WristFlick: Design and Evaluation of a Smartwatch-Based System for Interacting with Smart Televisions

Yuan Ren
Inclusive Interaction Lab
University of California, Merced
Merced, California, USA
yren5@ucmerced.edu

Ahmed Sabbir Arif
Inclusive Interaction Lab
University of California, Merced
Merced, California, USA
asarif@ucmerced.edu

(a) Media Control & Navigation **(b) Focused Content Exploration** **(c) Searching by Writing on the Screen**

Figure 1: WristFlick is a smartwatch-based system designed for interacting with smart televisions. It allows users to browse channels, manage media playback, and view extra content like cast and soundtrack information using simple taps and flicks on the smartwatch. Users can also write directly on the smartwatch screen to search for specific channels or titles.

Abstract

WristFlick is a smartwatch-based system designed to interact with smart televisions. It allows users to navigate channels, control media, and access additional content, such as cast and soundtrack details, using taps and flicks on the smartwatch display. It also supports text input to search for specific channels or titles by writing directly on the smartwatch screen. Its design was refined through multiple pilot studies and evaluated in a three-session user study. The results revealed that WristFlick is significantly faster, requires fewer actions, and leads to fewer errors compared to a traditional remote control. Furthermore, participants preferred WristFlick over remote control and experienced a greater sense of flow during usage. In search tasks, WristFlick achieved comparable speeds with significantly fewer actions. Participants also demonstrated improved performance over time, with faster input speeds in later blocks. These findings suggest that WristFlick is an effective alternative to traditional remote controls for operating and controlling media on smart televisions.

IMX '25, Niterói, Brazil

ACM ISBN 979-8-4007-1391-0/25/06
https://doi.org/10.1145/3706370.3727864

CCS Concepts

• **Human-centered computing** → **Empirical studies in interaction design**; **Pointing**; *Gestural input*; **Text input**; **Empirical studies in HCI**; • **Information systems** → *Multimedia information systems.*

Keywords

TV, Large Display, Multimedia, Video, Media Control, Text Entry, Media Search, Television, Media Navigation, Remote Control, X-Ray

ACM Reference Format:
Yuan Ren and Ahmed Sabbir Arif. 2025. WristFlick: Design and Evaluation of a Smartwatch-Based System for Interacting with Smart Televisions. In *ACM International Conference on Interactive Media Experiences (IMX '25), June 03–06, 2025, Niterói, Brazil.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3706370.3727864

1 Introduction

Interactive smart televisions, commonly known as smart TVs or connected TVs, are gaining widespread popularity. These devices combine the functionality of traditional televisions with the capabilities of computers, enabling users to connect to the Internet through broadband or Wi-Fi [33, 34]. This connectivity grants access to digital content through platforms such as Roku and Amazon Fire TV and supports streaming through over-the-top (OTT) services such as Netflix and Hulu. A recent survey highlighted a significant increase in the adoption of smart TV in the United States and Europe, with penetration rates reaching nearly 75% in some regions [34].

As televisions become more advanced, so do the remote controls designed to operate them. Modern remotes are equipped with innovative features, such as voice commands and gesture-recognizing trackpads [40]. However, despite these advances, traditional button-based remotes continue to be the most widely used method for interacting with televisions [21, 27, 54].

Wearable technology, particularly smartwatches, offers a promising alternative for interacting with smart TVs. With touch-sensitive displays and portability, smartwatches have the potential to provide a more intuitive, efficient, and pleasant user experience compared to traditional remotes. Furthermore, being a personal device, smartwatches can enable multiple members of the household to interact with TV independently, eliminating the need for sharing a single remote. Unlike speech- or gesture-based interactions, smartwatch-based interactions do not rely on recognition algorithms, making them more reliable. In addition, they do not require expressive hand or wrist gestures, reducing the likelihood of user fatigue (§2).

This paper introduces WristFlick, a smartwatch-based system offering an alternative way to interact with smart televisions. WristFlick enables users to perform both basic and advanced tasks, such as navigating channels, controlling media playback, and accessing additional content such as cast details, using intuitive taps and flicks on the smartwatch. It also supports text input, allowing users to write directly on the smartwatch screen, simplifying search tasks compared to conventional remote control methods.

The design of WristFlick was informed by extensive lab trials and pilot studies, and its effectiveness was evaluated in a comparative study against a remote control. The study examined its impact on user experience, task efficiency, and interaction flow with a smart TV. Building on the fundamentals of wearable technology, WristFlick aims to provide a more streamlined, intuitive, and enjoyable interaction method, reducing the dependence on cumbersome remotes and improving the overall viewing experience of television.

2 Related Work

This section reviews the most relevant work for this project, including interactive systems for television control, text entry methods, and platform-specific media control options. Studies on traditional TV interactions are excluded, as the functionalities and role of televisions have evolved significantly with the emergence of interactive televisions [46]. Comprehensive reviews of traditional TV interaction research are available in the literature [14, 61]. Research has also investigated virtual televisions in augmented and virtual reality [39, 64], which are outside the scope of this work.

2.1 Tablet & Smartphone-based Methods

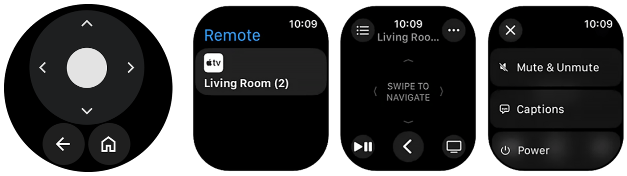
Cox et al. [13] investigated tablet-based control methods for interactive TVs, testing two variants of an open-source application. One variant transformed the tablet into a trackpad with virtual left and right mouse buttons, while the other mirrored the television display to enable absolute coordinate selection. These methods were compared with mid-air gestures and a Wiimote in a simple drag-and-drop task. The mirrored selection method demonstrated superior precision, while text entry using an on-screen keyboard favored the Wiimote in terms of speed and accuracy. However, the study's findings are difficult to interpret due to the use of unconventional performance metrics. Similarly, Forsling Parborg [23] developed three smartphone-based control methods: a virtual trackpad, a directional gesture-based method that replaces the arrow keys and the OK button on a traditional remote, and a spatial method in which users move the cursor on a television by pointing the smartphone in different directions. Unlike tablet methods, these approaches did not mirror the TV display, eliminating the need for users to look at the phone. An evaluation revealed that these methods were not significantly faster than a traditional remote control and were more error-prone. Furthermore, users did not find these methods noticeably more user-friendly than the remote control. However, these methods were evaluated only in basic interactions, such as volume adjustments and playing and pausing media.

2.2 Smartwatch-based Methods

Smartwatches have been used in various systems to control interactive televisions. For example, Verweij et al. [65] developed a method that utilizes the inertial sensors of a smartwatch, enabling users to perform mid-air gestures for basic television control. Similarly, Seetharamu et al. [59] introduced a smartwatch-based system that allowed users to switch channels by performing left and right wrist gestures and scroll by performing a wrist-up gesture. Based on this work, Luna et al. [36] added push and pull gestures to a similar system, achieving relatively high accuracy rates for gesture recognition (65–97%) in standing and seated positions. Furthermore, Popovici et al. [49] designed a system that allowed viewers to switch to their favorite channels by pointing to a specific pocket on their clothing and then at the screen. This system used a Myo armband that uses electromyography (EMG) to measure muscle activity for gesture recognition. However, none of these systems were empirically evaluated in comparative studies. Nascimento et al. [41], on the other hand, proposed a method that enabled viewers to interact with and control Netflix content by self-assigning gestural shortcuts, such as drawing shapes on the screen [52], for common tasks such as pause, resume, rewind, fast forward, and volume adjustment. Although the method was not evaluated in an empirical study, participants in a usability study described it as *"nice to use."*

Roberts et al. [55] developed a smartwatch-based interactive system that allowed users to perform basic controls, such as turning the TV on/off, adjusting the volume, and switching channels, using virtual buttons on the smartwatch. In an evaluation, the system achieved a task completion time of 9.40 seconds. Subjectively, most of the participants preferred this method over a speech-based alternative. In addition, various third-party applications are available for smartwatches to control televisions. These applications are comparable to the prototype by Roberts et al. [55] and support only basic functionalities. Recently, both Google and Apple introduced TV control features on their flagship smartwatches[1], offering functionality comparable to the systems mentioned earlier (Fig. 2), further discussed in Section 4.

[1]Google introduced television control features on the Pixel Watch 3 [25], released on September 10, 2024 [24], and similarly, Apple added these features to watchOS 11 [10], launched on September 16, 2024 [56].



(a) Android Wear OS **(b) Apple watchOS**

Figure 2: Default Android Wear OS and Apple watchOS TV control applications, offering basic functionality for media interaction.

2.3 Speech-Based Methods

Speech-based methods, or voice commands, are becoming increasingly popular for interacting with smart TVs. Although the reliability of speech recognition has improved significantly over time, research shows that viewers are still reluctant to use it as their primary mode of interaction with televisions.

Berglund and Johansson [8] evaluated the user experience of voice commands for operating a television. They found that while viewers were generally satisfied with the usability of this method, they identified several scenarios where non-speech methods were preferable. These included situations where viewers needed to remain quiet (e.g., watching in mute to avoid disturbing sleeping family members), speaking with others, noisy environments, difficulty in verbally expressing intent, or browsing content without a specific goal. Eriksson and Sjogren [21] also highlighted these limitations of speech interaction and recommended that a non-speech alternative, such as a traditional remote control, be always available along with speech-based systems.

In a more recent study, Santos et al. [58] found that, although viewers enjoy using speech-based controls, they still prefer having a manual alternative available. In a follow-up study, Santos et al. [57] concluded that viewers consider speech a natural and user-friendly mode of interaction with televisions, but not always immediate or efficient for all use cases. They also reported privacy concerns and noted that failures during repeated attempts could cause frustration, described as a "brick-wall effect." Consequently, they recommended offering multiple interaction methods to accommodate different tasks. Pandey et al. [47] identified these issues as general challenges of speech-based interaction.

2.4 Spatial Interactions

Research has also explored spatial interaction with TVs, including the use of mid-air gestures for control [5] and mid-air gestural shortcuts for preferred TV channels [48]. Zaiţi et al. [67] conducted a gesture elicitation study for TV control, identifying fine-resolution gestures of the finger and hand pose for 21 television control tasks. However, these gestures were not evaluated in representative scenarios involving a television. Building on this work, Wu et al. [66] conducted a gesture elicitation study to identify free-hand gestures for 19 common TV control tasks. Unlike Zaiţi et al. [67], their set included bimanual gestures. They compared these gestures with those identified by Zaiţi et al. [67] and a conventional remote control, finding that both gesture sets were significantly faster, although users perceived the remote control as more efficient. In addition, their evaluations were limited to basic functionalities, such as adjusting volume, muting, and changing channels.

Several studies have reported fatigue and discomfort associated with mid-air gestures [5, 7], which limits their adoption as a dominant mode of interaction. Eriksson and Sjogren [21] argued that mid-air gestures might not support all TV interactions due to the limited number of practical gestures users can perform. Similarly, Kim [31] cautioned that, while gestural interaction can be effective in showcasing new technology, it often leads to a suboptimal user experience.

2.5 Contemporary Control Interfaces

In recent years, several studies have explored contemporary modes of television control. These include brain-computer interfaces that use electroencephalogram (EEG) headsets to monitor brain signals and interpret viewer intent [2], as well as tangible user interfaces, such as a squeeze-based interface that allows users to control a television by squeezing, pulling, or pushing different parts of a digital cushion [1]. Furthermore, Eriksson and Sjogren [21] developed a custom remote control combining buttons, a joystick, and an accelerometer for tilt-based interaction. However, these methods are still in their early stages and currently support only basic functions.

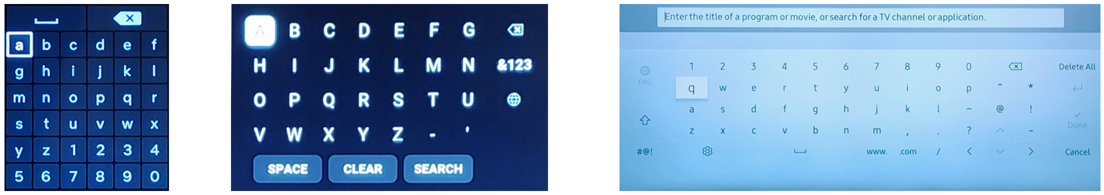
2.6 Text Entry on Televisions

The goal of text entry on televisions differs from that of computer systems, which typically involve tasks such as composing messages. On televisions, text entry is often limited to search tasks, allowing users to find content by entering just a few letters, enabled by prefix-based search completion and recommendations.

Text entry on smartwatches has been extensively studied, with research exploring various input methods, including miniature virtual Qwerty keyboards [26, 45, 60], keypads inspired by feature phones [18, 32], and gestural input techniques [44, 50]. Arif and Mazalek [3] provides a detailed review of these approaches. Similarly, text entry techniques for virtual reality have explored the use of controllers and other devices, as detailed in a comprehensive review [17]. Although many of these methods could theoretically be adapted for televisions, this potential application remains largely unexplored.

Commercially, there are numerous external devices available for entering text on televisions, including external keyboards and trackpads. Barrero et al. [7] compared four such devices for text entry on TVs: a full-length Qwerty keyboard, a miniature Qwerty keyboard, a remote pointer, and a trackpad. In the study, participants entered short English phrases using these methods. The keyboards were significantly faster than the pointer and trackpad, although all methods produced comparable error rates. However, the pointer and trackpad caused fatigue, making them uncomfortable to use for extended periods, while the participants noted that the keyboards were difficult to use in dark settings.

Unfortunately, there is no standard keyboard layout for televisions, as different manufacturers and streaming services employ varying designs. Fig. 3 illustrates three examples of keyboard layouts on streaming platforms. The first two layouts are alphabetical,



(a) Netflix (b) YouTube Premium (c) Samsung TV

Figure 3: Examples of different keyboard layouts used across three streaming platforms.

while the third follows the QWERTY arrangement. Even among alphabetical layouts, there are subtle differences. For instance, Netflix (Fig. 3a) uses a 6×5 grid, while YouTube Premium (Fig. 3b) employs a 7×4 grid. Additionally, Netflix allows numeric input directly from the main layer, whereas YouTube Premium requires switching to a dedicated numeric layer. These variations, combined with the use of remote controls for navigation, make text entry on TVs challenging, as users cannot rely on a single consistent layout for all platforms.

2.7 Focused Content Exploration

In 2011, Amazon introduced the X-ray widget in the Amazon Kindle Touch, later expanding it to Amazon Prime Video [11]. This feature acts as a digital concordance, providing users with focused information about the content. When users press the up button on the remote, they can access details such as actor bios, photos, filmographies, character backstories, and soundtrack information, including song titles and performers [30]. In addition, the X-ray feature offers trivia, Easter eggs, and bonus video content. The most relevant details are displayed in a bar above the lower bezel, and users can scroll through the bar horizontally using the left and right buttons on the remote (Fig. 4a). Selecting the "X-Ray" option expands the view to full-screen, allowing deeper exploration through categorized tabs (Fig. 4b).

3 Pilot Study: Interaction Modalities

We first conducted a pilot study to investigate different interaction modalities on smartwatches, including directional flicks, wrist gestures, taps on the screen, and a force-based interaction approach. The goal was to determine ideal interaction modalities that could be integrated into a comprehensive system for controlling smart televisions. Although prior research identified directional flicks and taps as the most effective interaction methods on smartwatches [53], we deemed this study necessary due to conflicting results in the relevant literature [13, 23], particularly when touchscreens were used as external control devices rather than for interacting with their own interface. We opted not to integrate mid-air and other gestural methods due to the challenges associated with these techniques, as discussed earlier (§2.4). Similarly, we excluded speech-based interaction, both due to the inherent limitations of this modality (§2.3) and our preference for a non-recognition-based system, which has been identified as more reliable [8, 57].

3.1 Participants

Twelve participants took part in the study (M = 26.83 years, SD = 6.1). Seven of them identified as female, and five as male. They were all right-handed and wore the watch on their left wrist. Eight of them were smartwatch owners (M = 3 years, SD = 2.4), while the other four, although not owners, were familiar with smartwatches. Each participant received US $10 for volunteering in the study.

3.2 Apparatus

All studies reported in this work used the same smartwatch and television, described in §5.1.

3.3 Design

We compared four indirect interaction methods for television via smartwatches: directional flicks (four directions: up, down, left, right), wrist gestures (two directions: twisting the wrist towards and away from the body), tapping on the display, and a force-based interaction approach (three pressure levels: soft, regular, and hard). The study protocol was reviewed and approved by the Institutional Review Board (IRB).

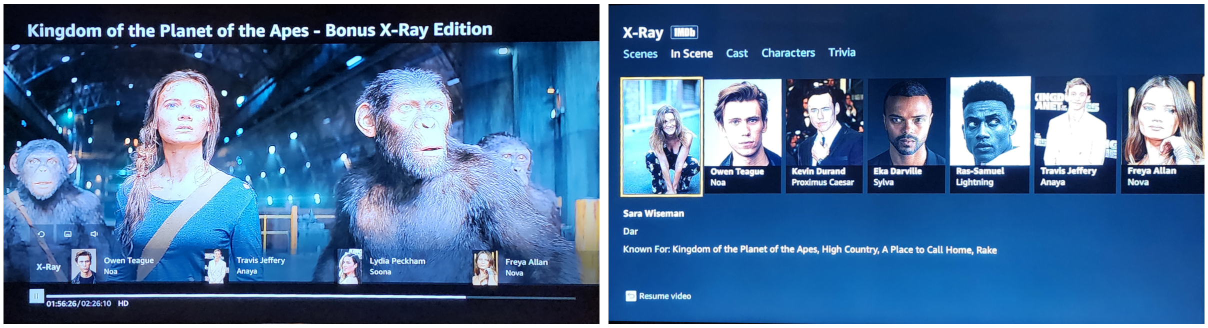
We conducted a separate pilot study (N = 3, 1 female, 2 male, M = 29 years) to determine optimal threshold values for these input methods on the study device. For flick detection, a movement threshold of 10 pixels was used, thus, finger motion needed to exceed this distance to be recognized as a flick. For wrist twists, a flat position was defined within a range of 0.70° to 10.97°. Any angular deviation beyond this range in either direction was registered as a twist. For force-based input, taps with a force below 0.01 and a duration over 132 ms were classified as soft, while those with a force above 0.02 and a duration over 551 ms were classified as hard.

In summary, the study included 12 participants performing 10 actions across four independent variables, with 20 trials per action, resulting in a total of 2,400 actions. The dependent variables were the following performance metrics.

- **Task completion time:** The average time, in seconds, that participants took to accurately perform the actions within a given method.
- **Error rate:** The average number of errors made while performing an action. An error was recorded whenever the performed action did not match the one presented.

3.4 Procedure

Upon arrival, participants were briefed about the study, gave their informed consent, and practiced each of the four methods by performing each action twice. The study then began. The television displayed the actions in a randomized order (Fig. 5), and participants were instructed to perform each action on the smartwatch



(a) (b)

Figure 4: The X-ray feature on Amazon Prime Video: (a) Users can access key details in a bar by pressing the up button on the remote. (b) Selecting the "X-Ray" option on the bar expands the view to full-screen.

without looking at its display. Once the correct action was recognized, the television displayed a success notification and moved to the next action. If an action was not recognized or was performed incorrectly, participants were asked to try again. After completing the study, participants took part in a debriefing session, where they shared their experiences and provided feedback.

Figure 5: A participant taking part in the pilot study.

3.5 Results & Discussion

An ANOVA revealed a significant effect of method on task completion time ($F_{3,11} = 14.54, p < .001$). A Tukey-Kramer multiple-comparison test identified three distinct groups: {tap}, {directional gesture}, and {force, wrist tilting}, with the first two being significantly faster (> 30%) than the last group. A significant effect on error rate was also identified ($F_{3,11} = 28.18, p < .0001$) . A Tukey-Kramer test showed that the force-based method had a significantly higher error rate, while other methods were comparable. Fig. 6 illustrates the average task completion time and error rate for each method. In the post-study debrief, all participants preferred tap and flick methods for their speed, ease of use, and lower error rates. Based on these findings, we designed WristFlick using taps and flicks to enhance efficiency and user-friendliness.

4 WristFlick: Design & Development

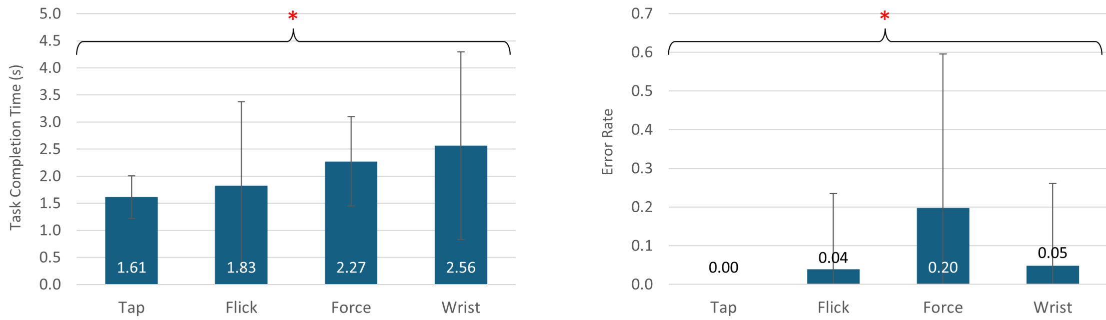
WristFlick is a comprehensive system that enables users to control a smart television through a smartwatch. Upon launch, WristFlick connects to the television using a client-server protocol over a shared Wi-Fi network. Users can choose from three interaction modes: media control and navigation, exploration of focused content, and searching, to control the television or streaming services. Since smartwatches are personal devices, they allow multiple household members to interact with the TV independently, eliminating the need for sharing a single remote. However, a smartwatch can also serve as a shared, active tangible device [19] for TV interaction when needed.

Its design and interaction approaches were determined through an iterative process. Initially, all possible options were identified based on the existing literature and the affordances of the smartwatch form-factor. These options were evaluated through multiple lab trials, in which lab members informally tested their effectiveness and usability, as well as through pilot studies with human subjects. The final selections were made based on performance metrics such as speed, accuracy, and user preference. We discuss the design of the system in the following sections and list the alternatives considered before selection.

4.1 Media Control & Navigation Behaviors

With WristFlick, users can navigate through the TV guide or explore different categories available on streaming services, such as genres, by flicking up or down on the smartwatch display (Fig. 7a). Once the desired listing or category is displayed on the TV screen, users can scroll through the available programs or titles by flicking left or right on the smartwatch display (Fig. 7b). To select a program or title, users can simply tap anywhere on the smartwatch display (Fig. 7c).

After selecting a program or title, the system automatically switches to media control mode. In this mode, users can pause or play by tapping the display, fast-forward 5 seconds by flicking right, and rewind 5 seconds by flicking left. For progressive fast-forwarding or rewinding, users can flick and hold their finger on the display (Fig. 7d). The system progresses based on default increments. In our implementation, similar to Netflix [42], the first two seconds advance or rewind by 5 seconds, from two to five seconds the interval increases to 15 seconds, between five and ten seconds it jumps by 1 minute per flick, and after ten seconds, the jump



(a) Task Completion Time **(b) Error Rate**

Figure 6: Average task completion time (seconds) and error rate across methods. Statistically significant differences are marked by a red asterisk. Error bars indicate ±1 standard deviation.

(a) Flick Up & Down **(b) Flick Left & Right** **(c) Tap** **(d) Flick Right & Hold** **(e) Rotate Upward**

Figure 7: A subset of interactions in WristFlick: (a) and (b) demonstrate category and media navigation, respectively, (c) shows a tap to select an item (channel or title), (d) demonstrates flicking right and holding for progressive fast-forward, and (e) shows rotating the crown upward to increase volume.

interval increases to 2 minutes. Users can adjust the volume by rotating the crown of the smartwatch. Rotating it upward increases the volume by one unit, while rotating it downward decreases the volume by one unit (Fig. 7e). Users can exit the system at any time by long-pressing the display for more than 500 milliseconds. Table 1 summarizes these interactions.

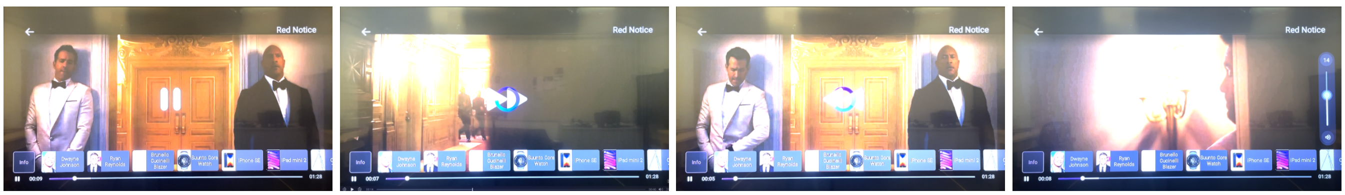
4.1.1 Pilot Studies: Alternative Designs. In addition to many informal lab trials, we conducted three pilot studies to explore alternative interaction approaches to those presented in Table 1. These studies measured task completion time, error rate, and user preferences, as described in Section 3.3. Some participants were shared across the studies, but not all. The first pilot study (N = 4, 1 female, 3 male, M = 29 years) compared virtual buttons with directional flicks for navigation. While virtual buttons were faster, participants reported frustration with precise selection on the small screen, resulting in significantly higher error rates. This aligns with prior findings [3]. We also considered an alternative directional mapping but chose not to test it, as participants in a pre-study session found the selected directions more intuitive, aligning with the arrow keys on a traditional remote. In the second pilot study (N = 4, 1 female, 3 male, M = 29 years), we compared dedicated virtual buttons for media control with tapping anywhere on the display. As in the first pilot study, participants found virtual buttons frustrating and error-prone. In contrast, the tap-based approach, which did not require precise target selection, was faster, more accurate, and preferred by the participants. This study also compared long-press with dedicated virtual buttons to turn on/off the TV. Long-presses were slower by design, yet outperformed the other methods in terms of accuracy and user preference. The third pilot study (N = 2 male, M = 29.5 years) compared sliders, flick gestures, and rotation of the crown to adjust the volume. Rotating the crown emerged as the best performer in terms of speed, accuracy, and user preference. All pilot studies used circular virtual buttons sized 40–90 dp.

Notably, the default features of Google and Apple smartwatches, as well as third-party applications, rely primarily on dedicated virtual buttons for most actions, such as returning, returning to the main menu, playing and pausing media, and muting/unmuting [25, 56]. These virtual buttons performed poorly in the pilot studies. However, the Apple Watch also uses the crown to adjust the volume and directional flicks to navigate the Apple TV menu options [56].

4.1.2 Visual Feedback on Television. The custom television interface provides visual feedback for all interactions, adhering to conventions used in popular streaming services (Fig. 8). For media selection, feedback is simply reflected by changes on the television screen. When pausing, fast-forwarding, or rewinding, corresponding icons (e.g., pause, forward, rewind) are displayed at the center of the screen and fade out after 1 second. During channel changes or volume adjustments, the interface shows the channel number or

Table 1: Navigation and media control actions in WristFlick.

Navigation Mode	Action	Unit	Media Control Mode	Action	Unit
Next category	Flick up	1	*Pause/Play*	Tap	1
Previous category	Flick down	1	*Fast-forward*	Flick right	5 seconds
Next channel or title	Flick left	1	*Rewind*	Flick left	5 seconds
Previous channel or title	Flick right	1	*Continuous rewind*	Flick right & hold	Progressive
Select a channel or title to play	Tap	1	*Continuous fast-forward*	Flick left & hold	Progressive
			Volume Up	Rotate the crown up	1
Exit the system	Long-press	–	*Volume Down*	Rotate the crown down	1



(a) Pause **(b) Fast-forward** **(c) Rewind** **(d) Volume Control**

Figure 8: Feedback displayed on the television interface during media control. The first three images show pause, forward, and rewind icons at the center of the screen, while the last image displays a volume control slider on the right side of the screen. This feedback aligns with standard television and streaming service interfaces.

volume level slider on the right side of the screen, which fades out after 2 seconds. These behaviors are based on standard practices in television interfaces.

4.2 Focused Content Exploration Behavior

During media viewing, users can activate focused content exploration by performing a two-finger tap on the smartwatch display (Fig. 9a). This brings up a focused content list on the television screen, similar to the X-ray bar in Amazon Prime Video (Fig. 4). The smartwatch displays a card for the first item on the list, and users can flick to the left or right to navigate through all cards (Figs. 9b, 9c). As users browse the cards, the corresponding item is highlighted on the television, so they do not need to constantly look at the smartwatch during interaction.

Once the desired item is on the smartwatch display and highlighted on the TV display, it can be tapped to flip the card and view additional details on the smartwatch (Fig. 9d). This action does not interrupt media playback, as the details are not shown on the television, preserving the viewing flow. If users wish to display the details on the television, they can flick up on the smartwatch with two fingers, effectively "throwing" the information onto the TV screen. Flicking down with two fingers brings the information back to the smartwatch, allowing it to resume playing media. This throwing behavior, however, was not evaluated in the user study. Users can scroll through the content on a card by flicking up and down (Fig. 9e). They can return to navigation mode by performing another two-finger tap. Table 2 summarizes the interactions used to explore focused content. These interactions were selected based on findings from the pilot studies (§4.1.1) and lab trials. The two-finger tap was chosen because it did not interfere with other smartwatch functions and was easily distinguishable from other gestures.

Table 2: Focused content exploration actions in WristFlick.

Focused Content Mode	Action
Show or hide the focused content cards	Two-finger tap
Navigate to the next card	Flick left
Navigate to the previous card	Flick right
Flip the card	Tap
Display the card on the television	Two-finger flick up
Remove the card from the television	Two-finger flick down
Scroll up on the flipped card	Flick up
Scroll down on the flipped card	Flick down

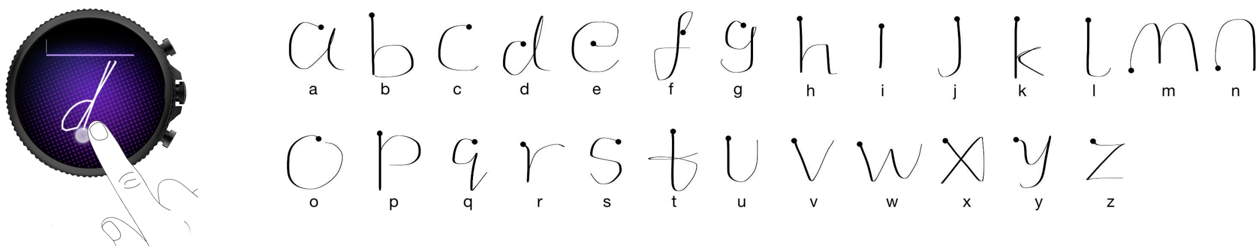
4.3 Searching Behavior

WristFlick allows users to search by writing directly on the smartwatch display. When users want to search for media, they can begin writing characters one at a time on the display. This action automatically activates the search mode on the television screen. We adopted a simple approach inspired by prior work [44], as our focus differs from traditional text entry research. As discussed in Section 2.6, our aim was not to develop a general-purpose input method for smartwatches or to optimize it using language models, but rather to create a system-specific solution for media search that takes advantage of the host system's predictive search functionality.

The input module was developed using the $Q Super-quick Recognizer, a 2-D gesture recognizer designed for rapid prototyping of gesture-based interfaces, especially on low-power mobile devices and wearables [63]. Although the algorithm supports multi-stroke gestures, we chose a unistroke gesture set (Fig. 10b) for better performance (§4.1.1). The algorithm used three templates per letter, with a minimum match score threshold of 0.20 for recognition, where

(a) Two-finger Tap **(b) Flick Left** **(c) Flick Right** **(d) Tap** **(e) Flipped Card**

Figure 9: Focused content exploration with WristFlick: (a) demonstrates performing a two-finger tap to activate focused content mode; (b) and (c) demonstrate flicking left and right to view the previous and next item cards, respectively; (d) shows a tap to flip the card for more details; and (e) displays the flipped card with additional information.



(a) Writing **(b) Gesture Set**

Figure 10: Searching with WristFlick: (a) writing letters on the smartwatch display, and (b) the gesture set used.

1.0 represents a perfect match. This effectively avoided conflicts between navigational gestures and text input, as all navigational gestures or flicks typically scored below 0.01 with the recognizer.

WristFlick recognizes a letter when users lift their finger and displays it on the smartwatch. This action also activates the input mode, showing a text field and a gesture input area (Fig. 10a). With the recognition model stored locally, there is no noticeable lag between writing and recognition. If the letter is correct, users press the smartwatch's physical button to send it to the television, which opens the search console and displays media recommendations on the television (Fig. 1c). To delete a letter, users flick left on the input field, likewise they flick right to add a space. Navigation through recommendations and search results adheres to the conventions described in Section 4.1. If an incorrect input is sent by mistake, users can press the physical button again to cancel and return to the previous state. Table 3 summarizes the interactions used for media searching.

Table 3: Searching actions in WristFlick.

Search Mode	**Action**
Activate search mode	Write on the display
Send a letter or text to the television	Crown press
Delete a letter	Flick left
Enter a space	Flick right
Send a letter or text to the television	Crown press
Clear letter or text sent to the television	Crown press

For simplicity, number entry was excluded from the experimental tasks. However, it can be easily integrated by expanding the gesture set and adding the corresponding templates. Similarly, WristFlick can support multiple languages either through dedicated gesture sets for each language or by using phonetic or transliteration approaches, where gestures are mapped to phonemes instead of individual letters [37]. While WristFlick could also be extended to work in combination with speech commands, that integration is beyond the scope of this work.

4.3.1 Pilot Studies: Alternative Designs. We conducted two pilot studies to explore alternative interaction approaches to those presented in Table 3. The first pilot study (N = 3, 1 female, 2 male, M = 28.3 years) compared a unistroke gesture set with a multi-stroke gesture set, where participants entered all letters of two alphabets. The unistroke set achieved significantly higher accuracy rate (over 99%) compared to the multi-stroke set. The flexibility of the multi-stroke option allowed users to draw characters or symbols in various ways, leading to increased recognition errors. This variability required extensive training data to ensure high proficiency and accuracy, making it impractical for the system. Furthermore, participants did not report any significant differences between the two sets in terms of learnability and usability. In the second pilot study (N = 2 male, M = 29.5 years), we compared a physical upper-side button on the smartwatch with a virtual button placed at the bottom of the screen (130×50 dp) for confirming search recommendations. Consistent with previous pilot studies (§4.1.1), the virtual button resulted in significantly more errors than the physical one. Also, participants found the physical button to be more user-friendly and reliable.

5 Evaluation Protocol

We conducted a three-session user study to evaluate WristFlick. Each session focused on comparing WristFlick's performance in media control, navigation, and search, respectively, against a traditional remote control. The first two sessions took place on the same day, separated by a 5- to 15-minute break, while the third session was scheduled up to five days later. Since not all participants were scheduled on the same day, the complete study took several months to complete. This session order was chosen because the features introduced in the earlier sessions were used in the latter ones, ensuring participants could focus on learning and performing the tasks specific to each session. In other words, since the study did not recruit new participants for subsequent sessions, re-learning the previously used method was not necessary.

All sessions used the same apparatus, setup, participants, and performance metrics. However, the tasks, study design, and procedures varied to align with the goals of each session, with some metrics being unique to specific sessions. The study protocol was reviewed and approved by the Institutional Review Board (IRB).

5.1 Apparatus

We used a 1,651 mm LG UL3G-B Series[2] commercial display monitor with a built-in Quad Core SoC and speakers. The monitor provides a resolution of 3,840 × 2,160 (UHD) and runs on the Linux-based operating system webOS 4.0 for smart TVs. However, we connected a 4th generation Chromecast[3] to the display to access the Google TV interface (Fig. 11a). The Chromecast runs on Android TV OS, which simplified the connectivity and data transfer between the Android-based smartwatch and the TV interface. It supports up to 4K HDR at 60 FPS and includes dual-band Wi-Fi 802.11ac (2.4/5 GHz) and Bluetooth connectivity. An AuviPal G9 smart remote (167.6 × 55.88 × 16 mm, 99.8 g) was used with the television (Fig. 11b). The remote supports both IR and 2.4GHz RF wireless technology, providing a control range of approximately 84 cm. It includes five programmable keys and integrates seamlessly with the Google TV interface.

We used a Fossil Gen 6 FTW4061V smartwatch (44 mm, 118 g) with a resolution of 416 × 416 at 326 ppi (Fig. 11c). The smartwatch features two configurable push buttons and a rotating crown that serves as the home button. It is powered by the Qualcomm Snapdragon Wear 4100+ processor, with 8 GB of storage and 1 GB of RAM, providing enough processing power for efficient on-device interactions. The built-in Bluetooth 5.0 LE ensures compatibility with the television. The smartwatch runs on Google Wear OS, which seamlessly integrates with Android TV OS.

(a) Chromecast (b) AuviPal G9 (c) Fossil Gen 6

Figure 11: The devices used in this work (not shown to scale).

5.2 Baseline Condition Selection

We chose to use a traditional remote control as the baseline condition for our evaluations for several reasons. First, both third-party and native applications on Wear OS and watchOS rely heavily on precise target selection for most actions, thus, requiring users to tap on small interactive elements like buttons or menu items (Fig. 2). In the pilot studies reported in Section 4.1.1, participants performed poorly with these approaches in terms of both speed and accuracy, and rated them unfavorably compared to our crown- and flick-based, target-agnostic methods. Furthermore, prior research has also shown that target-agnostic and physical crown-based approaches are more accessible for individuals with motor disabilities [51]. As such, including these alternatives in the final evaluation was neither meaningful nor fair. In addition, many native and third-party applications were incompatible with the devices used in our study, limiting their practical utility.

Second, neither third-party applications nor the default smartwatch features supported all three interaction modes offered by WristFlick. As a result, they could not serve as consistent baselines across all sessions. While it was technically possible to use different baseline methods for different tasks (e.g., using a miniature physical QWERTY keyboard for search), this would have introduced an unfair advantage since such tools are optimized for specific tasks rather than general-purpose use. Moreover, this approach would have introduced practical challenges, including the need to train participants on multiple baseline methods per session. This would have significantly extended session durations, complicating both recruitment and scheduling.

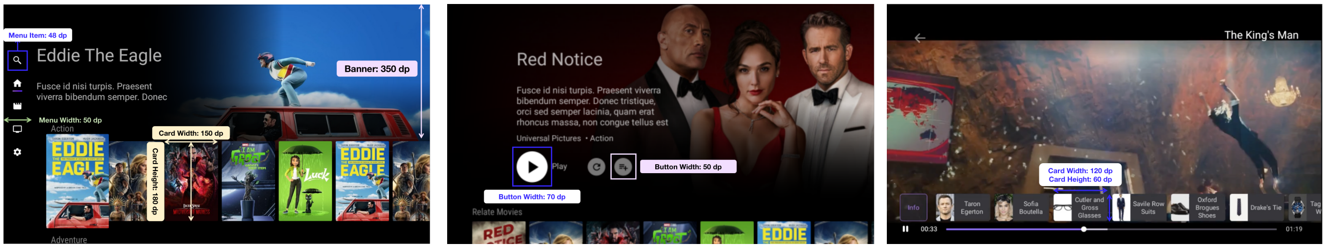
Finally, despite advancements in smart features and the growing presence of voice-based interaction, conventional remote controls remain the most widely used and preferred method for interacting with televisions [27, 54]. Therefore, we argue that using a remote control as the baseline remains a valid and relevant choice.

5.3 Implementation & Source Code

The complete system was developed using Android 13 (API level 33)[4]. The television interface was built with the Android Leanback library, while the smartwatch interface was developed using the Wear library. To establish communication between the television and the smartwatch via Wi-Fi, we used the Android TV Remote Service. Since there is no publicly available official documentation for this service, our implementation relied on unofficial documentation from GitHub [6] and a publicly accessible repository [20].

We invested significant time and effort into ensuring that both the smartwatch and television interfaces appeared professional-grade by focusing on even the smallest details. For the smartwatch

[2]HP Display Monitor: https://www.lg.com/us/business/digital-signage/lg-65ul3g-b
[3]Google Chromecast: https://store.google.com/us/product/chromecast_google_tv
[4]Android 13: https://developer.android.com/about/versions/13



(a) Content Browsing **(b) Movie Details** **(c) Media with Focused Content**

Figure 12: Different views of the custom television interface, designed to closely replicate the look and feel of Netflix. The figure also includes dimensions of key interactive elements.

interface, we adhered to the official design guidelines for Android-based devices [16], including the recommended dimensions for applications, interactive elements, behaviors, fonts, and color palettes.

For the television interface, we closely modeled our design after the Netflix interface while also adhering to the Android TV design guidelines [15]. We replicated the look and feel, fonts, color palette, and subtle animations for feedback, such as a 400 ms sliding transition when changing views and a 1.2x zoom-in/out animation when highlighting cards. However, since Netflix does not support focused content, we adopted the design of Amazon Prime Video's X-ray feature, maintaining the same visual style and behavior. In fact, during the study, many participants could not distinguish between our custom interface and Netflix until we demonstrated all the features explored in the study. Fig. 12 presents the template used for the television interface with dimensions. The implementation of the search system is discussed in §4.3. We have made the source code for both the television interface[5] and the smartwatch interface[6] publicly available to support replication, validation, and further research.

5.4 Movie Database

For evaluation, we added 250 movie titles to the TV interface, along with their posters and metadata, including titles, synopses, runtimes, and release dates, which were displayed on the preview page (Fig. 12b). These media files and data were sourced from various movie databases. Due to copyright restrictions, we were unable to use full movies in our system. Instead, we downloaded 10-15 minute publicly available clips of these films. The movie titles were strategically selected to represent a diverse mix of classic, critically acclaimed, independent, and blockbuster films, as well as the twelve most popular movie genres [43]. Some titles were classified under multiple genres. In addition, the titles were chosen to ensure the representation of the English alphabet, aligned with typical letter frequencies [35, pp. 36–37]. The Pearson correlation coefficient between the letter frequencies in the titles and those in the English language indicated a very strong positive correlation ($r = 0.956$), which is particularly relevant for the search session (§8). The length of the movie titles ranged from 3 to 43 characters, including spaces, with an average of 14 characters (SD = 6).

[5]Television Interface (Server): https://github.com/theiilab/TVInteractions
[6]Smartwatch Interface (Client): https://github.com/theiilab/SmartwatchInteractions

5.5 Experimental Setup

We set up the experiment in a laboratory to resemble a typical living room by placing a couch and a side table, creating a comfortable and familiar space for participants. The television was mounted on an Onkron TV stand with a rolling cart, 90 cm above the ground and 360 cm from the couch to ensure an optimal viewing angle. Participants were asked to sit on the couch and interact with the television using the remote control or the smartwatch, as they would in their own living room. Only the researcher and the participant were present during the study. Once a session began, the researcher moved to the other side of the room, out of the participant's line of sight, to minimize distractions. The setup is illustrated in Fig. 13.

(a) WristFlick **(b) Remote Controller**

Figure 13: Two participants taking part in the main study.

5.6 Performance Metrics

The following performance metrics were calculated for all sessions. These metrics were automatically computed and recorded by the experimental devices.

- **Task Completion Time:** This metric represents the average time participants took to complete an experimental task. While it was calculated for all sessions, the tasks in each session differ and are discussed in their respective sections.
- **Actions per Task:** This metric reflects the average number of actions, including taps, flicks, and crown rotations, performed to complete each experimental task.
- **Task Efficiency:** This metric measures the ratio of the expected number of actions (A_e) needed to perform a task to

the actual number of actions (A_a) performed by participants. Task efficiency is calculated using the formula: $\frac{A_e}{A_a} \times 100$. An efficiency ratio of 100% indicates that the participant performed the exact number of expected actions, while a lower ratio signifies inefficiency (the participant performed more actions than expected).

5.6.1 Subjective Measures. We also administered the following questionnaire at the end of the study to assess WristFlick's usability and participants' engagement with the system.

- **Usability Questionnaire:** We used a 5-item scale [53], modeled after the System Usability Scale (SUS) [9], to assess perceived speed, accuracy, learnability, ease of use, and willingness to use the methods, which are rated on a 5-point Likert scale (Appendix A.1).
- **Core Flow Scale:** Jackson and Csikszentmihalyi [29] defined "flow" as an optimal psychological state that occurs when users are fully immersed in a task, with all aspects of the activity coming together seamlessly. As a result, flow is considered a positive and desirable psychological experience, particularly with entertainment systems [28]. We used the 10-item scale [38], which assesses the central subjective (or phenomenological) experience of flow. This scale asks users to rate ten statements on a 7-point scale (Appendix A.2), with the average of these ratings representing the core flow of the examined system [28].

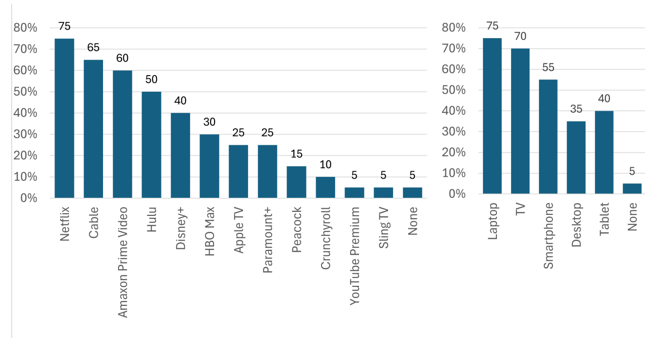
5.7 Participants

Twenty participants from the local university and community college took part in the study. They each received US $35 for participation. Their ages ranged from 20 to 31 years (M = 22.8, SD = 3.3), with two participants opting not to disclose their age. Ten participants identified as female and ten as male. Eighty percent (N = 16) were undergraduate students, while 20% (N = 4) were pursuing post-secondary degrees.

5.7.1 Language Proficiency. Based on the 5-point Interagency Language Roundtable (ILR) scale [22], the majority (60%, N = 12) rated their English proficiency as "Level 5: Native or bilingual proficiency," six participants (30%, N = 6) rated themselves at "Level 4: Full professional proficiency," one participant (5%, N = 1) rated at "Level 3: Professional working proficiency," and one (5%, N = 1) rated at "Level 2: Limited working proficiency."

5.7.2 Technology Experience. All participants were frequent smartphone users, with an average of 9.4 years of experience (SD = 2.7). Eight participants (40%) owned smartwatches with an average of 2.6 years of ownership (SD = 2.5), while the remaining 60% (N = 12) had used smartwatches but did not own one. Nineteen participants (95%) were right-handed and wore the smartwatch on their left wrist, while one participant was left-handed and wore it on their right wrist.

5.7.3 Video Consumption. Most participants (65%, N = 13) reported subscribing to both cable TV and streaming services to access video media, while 20% (N = 6) exclusively used streaming services. One participant (female, 20 years) indicated that she did not subscribe to either cable or streaming services. Among streaming platforms,

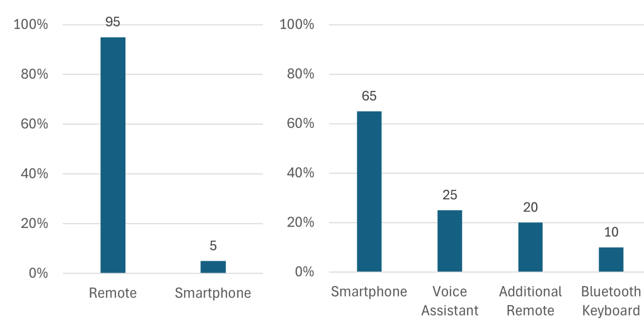


(a) Media Services (b) Devices

Figure 14: Most commonly used media services and devices for video content consumption among participants. Totals exceed 100% as participants often subscribe to multiple services and use more than one device.

Netflix was the most popular (75%, N = 15), followed by Amazon Prime Video (60%, N = 12), and Hulu (50%, N = 10). The majority (75%, N = 15) subscribed to more than one streaming service, with an average of three services per participant. Fig. 14a illustrates the percentage of media services used by participants.

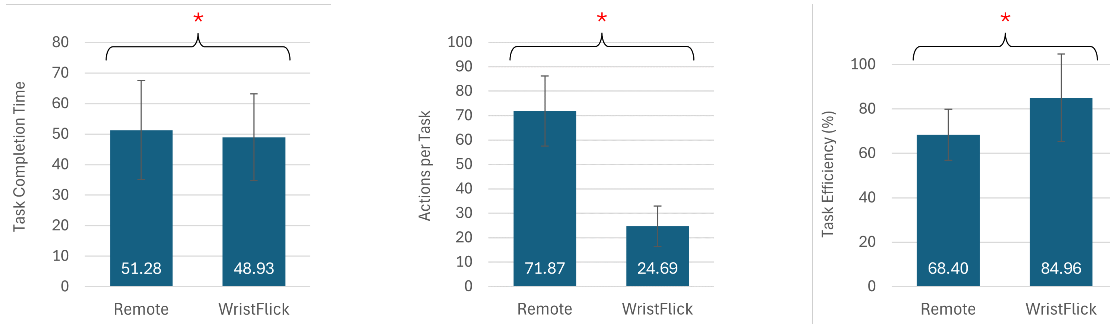
Most consumed media via laptop computers (75%, N = 15), followed by televisions (70%, N = 14) and smartphones (56%, N = 11). Some participants also used desktop computers and tablets. Fig. 14b illustrates the percentage of devices frequently used by participants.



(a) Primary (b) Secondary

Figure 15: Percentages of primary and secondary devices used to control televisions and media, and to search for content. Totals exceed 100% as participants often use multiple devices.

5.7.4 Media Control & Navigation. When asked about which devices they use to control media on television, nearly all participants (95%, N = 19) reported using the default remote control that came with the television. One participant (female, 20 years) stated that she used a smartphone application to operate her television. All participants (100%) indicated that they sometimes used a secondary or tertiary device to control the TV. The majority mentioned smartphone applications (65%, N = 13), followed by voice assistants or speech-based methods (25%, N = 5). Furthermore, some participants



(a) Task Completion Time **(b) Actions per Task** **(c) Task Efficiency (%)**

Figure 16: Average task completion time (seconds), actions per task, and task efficiency (%) across methods in Session 1. Statistically significant differences are marked by a red asterisk. Error bars indicate ±1 standard deviation.

indicated using an additional remote either provided by their cable service provider or purchased separately (20%, N = 4). Some (10%, N = 2) also mentioned using Bluetooth keyboards, particularly for searching, as entering text with a remote control is difficult. Fig. 15 illustrates the distribution of devices used to control televisions.

6 Session 1: Media Control & Navigation

This session focused on comparing WristFlick's media control and navigation performance with that of a traditional remote control.

6.1 Experimental Tasks

An experimental task in the session involved performing the following media control and navigation actions:

(1) Find a specific movie title (x)
(2) Play the movie for t seconds
(3) Increase or decrease (randomized) the volume by n units
(4) Fast-forward by t seconds
(5) Pause
(6) Play
(7) Rewind by t seconds
(8) Fast-forward to the end of the movie
(9) Rewind to the start of the movie

The movie titles (x) were randomly selected from the movie list (§5.4), ensuring that one title from each of the 12 genres, with no title appearing twice in a single session. The values for t (time in seconds) were randomly selected by the system between 5 and 10 seconds, and the volume adjustments (n) were randomly selected between 2 and 5 units. The sum of these values for each action was balanced across all participants. These actions were carefully chosen to ensure that all the media control and navigation interactions proposed in the system were performed by the participants.

6.2 Design & Procedure

The session followed the protocol outlined in Section 5. After introducing the study and collecting informed consent, participants were given the opportunity to practice media control and navigation tasks by performing two tasks using both WristFlick and the remote control. These practice tasks and movies were not repeated in the main session. Participants were then asked to sit on a couch and relax, as they would at home while watching TV. The tasks were provided to them on a printed sheet, and they were instructed to follow the actions listed. If an action was performed incorrectly, they were asked to repeat it. A side table was placed next to the couch for participants to use if they wished to set down the instruction sheet. Upon completing the session, participants were instructed to take a break of at least 5 minutes (and up to 15 minutes) before starting the next session. In summary, the design of this session was: 20 participants × 2 methods × 12 tasks (9 actions per task), resulting in a total of 480 tasks (4,320 actions).

6.3 Results

A Shapiro-Wilk test confirmed that the residuals of the response variables were normally distributed. Therefore, we conducted a two-way paired-samples t-test to compare all quantitative measures. We also report effect sizes using Cohen's d, where $d = 0.2$ indicates a small effect, $d = 0.5$ indicates a medium effect, and $d = 0.8$ indicates a large effect [12]. The entire session lasted between 30 and 45 minutes, including demonstration and practice.

6.3.1 Task Completion Time. A paired-samples t-test revealed a significant effect of method on task completion time ($t = 2.21$, $df = 19$, $p < .05$, $d = 0.50$). On average, participants completed media control and navigation tasks 5% faster using WristFlick compared to Remote. Fig. 16a illustrates the average task completion times for the two methods.

6.3.2 Actions per Task. A paired-samples t-test revealed a significant effect of method on actions per task ($t = 37.99$, $df = 19$, $p < .001$, $d = 8.49$). On average, WristFlick required 59% fewer actions to complete a media control and navigation task compared to Remote, as illustrated in Fig. 16b.

6.3.3 Task Efficiency. A paired-samples t-test revealed a significant effect of method on task efficiency rate ($t = -7.90$, $df = 19$, $p < .001$, $d = 1.77$). On average, participants were 24% more efficient in performing media control and navigation tasks with WristFlick compared to Remote. Fig. 16c illustrates the average task efficiency for each method.

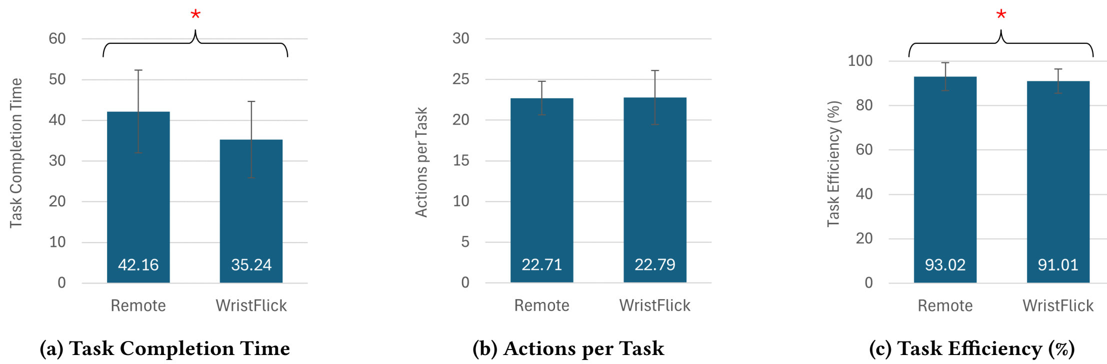


Figure 17: Average task completion time (seconds), actions per task, and task efficiency (%) across methods in Session 2. Statistically significant differences are marked by a red asterisk. Error bars indicate ±1 standard deviation.

7 Session 2: Focused Content

This session focused on comparing WristFlick's performance in accessing and acquiring focused content with that of a traditional remote control.

7.1 Experimental Tasks

A task in this session required participants to retrieve 10 specific details from the focused content section of a movie. These tasks were predetermined and identical for all participants. To ensure consistency and avoid confounding factors (such as some participants already knowing the answers) we used fictitious data in the focused content for participants to retrieve, such as the price of an item shown in the movie.

7.2 Design & Procedure

This session followed the same procedure as the previous session. However, instead of navigation and media control, participants were asked to acquire details about specified movies from the provided list. Upon completion, we scheduled the final session with each participant within five days of this session. In summary, the design was 20 participants × 2 methods × 8 movies per method × 7 details per movie, resulting in a total of 2,240 pieces of information collected.

7.3 Results

The entire session lasted between 15 and 25 minutes, including demonstration and practice. In this session, we conducted the same statistical tests as in the previous session.

7.3.1 Task Completion Time. A paired-samples t-test revealed a significant effect of method on task completion time ($t = 4.43$, $df = 19$, $p < .001$, $d = 0.99$). On average, participants completed the focused content tasks 16% faster using WristFlick compared to Remote. Fig. 17a illustrates the average task completion times for the two methods.

7.3.2 Actions per Task. A paired-samples t-test did not find a significant effect of method on actions per task ($t = -0.18$, $df = 19$, $p = .86$, $d = 0.04$). On average, participants performed approximately 23 actions to complete each focused content task. This is illustrated in Fig. 17b.

7.3.3 Task Efficiency. A paired-samples t-test revealed a significant effect of method on task efficiency rate ($t = 2.55$, $df = 19$, $p < .05$, $d = 0.57$). On average, participants were about 2% more efficient in performing the focused content tasks with Remote compared to WristFlick. Fig. 17c illustrates this.

8 Session 3: Searching

This session compared the search performance of WristFlick with that of a traditional remote control.

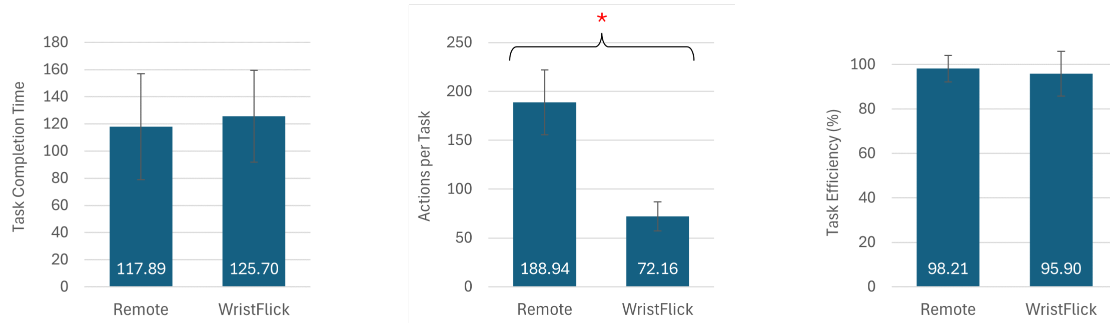
8.1 Experimental Tasks

The experimental task in this session involved searching for specific movie titles either by performing gestures or writing directly on the smartwatch, or by using a traditional remote control to navigate Netflix's virtual keyboard through its conventional arrow-based system (Fig. 3a).

8.2 Design & Procedure

This session followed the same procedure as the previous sessions. However, to familiarize participants with the gestural alphabet used in the study, they were asked to accurately enter each letter of the English alphabet at least twice. Similar training was provided for the remote control, although participants were already familiar with that method. Since the television system offered prefix-based suggestions as letters were entered, participants were instructed to select the target movie as soon as it appeared on the screen, rather than typing the entire title. As in the other sessions, error correction was mandatory, meaning participants had to select the correct movie from the list or try again. For navigation, they used the respective system's navigational approach. In summary, the design was: 20 participants × 2 methods × 3 blocks × 30 movies per block, resulting in a total of 3,600 search attempts.

Upon completion of this session, participants were asked to complete usability and flow questionnaires to evaluate various aspects of their experience with both methods and to provide additional comments or feedback on the study.



(a) Task Completion Time **(b) Actions per Task** **(c) Task Efficiency (%)**

Figure 18: Average task completion time (seconds), actions per task, and task efficiency (%) across methods in Session 3. Statistically significant differences are marked by a red asterisk. Error bars indicate ±1 standard deviation.

8.3 Results

The entire session lasted between 20 and 30 minutes, including demonstration, practice, and questionnaire. We conducted the same statistical tests as in the previous sessions.

8.3.1 Task Completion Time. A paired-samples t-test did not identify a significant effect of method on task completion time ($t = -1.86, df = 19, p = .07, d = 0.42$). On average, participants were 6% faster in performing search tasks with Remote compared to WristFlick. Fig. 18a illustrates the average task completion times for both methods.

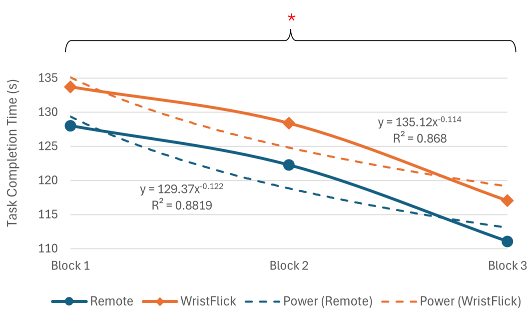


Figure 19: Average task completion times by block for the two methods fitted to power trendlines. Red asterisks indicate statistically significant differences.

Further analysis revealed a learning effect with both methods. A repeated-measures ANOVA showed a significant effect of block on task completion time ($F_{2,38} = 13.46, p < .0001, \eta^2 = 0.08$). Participants' task completion times with both methods were significantly faster in the final block compared to the first ($p < .05$). Fig. 19 illustrates task completion times by block for both methods, fitted to power trendlines.

8.3.2 Actions per Task. A paired-samples t-test revealed a significant effect of method on actions per task ($t = 16.21, df = 19, p < .001, d = 3.63$). On average, participants performed 62% fewer actions for search tasks with WristFlick compared to the Remote, as shown in Fig. 18b.

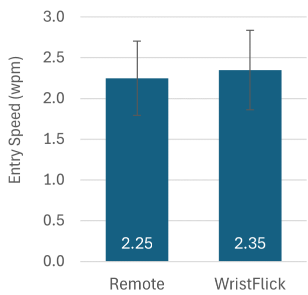
8.3.3 Task Efficiency. A paired-samples t-test did not find a significant effect of method on task efficiency rate ($t = 1.42, df = 19, p = .17, d = 0.32$). Participants demonstrated high efficiency in performing search tasks using both methods, with efficiency rates between 96–98%, as illustrated in Fig. 18c.

8.4 Text Entry Results

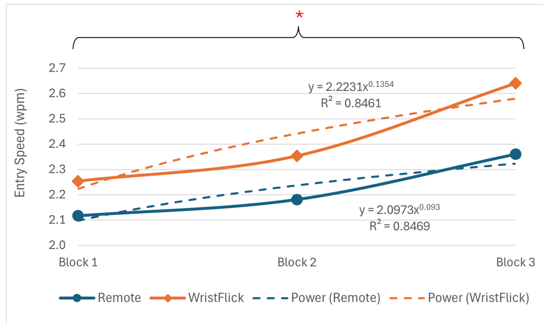
Although the focus of this session was not to evaluate text entry performance, we recorded commonly used text entry metrics [4, 62]: words per minute (average number of words entered per minute) and corrected error rate (percentage of errors committed and corrected) for both methods. We did not calculate final error rates, as error correction was mandatory, resulting in no errors in the final text. To compare performance across blocks, we conducted a repeated-measures ANOVA, as a Shapiro-Wilk test confirmed normality of residuals and a Mauchly's test confirmed sphericity.

8.4.1 Entry Speed. A paired-samples t-test did not reveal a significant effect of method on entry speed ($t = -0.98, df = 19, p = .34, d = 0.22$). On average, Remote and WristFlick yielded 2.25 wpm and 2.35 wpm, respectively (Fig. 20a). However, further analysis using a repeated-measures ANOVA identified a significant effect of block on entry speed ($F_{2,38} = 10.73, p < .0005, \eta^2 = 0.02$). A post-hoc Tukey-Kramer multiple-comparison test showed that entry speed with WristFlick was significantly faster in the final block compared to the first ($p < .05$), whereas no such effect was observed for the Remote. Fig. 20b illustrates entry speed for both methods by block, fitted to power trendlines.

8.4.2 Corrected Error Rate. A paired-samples t-test revealed a significant effect of method on corrected error rate ($t = -2.40, df = 19, p < .05, d = 0.54$). On average, participants corrected 0.61% of errors with the Remote and 18.62% with WristFlick (Fig. 21a). Further analysis using a repeated-measures ANOVA found a significant effect of block on corrected error rate ($F_{2,38} = 4.62, p < .05, \eta^2 = 0.02$). A post-hoc Tukey-Kramer multiple-comparison test showed that participants corrected significantly fewer errors with WristFlick in the final block compared to the first ($p < .05$), while no such effect was observed for the Remote. Fig. 21b illustrates corrected error rates for both methods by block, fitted to power trendlines.

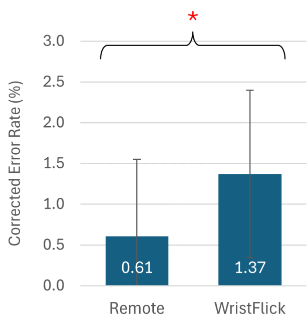


(a) Entry Speed (wpm)

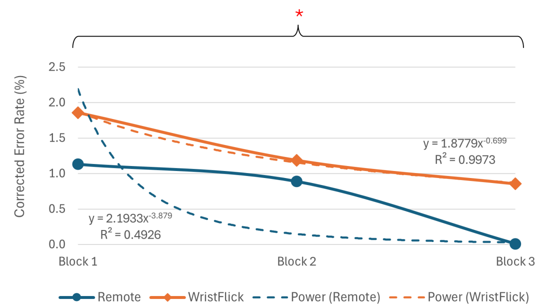


(b) Entry Speed (wpm) per Block

Figure 20: (a) Average text entry speed (wpm) across methods and (b) average text entry speed by block across methods fitted to power trendlines. Red asterisks indicate statistically significant differences. Error bars represent ±1 standard deviation.



(a) Corrected Error Rate (%)



(b) Corrected Error Rate (%) per Block

Figure 21: (a) Average corrected error rates (%) across methods and (b) average corrected error rates (%) by block across methods fitted to power trendlines. Red asterisks indicate statistically significant differences. Error bars represent ±1 standard deviation.

9 Overall Subjective Results

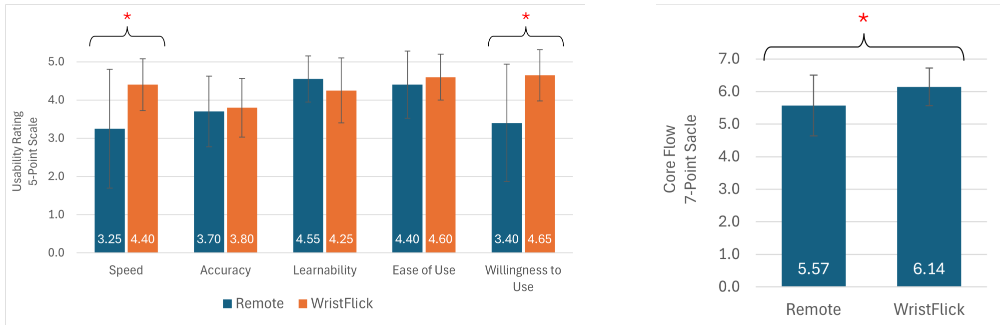
A Wilcoxon signed-rank test revealed a significant effect of method on both perceived speed ($z = -2.56, p < .05, r = -0.20$) and willingness to use ($z = -2.77, p < .01, r = 0.10$). However, no statistically significant differences were found for perceived accuracy ($z = -0.30, p = .77, r = -0.02$), learnability ($z = -1.30, p = .19, r = 0.13$), or ease of use ($z = -0.91, p = .37, r = 0.32$). Fig. 22a illustrates the average usability ratings for the two methods. Furthermore, a Wilcoxon signed rank test identified a significant effect of the method on the core flow ($z = -2.52, p < .05, r = 0.59$). Fig. 22b shows the average flow ratings for the two methods.

10 Discussion

In the study, WristFlick outperformed the traditional remote control in almost every aspect. In media control and navigation tasks, WristFlick was 5% faster and required 67% fewer actions than the remote control, both of which were statistically significant. Most notably, WristFlick demonstrated significantly higher task efficiency, with a 24% improvement. This was unexpected, as we had anticipated its efficiency to be lower or comparable to the remote control, given participants' familiarity with remote controls, and thus their reduced likelihood of making errors. Another surprising result was the significantly higher number of actions per task with the remote control compared to WristFlick. We had expected the number of actions to be somewhat comparable. A deeper analysis revealed that this was caused by participants repeatedly pressing the buttons for tasks such as volume control rather than using the press-hold functionality, increasing the total number of actions. For example, to increase the volume by 5 units, participants usually pressed the volume button five times, whereas the same could typically be achieved with just one or two crown rotations using WristFlick.

In the focused content tasks, WristFlick was also significantly faster than the remote control (16% faster). However, there was no statistical difference in actions per task. Both WristFlick and the remote control required an average of 23 actions to complete a focused content task. Unlike in navigation and control, WristFlick yielded a significantly lower task efficiency (2% lower) than the remote control. The post-study debrief suggested that this was primarily because many participants were unfamiliar with the focused content feature, which is available only on Amazon Prime Video. This unfamiliarity led to confusion and errors during the task.

In the search tasks, both methods demonstrated similar task completion times, with no significant difference. On average, participants completed a search in approximately 120 seconds using either method. However, WristFlick required significantly fewer actions, with participants performing 62% fewer actions compared



(a) Usability Rating **(b) Core Flow**

Figure 22: Average usability and flow ratings on five- and seven-point Likert scales, respectively. Statistically significant differences are marked by a red asterisk. Error bars indicate ±1 standard deviation.

to the traditional remote control. Task efficiency was also comparable between the two methods, with no statistically significant difference. The resemblance of WristFlick's gesture set to actual written alphabets may have contributed to this outcome.

To assess learning effects, we compared task completion times across blocks, anticipating that participants were becoming more proficient with WristFlick as they progressed. Results showed that participants were indeed significantly faster in the last block compared to the first for both methods. Since the learning curve had not yet flattened, it is unclear whether further practice with WristFlick would allow it to surpass the performance of the remote control. We observed a similar trend in text entry metrics. There was no significant difference in entry speed (around 2.3 wpm for both methods), but block-wise comparisons were statistically significant for both. Furthermore, participants committed and corrected more errors with WristFlick than with the remote control. This could improve with practice, and further investigation is encouraged.

Although it was not the focus of this work, we evaluated the proposed crown- and flick-based target-agnostic selection methods in a separate investigation involving users with severe motor disabilities. In that study, participants performed significantly better with these methods in terms of both speed and accuracy, and expressed a strong preference for them as they eliminate the need for precise target selection. In fact, some participants with limited dexterity due to conditions such as brain injury, spinal cord injury, arthritis, and dwarfism were unable to tap small buttons on the smartwatch, making these alternative interaction methods especially valuable.

10.1 Participant Feedback

Overall, participants found WristFlick to be significantly faster than the remote control, expressing a preference for continuing to use it. Many participants expressed frustration with the remote control. One participant (female, 23 years) commented, *"The remote made it harder to go faster and keep up."* Another participant (female, 46 years) noted, *"The remote felt cumbersome and bulky."* However, some also appreciated the precision of the remote, particularly for searching. One participant (male, 31 years) commented, *"The remote was more precise and allowed me to see the search options as I typed in the letters."*

Participants also found the smartwatch more comfortable and convenient to use than the remote control. One participant (male, 22 years) noted, *"The smartwatch was really comfortable to use."* In contrast, some participants reported fatigue when using the remote control. One participant (male, 31 years) commented, *"The remote relied heavily on my thumb. I found that it made my thumb a little sore compared to WristFlick."*

Participants did not perceive any noticeable difference between the two methods in terms of accuracy, learnability, and ease of use. These are positive results as they could suggest that a new approach like WristFlick was as easy to learn and use as a commonly used everyday device. One participant (female, 23 years) remarked, *"Once you get the hang of it, the watch is very cool, accurate, and fast."*

Participants also rated WristFlick significantly higher on flow than the conventional method (9% higher). This suggests that our aim of not disrupting the flow of watching was achieved. One participant (female, 46 years) commented, *"I appreciated how fast I could maneuver this. I was also able to focus more on watching the movie. The smartwatch provided a smooth experience."* Some participants commented on the specific benefits of using WristFlick during group viewing. They appreciated being able to explore focused content on the smartwatch without pausing the video, which could potentially disrupt others. One participant (male, 31 years) remarked, *"Having a second screen was handy as it would allow me to view the extra info without pausing the film, which may annoy others who are also watching."*

11 Limitations

We acknowledge several limitations of this work. First, we did not fully examine potential system-level constraints of using WristFlick, such as battery drain, device heating due to processing load, or long-term hardware wear. However, given the increasing battery capacity and processing power of modern smartwatches, we believe that such issues, if present now, are likely to diminish over time.

Second, the participant sample lacked demographic diversity. All participants were students from a local university or community college and were relatively young. Future studies involving a more diverse and representative population would provide a deeper understanding of the performance and preferences related to WristFlick.

However, all statistically significant differences showed medium to large effect sizes ($d > 0.50$), suggesting that these findings are likely generalizable to a broader population. However, the effect of block in text entry tasks yielded small effect sizes ($\eta^2 < 0.06$), indicating that further investigation is needed to better understand the impact of practice on text entry performance. Similarly, perceived performance metrics also showed small effect sizes ($r < 0.30$), warranting additional research. In contrast, flow demonstrated a large effect size, highlighting its strong impact in the study.

Finally, the text entry session was relatively short and may not have captured the full extent of the learning curve. A longer-term, longitudinal study would help to better assess the effects of practice on performance with the proposed text entry method.

12 Conclusion

This work demonstrated that WristFlick, a smartwatch-based system for interacting with smart TVs, offers significant improvements over a traditional remote control in terms of speed, efficiency, and overall user experience. In media control and navigation tasks, WristFlick significantly reduced the number of required actions and improved task efficiency. Even though participants were less familiar with the focused content feature, WristFlick still enabled faster access to relevant information. In search tasks, WristFlick performed comparably to the remote control in terms of speed, but required significantly fewer actions, highlighting the potential of gesture-based input for more complex interactions. Participants also found WristFlick intuitive and enjoyable to use, with many expressing a clear preference for it over the traditional remote. Furthermore, the positive feedback on flow indicates that WristFlick allows users to remain more engaged with the content itself, minimizing disruptions caused by interaction.

13 Future Work

While the results of this study are promising, several directions for future research remain. First, evaluating WristFlick over extended periods could offer valuable insights into its long-term usability and the progression of its learning curve. Understanding how performance evolves with sustained use will be essential for refining its design and enhancing efficiency. Second, investigating further customization options, such as gesture personalization or integration with additional devices (e.g., voice assistants), could increase the system's flexibility and user appeal. Lastly, exploring the potential of WristFlick in smart home applications, such as controlling lighting or appliances, could expand its utility beyond TV interactions.

References

[1] Mehul Agrawal, Vero Vanden Abeele, Karen Vanderloock, and Luc Geurts. 2015. Skweezee-Mote: A Case-Study of a Gesture-Based Tangible Product Design for a Television Remote Control. In *ICoRD'15 – Research into Design Across Boundaries Volume 2*, Amaresh Chakrabarti (Ed.). Springer India, New Delhi, 409–419. https://doi.org/10.1007/978-81-322-2229-3_35

[2] Haider Abdullah Ali, Liwa Abdullah Ali, Andrei Vasilateanu, Nicolae Goga, and Ramona Cristina Popa. 2023. Towards Design and Implementation of an EEG-Based BCI TV Remote Control. *International journal of online and biomedical engineering* 19, 10 (2023), 118–139. https://doi.org/10.3991/ijoe.v19i10.38843

[3] Ahmed Sabbir Arif and Ali Mazalek. 2016. A Survey of Text Entry Techniques for Smartwatches. In *Human-Computer Interaction. Interaction Platforms and Techniques*, Masaaki Kurosu (Ed.). Vol. 9732. Springer International Publishing, Cham, 255–267. https://doi.org/10.1007/978-3-319-39516-6_24 Series Title: Lecture Notes in Computer Science.

[4] Ahmed Sabbir Arif and Wolfgang Stuerzlinger. 2009. Analysis of Text Entry Performance Metrics. In *2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*. IEEE, Toronto, ON, Canada, 100–105. https://doi.org/10.1109/TIC-STH.2009.5444533

[5] Ahmed Sabbir Arif, Wolfgang Stuerzlinger, Euclides Jose de Mendonca Filho, and Alec Gordynski. 2014. Error Behaviours in an Unreliable in-Air Gesture Recognizer. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. Association for Computing Machinery, New York, NY, USA, 1603–1608. https://doi.org/10.1145/2559206.2581188

[6] Aymeric. 2023. Google TV Remote Control. https://github.com/Aymkdn/assistant-freebox-cloud/wiki/Google-TV-(aka-Android-TV)-Remote-Control-(v2)

[7] Aurora Barrero, David Melendi, Xabiel G. Pañeda, Roberto García, and Laura Pozueco. 2016. Evaluation of Text Entry Methods for Interactive Digital Television Applications with Devices Alternative to Conventional Remote Controls. *International Journal of Human–Computer Interaction* 32, 10 (Oct. 2016), 765–776. https://doi.org/10.1080/10447318.2016.1195586 Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10447318.2016.1195586.

[8] Aseel Berglund and Pontus Johansson. 2004. Using Speech and Dialogue for Interactive TV Navigation. *Universal Access in the Information Society* 3, 3 (Oct. 2004), 224–238. https://doi.org/10.1007/s10209-004-0106-x

[9] John Brooke. 1996. SUS: A 'Quick and Dirty' Usability. In *Usability Evaluation in Industry* (1st ed.), Patrick W. Jordan, B. Thomas, Ian Lyall McClelland, and Bernard Weerdmeester (Eds.). Vol. 189. CRC Press, Boca Raton, FL, USA, 189–194. Publisher: Taylor & Francis.

[10] Jade Bryan. 2024. watchOS 11 Makes Your Apple Watch a More Powerful TV Remote. https://www.nextpit.com/apple-watch-tv-remote-control-features-siri-volume-power-off

[11] Kenneth Butler. 2011. Video: Amazon Kindle Touch's X-Ray Reference Tool Makes eReading Easier. https://www.laptopmag.com/articles/video-amazon-kindle-touchs-x-ray-reference-tool-makes-ereading-easier

[12] Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences* (2nd ed. ed.). L. Erlbaum Associates, Hillsdale, N.J.

[13] Dale Cox, Justin Wolford, Carlos Jensen, and Dedrie Beardsley. 2012. An Evaluation of Game Controllers and Tablets as Controllers for Interactive TV Applications. In *Proceedings of the 14th ACM international conference on Multimodal interaction (ICMI '12)*. Association for Computing Machinery, New York, NY, USA, 181–188. https://doi.org/10.1145/2388676.2388713

[14] Volnei da Silva Klehm, Rodrigo de Souza Braga, and Vicente Ferreira de Lucena. 2022. A Survey of Digital Television Interactivity Technologies. *Sensors* 22, 17 (Jan. 2022), 6542. https://doi.org/10.3390/s22176542 Number: 17 Publisher: Multidisciplinary Digital Publishing Institute.

[15] Android Developers. 2023. Design for TV. https://developer.android.com/design/ui/tv/guides/foundations/design-for-tv

[16] Android Developers. 2024. Design Principles - Wear. https://developer.android.com/design/ui/wear/guides/foundations/design-principles

[17] Tafadzwa Joseph Dube and Ahmed Sabbir Arif. 2019. Text Entry in Virtual Reality: A Comprehensive Review of the Literature. In *Human-Computer Interaction. Recognition and Interaction Technologies (Lecture Notes in Computer Science)*, Masaaki Kurosu (Ed.). Springer International Publishing, Cham, 419–437. https://doi.org/10.1007/978-3-030-22643-5_33

[18] Mark D. Dunlop, Andreas Komninos, and Naveen Durga. 2014. Towards High Quality Text Entry on Smartwatches. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. Association for Computing Machinery, Toronto, Ontario, Canada, 2365–2370. https://doi.org/10.1145/2559206.2581319

[19] Brien East, Sean DeLong, Roozbeh Manshaei, Ahmed Arif, and Ali Mazalek. 2016. Actibles: Open Source Active Tangibles. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces - ISS '16*. ACM Press, Niagara Falls, Ontario, Canada, 469–472. https://doi.org/10.1145/2992154.2996874

[20] EowynREN. 2023. EowynREN/androidtv-remote. https://github.com/EowynREN/androidtv-remote original-date: 2023-10-21T07:38:32Z.

[21] Robert Eriksson and Filip Sjogren. 2007. *Enhancing the user experience with new interaction techniques for interactive television.* Master's thesis. Department of Computer Science, Umeå University, Umeå, Sweden.

[22] International Center for Language Studies. 2023. The Interagency Language Roundtable (ILR) Proficiency. https://www.icls.edu/foreign-language-programs/ilr-proficiency-levels

[23] Emma Forsling Parborg. 2017. *Towards Replacing the Remote Control with Commodity Smart-Phones Through Evaluation of Interaction Techniques Enabling Television Service Navigation.* Ph. D. Dissertation. Department of Science and Technology, Linköping University, Norrköping, Sweden. https://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-144581

[24] GSMArena. 2024. Google Pixel Watch 3 - Full Phone Specifications. https://www.gsmarena.com/google_pixel_watch_3-13253.php

[25] Google Pixel Watch Help. 2024. Control Your TV with Your Watch. https://support.google.com/googlepixelwatch/answer/15294495

[26] Jonggi Hong, Seongkook Heo, Poika Isokoski, and Geehyuk Lee. 2015. SplitBoard: A Simple Split Soft Keyboard for Wristwatch-Sized Touch Screens. In *Proceedings*

of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 1233–1236. https://doi.org/10.1145/2702123.2702273

[27] Business Research Insights. 2024. Remote Control Systems & Kits Market Size, Share 2031, Report. https://www.businessresearchinsights.com/market-reports/remote-control-systems-kits-market-100201

[28] Sue Jackson. 2014. Flow Scales. In *Encyclopedia of Quality of Life and Well-Being Research*, Alex C. Michalos (Ed.). Springer Netherlands, Dordrecht, 2305–2308. https://doi.org/10.1007/978-94-007-0753-5_1065

[29] Susan A. Jackson and Mihaly Csikszentmihalyi. 1999. *Flow in Sports*. Human Kinetics, Champaign, IL, USA. Google-Books-ID: Jak4A8rEZawC.

[30] Cosette Jarrett. 2022. Who's That Actor? What's That Song? Prime Video's Interactive X-Ray Feature Has the Answer. https://www.aboutamazon.com/news/entertainment/what-is-x-ray-on-prime-video

[31] Sung-Woo Kim. 2012. An Outlook for Interaction Experience in Next-generation Television. *Journal of the Ergonomics Society of Korea* 31, 4 (2012), 557–565. https://doi.org/10.5143/JESK.2012.31.4.557 Publisher: Ergonomics Society of Korea.

[32] Andreas Komninos and Mark Dunlop. 2014. Text Input on a Smart Watch. *IEEE Pervasive Computing* 13, 4 (Oct. 2014), 50–58. https://doi.org/10.1109/MPRV.2014.77

[33] Federica Laricchia. 2024. Smart and Connected TVs - Statistics & Facts. https://www.statista.com/topics/4761/smart-and-connected-tvs/

[34] Federica Laricchia. 2024. TV Technology - Statistics & Facts. https://www.statista.com/topics/6187/tv-technology/

[35] Robert Edward Lewand. 2000. *Cryptological Mathematics*. Vol. 16. Mathematical Association of America, Washington, D.C., USA.

[36] Mateus M. Luna, Thyago P. Carvalho, Fabrizzio Alphonsus A. M. N. Soares, Hugo A. D. Nascimento, and Ronaldo M. Costa. 2017. Wrist Player: A Smartwatch Gesture Controller for Smart TVs. In *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, Vol. 2. IEEE, Turin, Italy, 336–341. https://doi.org/10.1109/COMPSAC.2017.266 ISSN: 0730-3157.

[37] Sabina Mammadzada. 2023. A Review of Existing Transliteration Approaches and Methods. *International Journal of Multilingualism* 20, 3 (July 2023), 1052–1066. https://doi.org/10.1080/14790718.2021.1958821 Publisher: Routledge _eprint: https://doi.org/10.1080/14790718.2021.1958821.

[38] Andrew J. Martin and Susan A. Jackson. 2008. Brief Approaches to Assessing Task Absorption and Enhanced Subjective Experience: Examining 'Short' and 'Core' Flow in Diverse Performance Domains. *Motivation and Emotion* 32, 3 (Sept. 2008), 141–157. https://doi.org/10.1007/s11031-008-9094-0

[39] Mark McGill, John H. Williamson, and Stephen Brewster. 2016. Examining The Role of Smart TVs and VR HMDs in Synchronous At-a-Distance Media Consumption. *ACM Trans. Comput.-Hum. Interact.* 23, 5 (Nov. 2016), 33:1–33:57. https://doi.org/10.1145/2983530

[40] Prenisha Naidoo. 2023. The Rise of Interactive TV. https://homelinkd.com/blog/the-rise-of-interactive-tv-next-gen-viewing-experiences,https://homelinkd.com/blog/the-rise-of-interactive-tv-next-gen-viewing-experiences

[41] Thamer Horbylon Nascimento, Fabrizzio Alphonsus A. M. N. Soares, Hugo A. D. Nascimento, Marcos Alves Vieira, Thyago Peres Carvalho, and Wesley Flávio de Miranda. 2019. Netflix Control Method Using Smartwatches and Continuous Gesture Recognition. In *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*. IEEE, Edmonton, AB, Canada, 1–4. https://doi.org/10.1109/CCECE.2019.8861610 ISSN: 2576-7046.

[42] Netflix. 2024. Using Keyboard Shortcuts on Netflix. https://help.netflix.com/en/node/24855

[43] The Numbers. 2024. Genres Movie Breakdown 1995-2024. https://www.the-numbers.com/market/genres

[44] Jonathan O'Callaghan. 2014. Microsoft Develops Handwriting Tool for Android Smart Watches. https://www.dailymail.co.uk/sciencetech/article-2791212/the-smartwatch-lets-scribble-messages-microsoft-develops-handwriting-tool-android-wrist-tech.html Section: Science.

[45] 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. https://doi.org/10.1145/2470654.2481387

[46] Origin. 2023. A Concise History of the Birth and Ascent of Smart (connected) TV - Everything You Don't Know. https://www.corp.originmedia.tv/ctvinsider/a-concise-history-of-the-birth-and-ascent-of-smart-connected-tv-everything-to-know

[47] Laxmi Pandey, Khalad Hasan, and Ahmed Sabbir Arif. 2021. Acceptability of Speech and Silent Speech Input Methods in Private and Public. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '21)*. ACM, New York, NY, USA, Yokohama, Japan, 13 pages. https://doi.org/10.1145/3411764.3445430

[48] Irina Popovici, Ovidiu-Andrei Schipor, and Radu-Daniel Vatavu. 2019. Hover: Exploring Cognitive Maps and Mid-Air Pointing for Television Control. *International Journal of Human-Computer Studies* 129 (Sept. 2019), 95–107. https://doi.org/10.1016/j.ijhcs.2019.03.012

[49] Irina Popovici, Radu-Daniel Vatavu, and Wenjun Wu. 2019. TV Channels in Your Pocket! Linking Smart Pockets to Smart TVs. In *Proceedings of the 2019 ACM International Conference on Interactive Experiences for TV and Online Video (TVX '19)*. Association for Computing Machinery, New York, NY, USA, 193–198. https://doi.org/10.1145/3317697.3325119

[50] Gulnar Rakhmetulla and Ahmed Sabbir Arif. 2021. SwipeRing: Gesture Typing on Smartwatches Using a Segmented QWERTY Around the Bezel. In *Proceedings of Graphics Interface 2021 (GI 2021)*. Canadian Information Processing Society, Toronto, ON, Canada, 166 – 177. https://doi.org/10.20380/GI2021.19 ISSN: 0713-5424 event-place: Virtual Event.

[51] Gulnar Rakhmetulla and Ahmed Sabbir Arif. 2023. Crownboard: A One-Finger Crown-Based Smartwatch Keyboard for Users with Limited Dexterity. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3544548.3580770

[52] Gulnar Rakhmetulla, Yuan Ren, and Ahmed Sabbir Arif. 2023. GeShort: One-Handed Mobile Text Editing and Formatting with Gestural Shortcuts and a Floating Clipboard. *Proceedings of the ACM on Human-Computer Interaction* 7, MHCI (Sept. 2023), 212:1–212:23. https://doi.org/10.1145/3604259

[53] Yuan Ren and Ahmed Sabbir Arif. 2021. Stepper, Swipe, Tilt, Force: Comparative Evaluation of Four Number Pickers for Smartwatches. *Proc. ACM Hum.-Comput. Interact.* 5, ISS (Nov. 2021), 500:1–500:21. https://doi.org/10.1145/3488545

[54] Grand View Research. 2023. *Smart Remote Market Size, Share & Trends Analysis Report By Application (Smart TV, Other), By Distribution Channel (Offline, Online), By Type, By Region, And Segment Forecasts, 2023 - 2030*. Technical Report GVR-4-68039-933-3. Grand View Research, Inc. 100 pages. https://www.grandviewresearch.com/industry-analysis/smart-remote-market-report

[55] Jonas Roberts, Johannes Wölfl, Christoph Märkl, Patricia Böhm, and Daniel Isemann. 2017. Comparing Smartwatch Input Modalities for Older Users in a TVControl Scenario. In *Mensch und Computer 2017 - Workshopband*. Gesellschaft für Informatik e.V., Regensburg, Germany, 10.18420/muc2017. https://dl.gi.de/handle/20.500.12116/3240

[56] Nikki Rothberg and Clare Varellas. 2024. watchOS 11 Is Available Today. https://www.apple.com/newsroom/2024/09/watchos-11-is-available-today/

[57] Rita Santos, Jorge Abreu, Pedro Beça, Ana Rodrigues, and Sílvia Fernandes. 2020. Voice Interaction on TV: Analysis of Natural Language Interaction Models and Recommendations for Voice User Interfaces. *Multimedia Tools and Applications* 79, 47 (Dec. 2020), 35689–35716. https://doi.org/10.1007/s11042-020-08710-2

[58] Rita Santos, Joana Beja, Mário Rodrigues, and Ciro Martins. 2019. Designing Visual Interfaces to Support Voice Input: The case of a TV application to request help in daily life tasks. In *Proceedings of the XX International Conference on Human Computer Interaction (Interacción '19)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3335595.3335637

[59] Vinod Keshav Seetharamu, Joy Bose, Sowmya Sunkara, and Nitesh Tigga. 2014. TV Remote Control Via Wearable Smart Watch Device. In *2014 Annual IEEE India Conference (INDICON)*. IEEE, Pune, India, 1–6. https://doi.org/10.1109/INDICON.2014.7030602 ISSN: 2325-9418.

[60] Tomoki Shibata, Daniel Afergan, Danielle Kong, Beste F. Yuksel, I. Scott MacKenzie, and Robert J.K. Jacob. 2016. DriftBoard: A Panning-Based Text Entry Technique for Ultra-Small Touchscreens. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 575–582. https://doi.org/10.1145/2984511.2984591

[61] Telmo Silva, Pedro Almeida, Jorge Abreu, and Eliza Oliveira. 2018. Interaction Paradigms on iTV: A Survey Towards the Future of Television. In *Proceedings of the 9th International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC '18)*. International Institute of Informatics and Systemics (IIIS), Orlando, Florida, USA, 18–23.

[62] R. William Soukoreff and I. Scott MacKenzie. 2003. Metrics for Text Entry Research: An Evaluation of MSD and KSPC, and a New Unified Error Metric. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, New York, NY, USA, 113–120. https://doi.org/10.1145/642611.642632 event-place: Ft. Lauderdale, Florida, USA.

[63] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O. Wobbrock. 2018. $Q: A Super-Quick, Articulation-Invariant Stroke-Gesture Recognizer for Low-Resource Devices. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, Barcelona Spain, 1–12. https://doi.org/10.1145/3229434.3229465

[64] Radu-Daniel Vatavu, Pejman Saeghe, Teresa Chambel, Vinoba Vinayagamoorthy, and Marian F Ursu. 2020. Conceptualizing Augmented Reality Television for the Living Room. In *Proceedings of the 2020 ACM International Conference on Interactive Media Experiences (IMX '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3391614.3393660

[65] David Verweij, Vassilis-Javed Khan, Augusto Esteves, and Saskia Bakker. 2017. Multi-User Motion Matching Interaction for Interactive Television using Smartwatches. In *Adjunct Publication of the 2017 ACM International Conference on Interactive Experiences for TV and Online Video (TVX '17 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 67–68. https://doi.org/10.1145/

3084289.3089906

[66] Huiyue Wu, Liuqingqing Yang, Shengqian Fu, and Xiaolong (Luke) Zhang. 2019. Beyond Remote Control: Exploring Natural Gesture Inputs for Smart TV Systems. *Journal of Ambient Intelligence and Smart Environments* 11, 4 (Jan. 2019), 335–354. https://doi.org/10.3233/AIS-190528 Publisher: IOS Press.

[67] Ionuţ-Alexandru Zaiţi, Ştefan-Gheorghe Pentiuc, and Radu-Daniel Vatavu. 2015. On Free-Hand TV Control: Experimental Results on User-Elicited Gestures with Leap Motion. *Personal and Ubiquitous Computing* 19, 5 (Aug. 2015), 821–838. https://doi.org/10.1007/s00779-015-0863-y

A Questionnaires

A.1 Usability Questionnaire

The usability questionnaire asked participants to rate the following five statements on a 5-point scale.

(1) The method is fast.
(2) The method is both accurate and precise.
(3) The method is easy to learn and understand.
(4) The method requires low physical and mental effort to use.
(5) I would use this technique on my device.

A.2 Core Flow Scale

The Core Flow questionnaire [38] asked participants to rate the following statements on a 7-point scale.

(1) I am “totally involved.”
(2) It feels like “everything clicks.”
(3) I am “tuned in” to what I am doing.
(4) I am “in the zone.”
(5) I feel “in control.”
(6) I am “switched on.”
(7) It feels like I am “in the flow” of things.
(8) It feels like “nothing else matters.”
(9) I am “in the groove.”
(10) I am “totally focused” on what I am doing.