Expanding Touch Interaction Capabilities for Smart-rings: An Exploration of Continual Slide and Microroll Gestures

Anuradha Herath*
caherath@student.ubc.ca
University of British Columbia
Kelowna, British Columbia, Canada

Bradley Rey
reyb@myumanitoba.ca
University of Manitoba
Winnipeg, Manitoba, Canada

Sandra Bardot
sandra.bardot@umanitoba.ca
University of Manitoba
Winnipeg, Manitoba, Canada

Sawyer Rempel
rempels8@myumanitoba.ca
University of Manitoba
Winnipeg, Manitoba, Canada

Lucas Audette
audettel@myumanitoba.ca
University of Manitoba
Winnipeg, Manitoba, Canada

Huizhe Zheng
huizhezheng@myumanitoba.ca
University of Manitoba
Winnipeg, Manitoba, Canada

Jun Li
jun.li3@huawei.com
Huawei HMI Lab
Toronto, Ontario, Canada

Kevin Fan
szu.wen.fan@huawei.com
Huawei HMI Lab
Toronto, Ontario, Canada

Da-Yuan Huang
dayuan.huang@huawei.com
Huawei HMI Lab
Toronto, Ontario, Canada

Wei Li
wei.li.crc@huawei.com
Huawei HMI Lab
Toronto, Ontario, Canada

Pourang Irani
pourang.irani@ubc.ca
University of British Columbia
Kelowna, British Columbia, Canada

ABSTRACT

As smart-rings emerge in both research and commercial markets, their limited physical size remains to restrict the interaction potential and input vocabulary possible. Thus, focusing on touch interaction for its natural and preferred input potential, this early work explores the combination of slide and microroll gestures performed by the thumb in continual motion on a smart-ring's touch capacitive surface. We first capture over 3000 slide and microroll gesture instances, extract features, and generate and test machine learning models that are able to discern the slide and microroll gestures within the same touch instance. Through the use of 18 features, our Random Forest model provides a 92.4% accuracy. We conclude with demonstrations of potential applications utilizing continual slide and microroll gestures, and a short discussion which provides future research directions stemming from the positive results obtained from this preliminary work.

CCS CONCEPTS

• **Human-centered computing → Gestural input**; **Touch screens**; **Mobile devices**.

KEYWORDS

smart ring, touch interaction, thumb interaction, one-handed, microroll, rolling/sliding gestures

ACM Reference Format:
Anuradha Herath, Bradley Rey, Sandra Bardot, Sawyer Rempel, Lucas Audette, Huizhe Zheng, Jun Li, Kevin Fan, Da-Yuan Huang, Wei Li, and Pourang Irani. 2022. Expanding Touch Interaction Capabilities for Smart-rings: An Exploration of Continual Slide and Microroll Gestures. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '22 Extended Abstracts), April 29-May 5, 2022, New Orleans, LA, USA*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3491101.3519714

1 INTRODUCTION

Smart-rings are becoming increasingly viable, through the miniaturization of hardware and components needed, to provide interaction for many mobile devices. Furthermore, smart-rings afford users multiple interaction modalities [36], such as touch, gesture, and proximal interaction, and provide a means for always-available input [3, 4, 20, 40, 48]. However, as seen in previous devices, when reducing the physical size, interaction capabilities are often reduced [5, 32, 37]. With a diminished interaction space on a smart-ring, due to the physical size, novel interaction techniques must be explored to fully take advantage of the hardware available and to increase the interaction bandwidth.

While smart-rings, with appropriate sensors included, provide the ability for multiple interaction modalities, we must recognize that we live in "a touch input world" [16]. Touch as in input modality provides users a preferred, natural, and at times even discrete form of interaction [3, 16, 31]. Research has focused on expanding touch interaction capabilities for devices such as smartphones and smartwatches. This includes using the bezels [24, 27, 35, 47], pressure

*Anuradha Herath conducted this work while with the University of Manitoba.

CHI '22 Extended Abstracts, April 29-May 5, 2022, New Orleans, LA, USA

ACM ISBN 978-1-4503-9156-6/22/04…$15.00
https://doi.org/10.1145/3491101.3519714

sensing [17, 28], touch-based haptic interactions [34], multi-modal input with common touch gestures [19, 22, 43], and rolling gestures [6, 9, 32]. As smart-rings are yet an increasingly miniature device, further exploration is needed for smart-rings to expand their capacity for potential touch interactions.

Thus, our motivation for this work focuses on expanding touch interaction vocabulary and capability for small hardware, specifically that of a smart-ring. Current smart-ring devices, if allowing for touch interaction, often solely utilize taps and swipes/slides [4, 15, 21][1]. We aim to extend on this common interaction technique through the incorporation of microroll gestures [32]. Microroll gestures provide us with a potential means of interaction that either allows expanded input expressivity over that of discrete, and limited, pressure based interaction [11, 25, 26, 30], or the ability to be conducted anywhere on the touch surface, unlike bezel-based interaction.

Throughout, we focus on a novel implementation allowing for continual slide and microroll gesturing on a smart-ring touch capacitive surface. Thus, increasing the interactive touch capability from simple slide gestures alone. We extend and differentiate on the work previously done by Roudaut et al. [32], through the continual nature of the slide and microroll gestures combined within the same touch instance. In this early work, our contributions are two-fold: **C1:** A machine learning model that discerns slide and microroll gestures that take place in the same touch instance. We achieve a model accuracy of 92.4% using a Random Forest model with 18 features; **C2:** Proposed applications that can benefit from the increased touch interaction space allowed by real-time discerning of slide and microroll gestures. Combined, and through final discussion, we show the potential for further expanding touch interaction capabilities on small hardware such as smart-rings. We also hope our work motivates future research and discussion in the expansion of touch interaction capabilities for miniature smart-ring devices, allowing for them to become increasingly viable devices we can utilize across our daily lives.

2 RELATED WORK

2.1 Finger Based Micro Interactions

Many forms of finger based micro interactions on smart-rings have been studied in previous literature. At the base of these interaction techniques lies classic thumb to smart-ring touch interaction [4, 8, 21]. These often incorporate simple slide/swipe, tap, and/or simple gesture recognition as input. As well, other more complex touch interaction methods have also been explored and are discussed below.

Area and Location Based Touch Interactions. A method of expanding touch interaction using area of the input device was presented by Boring et al. [10], where they introduced a novel one-handed interaction technique called the *"The Fat Thumb"*. Here, the contact size of the thumb was used to simulate pressure augmenting interactions such as panning and zooming within a single touch motion. A similar method was used in *TouchSense* [18] where different areas of the pad of the finger acted as input for smaller touch screen devices. The different touch areas of the finger pad were distinguished through an IMU device attached to the fingernail of the index finger.

Utilizing different contact locations of a finger have been utilized for text entry methods [23, 46]. The inner-segments of the fingers, where thumb-to-finger touching is possible, was used in *FingerT9* [46]. An experimental prototype with force sensors attached to finger segments was used to distinguish different segments and the T9 input was used for the key to finger-segment mapping. In *FingerText* [23], the use of finger nails as a design space for one-handed text input was explored by placing touch sensors on the nails of the fingers.

Rolling Based Interactions. One of the earliest works on finger microrolling was explored by Benko et al. as a clicking operation for multi-touch displays [6]. Within this work, an interaction technique proposed, *SimPress*, used the changes in the contact area of the finger with the touch display to simulate pressure to distinguish clicking actions. Furthermore, Bonnet et al. [9] explored rocking of the thumb (rolling the thumb back and forth on the touch surface) as an alternative to tap and long press interactions. They developed an algorithm to track both the trajectory, relative variation of the direction, and area touched by the thumb when performing the rocking gesture. The rocking was successfully applied, through experimentation, for both discrete and continuous interaction.

The most detailed work on finger microrolling interaction was carried out by Roudaut et al. [32] in the paper titled *MicroRolls*. In their work, they studied the effectiveness of 6 types of discrete thumb microrolling gestures together with slides, swipes and rubbing gestures. With an automatic gesture recognizer, which was based on a K-Nearest Neighbours model, and using 10 of Rubine's features [33] extracted from the recorded gestures they managed to obtain an overall recognition rate of 95.3% and a microroll recognition rate of 96%. As this study showed promising results, we look to extend on this research through two distinguishable differences. First, we are mainly focusing on live detection of continuous and combined gestures of slides and microrolls rather than discrete instances of each. Second, our interest lies in discerning slides and microrolls on miniature touch surfaces for micro devices, such as smart-rings, where the touch capacity has a lower fidelity to that of more advanced displays used on smartwatches and smartphones.

2.2 Stroke Based Gesture Recognition and Feature Extraction

Feature extraction, based on stroke paths, for gesture recognition has been heavily used in fields like optical character [1, 14, 29] and hand gesture [38, 49] recognition. Apart from these, research has also utilized machine learning and feature extraction for stroke path input on a range of mobile devices [32, 47], similar to our goals for this work. In this section we describe and discuss feature extraction algorithms that are more commonly adapted for use on machine learning models.

One of the most frequently used family of recognizers for stroke gesture recognition is the $-family recognizers [2, 41, 42, 45]. The availability of the pseudocode and implementations in many programming languages and the fast recognition with low amount of resource utilization are some positive factors that has resulted in their frequent use. Alternatively, gesture recognition can be done

[1]Additionally, the FinchRing is a consumer smart-ring allowing for tap and swipe gestures: https://www.finch-xr.com/ring/

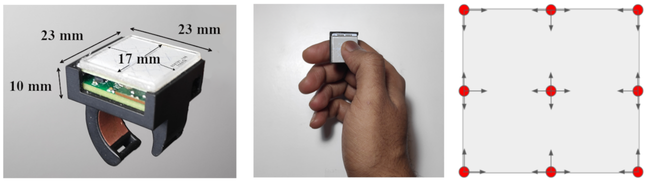

Figure 1: Left: Dimensions of the smart-ring; Middle: Scale relative to the hand while wearing the smart-ring; Right: The 9 touch surface locations and the corresponding 24 directions explored in the study.

through feature extraction and machine learning models. Rubine's features [33], and an extended set of Rubine like features that are used for sketch recognition, can be easily used for stroke gesture recognition [7, 32]. These features which are mainly based on geometry, pressure, and motion are popular due to the ease of extraction without causing any additional stress on the system resources. Early testing, proved the $-family recognizers were not complex enough, even though they provide a more lightweight approach, to distinguish between slides and microrolls. Thus, we opt for a machine learning and feature extraction approach that is further highlighted below and similarly used in previous works [32, 47].

3 SLIDES AND MICROROLLS: DISTINGUISHING THE INTERACTIONS IN CONTINUAL MOTION

Our main focus in this early work is to enable location independent continuous slide and microroll gesture interactions on micro touch surfaces. Thus, a user can slide their thumb around the touch surface, then without lifting their thumb, microroll in any direction, then switching back to sliding, and continually repeating the interaction from there. The interaction can also be performed with microrolling occurring first and the same gesture being performed back-to-back. To accomplish this, we used a smart-ring designed and built by our research team, collected input data, and generated and evaluated models, all of which are described below.

3.1 The Smart-Ring

Custom ring hardware was designed and assembled in-house. The electronics are built around a NINA-B306 NRF52840 Bluetooth Low Energy module, this unit is responsible for interfacing with the on-board sensors. For sensing capabilities, a BMI160 6-axis axis Inertial Measurement Unit enables orientation detection and an IQS572 capacitive touch driver is used for driving the trackpad display board, which stacks on-top of the main PCB using a header connector, providing a touch interaction space of approximately 17x17 mm. A curved 19mAh Lithium Polymer battery provides power, and the on-board power management circuitry handles power regulation and enables charging through the on-board micro USB port. All the electronics are then housed in an ergonomically shaped SLA 3D printed enclosure, with the curved battery sitting inside the band to minimize the overall form-factor. The final design of the ring and the scale relative to the hand can be seen in Figure 1.

3.2 Data Collection

To capture and understand the gestures being performed across the touch surface, we designed an experiment to collect interaction data from volunteers. This data allowed us to create and test machine learning models for which we utilize to allow continuous slide and microroll gesturing.

Volunteers. Seven volunteers (6 men, 1 woman) aged between 25-30 participated in the data collection. Volunteers had prior experience interacting with smartwatches, however had limited experience with a smart-ring.

Apparatus. The application for data collection was developed in Java for Android using Android Studio. The experiment and all the developed applications were run on a Samsung Galaxy S8 running Android version 9 (API level 28) together with the smart-ring connected via Bluetooth for obtaining the input.

Design. The volunteers were required to participate in 10 sessions performing individual slides and microrolls. Each session consisted of slides and microrolls in all horizontal and vertical directions possible from nine locations of the smart-ring (4 directions x 1 center + 3 directions x 4 edges + 2 directions x 4 corners); see Figure 1. This yields 24 slide and 24 microroll gestures for each session, contributing to a total of 480 gestures per volunteer across the 10 sessions, and a total of 3360 (1680 slides + 1680 microrolls) gestures from all the volunteers.

Task. First, volunteers were given instructions and allowed to ask questions. When ready to begin the data collection, volunteers were then instructed to place the ring on the second phalanx of their index finger on their dominant hand, enabling effective thumb-to-smart-ring touch interactions; for all volunteers the smart-ring fit snug. The task was carried out while seated [32], with their interacting hand resting on a table. During the sessions, the gesture to complete was displayed on a connected smartphone. The locations, directions and gesture types (slide, microroll) were randomized in every session. Volunteers were advised to complete the gesture in a manner that was as comfortable as possible and natural to them. A gesture was considered complete when the connection between the thumb and touch surface was released (i.e, when the finger was raised from the touch surface), and would trigger the next gesture to be performed to be displayed to the volunteer. If a wrong gesture was completed (e.g., a slide when a microroll was asked for) the volunteer would redo the trial. A session lasted on average roughly four minutes, with volunteers being asked to rest for at least one minute after each session. On average each volunteer spent roughly an hour completing the data collection.

3.3 Data Processing

Once all the data had been collected, we removed the first session as it was considered training in order to minimize errors found while volunteers were learning to perform the gestures. We then concatenated all the captured gestures into a final single series of connected and continual gestures, taken at random, with the restriction that the locations of the end of the current motion and start of the next motion were within a distance of 0.15 units (the full horizontal and vertical distance of the trackpad is 1.0 units). We opted for the collection of separated gestures, concatenating them during data processing for three reasons: First, in order to

Table 1: Details of the top 5 configurations of the top 3 models: t = number of tree nodes, K = neighbours, i = maximum considered iterations, n = number of nodes in the hidden layers, l = number of hidden layers.

	Random Forest		**K Nearest Neighbours**		**Multi-Layer Perceptron**	
	accuracy %	details	accuracy %	details	accuracy %	details
1	92.4	t = 500	82.0	K = 1	80.3	i = 700, l = 2, n = (300, 300)
2	92.4	t = 700	79.5	K = 3	79.2	i = 500, l = 2, n = (300, 300)
3	92.4	t = 900	78.8	K = 5	78.9	i = 700, l = 2, n = (200, 200)
4	92.2	t = 300	78.4	K = 2	78.8	i = 700, l = 2, n = (500, 500)
5	91.9	t = 100	77.9	K = 7	78.6	i = 500, l = 2, n = (500, 500)

capture all possible combinations of combined gestures from all locations, the number of trials and the length of each trial would be significantly larger than the method utilized. This would be infeasible for volunteers and paid participants to complete and would make error correction more time consuming. Second, as a byproduct of the first issue, lengthy trials can begin to affect the interaction characteristics (such as pauses and/or slower gesturing) of the gesture being completed. Third, while we only created and utilized one series of combined gestures, capturing the gestures in a separated manner allows for easy creation of multiple new and unique gesture series, thus providing an increased amount of unique data for model training.

The complete dataset containing the series of connected and continual gestures was then used for extracting features using a sliding window approach [13]; through testing, a window of 25 samples and a slide of 5 samples was used. The number of samples per window was determined by the minimum samples for a slide or a microroll motion that allowed for high accuracy while also taking into account natural gesture speed seen within the data. The gesture within each window is determined by the majority of points of a gesture type from samples within a window. As a microroll has more touch points captured, due to a longer gesture duration, there are inherently more samples for a microroll than a slide. Thus, to remove the bias that was added due to the sliding window approach, we balanced the dataset using SMOTE [12] with minority re-sampling. After, and in total, we had 91,804 extracted feature samples that we used for model generation and testing.

Following a similar approach by Roudaut et al., we extracted and used 18 features (from each window) of the 114 features described by [7]. These 18 features[2] are: *mean pressure per unit length, diagonal length of the bounding box of all sample points in the window, average x distance between touch points, average y distance between touch points, mean pressure, average speed, total number of samples with pressure lower than the mean pressure, time, mean acceleration in x, total gesture length, mean acceleration in y, density 2 (absolute length/area of the bounding box), maximum pressure, minimum pressure, mean force, density 1 (absolute length of the stroke/distance from first point to the last point), area of the bounding box, and ratio of diagonal length to the area.*

[2]These features are ordered from most to least important (using permutation feature importance) as they pertain to the RF model.

3.4 Model Generation and Evaluation

To distinguish between the two gestures in real-time we developed four machine learning models, using binary classification, and trained them with the features extracted from our gesture dataset. The models used were, Support Vector Machine (SVM), Random Forest (RF), K-Nearest Neighbour (K-NN) and a Multi-Layer Perceptron (MLP). Binary classification was used, as directionality could be discerned through simply analyzing the difference between touch points as well as to minimize complexity of the models.

From the entire processed dataset, we split the data in a stratified fashion, using 80% of all samples for training and the remaining 20% for model testing. Of the four models, the SVM model was trained with the radial basis function kernel [39]. The RF model and K-NN model were trained and tested multiple times with adjusted parameters (number of trees for RF and number of neighbours for K-NN). The values used were 100, 300, 500, 700 and 900 for RF and values from 1 to 20 for K-NN). For the MLP model we used the same set of iterations as above and for each iteration we used up to three hidden layers starting from 100 nodes in each layer.

From the four models used, the highest accuracy (92.4%) was produced by the RF model with its number of trees set to 500 followed by the K-NN model (82% with K=1) and MLP model with an accuracy of 80% (700 iterations with two hidden layers of 300 nodes). The least accurate model was the SVM with 68% accuracy. A detailed description of the top 5 accuracy scores for the first 3 models is given in Table 1. To further confirm that the accuracy of our RF model was independent of the testing dataset we carried out a 5-fold stratified cross validation on the entire randomized dataset. The average accuracy of the cross validation results were found to be 90.6% which further clarified our findings.

4 SAMPLE APPLICATIONS

Having produced a model with high accuracy to be able to discern continual slide and microroll gesturing, through the following two applications we demonstrate the potential of continual slide and microroll gestures using a smart-ring. Our applications were built for a Samsung Galaxy S8 running Android. To avoid memory limitation issues we hosted our model on a local Python server with an API endpoint for touch data and model prediction communication. Demo clips can be viewed here.

4.1 Mobile Gaming

Due to the nature of mobile gaming, typical physical joystick controllers are not present. This leads to the the inclusion of touch

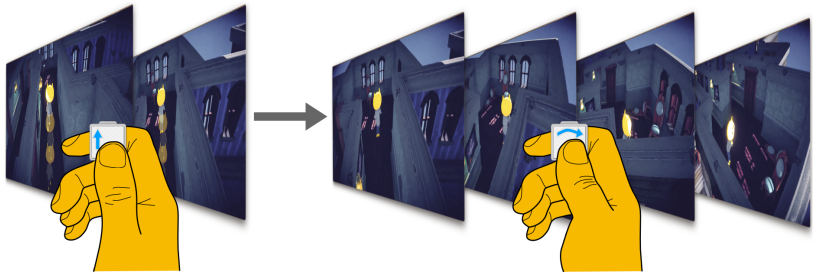

Figure 2: Game application using continual slide and microroll gestures. First, the user slides up to have the character move up through the hall (frames left of the grey arrow and faded character depict this movement). Then, the user microrolls to the right to have the camera circle around the character (frames right of the grey arrow) before being able to move back to a slide to keep the character walking.

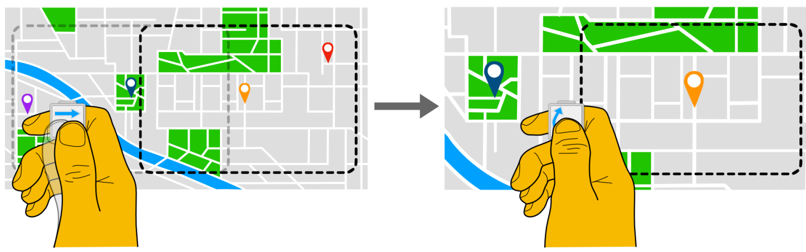

Figure 3: Map application using continual slide and microroll gestures. Here, the user can pan around the map using slide gestures (image left depicts sliding to the right using a slide to the right). When needed, the user can zoom-in or -out using up and down microrolling gestures respectively (image right depicts zooming-in through an upward microroll).

screen joysticks. Typically, a game may require one joystick for character movement and another for camera movement. Through the use of slide and microroll gestures, we can incorporate these two interactions within the same smart-ring. As seen in Figure 2, the character can be moved throughout the game environment using slide gestures, as one typically would on a touch screen joystick. When needed, the user can microroll their thumb to adjust the camera, before changing back to a slide to keep the character moving. This allows for a very fluid interaction, utilizing a single smart-ring. Additionally, the microrolling gesture could be implemented to move the character at a faster pace, or allow for quick item selection throughout game play.

4.2 Map Navigation

When searching for a location on a map, the entire interaction is often a set of discrete slide and pinch gestures. Here we propose utilizing the continual slide and microroll gestures implemented in our work to perform the panning and zooming interactions respectively. As demonstrated in Figure 3, a user can slide on the smart-ring to pan around the map. They can then perform an upwards microroll gesture to zoom-in, or conversely a downward microroll gesture to zoom-out. The user can then, as before, move to a sliding gesture to continue panning. This continuous interaction between slide and microroll gestures mimics the discrete interactions typically taken place. Additionally, with the limited touch area on a smart-ring, the incorporation of both slide and microroll gestures allows the interaction to remain fluid, without the need for mode switching through a separate gesture.

5 DISCUSSION

As seen in many previous works [5, 32, 37], limitations in interaction occur as devices become smaller. Novel methods of interaction can enable an increased input vocabulary, something that is necessary so that the device can perform functions as needed. While devices such as smart-rings are becoming more feasible, their size greatly effects the interactions that we can utilize. Thus, we propose that microrolling can be an optimal gesture of choice to further expand smart-ring touch interaction due to its limited need for touch area and the potential for combination with an already familiar sliding gesture. Our early results, with a 92.4% model accuracy, suggest that combined slide and microroll gestures have the ability to be discerned within a single touch instance. Thus, in doubling the interaction vocabulary, not only can we discern independent interactions, but we can reduce the need to clutch by allowing the gestures to occur in the same touch instance.

As our work extends that of the research done by Roudaut et al. [32], we note both similarities and differences regarding our implementation. First, while we did not capture quantitative data for performance metrics and qualitative comments from volunteers, we note that the work done Roudaut et al. suggests that microrolls have the potential to outperform other menuing techniques and while not statistically significant, microrolls were rated as the interaction technique of choice. Second, model results were relatively similar, with Roudaut et al. achieving 96.1% accuracy for microrolls and 95.1% accuracy for slides. Notably, in their work, slides and microrolls caused the most error when discerning the two gestures. Due to the many similarities in the gesture, we attempted to correct for this error in recognition through an increased feature set, primarily through the use of pressure detection. Finally, as the smart-ring has a smaller touch area and resolution than that used by Roudaut et al., the model holds up well for our goal of achieving the interaction on a micro-device, justifying its use in future smart-ring iterations and interaction use cases.

From the models that we had trained and tested it was evident that when the length of the sliding window was increased we could obtain a better model with higher accuracy. However, we found that when predicting gestures in real time, the gesture had to be performed very slowly in order to obtain the necessary amount of data samples to make a prediction (i.e., to fulfill the size of the sliding window). It should also be noted that through our applications we detected a slight increase in the number of correct sliding gesture predictions when the sliding gesture occurred across the length of the touch surface. Thus, we believe that while our model was able to discern the two gestures with high accuracy, one of the defining features of the two gestures are their overall length and speed of the interaction (as touch points are recorded further apart with greater speed) used for interaction.

There are limitations in both the hardware and model generation methods to note. Through iteration, we would aim to correct for these in future works. First, the Bluetooth module used within the smart-ring prototype has limited transmission rates. Thus, this limited transmission could affect both the model training as well as real-time gesture predictions in applications. Secondly, our current model utilization is limited to android based mobile devices. The lack of memory to embed the model within our applications leads us to making predictions remotely through an API based, locally hosted, web service. The latency issues due to data transmission speeds with such a system could also affect the gesture prediction speeds of our applications. This is especially a concern in scenarios like a first person shooter game where reaction speed is a crucial factor. For applications, such as map navigation and search, where a slight lag between when the gesture and the required action is permitted, more complex models can be utilized to provide predictions with better accuracy. Furthermore, with the rapid increase in capable and miniaturized hardware, these increasingly smaller and more powerful chips will quickly mitigate this issue in the near future.

In future, we aim to further explore the potential that slide and microroll gestures allow, especially as they pertain to small form factors such as smart-ring devices. While in this work we only explore four directions, we hope to increase the number of allowable interactive directions to eight. Through a larger study, with an expanded set of participants and trials, and updated hardware, we believe we can improve upon the model's performance. Furthermore, we believe the potential for use is not limited to the applications shown. Our research team aims to explore text entry utilizing slide and microroll gestures. Text entry is notoriously difficult on small form factors [48] or through joystick interaction [44], and microrolls could allow for layer and function commands to be invoked with ease. We also look to demonstrate the potential capacity for interaction while eyes-free of the smart-ring. As our implementation was location independent, unlike precise tapping or edge based interaction, we feel the use of slide and microroll gestures could allow for unimpeded interaction across a range of eyes-free scenarios, yet needs further study.

6 CONCLUSION

This work provides an early look at our research focusing on expanding touch interaction capabilities for smart-ring devices. Due to the limited interaction space available, touch interactions are often held to simple slide and tap gestures; this limits the total usable interaction vocabulary, thus restricting overall use cases for smart-rings. In order to expand current touch capabilities, we explore the combination of slide and microrolling gestures that can occur within the same touch instance. Through data capture, feature extraction, model creation and validation, we achieve an accuracy of 92.4% through an RF model to discern slide and microroll gestures within the same touch instance. We highlight our findings and provide two applications that can benefit from continual slide and microroll gesturing. Finally, we discuss limitations and future work that will carry this research forward, allowing us, and the broader research community, to further explore the potential interaction capabilities of smart-rings.

ACKNOWLEDGMENTS

We would like to thank all of our volunteers and the support from Huawei Canada. We also acknowledge funding support from the NSERC CREATE grant on Visual and Automated Disease Analytics for the first two authors.

REFERENCES

[1] M Usman Akram, Zabeel Bashir, Anam Tariq, and Shoab A Khan. 2013. Geometric feature points based optical character recognition. In *2013 IEEE Symposium on Industrial Electronics & Applications*. IEEE, 86–89.

[2] Lisa Anthony and Jacob O Wobbrock. 2010. A lightweight multistroke recognizer for user interface prototypes. In *Proceedings of Graphics Interface 2010*. 245–252.

[3] Daniel Ashbrook, Patrick Baudisch, and Sean White. 2011. Nenya: subtle and eyes-free mobile input with a magnetically-tracked finger ring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2043–2046.

[4] Sandra Bardot, Surya Rawat, Duy Thai Nguyen, Sawyer Rempel, Huizhe Zheng, Bradley Rey, Jun Li, Kevin Fan, Da-Yuan Huang, Wei Li, et al. 2021. ARO: Exploring the Design of Smart-Ring Interactions for Encumbered Hands. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*. 1–11.

[5] Patrick Baudisch and Gerry Chu. 2009. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1923–1932.

[6] Hrvoje Benko, Andrew D. Wilson, and Patrick Baudisch. 2006. Precise Selection Techniques for Multi-Touch Screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1263–1272.

[7] Rachel Blagojevic, Samuel Hsiao-Heng Chang, and Beryl Plimmer. 2010. The Power of Automatic Feature Selection: Rubine on Steroids. *SBIM* 10 (2010), 79–86.

[8] Roger Boldu, Alexandru Dancu, Denys JC Matthies, Pablo Gallego Cascón, Shanaka Ransir, and Suranga Nanayakkara. 2018. Thumb-In-Motion: Evaluating Thumb-to-Ring Microgestures for Athletic Activity. In *Proceedings of the Symposium on Spatial User Interaction*. 150–157.

[9] David Bonnet, Caroline Appert, and Michel Beaudouin-Lafon. 2013. Extending the Vocabulary of Touch Events with ThumbRock *(GI '13)*. Canadian Information Processing Society, 221–228.

[10] Sebastian Boring, David Ledo, Xiang'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The fat thumb: using the thumb's contact size for single-handed mobile interaction. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*. 39–48.

[11] Jared Cechanowicz, Pourang Irani, and Sriram Subramanian. 2007. Augmenting the mouse with pressure sensitive input. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1385–1394.

[12] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

[13] Mayur Datar, Aristides Gionis, Piotr Indyk, and Rajeev Motwani. 2002. Maintaining stream statistics over sliding windows. *SIAM journal on computing* 31, 6 (2002), 1794–1813.
[14] Ramy El-Hajj, Laurence Likforman-Sulem, and Chafic Mokbel. 2005. Arabic handwriting recognition using baseline dependant features and hidden markov modeling. In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*. IEEE, 893–897.
[15] Barrett Ens, Ahmad Byagowi, Teng Han, Juan David Hincapié-Ramos, and Pourang Irani. 2016. Combining ring input with hand tracking for precise, natural interaction with spatial analytic interfaces. In *Proceedings of the 2016 Symposium on Spatial User Interaction*. 99–102.
[16] Bogdan-Florin Gheran, Jean Vanderdonckt, and Radu-Daniel Vatavu. 2018. Gestures for smart rings: empirical results, insights, and design implications. In *Proceedings of the 2018 Designing Interactive Systems Conference*. 623–635.
[17] Kyohei Hakka, Toshiya Isomoto, and Buntarou Shizuki. 2019. One-Handed Interaction Technique for Single-Touch Gesture Input on Large Smartphones. In *Symposium on Spatial User Interaction*. 1–2.
[18] Da-Yuan Huang, Ming-Chang Tsai, Ying-Chao Tung, Min-Lun Tsai, Yen-Ting Yeh, Liwei Chan, Yi-Ping Hung, and Mike Y Chen. 2014. TouchSense: expanding touchscreen input vocabulary using different areas of users' finger pads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 189–192.
[19] Jingun Jung, Sangyoon Lee, Jiwoo Hong, Eunhye Youn, and Geehyuk Lee. 2020. Voice+ Tactile: Augmenting In-vehicle Voice User Interface with Tactile Touchpad Interaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.
[20] Wolf Kienzle and Ken Hinckley. 2014. LightRing: always-available 2D input on any surface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 157–160.
[21] Junhyeok Kim, William Delamare, and Pourang Irani. 2018. Thumbtext: Text entry for wearable devices using a miniature ring. In *Graphics Interface*.
[22] Young-Ho Kim, Bongshin Lee, Arjun Srinivasan, and Eun Kyoung Choe. 2021. Data Hand: Fostering Visual Exploration of Personal Data on Smartphones Leveraging Speech and Touch Interaction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–17.
[23] DoYoung Lee, Jiwan Kim, and Ian Oakley. 2021. FingerText: Exploring and Optimizing Performance for Wearable, Mobile and One-Handed Typing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.
[24] Wing Ho Andy Li, Hongbo Fu, and Kening Zhu. 2016. BezelCursor: Bezel-initiated cursor for one-handed target acquisition on mobile touch screens. *International Journal of Mobile Human Computer Interaction (IJMHCI)* 8, 1 (2016), 1–22.
[25] Hyunchul Lim, Jungmin Chung, Changhoon Oh, SoHyun Park, and Bongwon Suh. 2016. OctaRing: examining pressure-sensitive multi-touch input on a finger ring device. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 223–224.
[26] David C McCallum, Edward Mak, Pourang Irani, and Sriram Subramanian. 2009. PressureText: pressure input for mobile phone text entry. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems*. 4519–4524.
[27] Ali Neshati, Bradley Rey, Ahmed Shariff Mohommed Faleel, Sandra Bardot, Celine Latulipe, and Pourang Irani. 2021. BezelGlide: Interacting with Graphs on Smartwatches with Minimal Screen Occlusion. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–13.
[28] Sebastien Pelurson and Laurence Nigay. 2016. Bimanual input for multiscale navigation with pressure and touch gestures. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*. 145–152.
[29] J Pradeep, E Srinivasan, and S Himavathi. 2011. Diagonal based feature extraction for handwritten character recognition system using neural network. In *2011 3rd international conference on electronics computer technology*, Vol. 4. IEEE, 364–368.
[30] Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. 2004. Pressure widgets. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 487–494.
[31] Mikko J Rissanen, Samantha Vu, Owen Noel Newton Fernando, Natalie Pang, and Schubert Foo. 2013. Subtle, Natural and Socially Acceptable Interaction Techniques for Ringterfaces—Finger-Ring Shaped User Interfaces. In *International Conference on Distributed, Ambient, and Pervasive Interactions*. Springer, 52–61.
[32] Anne Roudaut, Eric Lecolinet, and Yves Guiard. 2009. MicroRolls: Expanding Touch-Screen Input Vocabulary by Distinguishing Rolls vs. Slides of the Thumb *(CHI '09)*. Association for Computing Machinery, 927–936.
[33] Dean Rubine. 1991. Specifying Gestures by Example *(SIGGRAPH '91)*. Association for Computing Machinery, 329–337.
[34] Hasti Seifi and Kent Lyons. 2016. Exploring the design space of touch-based vibrotactile interactions for smartwatches. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*. 156–165.
[35] Marcos Serrano, Eric Lecolinet, and Yves Guiard. 2013. Bezel-Tap gestures: quick activation of commands from sleep mode on tablets. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 3027–3036.
[36] Roy Shilkrot, Jochen Huber, Jürgen Steimle, Suranga Nanayakkara, and Pattie Maes. 2015. Digital digits: A comprehensive survey of finger augmentation devices. *ACM Computing Surveys (CSUR)* 48, 2 (2015), 1–29.
[37] Gaganpreet Singh, William Delamare, and Pourang Irani. 2018. D-SWIME: A design space for smartwatch interaction techniques supporting mobility and encumbrance. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.
[38] Eugene M Taranta II, Amirreza Samiei, Mehran Maghoumi, Pooya Khaloo, Corey R Pittman, and Joseph J LaViola Jr. 2017. Jackknife: A reliable recognizer with few samples and many modalities. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 5850–5861.
[39] Karl Thurnhofer-Hemsi, Ezequiel López-Rubio, Miguel A Molina-Cabello, and Kayvan Najarian. 2020. Radial basis function kernel optimization for Support Vector Machine classifiers. *arXiv preprint arXiv:2007.08233* (2020).
[40] Hsin-Ruey Tsai, Min-Chieh Hsiu, Jui-Chun Hsiao, Lee-Ting Huang, Mike Chen, and Yi-Ping Hung. 2016. TouchRing: subtle and always-available input using a multi-touch ring. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. 891–898.
[41] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O Wobbrock. 2012. Gestures as point clouds: a $ P recognizer for user interface prototypes. In *Proceedings of the 14th ACM international conference on Multimodal interaction*. 273–280.
[42] 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*. 1–12.
[43] Bryan Wang and Tovi Grossman. 2020. BlyncSync: Enabling Multimodal Smartwatch Gestures with Synchronous Touch and Blink. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14.
[44] Andrew D Wilson and Maneesh Agrawala. 2006. Text entry using a dual joystick game controller. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 475–478.
[45] Jacob O Wobbrock, Andrew D Wilson, and Yang Li. 2007. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*. 159–168.
[46] Pui Chung Wong, Kening Zhu, and Hongbo Fu. 2017. FingerT9: Leveraging thumb-to-finger interaction for one-handed text entry on smartwatches. In *SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications*. 1–3.
[47] Pui Chung Wong, Kening Zhu, Xing-Dong Yang, and Hongbo Fu. 2020. Exploring eyes-free bezel-initiated swipe on round smartwatches. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–11.
[48] Zheer Xu, Weihao Chen, Dongyang Zhao, Jiehui Luo, Te-Yen Wu, Jun Gong, Sicheng Yin, Jialun Zhai, and Xing-Dong Yang. 2020. Bitiptext: Bimanual eyes-free text entry on a fingertip keyboard. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.
[49] Yina Ye and Petteri Nurmi. 2015. Gestimator: Shape and stroke similarity based gesture recognition. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. 219–226.