Minuet: Multimodal Interaction with an Internet of Things

Runchang Kang[‡] Anhong Guo[‡] Gierad Laput[¶] Yang Li[§] Xiang 'Anthony' Chen[†]
[†]UCLA HCI Research [‡]Carnegie Mellon University [¶]Apple [§]Google

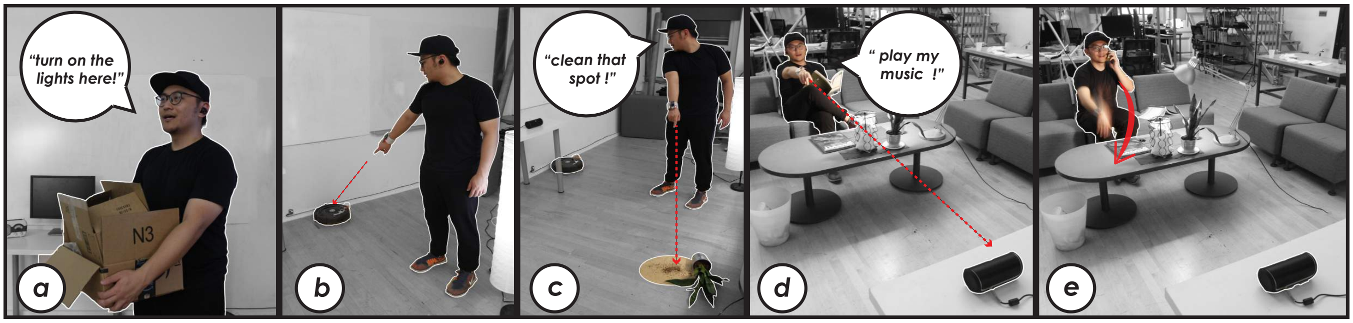

Figure 1: Interaction scenario of Minuet: after returning home, the user turns on the lights in the same room through voice input (a); the user points at the Roomba (b) and then the dirty area to ask Roomba to clean it up (c); the user points at the music player to listen to a personalized playlist (d); finally the user gestures to lower the volume while picking up a phone call (e).

Abstract

A large number of Internet-of-Things (IoT) devices will soon populate our physical environments. Yet, IoT devices' reliance on mobile applications and voice-only assistants as the primary interface limits their scalability and expressiveness. Building off of the classic 'Put-That-There' system, we contribute an exploration of the design space of voice + gesture interaction with spatially-distributed IoT devices. Our design space decomposes users' IoT commands into two components—selection and interaction. We articulate how the permutations of voice and freehand gesture for these two components can complementarily afford interaction possibilities that go beyond current approaches. We instantiate this design space as a proof-of-concept sensing platform and demonstrate a series of novel IoT interaction scenarios, such as making 'dumb' objects smart, commanding robotic appliances, and resolving ambiguous pointing at cluttered devices.

CCS Concepts

• **Human-centered computing → Human computer interaction (HCI)**.

Keywords

Internet-of-Things, multimodal interaction, gesture, voice

ACM Reference Format:
Runchang Kang[‡] Anhong Guo[‡] Gierad Laput[¶] Yang Li[§] Xiang 'Anthony' Chen[†]. 2019. Minuet: Multimodal Interaction with an Internet of Things. In *Symposium on Spatial User Interaction (SUI '19), October 19–20, 2019, New Orleans, LA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357251.3357581

1 INTRODUCTION

Internet-of-Things (IoT) promises a future where homes are populated by smart and connected objects, from intelligent appliances, to automated furniture, to service robots. To control these smart and connected objects, one approach is to develop accompanying apps on personal devices (*e.g.*, phones and watches). However, as the number of IoT devices continues to increase, it costs users unscalable amount of attention to manage and retrieve device-specific control apps.

An alternative approach is interacting with IoT devices through natural language commands via intelligent voice assistants (*e.g.*, Amazon Alexa, Google Home). Voice input alone, however, is limited for issuing commands with spatially-distributed IoT devices. For example, if a person reading a book wants to dim all the lights except for the one above him/her: it would be quite cumbersome to express such intent using natural language commands alone. Further, such verbose commands would be challenging for the assistant to recognize and execute.

To enable spatial expressiveness in interacting with IoT devices, prior work such as Snap-to-It [13] explores connecting to an appliance and controlling it by simply taking a picture. However, it cannot address situations where the IoT device is at a distance or out of sight from the user. To enable remote interaction from afar, WristQue [29], SeleCon [1], and Scenariot [20] employ Ultra-Wide Band (UWB) technology to point at and control IoT devices. While spatially expressive, such freehand gestures alone can be *ambiguous*—pointing at close-by devices can be ambiguous, and gestural

SUI '19, October 19–20, 2019, New Orleans, LA, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6975-6/19/10…$15.00
https://doi.org/10.1145/3357251.3357581

commands can also be ambiguous (*e.g.*, volume up vs. brightness up).

To resolve ambiguity, one approach is multimodal interaction, *e.g.*, combining voice and gesture. The classic 'Put-That-There' system employs voice commands in tandem with pointing gestures to spatially manipulate elements on a large display [8]. Another motivation for adding the gestural modality is for social appropriateness: participants in our formative study mentioned preferences of a quiet input technique in certain situations. Despite the many prior examples and benefits of multimodal interaction, there is still a lack of systematic exploration into the various possibilities of combining voice and gestures in an IoT interaction scenario.

Our research contributes a design space of voice + gesture interaction with spatially-distributed IoT devices. We conducted a formative study to investigate how users would interact with IoT devices using voice and freehand gestures. Findings suggest that users express their intent of controlling IoT devices as two components—*selection* and *interaction*. We illustrate how the permutations of voice and gesture for these two components can complementarily afford interaction possibilities that go beyond existing approaches. Compared to prior work that discusses the general relationships between multiple modalities [30], our design space focuses on a concrete scenario with IoT devices and articulates the interplay between two specific modalities: voice and gesture.

To instantiate the design space, we then developed Minuet—a proof-of-concept sensing platform using UWB and motion sensors to accurately and robustly detect which IoT device a user is pointing at (selection), as well as to recognize the control interaction expressed by voice, gesture, or a combination of both. A technical evaluation shows that Minuet achieved a localization accuracy of $0.330m$ and a low false positive rate in detecting the occurrence of a pointing gesture (one in the entire 45-min study involving 10 participants). Further, to better understand users' performance of the proposed interaction, we measure and model users' freehand pointing behavior using an angular offset: on average participants' pointing in a $6m \times 10m$ room was off the target (IoT devices) by $9°$. Finally, a qualitative study provides empirical evidence on the benefits of using voice + gesture to interact with IoT devices, and reveals areas of improvement in future work.

1.1 Scenario Walkthrough

Fig. 1 illustrates several voice + gesture interaction techniques sampled from our design space and implemented with our proof-of-concept sensing platform. Larry walks into the kitchen carrying several boxes. It is too dim to see, so he says *"turn on the lights here."* Only the kitchen lights up but not the other rooms. Larry notices a plant was knocked over by his dog Bella. Larry points at the Roomba and says *"clean that spot!"* while circling the dirty area on the floor. The Roomba promptly travels to the crime scene and starts cleaning. To cover Roomba's noise, Larry points to a speaker of his whole-home audio system and says *"play my music."* Knowing it is Larry, the system selects a playlist from his favorites. Larry enjoys the music as the Roomba almost finishes cleaning. Suddenly, the phone rings and it is Larry's boss. Hurrying to pick up the call, Larry points to the speaker and waves his hand down to lower the volume.

As demonstrated in this scenario, by combining voice and gesture, users can expressively interact with IoT devices, more so than relying on mobile apps or voice-only assistants.

1.2 Contribution

Our main contribution is a design space that depicts various voice + gesture techniques to solve the long-standing problem of expressively interacting with spatially-distributed IoT devices. Our proof-of-concept system provides concrete implemented examples (as shown in the video figure) from the design space, and further demonstrates the technical feasibility of our proposed interactions.

2 RELATED WORK

Two streams of prior work is related to our research: multimodal interaction and interacting with IoT devices.

2.1 Multimodal Interaction

Multimodal interaction exploits the synergic use of different modalities to optimize how users can accomplish interactive tasks [32]. Early systems such as 'Put-that-there' let a user simply point at a virtual object and literally tell the the system to put that object somewhere else by pointing at a different location on a 2D projected display [8]. Quickset demonstrated a military planning interface that allows a user to pen down at a location on a map and utter the name of a unit (*e.g.*, "red T72 platoon") to place at that location [12].

To optimally combine different modalities, it is important to understand their individual characteristics. In general, pointing and gestures were found to be more useful when specifying targets that are "*perceptually accessible*" to a user while speech is better at specifying "*abstract or discrete actions*" [39]. Oviatt et al. summarized that "*basic subject, verb, and object constituents almost always are spoken, whereas those describing locative information invariably are written or gestured*" [33]. Our work is inspired by these findings: we introduce pointing gesture to complement existing voice assistants' limited capability in specifying multiple spatially-distributed IoT devices. Indeed, as Oviatt et al. pointed out, multimodal input "*occurred most frequently during spatial location commands*" [34], which suggests the potential of multimodally specifying and interacting with IoT devices situated across the physical space.

To interpret multimodal input, it is important to understand the 'interaction' between modalities. Oviatt found that users' natural language input is simplified during multimodal interaction, as the complexity is offloaded to more expressive modalities [31]. Cohen et al. pointed out that direct manipulation can effectively resolve anaphoric ambiguity and enable deixis commonly found in natural language communication [11]. Building off of this work, our system demonstrates how the recognition of freehand pointing catalyzes voice-only interaction, allowing users to explicitly and concisely voice-control a specific IoT device.

2.2 Interacting with IoT Devices

Prior work has explored various techniques and technologies to enable novel interactions with IoT devices, which we categorize below by the style of interaction.

By voice Recent development in natural language processing and cloud computing democratized voice-based interaction with IoT devices, such as Amazon Alexa [2], Google Home [14], and Apple HomePod [3].

By a remote control One of the most popular approaches has been extending the conventional device-specific remote control. Beigl proposed a laser-based universal remote control instrumented with a transceiver so that specific interactions with a given device can be provided to the user on the remote control [6]. Infopoint was a device with a uniform user interface for appliances to work together over a network, such as picking up data from a display and dropping it onto another [23]. Gesture Connect enabled a user to first scan a device's NFC tag and then use physical gesture on a mobile phone to control that device [36]. Improv allowed users to create custom cross-device gestures (*e.g.*, on a smart phone) to control an IoT device on-the-fly by demonstration [10].

By proximity Schilit et al.'s seminal paper on context-aware computing discussed how devices' information, commands and resources can be dynamically made available to a user based on where a user is at, with whom, and what is nearby [38]. Proxemic interaction [5] leveraged the proximity between a user and devices in the environment to fluidly transition between implicit and explicit interaction [22], gradual engagement between personal devices [28], and the control of appliances [26]. Deus EM Machina enabled a mobile phone to identify a close-by IoT device, and to provide custom control for a user to interact with that device [41].

By gaze Zhang et al. augmented an eyewear with an IR emitter to select objects instrumented with IR receivers [42]. To address the 'Midas touch' problem, Velloso et al. let a user gaze at and follow the animation overlaid on a device to engage with it [40].

By camera capture or discovery Going beyond fiducial markers in augmented reality [37], iCam augmented a location-aware handheld device with a camera and a laser range-finder, thus enabling users to annotate surrounding appliances with digital information [35]. Snap-to-It [13] and SnapLink [9] are systems that let a user select and interact with an IoT device by taking a picture of the unmarked device. Heun et al. explored interaction techniques that overlay on-screen virtual controls when the camera is facing a real-world device [19]. Similarly, Scenariot combines SLAM (simultaneous localizing and mapping) and UWB technologies to localize and discover available interactions with surrounding IoT devices [20]. Although requiring instrumenting each individual device with an UWB module, Scenariot's system served as an initial inspiration for our system implementation.

By freehand pointing Perhaps the more related approach to our work is freehand pointing at an IoT device to select and interact with it, which was demonstrated in [1]. However, the drawback is that it requires a dedicated UWB module for each IoT device. To address this problem, WristQue [29] instruments only the user and the environment, and employs motion sensors to infer which device a user is pointing at based on their current location. The problem is that freehand pointing alone can be ambiguous, especially when the devices are too close to each other; once a device is selected, there is gestural ambiguity (*e.g.*, the same swipe-up gesture might refer to increasing brightness or volume). To solve these problems, our system provides interaction techniques for disambiguation, while incorporating the voice modality to increase the expressiveness of specifying IoT devices' control commands.

3 FORMATIVE STUDY

To inform the creation of the design space, we conducted a formative study to investigate how users would remotely (*i.e.*, from a distance) and multimodally interact with IoT devices.

3.1 Participants & Apparatus

We recruited 10 participants (aged 18-28, 6 males, 4 females). Five were native English speakers; eight had experience interacting with intelligent home assistants. Fig. 2 shows an arrangement of eight typical household appliances around a user in a $6m \times 10m = 60m^2$ space. We chose these appliances so that there were both single (*e.g.*, one Roomba, one oven, one AC) and multiple (*e.g.*, four lamps, two electronic door locks) devices with various tasks described below.

3.2 Tasks & Stimuli

Participants were asked to remotely control the appliances while standing at the center of the room. For task design, we considered the distinction between commands (*e.g.*, turn on/off, change volume, clean an area) and information (*e.g.*, when will the coffee be ready, what is the AC's set temperature) based on Schilit et al.'s dichotomy [38]. We also intentionally placed several appliances close to each other, *e.g.*, the lamps were $30cm$ apart from each other (Fig. 2). To avoid priming the participants, we described each command-task [38] by showing the participants a pre-recorded video of an appliance's state change (*e.g.*, a lamp going from off to on), similar to that in PixelTone [25]. Specifically, participants were given the following instructions:

"*Assuming all appliances can understand any of your hand or arm gestures and/or natural language, how would you remotely cause the appliance(s) into the effect shown in the video?*"

Participants were asked to come up with and perform gesture and/or voice input. The studies were video recorded, and participants' comments and responses during the tasks were also gathered.

3.3 Analysis & Results

We found that the way participants expressed their intent of controlling IoT devices almost always consisted of two components: *(i)*

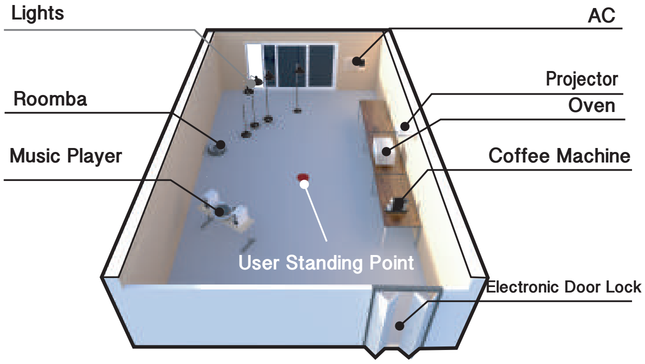

Figure 2: Room and appliances layout in the formative study.

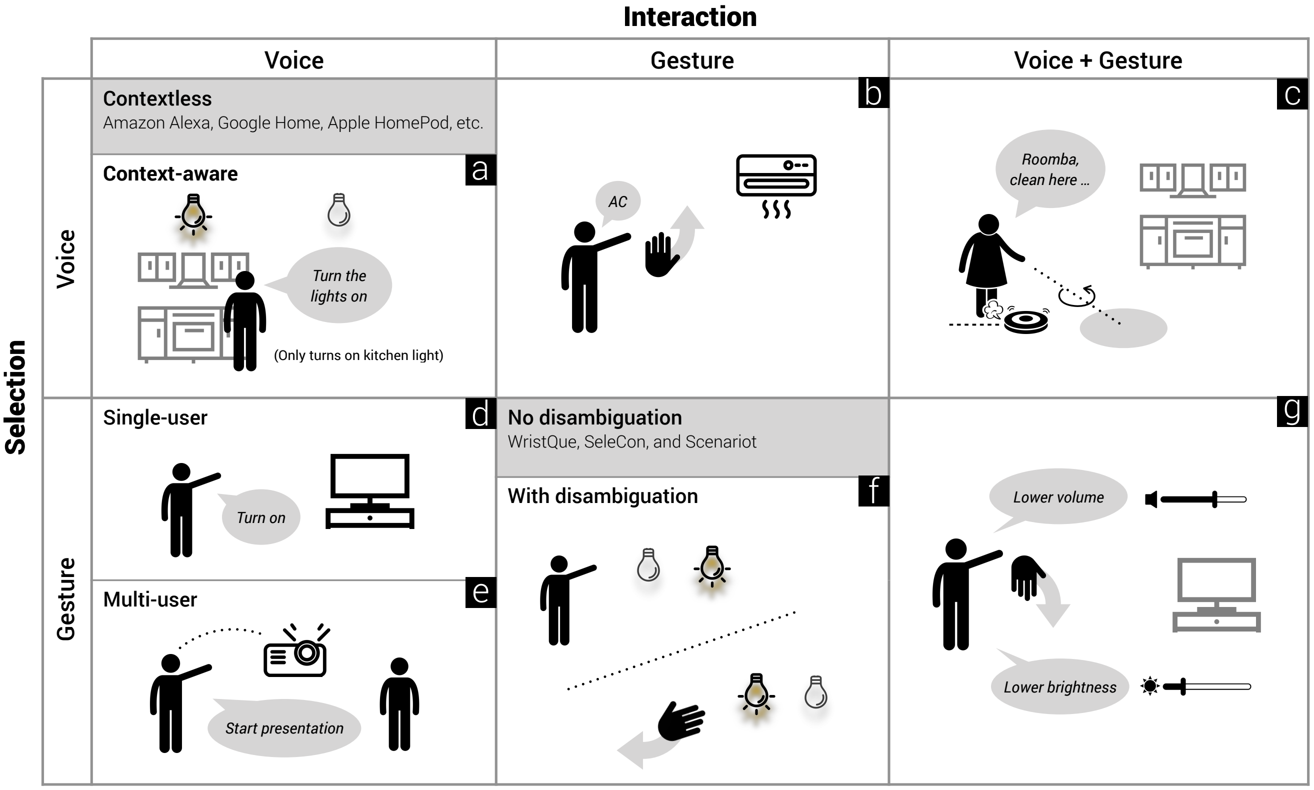

Figure 3: A design space of voice + gesture with spatially-distributed IoT devices. Note that we further subdivide a few cells where we extend voice + gesture with design considerations such as contextual awareness, user identification and disambiguation, as a way to demonstrate the richness and extensibility of our design space.

Selection—specifying a target device, and *(ii)* Interaction—specifying the command or requesting for information. However, participants' preferences varied in using gestures and/or voice as their desired interaction modality.

For selection, eight of the ten participants moved their arm towards the target appliance, although their specific hand gestures differed (pointing, circling, tapping, waving). For interaction (specifying commands or requesting information), six participants interacted with appliances multimodally (*i.e.*, using both gestural and verbal commands) while one participant chose to use gestures only and three others used voice only. Despite their preferences, participants mentioned reasons for using multiple modalities. The most common consideration was social appropriateness, *e.g.*, when in a quiet environment or when having a conversation with others, gestures can complement voice input.

One specific question we were interested in is how participants handled ambiguous situations (*i.e.*, appliances close to each other). Some participants suggested left/right-swipe gesture to switch amongst close-by appliances until the correct one was selected. Some others preferred using relative spatial references in voice input to disambiguate, such as *"turn on the **left** light"*, *"the one in the **middle**"*.

3.4 Implications for Design

- Voice + gesture interaction with IoT devices consists of two components: selection and interaction;
- Freehand pointing (moving arm/hand towards the target) is an intuitive way for selecting a device;
- Verbal and gestural commands should be able to work both interchangeably (to cater to various contexts) and collaboratively (to increase expressiveness);
- The system should clearly indicate ambiguous device selections and allow for follow-up disambiguation either verbally or gesturally.

4 DESIGN SPACE & EXAMPLES

Based on the findings from our formative study, we constructed a design space to lay out various multimodal interaction possibilities with spatially-distributed IoT devices, as well as comparing our work with prior or existing systems. As shown in Fig. 3, the two design dimensions correspond to the two components commonly found in participants' expression of commands to control IoT devices: *selection* of a device and specific *interaction* with the device. We further subdivide a few cells where we extend voice + gesture with design considerations such as contextual awareness, user identification and disambiguation, as a way to demonstrate the richness and extensibility of our design space.

Below we illustrate this design space by walking through each of its 'cells', using examples we later implemented in a proof-of-concept sensing platform (detailed later in §5).

Voice Selects, Voice Interacts This part of the design space is most commonly found in commercial voice-only assistants without

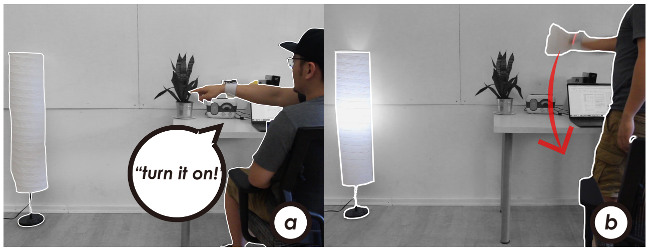

Figure 4: Using voice to turn on a light (a); later when it is night time and quiet, a gesture is more contextually appropriate for turning the light off (b).

any contextual information about the user. However, as localization technology (*e.g.*, using sound or radio) becomes increasingly available, we can expect to add contextual awareness to voice input that allows for an 'unspoken' selection of appliances. For example, as shown in Fig. 3a and implemented in Fig. 1a, a user walking into a dark kitchen asks to turn the lights on; knowing where the user is, the system turns on only the lights in the kitchen but not those in other rooms.

Voice Selects, Gesture Interacts Such a combination works for scenarios where the user can *continuously* gesture their intent with an IoT device, rather than having to repeat a voice command. For example, as shown in Fig. 3b, a user—feeling too cold—can call out "AC" and then keep gesturing the temperature up until s/he feels comfortable. In this case continuous gesturing is more natural and efficient than otherwise having to repetitively say "warmer" or "increase temperature."

Voice Selects, Voice + Gesture Interacts Compared to Fig. 3a, adding gestures to voice interaction further provides spatial references. For example, as shown in Fig. 3c and implemented in Fig. 1b, a user can call out "Roomba", speak out the task ("clean") and gesture to specify spatial references ("here" while gesturing to circle a dirty area on the floor). This part of the design space bridges IoT interaction with related research in robotics, where similar multimodal commands have been developed to control robots [17, 18].

Gesture Selects, Voice Interacts This is the most common way in our formative study to remotely control an IoT device. Importantly, in this cell we illustrate the distinction between single- and multi-user interaction—as shown in Fig. 3e, in a meeting with multiple presentations, each presenter can start their slides by simply pointing at the projector and saying "start presentation", which will be interpreted in conjunction with the user's identity to retrieve the corresponding slides. As shown in Fig. 6, during the presentation, the presenter can point to an audience member, which grants his/her personal device temporary access to navigate to specific slides for questions.

Gesture Selects, Gesture Interacts This combination provides useful options during situational impairment or when voice input might be socially inappropriate (*e.g.*, in a quiet place). For example, when on the phone, a user can point to the music player and swipe down to decrease the volume (Fig. 1e). While prior work has explored similar point-and-gesture interactions [1, 20, 29], the problem of disambiguating close-by devices was never addressed in a real-world setting. In contrast, we propose both a gesture (Fig. 3f) and a voice (discussed in §5) mechanisms for disambiguation: in Fig. 3f, a follow-up gesture 'swipes' through ambiguous candidates until the correct device is selected.

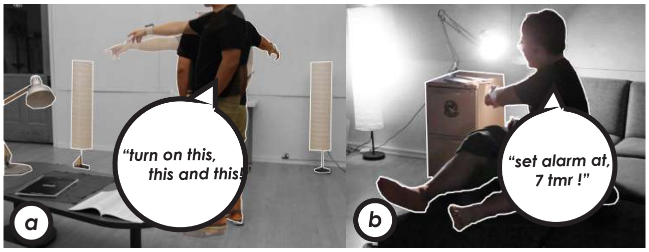

Figure 5: A user can point and select multiple IoT devices in a single sequence of action (a); a built-in alarm clock function can be assigned to a static object, such as a night stand (b).

Gesture Selects, Voice + Gesture Interacts As a user points at an IoT device (*e.g.*, TV), voice is utilized to issue a one-shot command (*e.g.*, lower volume) and a gesture to continuously perform that command (*e.g.*, waving down to keep lowering the volume). Unlike an AC (Fig. 3b), a TV has more control options than a simple up/down gesture can specify (Figure 3g); adding voice here disambiguates the user's intent (*e.g.*, volume up/down vs. brightness up/down).

4.1 More Exemplar Interactions Inspired by the Design Space

Below we showcase more exemplar interactions inspired by, and extended from, the aforementioned design space and implemented using our proof-of-concept sensing platform.

Contextually-appropriate interaction As mentioned in Fig. 3a, the combination of voice and gesture allows for contextually appropriate interaction with IoT devices. As shown in Fig. 4, normally a user can point to a floor lamp and say "turn it on"; however, at night time with family members asleep, a silent gesture is more contextually appropriate when turning the lamp off.

Multi-device selection & disambiguation As shown in Fig. 5a, a user returning home late can say "turn on this, this and this" while pointing to two floor lamps and the table lamp. The system groups these consecutively selected devices in the same input frame, and applies the same action.

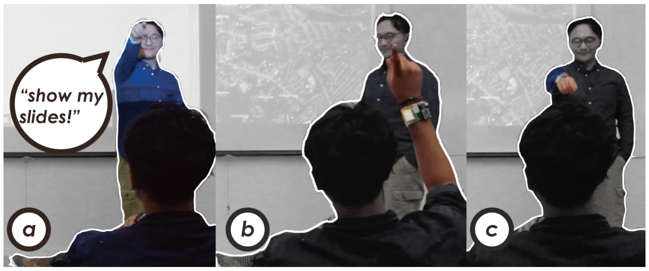

Figure 6: After starting the slides, the presenter points to an audience member who raised his/her hand, granting him/her with temporary access to navigate to specific slides for questions.

Interacting with 'dumb' physical objects Fig. 5b shows an example of interacting with 'dumb' objects as a virtual IoT device. Discussed later in §5, our proof-of-concept system provides an interface for a user to register the position of an object (*e.g.*, a nightstand) or a zone (*e.g.*, the living room). Such objects and zones then become interactive to pointing gesture. For example, as shown in Fig. 5b, a user can point at a nightstand it and say "set alarm 7 am for tomorrow", which effectively turn the physical nightstand to a virtual alarm clock.

Multi-user scenario As shown in Fig. 6, a user points to the projector and uses voice to start a presentation, which can then be controled by gestures. As an audience member raises his/her hand to ask a question, the presenter points at him as an acknowledgement. The synchronized hand-raising and pointing act as a 'handshake', granting the audience member temporary access to the presentation system, *e.g.*, navigating to a specific slide for questions.

5 SYSTEM DESIGN & IMPLEMENTATION

In this section we describe the design and implementation of *Minuet*—a proof-of-concept sensing platform that instantiates our design space. As shown in Fig. 7, the hardware of Minuet consists of instrumented UWB transceivers in the environment; a mobile component that is worn on a user's wrist, which consists of a microcontroller with built-in Wi-Fi and Bluetooth modules, a UWB localization module, an IMU, and a battery. Building on top of this hardware setup, the software can recognize when a user points at an IoT device, which device is pointed at, how to disambiguate, and how to interpret users' subsequent voice and/or gesture input to interact with the selected device.

5.1 Selecting an IoT Device via Pointing

Minuet allows a user to select an IoT device by simply pointing at it, which is enabled by drawing a pointing vector from a user's location to 'intersect' with IoT devices' pre-registered locations (Fig. 8). Below we explain how each of these components works.

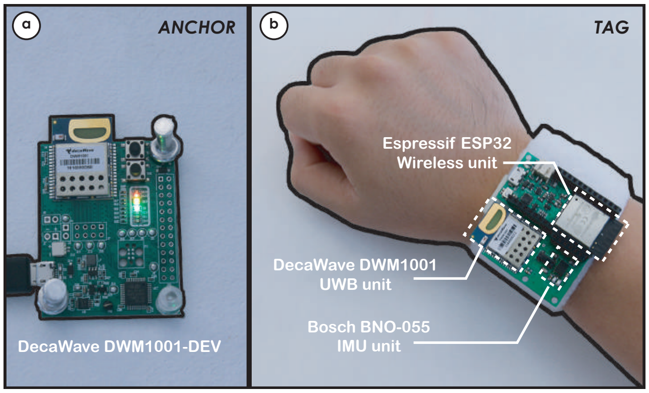

Figure 7: Hardware components of Minuet: four UWB anchors (a) are instrumented in the environment to triangulate a user that wears a UWB tag, an IMU and a wireless module on a wristband (b).

Obtaining locations of users and IoT devices For localizing a user, we use an off-the-shelf Ultra-Wide Band (UWB) Real-Time Localization System[1], which is comprised of four *anchors*—UWB radio transceivers that set up a UWB coordinate system and triangulate a *tag* device worn on the user's wrist. We measured a localization accuracy of 30 *cm* in our lab space (detailed in the next section).

For localizing IoT devices, instead of requiring a laborious scanning process that suffers from noise and errors, Minuet leverages a user's wrist-worn UWB tag and provides a light-weight mechanism to register an IoT device's location. As shown in Fig. 9a, in the registration mode, a user simply uses the hand that wears the UWB tag to tap at one or multiple points over a device's surface (*e.g.*, the vertices of its bounding box); the system then automatically computes the spatial and geometric representation of the IoT device. Even if the device is later moved to a different location, with a few taps the user can easily update its spatial information. Similarly, the user can also spatially register different zones (*e.g.*, kitchen, living room) by walking around their enclosing space in the registration mode (Fig. 9b).

Detecting when a pointing gesture occurs Prior work has explored thresholding the magnitudes of accelerometer and gyroscope to detect the onset of pointing [1]. The problem is that the high threshold value requires a user to extend their arm very fast in order for the pointing to be recognized. To overcome this problem we take a data-driven approach: as a pointing gesture is performed, we collect raw data from the accelerometer and gyroscope over an empirically defined 1.5 second window. We trained a Random Forest [27] classifier over 160 examples of pointing gestures (varied in directions and speeds) and 900 samples of inactivity (*e.g.*, walking around, hand waving, typing on a computer, reading books). Our approach runs a sliding time window to 'pick up' a user's pointing gesture, producing only one false positive over a collective 45 minutes of inactivity[2] (10 participants × 270 seconds per participant).

Detecting which device is pointed at As the system detects the occurrence of a pointing gesture, it first computes the pointing direction $\mathbf{v}$ computed by transforming IMU's absolute pointing

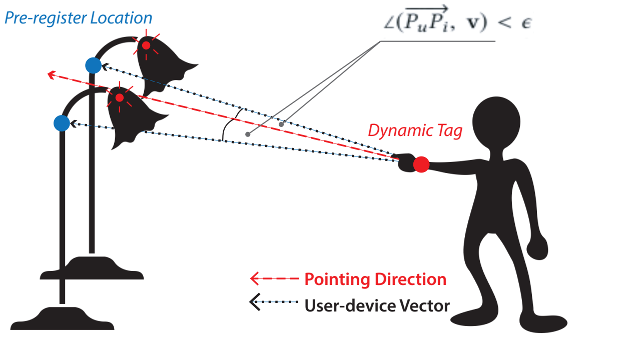

Figure 8: Close-by devices are too ambiguous to point and select; we first compute a set of ambiguous devices, which is further filtered with gesture (*e.g.*, a left-swipe) or verbal disambiguation (*e.g.*, "the left one").

[1] https://www.decawave.com/sites/default/files/dwm1001_system_overview.pdf
[2] We asked participants to perform eight different common daily indoor activities which were adapted from [24]

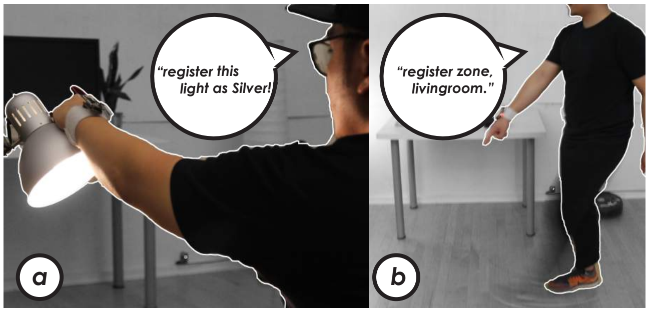

Figure 9: A user can tap at an IoT device and use voice input to register a name and the spatial location of that device (a); further, a zone can be registered by walking around a specific region after entering the registration mode (b).

orientation (in Euler angles) to a vector in the aforementioned UWB coordinate system. Then based on the user's current location P_u and the pre-registered locations of the IoT devices $P_1, P_2, ..., P_N$, the system computes $\arg\min_i \angle(\overrightarrow{P_u P_i},\ \mathbf{v})$ to select an IoT device that intersects with the user's pointing direction. However, as illustrated in Fig. 8, the problem is ambiguity: for devices that appear too close to each other from a user's point of view, it would be difficult to point exactly at one specific device. Below we discuss Minuet's solution for disambiguation. Although prior work in Virtual Reality (*cf.* [4]) has explored solutions to a similar problem (*i.e.*, pointing at virtual objects in 3D), little is known about how disambiguation can be achieved when pointing at real-world objects (*i.e.*, spatially-distributed IoT devices) and how to model a user's performance. Below we discuss our approach based on [4].

Disambiguation We first find a set of ambiguously selectable devices

$$S_{\text{ambiguous}} = \{i \mid \angle(\overrightarrow{P_u P_i},\ \mathbf{v}) < \epsilon\} \quad (1)$$

The value of ϵ is obtained and discussed later in our technical evaluation. When there is more than one device in the ambiguous set, Minuet enters an disambiguation mode, indicated by a blinking LED light[3] attached to each of the ambiguous devices. Next, inspired by the formative study, Minuet allows the user to verbally disambiguate using spatial references, *e.g.*, "the left one" or "the second one from the left." To enable spatial references, our system computes the spatial relationship between each ambiguous devices based on their relative locations to the user. Alternatively, users can swipe left or right to go through the set of devices, similar to switching amongst different application windows in the OS.

5.2 Interacting with an IoT Device Multimodally

Once an IoT device is selected (possibly with disambiguation), the user can immediately speak a voice command or continually perform a hand gesture followed by pointing.

Processing voice input As with most commercial voice assistants, we stream voice input through a Bluetooth-enabled wireless ear piece (although other approaches are possible). Users' speech is then converted to text [16] and parsed as a syntax tree [15] (*e.g.*, Fig. 10 shows a representation of the command "increase the volume").

Processing gesture input Currently, Minuet supports six different gestures (swiping up/down/left/right, wrist rotation clockwise / counter-clockwise) aggregatively elicited from participants' input in our formative study. Similar to pointing recognition, we take a data-driven approach to recognize the onset of each gesture, which is trained with 100 demonstrative examples and 1750 examples of non-activities. Our Random Forest classifier achieved 98.59% in a 10-fold cross-validation.

Mapping voice + gesture input to controlling IoT devices We take a frame-based approach [31] to handle voice+gesture input. As a user points at a device, the system creates an empty frame, *e.g.*, `{device: [speaker_2]; property: []; action: []; parameter: []}`, and starts looking for co-occurred or subsequent voice or gesture input to fill in the frame.

For voice input, we compare the parsed dependency graph with the selected device's repertoire of executable verbal commands, which we constructed from the corpus collected in the formative study. For example, the comparison will match "increase the volume" to "raise the volume" by the DOBJ 'volume' while considering the synonymous ROOT phrases 'increase' and 'raise.' As a result, the system will update the frame `{property: volume; action: increase; parameter: '5%'}`[4].

For gesture input, we map our gesture set to each IoT device's most common controls. Up/down/left/right swipes perform discrete actions (*e.g.*, swiping up to a speaker increases the volume by a default delta of 5%) while rotation enables continuous control (*e.g.*, counter-clockwise to continuously lower the volume).

For voice+gesture input, *e.g.*, swiping up towards a TV and saying "volume", the system will partially fill up the frame until it has enough information to be mapped to a specific command on the selected device. For example, if the system first detects a swipe-up gesture, it will realize that for the TV this gesture only maps to `increase(5%)` but does not specify a property (because there are more than one applicable property, *e.g.*, volume and brightness). Thus the system keeps waiting for further information, *e.g.*, a subsequent voice input of "volume", which now enables the system to fill the `property` slot and to execute a specific command on the TV.

6 TECHNICAL EVALUATION

We conducted a technical evaluation investigating: *(i)* how accurately the system can locate a user; and *(ii)* how accurately a user can point at a target appliance (*i.e.*, $\angle(\overrightarrow{P_u P_{\text{target}}}, \mathbf{v})$). The focus of our evaluation is to demonstrate the technical feasibility of our proposed voice + gesture interaction with IoT devices; a longitudinal field study to fully understand the reliability of the proof-of-concept

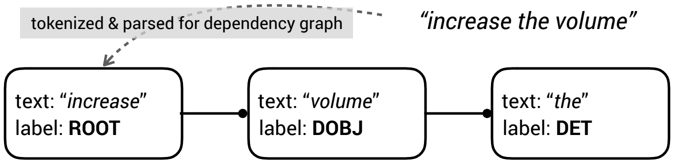

Figure 10: An exemplar parse tree of a voice input.

[3] Or in some cases, simply light up all ambiguous lamps or make a sound from all ambiguous speakers.

[4] '5%' is the default value when the parameter is not explicitly specified.

system is beyond the scope of our paper, which we plan to address in future work.

6.1 Participants & Apparatus

We recruited 10 users to participate in our evaluation: six males, four females, aged 21 to 30, all but three had experience using home assistants, and two are native English speakers. As shown in Fig. 11, we used the same lab space in the formative study. A $4m \times 5m$ grid (each cell $1m \times 1m$) was marked as references when sampling the measurement. We selected 5 common household appliances spatially distributed in 3D space (Fig. 11).

6.2 Localization and Pointing Accuracy

As localization is the immediate step that precedes pointing, we conducted the first two measurements in the same set of tasks where participants were asked to point at each appliance at various locations.

We first introduced the pointing gesture: participants were asked to extend their arm towards an appliance, similar to our everyday behavior of referring to an object from afar or showing direction to someone else.

Participants were standing while performing tasks throughout the study. We assigned each participant six random locations from the grid (we did, however, balanced the number of participants' pointing trials at each intersection to ensure a uniform sampling across the entire grid). At each location, each participant performed two rounds of pointing tasks for all five spatially-distributed IoT devices before moving on to the next location. We did not provide feedback on whether or not a appliance was successfully selected; instead a simple beeping sound was played to indicate the completion of each trial. In total, we collected: 10 participants × 6 locations per participant × 2 rounds per location × 5 appliances to point at for each round = 600 data points.

6.3 Analysis & Results

For localization, across all 600 pointing trials, we calculated the mean error between the ground truth coordinate values[5] and the localization results. As the Z value was fixed for each participant (standing), we entered it manually into the system. The overall mean localization error across all participants was $0.330m$ (SD = $0.169m$).

For pointing detection, 12 out of 600 pointing trials were *not* picked up by the system (2% overall). For the other 98% of the examples, Fig. 12 shows the distribution of $\angle(\overrightarrow{P_u P_{\text{target}}}, \mathbf{v})$—that is, how much each participant's pointing deviated from the target appliance, measured by the angle between the participant-device vector and the participant's pointing vector. The result shows that on average participants' pointing was off about 9.0° (SD=4.7°) and collectively 95% of the pointing trials fell within a margin of error of 17.7°. These findings inform the choice of ϵ's value when deploying Equation 1 for disambiguation.

[5] Measured by an ACKLIFE 40M Advanced Laser Distance Meter.

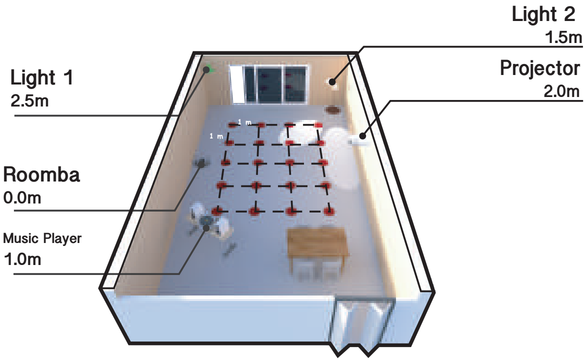

Figure 11: Room and appliances layout for our technical evaluation (numbers below appliance names are heights).

7 QUALITATIVE STUDY

We invited users to try out our system and gathered their initial reactions and feedback.

7.1 Participants & Apparatus

We recruited the same 10 participants in the technical evaluation, as they were already familiar with the pointing part of our system. The appliances and their spatial layout remained the same (Fig. 11).

7.2 Tasks & Stimuli

Before the tasks started, we provided a quick tutorial of Minuet for each participant, including the six gestures and exemplar voice commands. Participants were free to practice how to interact with Minuet multimodally for five minuets. As shown below, the main tasks consisted of asking participants to act out three representative scenarios that encompass interaction examples distributed in our design space. The bolded texts are the specific tasks participants performed.

*Task 1 It is reading time. As soon you as you enter the room you realize it is too dim so you **want the wall-mounted light on**, which is located on the other side of the room above an lounge chair. You then sit down and read. However, soon you feel sleepy and want to take a snap. You **want the light off** without having to stand up and reach its switch.*

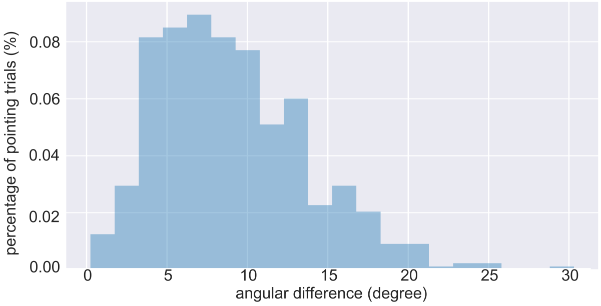

Figure 12: The distribution of angular difference—how much participants' pointing deviated from the target device—amongst all pointing trials.

Task II *Your dog knocked over a plant and now the floor is covered with soil. You* ***command the Roomba to clean the dirty area****. The Roomba is loud so you* ***want the music player to play some music*** *to cover the noise.*

Task III *Having finished your slides for tomorrow's meeting, you want to rehearse it. You* ***start the presentation using a public projector*** *in the lab. The slides look good and it is time to go home. However, before walking out of the room, you realize you just need to* ***turn off the light above your desk but not the light right next to it****, which is for your neighboring lab mate who is still working.*

Immediately after each task, we asked the participant to comment on both the interaction techniques and the system: *(i)* whether the voice + gesture interaction style was useful in the tested scenario; *(ii)* whether the system performed well as you expected; and *(iii)* whether the system was easy to use. The entire study was video and audio recorded.

7.3 Analysis & Results

To analyze the data, we used a method akin to the Affinity Diagram [7]: we organized notes of participants' comments to iteratively develop meaningful and coherent themes that captured their reactions to both the system and the underlying idea of voice + gesture interaction with IoT devices.

In general, participants (P1-P10) responded positively to many of our proposed interaction techniques while also pointing out issues and concerns with some others.

Overall, the participants welcomed the idea of voice + gesture interaction with IoT devices: *"I like the idea of supporting both voice and gesture command"* (P4); *"The gestures are very logical, the system understands my verbal commands well."* (P5); *"The voice commands are flexible"* (P6). Some pointed out that Minuet complements existing voice assistants: *"Compare with Alexa, I don't have to talk if I don't want to [but still be able to control]."* (P7). *"Easy object classification without naming or grouping—imagine you have 50 light bulbs, you can control through embodied interaction"* (P2). Pointing, in particular, was considered an *"intuitive"* (P1) and expressive—*"You can point to the device you want to control"* (P4), *"Pointing to start makes me feel easy"* (P6). Moreover, our scalable and extensible system also raised some participants' interest. *"Handling multiple devices in one system is important"* (P7), *"This system can realize multiple devices controls with one device"* (P1).

Participants also mentioned three main areas for improvement, mostly related to implementation: *(i)* Shorten the pointing recognition time (P[1,3,6,10]): currently we employ a 1.5 second time window (§5) to accommodate for people's varied speeds of pointing motion; future work will experiment using time windows of multiple sizes to increase responsiveness for fast pointing behaviors. *(ii)* Robustness of speech recognition: eight of our participants were *not* native English speakers and two in particular struggled to be understood by the speech recognition engine, which, in the future, could be improved by tailoring the recognition to individual users. *(iii)* Integrating a microphone into the wearable platform: currently to ensure quality we use a pair of Bluetooth ear pieces for sending/receiving audio, which can be replaced with an on-board component in our next iteration.

8 DISCUSSIONS

We discuss the limitations and directions for future work.

Feedback to users Currently Minuet provides simple audio and visual feedback to users. Our future work will explore other types of feedback, *e.g.*, haptics that are amenable to be added on Minuet's wearable modules. We will also investigate how to provide sufficient information without slowing down or distracting users' interaction with the IoT devices.

Localization Accuracy As reported in §6, the localization position accuracy of our UWB system is $0.330m$. The error might be induced by various factors, including but not limited to: anchors' geometrical arrangement, and obstacles in the environment (*e.g.*, furniture). Our future work will apply external filters while streaming data to improve accuracy.

Our current UWB system provided by DecaWave has a known Z accuracy problem [21]. While it is beyond the scope of our research to debug the product, we plan to find data-driven solutions to mitigate this problem, or explore alternative localization systems for implementation.

Power consumption Compare with IMU sensors, the UWB module is highly power-consuming. From our test result, a 500 *mAh* battery can support 1.5 hours of continuous usage. Our future work will explore using the IMU sensors to wake up the UWB only when movement is detected.

IoT devices moved? Currently we only support static registration of an IoT device and do not track a device if it is moved to a new position. Nevertheless, updating an IoT's position is easy, as the user can simply redo the registration step (Fig. 9).

Fatigue and accessibility Fatigue is a known problem for freehand gestures. Although our participants did not report any fatigue during the study, such problem might still occur if freehand gestures are used at a regular basis. A related issue is accessibility—freehand gestures become unavailable when users have motor difficulty (*e.g.*, shoulder injury, arthritis). To address both issues, we plan to explore the design of voice + alternate 'hands-free' pointing techniques, *e.g.*, using head pose and gaze to interact with IoT devices.

Integrating with smartphones We implemented the mobile module of Minuet as a wearable device; however, it is possible to integrate it with other personal devices, such as a smart phone. Leveraging the phones' existing IMU, our future work will integrate a lightweight UWB tag as part of a phone case, which will turn the phone into a universal IoT remote control, enabling Snap-to-It [13] style interaction even with IoT devices at a distance.

Virtual canvas for creativity Similar to spatially registering an IoT device or a static object, Minuet can allow a user to create a virtual canvas on any surface (provided there is a uniformly accurate X/Y/Z localization). A user can simply tap at two diagonal reference points on a wall to define a rectangular canvas. The tag's relative positions with the reference points enables interactive touch interaction on the wall, similar to interaction styles demonstrated in Wall++ [43].

REFERENCES

[1] Amr Alanwar, Moustafa Alzantot, Bo-Jhang Ho, Paul Martin, and Mani Srivastava. 2017. SeleCon: Scalable IoT Device Selection and Control Using Hand Gestures. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation (IoTDI '17)*. ACM, New York, NY, USA, 47–58. https://doi.

org/10.1145/3054977.3054981

[2] Amazon. 2019. Amazon Alexa. https://developer.amazon.com/alexa

[3] Apple. 2019. HomePod. https://www.apple.com/homepod/

[4] Ferran Argelaguet and Carlos Andujar. 2013. A survey of 3D object selection techniques for virtual environments. *Computers & Graphics* 37, 3 (2013), 121–136.

[5] Till Ballendat, Nicolai Marquardt, and Saul Greenberg. 2010. Proxemic Interaction: Designing for a Proximity and Orientation-aware Environment. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, New York, NY, USA, 121–130. https://doi.org/10.1145/1936652.1936676

[6] Michael Beigl. 1999. Point & Click - Interaction in Smart Environments. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing (HUC '99)*. Springer-Verlag, London, UK, UK, 311–313. http://dl.acm.org/citation.cfm?id=647985.743710

[7] Hugh Beyer and Karen Holtzblatt. 1997. *Contextual design: defining customer-centered systems*. Elsevier.

[8] Richard A. Bolt. 1980. &Ldquo;Put-that-there&Rdquo;: Voice and Gesture at the Graphics Interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '80)*. ACM, New York, NY, USA, 262–270. https://doi.org/10.1145/800250.807503

[9] Kaifei Chen, Jonathan Fürst, John Kolb, Hyung-Sin Kim, Xin Jin, David E. Culler, and Randy H. Katz. 2018. SnapLink: Fast and Accurate Vision-Based Appliance Control in Large Commercial Buildings. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 129 (Jan. 2018), 27 pages. https://doi.org/10.1145/3161173

[10] Xiang 'Anthony' Chen and Yang Li. 2017. Improv: An Input Framework for Improvising Cross-Device Interaction by Demonstration. *ACM Trans. Comput.-Hum. Interact.* 24, 2, Article 15 (April 2017), 21 pages. https://doi.org/10.1145/3057862

[11] PR Cohen, M Darlymple, FCN Pereira, JW Sullivan, RA Gargan Jr, JL Schlossberg, and SW Tyler. [n.d.]. Synergic use of direct manipulation and natural language. In *Proc. Conf. human Factors in Computing Systems (CHI'89)*. 227–233.

[12] Philip R. Cohen, Michael Johnston, David McGee, Sharon Oviatt, Jay Pittman, Ira Smith, Liang Chen, and Josh Clow. 1997. QuickSet: Multimodal Interaction for Distributed Applications. In *Proceedings of the Fifth ACM International Conference on Multimedia (MULTIMEDIA '97)*. ACM, New York, NY, USA, 31–40. https://doi.org/10.1145/266180.266328

[13] Adrian A. de Freitas, Michael Nebeling, Xiang 'Anthony' Chen, Junrui Yang, Akshaye Shreenithi Kirupa Karthikeyan Ranithangam, and Anind K. Dey. 2016. Snap-To-It: A User-Inspired Platform for Opportunistic Device Interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5909–5920. https://doi.org/10.1145/2858036.2858177

[14] Google. 2019. Google Home - Smart Speaker & Home Assistant - Google Store. https://store.google.com/us/product/google_home

[15] Google Cloud. 2019. Cloud Natural Language. https://cloud.google.com/natural-language/

[16] Google Cloud. 2019. Cloud Speech-to-Text - Speech Recognition Cloud. https://cloud.google.com/speech-to-text/

[17] Boris Gromov, Luca M Gambardella, and Gianni A Di Caro. 2016. Wearable multi-modal interface for human multi-robot interaction. In *Safety, Security, and Rescue Robotics (SSRR), 2016 IEEE International Symposium on*. IEEE, 240–245.

[18] Boris Gromov, Luca M Gambardella, and Alessandro Giusti. 2018. Robot Identification and Localization with Pointing Gestures. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3921–3928.

[19] Valentin Heun, Shunichi Kasahara, and Pattie Maes. 2013. Smarter objects: using AR technology to program physical objects and their interactions. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. ACM, 961–966.

[20] Ke Huo, Yuanzhi Cao, Sang Ho Yoon, Zhuangying Xu, Guiming Chen, and Karthik Ramani. 2018. Scenariot: Spatially Mapping Smart Things Within Augmented Reality Scenes. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 219, 13 pages. https://doi.org/10.1145/3173574.3173793

[21] Antonio Ramón Jiménez and Fernando Seco. 2016. Comparing Decawave and Bespoon UWB location systems: Indoor/outdoor performance analysis.. In *IPIN*. 1–8.

[22] Wendy Ju, Brian A. Lee, and Scott R. Klemmer. 2008. Range: Exploring Implicit Interaction Through Electronic Whiteboard Design. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work (CSCW '08)*. ACM, New York, NY, USA, 17–26. https://doi.org/10.1145/1460563.1460569

[23] Naohiko Kohtake, Jun Rekimoto, and Yuichiro Anzai. 2001. InfoPoint: A Device That Provides a Uniform User Interface to Allow Appliances to Work Together over a Network. *Personal Ubiquitous Comput.* 5, 4 (Jan. 2001), 264–274. https://doi.org/10.1007/s007790170005

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

[25] Gierad P. Laput, Mira Dontcheva, Gregg Wilensky, Walter Chang, Aseem Agarwala, Jason Linder, and Eytan Adar. 2013. PixelTone: A Multimodal Interface for Image Editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2185–2194. https://doi.org/10.1145/2470654.2481301

[26] David Ledo, Saul Greenberg, Nicolai Marquardt, and Sebastian Boring. 2015. Proxemic-Aware Controls: Designing Remote Controls for Ubiquitous Computing Ecologies. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, New York, NY, USA, 187–198. https://doi.org/10.1145/2785830.2785871

[27] Andy Liaw, Matthew Wiener, et al. 2002. Classification and regression by randomForest. *R news* 2, 3 (2002), 18–22.

[28] Nicolai Marquardt, Till Ballendat, Sebastian Boring, Saul Greenberg, and Ken Hinckley. 2012. Gradual Engagement: Facilitating Information Exchange Between Digital Devices As a Function of Proximity. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*. ACM, New York, NY, USA, 31–40. https://doi.org/10.1145/2396636.2396642

[29] B. D. Mayton, N. Zhao, M. Aldrich, N. Gillian, and J. A. Paradiso. 2013. WristQue: A personal sensor wristband. In *2013 IEEE International Conference on Body Sensor Networks*. 1–6. https://doi.org/10.1109/BSN.2013.6575483

[30] Laurence Nigay and Joëlle Coutaz. 1993. A design space for multimodal systems: concurrent processing and data fusion. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*. ACM, 172–178.

[31] Sharon Oviatt. 1999. Mutual Disambiguation of Recognition Errors in a Multimodel Architecture. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, New York, NY, USA, 576–583. https://doi.org/10.1145/302979.303163

[32] Sharon Oviatt. 1999. Ten Myths of Multimodal Interaction. *Commun. ACM* 42, 11 (Nov. 1999), 74–81. https://doi.org/10.1145/319382.319398

[33] Sharon Oviatt, Phil Cohen, Lizhong Wu, Lisbeth Duncan, Bernhard Suhm, Josh Bers, Thomas Holzman, Terry Winograd, James Landay, Jim Larson, et al. 2000. Designing the user interface for multimodal speech and pen-based gesture applications: state-of-the-art systems and future research directions. *Human-computer interaction* 15, 4 (2000), 263–322.

[34] Sharon Oviatt, Antonella DeAngeli, and Karen Kuhn. 1997. Integration and Synchronization of Input Modes During Multimodal Human-computer Interaction. In *Referring Phenomena in a Multimedia Context and Their Computational Treatment (ReferringPhenomena '97)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1–13. http://dl.acm.org/citation.cfm?id=1621585.1621587

[35] Shwetak N Patel, Jun Rekimoto, and Gregory D Abowd. 2006. icam: Precise at-a-distance interaction in the physical environment. In *International Conference on Pervasive Computing*. Springer, 272–287.

[36] Trevor Pering, Yaw Anokwa, and Roy Want. 2007. Gesture Connect: Facilitating Tangible Interaction with a Flick of the Wrist. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, New York, NY, USA, 259–262. https://doi.org/10.1145/1226969.1227022

[37] Jun Rekimoto and Katashi Nagao. 1995. The World Through the Computer: Computer Augmented Interaction with Real World Environments. In *Proceedings of the 8th Annual ACM Symposium on User Interface and Software Technology (UIST '95)*. ACM, New York, NY, USA, 29–36. https://doi.org/10.1145/215585.215639

[38] Bill Schilit, Norman Adams, and Roy Want. 1994. Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. Proceedings., Workshop on*. IEEE, 85–90.

[39] EDWARD TSE, SAUL GREENBERG, CHIA SHEN, and CLIFTON FORLINES. 2007. Multimodal Multiplayer Tabletop Gaming. *Comput. Entertain.* 5, 2, Article 12 (April 2007). https://doi.org/10.1145/1279540.1279552

[40] Eduardo Velloso, Markus Wirth, Christian Weichel, Augusto Esteves, and Hans Gellersen. 2016. AmbiGaze: Direct Control of Ambient Devices by Gaze. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 812–817. https://doi.org/10.1145/2901790.2901867

[41] Robert Xiao, Gierad Laput, Yang Zhang, and Chris Harrison. 2017. Deus EM Machina: On-Touch Contextual Functionality for Smart IoT Appliances. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4000–4008. https://doi.org/10.1145/3025453.3025828

[42] Ben Zhang, Yu-Hsiang Chen, Claire Tuna, Achal Dave, Yang Li, Edward Lee, and Björn Hartmann. 2014. HOBS: Head Orientation-based Selection in Physical Spaces. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction (SUI '14)*. ACM, New York, NY, USA, 17–25. https://doi.org/10.1145/2659766.2659773

[43] Yang Zhang, Chouchang (Jack) Yang, Scott E. Hudson, Chris Harrison, and Alanson Sample. 2018. Wall++: Room-Scale Interactive and Context-Aware Sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 273, 15 pages. https://doi.org/10.1145/3173574.3173847