Doppio: A Reconfigurable Dual-Face Smartwatch for Tangible Interaction

Teddy Seyed[1], Xing-Dong Yang[2], Daniel Vogel[3]
University of Calgary[1], Dartmouth College[2], University of Waterloo[3]
teddy.seyed@ucalgary.ca, xing-dong.yang@dartmouth.edu, dvogel@uwaterloo.ca

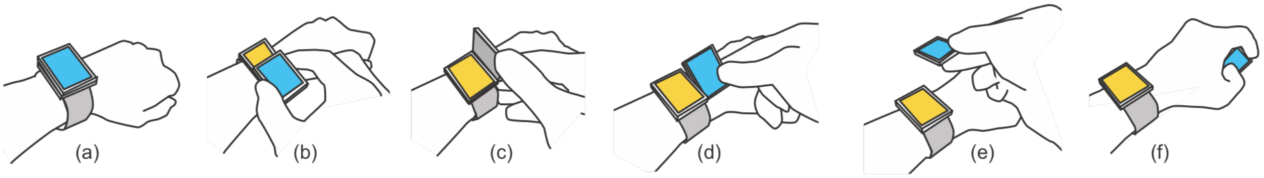


Figure 1. Illustration of Doppio interaction concept with tangible input examples: (a) top stacked on base for default use; (b) slide down to peek at information; (c) hinge to view a background app; (d) adjacent with hinge and pivot manipulation; (e) detached, distant top for sharing; (f) placing top in hand for indirect input. See also the accompanying video figure.

ABSTRACT

Doppio is a reconfigurable smartwatch with two touch sensitive display faces. The orientation of the top relative to the base and how the top is attached to the base, creates a very large interaction space. We define and enumerate possible configurations, transitions, and manipulations in this space. Using a passive prototype, we conduct an exploratory study to probe how people might use this style of smartwatch interaction. With an instrumented prototype, we conduct a controlled experiment to evaluate the transition times between configurations and subjective preferences. We use the combined results of these two studies to generate a set of characteristics and design considerations for applying this interaction space to smartwatch applications. These considerations are illustrated with a proof-of-concept hardware prototype demonstrating how Doppio interactions can be used for notifications, private viewing, task switching, temporary information access, application launching, application modes, input, and sharing the top.

Author Keywords

Wearable; tangible; smartwatch; interaction techniques

ACM Classification Keywords

H.5.2. Information Interfaces (e.g., HCI): Input devices

INTRODUCTION

Smartwatches may be a convenient way to access information, but interaction is challenging due to the small form factor of the screen area. A small display size means only a limited amount of information can be shown. Proposed solutions to this problem include switching to a smartphone or tablet for some usage scenarios [14], extending the display by projecting onto the forearm [24], or placing touch displays around the entire watch band [21]. Additionally, since most watches use direct touch on the display, the small display combined with fat fingers is problematic [31]. Using off-screen touch input [16, 20, 27] or physical knobs and buttons, solves occlusion but limits the input space, while using voice input is conspicuous and slow [11]. Researchers have responded with more expressive sensing to enable gestural interaction around the watch [26, 30] or by manipulating the entire watch face, similar to a joystick [32].

We significantly extend the idea of manipulating a watch as input by adding a second face that can be attached, oriented, and manipulated around a base watch face. This not only doubles the display area and can keep fingers off the display, but also creates a tangible input language with manipulations like stacking, peeking, hinging, adjacency, distance, and indirect input (see examples in Figure 1). We call this concept "Doppio", meaning double. It can be thought of as combining and tailoring aspects of Codex [17], Paddle [29], and Siftables [22] into a smartwatch form factor.

The primary contributions of our work are: 1) the concept of a reconfigurable dual-face smartwatch that enables tangible interaction and a larger display area; 2) the results of an exploratory study with a passive prototype probing how people might use this design space; 3) the results of a controlled experiment with an instrumented prototype to evaluate transition times between configurations and subjective preferences for different configurations and transitions; and 4) a proof-of-concept hardware prototype demonstrating how this style of interaction can be used.

RELATED WORK

After surveying smartwatch interaction techniques, we describe inspirational dual-faced conventional watches and relate our work to tangible multi-display devices.

Enhancing Smartwatch Input

Since touch is arguably the most common input method for smartwatches, mitigating the fat-finger problem and increasing the touch input space are well explored areas. For

CHI'16, May 07-12, 2016, San Jose, CA, USA

DOI: http://dx.doi.org/10.1145/2858036.2858256

example, to accurately type on very small keyboard keys, Zoomboard [25] uses multiple taps to zoom in on a specific key, while Swipeboard [13] uses consecutive directional swipes to select a character. TouchSense [18] is an example of increasing the touch input space, in this case by exploiting differences in finger contact area when touching at different angles. Other approaches move touch input off of the display to reduce finger occlusion, sometimes also increasing the input space. For example, Ashbrook et al. [7] investigate touch on a round bezel, Blasko and Feiner [8] propose stroke gestures around the watch face guided by tactile patterns, Oakley et al. [23] explore capacitive touch sensing around the outside of the watch case, Watchit [27] enables a touch interaction language on the watch band, Laput at al. [20] demonstrate touch sensitive buttons on the skin near the watch, Blasko et al. [9] prototype watch interaction using a retractable string, Abracadabra [16] and Gesture Watch [19] detect finger movements in the air near the watch, and Gesturewrist [30] detects posture of the hand using an instrumented wrist band. Expanding touch input space is not our focus, but in principle, all these ideas could be combined with Doppio's tangible interaction.

Two examples more closely related to our focus are Pasquero et al.'s haptic wristwatch [26] that combines input by twisting the watch frame with finger gestures and touch, and Xiao et al.'s [32] method of twisting, tilting, and pushing the entire watch face when attached to the band on a joystick-like mount. These examples demonstrate interactions that go beyond form and appearance of typical smart watch. Doppio also supports similar interactions and many more through its detachable top face design.

Extending the Smartwatch Display

Duet [16] expands the conventional approach of pairing a smartwatch with a smartphone into a cross-device and contextually-linked interaction language and WatchConnect [20] is a toolkit with a similar goal with wall-sized displays. More relevant are techniques that do not rely on any external device or display. Xu et al. study minimal smartwatch displays [33] and Lenovo demonstrated a watch with a second display viewable by holding it to your eye [17], but a more typical approach is adding displays near the wrist. AugmentedForearm [24] proposes using the entire forearm as a watch-like display, which is demonstrated with a prototype constructed from four smartwatches. Facet [21] is a watch-like bracelet made of multiple touchscreens and DisplaySkin [10] has a similar goal using a single flexible display wrapped around the wrist. Although these approaches significantly expand the display size, they also move away from the standard look and feel of a watch. The Doppio concept builds multiple small displays worn near the wrist, but adds the dimension of reconfigurable displays.

Novel wristwatch designs

Conventional watches are both utilitarian and aesthetic objects [7, 33], so we were inspired by how conventional wristwatches position and manipulate two watch faces. The *Titanium Two Face* [6] has two adjacent watch faces to display two time zones. The *Piaget Altiplano Double Jeu* [3] has a hinged top face that opens to reveal a second dial on the watch base and the *Porsche Design P'6520 Heritage Compass* [4] uses a similar hinged top face that opens to reveal a compass underneath. The *Jaeger-LeCoultre Reverso* [2] and *Ritmo Mvndo Persepolis* [5] place the two faces on either side of the watch case, such that it can be flipped to switch between two dials. *Halda's Space Discovery* [1] comes with two faces, analog and digital, that can be swapped into a single housing on the band. The Doppio interaction language has a similar adjacent configuration with two faces side-by-side, a hinge action to access a secondary face, and a top face that can be detached.

Multi-display Tangible Devices

Multiple connected displays can also expand the interaction space. The Codex [17] tablet has two hinged displays and uses hinge angle and orientation to changes between eleven modes including a detached mode by removing a display from the case for sharing. PaperFold [15] extends Codex to three hinged displays, but with permanently connected displays. Paddle [32] has mechanical connections to fold into seven different shapes forming a tangible interaction language. The device requires two-hands, but introduces a "peeking" interaction expanded in our work. Siftables [22] demonstrate how multiple small displays can use orientation, proximity, and side-by-side adjacency for tangible interaction. The displays are not physically connected and are designed to be used on a table.

The Doppio concept combines and extends these ideas: hinging from Codex; peeking from Paddle; and orientation and adjacency from Siftables. However, Doppio is based around a smaller form factor and is specifically designed for a smartwatch context with tangible constraints such as an anchored display and one-handed manipulation. In spite of these constraints, a larger interaction space is created exploiting the reconfigurable nature of two displays that can be mechanically detached and joined in many ways.

DEVICE DESIGN OPTIONS

A number of different options influence the form factor and design of a Doppio device. As a convention, we refer to the two display faces in a Doppio smartwatch as the "top" and the "base." The base is securely attached to the band like a standard smartwatch, but the top can be attached to the base face, or any of its sides. While attached, the top can be slid, rotated, tilted or detached completely. When attached to the face of the base (called a "stacked" configuration), the combined top and base resemble a standard smartwatch. The following general requirements inform design choices for the top and base, attachment mechanisms, and sensors.

Top and Base Shapes

The shape of the top and base faces influences interaction affordances and capabilities. If both faces are rectilinear (e.g. square) or have large facets (e.g. hexagonal), the top may be stably attached to the sides of the base and different attachment points are well-defined. If both faces are curvilinear (e.g. circle), attaching the top to the base is less stable and less well-defined, but the top can be continuously rolled around the base. If the faces have an aspect ratio other than one (e.g. rectangular) there can be a landscape versus portrait distinction, and rotating the top while stacked on the base creates "off-axis" orientations that can be differentiated visually and by touch. We use rectangular faces in our prototypes and illustrations.

Mechanism to Attach Top to Base

Enabling high degree-of-freedom movements for the top when attached to the base, while also enabling the top to be detached, is challenging. We experimented with mechanical methods using LEGO and low-fidelity mockups and found ball joints mounted on a turntable a reasonable subset of desired movements. However, mechanical joints are difficult to fit into a small device and the movement constrains are limiting. We use magnets placed at key locations. These enable a variety of physical attachment locations that can be "broken" to manipulate or detach the top. Magnets can be easily integrated and allow large movement flexibility. The magnetic force also has an exploitable quality as the top is pulled towards some locations and repelled from others. However, a magnetic connection is less strong and the top could fall. A hybrid approach is tethering the top to the base using an elastic or retractable string. This would enable string-based interaction [9, 28], but would also be technically challenging to integrate in such a small form factor.

Sensors

Both faces have a digital display and touch screen, but other sensors are needed to detect orientation, attachment location, and position of the top relative to the base. Magnetic sensors can track position [12] and attachment location, but precision is reduced due to attachment magnets. IMUs and magnetometers can track relative orientation of the top for rotating, hinging and pivoting. Proximity, light, force, or capacitive sensors can detect when and how the top is attached to the base. We use capacitive and IMU sensing.

INPUT SPACE

There are numerous ways to attach the top to the base and different ways for the top to be in a detached state. This creates a set of 124 *configurations* with many possible *transitions* between them. Additional *manipulations* can vary a configuration or transition, e.g. rotating, tilting, sliding, or moving the top relative to the base and touching the display of the base or top. In this section, we define and enumerate possibilities and define a concise terminology and syntax used throughout the remainder of the paper. We have not yet evaluated cognitive demands or semantics associated with this interaction vocabulary. Our aim is to show the range of possible interactions. In later sections, we observe how this input space could be used, evaluate times and preference for key transitions, and demonstrate how this input space can be applied to real applications.

Configurations

The configuration is determined by how the top is attached (or not attached) to the base and the orientation and position of the top relative to the base. The number of variations for each configuration type is noted in parenthesis and Figure 2 illustrates example configurations.

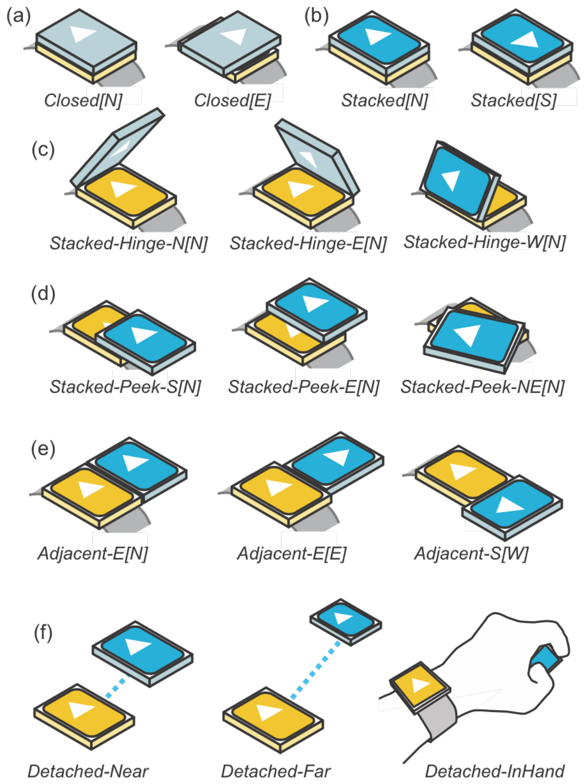


Figure 2. Example Configurations.

Closed (4) — The top is face down, attached to the face of the base such that no display is visible. The top can be rotated around the normal vector of the base, with the top orientation expressed as a compass direction relative to the base (N, S, E, W). As a convention, the relative orientation of the top is given in square brackets after the configuration name, e.g. "*Closed[N]*", "*Closed[E]*", etc. (Figure 2a).

Stacked (4) — The top is face up, attached to the top of the base such that only the top display is visible. The relative orientation of the top is the same as *Closed, e.g. "Stacked[N]", "Stacked[E]", etc.* (Figure 2b).

Stacked-Hinge (16) and *Closed-Hinge* (16) — One side of the top is attached to the base, resembling a hinged door or partially opened book, revealing the base face beneath. The relative orientation of the top is the same as *Closed.* The side of the base where the top is attached is given as compass direction (relative to base) after the configuration name, e.g. "*Stacked-Hinge-N[N]*", etc. (Figure 2c).

Stacked-Peek (32) and *Closed-Peek* (32) — The top is attached to the base face, but partially slid in a direction so that a portion of the base face is revealed. The relative orientation of the top is the same as *Closed*, given as the pre-peek orientation. The direction that top has slid is given as compass direction (relative to base) after the configuration name, e.g. "*Stacked-Peek-S[N]*". Off axis peeks with ~45°

rotation are possible for peeking at a corner of the base, e.g. “*Stacked-Peek-SW[N]*”. (Figure 2d).

Adjacent (16) — The top and base are side-by-side, planar or near planar. The side of the base to where the top is attached is given as a compass direction (relative to base) after configuration name, e.g. “*Adjacent-E[N]*”. The orientation of the top is the same as *Closed* (Figure 2e).

Detached (4) — When the top and base are detached, the top can be near the base (“*Detached-Near*”) or far from the base (“*Detached-Far*”). The exact threshold for near and far is determined by hardware constraints, but the guideline is that the top is near when the user is holding the top. Two special detached cases are when the top is held in the watch hand, “*Detached-InHand*”, or when mounted on the watch band, “*Detached-Band* (Figure 2f).

Transitions

Different physical actions are employed when transitioning between configurations, such as sliding the top on the base, sliding the top off of the base, attaching and detaching the top from the base, and rotating the top on the base of the face, its side, or edges (see Figure 3 for examples).

Transitions often pass through intermediate configurations. Consider transitioning from *Stacked[N]* to *Adjacent-E[E]*: one path is to slide from *Stacked[N]* to *Adjacent-E[N]* then a momentary detach, rotate, and attach the top to arrive at *Adjacent-E[E]*; another path is a rotation from *Stacked[N]* to *Stacked[E]*, then a slide to *Adjacent-E[E]*. Intermediate configurations can be filtered with a threshold, but the transition path can also form part of the interaction language.

Manipulations

The top can also be manipulated within the current configuration (see Figure 4 for examples).

Rotation — In *Adjacent* configurations, the top can *pivot* around the normal vector of the attached side or *hinge* around the attached side. The top can also pivot slightly around the normal vector of the face without leaving a specific *Stacked* or *Closed* configuration. The concise notation to specify these manipulations adds the axis (*hinge* or *pivot*) and angle between the square brackets (e.g. “*Adjacent-E[N, pivot 15°]*”, “*Adjacent-E[N, hinge -45°]*”, or “*Stacked-N[N, pivot -15°]*”). Note that rotating beyond a threshold will transition to another configuration, for example pivoting more than 45° in *Stacked[N]* transitions to *Stacked[E]*.

Translation — In *Detached-Near*, the position of the top relative to the base could be tracked. This could act as input for the base, or navigate information in the top.

Touch — Manipulations can also overload the meaning of a transition. For example, transitioning from *Stacked-Peek-S[N] to Stacked[N]* could be done while simultaneously touching the top face to change system behavior.

EXPLORATORY STUDY

To understand how people might use this input space, we conducted an exploratory study with six participants familiar with interaction design and wearable technologies. Our goal was to discover usage themes rather than thoroughly explore specific interactions. To help ground the discussion and provide the participants with a way to demonstrate their ideas, we supplied a simple cardboard prototype (Figure 5). The prototype used magnets to connect the top and base together such that the top was free to be moved and detached.

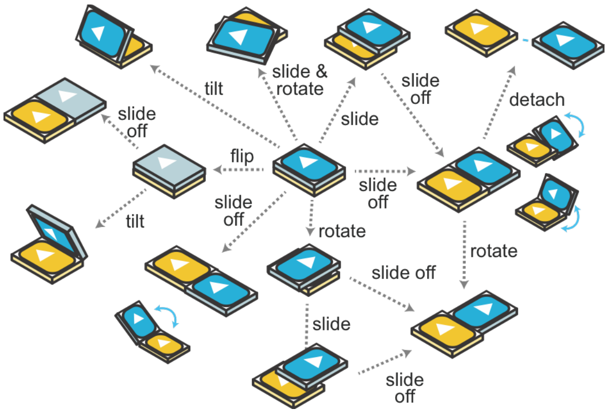
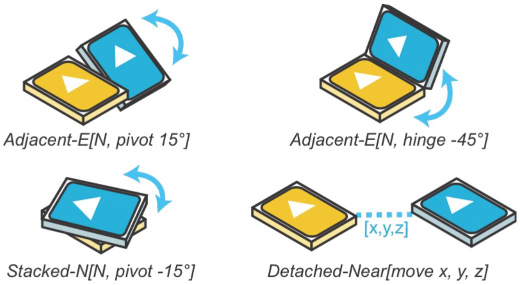


Figure 3. Example Transitions.



Adjacent-E[N, pivot 15°] *Adjacent-E[N, hinge -45°]*

Stacked-N[N, pivot -15°] *Detached-Near[move x, y, z]*

Figure 4. Example Manipulations.

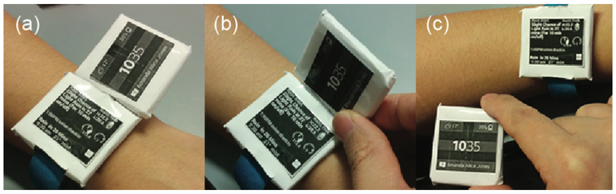


Figure 5. Paper prototype: (a) top attached to the side; (b) top hinged open; (c) top detached from the base.

Protocol

Participants were shown how to form different configurations and then encouraged to discuss situations in which they might use such a dual-face watch. To prompt thoughts and ideas, we provided pictures of common mobile applications (e.g. clock, calendar, weather). The study was video recorded so actions and behaviors could be examined. Comments were transcribed for thematic grouping.

Results

Overall, participants were positive about the form factor. Several commented on the usefulness of multiple displays, for example “*… with a regular smartwatch, I don’t feel like reading emails on it, but a dual-screen watch adds more space and makes me feel that reading emails becomes possible*” [P3], and “*I like the form factor of the dual-screen* *watch and the fact that I can reconfigure the position of the top screen*" [P5]. Eight usage themes emerged:

E1 Attaching the top to one side of the base can form a single unified display and interaction space, or each display can function separately for different tasks.

E2 Some types of tangible input, like sliding the top a short distance, lets you provide input without covering the display with a finger.

E3 The physical nature of tangible input seems like it would enable muscle memory for eyes-free interaction.

E4 Attaching the top to the left or right side is more suitable for touch input since the top can rest on the arm.

E5 When attached to the left or right side of the base, the maneuverability of the top can be impeded by the arm.

E6 Once detached, the top could be handed to another person for sharing content like photos. However, a shared top should restrict access to personal information.

E7 While attached to the base, the top can be hinged up and the face oriented away like a small public display, leaving the base as a private display.

E8 When the top is hinged up and attached to the top of the base, it acts like a privacy shield.

CONTROLLED EXPERIMENT

From our exploratory study, we observed ergonomics and movement difficulty influenced input space usability. The goal of our controlled study is to measure how these issues affect transitions between key configurations by measuring time and subjective preference. The results inform interaction design considerations discussed later.

Participants

We recruited 12 participants (3 female), ages 18 to 45. All were right handed, and 2 currently owned a smartwatch.

Apparatus

An instrumented Doppio prototype was used to log transition times between configurations (Figure 6). The top and base are each 42 × 36 × 7 mm (similar size as the *Apple Watch)*. Capacitive sensing is used to detect configurations. Four steel electrodes are attached to the base such that each electrode covers the side with a small portion wrapped around to the face (Figure 6b). Portions of the top are made conductive by completely wrapping the sides and face with copper tape, but leaving only eight smaller conductive areas on the back. The four base electrodes are wired to an Arduino to measure capacitive strength. Neodymium magnets are embedded into the top to increase connection reliability and provide a feeling of physical connection. A thumbpiece (Figure 6c) with a single electrode is used to accurately detect when the top was held in the hand.

The pattern of conductive areas on the base and top enable robust detection of groups of configurations, but not specific configurations. It can detect when the top is in a *Stacked* or *Closed* configuration, but not which one or the orientation of the top. Likewise, it can detect a *Stacked-Peek* configuration, but not the peek direction. It can also distinguish between *Adjacent-N, Adjacent-E, Adjacent-S,* and *Adjacent-W*, but not which side of the top is attached to the base. However, detecting when a configuration in a group is entered and exited provides the necessary data.

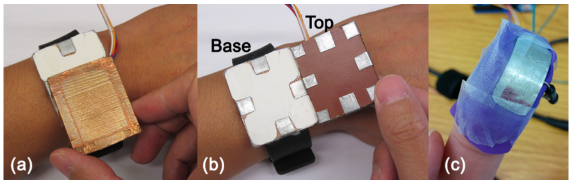


Figure 6. Prototype for timing transitions: (b) base electrode placement and top conductive pattern; (c) thumb piece.

A C# application prompted participants to perform transitions according to the experiment design. By prompting participants to transition from a certain start configuration to a certain end configuration, we could record specific transition times. During the experiment, we monitored their compliance and provided a redo function (accessible to both experimenter and participant) when a wrong configuration was accidently transitioned to. The C# application logged events from the Arduino synchronized to the prompted transition. In many cases, the transition time is simply the duration between two detected configurations, but there are cases where we ignore an assumed intermediate configuration. For example, when rotating the top to transition from *Stacked[N]* to *Stacked[S]*, we ignore the intermediate con figuration since it must be *Stacked[W]* or *Stacked[E].*

Task

Participants were required to transition from a specific starting configuration to a specific ending configuration. The start and end configurations were shown as graphical illustrations (like Figures 1-4). Illustrations and instructions were shown together on a notebook computer placed such that participants could see the watch and visual instructions simultaneously. All tasks were performed while standing.

The participant positioned the top in the starting configuration before a trial started, and transitioned to the end configuration as quickly and accurately as possible. When a trial was completed, a sound was played and the visuals and instructions were updated to reflect the next trial. Task completion time is the time from when the start configuration was exited until the ending configuration was entered.

An error occurred when the device did not detect the correct ending configuration, if the experimenter observed a wrong configuration or a detection failure, or if the participant self-reported a mistake or a technical problem. In all cases, an error tone was played and the trial was repeated.

Design and Protocol

From the input space, we evaluated 37 TRANSITIONS spanning 7 transition FAMILIES (listed in Figure 7). Many tested transitions are equivalent to non-tested ones by symmetry (e.g. the tested transition from *Stacked[N]* to *Stacked[S]* is equivalent to *Stacked[S]* to *Stacked[N]*). *Detached-Far, Stacked-Hinges,* and manipulations were not included due to ambiguity in the ending configuration position. We included *Stacked-Hinge* in a preference questionnaire.

The experiment design used 1 full practice block and 2 measurement blocks. Each block presented all TRANSITIONS grouped by FAMILY, with FAMILY ordered by increasing complexity of movement: *Stacked Rotations, Closed, Detaching, Straight Peeks, Corner Peeks, Stacked to Adjacent, Adjacent Rotations*. This aided learning compared to a counter-balanced design, and order effects are minimized because all families are presented in each block. Within each FAMILY, 3 repetitions of related transitions were presented in randomized order (e.g. the 3 related transitions in *Stacked Rotations* from *Stacked[N]* to Stacked[E], *Stacked[S]*, and *Stacked[W]*, see arrows in Figure 7). Participants took breaks as needed. In summary: 2 blocks × 37 TRANSITIONS (grouped by 7 FAMILIES) × 3 repetitions.

After the timed portion of the study, participants completed a questionnaire asking for subjective preference for all transitions in the timed portion with the addition of 4 *Stacked-Hinge* transitions. Ratings were from 1 to 7 using a continuous numeric scale (1 least preferred, 7 most preferred with decimal ratings like 3.5 permitted). Including questionnaire, the experiment averaged 40 minutes.

Data Processing and Statistical Analysis

Outlying trials with times more than three standard deviations from the mean were removed (34 trials, 1.3%). Repeated-measures ANOVA and pairwise t-tests with Holm correction were used for all measures. Trials are aggregated by participant and the factors under analysis. To correct for non-normally distributed time data, all statistical tests are performed on log transformed times.

Results

Time by Transition Family

There is a significant effect for FAMILY on time ($F_{6,66} = 15.21$, $p < .001$, $\eta^2 = .33$). Post hoc tests show: *Adjacent Rotations* (860 ms, SEM 14) are slower than all other families except *Detaching* (795 ms, SEM 29) and *Closing* (584 ms, SEM 37) (all $p < .05$); *Detaching* is slower than *Stacked Rotations* (572 ms, SEM 21), *Stacked to Adjacent* (541 ms, SEM 11), *Corner Peeks* (499 ms, SEM 17), and *Straight Peeks* (383 ms, SEM 8) (all $p < .01$); and *Straight Peeks* are faster than *Stacked Rotations* and *Stacked to Adjacent*. *Adjacent Rotations* and *Detaching* incur more movement overhead, but all family transition times are reasonable suggesting this style of tangible interaction is feasible.

Time by Transition

Figure 7 illustrates times for all tested transitions. We analyze each FAMILY with more than 1 transition separately. For *Straight Peek* and *Corner Peek* families, there are no significant effects of TRANSITION on time. *Peeking* is already among the fastest families, and this indicates there is little difference in time for different peek directions.

For the *Stacked Rotation* family, there is a significant effect of TRANSITION on time ($F_{2,22} = 23.748$, $p < .001$, $\eta^2 = .19$). Post hoc tests show *Stacked[N] to Stacked[S]* (755 ms, SEM 46) was slower than other transitions (mean 482 ms). Turning 180° is almost twice as slow as 90°: our observations suggest this is due to the majority of participants using a two-part turning strategy for *Stacked[N] to Stacked[S]* where the top is turned by 90 in two operations. Some participants discovered a faster strategy by grasping the top from the North side and turning it 180° in one operation.

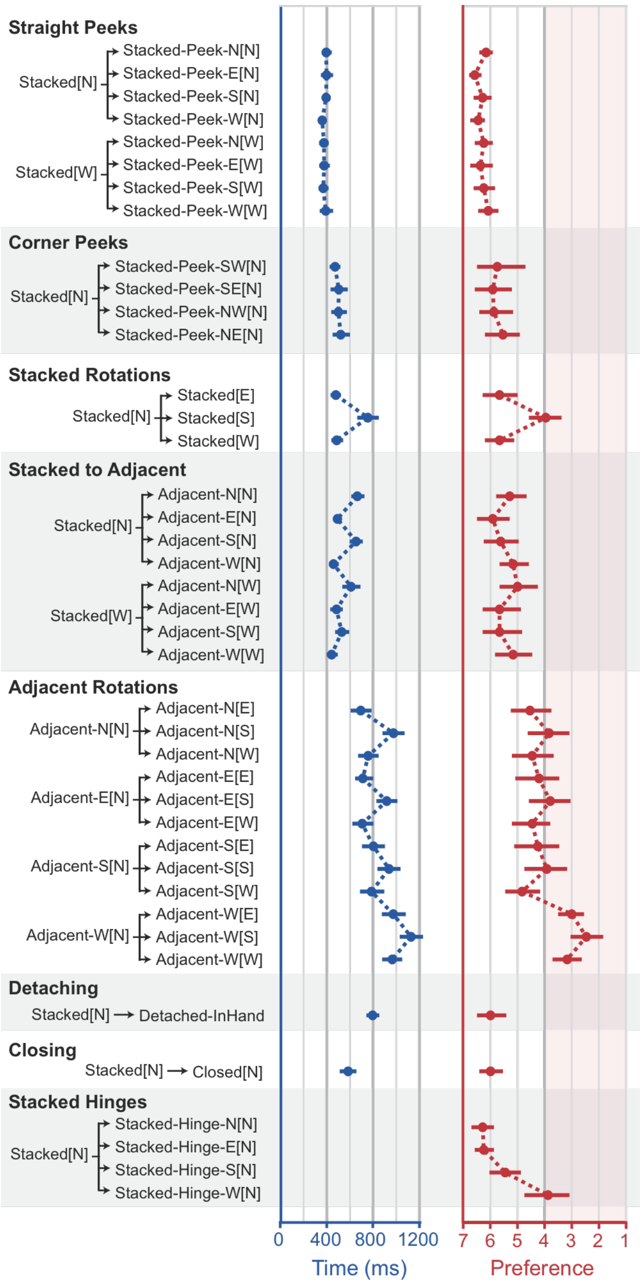


Figure 7. Time and Preference by Transition. Preference axes flipped to show correspondence with Time. Stacked Hinge times not tested. Error bars are 95% confidence interval.

For the *Stacked to Adjacent* family, there is a significant effect of TRANSITION on time ($F_{7,77} = 7.1413$, $p < .001$, $\eta^2 = .20$). Post hoc tests show *Stacked[N] to Adjacent-N[N]* and *Stacked[N] to Adjacent-S[N]* are both slower than *Stacked[N] to Adjacent-E[N], Stacked[N] to Adjacent-*

W[N], Stacked[W] to Adjacent-W[W], and *Stacked[W] to Adjacent-E[W].* The general pattern is that attaching the top to the East or West sides is faster than North or South. We believe the advantage may be attributed to the way participants rest their hand holding the top on the other arm to make alignment of the top with the side of base easier.

For the *Adjacent Rotations* family, there is a significant effect of TRANSITION on time ($F_{11,121} = 8.911$, $p < .001$, $\eta^2 = .15$). Post hoc tests show: *Adjacent-E[N] to Adjacent-E[S]* is slower than *Adjacent-E[N] to Adjacent-E[E]; Adjacent-N[N] to Adjacent-N[S]* is slower than *Adjacent-N[N] to Adjacent-N[E]; Adjacent-S[N] to Adjacent-S[S]* is slower *than Adjacent-N[N] to Adjacent-N[W]; Adjacent-W[N] to Adjacent-W[S]* is slower than *Adjacent-E[N] to Adjacent-E[E] and Adjacent-N[N] to Adjacent-N[E].* Similar to the results for *Stacked Rotations*, 180° rotations incur overhead. In addition, there were some differences in 90° rotations: *Adjacent-W[N] to Adjacent-W[W]* is slower than *Adjacent-E[N] to Adjacent-E[E], Adjacent-N[N] to Adjacent-N[E],* and *Adjacent-N[N] to Adjacent-N[W]*. This provides some statistical evidence supporting the general trend visible in Figure 7: *Adjacent-W* rotations are more costly.

Subjective Preference

Figure 7 illustrates mean preferences for all tested transitions. We analyze each FAMILY with more 1 transition separately. For the most part, the pattern of preference corresponds to the pattern of times, and the significant differences follow a similar trend. For *Straight* and *Corner Peeks*, and *Stacked to Adjacent* there are no significant TRANSITION on preference effect. For the *Stacked Rotation* family, there is a significant effect of TRANSITION on preference ($F_{2,22} = 17.406$, $p < .0001$). Post hoc tests show *Stacked[N] to Stacked[S]* is less preferred (all $p < .01$), the same pattern as time. For the *Adjacent Rotations* family, there is a significant effect of TRANSITION on preference ($F_{11,121} = 6.721$, $p < .0001$). Post hoc tests found 15 pairwise differences, all involving *Adjacent-W[N] to Adjacent-W[E], Adjacent-W[N] to Adjacent-W[S],* and *Adjacent-W[N] to Adjacent-W[W].* In each case these 3 transitions were less preferred than the others (all $p < .05$). This supports and extends the partial trend in significant differences of time.

The *Stacked Hinge* family was not included in the timing portion, but transitions in the family have a significant effect of TRANSITION on preference ($F_{3,33} = 15.793$, $p < .0001$). Post hoc tests revealed a strong preference order: *Stacked-Hinge-W[N]* is least preferred, then *Stacked-Hinge-S[N],* and then *Stacked-Hinge-E[N]* and *Stacked-Hinge-N[N]* which are both most preferred.

Discussion

The combined time and preference results from the controlled study indicate that the majority of transitions are reasonable or cumbersome. *Peeks* are fastest (less than 400ms on average) and all variations received similar preference. *Adjacent Rotation* is the slowest family, but within an acceptable 860ms on average and the majority of its variations are rated neutral or marginally positive. Our results do identify transitions that should be used sparingly or avoided: all *Adjacent Rotations* on the west side and perhaps *Stacked[N] to Stacked-Hinge-W[N]* based on preference. Transitions requiring 180° top rotations were somewhat slower in all families and less preferred in many cases. We believe this is partly due to a commonly used inefficient rotation movement strategy, but regardless, transitions with more than 90° top rotations should be used with caution.

DESIGN CONSIDERATIONS

Based on device options, input space, and results of the formative and controlled studies, we generated a set of design considerations for functional characteristics and mappings for Doppio interaction. Related formative study themes and controlled study results noted when applicable.

Functional Characteristics

Characteristics are opportunities, affordances, and constraints that influence how the input space is used.

One-Handed - Interaction is performed with the non-watch hand, but interesting counter-examples are when the top is held by the watch hand or operated by both hands. For example, the top may be held with the watch hand so thumb touches provide indirect input to the base (theme E6).

Stability – The magnets create a spring-like resistance favouring certain positions and orientations in attached configurations. During transitions, the top snaps into most configurations, and manipulations can be reset by snapping back to a preferred position and orientation (theme E3).

Shape - Given the rectangular shapes, *Stacked[E]* and Stacked*[W]* feel temporary. A wide *Adjacent* configuration is well suited to displaying larger information spaces (E1).

Touch Constraints - Touch input is not always possible or comfortable with the base and top. For example, the base face is impossible to touch in *Stacked*, possible to touch in *Stacked-Hinge,* and easy to touch in some *Adjacent* and *Detached* configurations. When *Adjacent*, the top is easiest to touch in *Adjacent-W* or *Adjacent-E* given arm support (E4).

Ergonomics - Configurations are not equal in terms of comfort and usability (E5). For example, compared to *Adjacent-E,* transitioning to *Adjacent-W* is awkward due to hand-over-hand action and occlusion. Configurations that orient the top away from the user (like *Stacked-Hinge-W[N]*) make the top awkward to view or manipulate.

Functional Mappings

Design characteristics create semantic mapping opportunities, general techniques, and useful qualities. Interaction techniques and scenarios encompassing many of these qualities are demonstrated in a following section.

Multitasking - *Stacked-Hinge* configurations can be used to monitor or switch applications (e.g. in *Stacked[N]*, the top shows the time, *Stacked-Hinge-E[N]* reveals weather app on the base, and a tap swaps weather to the top). Relative orientation is another way to multi-task (e.g. *Stacked[N]* shows time, *Stacked[S]* shows fitness app). The controlled study found that 180° top rotations are slower and less preferred, but this may be due to the type of grasp.

Extra Information - Fast and unstable *Peeks* are suited to temporarily viewing small bits of information like notifications or system status. They also increase privacy (e.g. generic notification icon on top, *Peek* to view content). *Corner Peeks and Side Peeks* can view extra information for the current app depending on the amount of information.

Slide Off for More - When a small bit of content is visible in *Peek*, the top can be slid off the base until it springs into an *Adjacent* configuration to view full information.

Sharing - When detached, the top can be momentarily given to someone else to play a game, or view content (e.g. a photo) (E6). A manipulation (e.g. pivoting) during the transition from *Detached-Near* to *Detached-Far* can indicate how much control the user has over the content (e.g. view current photo only, view the whole album). Manipulations on the base can also remotely control top content.

Modes - Application modes can be selected according to the *Adjacent* configuration (e.g. weather by city vs weather map) and *Adjacent Rotations* can refine the current mode further (show forecast or satellite map).

Tuning - Discretizing the hinging and/or pivoting rotation space can be used to change a viewing mode of an application from an *Adjacent* configuration (e.g. small hinge rotations to toggle a map between streets, satellite, and both).

Reduced Occlusion - In *Adjacent* or *Detached* configurations, touch input on the top can control an interface shown on the base eliminating occlusion (E2). For long term single-handed navigation, the top can be held in the watch hand and used like a thumb-activated touch pad to navigate content on the base (e.g. pan and zoom a map).

Silence and Reset - The *Closed* configuration can be used to silence notifications or if held temporarily, as a way to reset an application state (transition from *Adjacent* to *Closed*, then back to *Adjacent* to reset fitness activity tracking).

Private and Public - Hinge configurations have built in affordances to suggest public sharing (e.g. *Stacked-Hinge-N[N]*) or private viewing (*Closed-Hinge-N[N]*) (E7, E8).

DOPPIO DEVICE PROTOTYPE

As a full proof-of-concept, we built a high-fidelity hardware prototype with two high resolution touch displays able to detect all primary Doppio configurations, transitions, and manipulations. Here we provide device technical details, and demonstrate the interaction vocabulary in the following section. The prototype uses two Sony SWR50 smartwatches with factory wristbands removed. Each SWR50 is placed in a custom 3D printed housing (8.75 × 6.75 × 2 mm) to form the top and base (Figure 8). The prototype is larger and thicker than we envision for a real device, but it is effective for validating and demonstrating the concept.

The base housing has four embedded capacitive sensors linked via conductive tape to the top face frame and wired to an off-board Arduino. The top housing is wrapped in conductive tape and has four magnets embedded at the corners to snap into *Stacked* and *Adjacent* configurations. The pattern of triggered capacitive sensors enables detection of *Stacked* and *Adjacent* configurations.

Each SWR50 has a 1200 MHz processor with a 1.6" (320×280px) capacitive touchscreen and orientation and inertial motion sensors. The relative orientation of the top to the base for detecting *Hinge* and *Pivot* movements is determined using the built-in accelerometer and gyroscope of each SWR50. The proximity of the top to the base for *Detached* configurations is measured with the built-in magnetometer on the base SWR50 (similar to the method used by Abracadabra [16]). 3D position tracking is possible via a pair of magnetometers [12], but not currently implemented.

Each SWR50 runs custom Android Wear software that passes sensor state to a server via WiFi. Combined with the Arduino's capacitive state, the server determines the current configuration, transition, or manipulation and updates the application display on both SWR50s also through WiFi.

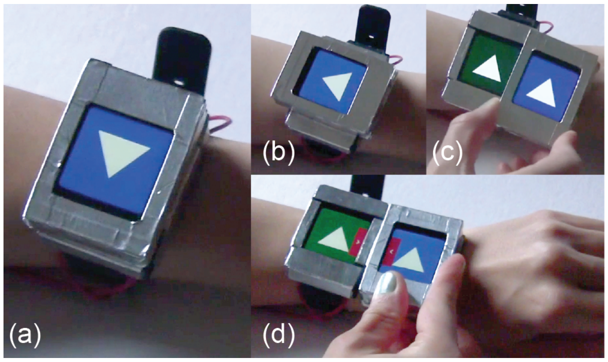


Figure 8. High Fidelity Prototype: (a) top stacked; (b) top rotated 90°; (c) top peek off base; (d) top attached to side of base.

INTERACTION DEMONSTRATIONS

We explore the characteristics and mappings in the Design Considerations section by applying them to specific system and application functions using the Doppio device prototype (see also the accompanying video).

Global Functions

Doppio interactions can be used for global functions like managing notifications, system status, and applications.

Managing Notifications - The transient nature of *Peeks* and the hidden base display in a *Stacked* configuration are combined to manage notifications. When a notification occurs, sliding the top down to *Stacked-Peek-S[N]* previews notification content on the base without altering the top display (Figure 9a). Sliding back to *Stacked* dismisses that notification, but sliding back while touching the top display saves it. While previewing a notification in a *Peek,* sliding off the base to an *Adjacent* configuration opens the associated application on both displays (Figure 9c). For more privacy, a notification preview can transition to a *Stacked-Hinge-N[N]* to shield the base display with top (Figure 9b).

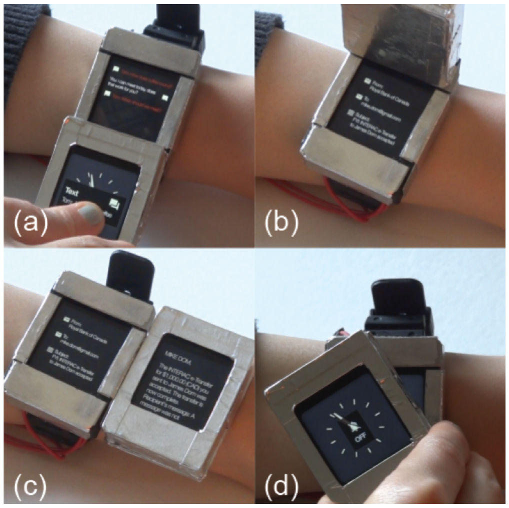


Figure 9. Notifications: (a) preview notification content; (b) viewing a notification more privately; (c) opening associated notification app using adjacent; (d) disabling notifications.

Managing System Information and Settings - Instead of using space on the top display for system information, *Corner Peeks* can easily reveal this information on the base display. For example, pivoting the top slightly counter-clockwise from *Stacked* to *Stacked-Peek-SW[N]* reveals the bottom-right corner of the base where the battery level can be shown. System settings could also be managed in this manner. For example, pivoting *Stacked* clockwise to *Stacked-Peek-SE[N]* could view the number of notifications in the bottom-left base corner, but if the same pivoting motion is done and the top allowed to immediately spring back this could silence notifications (Figure 9d).

Multitasking - Launching and switching between apps is challenging on a smartwatch. Transitioning to different *Stacked* and *Stacked-Hinge* configurations is a tangible way to switch between apps. For example, two foreground apps (like weather and fitness) can be mapped to *Stacked[N]* and *Stacked[S]* so that a half-turn of the top switches between them (Figure 10a-c). Alternatively, *Hinging* the top can access background apps like calendar on the base, and the background app can be swapped to the top by tapping the base display (Figure 10 d-f). One background app can be dedicated to full system information and settings, accessed with a less preferred *Hinge* like *Stacked-Hinge-S[N]*.

Launching Applications - Apps can be launched by turning the top one-quarter turn to *Stacked[W]* or *Stacked[E]* to view icons for two groups of frequently accessed apps (Figure 11a,b). Tapping an icon launches the app. Additional apps can be viewed and launched by sliding the top off the base to an *Adjacent-S* configuration (Figure 11c).

Applications

We designed and built four applications to show how different design characteristics and mappings can be applied.

Weather - The default view of the weather app shows the current temperature at the current location (Figure 12a). A top *Corner Peek* views *extra information* like the humidity. We reserve top *Corner Peeks* for app information and bottom corners for system information. Additional extra information is revealed with a *Stacked-Peek-E[N]* to show an hourly breakdown of the weather forecast. Sliding the top to *Adjacent-S[N]* changes the *mode* to a four-day forecast and *Adjacent-N[N]* changes the *mode* to a weather forecast of four cities (Figure 12b). In this configuration, discrete input is provided with a hinge manipulation *Adjacent-N[N,hinge]* to show more cities (Figure 12c). Rotating the top from *Adjacent-N[N]* to *Adjacent-N[E]* changes the sub *mode* to an hourly forecast for individual cities.

Map – The wide *shape* when *Adjacent* is ideal for viewing a map spanning both displays. Pivoting the top can zoom for *continuous input* while avoiding *occlusion*. Alternatively, panning or pinching touch input can be performed on one display leaving an occlusion free view of the map on the other (Figure 13a). In situations where a hand is unavailable (e.g. carrying a bag), the top can be used *one-handed* by the hand wearing the watch for indirect input (Figure 13b).

Photos - Handing the top to another person is an example of *sharing* in a photo app. The top can be detached from the

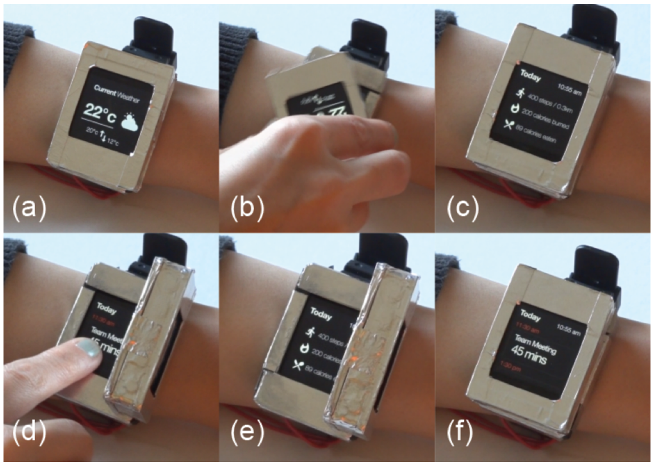


Figure 10. Multitasking: (a) foreground app; (b,c) turning to switch foreground app; (d) stacked hinge to view background apps; (e,f) swap background app to foreground with tap

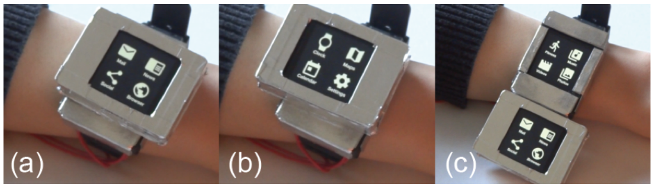


Figure 11. Launching apps: (a, b) frequent app icons in left and right groups; (c) viewing more app icons in group.

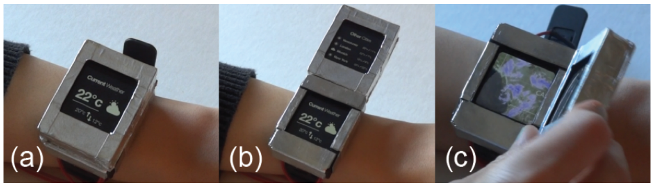


Figure 12. Weather app: (a) Stacked shows local conditions; (b) Adjacent shows city forecasts; (c) hinging alters map.

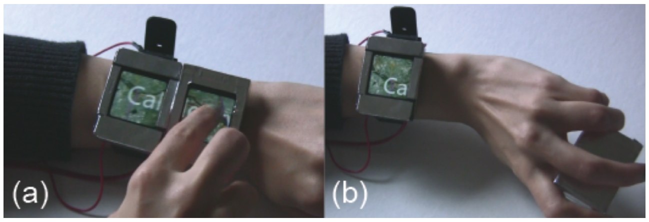


Figure 13. Map app: (a) Occlusion free map navigation on one Doppio screen; (b) One-handed interaction on Doppio.

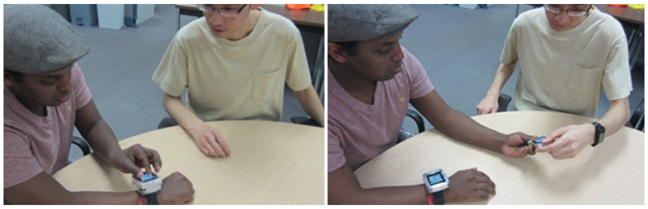


Figure 14. Photo app: sharing photos by sharing the top.

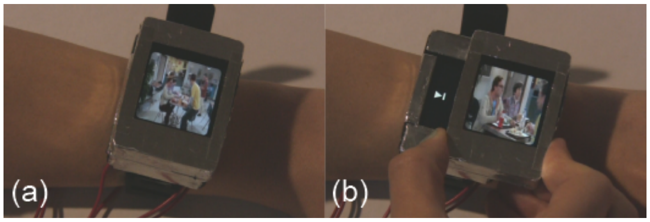


Figure 15. Media player app: (a) Pushing the top right to fast forward the video; (b) Pushing the top up/down for volume.

base and shared with others on two levels. The default is *Locked* where the top displays the current photo when the top was detached and disables all interaction (Figure 14a). Alternatively, if the top touch screen is held while detaching, the top is s*emi-locked* enabling the other person to interact, but only with the photo app (Figure 14b).

Media Player - Media player control can be accessed without a graphical user interface detracting from media display. Similar to Xiao et al. [32], video playback can be controlled by physically manipulating the top. For example, *Stacked-Peek-E[N]* and *Stacked-Peek-W[N]* to seek forward and backward (Figure 15) and Stacked-Peek-N[N] and Stacked-Peek-S[N] to increase or decrease volume.

LIMITATIONS AND FUTURE WORK

We discuss limitations of our work and suggest future research for exploring similar interaction spaces.

Device Size and Thickness - The size and thickness of Doppio can be further reduced with additional engineering efforts. The smartwatches in our current implementation can be replaced by mini touchscreen LCDs with custom designed PCBs and batteries. We believe that with today's technology, it is possible to create a dual-face smartwatch similar in size to a regular smartwatch. We also see alternative design options that can improve the Doppio form factor. For example, a better battery size strategy can be used. The base can have a larger battery than the top, as it needs power in smaller periods of time in the detached stage. The top can also be powered by the base when attached, allowing it to be made even smaller to reduce size and weight.

Battery Life - Batteries drain quickly in our current implementation due to frequent WiFi data communication between the two watch faces and because both displays are always on. Battery capacity can be increased, but power consumption can also be reduced. For example, switching to a low-power protocol like Bluetooth LE and automatically turning off the base when covered by the top.

Shape of the Watch - While we explore the notion of dual-screen smartwatches in a rectangular shape, a round watch face may provide a unique affordance for new types of interactions. For example, a round face allows users to perform a number of unique continuous input methods, such as pivoting or rotating the top along the curved edge of the base. Future research will investigate different shape options and interactions enabled by different shapes.

Alternative Designs - Doppio can further benefit from an additional screen on the back face of the top. This may introduce new interactions that are not possible using two screens. Future research should focus on exploring new interactions enabled by a double-sided top face, as well as methods to overcome the technical and engineering challenges introduced by an additional screen in a small watch form factor. Tethering the top using a string provides potential for new interactions, warranting further research.

Evaluation - Doppio warrants careful evaluation in the field. For example, understanding usability in real-world usage scenarios to identify issues not revealed in our controlled experiment and examining the benefits and limitations of sharing the top in different social settings.

CONCLUSION

Our work introduces the concept of a reconfigurable dual-face smartwatch, designed to address the limited interaction space in a small smartwatch form factor. Doppio's two display faces, including a detachable top, can form 124 different configurations with associated transitions and manipulations. Through an exploratory study, we investigated how users could practically utilize the Doppio concept. We also conducted a controlled experiment using an instrumented prototype to measure the transition time between the different configurations. The results of both studies and subjective preferences provided insight into a set of practical characteristics and design considerations to be applied to Doppio applications. Finally, to demonstrate the wide variety of novel Doppio interactions, we built a proof-of-concept hardware prototype, demonstrating how Doppio's interactions can be used for common watch scenarios and applications. As research in smartwatches and wearable devices continues to increase, we believe our work can inspire new ideas and designs for the future wearable devices.

REFERENCES

1. Halda Space Discovery Watch, http://www.ablogtowatch.com/halda-space-discovery-watch-review/

2. Jaeger LeCoultre Reverso Watches, http://www.jaeger-lecoultre.com/US/en/collections/reverso
3. Piaget Altiplano Double Jeu Watch, http://www.piaget.com/watches/rose-gold-ultra-thin-mechanical-watch-g0a35153
4. Porsche's P'6520 Heritage Compass Watch, http://www.ablogtowatch.com/porsche-design-p6520-heritage-compass-watch-hands-on/
5. Ritmo Mundo Persepolis, http://www.wristwatchreview.com/2012/04/05/review-ritmo-mundo-persepolis-part-1/
6. Titanium Two Face watch, http://shop.nordstrom.com/s/titanium-two-face-watch-20mm-juniors/3384115
7. Daniel Ashbrook, Kent Lyons, and Thad Starner. 2008. An investigation into round touchscreen wristwatch interaction. In *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services* (MobileHCI '08). ACM, New York, NY, USA, 311-314. DOI=10.1145/1409240.1409276 http://doi.acm.org/10.1145/1409240.1409276
8. Gabor Blasko and Steven Feiner. 2004. An Interaction System for Watch Computers Using Tactile Guidance and Bidirectional Segmented Strokes. In *Proceedings of the Eighth International Symposium on Wearable Computers* (ISWC '04). IEEE Computer Society, Washington, DC, USA, 120-123. DOI=10.1109/ISWC.2004.6 http://dx.doi.org/10.1109/ISWC.2004.6
9. Gabor Blasko, Chandra Narayanaswami, and Steven Feiner. 2006. Prototyping retractable string-based interaction techniques for dual-display mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '06), Rebecca Grinter, Thomas Rodden, Paul Aoki, Ed Cutrell, Robin Jeffries, and Gary Olson (Eds.). ACM, New York, NY, USA, 369-372. DOI=10.1145/1124772.1124827 http://doi.acm.org/10.1145/1124772.1124827
10. Jesse Burstyn, Paul Strohmeier, and Roel Vertegaal. 2015. DisplaySkin: Exploring Pose-Aware Displays on a Flexible Electrophoretic Wristband. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction* (TEI '15). ACM, New York, NY, USA, 165-172. DOI=10.1145/2677199.2680596 http://doi.acm.org/10.1145/2677199.2680596
11. Ron Van Buskirk and Mary LaLomia. 1995. A comparison of speech and mouse/keyboard GUI navigation. In *Conference Companion on Human Factors in Computing Systems* (CHI '95), I. Katz, R. Mack, and L. Marks (Eds.). ACM, New York, NY, USA, 96-. DOI=10.1145/223355.223447 http://doi.acm.org/10.1145/223355.223447
12. Ke-Yu Chen, Kent Lyons, Sean White, and Shwetak Patel. 2013. uTrack: 3D input using two magnetic sensors. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (UIST '13). ACM, New York, NY, USA, 237-244. DOI=10.1145/2501988.2502035 http://doi.acm.org/10.1145/2501988.2502035
13. Xiang 'Anthony' Chen, Tovi Grossman, and George Fitzmaurice. 2014. Swipeboard: a text entry technique for ultra-small interfaces that supports novice to expert transitions. In *Proceedings of the 27th annual ACM symposium on User interface software and technology* (UIST '14). ACM, New York, NY, USA, 615-620. DOI=10.1145/2642918.2647354 http://doi.acm.org/10.1145/2642918.2647354
14. Xiang 'Anthony' Chen, Tovi Grossman, Daniel J. Wigdor, and George Fitzmaurice. 2014. Duet: exploring joint interactions on a smart phone and a smart watch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 159-168. DOI=10.1145/2556288.2556955 http://doi.acm.org/10.1145/2556288.2556955
15. Antonio Gomes and Roel Vertegaal. 2015. PaperFold: Evaluating Shape Changes for Viewport Transformations in Foldable Thin-Film Display Devices. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction* (TEI '15). ACM, New York, NY, USA, 153-160. DOI=10.1145/2677199.2680572 http://doi.acm.org/10.1145/2677199.2680572
16. Chris Harrison and Scott E. Hudson. 2009. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology* (UIST '09). ACM, New York, NY, USA, 121-124. DOI=10.1145/1622176.1622199 http://doi.acm.org/10.1145/1622176.1622199
17. Ken Hinckley, Morgan Dixon, Raman Sarin, Francois Guimbretiere, and Ravin Balakrishnan. 2009. Codex: a dual screen tablet computer. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09). ACM, New York, NY, USA, 1933-1942. DOI=10.1145/1518701.1518996 http://doi.acm.org/10.1145/1518701.1518996
18. Da-Yuan Huang, Ming-Chang Tsai, Ying-Chao Tung, Min-Lun Tsai, Yen-Ting Yeh, Liwei Chan, Yi-Ping Hung, and Mike Y. Chen. 2014. TouchSense: expanding touchscreen input vocabulary using different areas of users' finger pads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 189-192. DOI=10.1145/2556288.2557258 http://doi.acm.org/10.1145/2556288.2557258
19. Jungsoo Kim, Jiasheng He, Kent Lyons, and Thad Starner. 2007. The Gesture Watch: A Wireless Contact-free Gesture based Wrist Interface. In *Proceedings of the 2007 11th IEEE International Symposium on Wearable Computers* (ISWC '07). IEEE Computer Society, Washington, DC, USA, 1-8.

DOI=10.1109/ISWC.2007.4373770 http://dx.doi.org/10.1109/ISWC.2007.4373770

20. Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. 2014. Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors. In*Proceedings of the 27th annual ACM symposium on User interface software and technology*(UIST '14). ACM, New York, NY, USA, 389-394. DOI=10.1145/2642918.2647356 http://doi.acm.org/10.1145/2642918.2647356
21. Kent Lyons, David Nguyen, Daniel Ashbrook, and Sean White. 2012. Facet: a multi-segment wrist worn system. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (UIST '12). ACM, New York, NY, USA, 123-130. DOI=10.1145/2380116.2380134 http://doi.acm.org/10.1145/2380116.2380134
22. David Merrill, Jeevan Kalanithi, and Pattie Maes. 2007. Siftables: towards sensor network user interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction* (TEI '07). ACM, New York, NY, USA, 75-78. DOI=10.1145/1226969.1226984 http://doi.acm.org/10.1145/1226969.1226984
23. Ian Oakley and Doyoung Lee. 2014. Interaction on the edge: offset sensing for small devices. In*Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 169-178. DOI=10.1145/2556288.2557138 http://doi.acm.org/10.1145/2556288.2557138
24. Simon Olberding, Kian Peen Yeo, Suranga Nanayakkara, and Jurgen Steimle. 2013. AugmentedForearm: exploring the design space of a display-enhanced forearm. In *Proceedings of the 4th Augmented Human International Conference* (AH '13). ACM, New York, NY, USA, 9-12. DOI=10.1145/2459236.2459239 http://doi.acm.org/10.1145/2459236.2459239
25. Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. 2013. ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 2799-2802. DOI=10.1145/2470654.2481387 http://doi.acm.org/10.1145/2470654.2481387
26. Jerome Pasquero, Scott J. Stobbe, and Noel Stonehouse. 2011. A haptic wristwatch for eyes-free interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*(CHI '11). ACM, New York, NY, USA, 3257-3266. DOI=10.1145/1978942.1979425 http://doi.acm.org/10.1145/1978942.1979425
27. Simon T. Perrault, Eric Lecolinet, James Eagan, and Yves Guiard. 2013. Watchit: simple gestures and eyes-free interaction for wristwatches and bracelets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 1451-1460. DOI=10.1145/2470654.2466192 http://doi.acm.org/10.1145/2470654.2466192
28. Norman Pohl, Steve Hodges, John Helmes, Nicolas Villar, and Tim Paek. 2013. An interactive belt-worn badge with a retractable string-based input mechanism. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 1465-1468. DOI=10.1145/2470654.2466194 http://doi.acm.org/10.1145/2470654.2466194
29. Raf Ramakers, Johannes Schöning, and Kris Luyten. 2014. Paddle: highly deformable mobile devices with physical controls. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '14). ACM, New York, NY, USA, 191-192. DOI=10.1145/2559206.2579524 http://doi.acm.org/10.1145/2559206.2579524
30. Jun Rekimoto. 2001. GestureWrist and GesturePad: Unobtrusive Wearable Interaction Devices. In*Proceedings of the 5th IEEE International Symposium on Wearable Computers* (ISWC '01). IEEE Computer Society, Washington, DC, USA, 21-.
31. Katie A. Siek, Yvonne Rogers, and Kay H. Connelly. 2005. Fat finger worries: how older and younger users physically interact with PDAs. In *Proceedings of the 2005 IFIP TC13 international conference on Human-Computer Interaction* (INTERACT'05), Maria Francesca Costabile and Fabio Paternò (Eds.). Springer-Verlag, Berlin, Heidelberg, 267-280. DOI=10.1007/11555261_24 http://dx.doi.org/10.1007/11555261_24
32. Robert Xiao, Gierad Laput, and Chris Harrison. 2014. Expanding the input expressivity of smartwatches with mechanical pan, twist, tilt and click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 193-196. DOI=10.1145/2556288.2557017 http://doi.acm.org/10.1145/2556288.2557017
33. Cheng Xu and Kent Lyons. 2015. Shimmering Smartwatches: Exploring the Smartwatch Design Space. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction* (TEI '15). ACM, New York, NY, USA, 69-76. DOI=10.1145/2677199.2680599 http://doi.acm.org/10.1145/2677199.2680599