MultiBezel: Adding Multi-Touch to a Smartwatch Bezel to Control Music

Marvin Reuter
Technische Hochschule Köln
Cologne, Germany
marvin.reuter@th-koeln.de

Ali Ünal
Technische Hochschule Köln
Cologne, Germany
ali_ihsan.uenal@smail.th-koeln.de

Jan Felipe Ribeiro Kolodziejski
Technische Hochschule Köln
Cologne, Germany
jan_felipe.kolodziejski_ribeiro@smail.th-koeln.de

David Petersen
Technische Hochschule Köln
Cologne, Germany
david.petersen@th-koeln.de

Matthias Böhmer
Technische Hochschule Köln
Cologne, Germany
matthias.boehmer@th-koeln.de

ABSTRACT

Limited screen space and reliance on touch input pose challenges for intuitive interaction on smartwatches, often leading to screen occlusion and hindered usability. To overcome these limitations, this paper presents MultiBezel, a novel multi-touch enabled bezel designed to enhance interaction on smartwatches. MultiBezel utilises a gesture-based interaction scheme, mapping distinct finger combinations to control the smartwatch. This approach shifts interaction away from the touchscreen enabling new ways of eyes-free control. We developed a functional prototype that recognises up to three simultaneous touch points. This prototype demonstrates the feasibility of multi-touch bezel interaction for enhancing the smartwatch user experience. We contribute the hardware and software of our prototype and implemented an application for music control.

KEYWORDS

mobile, smartwatch, eyes-free interaction, multi-touch, gesture control

ACM Reference Format:

Marvin Reuter, Ali Ünal, Jan Felipe Ribeiro Kolodziejski, David Petersen, and Matthias Böhmer. 2025. MultiBezel: Adding Multi-Touch to a Smartwatch Bezel to Control Music. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25), April 26-May 1, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3706599.3720156

1 INTRODUCTION

Smartwatches, despite their potential for ubiquitous access, present interaction challenges due to their limited screen size and reliance on touch input, causing screen occlusion, hindering visibility and requiring visual attention in mobile, hands-busy or socially sensitive scenarios [4, 6, 11]. Prior research has explored bezel interactions like bezel-initiated swipes (BIS) [6, 11] and bezel-to-bezel (B2B) gestures [5, 8] for eyes-free control, however, these techniques are typically limited to single-touch input.

This paper presents MultiBezel, a novel multi-touch capable bezel for smartwatches designed to enhance music control and mitigate screen occlusion issues. Our approach integrates multi-touch technology into the bezel, with the goal to enable intuitive, gesture-based control without the need for direct visual focus on the device. The significance of this research lies in its potential to redefine how users interact with wearable devices. By shifting interaction away from the small and often occluded touchscreen to the bezel. This is particularly relevant in scenarios where visual focus is limited or impractical, such as during exercise, cooking, or commuting.

We provide insights into the benefits and challenges of multi-touch technology in wearable devices. The development process, the results achieved, and the potential for future advancements in this project are discussed in detail. MultiBezel improves smartwatch interaction by mitigating screen occlusion and enabling more intuitive, eyes-free control through a multi-touch enabled bezel, thus representing a significant contribution to wearable device design and usability.

2 RELATED WORK

This research builds upon a burgeoning field exploring innovative input methods and eyes-free interaction techniques for smartwatches, specifically focusing on bezel interaction and eyes-free techniques. However, existing work primarily explores single-touch bezel interactions or focuses on general input tasks. This research addresses a key gap by introducing multi-touch capabilities to the bezel, enabling a richer gesture set tailored specifically for intuitive and efficient music control.

Screen Occlusion Mitigation. Screen occlusion, a significant usability issue on small touchscreens, has been a focus of research in smartwatch interaction. Neshati et al. [6] investigated the use of bezel interaction for mitigating occlusion when exploring data visualizations on smartwatches. They found that Partial BezelGlide (PBG), which limits interaction to the bezel region with maximum screen visibility, significantly reduces occlusion compared to full bezel interaction and touchscreen-based techniques.

CHI EA '25, April 26-May 1, 2025, Yokohama, Japan

ACM ISBN 979-8-4007-1395-8/2025/04
https://doi.org/10.1145/3706599.3720156

Bezel Interaction Techniques. Previous research has demonstrated the potential of utilizing the smartwatch bezel for interaction. Bezel-initiated swipe (BIS) gestures, where a swipe originates from the bezel and moves inwards towards the screen, have been explored for command invocation and text entry on both square and circular smartwatches [1, 6]. Wong et al. [11] investigated the performance of eyes-free BIS on circular smartwatches, finding that accuracy is highly dependent on bezel segment size and user-specific variations. Building on the concept of BIS, Kubo et al. [5] investigated bezel-to-bezel (B2B) gestures, where swipes initiate and end on different bezel segments, enabling a larger set of commands for eyes-free interaction. Rey et al. [8] further explored the feasibility of B2B gestures on circular smartwatches in mobile and encumbered scenarios, revealing that while accuracy decreases under these conditions, B2B remains viable with carefully chosen segment layouts. Our work takes inspiration from these bezel-based approaches, extending them by introducing multi-touch capabilities to the bezel. Enabling richer interaction possibilities beyond single-touch swipes.

Eyes-Free Interaction. The need for eyes-free smartwatch interaction has been recognized in a variety of contexts, including mobile, hands-busy, and socially sensitive scenarios [4, 7]. Various techniques have been proposed, including wrist gestures [3], non-voice acoustic input [9] and gaze input [2] for hands-free interactions. Wong et al. [11] demonstrated that BIS gestures can be performed effectively in eyes-free conditions with appropriate bezel segmentation and personalized machine learning models. Rey et al. [8] further validated the feasibility of eyes-free B2B gestures, even in mobile and stressful situations.
MultiBezel builds on this research by enabling eyes-free interaction with a multi-touch bezel, allowing users to control music playback without needing to look at the watch face. This is particularly beneficial for hands-busy situations, such as exercising or cooking, where conventional touch interaction is challenging.

Music Control Interfaces. Roth et al. [10] presented TempoWatch, a smartwatch-based music control interface designed specifically for dance instructors. It employs a circular touch interface for controlling tempo and secondary interfaces for volume and track selection. TempoWatch highlights the value of designing specialized interfaces tailored to specific user needs and contexts.
While our focus is on a multi-touch bezel, MultiBezel shares a similar goal of providing a more efficient and convenient way to control music on smartwatches. By exploring different multi-finger gestures and enabling eyes-free interaction, we aim to enhance the music player experience for a broader range of users and situations.
This work on MultiBezel is uniquely positioned within the landscape of smartwatch interaction research. It draws inspiration from previous work on bezel-based interaction, screen occlusion mitigation, and eyes-free techniques, while extending these approaches by introducing multi-touch capabilities to the bezel for more expressive and efficient smartwatch control. By exploring this novel interaction method, we provide a valuable contribution to understanding how multi-touch bezels can enhance the smartwatch user experience, particularly for tasks like music control in eyes-free contexts.

3 CONCEPT AND PROTOTYPE

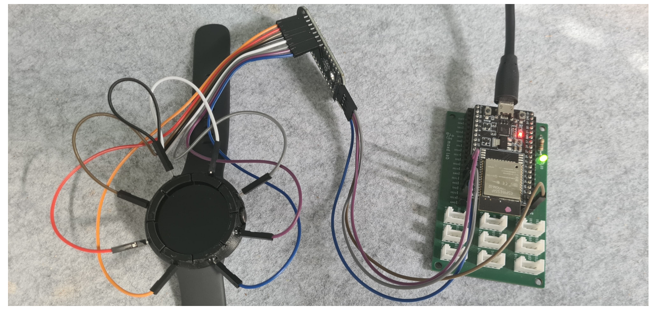

Figure 1: MulitBezel prototype with smartwatch, 3D-printed capacitive sensor technology and ESP32 microcontroller.

To explore the potential of a multi-touch capable smartwatch bezel for eyes-free music control, we developed a functional prototype called "MultiBezel" (fig. 1). This prototype allows users to perform various music playback actions using intuitive multi-finger gestures on the bezel, without needing to look at the watch face. This section describes the hardware and software components of the MultiBezel prototype, detailing its design and functionality.

3.1 Concept

The core concept of this project is to integrate multi-touch functionality into the bezel of a smartwatch. The goal is to move beyond the limitations of single-touch bezel interactions, typically restricted to basic swipe gestures, and explore the potential of more complex and expressive multi-finger inputs. By expanding the interaction space beyond the small display, MultiBezel aims to create a more intuitive and efficient user experience.

3.2 Hardware

Our MultiBezel prototype consists of three primary components: a custom-designed 3D printed bezel, a TrillCraft sensor[1] and an ESP32 microcontroller[2]. Figure 2 illustrates the system architecture. The 3D printed bezel, produced using conductive filament, features eight distinct channels separated by non-conductive gaps. Each channel acts as a capacitive sensor, providing raw capacitance readings for a specific region of the bezel. When a finger touches a conductive region, the corresponding capacitance value increases. We opted for eight channels as this configuration provided a suitable balance between touch detection accuracy (supporting up to three simultaneous fingers) and manageable wiring complexity.
Furthermore, the prototype is designed to interpret the simultaneous activation of two adjacent channels as a single touch event. This decision ensures that when a finger contacts the bezel across two neighbouring channels, it is correctly registered as one touch, rather than two.
The TrillCraft interfaces with the eight conductive channels of the 3D printed bezel via jumper wires, collecting raw capacitance readings from each channel. The ESP32 microcontroller processes

[1]https://learn.bela.io/tutorials/trill-sensors/working-with-trill-craft/
[2]https://www.espressif.com/en/products/socs/esp32

the data received from the TrillCraft, calculating the number of touch points and their positions.

To transmit touch data to the smartwatch, the ESP32 establishes a local wireless network using an access point, with the smartwatch connecting to this network. Data is sent over this wireless link using the UDP protocol.

While this approach temporarily disables the smartwatch's internet connectivity, this limitation is acceptable for our prototype, as our focus is on evaluating the user experience of the multi-touch bezel for music control.

Using this approach, our prototype simulates a digital bezel with multitouch recognition.

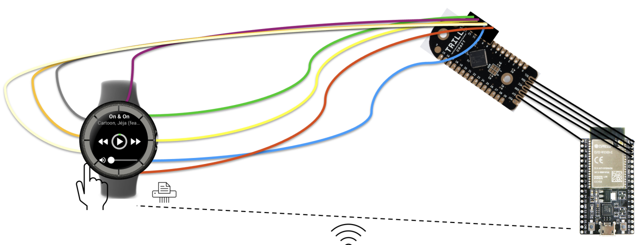

Figure 2: Representation of the system architecture

3.3 Software

3.3.1 Touch detection. A touch detection algorithm, implemented on an ESP32 microcontroller, determines precise finger positions. The algorithm compares capacitive sensor values against a predefined threshold. The sensing area is divided into eight segments, referred to as pads, numbered 0 through 7 in a clockwise direction starting at the top (see Fig. 3).

A touch is registered when a pad's capacitive value exceeds the threshold. Touches between adjacent pads are also detected. If two neighboring pads simultaneously register values above the threshold, the algorithm calculates a weighted average of their values to determine the precise touch position. This allows for position resolution finer than the individual pads (e.g. a touch between pad 1 and pad 2 might result in a calculated position of 1.4).

This calculated position is then multiplied by 100 and transmitted, resulting in a position value ranging from 0 to 799. The transition from pad 7 to pad 0 results in a jump from position 799 to 0. The system continuously calculates and transmits these position values. This continuous position data allows for tracking the angular delta between current and previous touch positions. By accumulating these angular deltas, changes in a single touch's position can be accurately measured and utilized for gesture recognition on the MultiBezel.

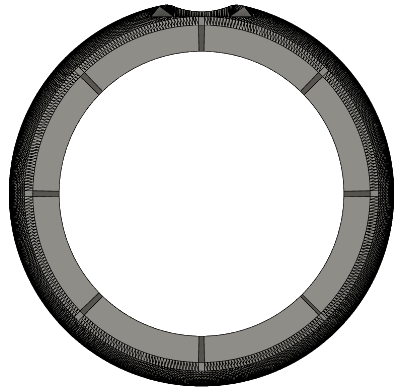

Figure 3: MultiBezel front view. Light grey indicates the capacitive areas, referred to as pads.

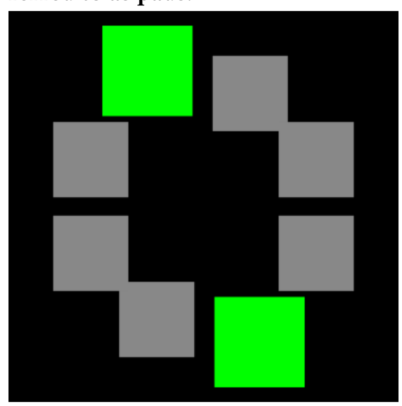

Figure 4: Raw data view

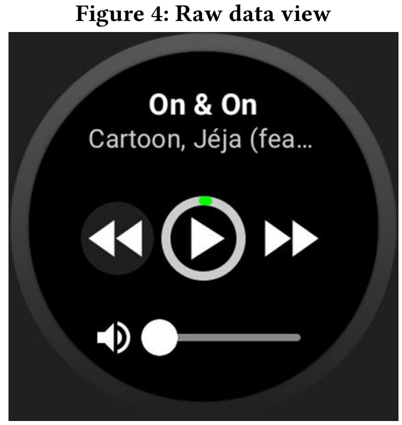

Figure 5: Music Player User Interface

3.3.2 Displaying raw data. To facilitate testing and verification of individual channel functionality, a dedicated raw data view was integrated into the application. This view comprises eight squares arranged circularly, as illustrated in Figure 4, with each square representing a corresponding pad on the MultiBezel. When a pad is touched, its associated square changes color, providing a visual indication of touch detection. This visual feedback mechanism serves two primary purposes:

- Channel Validation: It enables rapid identification of any malfunctioning channels that fail to register touch or transmit signals correctly.
- Touch Response Assessment: It allows for qualitative observation of the latency between a touch event and the corresponding visual feedback, providing insight into the system's responsiveness.

The raw data view also supports the visualization of simultaneous multi-touch interactions and finger movements across channels, further aiding in the diagnosis of potential issues and optimizing the touch recognition algorithm. By visualizing these interactions, the development process was significantly streamlined, allowing for faster identification and resolution of faulty channels. Future iterations of the raw data view could also incorporate logging capabilities to record precise touch timestamps and durations for quantitative analysis.

4 APPLICATION FOR MUSIC CONTROL

This work focuses on a custom-developed music player application, chosen as a tangible platform to showcase the capabilities of MultiBezel. The experience gained in developing this application will provide valuable insights for implementing MultiBezel's functionality within a broader range of applications in the future. The music player was designed to incorporate three core functions:

- Volume Control
- Track Navigation (Skipping Songs)
- Playback Control (Pausing and Resuming)

The goal is to allow users to seamlessly control these functions through intuitive, multi-touch gestures on the bezel – without needing to look at the smartwatch screen. The music player application is a fully custom-built solution designed specifically to leverage the multi-touch capabilities of the bezel. It is developed for the Android Wear operating system and utilizes the Android MediaPlayer framework[3] as its foundation for audio playback and control. This provides a robust and reliable platform for managing audio streams and interacting with the device's audio hardware.
The custom implementation allows for tight integration with the bezel touch events and enables precise control over the gesture recognition and mapping processes.
The music player provides a streamlined interface for controlling audio playback directly from the smartwatch bezel. The user interface reflects a simple music player design, which is not required for interaction but available for reference (see Fig. 5).
Haptic feedback is provided after each action performed. This means that the user is directly informed when an action has been performed, which aims to help when using the prototype eyes-free.
The following section details the individual music player functionalities and their corresponding multi-touch bezel gestures. Figure 6 illustrates the gesture set. We designed these mappings as basic gestures, which are intended to serve as simple and potentially intuitive foundations for user interaction.

[3]https://developer.android.com/media/platform/mediaplayer

(a) Control the music player's volume

(b) Skip through songs

(c) Pause/Resume songs

Figure 6: Overview of the Prototype Functions: The numbers indicate the chronological order of the steps. The red dots represent the approximate locations where the fingers should be placed.

Controlling the volume. Volume adjustments are achieved through one-finger rotational gestures on the MultiBezel (see Fig. 6a). A clockwise rotation increases the volume, while a counter-clockwise rotation decreases it. Each 22.5-degree rotation results in a 5% change in volume. If the finger is lifted from the bezel before a 22.5-degree rotation is completed, the volume adjustment action is cancelled. Heo et al. [4] have already pointed out that delimiters should be used to minimise accidental touches. By using these necessary thresholds to change the volume, a delimiter is added, which should minimise the risk of accidental actions.

Skipping through songs. Navigating through the music library is facilitated by two-finger rotational gestures on the MultiBezel. To skip to the next song, users perform a clockwise rotation of at least 45 degrees. To skip to the previous song, users perform a counter-clockwise rotation of at least 45 degrees.

Pausing and Resuming a song. Users can play and pause music using a discrete three-finger tap on the bezel (see Fig. 6c). The three-finger tap was chosen as the triggering gesture for this function to minimize accidental activation. Requiring three fingers to be used simultaneously should significantly reduce the likelihood of unintentional triggering, ensuring that playback control remains deliberate and reliable.

5 DISCUSSION

5.1 Results

While a formal user study is planned for future work, our prototype development and initial exploration yielded several key findings.
We successfully implemented a functional prototype demonstrating the feasibility of using a multi-touch bezel for music control. The bezel accurately recognizes distinct touch events from up to three fingers, enabling the execution of a range of gestures mapped to core music player functions (play/pause, volume change, track navigation). This provides a proof-of-concept for a novel interaction method that directly addresses limitations of conventional touchscreen-based controls.
Although subjective user experience will be formally evaluated in future studies, we hope that the use of multi-touch gestures will enable intuitive control mapping that utilises natural hand movements, potentially streamlining interaction compared to sequential button presses or scrolling on a small touchscreen. However, an important consideration is whether users will find multi-touch gestures comfortable and practical in real-world scenarios. While multi-touch capabilities expand the possible interaction space, further investigation is needed to determine how many gestures are genuinely useful and whether single-touch gestures could perform just as effectively in certain contexts.
Unlike prior work focusing on single-touch bezel swipes, MultiBezel introduces multi-touch sensing. With the goal to significantly expand the interaction vocabulary and design space for bezel-based input. While some research has explored bezels for general input, MultiBezel focuses specifically on music control, as a first way to test the plausibility of implementing it on smart watches. MultiBezel directly addresses the issue of screen occlusion on smartwatches, by moving the interaction to the bezel, which has primarily been tackled through touchscreen-based solutions. By moving interaction entirely to the bezel, MultiBezel tries to offer a more effective and elegant solution for enabling eyes-free interactions.

5.2 Optimization of Hardware

While our MultiBezel prototype demonstrates the potential of multi-touch bezels to enhance smartwatch interaction, it is important to acknowledge its current limitations. The wiring of the current prototype is not fully isolated, resulting in occasional noise interference when touched. This highlights the need for a more sophisticated hardware design with proper shielding and isolation in future iterations. The MultiBezel is intended to be incorporated into the smartwatch without the need for wireless data transmission once the sensor is integrated into the device's circuit board, which could help to solve the problem. For this reason, integration into the watch itself is a high priority for future iterations of the MultiBezel. In addition, the current sensor configuration of the prototype can only detect up to three touch points simultaneously. Expanding the capacity spaces to support a greater number of fingers could enable even richer and more expressive gesture sets. However, one question remains: how many such gestures are actually necessary and practical? It is crucial to evaluate whether users will actually benefit from an expanded gesture set or whether a more limited set of intuitive, well-defined interactions would be preferable. Furthermore, future research should investigate whether single-touch gestures could offer comparable functionality while reducing complexity and minimizing cognitive load.
The current prototype does not yet include a delimiter gesture to distinguish intended interactions from accidental touches. This can lead to unintentional activation of music player functions during everyday movements. While a delimiter may not be as critical for continuous inputs, which are designed to minimise accidental triggering, a delimiter mechanism would be particularly beneficial for discrete inputs such as play/pause, where a single touch can trigger an action. The implementation of a subtle and reliable demarcation mechanism, such as a specific bezel pressure pattern or a combination gesture, would be crucial to improving usability in these cases.
The effectiveness of such a delimiter could be tested in everyday life by testing for false positives while the user is wearing a smartwatch with the MultiBezel integrated. Moreover, the interaction between multi-touch gestures and single-touch inputs must be carefully considered to avoid unintended conflicts. Potential interference with touchscreen-based interactions should also be examined to ensure a seamless and efficient user experience.

5.3 Other Applications

The potential of a multi-touch bezel extends beyond music player control. Future research will investigate the applicability of MultiBezel to other smartwatch applications, such as:

- **Call Management:** Intuitive gestures could be employed for answering, rejecting, or muting calls without requiring visual attention to the smartwatch screen.
- **Navigation:** The bezel could enable zooming, panning, and rotating maps with multi-finger gestures, providing a more convenient and immersive navigation experience.
- **Smart Home Control:** MultiBezel could facilitate interaction with smart home devices, allowing users to control lights, thermostats, or appliances directly from their wrist using simple gestures.

With further optimization, the MultiBezel could evolve beyond a dedicated music controller, serving as a versatile multi-touch input surface for a wide range of smartwatch functionalities. This could potentially extend the smartwatch's capabilities to control and interact with other devices in a user's ecosystem, creating a more seamless and integrated user experience.

6 CONCLUSION

This work introduced MultiBezel, a novel approach to smartwatch interaction that leverages a multi-touch capable bezel for enhanced music control, addressing the limitations of small touchscreens and aiming for eyes-free, intuitive interaction.

Our functional MultiBezel prototype implements multi-touch sensing on a smartwatch bezel, enabling multi-touch gestures for music player control.

We contribute the source code of all components and the hardware design as open source.[4] While further study is needed, this research indicates that multi-touch bezel interaction has significant potential to improve smartwatch usability and unlock new possibilities for user-centered wearable technology.

ACKNOWLEDGMENTS

This work has been partly funded by the European Regional Development Fund under grant EFRE-20500002.

[4]Microcontroller Github repository: https://github.com/moxdlab/MultiBezel_firmware, Android smartwatch Github repository: https://github.com/moxdlab/MultiBezel

REFERENCES

[1] Sunggeun Ahn, Jaeyeon Lee, Keunwoo Park, and Geehyuk Lee. 2018. Evaluation of edge-based interaction on a square smartwatch. *International Journal of Human-Computer Studies* 109 (2018), 68–78. https://doi.org/10.1016/j.ijhcs.2017.08.004

[2] Andrés Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. ACM, New York, NY, USA, 457–466. https://doi.org/10.1145/2807442.2807463

[3] Jun Gong, Xing-Dong Yang, and Pourang Irani. 2016. WristWhirl: One-handed Continuous Smartwatch Input Using Wrist Gestures. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. ACM, New York, NY, USA, 861–872. https://doi.org/10.1145/2984511.2984563

[4] Seongkook Heo, Michelle Annett, Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. 2020. No Need to Stop What You're Doing: Exploring No-Handed Smartwatch Interaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. ACM, New York, NY, USA, 1–11. https://doi.org/10.1145/3313831.3376488

[5] Yuki Kubo, Buntarou Shizuki, and Jiro Tanaka. 2016. B2B-Swipe: Swipe Gesture for Rectangular Smartwatches from a Bezel to a Bezel. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 3852–3856. https://doi.org/10.1145/2858036.2858216

[6] Ali Neshati, Bradley Rey, Ahmed Sharif Mohommed Faleel, Sandra Bardot, Celine Latulipe, and Pourang Irani. 2021. BezelGlide: Interacting with Graphs on Smartwatches with Minimal Screen Occlusion. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. ACM, New York, NY, USA, Article 501, 13 pages. https://doi.org/10.1145/3411764.3445201

[7] 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* (Toronto, Ontario, Canada) *(CHI '14)*. ACM, New York, NY, USA, 169–178. https://doi.org/10.1145/2556288.2557138

[8] Bradley Rey, Kening Zhu, Simon T. Perrault, Sandra Bardot, Ali Neshati, and Pourang Irani. 2022. Understanding and Adapting Bezel-to-Bezel Interactions for Circular Smartwatches in Mobile and Encumbered Scenarios. In *Proceedings of the ACM on Human-Computer Interaction (MHCI, Vol. 6)*. ACM, New York, NY, USA, Article 201, 28 pages. https://doi.org/10.1145/3546736

[9] Gierad Reyes, Daniel Zhang, Sidhant Ghosh, Parth Shah, Jerry Wu, Ashutosh Parnami, Bryan Bercik, Thad Starner, Gregory D. Abowd, and W. Keith Edwards. 2016. Whoosh: Non-voice Acoustics for Low-cost, Hands-free, and Rapid Input on Smartwatches. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers* (Heidelberg, Germany) *(ISWC '16)*. ACM, New York, NY, USA, 120–127. https://doi.org/10.1145/2971763.2971775

[10] Josef Roth, Jan Ehlers, Christopher Getschmann, and Florian Echtler. 2021. TempoWatch: a Wearable Music Control Interface for Dance Instructors. In *Fifteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Salzburg, Austria) *(TEI '21)*. ACM, New York, NY, USA, 1–6. https://doi.org/10.1145/3430524.3442461

[11] Pui Chung Wong, Kening Zhu, Xing-Dong Yang, and Hongbo Fu. 2020. Exploring Eyes-free Bezel-initiated Swipe on Round Smartwatches. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. ACM, New York, NY, USA, 1–11. https://doi.org/10.1145/3313831.3376393