AirTouch: Synchronizing in-air hand gesture and on-body tactile feedback to augment mobile gesture interaction

Seungyon Claire Lee
HP Labs
Palo Alto CA US
claire.sylee@hp.com

Bohao Li and Thad Starner
GVU Center, Georgia Institute of Technology
Atlanta GA US
{bhli1989, thad}@gatech.edu

Abstract

We present the design and evaluation of AirTouch, a wristwatch interface that enables mobile gesture interaction through tactile feedback during limited visual attention conditions. Unlike its predecessor, the Gesture Watch, AirTouch is supported by a push-to-gesture mechanism (PTG) where the user performs a gesture and then confirms it afterward with a trigger gesture. The Gesture Watch, in contrast, requires the user to hold a trigger gesture while performing an interaction, and its PTG method does not allow the user to preview nor reverse the action. The effect of the new PTG mechanism and tactile feedback are evaluated through two experiments. The first experiment compares AirTouch's PTG mechanism to that of the Gesture Watch both with and without visual access to the watch. The second experiment examines mobile gesture interaction with the new PTG mechanism in four conditions (with and without tactile feedback and with and without visual restriction). We found that the new PTG mechanic enabled more accurate and faster interaction in the fully visible condition. Additionally, tactile feedback in the limited visual access condition successfully compensated for the lack of visual feedback, enabling similar performance times and perceived difficulties as in the fully visible condition without tactile feedback.

1 Introduction and motivation

Gesture-based interaction is gaining attention in the consumer electronics market (e.g., Nintendo Wii and Microsoft Kinect) as a viable mode of interaction to control devices in a more natural and intuitive way. Mobile gesture interaction is a logical next step for future investigation. In a mobile interaction, users are often on-the-go interacting with their mobile device as they navigate the physical environment. When mobile, users need to split their visual attention between their device and the environment to ensure accurate hand-eye coordination while avoiding obstacles in their path. However, interacting with a mobile device while in motion raises concerns for safety [16, 19, 22].

When developing our first wristwatch user interface (UI) for mobile gesture interaction, the Gesture Watch [9], we observed similar problems. The Gesture Watch captured in-air hand gesture through wrist-mounted IR sensors and sent the gesture patterns to a recognizer. With the Gesture Watch, users could control electronic devices (e.g., MP3 player) while on-the-go.

AirTouch pairs each proximity sensor with a vibration motor pressed against the user's wrist. When the proximity sensor detects a hand above it, the corresponding motor buzzes. We designed a new push-to-gesture mechanism (PTG) for AirTouch which takes advantage of this tactile feedback. The mechanism follows two design principles of *direct manipulation: representation of the object of interest* (in this case, the tactile representation of hand movement in relation to the device) and *reversible operations* [21]. With AirTouch, a gestural command is implicitly canceled to reverse the action when the user does not commit the command with a trigger gesture. Unlike the PTG mechanism of the Gesture Watch, the new PTG of AirTouch provides an eyes-free representation of what the sensors are perceiving and allows the user to cancel the interaction implicitly in case the gesture was made in error or the system's perception of the gesture seems likely to be wrong (as judged by the user). We hypothesize that tactile feedback and the new PTG of AirTouch will assist eyes-free mobile gesture interaction when a user's vision is limited.

In this paper we present the results of two experiments that investigate mobile gesture input and vibrotactile feedback in conditions of limited visibility. The rest of the paper reports related work in mobile gesture interaction and tactile perception space, the introduction of the AirTouch system and pilot test for PTG design, and two experiments. Our first experiment evaluates the appropriateness of the new PTG technique while the second experiment evaluates the effect of tactile feedback on participants' use of AirTouch.

2 Related work

As a design principle, direct manipulation has shaped graphical user interfaces (GUIs) and even physical UIs [8, 21]. For example, a text label or tooltip box that changes color or appears upon a mouse-over event in a GUI visually represents the *object of interest* and enables *reversible actions* as suggested by direct manipulation. By using capacitive sensing integrated with physical keys, Rekimoto and colleagues created keyboards which would display what action a given button would perform when a user touched the button but had not yet pressed it [15]. In this manner, users could interact with the physical buttons in smaller steps and retreat from an action before committing it.

Touch and sensor-based interactions in mobile and wearable computing often raise new concerns on motor performance. Proprioception [20] (perception of motion and position of body parts in accordance with joint and muscle control) is essential in motor performance [17] and highly affected by force feedback [18] and vision [14]. Even expert typists suffer performance difficulties when using touch-based soft keyboards due to the lack of force feedback [4]. This difficulty can be reduced by providing tactile or auditory feedback [12]. To reduce visual distraction in mobile interactions, researchers have explored the benefit of using haptics. Users can perceive tactile patterns on the wrist while visually distracted [11], receive tactile directional cues on the torso while driving [6], and navigate the environment helped by navigational cues on the waist [23].

Motor distraction in mobile computing (e.g., walking) can also limit human performance [2, 3]. To compensate for limited dexterity (and improve social appropriateness) while mobile, Whack Gestures [7] utilized gross gesture for inexact and inattentive interaction rather than fine gesture that requires precise hand-eye coordination. Ashbrook and colleagues tested the importance of device placement in mobile computing. They observed significantly faster interaction time on wrist-worn mobile devices than devices stored in pockets [1], suggesting that wrist-worn devices may require less motor distraction than devices placed elsewhere.

3 Apparatus and configuration

As with the Gesture Watch, AirTouch performs gesture recognition using the Gesture and Activity Recognition toolkit (GART) [13], which utilizes hidden Markov models (HMMs). GART links the patterns of hand gesture to corresponding commands that can be sent to electronic devices.

3.1 Wristwatch interface and GART

The Air Touch consists of two parts, the sensors in a wristwatch UI and a tactile display which are fastened by

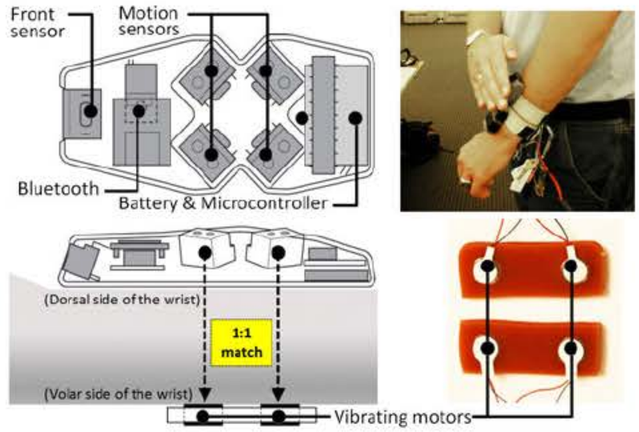

Figure 1. AirTouch wristwatch interface

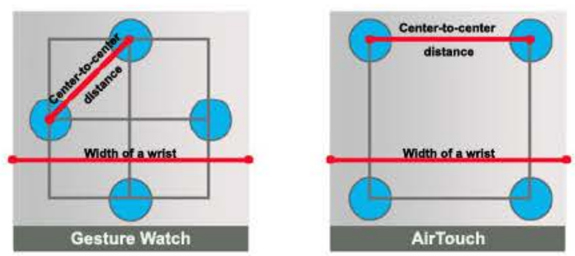

Figure 2. Sensor layout comparison

an elastic strap and worn on the dorsal and volar sides of the wrist, respectively (Figure 1). Since we observed user difficulty in localizing vibrators on the dorsal side of the wrist in our previous studies [5], we located the tactile display on the volar side, where high perception accuracy has been shown previously [10, 11]. The size of the wristwatch UI is 109mm x 20.5mm x 49.5mm (L x H x W).

We use four SHARP GP2Y0D340K IR proximity sensors to capture in-air hand gestures between 10 - 60 cm above the wrist. Four vibrating motors (Precision Microdrives #301-101, 200Hz, d = 10 mm, h = 3.4 mm) are arranged in a square with 30 mm center-to-center distances. A rubber housing ensures constant center-to-center distance between the motors. Unlike the cardinal layout of the Gesture Watch, AirTouch sensors are arranged in an orthogonal layout (Figure 2) to assist easier perception of tactile feedback. The orthogonal motor layout that is synchronous with sensor layout enables longer center-to-center distances between motors compared to the cardinal layout. We believe that the longer center-to-center distances and the simpler grid of AirTouch enable easier perception.

The microcontroller synchronizes sensors and motors by turning on and off vibration motors based on the sensors' input. Users can mentally synchronize the in-air hand gesture with on-body tactile feedback. Power is supplied by a 3.7

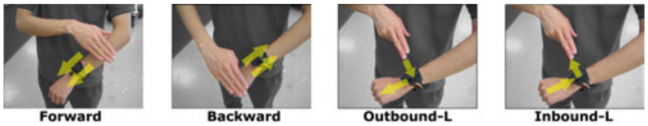

Figure 3. Gestures tested in the user study.

Figure 4. GART GUI: training (left), recognition (right)

V Lithium-ion battery. A power regulator is used to guarantee a stable 3.3 V power supply. The front sensor is placed at the side of the watch facing toward the user's hand. To avoid false triggering caused by the hand, the front sensor is tilted 20 degrees upward.

A remote computer receives sensor data from AirTouch through Bluetooth and processes the gestures. The GART GUI is implemented in Java, and it assists visual training of new gestures and recognizes trained gestures during experiments (Figure 4).

3.2 Push-to-gesture mechanism (PTG)

Once captured, data from the sensors are passed to an ATmega 168 microcontroller. The microcontroller stores sensor data, turns on and off motors, and waits for user confirmation rather than immediately sending the gesture to GART. This storage function of the microcontroller is added to AirTouch to support our new PTG. The new PTG in AirTouch has a time-out period for user confirmation in the interaction. Within the time-out period, users can make a decision to confirm or abort the gesture (Figure 5). If the tactile sensation of the hand gesture does not match user's intention ('3-1.Receive tactile feedback' in Figure 5), the user can cancel the incorrect gesture by not triggering the confirmation sensor within the time-out period ('4. Confirm or abort' in Figure 5). The microcontroller will send data to GART only when the user confirms the gesture by quickly tilting up and down his wrist within the time-out period. Stored sensor data is discarded after the time-out. Following the principles of *direct manipulation* [21], the on-off status of the motor indicates the state change of the *object of interest* (sensors) and the time-out period that waits for user confirmation enables *reversible user operations*.

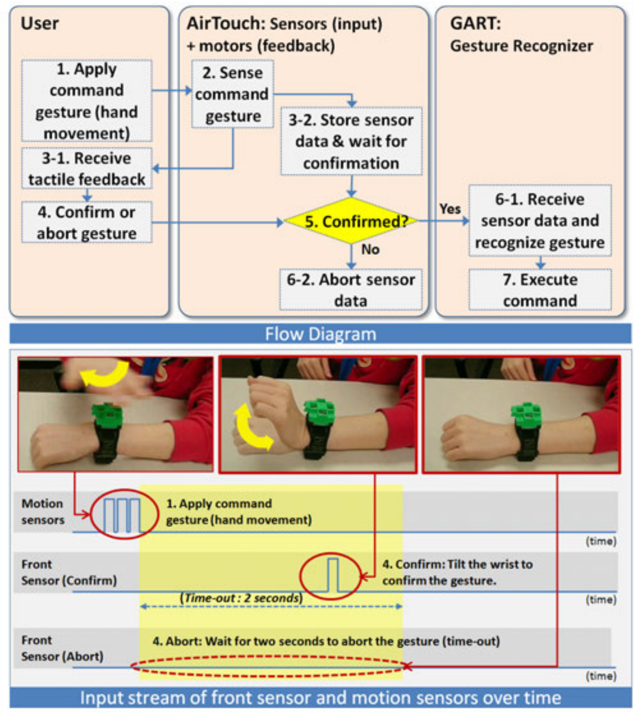

Figure 5. AirTouch interactions.

Unlike the PTG of the Gesture Watch which required a pair of wrist tilting gestures for segmenting the gesture to be recognized (Figure 6), the time-out period of AirTouch enables automatic segmentation of the gesture by taking advantage of the 'idle period.' With this PTG, the semi-automatic gesture segmentation in AirTouch is simpler than the 'do-while' type interaction of the Gesture Watch (i.e., hold segmentation gesture while applying command gesture).

To find the appropriate length of the time-out period, we performed a pilot test with seven participants. During the pilot test, participants were asked to apply four gestures (Figure 3) six times (4 gestures x 6 times = 24 trials) in random order while wearing the AirTouch system on their non-dominant wrist. Participants listened to voice commands from a computer and applied the gesture with the new PTG (Figure 5). We measured the time lapse between the last data input from the motion sensor and the front sensor activation. The result calculated from 168 data points showed that the maximum time between the hand gesture and confirmation was 1.3 seconds. Thus, we decided to set the time-out period for two seconds in our formal experiment.

4 Study design overview

To investigate the possible benefits of tactile feedback with respect to visual attention in mobile gesture interac-

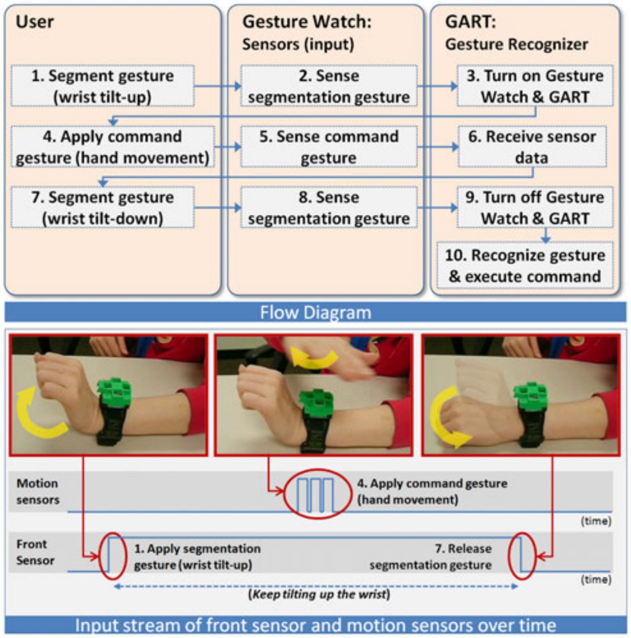

Figure 6. Gesture Watch interactions.

Table 1. Conditions for experiments 1 and 2

	Experiment 1 (old PTG vs. new PTG)		Experiment 2 (effect of tactile feedback)	
	old PTG	new PTG	no feedback	has feedback
full vision	Old-full	New-full	nT-full	T-full
limited vision	Old-limited	New-limited	nT-limited	T-limited

tions, we conducted two experiments involving limited and full visual access to the interface. Experiments addressed the following research questions: how are the two PTG mechanisms different and what is the impact of the new PTG on user performance compared to the previous PTG? (experiment 1) and does tactile feedback during gestural interaction compensate for limited visual access while the user walks (experiment 2)? The study design for each experiment is composed of 2x2 conditions (Table 1). Tasks in all conditions are performed while the participants walk in a designated path (Figure 7-right). The walking path is set in a quiet lab setting. A pair of orange flags visually mark each 'gate' in the walking path to guide the user. All participants were recruited from the Georgia Institute of Technology.

5 Experiment 1

Fourteen participants volunteered for the experiment (mean age = 22.36, eleven male). The mean width and circumference of their wrists were 54.53 mm and 162.93 mm, respectively. All participants were right-handed except one. 50% of the participants wore a wristwatch daily. During the experiment, we measured accuracy and amount of time required to gesture (4 gestures x 6 times x 2 conditions x 2 groups x 14 participants = 1344 trials). Participants completed a NASA Task Load Index (TLX) for each condition and a survey on their impressions of the interfaces.

5.1 Task and procedure

The experiment was conducted with a mixed between-within subject method in a balanced order. We selected this method because interactions with different PTG mechanisms is likely to cause confusion when used consecutively by one person. Seven participants (five male) controlled the watch with the old PTG method (Figure 6), whereas the other seven participants (six male) used the new PTG mechanism (Figure 5). No tactile feedback was provided in this experiment. Participants in each group had two conditions, full or limited vision. To simulate limited visual access to the watch, participants wore half-blocked goggles. The goggles limited the wearer's sight below the eye level when interacting with the wristwatch interface. However, the goggles allowed full vision for walking.

The experimental procedure was composed of four sessions: training, walking practice, main test, and survey session. During the training session, participants wore the watch UI on the non-dominant wrist, stood in front of the computer, and attempted four gestures three times (4 gestures x 3 times = 12 trials) guided by the researcher. The training GUI (Figure 4-left) and a synthetic voice on the computer presented the name and direction of the gesture. Once the participants heard the voice command for the gesture, they applied the gesture to the watch UI. This procedure is identical to the full vision condition in the main session except for the absence of the visual guide (gesture direction and name) and additional mobility condition (standing instead of walking). After finishing the training session, participants practiced walking twice along the designated path. The researcher led the participant for the first trial and followed the participant for the second trial (Figure 7-right).

In the main test session, participants wore headphones and carried a laptop in a backpack (Figure 7-left). On the laptop computer, the GART test GUI (Figure 4-right) was played to provide tasks for the user. This GUI was also remotely shared to another computer in the lab so that the researchers could monitor the live status of the performance. The voice command from the headphones prompted the study by saying 'Welcome to the user study. Please start walking.' Within 5 seconds, the first task was given by a voice command. The name of the gesture was played twice to ensure clear delivrey of the voice command (e.g., "Forward gesture. Forward."). Then the participant applied a hand gesture to the wristwatch UI. For the first group that used the old PTG mechanism, no feedback for confirmation or failure of the gesture was provided as the old system was

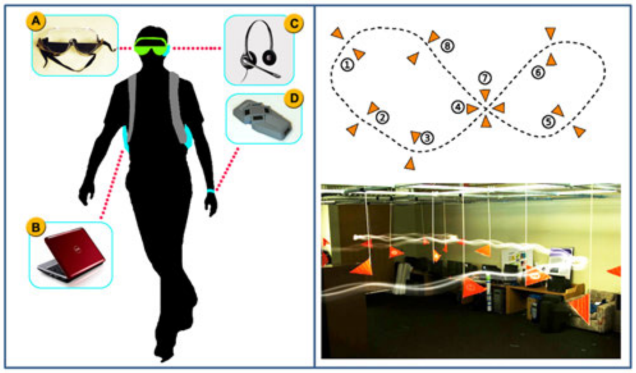

Figure 7. Test setting (left, A: Half-blocked goggle, B: Laptop computer, C: Headphone, D: AirTouch), walking track (right).

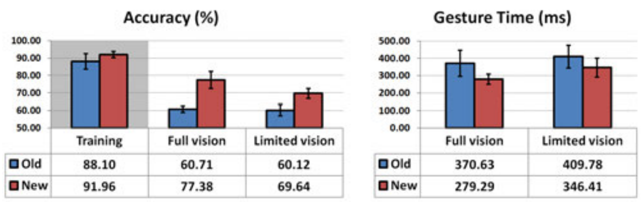

Figure 8. Experiment1: accuracy and time.

not designed to support it. On the other hand, for the second group that used the new PTG method, voice feedback was provided to indicate the user's decision (e.g., "Confirmed" or "Aborted, please try again"). The result of the gesture performance (correct or incorrect) was not provided for either groups. After a random interval that ranged between 10 and 20 seconds, another voice command was played to guide the next trial.

While participants walked and performed the task, researchers sat on the desk next to the walking path. One researcher controlled the GART GUI, and another researcher observed the participants to take notes or help them upon request. A subjective rating survey and the NASA-TLX was administrated after the main test session.

5.2 Result

The mean accuracy of the training session for the new and old PTG methods was 91.96% and 88.10%, respectively. In the main test, we measured performance accuracy (Figure 8-left) and gesture time (Figure 8-right), which is the time difference between the first and last incoming sensor data captured. The mean accuracy of the new PTG system was higher than the old system with statistical significance in the full vision condition (paired t-test with Bonferroni correction, $p < .05$), but not in the limited vision condition. The effect of visual restriction (full vs. limited vision) was not statistically significant in either PTG types (old and new).

Gesturing using the new PTG method was faster than the old PTG method with statistical significance in the full vision condition (paired t-test, Bonferroni correction, $p < .05$), but not in the limited vision condition. The effect of the visual condition (full vs. limited vision) was statistically significant only in the new PTG method (paired t-test, Bonferroni correction, $p < .05$).

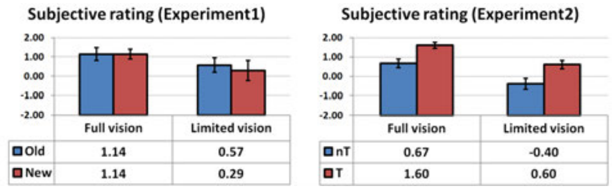

Figure 9. Subjective rating (-2.0 = very difficult, -1.0 = difficult, 0.0 = neutral, 1.0 = easy, 2.0 = very easy.

5.3 Subjective rating and workload

The difficulty (Figure 9-left) of all conditions was perceived as easy to neutral. The subjective rating of each condition indicated that the performance with full vision was perceived as easy (≈ 1.0) both with the old and new PTG methds. The limited vision condition was rated slightly lower as neutral to easy (0.0 - 1.0).

Results from the participants' self reports that were collected with the NASA-TLX assessment indicated that the design of the gestures was simple and easy, but using the system required a bit of familiarization time. Although the gesture (command, segmentation, and confirmation gesture) was awkward at first for some participants, soon they felt that the gestures became natural. Additionally, some participants reported needing extra effort while applying the L-shaped gesture. We will discuss the learnability of command gestures and PTG interactions later.

Ten out of fourteen participants mentioned that they could not coordinate hand and sensor correctly in the limited vision condition (which indicates the visual attention needed during the interaction). This difficulty was consistently observed both with the old and the new PTG mechanisms. Increased confidence was frequently mentioned in the full vision conditions. Some old PTG participants reported that holding the tilted non-dominant wrist for segmentation was obtrusive because they put extra effort in to avoid hitting the wrist during a dominant hand gesture.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase®
Smarter legal research.