform across a broad set of smartphones in a typical indoor environment. We selected five smartphones representing a broad spectrum of release dates, prices, and manufacturers (Table 3). We tested each scheme using 21 different transmitter-receiver combinations. Our setup involved one smartphone as the transmitter, positioned 50 cm away on-axis from two to three simultaneous receivers to make the measurement process more efficient. As we had two Pixel 4a smartphones, we also evaluated performance with identical transmitter and receiver models.

6.4.2 Results. As shown in Figure 6, Lee et al.’s scheme and both ggwave variants consistently exhibited very low error rates across all device combinations. DigitalVoices showed varying performances: the best results occurred with the Pixel 6 Pro transmitting to the Nexus 6P (TER \approx 12%), while the Pixel 4a and the Pixel 6 Pro largely failed as receivers. Nearby generally performed well, except when using the Samsung S20 as the transmitter and for some transmissions of the Pixel 4a. PriWhisper was effective in certain combinations, notably with the Pixel 6 Pro as the transmitter (TER as low as 3%), but failed in others due to synchronization issues. Its poor overall performance

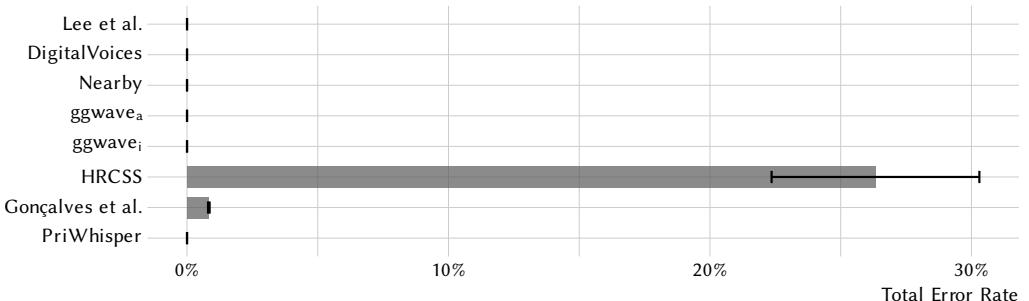


Fig. 4. Preliminary testing in a best-case hardware scenario. Shorter bars are better. Schemes are sorted by throughput (see Table 2). The error bars show the standard error. Parameters: $d = 50$ cm; D_{TX} = loudspeaker Neumann KH 80 DSP; D_{RX} = microphone Earthworks M23R; $V_{TX} = -40$ dB interface gain; quiet office environment; $N=240$ measurements.

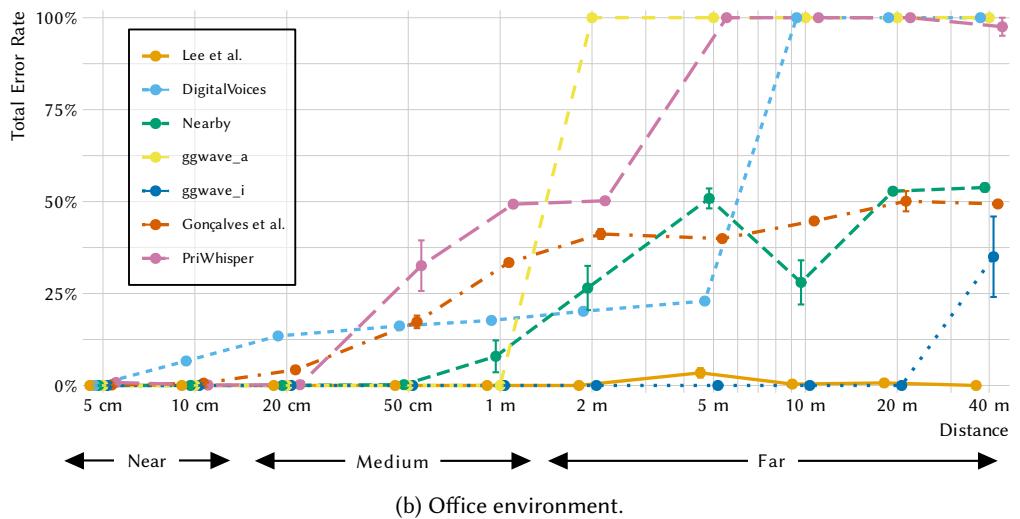
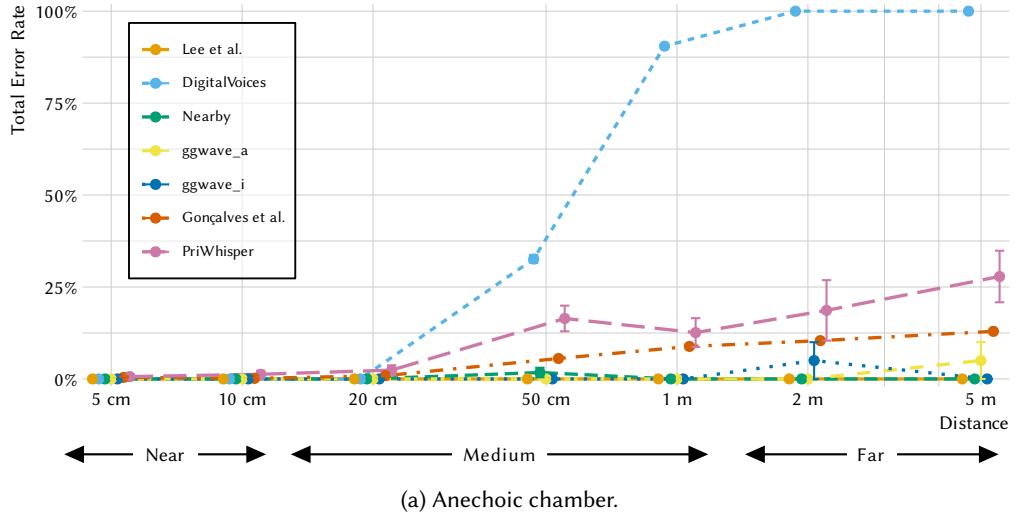


Fig. 5. Reliability of acoustic transmissions with varying distances. Smaller TER is better. The anechoic chamber allows for measurements up to a maximum distance of 5 m. In the office environment, we were able to measure up to a distance of 40 m, but the measurements at 10 – 40 m were conducted in the hallway. The distance axis is logarithmic, but we only measured at the indicated discrete distances. The error bars show the standard error. The data points of different schemes are dodged to the side to avoid overlap. Parameters: D_{TX} = smartphone Pixel 4a; D_{RX} = smartphone Samsung S20 Ultra; V_{TX} = volume index 19/25; anechoic chamber (Figure 5a) and quiet office environment (Figure 5b); $N=3740$ measurements.

can be attributed to the 50 cm device distance, which highlights its limitations outside its intended usage range of a few cm. In this case, the impact of different device models on the reliability varied significantly, ranging from satisfactory to completely ineffective. The performance of Gonçalves et al.'s scheme was more consistent, with error rates from 5% (best case: Oppo Reno 6 to Pixel 4a) to 32% (worst case: Pixel 4a to Nexus 6P).

Table 3. **Smartphones in the device models experiment** in Section 6.4. The volume column indicates the selected and maximum volume indices for each device. We set each device to a volume of ca. 75% of its maximum volume, which we experimentally identified to be a good trade-off between achieving high volume while maintaining low distortions due to amplifier and speaker non-linearities.

Device Name	Origin	Release Date	Price	Volume
Google Pixel 4a	USA	Oct 2020	349€	19/25
Google Pixel 6 Pro	USA	Oct 2021	900€	19/25
Huawei Nexus 6P	China	Sep 2015	700€	11/15
Oppo Reno 6	China	Sep 2021	499€	12/16
Samsung Galaxy S20 Ultra	South Korea	Feb 2020	1300€	11/15

Overall, the Pixel 6 Pro as the transmitter consistently resulted in the lowest error rates. The Samsung Galaxy S20 Ultra generally performed well as a receiver, except with certain transmitters for DigitalVoices and PriWhisper. This highlights a large diversity in device characteristics, which complicates the generalization of some schemes.

6.5 Impact of different types of noise and device handling

6.5.1 *Setup.* The preceding tests were conducted in relatively quiet environments, but practical scenarios often contain different types of interference, which can degrade the reliability of acoustic transmissions either due to the ambiance and the surroundings, or due to human handling of the devices. In this experiment, we aimed to assess the influence of different kinds of noises and handling on the receiver. We used the Pixel 4a as the transmitter and the Samsung Galaxy S20 Ultra as the receiver, at a distance of 50 cm aligned on-axis in a quiet indoor environment. We performed experiments for each of the following seven types of noise and handling, designed to mimic real-world disturbances:

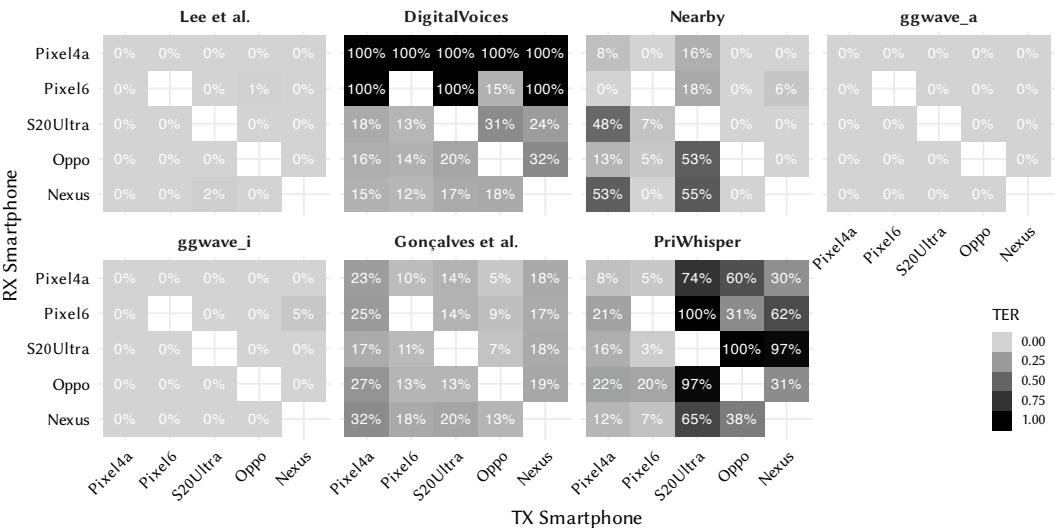


Fig. 6. **Reliability of acoustic transmissions for different device combinations.** Smaller TER is better. Schemes are sorted by throughput. The heatmap shows each scheme's reliability for 21 different TX-RX combinations. Parameters: $d = 50$ cm; $D_{TX}/D_{RX}/V_{TX}$ = see Table 3; office environment; $N=4620$ measurements.

- **Ambient.** Playing ambient field recordings from a café, a train station, and a market place on a Neumann loudspeaker 1 m away from the receiver. The experiment was conducted separately for all three ambient sounds (see [Section A.4.2](#) for detailed setup information).
- **Circle movement.** Shuffling the receiver in a circular motion on the table, which varies the distance by ± 5 cm.
- **Clapping.** Clapping hands near the receiver every 2 s.
- **Pocket.** Inserting the receiver into and removing it from a trouser pocket every 2 s, aligned so that the transmitter's speaker faces the receiver in the pocket.
- **Up and down.** Lifting and firmly placing the receiver on the table approximately every 2 s.
- **Opposite.** The smartphones are positioned with their microphones and speakers facing opposite directions. This arrangement simulates two people holding phones while facing each other.
- **Side-by-side.** The smartphones are placed next to each other, with their long edges adjacent.
- **Handheld.** The transmitter smartphone is held by a person, introducing the natural movement of typical handheld use.

6.5.2 *Results.* All schemes experienced significant performance degradation when the smartphone was pocketed during transmission ([Figure 7](#)). However, *Nearby*, the inaudible *ggwave* variant, and Lee et al.'s scheme were robust against all other noise types tested. The audible *ggwave* variant was affected by ambient noise, which made it completely unusable, likely because the ambient noise interfered with the transmission bandwidth. Clapping dramatically impaired *DigitalVoices*, *PriWhisper*, and Gonçalves et al.'s scheme, rendering them largely ineffective. Noise interactions with the table, whether through circular movement or repetitive lifting and placing, had a relatively minor impact on performance. Notably, *DigitalVoices* struggled significantly with clapping and pocketing noises ($TER > 50\%$), yet coped better with table-related movements ($TER \approx 6\%$). The side-by-side orientation was more detrimental to reliability than the opposite orientation, although not all schemes were affected.

6.6 Impact of different environments

6.6.1 *Setup.* To determine the influence of the environment and its multipath profile, we measured transmissions between a Pixel 4a (transmitter) and a Samsung Galaxy S20 Ultra (receiver) aligned on-axis at 50 cm. The evaluation took place in several locations: small and large offices, a meeting room, and a lecture room. In the lecture room, we evaluated both a static setup and a dynamic one in which people walked quietly throughout the room while maintaining an unobstructed line of sight between the devices, dynamically changing the multipath environment (see [Section A.4.3](#)).

6.6.2 *Results.* As shown in [Figure 8](#), Lee et al.'s scheme, *Nearby*, and both *ggwave* variants achieved perfect reliability in all tested environments. The remaining schemes performed worst in the offices, but showed improvement in the meeting and lecture rooms. Interestingly, quiet movements in the vicinity do not seem to negatively impact reliability.

6.7 Summary for each use case

We present a summary of our previous measurement results by scheme and use case in [Figure 9](#). This plot shows how each scheme performs for its intended use case (as per [Table 2](#)) in typical indoor environments, therefore we excluded measurements in the anechoic chamber and with the high-quality studio equipment, as they do not represent typical usage. We discuss these results in more detail in [Section 7.2](#).

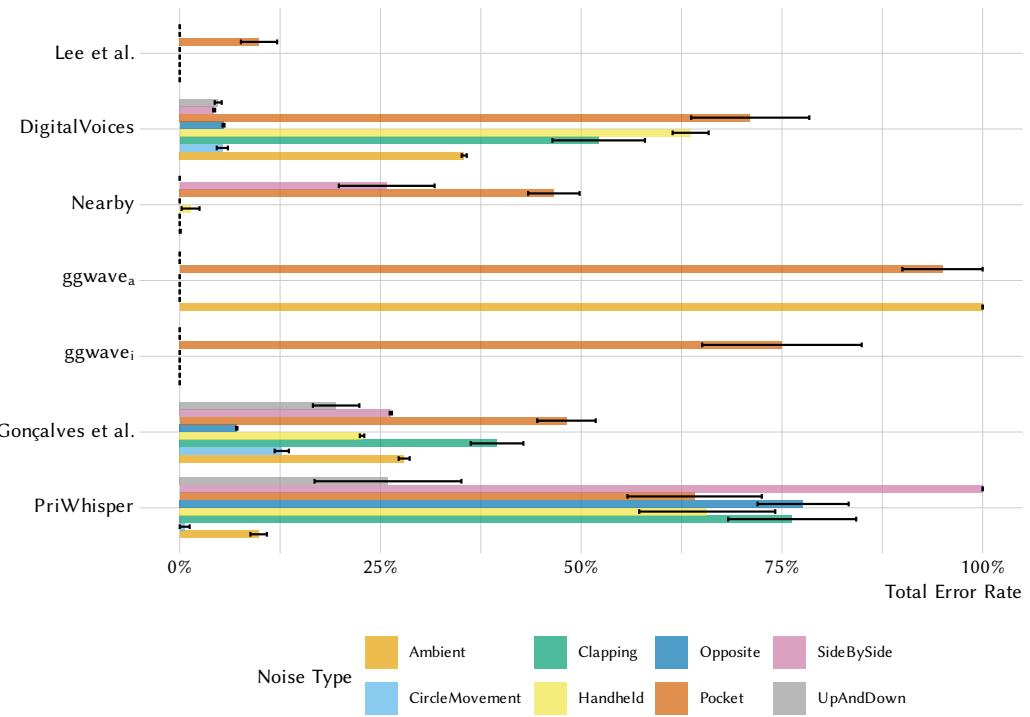


Fig. 7. Reliability for different types of noise disturbing the transmission. Smaller bars are better. Schemes are sorted by throughput. For the ambient noise, we performed three separate experiments with different field recordings. The error bars show the standard error. Parameters: $d = 50$ cm; D_{TX} = smartphone Pixel 4a; D_{RX} = smartphone Samsung S20 Ultra; V_{TX} = volume index 19/25; office environment with different types of controlled noise (see Section 6.5); N=2200 measurements.

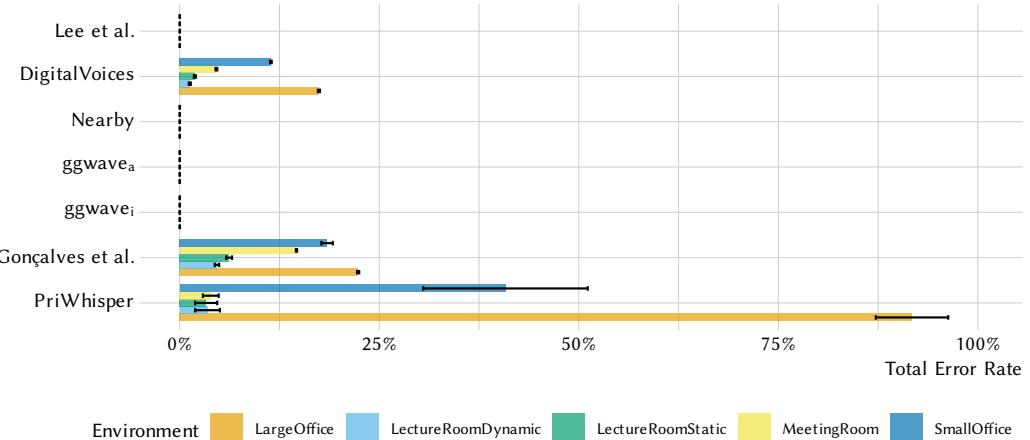


Fig. 8. Reliability in different transmission environments. Smaller bars are better. Schemes are sorted by throughput. The error bars show the standard error. Parameters: $d = 50$ cm; D_{TX} = smartphone Pixel 4a; D_{RX} = smartphone Samsung S20 Ultra; V_{TX} = volume index 19/25; N=1100 measurements.

6.7.1 Outliers. For the inaudible ggwave variant, one transmission failed during the medium-distance use case ($d = 50$ cm, $D_{TX} = \text{Nexus 6P}$, $D_{RX} = \text{Pixel 6 Pro}$), and some transmissions failed for the far-distance use case ($d = 40$ m, $D_{TX} = \text{Pixel 4a}$, $D_{RX} = \text{Samsung S20 Ultra}$). The outliers for the scheme by Lee et al. were mostly at a distance of 5 m as shown in Figure 5b. Nearby performed well in the medium-distance use case (TER median = 0%), except for a few anomalies in specific device combinations (e.g., $D_{TX} = \text{Samsung S20 Ultra}$, $D_{RX} = \text{Nexus 6P}$).

7 Discussion

7.1 Obtaining software implementations is challenging

To answer our first research question (**RQ1**) of how researchers can obtain working implementations of acoustic data transmission systems for evaluation, we conducted a systematic literature study and documented our experience in trying different acquisition methods (Section 5). Out of the 31 publications identified in our survey, none referenced a publicly available implementation, and only four authors provided their implementation upon request. This lack of accessible research artifacts severely hinders replication of research in the field of acoustic data transmission, thereby slowing scientific progress. Without straightforward access to verify and build on existing work, demonstrating improvements over the state of the art becomes significantly more challenging.

We also attempted to re-implement the schemes based on the description in the paper, a process that is both time-consuming and rarely undertaken by other researchers. During this effort, we encountered several inaccuracies and omissions in the original publications, which often required educated guesses to resolve (Appendix B). This challenge is not unique to our field but has been documented in other areas of computer science as well [7]. Such re-implementations carry a

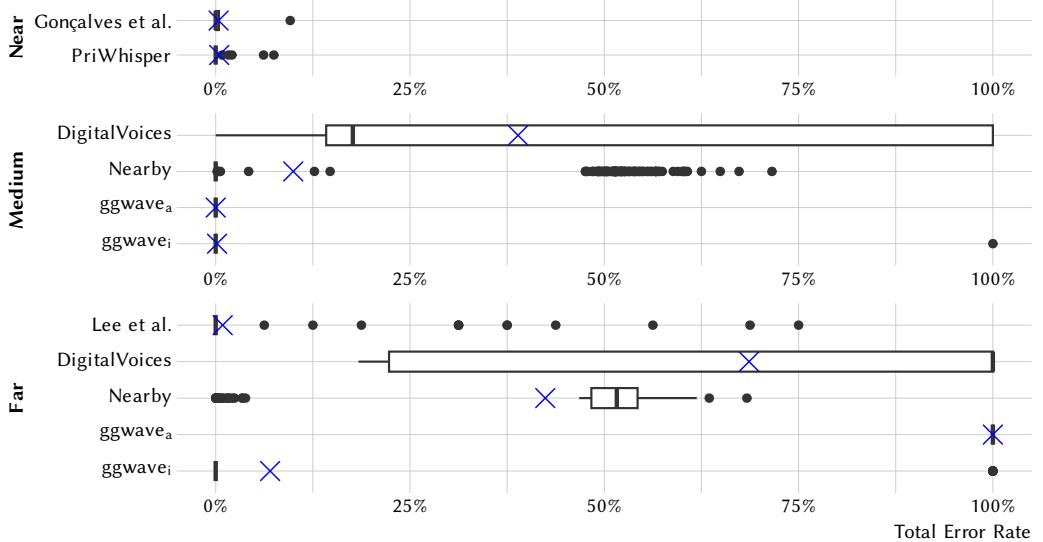


Fig. 9. Summary of our measurement results from a quiet indoor environment (Section 6.3, Section 6.4, and Section 6.6), for the three use case categories (defined in Section 2.1 and mapped in Table 2). Smaller TER is better. Schemes are sorted by throughput in each category. The box plots display the median (black bar), mean (blue cross), and outliers (black circles), with hinges marking to the 25th and 75th percentiles. Parameters: d = variable (according to the scenarios in Section 2.1); $D_{TX}/D_{RX}/V_{TX}$ = variable according to Table 3; quiet office environment; N=3300 measurements.

Table 4. Suitability of each scheme for different use case categories (RQ2). Schemes are sorted by throughput. A scheme is considered suitable (✓) if its mean TER is approximately 0%. Otherwise, it is unsuitable (✗). When only a few outliers appear and the median TER remains at 0%, the scheme may still be suitable but requires additional modifications (e.g., error correction). The highlighted entries indicate novel results that changed compared to prior knowledge (Table 2). The first column indicates how we obtained the implementation: P – referenced in the paper; A – by contacting the authors; R – by re-implementing them ourselves; N – non-academic solution for comparison.

Via	Name	Frequencies	Bandwidth	Inaudible	Throughput ¹	Near	Medium	Far
R	Lee et al. [33]	19.5–22 kHz	2.5 kHz	✓	15 bps	✗	✗	✓
A	Digital Voices [38]	1–3 kHz	2 kHz	✗	80 bps	✗	✗	✗
R	Nearby [13]	18.5–20 kHz	1.5 kHz	✓	84 bps	✗	✓ ²	✗
N	ggwave _a [12]	1.8–6.3 kHz	4.5 kHz	✗	268 bps	✗	✓	✗
N	ggwave _f [12]	15–19.5 kHz	4.5 kHz	✓	268 bps	✗	✓	✓
A	HRCSS [4]	18–22 kHz	4 kHz	✓	500 bps	✗	✗	✗
A	Gonçalves et al. [14]	0.1–4 kHz	3.9 kHz	✗	600 bps	✓ ²	✗	✗
R	PriWhisper [70]	9–17 kHz	8 kHz	✗	729 bps	✓ ²	✗	✗

¹ We list the net data rate, as this is the usable data rate after error correction from the user's perspective.

² Requires further error correction.

high risk of errors or deviations from the original, which is frustrating given the significant time investment. To improve replicability and transparency in research, we encourage other researchers to follow our suggestions in Section 7.5 and publish their software artifacts.

7.2 Generalizability towards ad-hoc smartphone communication use cases

To assess the practicality of the acoustic data transmission schemes proposed in previous studies (RQ2), we evaluated how well these schemes generalize towards the three use case categories defined in Section 2.1. For each use case category, we first reviewed the original publications of each scheme to identify their a priori suitability based on their specified throughput and maximum distance. For example, the scheme by Lee et al. only has a throughput of 15 bps, so by design it is not suitable for transmission of public keys or small files at near distances, as this would take far too long. In many cases, our experiments revealed limited practical suitability of the schemes, which was surprising given the claims in their original publications. Table 4 shows our a posteriori assessment of each scheme based on our evaluation. In particular, all highlighted results in the table represent novel insights which were previously not known based on the limited testing in the current literature. This discrepancy shows the importance of extensive testing across a range of distances and devices to demonstrate practical suitability.

For *far distances* (up to 40 m), the only reliable schemes were the inaudible growave variant, effective up to 20 m, and the scheme by Lee et al., which maintained reliability up to 40 m. All other schemes failed at these distances. At *medium distances*, both growave variants were effective, with the inaudible one being particularly robust. The Nearby scheme also performed well but struggled to generalize over all tested device combinations. The other schemes are not suited for this use case, either because of high error rates or because their throughput is not high enough. For the *near-distance* use case, only PriWhisper and the scheme by Gonçalves et al. had sufficient throughput, while maintaining low error rates. However, they did experience some erroneous packets, indicating a need for better or additional error correction techniques.

None of our analyzed schemes are suitable for all use cases (see [Table 4](#)), as different applications prioritize different trade-offs between reliability, throughput, latency, and audibility. Some schemes offer flexibility; for instance, ggwave can be configured for higher throughput at the cost of reliability or shifted to the near-ultrasonic range to achieve inaudibility.

For applications like secure pairing or mobile payments, where data integrity is paramount, a low TER is the most critical requirement, while high throughput is secondary. The scheme from Lee et al. and PriWhisper are well-suited for this context, although PriWhisper may benefit from additional error correction to guarantee performance in challenging environments. In contrast, use cases such as multimedia applications and content delivery (e.g., second-screen services, location-based information) often prioritize high throughput and inaudibility to ensure a smooth user experience, and can typically tolerate minor, correctable errors. The inaudible ggwave variant is a candidate here, though it cannot reach data rates higher than 268 bps. Notably, schemes designed for higher data rates generally struggled to perform reliably across varied indoor environments.

Given that high throughput generally decreases reliability, a robust design pattern for many practical applications is to treat the acoustic channel as a low-bandwidth trigger: a short, reliable acoustic message (e.g., a unique identifier or hash) can initiate a connection over a high-throughput channel like Wi-Fi or cellular to transfer the bulk of the data.

7.3 Practical challenges for acoustic data transmission

Our measurements show that acoustic data transmission on commodity smartphones and IoT devices face challenges that rarely appear in simulations or controlled lab settings (**RQ3**). Because audio hardware and propagation channels differ widely across devices and environments, there is no single transmission specification. Rather than prescribing a new modem architecture, we therefore summarize the *design requirements* that any future scheme should respect for real-world applicability.

7.3.1 Severe multipath propagation. Reverberation produces long delay spreads due to multipath propagation at the relatively slow speed of 343 m/s. Because this delay spread depends on the environment and geometry, a scheme that performs well in one setting may behave differently in another. We observed that even in the same environment, device position and orientation can significantly affect reliability, leading to considerable error rate variability. These findings support the observation by Getreuer et al. that acoustic transmission reliability is sensitive to environmental conditions [13, section 8A]. **Design requirement:** Operate correctly with delay spreads up to tens of milliseconds [33, 59] and the resulting inter-symbol interference (e.g., using guard intervals).

7.3.2 Device heterogeneity. Speaker and microphone responses, amplifier gains, and nonlinearities vary widely: our tests reveal up to 30 dB SPL spread between phones at the same volume setting. Prior research often considered only a specific device model, limiting its practical generalizability. Most acoustic links are unidirectional, so no feedback-based gain control is possible. Developers can only adjust volume relative to each device’s maximum output, and our tests show that most devices suffer from nonlinear distortions at volume settings above approximately 75%, though exact thresholds vary. **Design requirement:** Withstand at least 30 dB SNR fluctuation and moderate nonlinear distortion without closed-loop calibration (e.g., using normalization and robust modulation).

7.3.3 Limited bandwidth. IoT audio hardware typically employs ADCs and DACs with sampling rates of 44.1 kHz or 48 kHz, which limits the practical bandwidth to less than 22 kHz. When transmissions must be inaudible, the usable bandwidth reduces to less than 4 kHz, and audio hardware is highly frequency selective in this band [21]. **Design requirement:** When choosing the near-ultrasonic band for inaudibility, prepare for strong frequency-selectivity and limited throughput.

7.3.4 *Ambient noise.* In contrast to RF communication, the acoustic frequency band is exposed to a wide range of common noise sources—such as traffic, machinery, human speech, and noises from device handling—that occupy a significant share of the available spectrum. Both burst and stationary noise components are common. **Design requirement:** Maintain link integrity in the presence of continuous and impulsive noise (e.g., using strong error correction and time interleaving).

7.3.5 *Throughput vs reliability.* Some of our surveyed publications in Table 1 pursue very high throughput, but our evaluation shows that increased throughput typically comes at the expense of higher error rates. Commercial systems often target ≤ 200 bps [9, 13, 27, 62], likely because of reliability concerns in practice. **Design requirement:** Favor robustness and aggressive error control unless bi-directional link adaptation is feasible (e.g., HRCSS-style probing), which is often impossible in one-way IoT scenarios.

7.4 Comparison with original evaluations

Several schemes we tested were also evaluated by their original authors. Although our experimental setup differs intentionally to assess generalizability, we compare our results to determine whether the original findings are *conceptually* replicable [61]. Overall, our results align with the original evaluations—considering that deviations are expected due to differences in device characteristics, environments, and our re-implementations. The only exception is HRCSS [4], whose provided implementation yielded impractically high error rates even under optimal conditions. DigitalVoices and ggwave lack prior reliability evaluations.

7.4.1 *PriWhisper.* Zhang et al. [70] conducted a basic reliability evaluation indoors using a Samsung Galaxy S3 and a Google Nexus S, reporting a packet error rate of 0.5%. In our near-distance tests, we observed a similar bit error rate (around 0.5%), though our packet error rate was somewhat higher. Despite these differences, our results are comparable, given that our re-implementation might not be as optimized as the original and that we used different smartphone models.

7.4.2 *Gonçalves et al.* Gonçalves et al. [14] evaluated their scheme indoors on two portable computers, reporting bit error rates of approximately 3% at 10 cm and 25% at 100 cm. Our experiments revealed a similar trend. Although their noise tests (using background music from a nearby loudspeaker) lack detailed parameters, our noise experiments at 50 cm confirmed a comparable degradation in reliability.

7.4.3 *HRCSS.* Cai et al.’s HRCSS [4] initially appeared promising, but the provided MATLAB implementation failed our preliminary tests with high-quality equipment. To rigorously justify its exclusion, we analyzed the provided MATLAB source code in detail. We found that the high error rates stemmed from synchronization failures. In particular, the preamble detection algorithm in the implementation deviates from the method described in the HRCSS paper [4]. Specifically, errors in the code cause the receiver to incorrectly calculate and select signal peaks, preventing it from correctly identifying the start of a transmission. Although we did not attempt to fix the code, we verified synchronization as the root cause by manually supplying the correct preamble start index to the decoder, which substantially improved the TER.

We note that HRCSS occasionally produced excellent results, although its performance was highly inconsistent across distances, volumes, and devices. While their paper mentions a Java implementation used in their evaluations, we only received the MATLAB code, so we were unable to test the Java version. This is particularly unfortunate because their study uniquely benchmarks their scheme against an existing one (by Lee et al. [33]) using real devices.

7.4.4 *Lee et al.* Lee et al. [33] evaluated their scheme on Android devices in indoor far-distance scenarios, achieving packet success rates above 97% at distances of 10 m and 25 m. Our experiments confirmed these high success rates (over 98% at 10 m, 20 m, and 40 m). Additionally, our compatibility tests across 21 device combinations—using smartphones as transmitters at 50 cm—demonstrated excellent robustness and versatility.

7.4.5 *Nearby.* Getreuer et al. [13] tested Nearby with a Nexus 6 transmitter and Nexus 5/5X receivers, reporting packet success rates over 94% up to 2 m, though error rates increased at larger distances. Our tests confirmed low error rates below 1 m and higher errors at 1 m and 2 m, likely due to differences in our re-implementation, device models, and packet sizes. Another difference is that Getreuer et al. employed a form of error correction by repeating each transmission three times and using the redundancy to correct bit errors, but we did not evaluate this as it effectively reduces throughput to a third. The packet error rate, however, improves when using this form of error correction.

7.5 Towards more rigorous and reproducible evaluation

Our evaluation reveals that many existing acoustic data transmission systems struggle in practical scenarios. This highlights a critical need for more rigorous and reproducible evaluation methodologies to properly assess how new schemes generalize to real-world conditions. The challenges in replicating prior work—due to environmental variations and incomplete documentation—have historically limited comparative analysis, with most studies evaluating new schemes in isolation.

To address this, we present a set of guidelines for more rigorous and reproducible evaluation. The black-box analysis framework we detail in Section 6 serves as a practical implementation of these principles. A key advantage of our methodology is its generality; being system-agnostic, it can be applied to any unidirectional acoustic data transmission system, ensuring its relevance for evaluating future schemes. We encourage future publications to adopt this framework.

7.5.1 *Prioritize real-world evaluation over simulation.* While simulations are useful for initial design, final evaluations must be conducted on real devices in scenarios representative of the target use case. Our approach in Section 6 provides a template for how to systematically test the impact of key practical challenges:

- **Distance.** Test at various distances relevant to the intended application.
- **Device heterogeneity.** Use a diverse set of transmitter and receiver devices (e.g., different smartphone models or IoT hardware).
- **Environmental noise.** Assess performance against both ambient noise (using field recordings or white noise) and intermittent burst noise (e.g., from device handling).
- **Multipath effects.** Evaluate in environments of varying sizes and acoustic properties.

7.5.2 *Focus on comprehensive and relevant metrics.* The TER is arguably the most critical metric, as it directly reflects the system’s reliability from the perspective of a developer wanting to integrate acoustic data transmission schemes into their IoT or smartphone projects. Other factors such as audibility and latency do not have to be experimentally determined as they follow directly from the system’s design. In time-critical applications, the additional processing latency during encoding and decoding can also be of importance.

7.5.3 *Ensure replicability through open science practices.* To enable verification and build upon prior work, we advocate for the following best practices [2, 7, 46, 55, 61]:

- **Publish software artifacts.** Release complete implementations (transmitter and receiver code), ideally along with clear run instructions and analysis scripts. None of the papers we

reviewed provided their code, forcing error-prone and time-consuming re-implementations. We contribute our re-implementations of Nearby [13], PriWhisper [70], and Lee et al.'s scheme [33] in our replication package [51].

- **Describe the experimental setup.** Clearly specify all system parameters and experimental conditions (e.g., TX/RX devices, volume, gain, payload length, distance, environment, background noise) and include images of the setup. Define the evaluation metrics precisely. Our setup is detailed in [Appendix A](#).
- **Publish data artifacts.** Share raw measurement data (such as audio recordings) to enable both replication and reproduction of the evaluation [49]. This transparency allows independent verification and enables other researchers to improve upon the work (e.g., by designing better decoders). We publish a dataset of 11 900 recorded audio transmissions from eight schemes along with a CSV file containing raw BER results for each recording [51].

7.5.4 Establish standardized testbeds. For maximum reproducibility, evaluations could be performed in a standardized testbed, such as an anechoic chamber. To simulate realistic conditions, controlled noise sources playing back ambient field recordings or white noise can be introduced.

7.6 Limitations

7.6.1 Implementation variability. Due to a lack of research transparency in the field of acoustic data transmission and the absence of publicly available implementations for proposed schemes, we had to re-implement some schemes ourselves. This process was laborious and prone to errors. Our re-implementations of Nearby [13], PriWhisper [70], and the scheme by Lee et al. [33] may differ from the originals because the publications did not provide full implementation details. In part, we had to rely on educated guesses, which we describe in [Appendix B](#). Furthermore, for our re-implementations of Nearby and PriWhisper, we used larger message lengths than those reported in the original publications to meet the demands of their intended use cases in a single transmission. We verified our re-implementations using simulations and practical experiments, which showed that Nearby and PriWhisper can handle these larger message lengths while maintaining high reliability, making them conceptually replicable under these more challenging conditions. This positive result indicates that our re-implementations did not degrade the performance of the original implementations. During our experiments, all of our re-implementations were generally robust and reliable for their intended use case.

7.6.2 Environmental variability. We conducted our experiments in quiet indoor rooms and a hallway, acknowledging that varying environmental conditions could have influenced our results. To ensure accuracy, we repeated any experiments that yielded inconsistent results or were affected by unintended noise interference. However, one measurement stood out: Nearby performed significantly worse in the device models experiment compared to the distance experiment, despite using the same devices and transmission distance. This was not a measurement error; the only difference was the change in office rooms. This supports the observation by the Nearby authors, who also observed that the reliability of acoustic data transmissions is sensitive to environmental conditions [13, section 8A]. Therefore, given that multipath environments such as office rooms are inherently variable and cannot be precisely replicated by others (see [Section 7.3.1](#)), we also conducted tests in an anechoic chamber, which is a controlled and reproducible testing environment.

7.6.3 Burst noise variability. The burst noise experiment presented several challenges not present in the ambient noise experiment, due to the need for manual handling of the devices. This not only made the process time-consuming—requiring manual effort for 560 transmissions—but also introduced variability in handling, which affected the replicability of the experiment. We attempted

to mitigate this by triggering the burst noise at 2 s intervals, but some variability remains. Note that this setup resulted in less burst noise exposure per transmission for the schemes from Gonçalves et al. [14] and Lee et al. [33], which had shorter transmission durations compared to the other schemes (Gonçalves et al. at 0.4 s, Lee et al. at 1.1 s, others approximately 5 s; all with a 2 s pause between transmissions). Consequently, the results from these two schemes are not directly comparable with those from the others.

8 Future work

Based on our evaluation, we have identified a list of design requirements in [Section 7.3](#), which will allow future schemes to be ready for real-world deployment. In particular, our comparison across multiple smartphones demonstrated that most schemes struggle to adapt to different device combinations, due to variations in audio hardware such as frequency responses of microphones and speakers, amplifier non-linearities, and potential on-chip post-processing. Conducting a comprehensive study of acoustic characteristics across a variety of devices, including low-cost IoT hardware and smart devices, could delineate the operational limits within which schemes must reliably function. Such knowledge could streamline future testing efforts and assist in selecting relevant test devices.

Additionally, we observed that most prior studies utilized varying experimental setups when testing with real devices, complicating comparisons between results. Our work highlights the necessity for increased research transparency, specifically in detailing how evaluations are conducted. Future efforts should aim to establish and define a standard for testing acoustic communication schemes. Our approach, which involved designing a unified experimental setup for multiple schemes, could serve as a foundation ([Appendix A](#)). However, standardizing testing environments remains a challenge, especially outside of controlled settings like anechoic chambers.

9 Conclusion

We conducted a systematic literature study on acoustic data transmission schemes and analyzed their generalizability by requesting implementations from authors, re-implementing selected schemes ourselves, and evaluating them on real devices in both realistic and controlled environments. Our findings reveal that acoustic data transmission schemes often face reliability issues, particularly at higher data rates and across device models. This problem has not been extensively acknowledged or discussed in previous research, which predominantly relied on theoretical simulations or conducted only limited real-device testing. Moving forward, we recommend stronger emphasis on evaluating real-world performance, and we offer our evaluation methodology as a foundation for future work.

RQ1. *“How challenging is it to obtain or re-implement systems proposed in this field?”* The state of replicability in this field is bleak, as not a single publication in our systematic literature study provided accessible implementations. Contacting authors for their implementations proved largely unsuccessful, and re-implementing these systems is both time-consuming and challenging due to the inconsistent information provided in the publications.

RQ2. *“How well do these schemes generalize to ad-hoc smartphone communication use cases?”* Most of the schemes we tested are not yet reliable for practical ad-hoc data transmission between nearby smartphones. Although many claim high throughput, they fail to sustain it in real conditions. However, the scheme by Lee et al. [33] proved reliable even at longer distances, and the non-academic ggwave [12] is effective for close to medium distances.

RQ3. *“Which practical challenges affect nearby acoustic data transmission between smart devices?”* Acoustic data transmission on smart devices faces additional, often overlooked limitations compared to RF communication, including severe inter-symbol interference from long delay spreads and the added complexity of non-Gaussian noise and device diversity. Because simulations cannot

accurately capture these real-world complexities, we systematically tested acoustic schemes across multiple device models under varying environmental and noise conditions to quantify their impact on reliability and generalizability. Our findings show that thorough practical testing clarifies the trade-off between throughput and reliability for specific use cases, guiding the selection of an optimal system for practical applications. To achieve practical applicability, future schemes should be resistant to delay spreads of tens of milliseconds, withstand large volume fluctuations due to device heterogeneity, and handle both continuous and impulsive noise.

Availability

To provide transparency about our research results, we provide a replication package [51] containing:

- A Methodological Transparency & Reproducibility Appendix (META) containing detailed descriptions of our experimental setup, including the exact settings for transmission and reception (see [Appendix A](#)).
- Our three re-implementations of acoustic communication schemes [13, 33, 70] in MATLAB (see [Section 5](#)).
- A collection of 11 900 WAV files with recorded transmissions from eight acoustic communication schemes, captured both in an office environment and an anechoic chamber (see [Section 6](#)).
- A CSV file with our raw data, including the BER for each scheme's recordings.
- The full MATLAB analysis code that decodes recordings from different schemes in parallel, along with R scripts that generate all result figures.

Acknowledgments

We thank all authors who replied to us when we requested their acoustic data transmission implementations following our literature study ([Section 5](#)). Furthermore, we thank the anonymous reviewers for their helpful suggestions. This work has been funded by the LOEWE initiative (Hesse, Germany) within the emergenCITY center [LOEWE/1/12/519/03/05.001(0016)/72].

References

- [1] Will Archer Arentz and Udana Bandara. 2011. Near Ultrasonic Directional Data Transfer for Modern Smartphones. In *Proceedings of the 13th International Conference on Ubiquitous Computing (UbiComp '11)*. Association for Computing Machinery, New York, NY, USA, 481–482. <https://doi.org/10.1145/2030112.2030181>
- [2] Vaibhav Bajpai, Anna Brunstrom, Anja Feldmann, Wolfgang Kellerer, Aiko Pras, Henning Schulzrinne, Georgios Smaragdakis, Matthias Wählisch, and Klaus Wehrle. 2019. The Dagstuhl Beginners Guide to Reproducibility for Experimental Networking Research. *SIGCOMM Comput. Commun. Rev.* 49, 1 (Feb. 2019), 24–30. <https://doi.org/10.1145/3314212.3314217>
- [3] Green Bang, Myoungbeom Chung, and Ilju Ko. 2016. Data Communication Method Based on Inaudible Sound at near Field. In *2016 18th International Conference on Advanced Communication Technology (ICACT)*. 639–642. <https://doi.org/10.1109/ICACT.2016.7423502>
- [4] Chao Cai, Zhe Chen, Jun Luo, Henglun Pu, Menglan Hu, and Rong Zheng. 2022. Boosting Chirp Signal Based Aerial Acoustic Communication Under Dynamic Channel Conditions. *IEEE Transactions on Mobile Computing* 21, 9 (Sept. 2022), 3110–3121. <https://doi.org/10.1109/TMC.2021.3051665>
- [5] Yongliang Chen, Tao Ni, Weitao Xu, and Tao Gu. 2022. SwipePass: Acoustic-based Second-factor User Authentication for Smartphones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3 (Sept. 2022), 106:1–106:25. <https://doi.org/10.1145/3550292>
- [6] Ming Ki Chong and Hans W. Gellersen. 2013. How Groups of Users Associate Wireless Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1559–1568. <https://doi.org/10.1145/2470654.2466207>
- [7] Christian Collberg and Todd A. Proebsting. 2016. Repeatability in Computer Systems Research. *Commun. ACM* 59, 3 (Feb. 2016), 62–69. <https://doi.org/10.1145/2812803>

- [8] CopSonic Technologies. 2025. *CopSonic - Ultrasonic Authentication*. <https://www.copsonic.com/> [Retrieved 2025-03-03].
- [9] CUE Audio, Inc. 2025. *Cue - Creating Unforgettable Experiences*. <https://www.connectwithcue.com/> [Retrieved 2025-03-03].
- [10] Anind K. Dey. 2001. Understanding and Using Context. *Personal and Ubiquitous Computing* 5, 1 (Feb. 2001), 4–7. <https://doi.org/10.1007/s007790170019>
- [11] Vadim Gerasimov and Walter Bender. 2000. Things That Talk: Using Sound for Device-to-Device and Device-to-Human Communication. *IBM Systems Journal* 39, 3.4 (2000), 530–546–530–546. <https://doi.org/10.1147/sj.393.0530>
- [12] Georgi Gerganov. 2025. *ggwave*. <https://github.com/ggerganov/ggwave> [Retrieved 2025-03-03].
- [13] Pascal Getreuer, Chet Gney, Richard F. Lyon, and Rif A. Saurous. 2018/june. Ultrasonic Communication Using Consumer Hardware. *IEEE Transactions on Multimedia* 20, 6 (2018/june), 1277–1290. <https://doi.org/10.1109/TMM.2017.2766049>
- [14] Monira A. Gonçalves, Rafael V. Louredo, Michel P. Tcheou, and Lisandro Lovisolo. 2017. Acoustic Modem Using OFDM and Least-Square Channel Estimation. In *2017 IEEE 9th Latin-American Conference on Communications (LATINCOM)*. 1–5. <https://doi.org/10.1109/LATINCOM.2017.8240178>
- [15] Michael T. Goodrich, Michael Sirivianos, John Solis, Claudio Soriente, Gene Tsudik, and Ersin Uzun. 2009/february. Using Audio in Secure Device Pairing. *Int. J. Secur. Netw.* 4, 1/2 (2009/february), 57–68–57–68. <https://doi.org/10.1504/IJSN.2009.023426>
- [16] Michael T. Goodrich, Michael Sirivianos, John Solis, Gene Tsudik, and Ersin Uzun. 2006. Loud and Clear: Human-Verifiable Authentication Based on Audio. In *26th IEEE International Conference on Distributed Computing Systems (ICDCS)*. IEEE. <https://doi.org/10.1109/icdcs.2006.52>
- [17] Daniel Halperin, Thomas S. Heydt-Benjamin, Benjamin Ransford, Shane S. Clark, Benessa Defend, Will Morgan, Kevin Fu, Tadayoshi Kohno, and William H. Maisel. 2008. Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses. In *2008 IEEE Symposium on Security and Privacy (Sp 2008)*. 129–142. <https://doi.org/10.1109/SP.2008.31>
- [18] Michael Haus, Aaron Yi Ding, Qing Wang, Juhani Toivonen, Leonardo Tonetto, Sasu Tarkoma, and Jorg Ott. 2019. Enhancing Indoor IoT Communication with Visible Light and Ultrasound. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. 1–6. <https://doi.org/10.1109/ICC.2019.8762001>
- [19] Yitao He and Junyu Bian. 2018. Novel Ultrasonic Broadcast Communication System. In *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*. 1–7. <https://doi.org/10.1109/WCSP.2018.8555902>
- [20] Yann Hornych, Javier Cañada Toledo, Boyang Wang, Won-Jae Yi, and Jafar Saniie. 2020. Near-Ultrasonic Communications for IoT Applications Using Android Smartphone. In *2020 IEEE International Conference on Electro Information Technology (EIT)*. 407–410. <https://doi.org/10.1109/EIT4899.2020.9208265>
- [21] Veronika Hromadová, Peter Kasák, Roman Jarina, and Peter Bráda. 2022. Frequency Response of Smartphones at the Upper Limit of the Audible Range. In *2022 ELEKTRO (ELEKTRO)*. 1–5. <https://doi.org/10.1109/ELEKTRO53996.2022.9803475>
- [22] Kwang Myung Jeon, Hong Kook Kim, and Myung J. Lee. 2016. Noncoherent Low-Frequency Ultrasonic Communication System with Optimum Symbol Length. *International Journal of Distributed Sensor Networks* 12, 5 (May 2016), 9713180. <https://doi.org/10.1155/2016/9713180>
- [23] Xiaoyu Ji, Chaohao Li, Xinyan Zhou, Juchuan Zhang, Yanmiao Zhang, and Wenyuan Xu. 2020. Authenticating Smart Home Devices via Home Limited Channels. *ACM Trans. Internet Things* 1, 4, Article 24 (Aug. 2020), 24 pages. <https://doi.org/10.1145/3399432>
- [24] Meng Jin, Yuan He, Yunhao Liu, and Xinbing Wang. 2022. Furtively Connecting IoT Devices with Acoustic Noise. In *2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. 195–207. <https://doi.org/10.1109/IPSN54338.2022.00023>
- [25] Tero Jokela, Ming Ki Chong, Andrés Lucero, and Hans Gellersen. 2015. Connecting Devices for Collaborative Interactions. *Interactions* 22, 4 (June 2015), 39–43. <https://doi.org/10.1145/2776887>
- [26] Tero Jokela and Andrés Lucero. 2014. FlexiGroups: Binding Mobile Devices for Collaborative Interactions in Medium-Sized Groups with Device Touch. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. Association for Computing Machinery, New York, NY, USA, 369–378. <https://doi.org/10.1145/2628363.2628376>
- [27] Daniel Jones. 2022. *How Near-Ultrasonic Audio Adds Spatial Awareness to the Sonos System*. <https://tech-blog.sonos.com/posts/how-near-ultrasonic-audio-adds-spatial-awareness-to-the-sonos-system/> [Retrieved 2025-03-03].
- [28] Soonwon Ka, Tae H. Kim, Jae Y. Ha, Sun H. Lim, Su C. Shin, Jun W. Choi, Chulyoung Kwak, and Sunghyun Choi. 2016. Near-Ultrasound Communication for TV's 2Nd Screen Services. In *Proceedings of the 22Nd Annual International Conference on Mobile Computing and Networking*. ACM, 42–54–42–54. <https://doi.org/10.1145/2973750.2973774>
- [29] Vijay A. Kanade. 2018. Audio-Internet [A-I]: A Novel Technique for Streaming [Bits-over-Sound]: SoundData. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. 1690–1694.

<https://doi.org/10.1109/ICACCI.2018.8554953>

[30] Soohyun Kim, Hyunsu Mun, and Youngseok Lee. 2019. A Data-Over-Sound Application: Attendance Book. In *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. 1–4. <https://doi.org/10.23919/APNOMS.2019.8892996>

[31] Greg Kumparak. 2013. *SlickLogin Aims To Kill The Password By Singing A Silent Song To Your Smartphone*. <https://techcrunch.com/2013/09/09/slicklogin-wants-to-kill-the-password-by-singing-a-silent-song-to-your-smartphone/> [Retrieved 2025-03-03].

[32] Chulyoung Kwak, Seongwong Kim, Soonwon Ka, Jihwan Lee, and Sunghyun Choi. 2019/june. No Entry: Anti-Noise Energy Detector for Chirp-Based Acoustic Communication. In *2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. 1–9. <https://doi.org/10.1109/SAHCN.2019.8824986>

[33] Hyewon Lee, Tae H. Kim, Jun W. Choi, and Sunghyun Choi. 2015/april. Chirp Signal-Based Aerial Acoustic Communication for Smart Devices. In *INFOCOM*. 2407–2415. <https://doi.org/10.1109/INFOCOM.2015.7218629>

[34] Jihwan Lee, Chulyoung Kwak, Seongwon Kim, and Saewoong Bahk. 2020. Reliable and Low-Complexity Chirp Spread Spectrum-Based Aerial Acoustic Communication. *IEEE Access* 8 (2020), 151589–151601. <https://doi.org/10.1109/ACCESS.2020.3017097>

[35] Min-Chun Lin, Fu-Yu Huang, and Tzi-Dar Chiueh. 2015. A-NFC: Two-way near-Field Communications (NFC) via Inaudible Acoustics. In *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*. 1–6. <https://doi.org/10.1109/IISA.2015.7388085>

[36] Qiongzheng Lin, Lei Yang, and Yunhao Liu. 2017. TagScreen: Synchronizing Social Televisions through Hidden Sound Markers. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. 1–9. <https://doi.org/10.1109/INFOCOM.2017.8057170>

[37] LISNR, Inc. 2025. *Radius 3.0 Ultrasonic SDK*. <https://lisnr.com/radius-ultrasonic-sdk-3/> [Retrieved 2025-03-03].

[38] Cristina V. Lopes and Pedro M. Q. Aguiar. 2001. Aerial Acoustic Communications. In *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*. 219–222. <https://doi.org/10.1109/ASPA.2001.969582>

[39] Cristina V. Lopes and Pedro M. Q. Aguiar. 2003/july. Acoustic Modems for Ubiquitous Computing. *IEEE Pervasive Computing* 2, 3 (2003/july), 62–71. <https://doi.org/10.1109/MPRV.2003.1228528>

[40] Anil Madhavapeddy, David Scott, and Richard Sharp. 2003. Context-Aware Computing with Sound. In *UbiComp 2003: Ubiquitous Computing (Lecture Notes in Computer Science)*, Anind K. Dey, Albrecht Schmidt, and Joseph F. McCarthy (Eds.). Springer, Berlin, Heidelberg, 315–332. https://doi.org/10.1007/978-3-540-39653-6_25

[41] Anil Madhavapeddy, Richard Sharp, David Scott, and Alastair Tse. 2005/july. Audio Networking: The Forgotten Wireless Technology. *IEEE Pervasive Computing* 4, 3 (2005/july), 55–60. <https://doi.org/10.1109/MPRV.2005.50>

[42] Adib Mehrabi, Antonella Mazzoni, Daniel Jones, and Anthony Steed. 2019. Evaluating the User Experience of Acoustic Data Transmission. *Personal and Ubiquitous Computing* (Dec. 2019), 1–14–1–14. <https://doi.org/10.1007/s00779-019-01345-7>

[43] Jonathon Miegel, Philip Branch, and Peter Blamey. 2018. Wireless Communication between Personal Electronic Devices and Hearing Aids Using High Frequency Audio and Ultrasound. *The Journal of the Acoustical Society of America* 144, 4 (Oct. 2018), 2598–2604. <https://doi.org/10.1121/1.5063813>

[44] Satoshi Moriyama, Junji Takahashi, and Yoshito Tobe. 2017. Creation of Communication Groups Using Sound of Smartphones. In *Intelligent Environments 2017*. IOS Press, 8–14. <https://doi.org/10.3233/978-1-61499-796-2-8>

[45] Rajalakshmi Nandakumar, Krishna K. Chintalapudi, Venkat Padmanabhan, and Ramarathnam Venkatesan. 2013. Dhwan: Secure Peer-to-Peer Acoustic NFC. *ACM SIGCOMM Computer Communication Review* 43, 4 (Aug. 2013), 63–74–63–74. <https://doi.org/10.1145/2534169.2486037>

[46] B. A. Nosek, G. Alter, G. C. Banks, D. Borsboom, S. D. Bowman, S. J. Breckler, S. Buck, C. D. Chambers, G. Chin, G. Christensen, M. Contestabile, A. Dafoe, E. Eich, J. Freese, R. Glennerster, D. Goroff, D. P. Green, B. Hesse, M. Humphreys, J. Ishiyama, D. Karlan, A. Kraut, A. Lupia, P. Mabry, T. Madon, N. Malhotra, E. Mayo-Wilson, M. McNutt, E. Miguel, E. Levy Paluck, U. Simonsohn, C. Soderberg, B. A. Spellman, J. Turitto, G. VandenBos, S. Vazire, E. J. Wagenmakers, R. Wilson, and T. Yarkoni. 2015. Promoting an Open Research Culture. *Science* 348, 6242 (June 2015), 1422–1425. <https://doi.org/10.1126/science.aaa2374>

[47] Ed Novak, Zhuofan Tang, and Qun Li. 2019. Ultrasound Proximity Networking on Smart Mobile Devices for IoT Applications. *IEEE Internet of Things Journal* 6, 1 (Feb. 2019), 399–409. <https://doi.org/10.1109/JIOT.2018.2848099>

[48] Matthew J. Page, Joanne E. McKenzie, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, Jennifer M. Tetzlaff, Elie A. Akl, Sue E. Brennan, Roger Chou, Julie Glanville, Jeremy M. Grimshaw, Asbjørn Hróbjartsson, Manoj M. Lalu, Tianjing Li, Elizabeth W. Loder, Evan Mayo-Wilson, Steve McDonald, Luke A. McGuinness, Lesley A. Stewart, James Thomas, Andrea C. Tricco, Vivian A. Welch, Penny Whiting, and David Moher. 2021. The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews. *BMJ* 372 (March 2021), n71. <https://doi.org/10.1136/bmj.n71>

[49] Prasad Patil, Roger D. Peng, and Jeffrey T. Leek. 2016. A Statistical Definition for Reproducibility and Replicability. , 066803 pages. <https://doi.org/10.1101/066803>

[50] Florentin Putz, Flor Álvarez, and Jiska Classen. 2020. Acoustic Integrity Codes: Secure Device Pairing Using Short-Range Acoustic Communication. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 31–41. <https://doi.org/10.1145/3395351.3399420> arXiv:2005.08572

[51] Florentin Putz, Philipp Fortmann, Jan Frank, Christoph Haugwitz, Mario Kupnik, and Matthias Hollick. 2025. *Replication Package: Evaluating Acoustic Data Transmission Schemes for Ad-Hoc Communication Between Nearby Smart Devices*. <https://doi.org/10.5281/zenodo.17661991> [Dataset].

[52] Florentin Putz, Steffen Haesler, and Matthias Hollick. 2024. Sounds Good? Fast and Secure Contact Exchange in Groups. In *Proceedings of the ACM on Human-Computer Interaction 8* (New York, NY, USA) (CSCW2). Association for Computing Machinery. <https://doi.org/10.1145/3686964>

[53] Florentin Putz, Steffen Haesler, Thomas Völk, Maximilian Gehring, Nils Rollhausen, and Matthias Hollick. 2024. PairSonic: Helping Groups Securely Exchange Contact Information. In *Companion Publication of the 2024 Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '24 Companion)*. Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/367884.3681818>

[54] Nafis Sadeq and Md. Shohrab Hossain. 2022. Privacy Breach in Android Smartphone Through Inaudible Sound. In *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. 2441–2446. <https://doi.org/10.1109/WCNC51071.2022.9771721>

[55] Kavous Salehzadeh Niksirat, Lahari Goswami, Pooja S. B. Rao, James Tyler, Alessandro Silacci, Sadiq Aliyu, Annika Aebli, Chat Wacharamanotham, and Mauro Cherubini. 2023. Changes in Research Ethics, Openness, and Transparency in Empirical Studies between CHI 2017 and CHI 2022. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–23. <https://doi.org/10.1145/3544548.3580848>

[56] Shigeru Sasano, Fumihiro Sasamori, Osamu Takyu, and Shiro Handa. 2014. Performance Evaluation of Acoustic OFDM with Repetition Coding. In *2014 International Conference on Information and Communication Technology Convergence (ICTC)*. 798–799. <https://doi.org/10.1109/ICTC.2014.6983292>

[57] Utkalika Satapathy, Soumyajit Chatterjee, and Sandip Chakraborty. 2020. SilentInformer: A Framework for Information Dispersion Using Inaudible Acoustic Signals. In *2020 International Conference on COMmunication Systems & NETworkS (COMSNETS)*. 543–546. <https://doi.org/10.1109/COMSNETS48256.2020.9027493>

[58] Markus Scheck, Florentin Putz, Frank Hessel, Hermann Leinweber, Jonatan Crystall, and Matthias Hollick. 2023. DEMO: Secure Bootstrapping of Smart Speakers Using Acoustic Communication. (2023). <https://doi.org/10.26083/tuprints-00024180> Demo presented at the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec 2023).

[59] Liming Shi, Xinheng Wang, Limin Yu, Wenwu Wang, Zhi Wang, Muddeswar Iqbal, Charalampos C. Tsimenidis, and Shahid Mumtaz. 2023. A Long-Range Aerial Acoustic Communication Scheme. *Physical Communication* 60 (Oct. 2023), 102135. <https://doi.org/10.1016/j.phycom.2023.102135>

[60] Shopkick, Inc. 2025. *Shopkick*. <https://www.shopkick.com/> [Retrieved 2025-03-03].

[61] Patrick E. Shrout and Joseph L. Rodgers. 2018. Psychology, Science, and Knowledge Construction: Broadening Perspectives from the Replication Crisis. *Annual Review of Psychology* 69, Volume 69, 2018 (Jan. 2018), 487–510. <https://doi.org/10.1146/annurev-psych-122216-011845>

[62] Stimshop. 2024. *Wi-Us by Stimshop. Wireless via Ultrasound*. <https://wi-us.eu/> [Retrieved 2024-11-06].

[63] The American Business Awards. 2014. *VeriFone's Way2ride*. <https://stevieawards.com/aba/verifones-way2ride> [Retrieved 2025-03-03].

[64] TrillBit Inc. 2025. *TrillBit - Contactless, Fast and Secure Authentication Protocol*. <https://www.trillbit.com/technology-trillbit> [Retrieved 2025-03-03].

[65] Qian Wang, Kui Ren, Man Zhou, Tao Lei, Dimitrios Koutsoukos, and Lu Su. 2016. Messages behind the Sound. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking - MobiCom '16*. ACM Press. <https://doi.org/10.1145/2973750.2973765>

[66] Tianben Wang, Daqing Zhang, Yuanqing Zheng, Tao Gu, Xingshe Zhou, and Bernadette Dorizzi. 2018. C-FMCW Based Contactless Respiration Detection Using Acoustic Signal. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4 (Jan. 2018), 170:1–170:20. <https://doi.org/10.1145/3161188>

[67] Weiguo Wang, Jinming Li, Yuan He, Xiuzhen Guo, and Yunhao Liu. 2022/march. MotorBeat: Acoustic Communication for Home Appliances via Variable Pulse Width Modulation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 1 (2022/march), 1–24–1–24. <https://doi.org/10.1145/3517255>

[68] Yu Wang, Xiaojun Zhu, and Hao Han. 2021. ChirpMu: Chirp Based Imperceptible Information Broadcasting with Music. In *2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS)*. 1–9. <https://doi.org/10.1109/IWQOS52092.2021.9521353>

- [69] Katsuyoshi Yamamoto and Hiroshi Kubo. 2022. 32kbps Terrestrial Acoustic Communications of Single Carrier Block Transmission Employing Transmit Precoding and Channel Interpolation. *IEICE Communications Express* 11, 3 (2022), 141–147. <https://doi.org/10.1587/comex.2021XBL0206>
- [70] Bingsheng Zhang, Qin Zhan, Si Chen, Muyuan Li, Kui Ren, Cong Wang, and Di Ma. 2014. PriWhisper : Enabling Keyless Secure Acoustic Communication for Smartphones. *IEEE Internet of Things Journal* 1, 1 (Feb. 2014), 33–45. <https://doi.org/10.1109/JIOT.2014.2297998>
- [71] Xiao Zhang, Jiqiang Liu, Si Chen, Yongjun Kong, and Kui Ren. 2019. PriWhisper+: An Enhanced Acoustic Short-Range Communication System for Smartphones. *IEEE Internet of Things Journal* 6, 1 (Feb. 2019), 614–627–614–627. <https://doi.org/10.1109/jiot.2018.2850524>
- [72] Yanchao Zhao, Yiming Zhao, Si Li, Hao Han, and Lei Xie. 2023. UltraSnoop: Placement-agnostic Keystroke Snooping via Smartphone-based Ultrasonic Sonar. *ACM Trans. Internet Things* 4, 4, Article 22 (Nov. 2023), 24 pages. <https://doi.org/10.1145/3614440>
- [73] Man Zhou, Qian Wang, Kui Ren, Dimitrios Koutsonikolas, Lu Su, and Yanjiao Chen. 2019. Dolphin: Real-Time Hidden Acoustic Signal Capture with Smartphones. *IEEE Transactions on Mobile Computing* 18, 3 (March 2019), 560–573–560–573. <https://doi.org/10.1109/tmc.2018.2842771>

A Methodological transparency & reproducibility appendix

This section offers detailed information about our experimental setup with the goal of improving the transparency and replicability of our work. We document the exact settings we used for transmission and reception. [Table 5](#) lists all the hardware and software we used, including version numbers.

A.1 General

For every experiment, we repeated the measurements in case of external effects, such as vocal or noise interference, or when there were problems with playback and recording of the transmission.

A.2 Transmission

Following the setup shown in [Figure 3](#), we used the smartphones listed in [Table 5](#). First, we generated transmission WAV files for each scheme, which are part of our dataset. Message lengths were set to the publication’s default or if the implementation supported custom message sizes we adjusted it for an approximate five-second transmission, which we consider a practical upper limit for user wait times. All audio signals were then normalized by dividing the entire signal by the highest absolute sample value and scaling it to -3 dBFS.

We transmitted these files on each smartphone using the open-source Android app *Vinyl*, which allows automating the playback of multiple WAV files using playlists. We compiled a playlist containing the transmission files from all schemes, separated by two-second-long silent WAV files to prevent reverberation from affecting subsequent transmissions. For convenience, we also provide a single WAV file containing repeated transmissions from all schemes and corresponding silent gaps for easy playback. We controlled the output volume using the open-source Android app *Volume Control*, which provides precise control of the Android volume index, as shown in [Table 3](#).

A.3 Reception

On the receiver smartphone, we recorded each transmission using the open-source Android app *Audio Recorder*, which allows detailed control over the microphone source. We selected the MIC source for all smartphones, except for the Google Pixel 4a, which also supported the UNPROCESSED source to bypass Android’s post-processing. Recordings were saved as 24 bit PCM float mono WAV files at a minimum of 44.1 kHz, with software gain at 100% and all extra filtering disabled.

We recorded multiple transmissions consecutively and later segmented them manually in the *REAPER* digital audio workstation. To streamline this process, we used a script⁹ for batch cutting

⁹REAPER script for splitting: <https://stash.reaper.fm/v/25677/SplitX.lua>

and a custom iterator for unique file naming. The segments were rendered as 32 bit PCM mono WAV files at a minimum of 44.1 kHz. Finally, we decoded these files with each scheme’s receiver implementation to calculate the TER. If a decoded message was longer than expected, we truncated it; if it was shorter, we counted the missing bits as errors.

A.4 Experimental setup

We conducted tests in several indoor locations and an anechoic chamber. For the experiments in the indoor environment, we mounted all smartphones on small tripods with rubber feet, which isolated them from vibrations caused by the their speakers. We positioned the smartphones so their speakers and microphones faced directly toward each other (on-axis), measuring the distance between. The dimensions (length x width x height) of the indoor spaces were: *small office* (5.2 m x 4 m x 3 m), *large office* (13.5 m x 5.3 m x 2.8 m), *meeting room* (7.9 m x 5.3 m x 3 m), *lecture room* (13.5 m x 6.6 m x 3.6 m), and the *hallway* for long-distance tests (40 m x 2 m x 3 m).

The *anechoic chamber* (5.2 m x 7.7 m x 5.8 m) provided a reflection-free environment, with walls lined with mineral wool cones coated in synthetic resin (cone length 100 cm, cone base area 24 cm x 24 cm, reflection coefficient below 0.01 for frequencies above 100 Hz).

A.4.1 Preliminary best-case setup. For the preliminary tests (Section 6.2), we established a best-case baseline using studio equipment instead of smartphones. For transmission, a Neumann KH 80 DSP studio speaker played the signals (local control off, acoustic control “free-standing”, output level 94 dB SPL, input gain at 0 dB). We connected the speaker to an RME ADI-2 Pro FS R BE audio interface (−40 dB gain at a reference level of 24 dBu).

For recording, a calibrated Earthworks M23R reference microphone captured the audio. The signal was fed through a Camden EC1 preamp (35.5 dB gain, no additional filtering) into the audio interface (without additional gain). The preamp’s output was split to both interface inputs and summed to mono using mid-side processing, increasing the SNR by 3 dB. We used REAPER for playback and recording. The speaker and microphone were aligned on-axis at a distance of 50 cm.

A.4.2 Ambient noise setup. For the ambient noise experiment (Section 6.5), we used the standard smartphone and introduced noise using the Neumann KH 80 DSP studio speaker. The speaker was placed 1 m from the receiver, orthogonal to the transmission axis. The speaker configuration was the same as in the preliminary tests (Section A.4.1), with the exception of the output gain of our interface, which was set to −35 dB to subjectively match the real-life noise levels of the scenarios at the receiver’s location. During each iteration of the experiment, one field recording was played repeatedly while the transmission sounds were played sequentially. This process was repeated for all three recordings (café, train station, marketplace), which are freely available online¹⁰ under a CC-BY license.

A.4.3 Dynamic environment setup. To evaluate performance in a time-varying channel (Section 6.6), we conducted tests in a lecture room with four people walking slowly and quietly (without talking) around the devices. The smartphones were mounted on small tripods placed on tables in the center of the room, 50 cm apart on-axis. For comparison, a static control test was performed with an identical furniture layout but without people.

¹⁰Café field recording (CC BY 3.0): <https://freesound.org/people/AshFox/sounds/172968/>, train station field recording (CC BY 4.0): <https://freesound.org/people/Kyster/sounds/121576/>, marketplace field recording (CC BY 3.0): https://freesound.org/people/le_abbaye_Noirlac/sounds/129677/

Table 5. **Hardware and software used during our experiments.**

Type	Manufacturer	Product	Version
Microphone	Earthworks	M23R	
Preamp	Camden	EC1	
Audio Interface	RME	ADI-2 Pro FS R BE	
Calibration Device	Galaxy Audio	CM-C200	
Loudspeaker	Neumann	KH80 DSP	
Smartphone	Google	Pixel 4a	128GB
Smartphone	Google	Pixel 6 Pro	5G Sub-6 128GB
Smartphone	Huawei	Nexus 6P	32GB
Smartphone	Oppo	Reno 6	Dual-SIM 5G 128GB
Smartphone	Samsung	Galaxy S20 Ultra	5G 128GB
Desk Microphone Stand	K&M	23150-3	
Ball Joint	Roadworx	Universal Ball Joint	
Smartphone Clamp	Roadworx	Smartphone Clamp	
Smartphone App	Adrien Poupa	Vinyl Music Player ¹	v1.5.0
Smartphone App	Stas Shakirov	Volume Control ²	v2.6.0
Smartphone App	axet	Audio Recorder ³	v3.5.15
DAW	Cockos	REAPER ⁴	v6.82

¹ Vinyl: <https://f-droid.org/en/packages/com.poupa.vinylmusicplayer/>² Volume Control: <https://f-droid.org/packages/com.punksta.apps.volumecontrol/>³ Audio Recorder: <https://f-droid.org/en/packages/com.github.axet.audiorecorder/>⁴ REAPER: <https://www.reaper.fm/>

A.5 Replicating our experiments

The procedures described above should allow for the replication of our experiments. However, the reliability of acoustic transmissions is sensitive to factors like specific device models, microphone/speaker quality, and environmental acoustics. While we expect the overall trends to be similar, absolute TER values will likely differ.

B Re-implementations

In this section, we detail the challenges we encountered during the re-implementation of Lee et al.’s scheme, Nearby, and PriWhisper.

B.1 Lee et al.

We were able to implement all aspects of this scheme as described in their paper [33]. Their practice of listing most system parameters clearly in a table was particularly helpful. We interpret the 16 bps data rate mentioned by Lee et al. as the gross data rate, as it appears they did not account for the preamble in their calculations. Based on the default system parameters provided, we calculate a net data rate of 14.55 bps.

B.2 Nearby

Implementing the transmitter for the Nearby system [13] was straightforward, following the information provided in the paper. However, re-implementing the receiver proved challenging, as some aspects of the paper were unclear to us.

B.2.1 Synchronization. To locate the transmission’s start within the recording, the system processes the signal by iterating over samples, segmenting the signal from each sample offset, and correlating these segments with a period of the code signal. This correlation indicates the likelihood that a symbol begins at each sample offset, as each MFSK symbol was spread by the code signal. After correlating across all time offsets, the results are normalized to compensate for any potential Doppler shifts. These results are termed “normalized acquisition scores” in their paper (Section VI.C in [13]). Next, we identify the spacer symbol’s position using the normalized acquisition scores. Ideally, a distinct peak at the spacer symbol’s position would appear when plotting these scores. However, since the same code signal spreads all token symbols, peaks appear for each symbol, not just the spacer. We implemented a method to identify the positions of the n highest peaks—where n represents the number of token symbols—and select the earliest peak as the start. The paper does not discuss this issue or how to resolve it.

B.2.2 Block processing. The authors mention that their implementation processes signals in 100 ms blocks to likely improve time efficiency, yet the paper does not detail how this feature is implemented. We chose not to implement this aspect, as it is unlikely to impact our evaluation results.

B.2.3 Ambiguity in Algorithm 2. Algorithm 2 in the paper [13] employs a counter variable n in line 2 that initializes at zero and increments by one in each iteration to compute a “raw acquisition score” for each sample offset n , representing the likelihood of a data symbol starting at that offset. However, the authors do not specify an upper bound for n . We have documented our method for determining this upper bound in the comments of our receiver function, although it remains unclear whether this method aligns with the authors’ intended approach.

B.2.4 Ambiguity in Equation 47. In Equation 47 in the paper [13], the variable c , representing the sinc-interpolated code sequence, is indexed at position $(kM_p + m)/F_b$. Here, F_b denotes a sample rate of 12 kHz, which means the division results in a time in seconds at which the code signal should be indexed. Since c is a discrete signal sampled at 48 kHz, it cannot be indexed at arbitrary times. To convert the time $(kM_p + m)/F_b$ to a corresponding sample offset in c , we multiply by 48 000, the sample rate. Again, it is not clear if the authors intended this.

B.2.5 Ambiguity in Algorithm 3. Algorithm 3 in the paper [13] uses the matrix of raw acquisition scores a^{raw} , generated by Algorithm 2. In line 8 of Algorithm 3, it accesses row $n - d$ of a^{raw} , where n is a counter variable starting at zero and increasing by one each iteration. The paper does not specify an upper bound for n , but we assume that n runs up to the number of rows in a^{raw} reduced by one. The variable d is set to 215, leading to the algorithm attempting to access negative row indices during the first 215 iterations, which is invalid. We resolved this issue by omitting line 8 in iterations where $n - d$ results in a negative index. However, as a consequence, the normalized acquisition score matrix ends up with 215 fewer rows than a^{raw} . It is unclear whether this reduction is in line with the authors’ intentions.

B.2.6 Ambiguity during noise suppression. The bandpass filter for noise suppression uses a variable η , which is supposed to be “a small positive parameter” (Section VI.A in [13]). Since the paper does not specify an exact value for η , we set it to 1.

B.2.7 Data rate. We interpret the stated data rate of 94.5 bps as the gross data rate. The system can operate without the token structure that includes additional spacers and a parity symbol, in which case 94.5 bps is also the net data rate. However, when using the full token structure as described in the paper and used in our evaluation, the net data rate decreases to 84 bps.

B.3 PriWhisper

When re-implementing PriWhisper [70], the major difficulty was designing a mechanism for synchronization and calibration, as the paper does not clearly explain these aspects. Additionally, the descriptions of the interleaving and checksum processes in the paper were ambiguous. The specified data rate was also unclear, further complicating our implementation efforts.

B.3.1 Synchronization. In the original implementation, a sinusoid preamble is used within the jamming signal to aid synchronization. However, for our study, we did not implement the security features of the system, focusing solely on data transmission. To minimize the impact on the data rate, we combine the synchronization and calibration sequence. Accordingly, we introduced a special symbol at the beginning of each transmission, which contains the sum of all available data waveforms $s(t)$ (Section III.A in [70]). For synchronization, we use this known synchronization symbol and cross-correlate it with the recording. The position of the highest peak in the output is considered the start of the transmission.

B.3.2 Calibration. To determine the frequency of an MFSK symbol, we correlate each symbol duration in the recording with all available frequencies using the following equation from Section III.A in [70]:

$$R_m = \left| \int_0^T r(t) e^{i2\pi(f_c+m\Delta f)t} dt \right|, m \in [0, M-1] \quad (1)$$

After iterating over all frequencies, we select the frequency with the highest correlation value and map it to the binary number it represents. We repeat these steps for all symbols to decode the entire data transmission in the recording. Before demodulation, calibration using the synchronization symbol is crucial due to potential variations in microphone sensitivity across frequencies. Without calibration, [Equation 1](#) might return a lower correlation value for the actual frequency present in a symbol due to the frequency-selectivity of the microphone.

To address this, we apply [Equation 1](#) to the synchronization symbol, containing all frequencies. This establishes a baseline correlation for each frequency. We then calculate a calibration factor for each frequency by dividing the highest of these baseline values by the others. This factor adjusts the correlation values during demodulation, ensuring that each frequency is appropriately represented despite the microphone's bias. However, this calibration step assumes that the channel characteristics remain constant throughout the transmission duration.

B.3.3 Interleaving and checksum. The paper mentions interleaving the data and adding a checksum, but it lacks clear guidance on how to implement the interleaving process and the sequence of these steps. In our implementation, we first perform regular random interleaving and then add the checksum.

B.3.4 Data rate. We calculate a gross data rate of 771 bps for $M = 8$, which differs from the 1027 bps reported by the authors. Our data rate calculation is as follows: Using M different frequencies allows encoding $\log_2(M)$ bits per symbol. For $M = 8$, this results in 3 bits per symbol. With a symbol duration of 2 ms, we transmit 500 symbols per second, equivalent to 1500 bits per second. As one coded data block contains 255 BCH encoded bits, we can transmit $\frac{1500}{255} \approx 5.88$ blocks per second. One block contains 131 bits of pure data, resulting in a gross data rate of $5.88 \times 131 \approx 771$ bps, excluding signal components for synchronization or calibration. The resulting net data rate is 729 bps.