Duet: Exploring Joint Interactions on a Smart Phone and a Smart Watch

Xiang 'Anthony' Chen[12], Tovi Grossman[1], Daniel Wigdor[3], George Fitzmaurice[1]

[1]User Interface Group
Autodesk Research
{firstname.lastname}@autodesk.com

[2]HCI Institute
Carnegie Mellon University
xiangchen@acm.org

[3]Department of Computer Science
University of Toronto
dwigdor@dgp.toronto.edu

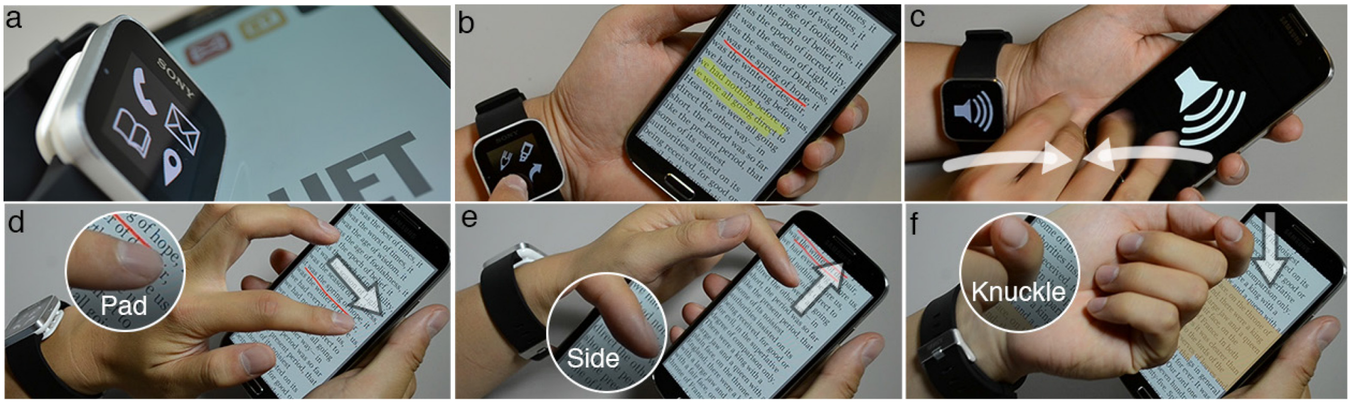

Figure 1. A duet of interaction between a handheld and a wrist worn device (a): the watch is used as a tool palette when annotating text on the phone (b); a simultaneous pinch-to-close swipe gesture on both devices mute their notifications (c); the watch's orientation indicates which hand part causes a touch, thus enabling a seamless transition between modes: for example, writing with the pad of the finger (d), scrolling with side of the finger (e), and text selection with the knuckle (f).

ABSTRACT
The emergence of smart devices (e.g., smart watches and smart eyewear) is redefining mobile interaction from the solo performance of a smart phone, to a symphony of multiple devices. In this paper, we present Duet – an interactive system that explores a design space of interactions between a smart phone and a smart watch. Based on the devices' spatial configurations, Duet coordinates their motion and touch input, and extends their visual and tactile output to one another. This transforms the watch into an active element that enhances a wide range of phone-based interactive tasks, and enables a new class of multi-device gestures and sensing techniques. A technical evaluation shows the accuracy of these gestures and sensing techniques, and a subjective study on Duet provides insights, observations, and guidance for future work.

Author Keywords
Duet, joint interaction, smart phone, smart watch.

ACM Classification Keywords
H.5.2 [User Interfaces]: Input devices and strategies, Interaction styles.

CHI 2014, April 26 - May 01 2014, Toronto, ON, Canada
Copyright 2014 ACM 978-1-4503-2473-1/14/04…$15.00.
http://dx.doi.org/10.1145/2556288.2556955

INTRODUCTION
Interactive computing technology is becoming increasingly ubiquitous. Advances in processing, sensing, and displays have enabled devices that fit into our palms and pockets (e.g., [2, 15]), that are wrist-worn [27, 40] or head-mounted [20, 29], or that are embedded as smart clothing [28, 37]. Commercialization is rapidly catching up with the research community's vision of mobile and ubiquitous form factors: smart phones, smart watches, and smart eyewear are all available for purchase. Soon, many of us may carry not one smart device, but two, three, or even more on a daily basis.

For interaction designers, this introduces a new opportunity to leverage the availability of these devices to create new interactions beyond the usage of a single device alone. At present, the space of interaction techniques making use of this opportunity is underexplored, primarily focusing on using a secondary mobile device such as a smart watch as a viewport and remote control of the smart phone [37]. To the best of our knowledge, we are unaware of any existing work that takes a different approach in designing a class of *joint interactions* on two smart mobile devices.

To address this limit, our research envisions a symphony of interaction between multiple smart mobile devices. To approximate this vision, we start by considering a scenario of two smart mobile devices as a joint interactive platform. Specifically, our goal is to explore various ways wherein these two devices can perform their individual input and output techniques to create new interaction possibilities for the users. To begin to realize this vision, we built *Duet* – an interactive system that enables joint interactions between a smart phone and a smart watch (Figure 1a). Inspired by research on conversational linguistics [9], and prior HCI work on 'foreground-background' interactions [5, 14], we explore a design space of interaction between the phone and the watch. Based on their spatial configurations, Duet coordinates the two devices' motion and touch input, and extends their visual and tactile output to one another, thus enabling the watch to enhance a wide range of phone-based interactive tasks. For example, we can divide an interface between the phone and the watch, such as reserving the phone for a canvas while hosting a tool palette on the watch (Figure 1b). We can create novel gestures, such as a cross-device pinch-to-close gesture to simultaneously mute both devices (Figure 1c). We can also use a watch's sensors to augment touch, such as using its orientation to infer which finger part (pad, side or knuckle) is touching the phone's screen (Figure 1d-f).

In the following sections, we first review techniques of handheld, wrist-worn and device-to-device interaction. Next we present our design space that encompasses the interaction between the phone and the watch based on their 'foreground-background' interactional relationships. We then introduce a suite of gestures and sensing techniques enabled by this joint interactive platform – and a technical evaluation of their recognition accuracy. We then demonstrate Duet's various application scenarios enabled by these gestures and techniques, and report users' reactions and feedback to inform future work.

RELATED WORK

We first review prior work that explores individual interaction techniques developed for *handheld* and *wrist-worn* devices. We further summarize various *device-to-device* interactions that demonstrate examples of using multiple devices to create new interaction possibilities.

Interaction Techniques for Handheld Devices

Touch is perhaps the most common input method for modern handheld devices. *Motion and spatial awareness*, enabled by a device's on-board sensors, can also be leveraged to enhance touch-based interaction [15]. Past research has demonstrated interaction by orienting [10], positioning [51], tilting [40], or whacking [19] a device. To go beyond the device's physical boundaries, others also explored interacting with the device using *freehand gesture* [4,21]. All these techniques, under Buxton's framework, fall into the 'foreground interaction' category [5]. Meanwhile, for 'background interaction', *context-awareness*, such as location, has long been proven useful in various mobile interaction scenarios [44]. Altogether, this work collects a toolbox of possible techniques for handheld devices.

Interaction Techniques for Wrist-worn Devices

The above interaction techniques for handhelds can also be found on wrist-worn devices. However, wrist worn devices have an even smaller form factor and are worn on our bodies, which demands a reconsideration of the techniques as well as explorations into new interaction possibilities.

Touch is more difficult on wrist-worn devices, which typically have small screens, exacerbating the *fat finger problem* [46]. Zoomboard used iterative zooming to ease target acquisition [36]. Facet utilized a multi-segment wrist-worn device, which allows touch to span multiple connected screens, yielding a richer input vocabulary [28].

Given the limitation of direct-touch on wrist-worn devices, the exploration of alternate techniques becomes more important. *Motion and spatial awareness* creates a variety of wrist-based interaction. GestureWrist and GesturePad use wrist-mounted accelerometer and capacitive sensors to recognize hand grips and pointing directions [42]. Past research also focuses on wrist rotation and tilting [8, 38, 46], most of which was implemented using a mobile phone held in the hand. In contrast to handhelds, a wrist-worn device is a potentially better solution: its wearability untethers the users' hands from any sensing devices, allows the sensing to be always available with one's day-to-day activities, and provides high-fidelity data by closely coupling the device to one' hand, wrist and arm movement.

A watch can also enable *freehand gestures* beyond its surface. A disappearing mobile device [35] can be mounted on the wrist and interacted with by 'scanning' fingers on top of it. Abracadabra enables spatial input with a small wrist-worn device by using the magnetometer to sense finger-mounted magnets [12]. Gesture Watch [23] and AirTouch [26] use multiple infrared sensors mounted on the back of the wrist to detect freehand gestures, such as hand swiping along the forearm. Instead of using the other hand to perform gestural input, earlier work also explores using wrist-mounted contact microphones to detect fingertip gestures [1]. Similarly, Digits reconstruct real-time 3D hand models by instrumenting sensors on the inner side of the wrist and facing the camera towards the palm [22].

Our review shows a plethora of interaction techniques developed for both handheld and wrist-worn devices individually. Yet few have considered marrying their techniques to design for scenarios where a user is carrying and using both devices. To better understand this issue, we review prior work on device-to-device interaction.

Device-to-Device Interaction

Device-to-device interaction associates multiple individual devices to create new interaction possibilities. We summarize three association principles from the literature: *synchrony, proxemic interactions,* and *distributed gestures.*

Synchrony associates devices by the synchronization of their inputs. Pick-and-drop synchronizes pen input across multiple computers to enable direct content manipulation between them [41]. Smart-Its Friends senses a handshake as a natural way to establish connections between smart artifacts [18]. Synchronous gestures detect the bumping between two tablets, thus allowing interactions such as spanning and sharing a photo across two screens [17]. Stitching applies a similar idea where a pen stroke across two tablets can be used to, for instance, transfer files [16]. Similar techniques were also used in Lucero et al's work where a pinching gesture between mobile devices spans a shared canvas across them [27]. Siftables proposes synchronized interactions with multiple networked tangible interfaces, such as bumping all devices at once to swap in a new set of data associations [34].

Proxemic Interaction associates devices by their spatial relationship (e.g., proximity and orientation) between one another. The Relate system built customized sensors into USB dongles, thus allowing peer-to-peer computation of devices' spatial relationship, and a set of spatial widgets to incorporate such relationship into the user interface [24]. A spatial proximity region around mobile devices can be used to mediate content access and sharing among a group of users [25]. Gradual engagement applies a similar idea to facilitate different levels of information exchange as a function of device-to-device proximity [31].

Distributed Interactions divides the tasks, features, or functions of an interface between multiple devices. Roomware envisions a room of inter-connected smart artifacts that augment people's individual or collaborative tasks [47]. ARC-Pad divides cursor positioning task into absolute pointing on a mobile device and relative adjustment on a large display [33]. A cross-device interaction style [45] designs interaction between a mobile device and a large interactive surface, such as selecting from a list of tools on the mobile and applying that tool in an application on the surface.

While this work shows the potential of certain device-to-device interactions, we are unaware of any existing research that has explored the opportunities of using the phone and the watch together. We see an immense potential to explore this opportunity of combining a smart phone and a smart watch to enhance our everyday mobile interactions.

DESIGN SPACE

The fundamental idea of our design space is to allow the phone and the watch, in various ways, to perform their individual input and output techniques, and together to create new interaction possibilities for the users. We construct a design space (Table 1) based on Falk's research on conversational linguistics [9], and Buxton's [5] and Hinckley et al.'s [14] 'foreground-background' frameworks.

In constructing this design space, our goal is for the two devices to carry out interactive tasks for the user as a single unified platform. This is similar to what Falk observed and described in her paper 'The conversational duet': "*In conversations between three or more persons, two of them may undertake jointly to carry out the communicative task to a third in such a way that a written version of their resultant in-sequence text would be indistinguishable from that of a single speaker.*" [9]

To attain this goal, the two devices should both perform their individual foreground or background interactions. Buxton defines foreground interaction as "*activities which are in the fore of human consciousness – intentional activities*" [5]. Hinckley et al. develops the definition of background interaction as "*sensing an action that the user would have had to perform anyway to accomplish their task*" [14]. In the past, these frameworks have been focusing on the context of a single device. Our design space extends this framework to a scenario when two mobile devices are present, guiding the design of interactions between them.

As shown in Table 1, the combination of foreground and background interactions, when two devices are present, creates a 2x2 design space encompassing a variety of interactions that leverage the availability of both devices. Current commercial designs have been focusing on the lower-left quadrant, where the watch is used as a temporal replacement for the phone, such as using the watch to check new emails or read text messages when the phone is not ready to hand [37]. The lower-right quadrant characterizes work that uses both devices for context and activity sensing [7, 31]. Less work has been done in the two upper quadrants where the phone remains in the foreground as an active input and output platform, and the watch transitions between foreground interaction (as an input device and extended display) and background sensing. Duet is a new system that focuses on and explores these two areas of the design space.

	Watch Foreground	Watch Background
Phone Foreground	Duet: • Phone as a primary input and output platform; • Watch as an input device or extended display.	Duet: • Phone as a primary input and output platform; • Watch as a sensor.
Phone Background	Current commercial designs: • Phone as an inactivated information portal [37] • Watch as a viewport or remote control [37]	Prior research: • Both phone and watch used for context and activity sensing [7, 31].

Table 1. A design space of interaction on a smart phone and a smart watch based on Buxton's framework [5].

IMPLEMENTATION DETAILS

To explore these areas of the design space, we built an interactive platform on a smart phone and a smart watch.

Hardware and System Setup

We used a Samsung Galaxy S4 smart phone and a Sony SmartWatch. The phone has a 1080x1920 capacitive multi-touch screen, quad-core 1.6 GHz processor, and a 3-axis accelerometer. The watch has a 128x128 pixels capacitive-touch color display, and is connected to and run by the phone via Bluetooth. The API of the watch provides limited touch input of seven pre-defined gestures: press, long press, release, and four swiping directions. Its accelerometer has a maximum rate of 10Hz.

Spatial Configuration

To fully explore the design space, we consider two possible ways in which the devices can be worn or carried in relation to one another. Figure 2 shows these two spatial configurations: with the face of the watch worn on the dorsal side (a), or on the ventral side of the wrist (Figure 2b). These two spatial configurations afford different ways in which the watch can augment phone-based interactions. In particular, wearing the watch on the ventral side provides additional visibility and quicker access to the watch while holding and using the phone. The current spatial configuration can be detected by constantly monitoring the two devices' relative orientation, or by using a motion-based gesture to explicitly register that information (we use the latter approach). While not necessary, we found the use of an expansion watchband can enable easy switching between these two configurations (Figure 2c).

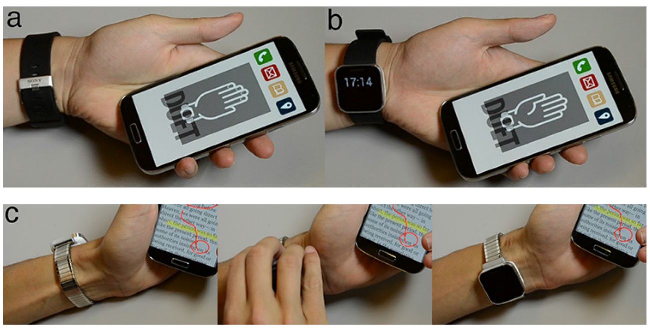

Figure 2. Devices' two spatial configurations: watch worn on the a) dorsal side or (b) ventral side of the wrist. c) An expansion watchband allows easy switching between the two.

GESTURES AND SENSING TECHNIQUES

The availability of a smart phone and a smart watch gives rise to a suite of new gestures and sensing techniques (Figure 3). Below, we describe the gestures and sensing techniques we explored, in the context of our design space (Table 1). Specifically, we developed two categories of techniques, where the watch is either in the foreground and background of interaction. This section provides a brief overview of these gestures and sensing techniques and a description of our gesture recognition methods. Later we describe the Duet system to demonstrate interactive scenarios and applications that utilize such techniques.

Watch in the Foreground, Phone in the Foreground

In the foreground of interaction, the watch can be used as an additional input device to complement the phone's motion and touch input. We implemented two gestures that utilize this additional input channel:

Double bump: Bumping the phone on the watch creates a synchronous gesture [17] that provides distinct input properties compared to bumping on other surfaces. To reduce the chance of false positives, we implemented a double bump gesture, where the phone bumps against the watch twice in succession (Figure 3a).

Multi-device gestures: We adapted the stitching technique [16] and the pinching gestures for interaction between the phone and the watch, to support four novel *multi-device gestures* (Figure 3b). The first two involve the finger swiping from the phone to the watch (*phone-to-watch swipe*) or from the watch to the phone (*watch-to-phone swipe*). The second two gestures are performed by two fingers swiping simultaneously, where the fingers move towards each other (*pinch-close*) or away from each other (*pinch-open*) (Figure 3b).

Watch in the Background, Phone in the Foreground

In the background of interaction, the watch can be used as an auxiliary sensor for the phone.

Flip and tap: To perform this gesture, a user flips her hand (that wears the watch) immediately before tapping on the phone's screen (Figure 3c).

Hold and flip: Inspired by DoubleFlip [43], this gesture consists of flipping the phone while holding the finger down on the screen. By detecting the synchronized motion of both devices, we also distinguish if the user is flipping the phone with the hand wearing the watch (Figure 3d).

Finger posture recognition: By sensing the watch's orientation when touching the phone, we can tell which finger part (pad, side, or knuckle) causes that touch (Figure 3e).

Handedness recognition: By correlating the watch's motion with the phone's motion at the onset of a touch event, we infer whether a touch is caused by a bare hand, or a hand that wears a watch (Figure 3f).

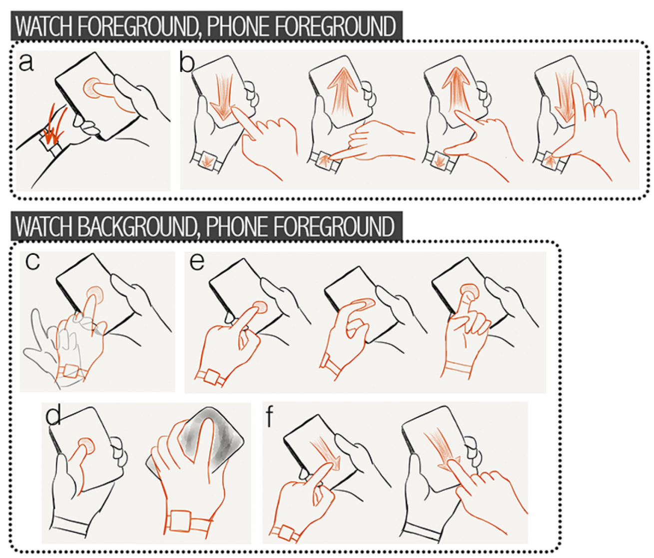

Figure 3. The core gesture and sensing techniques on a joint interactive platform of a phone and a watch: a) *Double bump*, b) *Multi-device gestures* (four different swipes: pinch-open, pinch-close, watch-to-phone swipe, and phone-to-watch swipe, c) *Flip and tap*, d) *Hold and flip*, e) *Finger posture recognition* (the pad, side or knuckle of the finger), and f) *Handedness recognition* (left- vs. right- hand touch).

Gesture Recognition

We used machine learning techniques for implementing our recognition system. For motion-related input, our general approach is to segment a chunk of accelerometer data (an array of X/Y/Z) pertinent to a particular gesture. We then flatten the data into a table of features – each axis value at a given time point is considered a feature. Using these features we can train a machine learning model (we use a Decision Tree – one of the most widely applied machine learning techniques) to recognize the gesture. These recognizers are used only if the watch-wearing hand is detected during the onset of a touch (using the aforementioned handedness recognition); otherwise a default interaction is applied.

TECHNICAL EVALUATION

To understand the feasibility of our designs, we tested the recognition accuracy of the six described techniques (Figure 3). In the evaluation, participants wore the watch on the dorsal side of their left wrist (Figure 2a) except for the *Multi-device gestures,* which require the watch to be moved to the ventral side (Figure 2b).

The accuracy of each technique was tested independently. Each technique recognizes several conditions, corresponding to the number of classes in the machine learning model (e.g., f*inger posture* recognition involves three: pad, side and knuckle, Figure 3e). Some techniques (e.g., flip and tap) included a baseline condition (e.g., a standard tap without any hand flipping) to test false positives. The conditions for each technique were as follows, with the number of conditions in parentheses:

Double bump (2): Users either performed a hold and double bump, or a hold without the double bump.
Multi-device gestures (4): Users performed the four multi-device gestures: pinch-open, pinch-close, phone-to-watch swipe, and watch-to-phone swipe.
Flip and tap (2): Users either performed a flip and tap, or performed a standard tap without first flipping the hand.
Hold and flip (2): Users either performed a hold and flip, or a hold without the flip.
Finger posture recognition (3): Users tapped the phone with either the pad of the finger, side of the finger, or knuckle.
***Handedness recognition (2)*:** Users tapped the phone with either the left (watch wearing) hand or the right (bare) hand.

Participants first learned to perform each technique condition by watching a demonstration by the experimenter. In the trials, participants were presented with visual cues instructing them to perform each condition.

Twelve participants (five male, seven female, ages 18 to 34, two left-handed) completed our study. Each participant performed five blocks of the six techniques, with the order of techniques counter-balanced using a Latin-square design. In each block, participants repeated 10 trials for each condition of a given technique. The first block for each technique was used for training. In total, the evaluation produced 12 participants × 15 conditions (across the 6 techniques) × 4 blocks × 10 trials per block = 7200 data points.

All techniques except for *Multi-device gestures* used machine learning based recognition. The results for *Multi-device gestures* will be discussed after our analysis on the first five techniques.

Ten-Fold Cross Validation

We conducted a conventional ten-fold cross validation using all the data from each technique. As shown in Table 2, all techniques achieved an accuracy of over 97% except for *Double bump* (93.87%). This result gives us a basic assessment where the interaction data from a group of users is known *a priori*, and a model can be trained and fine-tuned to a particular group of users. To challenge our techniques in more realistic scenarios, we conducted two further evaluations and analyses.

Per User Classifiers

It is important to know how the features perform at a per user level [13]. For each technique, we separated the data between the participants, and ran a ten-fold cross validation within the data of each participant. As shown in Table 2, the features are indicative for each technique for specific users (accuracy > 90% for all techniques). However, the results also show some users were inconsistent in performing the techniques, especially *Double bump* (SD 5.34%) and *Hold and flip* (SD 11.24%). These two techniques, by nature, are more complicated than the others, and demand clearer instructions and perhaps a larger set of training data.

	Double bump	Flip and tap	Hold and flip	Handedness recognition	Finger posture recognition
10-fold cross val.	93.87%	97.90%	97.56%	99.06%	99.34%
Per user classifiers	92.10% (5.34%)	95.92% (2.89%)	90.11% (11.24%)	97.33% (1.92%)	97.95% (0.80%)
General classifiers	88.33% (9.89%)	94.38% (9.91%)	85.29% (10.90%)	98.23% (2.64%)	93.33% (9.07%)

Table 2. Accuracy (SD in parentheses) of our gestures and sensing techniques: ten-fold cross validation, per user classifiers, and general classifiers.

Pinch to open	Pinch to close	Phone to watch	Watch to phone
97.69% (5.67%)	98.61% (2.32%)	95.83% (3.83%)	96.76% (3.25%)

Table 3. Accuracy (SD in parentheses) for *Multi-device gestures.*

General Classifiers

It is also important to know how the features can be generalized to new users whose data has not been used for training [13]. To simulate this scenario, we separated out one participant's data as a *test set* (new user), and the others' aggregated as a *training set* (existing users). For each technique, we repeated this process 12 times (i.e., all the combinations from the 12 users). We then calculated the average and the standard deviations of the accuracy. As shown in Table 2, the results indicate that for most techniques, there was some inconsistency between participants (SD between 9.00% and 11.00% except for *Handedness recognition*). As a result, their performance dropped compared to the previous two analyses. A solution to mitigate this problem is using some online learning mechanisms that dynamically incorporate a new user's data into the existing model.

Multi-Device Gestures
The multi-device gestures were recognized using hard coded heuristics, based on the gesture length, duration, and timing. The results of our evaluation (Table 3) show a fairly high accuracy of our implementation.

DUET: AN EXPLORATION OF JOINT INTERACTIONS
We now introduce the Duet system, which demonstrates how the novel gestures and sensing techniques we have described could be utilized to enhance a wide range of interactive tasks across various applications. Duet is an interactive system that explores the joint interactions between a smart phone and smart watch. The system can be thought of a smart phone shell that is enhanced by the watch. The shell consists of a home screen and four common mobile apps. The interactions we present are meant to explore the areas of interest within our design space (Table 1). In particular, we demonstrate how the watch can perform foreground interactions as an *input device* or *extended display*, or serve in the background as an *auxiliary sensor*. Meanwhile, the phone remains in the foreground, whose interaction is enhanced by these three different roles of the watch (as labeled in the headings of each interaction subsection below).

Home Screen
The Home Screen provides techniques for managing the device and its applications (apps).

Hold and Flip to Unlock AUXILIARY SENSOR
To unlock the device from an inactive mode, a user performs the *hold and flip* gesture (Figure 4). This gesture requires a synchronized motion of both the phone and the watch, thus reducing recognizer false positives. Optionally, one can use it as an additional security layer that requires the ownership of both devices in order to gain access.

Figure 4. *Hold and flip* unlocks the phone and registers the devices' spatial configuration (in this case, the watch is worn on the dorsal side of the wrist, also see Figure 2).

App Selection and Arrangement AUXILIARY SENSOR
Four app icons are displayed on the home screen. The user can touch an icon to open an app, or use a knuckle-touch to move the icons (Figure 5ab). Contrary to existing designs, this requires no extra steps to distinguish between opening and navigating the apps, and repositioning their icons.

App Selection Shortcut EXTENDED DISPLAY | INPUT DEVICE
A person can also use the watch to quickly switch between apps. Pressing and holding the watch brings up an app selection screen on the watch, which displays the app icons in a 2x2 grid (Figure 5c). Additional app icons would be organized on pages that a user would swipe between. Tapping on an app loads it on the phone, and pressing and holding on the app selection screen dismisses it.

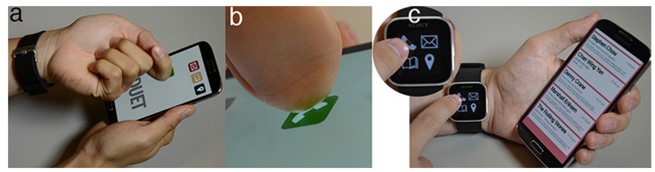

Figure 5. On the home screen, a, b) A knuckle-touch repositions app icons on the phone. c) The watch can be used to switch between apps on the phone.

Email
The Email app provides techniques to support reading and organizing a list of emails.

List Management AUXILIARY SENSOR
Tapping an email opens it; while a knuckle-touch can be used to select and apply actions to multiple emails, such as 'archive', 'mark as read' or 'delete'. This technique requires no extra widgets (e.g., checkboxes) for selection, thus saving more screen space for the other interactions.

Notification Management EXTENDED DISPLAY
In social occasions like meetings and movies, a person can use the *multi-device gestures* to manage which device(s) email notifications are received on. A pinch-to-close mutes both devices simultaneously (Figure 1c). A pinch-to-open resumes their notifications (Figure 6a). A stitching gesture from the phone to the watch directs all the notifications to the watch (Figure 6b). The opposite direction pushes all the notifications to be shown on the phone (Figure 6c). We also use tactile feedback to inform a gesture's direction, e.g., when swiping from the phone to the watch, a user can feel two vibrations – first on the phone, then on the watch, as if a single vibration was 'transferred' across the devices. This technique provides a way to customize notifications on multiple devices without resorting to extra physical buttons or UI elements.

Figure 6. In Email, Multi-device gestures are used to manage new email notifications on both devices.

Map

The Map app enhances a user's search and navigation task.

One-Handed Zoom AUXILIARY SENSOR

Existing map apps only partially support *one-handed use* – a common scenario that happens when, for instance, the user is holding a coffee. While zooming in can sometimes be accomplished with a double tap, it is difficult to zoom out with a single hand. With Duet, we use the *double bump* as a gestural shortcut for zooming out (Figure 7).

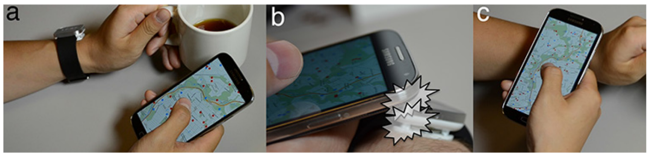

Figure 7. In a one-handed scenario, double bumping the phone on the watch creates a gestural shortcut to zoom out the map.

Multi-Device Target Selection EXTENDED DISPLAY | INPUT DEVICE

Another difficult task on a map app is selecting tiny and cluttered location markers (Figure 8a). Inspired by *Shift* [49], we design a mechanism for using the watch to facilitate small target acquisition on the phone. To start, the user thumbs down an area of interest (Figure 8b), which is then zoomed in on the watch (Figure 8bc). The user can select an enlarged target on the watch (Figure 8d), or swipe to pan and adjust the zoomed-in area (Figure 8e). Releasing the thumb brings the user back to the map navigation on the phone. The watch assists users in selecting *multiple* small targets without invoking widgets that take up screen space.

Figure 8. Using the watch to zoom in and select small map location markers on the phone.

Toggle View Modes INPUT DEVICE

The watch can also provide shortcuts for interactions on the phone. Swiping on the watch's screen is used as a shortcut to switch between normal and satellite views (Figure 9). In general, such simple interactions with the watch can replace tedious menu selections typically carried out on the phone.

Figure 9. Swiping on the watch's screen (b) switches map views – in this case, from normal (a) to satellite (c).

Reader

The Reader app allows users to read and annotate text.

Menu Access AUXILIARY SENSOR

A normal tap on the page brings up the menu with basic and frequently used options (Figure 10a). Alternatively, with the watch as a sensor, one can use the *flip and tap* gesture to display an advanced menu that contains additional commands (Figure 10bc).

Implicit Tool Selection AUXILIARY SENSOR

We use the finger postures recognition to implicitly select tools in the reader app. For example, after selecting a pen tool from the menu, the finger pad is used to annotate the text (Figure 1d), the side of the finger to scroll the page (Figure 1e), and the knuckle to start text selection (Figure 1f). This allows for a seamless transition between three frequent operations without having to explicitly specify any modes.

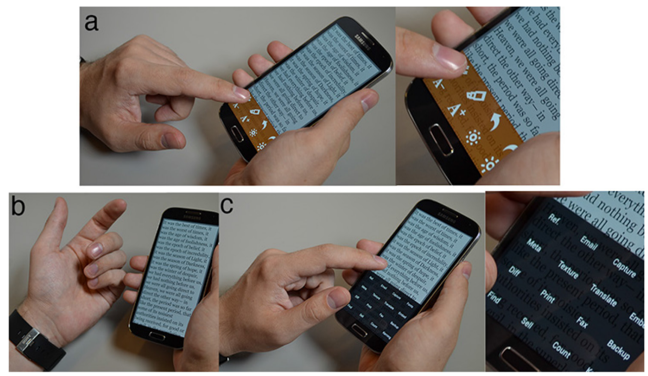

Figure 10. A tap accesses a basic menu (a), while a flip-and-tap accesses an advanced menu in the Reader app (bc).

Multi-Device Clipboard EXTENDED DISPLAY

In addition to the phone's default copy and paste functions, the watch can also be used as a clipboard that holds multiple pieces of text. Upon selecting the text (Figure 11a), the text will be displayed on the watch. Users can then add it to the clipboard by swiping right on the watch (Figure 11b). One can also retrieve an earlier selection by swiping down the clipboard (Figure 11cd).

Figure 11. Using the watch as an additional way to copy and paste from a clipboard of text. Selected text (a) is added to the clipboard with a swipe to the right (b), which then displays all the selected text (c). Swiping down goes through the text (c).

Multi-Device Tool Palette EXTENDED DISPLAY

By positioning the watch to the ventral side of the wrist, the watch can be used as a tool palette (Figure 1b). Swiping on the watch shows more menu options. Hosting the tool palette on the watch saves screen space on the phone for other interactions. It also resembles how a painter holds and uses a color palette while drawing on a canvas.

Call

The call app shows an exemplar interaction using the watch to retrieve information while holding the phone for a call.

Information Retrieval EXTENDED DISPLAY

In this situation, back-of-device touch [3,50] might be a useful input solution. To enable a quick exploration of this idea, we flipped the phone and turned its front screen into a back-of-device touch area (Figure 12a). This proof-of-concept prototype allows a person in a phone call to retrieve information that can be displayed on the watch. Users can navigate between a list of frequently used apps, by swiping up and down. Once the desired app is located (Figure 12b), details can be retrieved. For example, swiping left/right on the Email app goes through the inbox where the watch shows one email (sender and subject line) at a time (Figure 12c). Tapping on an email opens it; and the user can read the email by scrolling through its text; another tap closes the email. This technique works for cases where the caller needs to quickly access information on the phone, such as recent emails, missed calls, or calendar events.

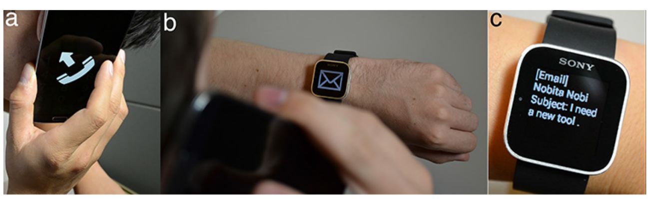

Figure 12. Enabling basic app access on the watch while using making a phone call.

USER FEEDBACK ON DUET

We gathered feedback of people using the Duet system. Our intention was not to carry out a task-based quantitative study, but rather to collect and learn from some initial user reaction and comments on our techniques.

Participants

We recruited 10 participants, five male, five female, ages 21 to 27. All participants were smart phone users. Six participants were students from three different local universities and the others were young professionals.

Procedure

We demonstrated all the Duet interactions to the participants, and asked them to comment on their *easiness* (how easy is it to perform the interaction) and *usefulness* (how is an interaction useful for some applications). The participants could also try out Duet by themselves and think aloud while exploring. The entire study took approximately 60 minutes.

Results

In general, participants gave positive feedback to Duet. The subjective rankings for each technique are shown in Figure 13. Below we summarize some of our key observations.

First, users liked how adding the watch can create lightweight interaction, which might otherwise be cumbersome on the phone alone. For example, *swipe to switch map views* was considered as a "*handy*" feature (P5), "better compared to [the] traditional way of doing it" (P1), and "reduce interaction steps" (P3).

Second, people liked using the watch as an extended display. For example, P3 liked how using the *knuckle to select emails* dispenses with UI widgets and "increases screen space", P2 commented that *flip and tap* to bring up the advanced menu "saves screen real-estate", and P8 liked how a tool palette on the watch "saves screen space" for the text in a reader.

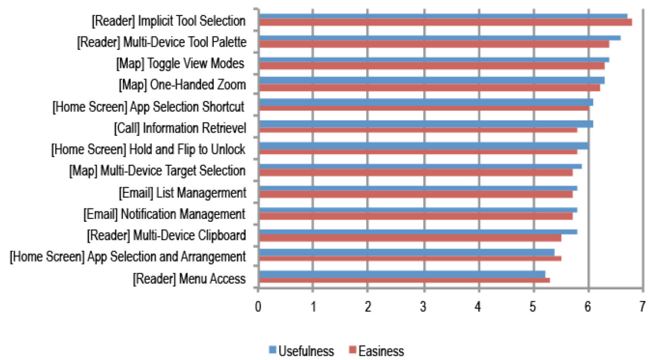

Figure 13. Subjective rankings show an overall positive reaction to Duet's interaction techniques.

Third, users enjoyed having a set of novel gestures enabled by using the watch as an auxiliary sensor. For example, most participants liked the possibility of using the watch to sense the *pad, side and knuckle* of the fingers. P2 and P5 further recommended designs that allow users to customize what each hand part does in the applications.

Last, users found it most compelling when the phone and the watch complemented one another and created interaction that went beyond their capabilities as individuals. For example, P2 considered *accessing apps on the watch while on a phone call* a feature he "can't think of any other [better] way to do it". The watch in this scenario fit in a 'niche' wherein phone-based interaction fell short. Some participants commented that *a tool pallet on the watch* resonated with their experience of using a physical toolbox (P5), pencil box (P6), and color pallet (P10): the combination of both devices did not just enable a certain function, but also created a unique and pleasant experience when using that function.

On the other hand, participants also pointed out issues and concerns with Duet. Foremost, participants often felt less enthusiastic when a Duet technique did not show significant improvement from what existing single-device designs could already achieve. For example, P7 thought *hold and flip* is good for some "niche applications" (e.g., banking), but would be "overkill" if using it to replace the existing locking/unlocking mechanisms.

People also gave mixed reviews for some of the watch-enhanced gestures. For example, both P8 and P9 pointed out that knuckle-touch could create screen occlusion and felt hard when dragging for precise positioning (e.g., arranging apps on the home screen); however, they liked using it for email selection, as this interaction required less precision and felt easier when performed with the knuckle. Finally, many participants noted the small display and touch area of the watch.

We also received valuable suggestions for additional features from the participants. Users suggested different mappings between techniques and applications, e.g., using *multi-device gestures* to copy text from the phone to the watch (P6), or as another way to authenticate the ownership of both devices (P7). A number of participants suggested including a 'fall back' option for situations where the user misplaces either device (e.g., how to unlock the phone when not having the watch for *hold and flip*). This further suggests a design challenge: how can we allow the users to transition to a multi-device paradigm and interface with them as if they were a unified interactive platform?

DISCUSSION AND FUTURE WORK

We discuss issues and questions to inform future work.

Recognition robustness. Despite the promising accuracy levels shown in the technical evaluation, it should be noted that the study was performed in a controlled lab environment. As such, there will likely be conditions where recognition rates may not perform as well. For example, our handedness detection (Figure 3d) is based on the assumption that, when wearing a watch, a touch down event will cause synchronized movement between the watch and the phone. However, a touch might only incur subtle finger motion, without detectable movement of the watch or the phone (false negatives); a bare hand's touch might also coincide with the devices' movement, thus resembling a touch caused by a watch-wearing hand (false positives). Our future work will explore software and hardware solution for mitigating this problem.

Watch wearing. Some of the Duet techniques require wearing the watch on the ventral side of the wrist to keep it readily visible and accessible (e.g., Figure 1bc). Although an elastic watchband greatly eases the switching between the two configurations (Figure 2c), a user still needs to perform the switch. Our future work will explore alternate input/output modalities on the form factor of a watch, e.g., extending the design solution of Facet [28].

Exploring the 'phrasing' of Duet. Musical communication research found that musicians use *phrasing* to structure their duet performance [11]. In particular, phrasing allows musicians to communicate with one another by delimiting their duet performance into temporal frames through which musicians *anticipate* each other's musical actions while finding room for their own musical *expression*. Similar to how Buxton articulates this concept in gesture design [6], our future work can learn from this 'phrasing' concept to extend our work to multi-device symphonic interaction. We can rethink how we can *phrase* the interaction between and by multiple devices into a fluid stream of action. Our *multi-device target selection* technique (Figure 8) has set foot in this exploration. As shown in Figure 14, this technique starts from the phone with a 'touch and hold on the map'. This leads to a 'showing touched area' on the watch, and leaves room for 'touch to adjust or select targets'. A touch release on the phone brings an end to this technique, dismissing map display on the watch, and leaving selected targets, if there is any, highlighted on the phone. All these 'components' of the techniques are phrased together, 'glued' by the muscle tension of thumb that holds down on the map [6]. By thinking in terms of phrasing in musical communication, we can explore more ways of designing interaction that spans multiple smart devices.

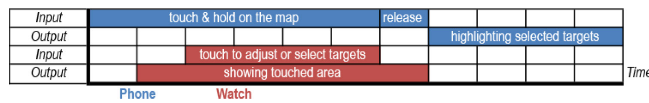

Figure 14. Phrasing between the phone and the watch in a target selection task on a map.

From Duet to Symphony. Our paper focuses on a duet of interaction between a smart phone and a smart watch. In the future, it would be interesting to consider how our design space and the interactions could be extended to not just a duet of devices, but perhaps a trio, a quartet, and eventually towards a symphony of devices and interaction [44].

CONCLUSION

Soon mobile interaction will no longer be the solo performance of the smart phone, but will rather be a symphony of a growing family of smart devices. Our Duet system reveals a design space of joint interaction between two smart devices and illustrates underexplored areas where the phone remains in the foreground of interaction, and the watch is used to enhance a wide range of phone-based interactive tasks. Our technical evaluation demonstrates the accuracy of the new gestures and sensing techniques used by Duet, and a subjective study on the Duet system provides insights, observations, and guidance for future work towards a symphony of interaction.

REFERENCES

1. Amento, B., Hill, W., and Terveen, L. The sound of one hand. *CHI '02*, 724-725.
2. Ballagas, R., Borchers, J., Rohs, M., and Sheridan, J.G. The Smart Phone. *IEEE Pervasive Computing 5*, 1 (2006), 70–77.
3. Baudisch, P. and Chu, G. Back-of-device interaction allows creating very small touch devices. *CHI '09*, 1923–1932.
4. Butler, A., Izadi, S., and Hodges, S. SideSight. *UIST '08*, 201–204.
5. Buxton, W. Integrating the periphery and context: A new taxonomy of telematics. *GI '95*, 239–246.

6. Buxton, W.A.S. Chunking and phrasing and the design of human-computer dialogues. *IFIP ‘86*, 494–499.
7. Chen, G., and Kotz, D., *A survey of context-aware mobile computing research*. Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.
8. Crossan, A., Williamson, J., Brewster, S., and Murray-Smith, R. Wrist rotation for interaction in mobile contexts. *MobileHCI ’08*, 435–438.
9. Falk, J. The conversational duet. *Proceedings of the Annual Meeting of the Berkeley Linguistics Society*, 2011.
10. Fitzmaurice, G.W. Situated information spaces and spatially aware palmtop computers. *CACM 36*, 7 (1993), 39–49.
11. Gratier, M. Grounding in musical interaction: Evidence from jazz performances. *Musicae Scientiae 12*, 1 Suppl (2008), 71–110.
12. Harrison, C. and Hudson, S.E. Abracadabra. *UIST ’09*, 121–124.
13. Harrison, C., Schwarz, J., and Hudson, S.E. TapSense. *UIST ’11*, 627–634.
14. Hinckley, K., Pierce, J., Horvitz, E., and Sinclair, M. Foreground and background interaction with sensor-enhanced mobile devices. *TOCHI ’12*, 1 (2005), 31–52.
15. Hinckley, K., Pierce, J., Sinclair, M., and Horvitz, E. Sensing techniques for mobile interaction. *CHI ‘00*, 91–100.
16. Hinckley, K., Ramos, G., Guimbretiere, F., Baudisch, P., and Smith, M. Stitching. *AVI ’04*, 23–30.
17. Hinckley, K. Synchronous gestures for multiple persons and computers. *UIST ’03*, 149–158.
18. Holmquist, L.E., Mattern, F., Schiele, B., Alahuhta, P., Beigl, M., and Gellersen, H. Smart-Its Friends. *Ubicomp ‘01*, 116–122.
19. Hudson, S.E., Harrison, C., Harrison, B.L., and LaMarca, A. Whack gestures. *TEI ’10*, 109–103.
20. Ishiguro, Y., Mujibiya, A., Miyaki, T., and Rekimoto, J. Aided eyes. *AH ’10*, 1–7.
21. Jones, B., Sodhi, R., Forsyth, D., Bailey, B., and Maciocci, G. Around device interaction for multiscale navigation. *MobileHCI ’12*, 83–92.
22. Kim, D., Hilliges, O., Izadi, S., et al. Digits. *UIST ’12*, 167–176.
23. Kim, J., He, J., Lyons, K., and Starner, T. The Gesture Watch. *ISWC ’07*, 1–8.
24. Kortuem, G., Kray, C., and Gellersen, H. Sensing and visualizing spatial relations of mobile devices. *UIST ’05*, 93.
25. Kray, C., Rohs, M., Hook, J., and Kratz, S. Group coordination and negotiation through spatial proximity regions around mobile devices on augmented tabletops. *3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*, 1–8.
26. Lee, S.C., Li, B., and Starner, T. AirTouch. *ISWC ’11*, 3–10.
27. Lucero, A., Keränen, J., and Korhonen, H. Collaborative use of mobile phones for brainstorming. MobileHCI '10, 337.
28. Lyons, K., Nguyen, D., Ashbrook, D., and White, S. Facet. *UIST ’12*, 123–130.
29. Mann, S. Smart clothing. *CACM 39*, 8 (1996), 23–24.
30. Mann, S. `WearCam’ (The Wearable Camera). *ISWC ’98*, 124–131.
31. Marquardt, N., Ballendat, T., Boring, S., Greenberg, S., and Hinckley, K. Gradual engagement. *ITS ’12*, 31–40.
32. Maurer, U., Rowe, A., Smailagic, A., and Siewiorek, D.P. eWatch. *BSN’06*, 142–145.
33. McCallum, D.C. and Irani, P. ARC-Pad. *UIST ’09*, 153.
34. Merrill, D., Kalanithi, J., and Maes, P. Siftables. *TEI ’07*, 75–78.
35. Ni, T. and Baudisch, P. Disappearing mobile devices. *UIST ’09*, 101–110.
36. Oney, S., Harrison, C., Ogan, A., and Wiese, J. ZoomBoard. *CHI ’13*, 2799–2803.
37. Pebble. Pebble E-Paper Watch. http://getpebble.com/.
38. Post, E.R. and Orth, M. Smart fabric, or “wearable clothing.” *ISWC ’97*, 167–168.
39. Rahman, M., Gustafson, S., Irani, P., and Subramanian, S. Tilt techniques. *CHI ’09*, 1943–1952.
40. Rekimoto, J. Tilting operations for small screen interfaces. *UIST ’96*, 167–168.
41. Rekimoto, J. Pick-and-drop. *UIST ’97*, 31–39.
42. Rekimoto, J. GestureWrist and GesturePad. *ISWC ’01*, 21–27.
43. Ruiz, J. and Li, Y. DoubleFlip. *CHI ’11*, 2717–2720.
44. Santosa, S. and Wigdor. D.. A field study of multi-device workflows in distributed workspaces. *UbiComp ’13*. 63-72.
45. Schilit, B., Adams, N., and Want, R. Context-Aware Computing Applications. *First Workshop on Mobile Computing Systems and Applications*, 85–90.
46. Schmidt, D., Seifert, J., Rukzio, E., and Gellersen, H. A cross-device interaction style for mobiles and surfaces. *DIS ’12*, 318–327.
47. Siek, K.A., Rogers, Y., and Connelly, K.H. Fat finger worries *INTERACT ’05*, 267–280.
48. Streitz, N.A., Konomi, S., and Burkhardt, H.-J., Roomware for cooperative buildings. *Cooperative Buildings: Integrating Information, Organization, and Architecture, 1998. 4-21.*
49. Strohmeier, P., Vertegaal, R., and Girouard, A. With a flick of the wrist. *TEI ’12*, 307–308.
50. Vogel, D. and Baudisch, P. Shift. *CHI ’07*, 657–666.
51. Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., and Shen, C. Lucid touch. *UIST ’07*, 269–278.
52. Yee, K.-P. Peephole displays. *CHI ’03*, 1–9.