CS 449 / 549 – Human-Computer Interaction

Final Project Report

Project Title: EchoSpaceAR: Visual and Spatial Simulation of Sound for Deaf or Hard of Hearing (DHH) Users

Course Code: CS449–549

Members of the Group:
Zeynep Şahin – 30553, UI/UX researcher.
Alper Çamlı – 30858, Unity/Python Developer
Seyma Kalender – 34800, Behavioral Data Analyst
Emre Berk Hamarat – 31188, Hardware Coordinator
Mehmet Selman Yılmaz – 31158, AR Developer
Korhan Erdoğdu – 30838, Android Developer

Date: January 10, 2026

Sabancı Üniversitesi

I. Abstract

EchoSpaceAR is an augmented reality (AR) prototype that turns sounds in the environment into quick visual-spatial cues to help people stay aware of their surroundings without having to hear them. The system works on Meta Quest 2/3 and Android, but Quest 3 is where it works best and is most optimized. EchoSpaceAR combines a Unity-based AR interface with a real-time audio analysis and classification pipeline. Detected sound events are turned into HUD indications that show direction and urgency while keeping the visual clutter and decision-making burden to a minimum. The project's primary contribution is the comprehensive integration of audio comprehension and augmented reality cue rendering within a limited field of view, underpinned by a human-computer interaction framework (learnability, effectiveness, safety/security) and established predictive principles (Hick–Hyman for decision complexity; Fitts for the interaction cost of a planned single control). We provide a within-subject pilot study contrasting a baseline condition (absence of AR feedback) with an EchoSpaceAR visual feedback condition. Objective metrics encompass reaction time to align with the specified direction and localization performance, evaluated through accuracy, angular error, and success within 15°. The Turkish Technology Acceptance Model questionnaire and the System Usability Scale are used to measure subjective acceptance. Because of the course schedule and access issues, the pilot is done with hearing participants instead of the targeted Deaf or Hard of Hearing population. This makes it less generalizable and is considered an initial test of feasibility and usefulness. Results show that EchoSpaceAR leads to faster orientation and a higher success rate, as well as good acceptance scores. However, latency and stereo-quality problems are major engineering risks for the next version.

II. Keywords

Augmented Reality, Accessibility, Environmental Awareness, Sound Visualization, Deaf and Hard of Hearing, Multimodal Interfaces, Unity, Meta Quest 3, Usability, Technology Acceptance Model, Real-time Audio Classification.

III. User-Interface Requirements (User Expectations & Context)

A. Introduction

Hearing is a key part of being aware of your surroundings, keeping an eye on the environment, and staying safe in everyday life. People who have trouble hearing, especially Deaf and Hard of Hearing (DHH) users, may have a hard time hearing and locating sounds in their environment, like approaching cars, alarms, or human voices. This can make it very hard for them to move around on their own and be aware of their surroundings. Hearing aids and cochlear implants can help in some cases, but they aren't always enough, especially when things are loud, complicated, or unfamiliar.

New breakthroughs in augmented reality (AR) make it possible to assist individuals hear by translating sound information into visual and spatial cues. AR systems can overlay real-time information over the user's actual surroundings, enabling them to engage additional senses without needing to fully focus on their environment. Still, Human-Computer Interaction (HCI) is very important when making AR-based assistive gadgets for DHH users. It is very crucial to keep DHH people's cognitive load low because they largely obtain their information through their eyes. Sign language, lip-reading, viewing the world around you, and utilizing the interface all consume up the same small amount of visual attention. When a lot of tasks that require a lot of visual attention happen at the same time, they can compete with each other and make it hard to think clearly and pay attention. For example, when a DHH person is talking to someone in sign language, they are particularly focused on the signer's hands, face, and body position. If something important happens in the environment at the same moment, such a warning sound or a car coming, the alarm could not go off unless it is clear and loud. But if an assisted AR interface presents more than one visual alert at the same time while the user is reading or conversing, it can make it harder for them to pay attention and require more mental strain instead of helping them stay conscious. There is real-world evidence to back up this fear. Luna et al. (2024) found that DHH participants had trouble understanding and had to work harder to think when many visual signals showed up at once in an AR interface. They wanted feedback that was easier to understand, like not showing so many signs at once.

This conclusion is in line with Cognitive Load Theory, which talks about intrinsic load (how hard a task is), relevant burden (how much time and effort is put into learning), and external stress that comes from bad design (Sweller, 1988; Sweller, Ayres, & Kalyuga, 2011). People who are DHH have a lot to think about because they have to see and hear things at the same time. This means that if a visual interface is too busy or cluttered, it could make you more anxious, which could make you do worse and not pay as much attention to what's going on around you as you should. It's hard to think about cognitive load when the DHH community speaks so many different languages. Many deaf people use sign language as their first language, so they may not be able to write or speak the same way. If the user doesn't speak the language of the text, it might be much harder to use interfaces with a lot of text. This makes it even more important to use clear pictures, a simple style, and cue styles that can change, like using pictures or symbols instead of long, dense text.

EchoSpaceAR fixes this by suggesting a small AR device that you can wear that turns sounds around you into visual clues that don't move. The system's goal is to help people make decisions quickly, see things clearly, and keep conversations simple. EchoSpaceAR doesn't want to completely replace hearing. Instead, it uses visual communications based on direction, meaning, and urgency to help people become more aware of their surroundings.

Making it easy to use is one of the main goals of the design. In the real world, people can't spend a lot of time going through menus, figuring out what all the symbols mean, or making a lot of changes by hand. So, EchoSpaceAR has a single-toggle interaction model, which means that users can turn the overlay on or off with just one action. This design is based on HCI ideas about how to avoid making mistakes, how to make decisions when you have too many options, and how to make your motor work better. There is a test with hearing participants in this report. This decision permits regulated evaluation of system usability, interaction efficiency, and cognitive load management prior to the system's implementation for DHH users, where ethical and accessibility factors demand rigorous design maturity.

The project includes (i) a technical pipeline that can capture sound in real time, sort it, and show it in augmented reality, and (ii) an interaction design based on theory that was tested using well-known models like Hick-Hyman Law, Fitts' Law, Technology Acceptance Model (TAM), and System Usability Scale (SUS). By linking design choices to theoretical frameworks and real-world data, EchoSpaceAR shows how augmented reality systems can strike a balance between being useful, easy to use, and safe in the real world.

B. Stakeholders and Design Process

1) Primary Stakeholders: Deaf and Hard of Hearing Users: The main stakeholders are DHH people who mostly use visual information. Sign language, lip-reading, navigation, and social contact all take up a lot of visual focus, so any assistive AR interface needs to avoid making things harder for the brain or splitting concentration. This led to design principles including keeping things simple, making decisions easier, using the same code, and focusing on safety while sending alerts.

2) Secondary Stakeholders: Hearing Pilot Participants: Hearing pilot participants endorse the prompt validation of interaction logic, usability, and cognitive load management. The study first tests with hearing users to find usability problems before putting stress on the main group of stakeholders. This is in line with ethical and methodological standards for accessibility research.

3) Tertiary Stakeholders: Designers, Researchers, Developers: EchoSpaceAR is a useful example for academics and developers of how to use HCI theory in an end-to-end AR pipeline with a way to measure success. The reproducible plots and CSV summaries provide you artifacts that you can use again and build on in later versions.

IV. Guidelines, Documents, and/or Process

A. HCI Foundations and Alignment

1) Four Pillars of HCI and EchoSpaceAR: EchoSpaceAR fits with the four pillars in the following ways.

(1) Requirements for the user interface. We got our requirements from the DHH context that was talked about in the literature. For example, visual attention is limited and needs to be protected, alarms should be few but useful, and safety-critical information should come first. In practical terms, this means things like limiting the number of cues that can happen at the same time, putting urgency first, and using icons instead of long explanations.

(2) Guidelines for papers and processes. Instead than explaining things after the fact, we used course ideas and rules as design limitations. The interface has learnability (consistency and visibility), effectiveness (quick feedback that can be acted on), and safety and security (preventing errors through constraints and giving users control to recover). We utilize Hick-Hyman Law to explain why we want to limit choices (bounded alternatives) and Fitts' Law to explain why we want one reliable control instead of a list of options.

(3) Software tools for the user interface. We used Unity and Quest hardware to make AR UI development easier, and Python-based analysis to make tables and charts that could be repeated. The deliverables in the project folder (plots, CSV summaries, and a structured results text file) are products of this pillar and help with clear reporting.

(4) Reviews by experts and tests of how easy it is to use. We executed a within-subject controlled pilot experiment and gathered behavioral and attitudinal data, including reaction time, localization outcomes, the Technology Acceptance Model (TAM), and the System Usability Scale (SUS). We also considered engineering diagnostics (latency spikes and stereo difficulties) to be related to usability because they have a direct effect on how timely and reliable feedback is.

2) Learnability, Effectiveness, and Safety/Security (Benyon's Principles):

3) Learnability: Visibility: The system status is meant to be simple: either the overlay is there (alerting) or it's not (listening/suppressed). Each event gives the system a short-lived cue, which makes it easy to understand how it works without having to look for hidden panels.

Sound categories correspond to fixed icons, while urgency corresponds to fixed salience rules. This continuous mapping makes it easier for users to acquire the cue language and lessens the need to remember things between trials.

Familiarity. The cue language uses basic icons and familiar directional metaphors (such arrows and indicators) to make the interface easy to use for people of all literacy levels.

4) Effectiveness: Feedback. The system gives rapid, actionable visual feedback (direction + urgency) for each detected event. This helps people respond quickly instead of waiting to figure out what it means.

The approach reduces down on the number of possibilities that are available at the same time by restricting concurrency and giving precedence to urgency. This makes it easy to choose what to do and raises the chances of finishing a task.

Affordance: The suggested single toggle makes it clear what to do (activate or disable overlay), which makes it easy for consumers to use without having to go through a lot of steps.

Confidence-aware rendering makes sure that outputs with low confidence don't look like signals with great certainty. Concurrency limitations keep overload from making things confusing.

Recovery: The idea of an overlay toggle allows you an instant "escape hatch" and backs up Shneiderman's idea of an internal center of control. Bounded cue persistence and fade-out also make it less likely that you'll get sidetracked for a long time.

Make it easier on short-term memory. The interface doesn't present a lot of things at once; it relies on recognition and constant mappings.

B. Process Model Fit: Spiral and Agile

EchoSpaceAR is a strong fit for the Spiral Process Model since it is risk-focused and iterative. You can think of each cycle in the spiral as: Planning: Set goals (increase awareness), restrictions (limited visual attention; XR region of vision), and success measures (RT, accuracy). Risk analysis: figure out what the biggest risks are, like cognitive overload, misclassification trust, latency instability, and stereo directional ambiguity. Engineering and prototyping: Set up the logs and the pipeline (audio $\rightarrow$ classification $\rightarrow$ mapping $\rightarrow$ HUD). Evaluation/review: conduct a pilot study, examine both objective and subjective indicators, and determine subsequent actions. This report shows that the first full cycle of the spiral is over. It includes an integrated prototype and a controlled evaluation that shows clear engineering risks and demonstrable benefits.

We also used short cycles that were similar to agile within each spiral step (implement, test on device, fix, iterate). This hybrid works well because spiral gives you systematic risk management and assessment gates, while agile gives you speed and ongoing input during development. In the end, the development process works with both course models and modern iteration methods.

C. Hick–Hyman and Fitt's Law Analysis

Hick–Hyman Law shows how the quantity of choices affects how long it takes to make a decision:

$$RT = a + b \cdot \log_2(n), \tag{1}$$

where n is the number of options available.

In AR sound awareness, "choices" include conflicting visual signals. The user has to decide what to pay attention to and how to respond, even if they don't do anything. Interfaces that show several stimuli at once effectively raise n, which makes decisions take longer and puts more strain on the brain. EchoSpaceAR cuts down on good choices by:

- a binary control model (intended toggle: on/off),
- restricted concurrency (limited number of visible stimuli),
- Prioritizing based on urgency (one main actionable cue under overlap),
- Confidence-aware attenuation (making it less likely that people will pay attention to occurrences that are not certain).

We also want to add buttons to the UI, like "close" or "modify." These buttons will be added to the UI along with these rules. This turns an open-ended multi-cue environment into a reduced set of choices, which helps reactions happen faster and more reliably.

Fitts' Law says that pointing time is:

$$MT = a + b\log_2\left(\frac{D}{W} + 1\right), \tag{2}$$

where D is the distance to the target and W is the width of the target. In XR, small targets and vast reaches make it take longer to choose and make more mistakes because they require more accuracy and tracking can change. Because of this, EchoSpaceAR prefers one toggle that is always in the same spot and big enough, rather than several small ones. This is in line with what the course says: raise W for actions that happen often, lower D by putting controls in easy-to-reach areas, and stay away from control panels that are too crowded, which raise both motor and cognitive costs. Also, as mentioned in the last section, we want to add buttons to the UI that let you close it or change it. Along with these rules, these buttons will be part of the UI.

V. User-Interface Software Tools (Implementation)

A. System Overview

1) Target Platforms: EchoSpaceAR is built in Unity and runs on head-worn XR hardware (Meta Quest 2/3) as well as an Android build for testing on mobile devices. The main focus of this research is Quest 3, which is chosen as the principal device for performance and usability testing. The Android build is not the major testing platform; it's just a way to verify for portability.

2) Supported Devices: We tested the system on Meta Quest 3 and Meta Quest 2. An Android version was made to check if it was possible. This report's pilot study data were gathered on Quest 3.

3) End-to-End Flow: The flow from start to finish is:

1) Recording sound from the environment with a microphone.
2) Buffering with low latency into small analytical windows.
3) Feature extraction and sound event classification (label and confidence).
4) Event mapping: category to icon, urgency to priority/salience, and confidence to rendering strength.
5) Real-time rendering in Unity as a head-up display (HUD) that shows the direction and urgency of a cue.
6) Logging behavioral and engineering signals to CSV for analysis and reports that can be repeated.

B. Technical Architecture

1) Audio Capture: The device's microphone picks up sound and stores it in brief windows so that it can be processed almost in real time. The pipeline is set up to cut down on the time between the start of sound and the start of the cue. The audio stream for the pilot setup has a single-channel input and a 16 kHz sampling rate. The analysis window is 1.0 seconds long, and the hop size is 0.25 seconds, which means there are four updates every second. This compromise gives adequate time context for strong categorization while yet making sure that cue updates happen often enough for awareness tasks.

2) Sound Classification: Audio windows that have been recorded are turned into log-mel spectrogram features and put into a small number of categories for environmental events. The classifier gives back:

- an anticipated **category label**,
- A scalar confidence score between 0 and 1,

- A timestamp that matches the Unity log clock.

To keep interactions safe, confidence is utilized to change the strength of the rendering instead of making a hard choice. When confidence is lower, the cue is rendered with less importance and for a shorter time. This stops cues from becoming too sure of themselves, which could lead users astray.

3) Event Mapping: A lightweight rule layer maps classifier outputs to visual events:

- **Category** $\rightarrow$ icon/semantic label (mapping that stays the same).
- Urgency means putting things in order of importance and strength (pulse strength).
- Confidence leads to a decrease in cue intensity and persistence.
- Timing: limited length and fade-out to keep things from getting too crowded.

This mapping makes cognitive load reduction happen by using a narrow cue vocabulary, consistent semantics, and limited concurrency when there is overlap.

4) Direction Definition in the Pilot: In the pilot trial, direction is determined by the written ground-truth event direction instead of the estimated direction-of-arrival (DOA). This guarantees experimental control and precise calculation of angular inaccuracy. The AR cue shows which way to go, so discrepancies in performance can be linked to the availability of feedback instead than sensor noise.

5) Unity—AR Pipeline: Events are sent to Unity in real time and shown as a HUD element in the user's field of vision. Unity also saves all trial events, including timestamps and participant responses, to a CSV file. The design promotes solid graphics and minimum UI movement relative to the headset to reduce perceptual jitter and keep things easy to see while users turn their heads.

C. AR UI Design

1) Cue Mapping: EchoSpaceAR has a little cue language:

- A directional indication in the HUD that tells you which way to turn your head encodes direction.
- Salience (pulse intensity / emphasis) and priority ranking under overlap are used to encode urgency.
- An icon, which may or may not have a short label (kept to a minimum in the pilot), is used to encode the category.

The mapping is meant to be the same for all sessions so that users can build a solid mental model and don't have to remember things on purpose.

2) Direction, Urgency, and Distance Encoding: The interface is meant to be useful: the direction cues are meant to help you get your bearings quickly, not give you a lot of information. The algorithm can prioritize because urgency separates threats from events that aren't as important. Distance encoding was not necessary for the pilot's dependent variables (reaction time and angular error) and was deliberately deprioritized to prevent increased visual complexity in the initial iteration.

3) Clutter Management Rules: To keep from having too many cues fighting for attention:

- The maximum number of visible cues that can happen at the same time is two.
- When events happen at the same time, the most important one takes precedence.
- Low-confidence events are shown with less importance and for a shorter time.
- Cues have a limited duration and fade out smoothly to keep the HUD clean.

This directly follows the course principle of lowering the load on short-term memory (recognition over recall) and stopping mistakes by using limits.

D. Interaction Design

1) Toggle-Based Interaction: EchoSpaceAR is made to require as little interaction as possible. The main control that is planned is a single **on/off toggle** that turns the overlay on or off. There are two reasons for this: (i) it makes Hick-Hyman choice easier (binary choice), and (ii) it helps safety and recovery by letting the user quickly turn off the overlay if it gets in the way. The pilot study primarily concentrated on perception and orientation performance instead of assessing toggle selection time; nevertheless, the toggle concept is a crucial design element for the subsequent spiral iteration.

2) System States and Error Handling: There are only a few predictable states that the system can be in:

- Listening: keeping an eye on and classifying active events.
- Alerting: A cue is shown with a set amount of time to stay on the screen and fade away.
- Suppressed: the user turned off the overlay, so events may still be logged but cues are not shown.

Confidence-aware rendering lowers misleading salience when things are uncertain. The system fails safely when there are pipeline problems because it doesn't show old cues.

E. Data Logging and Format

Unity keeps track of trial-level data that is needed for analysis:

- participant ID, condition, trial number,
- kind of event and level of urgency,
- timestamps: the start of the event, the start of the cue, and the response marker
- reaction time (ms),
- Localization results: accuracy (0/1), absolute angular error (degrees), success within 15° (0/1),
- Signals for engineering: latency, spikes in latency, and flags for stereo issues.

Data is exported to CSV files and put into objective tables and figures. For example, there are CSV files for the cs449_objective_performance_table.csv, cs449_objective_summary_table.csv, and cs449_issue_summary_by_condition.csv files.

VI. Expert Reviews & Usability Testing (Evaluation)

A. Method

1) Experimental Design: A within-subject design compared:

- No AR feedback is the baseline.
- EchoSpaceAR: AR cues turned on (direction + urgency).

The within-subject structure diminishes inter-individual variability and enhances sensitivity to interface effects. Condition order was balanced to lessen the effects of learning and tiredness.

2) Participants: The participants (N = 15) were hearing individuals with normal or corrected-to-normal vision. The number of people with normal hearing profiles was 11, and the number of people with mild HoH was 4. The mean age was 23.07 years (SD = 1.67), with a range of 22–27 (median = 23). Gender distribution was 10 male and 5 female. Education level was 13 undergraduate and 2 PhD. Self-reported technology adaptation level (1–5) was 5 for 13 participants and 4 for 2 participants. The mean, standard deviation, and median of prior VR experience were 1.05 years, 0.97 years, and 0.7 years, respectively. This supports the idea that there may be impacts of familiarity.

We did not conduct multivariate analysis in this pilot because age range was narrow (22–27) and we did not observe meaningful differences; additionally, technology adaptation level showed a strong ceiling effect (most participants self-reported 5). This is planned for the next spiral evaluation with a more diverse participant range.

3) Apparatus and Setup: The Meta Quest 3 was used for the experiment. Simulated events with scripted guidance were used to set a clear ground truth. The AR HUD showed directions and hints about how urgent something was. Unity kept track of the times of events, cues, reactions, and diagnostic signals like latency, spikes, and stereo flags.

4) Procedure: Participants underwent a brief familiarization phase to acquire cue meanings. After that, they finished blocks in both situations. Participants were told to swiftly and properly turn toward the direction that was shown. After finishing the exercises, the participants filled out the Turkish TAM questionnaire and the SUS.

5) Measures: We got:

- Time it takes to react (ms),
- 0–1 for accuracy,
- mean absolute angular error (in degrees),
- success inside 15° (0–1),
- TAM subscales and item distributions,
- Distribution of SUS scores,
- Engineering diagnostics: delay, latency spikes, and problems with stereo.

6) Analysis: Objective indicators were aggregated at the participant level and analyzed within-subject across circumstances. We present descriptive statistics and the inferential results contained in our exported result summaries. Engineering diagnostics were examined as risk indicators and correlated with performance measures to detect stability concerns affecting usability.

B. Results

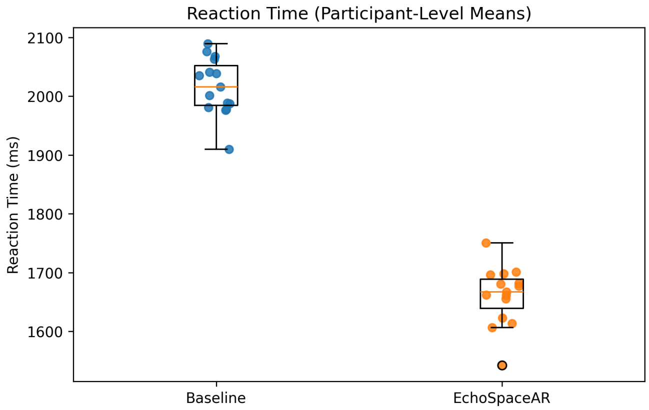
1) Participant Summary: Fifteen pilot participants (N = 15) finished both conditions within the same subject. There were 11 normal hearing profiles and 4 mild HoH hearing profiles. The mean age was 23.07 years (SD = 1.67), range 22–27. Gender distribution was 10 male and 5 female. The mean for years of prior VR experience was 1.05, the SD was 0.97, and the median was 0.7.

Metric	Baseline	EchoSpaceAR	Diff (E−B) [95% CI]	p
Reaction Time (ms)	2016.37 ± 48.40	1661.04 ± 49.31	-355.32 [−382.50, −328.15]	< .001
Accuracy (0–1)	0.62 ± 0.15	0.72 ± 0.11	+0.10 [0.00, 0.19]	= 0.048
Mean Abs. Angular Error (deg)	31.61 ± 7.04	26.21 ± 7.15	-5.41 [−11.16, 0.34]	= 0.063
Success Within 15° (0–1)	0.38 ± 0.12	0.61 ± 0.12	+0.23 [0.13, 0.33]	< .001

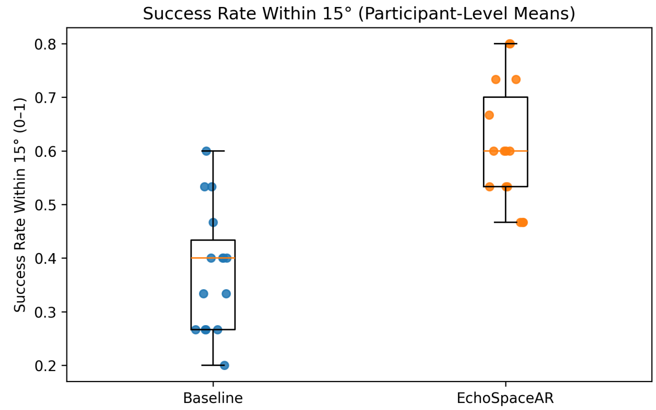
TABLE I: Objective performance (means for each participant; within-subject). Values are presented as mean ± standard error among participants. The EchoSpaceAR condition shortens reaction time and raises the chance of success within a workable 15° tolerance.

2) Objective Performance Table (on the above):

3) Objective Performance Visualizations: To keep things clear and cut down on context switching, figures are placed right next to the results they relate to, not at the end of the section. Full result figures are provided in Appendix B.



(a) Reaction time by condition (average for each subject). The boxplot shows the distribution, and each point shows the average reaction time of one participant. EchoSpaceAR moves the distribution down, which means that people are better at finding their way about with AR cues. This helps to achieve the goal of making decisions and responding faster in tasks that need awareness.



(b) Success within 15° by condition (average of participants). This thresholded metric measures localization that is "good enough" in the real world. EchoSpaceAR makes a big difference, which means that participants are more likely to reach a useable orientation tolerance.

Fig. 1: Key objective visualizations kept in the main report for readability; additional result figures are provided in Appendix B.

4) Technology Acceptance Model (TAM): The TAM questionnaire used in this pilot contains 28 Likert items (1–5) tailored to EchoSpace usage. Items 1–14 measure *Perceived Usefulness (PU)* (e.g., supporting task performance, effectiveness, quality, time saving). Items 25–28 measure *Perceived Ease of Use (PEOU)* (e.g., overall ease, ease of learning/remembering how to do tasks, and guidance). In addition, we report a *Low-Friction* construct that captures interaction smoothness and low cognitive burden, using items that reflect confusion, errors, annoyance, need for a manual, mental effort, rigidity, unexpected behavior, and perceived uselessness (items 15–19, 21, 23–24). These negatively phrased items are reverse-coded so that higher values consistently indicate more positive acceptance. Item-level distributions are provided in Appendix B (Fig. 9) to show which statements received more uncertainty or dispersion, beyond subscale means. In addition to descriptive subscale statistics, we treat the item-level plots as a *qualitative* diagnostic: the goal is not to infer new constructs, but to visually identify which statements show higher dispersion, uncertainty, or disagreement. The full questionnaire (item wording) used in our qualitative usability test is provided in Appendix C.

Responses to the TAM were gathered from N = 15 individuals (28 items). When negatively phrased items were reverse-coded, the subscale descriptive statistics were as follows: Perceived Usefulness (PU) mean = 4.04 (SD = 0.12), Perceived Ease of Use (PEOU) mean = 3.99 (SD = 0.15), and Low-Friction mean = 4.04 (SD = 0.27). These numbers show that people thought the pilot was very useful and easy to use. The TAM figures are provided in Appendix B.

5) System Usability Scale (SUS): We calculated System Usability Scale (SUS) scores for 15 participants. The average SUS score was 66.9 (SD = 6.7), and the 95

6) Engineering Diagnostics: Latency, Spikes, and Stereo Problems: The pilot logs have engineering diagnostic signals that have a direct effect on how easy they are to use. Two things stood out: EchoSpaceAR had more end-to-end latency than the baseline, and there were occasional stereo-issue events that made directional cue quality worse. The diagnostic figures are provided in Appendix B.

C. Discussion

The main idea of EchoSpaceAR is to make people more aware of their surroundings while still keeping their attention. This is shown in the choices we made for our UI. We limit the number of cues and the number of concurrent tasks to lower Hick–Hyman decision overhead. Instead of having several configuration menus that make interactions more expensive, we focus on a single dependable control (planned overlay toggle) that follows Fitts's Law.

The pilot results show that we are reaching our goals. EchoSpaceAR shortens the time it takes to respond and makes it easier to find your way around with AR feedback, even when you're off by 15 degrees. The accuracy goes up a lot, and the angle error goes down, which means that localization precision is getting better. The TAM results reveal that people think the cue language is very useful and easy to use, which is in line with the fact that participants can learn it quickly. SUS says that the usability of an early prototype is okay but could be better. This is to be expected because of XR's limitations and the existing problems with technical stability.

Engineering diagnostics show that system stability is a top priority for usability. Latency spikes make it take longer to respond, and stereo problems make it less accurate. These results are crucial because assistive AR systems depend on user confidence. If cues are late or aren't dependable, users may stop using the system even if the interface is well-designed. So, the next version has to see latency stability and directional dependability as usability problems that need to be fixed.

D. Limitations

The primary drawback is participant mismatch: the pilot employs hearing volunteers instead of DHH users, hence constraining external validity for the target population. Consequently, the study should be seen as a preliminary assessment of the feasibility and usefulness of interaction logic, rather than a conclusive assertion of accessible efficacy. A second disadvantage is that direction was written instead of being guessed from physical acoustics. This was done on purpose to keep the experiment under control, but future field tests should use genuine sensor-driven direction inference to check for ecological validity. A third constraint is the presence of engineering instability (latency spikes, stereo difficulties), which likely diminished the maximum feasible benefit and may affect subjective judgments.

E. Future Work

The spiral model immediately leads to future work: each new cycle should focus on the biggest risks and check that changes have had a positive effect by measuring them. In the next spiral, we will (i) make the pipeline more stable to cut down on latency spikes and lower end-to-end latency, (ii) make directional reliability better to cut down on stereo problems and boost cue trust, and (iii) make clutter management better when events overlap so that the user only sees one clear actionable cue instead of a lot of competing stimuli. At the same time, the assessment has to move toward DHH-centered validation through co-design sessions and usability testing with accessibility partners. This will give us qualitative information on how attention competition works in sign language communication and real-world mobility situations. Lastly, the intended overlay toggle should be put into place and tested directly using Fitts-style metrics (selection time, error rate, throughput). This will make sure that the control is always targetable and can be quickly recovered without adding to the cognitive load.

F. Final Reflection: Design Motto

Designing environmental awareness without relying on sound.

Appendix A
Appendix A: Additional Prototype Figures

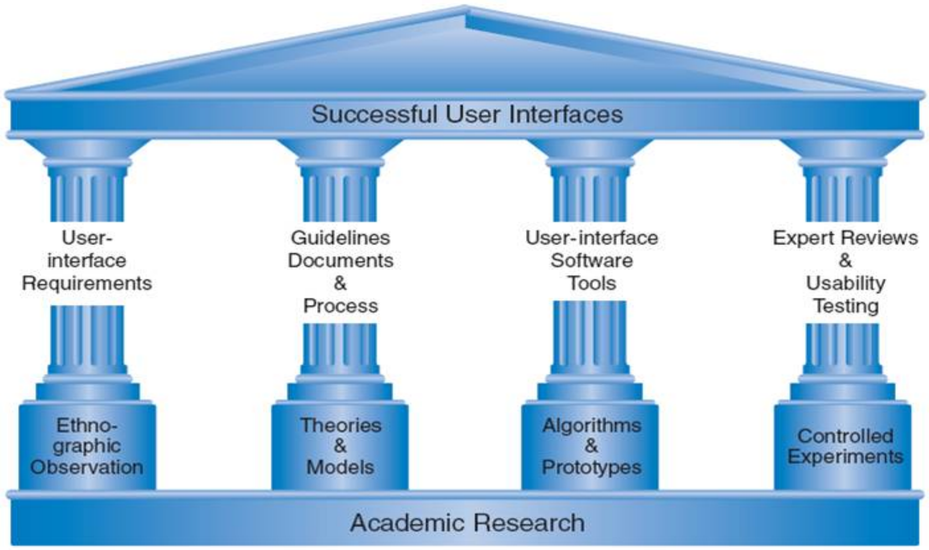
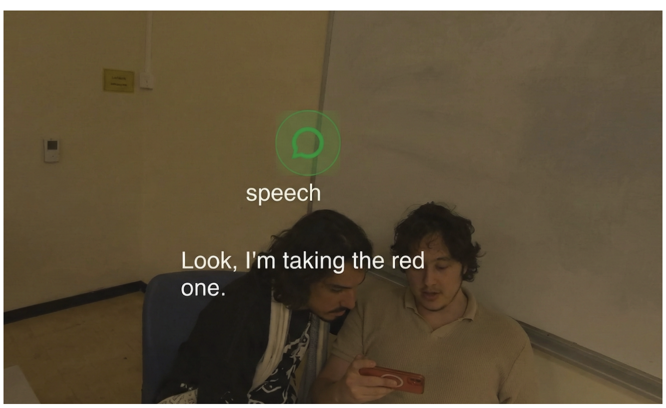


Fig. 2: The "Four Pillars of HCI" framing utilized in CS449/549 course materials are user-interface requirements, guidelines/process, tools, and evaluation (expert reviews and usability testing).



(a) View of the AR HUD in the app. The overlay uses a small set of visual words to show *direction* (where to turn) and *urgency* (how important it is) in a way that keeps things simple and doesn't add to the amount of decisions you have to make. This screenshot shows the "glanceable" design goal: the user should be able to comprehend the cue without having to read a lot of text or figure out how to use the controls.

(b) This is a picture of how the pilot setup works from the outside. This picture shows the real-world setting and limitations (head-worn AR field of view, posture, and distractions in the environment). Showing the physical setup helps us understand later outcomes and engineering diagnostics (such latency spikes and stereo problems) as things that affect usability rather than just technical problems.

Fig. 3: EchoSpaceAR prototype in action: (a) the AR HUD as the user sees it, and (b) the pilot usage context as seen from the outside. These numbers help to make the system description more concrete by linking the interface design to the real-world limits of head-worn AR.

APPENDIX B
APPENDIX B: FULL RESULTS FIGURES

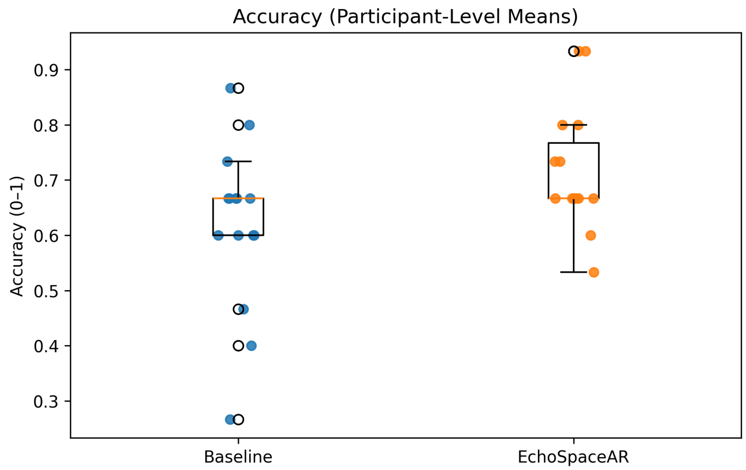


Fig. 4: Accuracy by condition (average of all participants). Most people who use EchoSpaceAR are more accurate. The steady rise shows that the visual-spatial clues help people guess less and chose the right way more often.

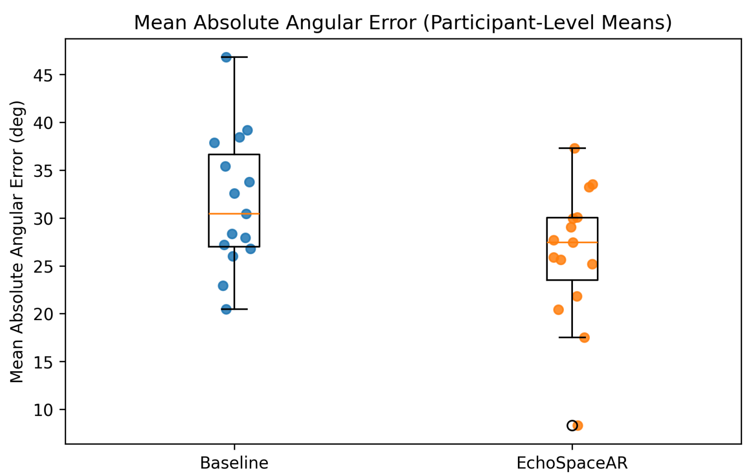


Fig. 5: Mean absolute angular inaccuracy by condition (average for each participant). Lower numbers mean that the location is more accurate. EchoSpaceAR lowers the central tendency of angular error, which shows that the interface increases not only the accuracy of head orientation toward the target direction but also the precision.

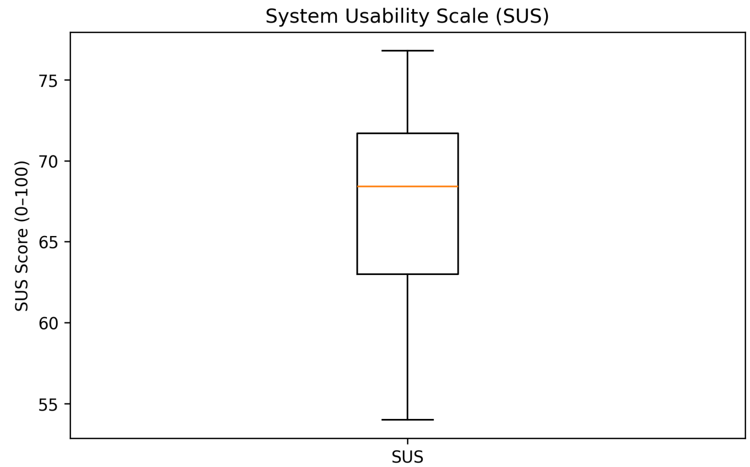


Fig. 6: SUS distribution (boxplot). Scores are mostly in the high 60s, which is a common range for early-stage prototypes that work from start to finish but still have some problems (like latency spikes, cue confidence variation, and XR comfort).

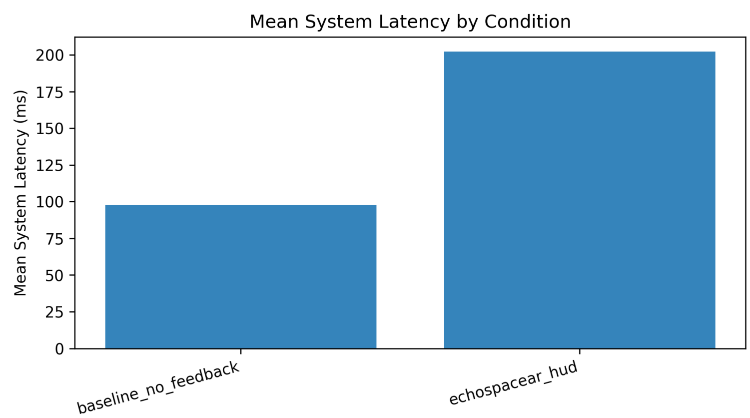


Fig. 7: Latency by condition. The average latency of EchoSpaceAR is higher than the baseline. In awareness tasks, extra latency makes cues less timely, which could shorten the time frame in which the cue can help make a safe choice.

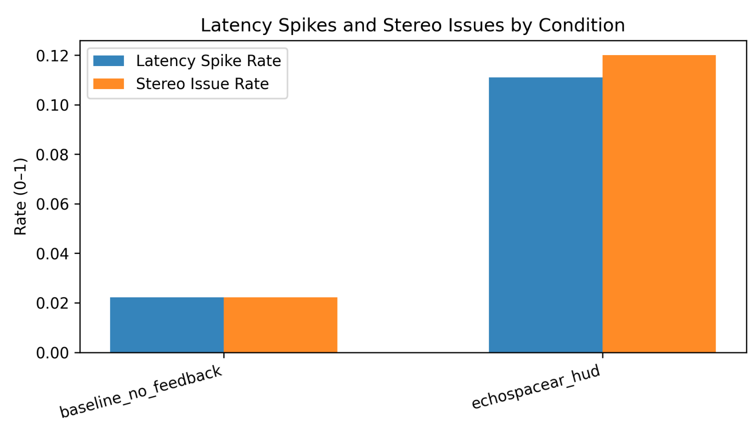


Fig. 8: Latency spike rate and stereo problem rate per situation. EchoSpaceAR has higher spike and problem rates, which means that the pipeline is unstable and could hurt both performance (response time) and confidence (directional reliability).

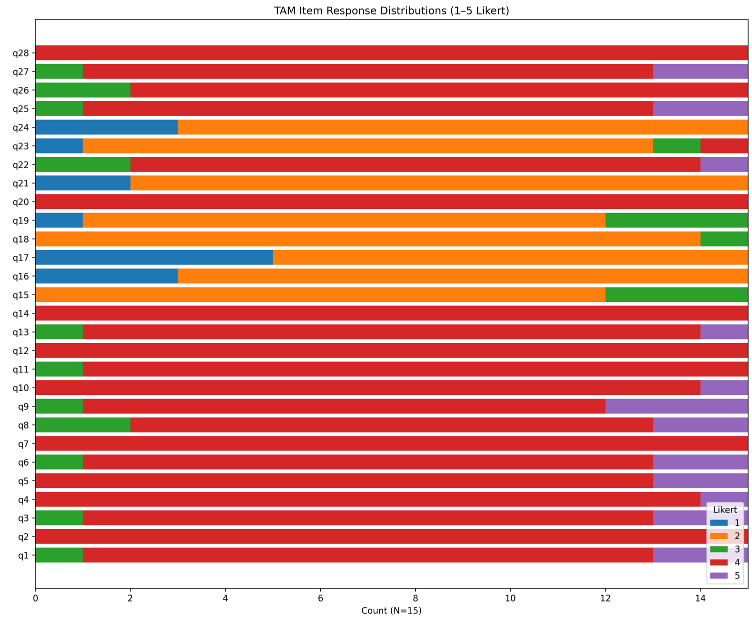


Fig. 9: TAM item distributions (1–5). When scoring, negatively phrased items are reverse-coded so that greater numbers always mean more positive acceptance. The fact that most ratings are higher than average shows that most people think EchoSpaceAR is useful and not too hard to use. Item-level spreads can show you where to focus your efforts in future versions, like where there is uncertainty or rivalry for attention.

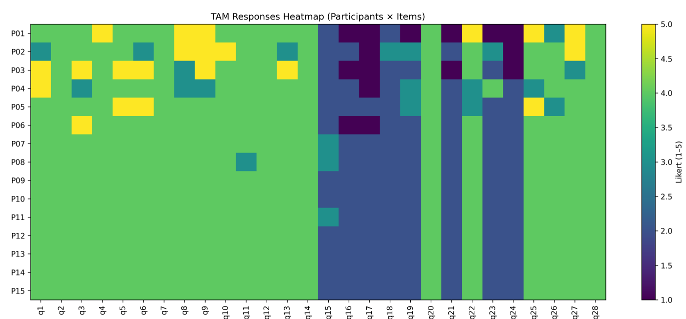


Fig. 10: TAM (1–5) heatmap showing how each participant rated each item. Most rows and columns have warm colors, which means that the evaluations are mostly favorable rather than polarized. Localized colder sections show possible topics where participants don't agree as much; these are helpful for deciding which adjustments to do first (for example, trust under uncertainty or perceived workload during overlaps).

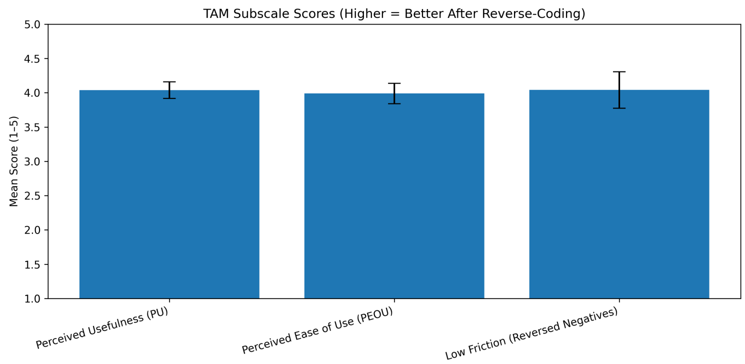


Fig. 11: This is a picture of the TAM subscale (PU, PEOU, Low-Friction) based on what the participants said. High, persistent subscale patterns indicate that the cue language is learnable and that the system is viewed as advantageous. The Low-Friction subscale fits with our design aim of having less interaction and limited clutter.

Appendix C
Appendix C: Qualitative Usability Test

Contents of our qualitative usability test (questions) can be found at the following link:

https://docs.google.com/forms/d/e/
1FAIpQLSdvifbay-hRChZcJHiQ1C5NZVHO05mV58daHyzAHGcSGVddlQ/viewform

Appendix D
Appendix D: Source Code and Supplementary Materials

The full source code, experimental logs, analysis scripts, and build artifacts for EchoSpaceAR are publicly available at:

https://github.com/mesely/SU-CS449-549-Final-Project-EchoSpaceAR.git

REFERENCES

1) Asakura, T., Shigeno, R., & Tokihiro, A. (2023). Development and evaluation of an augmented reality system for recognizing household sounds. *Sensors, 23*(17), 7616. https://doi.org/10.3390/s23177616
2) Bach-y-Rita, P., Tyler, M. E., & Kaczmarek, K. A. (2003). Seeing with the mind. *International Journal of Human-Computer Interaction, 15*(2), 285–295. https://doi.org/10.1207/S15327590IJHC1502_6
3) Benyon, D. (2013). *Designing interactive systems: A comprehensive guide to HCI, interactive design and software engineering* (2nd ed.). Pearson.
4) Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), *Usability evaluation in industry* (pp. 189–194). Taylor & Francis.
5) Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340.
6) Findlater, L., Chinh, B., Jain, D., Froehlich, J. E., Kushalnagar, R., & Lin, A. (2019). Deaf and hard-of-hearing individuals' preferences for wearable and mobile sound awareness technologies. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '19)* (Article 46). Association for Computing Machinery.
7) Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 47*(6), 381–391.
8) Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In P. A. Hancock & N. Meshkati (Eds.), *Human mental workload* (pp. 139–183). North-Holland.
9) Hick, W. E. (1952). On the rate of gain of information. *Quarterly Journal of Experimental Psychology, 4*(1), 11–26.
10) Hyman, R. (1953). Stimulus information as a factor in reaction time. *Journal of Experimental Psychology, 45*(3), 188–196.
11) International Organization for Standardization. (2018). *ISO 9241-11:2018 Ergonomics of human-system interaction—Part 11: Usability: Definitions and concepts*. ISO.
12) International Organization for Standardization. (2019). *ISO 9241-210:2019 Ergonomics of human-system interaction—Part 210: Human-centred design for interactive systems*. ISO.
13) Lee, Y. C., Lin, C. C., Jain, D., et al. (2025). SonoCraftAR: Authoring custom sound-reactive augmented reality experiences for deaf and hard-of-hearing users. *arXiv*. https://arxiv.org/abs/2505.17655
14) Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research methods in human-computer interaction* (2nd ed.). Morgan Kaufmann.
15) Li, Z., Connell, S., Dannels, W., & Peiris, R. (2022). SoundVizVR: Sound indications allowing Deaf or Hard-of-Hearing users to find noises in virtual reality. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '22)* (pp. 1–13). Association for Computing Machinery. https://doi.org/10.1145/3517428.3544817
16) Luna, S., Mojib, S., Tigwell, G., Papangelis, K., & Xu, J. (2024). Investigating visual scanning in augmented reality: Insights from Deaf and Hard of Hearing users. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '24)*. Association for Computing Machinery. https://doi.org/10.1145/3663548.3688535
17) Mirzaei, M., Kan, P., & Kaufmann, H. (2021). Head-up display of spatial sound sources in virtual reality for individuals who are Deaf or Hard of Hearing. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)* (pp. 582–587). IEEE. https://doi.org/10.1109/VR50410.2021.00083

18) Nielsen, J. (1993). *Usability engineering*. Morgan Kaufmann.
19) Norman, D. A. (2013). *The design of everyday things* (Revised and expanded ed.). Basic Books.
20) Preece, J., Rogers, Y., & Sharp, H. (2015). *Interaction design: Beyond human-computer interaction* (4th ed.). Wiley.
21) Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.
22) Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science, 12*(2), 257–285.
23) Sweller, J., Ayres, P., & Kalyuga, S. (2011). *Cognitive load theory*. Springer.