Investigating Multi-Touch Vibrations on Mobile Touchscreens

John Pasquesi[1] and Jenna L. Gorlewicz[2]

Abstract— **While many advanced haptic devices are under development, touchscreens are one of the most readily available platforms. In this paper, we leverage a leap forward in vibration-based haptics, Apple's new CoreHaptics API, and investigate its potential for a multi-finger vibration experience. We present a perceptual user study (N=15) that investigates multi-finger vibration identification and exploration strategies and we conduct a laser doppler vibrometry study, uncovering the challenges of developing consistent, high-quality vibration feedback across hardware platforms that vary in actuation principle, screen size, and external attachments. Repeated-measures ANOVA tests showed no statistically significant results in time, error, or weighted error for vibration identification across all participants based on number of fingers. However, one-way ANOVAs did show significant results within individuals, illustrating benefits of multiple-finger exploration. The most effective strategies for locating vibrations involved multiple fingers grouped together using sweeping motions across the screen. Our LDV study demonstrated that CoreHaptics and the Taptic Engine in the iPhone were capable of more accurately recreating specific frequencies than coin motors in Android devices. This research supports a move to a multi-finger, vibrotactile touchscreen experience, while highlighting the challenges of creating a consistent vibration experience across hardware platforms.**

I. BACKGROUND AND MOTIVATION

Vibrations have long been used as tertiary cueing mechanisms on mobile platforms. The buzz of an incoming message or phone call has become the norm. Engaging the tactile channel in primarily visual I/O devices like smartphones has demonstrated value in numerous ways - typing is more accurate, clicking and confirmation cues result in user satisfaction, and drag and drop tasks are more efficient [1], [2]. Vibrations have, in some capacity, brought touch back to touchscreens, though it is often noted that this is in limited form [1], [3], [4]. Yet, haptic capabilities in touchscreens provide an important tool for multimodal interaction [5]–[7], and the wide availability of touchscreens also makes them a common tool for assistive technologies [4], [8]–[11]. There has also been work in developing new types of touchscreens that allow richer haptic interactions involving variable friction and electrostatics [12]–[15]. While such enhanced haptic capabilities may one day be streamlined into mainstream touchscreens, we focus this paper on readily available, off-the-shelf hardware with vibration capabilities.

*This work is supported by the National Science Foundation under Grant Nos. 1644538 and 1845490. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

[1]John Pasquesi is a graduate student in Computer Science, Saint Louis University, St. Louis, MO, USA

[2]Jenna Gorlewicz is an Associate Professor in Mechanical Engineering, Saint Louis University, St. Louis, MO, USA jenna.gorlewicz@slu.edu

Previous work has informed our understanding of human perceptual capabilities within the field of vibrotactile displays (e.g. [11]). The key features of vibration are frequency and amplitude, and to distinguish one vibration from another a change in frequency or amplitude of 18-30%, with a minimum amplitude of no less than 5 micrometers [11], [16], is necessary. There are three distinct frequency ranges in the context of perceptual effects created by vibration: <3 Hz for slow kinesthetic motion, which feels like a slow pulse, 10-70 Hz for fluttering, which feels like tapping or a rapid pulse, and 100-300 Hz for smooth vibration, which feels similar to a steady buzz [11]. While these findings inform the perception of vibration at the fingertip, it isn't clear if these are upheld in the context of perceiving vibrations on a flat, rigid touchscreen platform, particularly with varying hardware and actuation capabilities, differing software capabilities, and in a multi-finger vibration experience. With vibrations being used in larger capacities on touchscreens, some of which require continuous perception of the signal (e.g. conveying graphical concepts to individuals with visual impairments (VI) [5], [6], [10], [17]–[19]), it is imperative that we better understand multi-vibration perception on touchscreens. In our previous research investigating line following and tracing via vibration on touchscreens, we have observed that users perceive a vibration signal to be different (even when it is the same) depending on their location on the screen and the screen size, and whether they have more than one finger on screen [7].

With major software platforms, namely Android and iOS, having infused vibrations into their mobile platforms in some form, one of the largest limitations of mobile vibration-based feedback to date is its restriction to a single finger, or lack of localization. Typically, commercial devices only allow a single vibration to be triggered, even if there are multiple fingers on screen, as the vibration effect is global across the screen. In some cases, this is also limited by the device being unable to independently track more than two fingers [20]. This limits exploration to a single finger or to psuedo multi-finger strategies. In our previous research, we observed that if a user needs to locate three key points of vibration on a multimodal, touchscreen-based graphic, they either need to use a single finger and carefully explore the whole screen, or alternate between different fingers, using one finger as the active finger at any given time (pseudo multi-finger). This type of non-visual, continuous interaction in the context of multimodal graphic exploration is one of the key use cases for this work. We are specifically interested in investigating 1) if and how multi-fingered vibration identification could enhance exploration strategies in this context and 2) to what extent a consistent vibrotactile experience can be provided to users across platforms.

Pattern	2.5Hz	5Hz	10Hz	25Hz	50Hz	100Hz	125Hz	150Hz	175Hz	200Hz	250Hz	500Hz	Knock	Quick Pulse	Slow Pulse	Rumble	Vibrate
Used In	Both	LDV	Both	LDV	LDV	LDV	LDV	LDV	LDV	LDV	Both	LDV	Both	LDV	Both	LDV	LDV

TABLE I
VIBRATION PATTERNS AND THE STUDIES THEY WERE USED IN IN THIS WORK. BOTH REFERS TO USE IN BOTH THE LDV AND HUMAN USER STUDY.

On each software platform, there are recommendations for using haptics for touch feedback, attention cues, notifications, impacts, selections, and specific events [21], [22]. These recommendations do not include how to utilize the haptic capabilities for more advanced cases, as they are not standard practice. However, with the introduction of iOS 13 and the Core Haptics API, multiple simultaneous vibrations and multi-finger tracking are now possible, due to the ability to instantiate several vibrations simultaneously. Even with these new capabilities, vibrations can differ substantially between devices. Libraries vary between Android and iOS, and even the physical vibration motors are different. The Taptic Engine within the iPhone claims faster startup times and greater accuracy than traditional coin motors. In this paper, we use these new software capabilities to investigate 1) the perception of vibration signals in multi-finger exploration on touchscreens and 2) how vibration profiles are altered across Android and iOS platforms with varying screen sizes. To do this, we developed two software applications (Android and iOS) that display vibration-based haptics on touchscreens, detailed in Section II. We then present a human user study investigating the perception of multiple touchscreen-based vibrations with single and multiple fingers in Section III. In Section IV, we conduct a scanning Laser Doppler Vibrometry (LDV) study investigating how vibration profiles are displayed and propagate across varying hardware platforms. We close with comparisons of hardware platforms for vibration-based haptics as well as recommendations on vibration profile characteristics and user strategies for interpreting multi-vibrational cues.

II. SOFTWARE DEVELOPMENT

In previous, collaborative work with Stefik and colleagues [10], we developed an open-source Android API for haptics [23]. The Android vibration API was developed as part of the Quorum Programming Language. It enables several key features: predefined patterns, variable frequency functions, and variable or infinite duration vibrations. Among the predefined patterns are quickPulse, slowPulse, rumble, knock, and vibrate - each of which can be looped indefinitely, or set to a specific duration. There is also the vibrateAtFrequency function, which allows the user to define a range of vibration frequencies to send. This API enables a wide range of vibration capabilities, with many distinct patterns possible.

Here, we extend this development work into iOS. Using the new CoreHaptics API, an analogous iOS API, which mirrors the Android API, was developed in Swift. This API offers the same predefined functions (rumble, quickPulse, slowPulse) as well as the ability to vibrate at different frequencies for a set duration of time, or until the vibration is cancelled [24]. This allows for selection of patterns from the existing options, as well as defining new patterns.

Using both the Android and iOS libraries, we created vibration patterns ranging from 2.5 Hz to 500 Hz (see Table I) for use in two applications: one for the user study (Section III) and one for the LDV study (Section IV). A larger number of vibration profiles were created for the LDV study to observe propagation of a variety of signals across hardware platforms. For the user study, five patterns were selected to provide a variety of both frequency and type of vibration, while keeping the user study time manageable. Of the five profiles, three changed only in frequency, and two changed in both frequency and amplitude. To build upon previous work, we selected three frequencies falling within the established ranges of perceptually distinct vibration: 2.5 Hz for slow kinesthetic motion, 10 Hz for fluttering, and 250 Hz for smooth vibration, all of which were at 100% amplitude and 50% sharpness [11]. The two patterns, which varied in both frequency and amplitude, were Knock (alternated between high (100%) and low (0%) amplitude in a two tap then one tap pattern, at 100% sharpness) and slowPulse (alternated between high (100%) and low amplitude (0%) in slow pulses), which were chosen due to their existing inclusion in the Android API. For exact, detailed implementation of the patterns, refer to [24]. These vibrations are used in the user study, as randomized sets of the selected vibrations to test user ability to find and distinguish profiles. The LDV study application plays vibrations until the user stops them, so that measurements could be taken via a scanning LDV across several hardware platforms.

III. HUMAN USER STUDY DESIGN AND RESULTS

The goal of the human user study was to investigate if and how users could identify and discern different vibration patterns displayed on touchscreens, as number of fingers used in exploration varies. Additionally, we wanted to understand what strategies users employed, when not prescribed methods, to complete the above tasks. Three research questions motivate this study:

- **RQ1:** Are the vibration profiles selected based on previous research distinguishable when displayed on a touchscreen platform?
- **RQ2:** What number of fingers is most efficient and accurate for distinguishing the selected vibration profiles?
- **RQ3:** What multi-finger exploration strategies support vibration identification and discrimination in the context of the selected vibration profiles?

Based on previous work utilizing multiple fingers [25], we expected that use of more than one finger would improve user performance.

Number of Unique Vibrations	Patterns
1	A, B, C, D, E, AA, BBB, CCCC, DDDDD, EE, DDD, AAAA, BBBBB, CC, EEEEE
2	AB, AC, AD, AE, BC, BD, BE, CD, CE, DE, AAE, BBCC, DDEEE, CCAAA, BBD
3	ABC, ABD, ABE, ACD, ACE, ADE, BCD, BCE, BDE, CDE, AACE, BBCCD, CDDA, BBEC, EEBA
4	ABCD, ABCE, ABBDE, ACDE, BCDE, AABCD, ABCCE, ABBDE, ACDDE, BCDEE
5	ABCDE

TABLE II
A=2.5 Hz, B=10 Hz, C=250 Hz, D=Knock, E=Slow Pulse

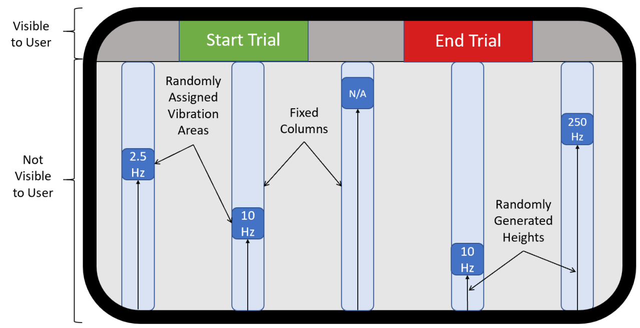

Fig. 1. Layout of the user study application during the main trials, as generated by the software. The user can see the top two buttons, but the rest of the screen is blank.

A. Methods

This study was approved by the university's Institutional Review Board. We utilized an iPhone 11 with popsockets attached to the back, a Mac laptop, and a camera with tripod for this study. The only criteria for study participation was that participants were able to use their fingers on a touchscreen. We recruited 15 individuals (age range: 18-34, 9 male, 6 female) to participate. The study lasted approximately 45 minutes-1.5 hours and consisted of two phases: a calibration phase and the main trials.

The calibration phase consisted of 15 trials and helped acclimate the user to the vibration profiles used in the study while also determining how well users could identify and distinguish the vibration profiles. In each trial, the participant was asked to determine if the two vibration profiles presented to them were the same or different patterns, regardless of strength of the vibration, and to answer with either "same" or "different". These 15 trials encompassed all possible combinations of the 5 different vibration patterns that were selected for use, ensuring the participant experienced every vibration profile before moving to the main trial phase. The application output the vibration pattern and time taken for each answer, and the researcher recorded the user's answer for correctness.

The main trial phase consisted of a total of 50 trials, broken into 5 sections of 10 trials each. Figure 1 illustrates the layout that is generated by the software for each individual main trial. Each of the dark blue squares represents an area in which a vibration was activated. These are the vibrations the users are trying to locate and identify, but they are invisible to the user. The vibration that each of these areas triggers was selected randomly by the program from the pool of vibrations in Table II. Within each of the five fixed columns on the screen, the vibration areas were assigned a randomly generated height, so that the participants would have to search for the vibrations. All of these randomly generated elements are created each time the start button is pressed. While the different vibration areas can be triggered simultaneously, most participants avoided doing so, as it added difficulty to distinguishing unique patterns.

For each of the 5 main sections, the participant is first assigned a number of fingers, between 1 and 5, to be used to complete the study section. The participant is instructed that they can explore the touchscreen's surface in whatever way they see fit, as long as they keep the prescribed number of fingers on the screen at all times. The participant is then informed that their goal is to identify the number of distinct (varying in pattern) vibrations found on the screen. To illustrate what is meant by distinct, we look to Figure 1. In this figure, there are three unique patterns displayed: 2.5 Hz, 10 Hz, and 250 Hz. Even though there are four active vibration areas, there are only three distinct patterns. Thus, the correct answer for this trial would be three. Once the participant feels confident that they have located all distinct vibrations, they hit the end button and give their answer to the researcher. This was repeated ten times with the assigned number of fingers. After these ten trials, the process is repeated four more times, each with a different number of fingers. During every trial, both the time and accuracy are recorded.

During all trials, the interaction of the participant's hands with the iPhone was video recorded. The recorded video was analyzed for user interaction on the screen. The categories used to describe the user's exploration were as follows: 1) *Touchscreen approach:* the direction in which the user primarily explored the screen, 2) *Speed of Hands:* how quickly the user's hands and fingers were moving, 3) *Search Methods:* Sweeping includes broad, organized strokes, while Searching is moving about the screen in no systematic fashion, 4) *Finger Placement:* Grouped or Separated fingers means the user either kept fingers closely grouped together or separated to explore independently with each, and 5) *Number of Hands:* the user primarily explored with fingers on both hands simultaneously, or only one hand.

B. Results

RQ1: In this investigation, we sought to confirm that the vibrations used in the study were perceptually distinct when displayed on a touchscreen platform, as previous research suggests they should be. The key data analyzed was the calibration data (N=225). To determine which vibration patterns were most distinguishable, we ran a repeated measures ANOVA test analyzing each pairing of vibrations across all trials and participants. Figure 2 illustrates these results for both time and error differences. Based on the resulting

	F Value	Num DF	Den DF	Pr > F
Time	1.3107	14.0000	196.0000	0.2036
Error	0.8042	14.0000	196.0000	0.6640

Fig. 2. Time and accuracy ANOVA results of calibration trials where users determined if two vibrations were the same or different.

	F Value	Num DF	Den DF	Pr > F
Time	0.2559	4.0000	56.0000	0.9048
Error	1.3665	4.0000	56.0000	0.2572
Weighted Error	0.6184	4.0000	56.0000	0.6512

Fig. 3. Time, error, and weighted error ANOVA results for the main trials where users determined the number of distinct vibration patterns hidden on the screen.

values, we rejected the null hypothesis that there were statistically significant differences in the mean times and errors for different pairings of vibrations. This suggests that the selected patterns were all roughly as easy or as difficult to distinguish. Based on the fact that the slowest time for any pairing was 12.29 seconds and the highest error was seven percent, we conclude that these patterns are all easily distinguished from one another. These empirical findings are supported by previous theoretical work [11], and confirm that the selected profiles were fitting for use in the remainder of the trials.

RQ2: In this investigation, we sought to determine which number of fingers resulted in the best user performance in terms of time and accuracy. To this end, the main trial data (N=750) was analyzed with a repeated measures ANOVA for time, error and a weighted error designed to value error-free trials more highly (Equation 1).

$$Time + 2 \times PercentError \tag{1}$$

This formula was selected to illustrate the methods that were most effective overall, by being both fast and accurate. The results of these ANOVA tests are presented in Figure 3. These results show that there is no statistically significant difference in any of these categories based on the number of fingers, when comparing across all participants. To further explore if any participants had varying performance based on the number of fingers used, we ran one-way ANOVA tests on each participant, comparing their time, error, and weighted error results based on the number of fingers. We found that there were participants within each of these categories that did experience a significant difference based on the number of fingers utilized. The participants that were identified as having statistically significant differences in these fields were then further investigated.

While there are results for time, error, and weighted error, we present the further investigation of weighted error because it includes the most relevant results, those that combine both accuracy and speed. In Figure 4, the results of a Tukey analysis on each participant's results are presented. For each participant, the results indicate which number of fingers was better than which other number of fingers in a statistically significant way. While cases like participant 4 and 9 tell us that some users might perform worse with 4 or 5 fingers, the

	Statistically Significant Weighted Error Results (P<0.05)
Participant 1	2>1, 3>1, 4>1, 5>1
Participant 4	3>4, 5>4
Participant 9	3>5, 4>5
Participant 10	4>1
Participant 12	4>1, 4>3

Fig. 4. Statistically significant results for weighted error based on Tukey analysis. > symbol indicates that the left value scored better than the right value.

most interesting results are from participants 1, 10, and 12. In particular, we see that participant 1 obtains better results with any number of fingers besides a single finger. Similarly, for participants 10 and 12, the use of four fingers outperformed the use of a single finger. These scenarios illustrate the potential value-add provided by a multi-finger experience, though we note that this performance enhancement is not significant across the study population.

This data demonstrates that a user can differentiate vibration patterns displayed on a touchscreen with multiple fingers. This study also illustrates that a multi-finger approach can provide increased performance for some participants, even if not for all. These results illustrate that multi-finger vibration perception, particularly when vibration signals are chosen strategically for best perception, is quite accurate on mobile touchscreens, even up to 5 fingers.

RQ3: In this investigation, we sought to uncover optimal exploration strategies for identifying vibrations on a touchscreen. To do this, we used all of the data collected coupled with recorded video. Each participant was ranked in terms of time performance, error performance, and weighted error. Figure 5 presents the data arranged in terms of a weighted sum. We found that in terms of time and error (combined), the majority of the best scores were obtained with sweeping motions, and all used grouped fingers. The worst scores used searching methods with separated fingers. These results are confirmed in the error assessment (not pictured). In terms of time assessment, no clear methods arose that were consistently faster than others. This provides a strong argument for why it is not only the presentation of the vibrations that matters, but also the exploration method that is employed.

C. Study Limitations

While our study design was created to find the answers to our research questions, we also acknowledge there are several limitations due to our design decisions. First, we left several variables open to interpretation by the participants. This was intentional, as we wanted to study the various strategies employed by users, but this also means there is less consistency in the user experience. Second, while we could have controlled for time and only measured accuracy, or vice-versa, we chose to capture both simultaneously. We chose this method to capture a realistic scenario, the exploration of an unknown graphic, where both time and accuracy matter. However, this means it is difficult to draw specific conclusions about these factors independently. We acknowledge that further testing could add to this study.

Participant	Touchscreen approach	Number of Hands	Speed of Hands	Search Methods	Finger Placement
12	Horizontal	1	Moderate	Sweeping	Grouped
7	Both	1	Moderate	Sweeping	Grouped
8	Vertical	2	Moderate	Searching	Grouped
6	Vertical	1	Moderate	Sweeping	Grouped
1	Horizontal	2	Moderate	Sweeping	Grouped
9	Horizontal	1	Moderate	Sweeping	Grouped
15	Vertical	2	Moderate	Sweeping	Inconsistent
3	Inconsistent	2	Moderate	Searching	Separated
2	Vertical	2	Slow	Searching	Separated
4	Inconsistent	2	Slow	Searching	Separated

Fig. 5. Analysis of participant video with the top and bottom five participants as sorted by weighted value of error and time.

IV. LDV STUDY AND RESULTS

Toward understanding how various vibration patterns propagate on different hardware devices, a series of benchtop studies were conducted to complement the perceptual user study. We used a single-point LDV (Polytech PDV-100), a scanning LDV (Polytech PSV-500), three different touchscreen devices, a foam pad, and a popsocket attachment. This study was motivated by a single question:

- **RQ4:** How do vibration patterns propagate in different hardware devices?

This question is equally important to the perceptual questions above, to understand if and how consistent vibrotactile feedback can be provided across varying platforms for a seamless user experience.

A. Methods

We chose to investigate the effects of screen size, hardware platform, and the use of external attachments by using the following devices (all units are in mm): iPhone 11 (151.0 x 76.0 x 8.4), Motorola Z Play Droid (160.0 x 81.0 x 7.7), Motorola Z Play Droid with Battery Pack (160.0 x 81.0 x 16.5), and Samsung Galaxy S3 Tablet (235.0 x 170.0 x 5.0). Note also that while the iPhone uses the Taptic Engine, the Moto and Tablet use coin motors.

Hardware devices were individually adhered to a foam pad (for reducing interference from external vibrations) via popsockets. The device and holding apparatus were placed on a vibration isolation optical table, upon which the LDV systems were mounted. A single-point LDV served as the reference point for the scanning LDV, which was set at the center of the screen for each trial. Prior to measurement, the device was centered in the view of the scanning LDV, and the focus was set to an appropriate level. Then a grid was overlaid on the device, defining measurement points approximately 5mm apart for the scan. Utilizing our APIs for vibration, each device was then set to vibrate for an indefinite period of time, and an averaging scan was started. These scans typically took between 5 and 20 minutes, and measured each point multiple times. For each device, a scan was run for all of the patterns referenced in Section II. The patterns were sent to the devices with the same software instructions, with the key difference being the device's actuation method. The complete set of measurements for all 4 device configurations provided the dataset that was drawn from for analysis. To analyze the data, we filtered by device type and attachment, comparing the amplitude of vibration at each frequency. We investigated effects from screen size, varying hardware platforms (iOS vs. Android), and external attachments (e.g. battery pack).

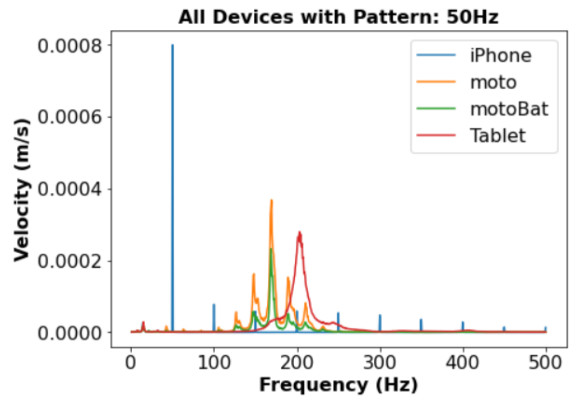

Fig. 6. All devices vibrating at 50 Hz, as measured by the scanning LDV.

B. Results

Key findings from our analysis demonstrate that vibration propagation is affected by all of the parameters investigated and that even within the same library, results vary between hardware devices. Not surprisingly, we observed that the addition of a battery pack changes amplitude by up to 20% but not frequency of vibration. As expected, we also observe that larger screen sizes (e.g. Tablet) suffer from vibration dampening and reduced amplitude of approximately 50% as the signal propagates across the screen. Findings demonstrate that the tested Android devices (Moto and Tablet) are able to maintain strongest amplitudes only at the natural frequency of the motor, which means that commanded vibration profiles may not always be achieved if they are far from the motor's natural frequency. While the overall impression of the vibration may still be conveyed, this may mean the vibration feels more like repeated 200 Hz vibrations (buzzing) than a continuous 50 Hz vibration (pulsing). On the other hand, iOS devices overcome this limitation because of the Taptic Engine, demonstrating much higher accuracy in replicating commanded vibration profiles at specific frequencies. When the devices are all set to vibrate at 50 Hz, as in Figure 6, the striking differences can easily be observed. While the Moto and Tablet still vibrate at roughly the same frequency as their default vibration (175 Hz and 205 Hz, respectively), the iPhone distinctly peaks at 50 Hz, with smaller amplitudes arising near harmonics of this frequency, rather than at its default vibration (135 Hz). A similar trend was observed with several of the other frequencies.

By observing the animations of vibration propagation obtained from the LDV, we confirmed these trends. To watch the full animations for all the situations that were measured, see the CHROME Lab site [26]. The iPhone shows different propagation patterns based on the frequency being transmitted and illustrates the general trend of the strongest vibrations occurring near the base of the phone, where the vibration motor rests. The Tablet and the Moto both show significantly less variance between the frequencies, indicating again that they do not do as well at providing distinct patterns. The tablet at least shows some variance at 250 Hz, while the Moto remains generally the same across all frequencies. These results illustrate that one must use caution when creating vibration profiles that are meant to have specific characteristics, particularly on Android platforms, as the hardware can have difficulty producing different profiles. On the other hand, these results raise the excitement about what is capable in CoreHaptics, particularly on the production of specific vibration characteristics.

V. CONCLUSIONS

In this paper, we present an investigation into touchscreen-based vibrations, their potential for a multi-finger experience, and their scalability across various hardware platforms. Through a perceptual-based user study, we illustrated that users can perceive distinct vibrations with multiple fingers on the same screen, even up to 5 fingers. We also observed that the most effective user exploration strategies involved the use of multiple fingers grouped together, exploring in sweeping motions across the screen. Based on these findings, we believe that best practices for multimodal, vibration-based graphics should allow for the use of multiple fingers and varied methods for exploration. Through a benchtop, LDV measurement study, we demonstrated the challenges of displaying consistent vibration patterns across hardware platforms that vary in actuation capabilities, screen size, and external attachments. We compared the accuracy with which one can command specific vibration profiles in software and see them replicated on the hardware, across iOS and Android, finding significant advantages with the iPhone's CoreHaptics library and hardware actuation, although it has yet to be determined if these advantages would scale to larger screen sizes. We recommend caution about any assumptions of consistent representation of vibrations when different actuators are involved. Future work will investigate how the creation of vibration profiles can be streamlined on the software side for cross-platform use and will compare Android and iOS renderings of vibration-based graphics and explorations to determine if the measured advantages are also perceived by users in the context of varying tasks.

REFERENCES

[1] K. E. MacLean, O. S. Schneider, and H. Seifi, "Multisensory haptic interactions: understanding the sense and designing for it," in *The Handbook of Multimodal-Multisensor Interfaces: Foundations, User Modeling, and Common Modality Combinations - Volume 1*. ACM, apr 2017, pp. 97–142.

[2] Z. Ma, D. Edge, L. Findlater, and H. Z. Tan, "Haptic keyclick feedback improves typing speed and reduces typing errors on a flat keyboard," in *IEEE World Haptics Conference, WHC 2015*. Institute of Electrical and Electronics Engineers Inc., aug 2015, pp. 220–227.

[3] R. Rastogi and D. T. Pawluk, "Toward an improved haptic zooming algorithm for graphical information accessed by individuals who are blind and visually impaired," *Assistive Technology*, vol. 25, no. 1, pp. 9–15, mar 2013.

[4] H. P. Palani, J. L. Tennison, G. B. Giudice, and N. A. Giudice, "Touchscreen-based haptic information access for assisting blind and visually-impaired users: Perceptual parameters and design guidelines," in *Advances in Intelligent Systems and Computing*, vol. 794. Springer Verlag, jul 2019, pp. 837–847.

[5] H. P. Palani, G. B. Giudice, and N. A. Giudice, "Haptic information access using touchscreen devices: Design guidelines for accurate perception of angular magnitude and line orientation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10907 LNCS. Springer Verlag, jul 2018, pp. 243–255.

[6] P. Gershon, R. L. Klatzky, H. Palani, and N. A. Giudice, "Visual, tangible, and touch-screen: Comparison of platforms for displaying simple graphics," *Assistive Technology*, vol. 28, pp. 1–6, jan 2016.

[7] J. L. Tennison, Z. S. Carril, N. A. Giudice, and J. L. Gorlewicz, "Comparing Haptic Pattern Matching on Tablets and Phones," *Optometry and Vision Science*, vol. 95, no. 9, pp. 720–726, 2018.

[8] S. O'Modhrain, N. A. Giudice, J. A. Gardner, and G. E. Legge, "Designing media for visually-impaired users of refreshable touch displays: Possibilities and pitfalls," *IEEE Transactions on Haptics*, vol. 8, no. 3, pp. 248–257, jul 2015.

[9] R. L. Klatzky, N. A. Giudice, C. R. Bennett, and J. M. Loomis, "Touch-screen technology for the dynamic display of 2d spatial information without vision: Promise and progress," *Multisensory Research*, vol. 27, no. 5-6, pp. 359–378, 2014.

[10] J. L. Tennison and J. L. Gorlewicz, "Non-visual Perception of Lines on a Multimodal Touchscreen Tablet," *ACM Trans. Appl. Percept*, vol. 16, no. 6, 2019. [Online]. Available: https://doi.org/10.1145/3301415

[11] S. Choi and K. J. Kuchenbecker, "Vibrotactile display: Perception, technology, and applications," *Proceedings of the IEEE*, vol. 101, no. 9, pp. 2093–2104, 2013.

[12] O. Bau, I. Poupyrev, A. Israr, and C. Harrison, "TeslaTouch: Electrovibration for touch surfaces," in *UIST 2010 - 23rd ACM Symposium on User Interface Software and Technology*, 2010, pp. 283–292.

[13] H. Xu, M. A. Peshkin, and J. E. Colgate, "UltraShiver: Lateral force feedback on a bare fingertip via ultrasonic oscillation and electroadhesion," in *IEEE Haptics Symposium, HAPTICS*, vol. 2018-March. IEEE Computer Society, may 2018, pp. 198–203.

[14] M. Dariosecq, P. Plénacoste, F. Berthaut, A. Kaci, and F. Giraud, "Investigating the semantic perceptual space of synthetic textures on an ultrasonic based haptic tablet," Tech. Rep., feb 2020.

[15] F. Giraud, T. Hara, C. Giraud-Audine, M. Amberg, B. Lemaire-Semail, and M. Takasaki, "Evaluation of a Friction Reduction Based Haptic Surface at High Frequency," Tech. Rep., mar 2018.

[16] H. Pongrac, "Vibrotactile perception: Differential effects of frequency, amplitude, and acceleration," in *Proceedings of the 2006 IEEE International Workshop on Haptic Audio Visual Environments and Their Applications, HAVE 2006*. Institute of Electrical and Electronics Engineers Inc., 2006, pp. 54–59.

[17] J. L. Tennison and J. L. Gorlewicz, "Toward non-visual graphics representations on vibratory touchscreens: Shape exploration and identification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9775. Springer Verlag, 2016, pp. 384–395.

[18] D. T. Pawluk, R. J. Adams, and R. Kitada, "Designing haptic assistive technology for individuals who are blind or visually impaired," *IEEE Transactions on Haptics*, vol. 8, no. 3, pp. 258–278, jul 2015.

[19] E. E. Abdallah and E. Fayyoumi, "Assistive Technology for Deaf People Based on Android Platform," in *Procedia Computer Science*, vol. 94. Elsevier B.V., jan 2016, pp. 295–301.

[20] "PackageManager." [Online]. Available: https://developer.android.com/reference/android/content/pm/PackageManager.html

[21] "Human Interface Guidelines: Apple Developer." [Online]. Available: https://developer.apple.com/design/human-interface-guidelines/ios/user-interaction/haptics/

[22] "Android haptics Design." [Online]. Available: https://material.io/design/platform-guidance/android-haptics.html

[23] A. Stefik and M. Uesbeck, "Android haptic library (java)," January 2020. [Online]. Available: https://bitbucket.org/stefika/androidquorum/src/master/AndroidHaptic/

[24] "chromelab/iosVibrationManager." [Online]. Available: https://github.com/chromelab/iosVibrationManager

[25] J. C. Craig, "Attending to two fingers: Two hands are better than one," Tech. Rep., 1986.

[26] "Chrome Lab Website." [Online]. Available: https://sites.google.com/slu.edu/gorlewicz-lab/research/ldv-data