Behavior Research Methods (2023) 55:2800–2812
https://doi.org/10.3758/s13428-022-01925-1

To BYOD or not: Are device latencies important for bring-your-own-device (BYOD) smartphone cognitive testing?

Jessica Nicosia[1] · Benjamin Wang[2] · Andrew J. Aschenbrenner[1] · Martin J. Sliwinski[3] · Scott T. Yabiku[4] · Nelson A. Roque[5] · Laura T. Germine[6,7] · Randall J. Bateman[1] · John C. Morris[1] · Jason Hassenstab[1,8]

Accepted: 6 July 2022 / Published online: 11 August 2022

Abstract
Studies using remote cognitive testing must make a critical decision: whether to allow participants to use their own devices or to provide participants with a study-specific device. Bring-your-own-device (BYOD) studies have several advantages including increased accessibility, potential for larger sample sizes, and reduced participant burden. However, BYOD studies offer little control over device performance characteristics that could potentially influence results. In particular, response times measured by each device not only include the participant's true response time, but also latencies of the device itself. The present study investigated two prominent sources of device latencies that pose significant risks to data quality: device display output latency and touchscreen input latency. We comprehensively tested 26 popular smartphones ranging in price from < $100 to $1000+ running either Android or iOS to determine if hardware and operating system differences led to appreciable device latency variability. To accomplish this, a custom-built device called the Latency and Timing Assessment Robot (LaTARbot) measured device display output and capacitive touchscreen input latencies. We found considerable variability across smartphones in display and touch latencies which, if unaccounted for, could be misattributed as individual or group differences in response times. Specifically, total device (sum of display and touch) latencies ranged from 35 to 140 ms. We offer recommendations to researchers to increase the precision of data collection and analysis in the context of remote BYOD studies.

Keywords Smartphones · Remote assessment · BYOD · Ambulatory assessment

Introduction

Over the past 30 years, there has been an exponential increase in the number of studies which have used remote technology for data collection—including, but not limited to, digital cognitive assessments, experience sampling, daily diaries, and ecological momentary assessments (Hamaker & Wichers, 2017). These types of studies offer a glimpse into participants' daily cognitive, physiological, and environmental experiences (Fahrenberg, 2006; Sliwinski, 2008; Sliwinski et al., 2018). There are many advantages to digital cognitive assessments as compared to traditional in-laboratory

✉ Jason Hassenstab
hassenstabj@wustl.edu

1 Charles F. and Joanne Knight Alzheimer Disease Research Center, Department of Neurology, Washington University School of Medicine, St. Louis, MO, USA

2 Mountain View, USA

3 Department of Human Development and Family Studies, The Pennsylvania State University, University Park, PA, USA

4 Department of Sociology and Criminology, The Pennsylvania State University, University Park, PA, USA

5 Department of Psychology, University of Central Florida, Orlando, FL, USA

6 Department of Psychiatry, Harvard Medical School, Boston, MA, USA

7 Institute for Technology in Psychiatry, McLean Hospital, Belmont, MA, USA

8 Department of Psychological & Brain Sciences, Washington University in St. Louis, St. Louis, MO, USA

or in-clinic assessments. These advantages include reduced recall bias (Mehl & Conner, 2014), higher ecological validity (Trull & Ebner-Priemer, 2014), and increased accessibility and sample sizes due to minimized recruitment and retention barriers (Germine et al., 2012; Kraut et al., 2004). Beyond facilitating larger and more representative samples, mobile studies of cognition are also more accessible for individuals who might otherwise be unable to participate due to financial or mobility reasons, making it easier for more people to engage with scientific research (de Liaño et al., 2012; Passell et al., 2021).

A critical decision when planning a remote digital study is whether to allow participants to use their own devices or to provide them with a study-specific device such that all participants' data is collected on identical hardware and operating systems. Remote bring-your-own-device (BYOD) studies can have several benefits. First, studies which allow participants to use their personal devices have been shown to elicit more diverse samples than studies which rely on conventional psychological research recruitment techniques (such as subject pool databases predominantly comprised of Western, educated, industrialized, rich, and democratic participants; Gosling & Mason, 2015; Henrich et al., 2010). Second, BYOD studies generally increase accessibility by reducing barriers to participation such as having to come into the clinic or adopting an unfamiliar device or operating system. Relatedly, for participants who already own a smartphone, BYOD studies avoid the hassles of managing an additional device. This is particularly relevant for clinical populations with cognitive impairment where learning to use a new device or attempting to manage a secondary personal device (e.g., transporting and charging two smartphones) can be particularly challenging. Finally, BYOD is more economical for researchers, which can translate to larger sample sizes due to reduced upfront costs.

The benefits of conducting clinical research using a BYOD model also come with some trade-offs. For example, while traditional laboratory-based studies afford researchers control over the testing environment and administration, the same cannot be said for remote assessments using personal devices (De Bruijne & Wijnant, 2013; Nosek et al., 2002; Reips, 2000). Researchers have little control over the technology participants use and the setting in which participants complete the session (although this is often touted as a benefit for more generalizable findings in the ecological momentary assessment, or EMA, literature; see Woods et al., 2015, for example). Perhaps most importantly, differences in participant device hardware and software have the potential to introduce unanticipated variance into the data which may impact results if not understood and controlled for (Germine et al., 2019; Passell et al., 2021; Woods et al., 2015). One critical source of this variance that may affect cognitive data collected in digital studies is differences in device latencies. In the context of smartphone cognitive assessments, device latency can be defined as the portion of the participant's measured response time that is not due to the participant. More specifically, the ***device*** latency is any duration from both (1) when software triggers an event/stimulus to when the event physically occurs (e.g., screen display or stimulus; referred to in this paper as display latency) and (2) when the user performs an action (e.g., taps the screen; referred to in this paper as touch latency) until the software registers the input (Foxlin, 2002; Pavlovych & Gutwin, 2012). Because these latencies are device-specific, their effects may carry over into multiple aspects of an experimental task including stimulus display, inter-trial intervals, and response time recordings.

Although most devices on the market today have relatively unnoticeable "lag," variability in device latency has the potential to influence cognitive assessments which rely on the precise display of visual stimuli and recording of participants' response times. Because the average simple response time can range from 200 to 300 milliseconds (ms) (Jain et al., 2015; Wilkinson & Allison 1989; Woods et al., 2015) and device latencies can range from 50 to 200 ms (for touchscreen devices), device latencies could introduce systematic error and exaggerate/suppress group differences in task performance (Henze et al., 2016; Pavlovych & Gutwin, 2012). Without further information on device characteristics, it may be difficult to control for device latency variation and take advantage of the benefits of digital cognitive assessments.

Evidence from Passell et al. (2021) indicated that cognitive test scores may vary with personal digital device. However, it is unclear whether these differences were due to cognitive and demographic factors that vary with personal device *choice* (e.g., tablet users tend to be older than users of other devices; Passell et al., 2021) or if these differences were due to variability in the devices themselves. Because response time data is essential for so many cognitive tasks, it is critical that we understand what is being measured and the extent to which variation in participants' personal devices may systematically bias results. If the variability introduced by BYOD models can be appropriately measured and mitigated, it can lead to greater research precision. Therefore, in the present study, we investigated device latencies (specifically, display and touch) across a series of popular smartphones and operating systems and provide researchers with a set of recommendations regarding digital assessments of cognition. Specifically, we used a custom-built robot to stimulate touchscreen input and measure display latencies to characterize the performance of 26 popular phones. Finally, we provide suggestions for researchers looking to optimize their digital cognitive assessment paradigms (see Table 1).

Table 1 BYOD study design considerations

(A) BYOD Study Design Choice		Potential Effect on Response Times
Within-Person	Mid-Study Change to Same Device	mild
	Mid-Study Change to Same Manufacturer	moderate
	Mid-Study Change to Different Manufacturer	severe
	Mid-Study OS Software Update	moderate
	Same Device, Same OS	mild
Between-Person	Different OS Versions (Same Device)	moderate
	Different Devices (Same Manufacturer)	moderate
	Different Devices (Different OS and/or Manufacturers)	severe

(B) BYOD Study Design Tradeoffs

Response Time Precision	Participant Sampling Pool	Cost	BYOD Design
~105 ms total device variability *max - min for all devices in the study*	Large	$	Full BYOD
~70 ms total device variability *max - min for iOS only devices in the study*	Moderate	$$	Selective BYOD
~17 ms total device variability *theoretical total latency variability of device with 120 Hz refresh & sampling rate*	Small	$$$	Device Provided

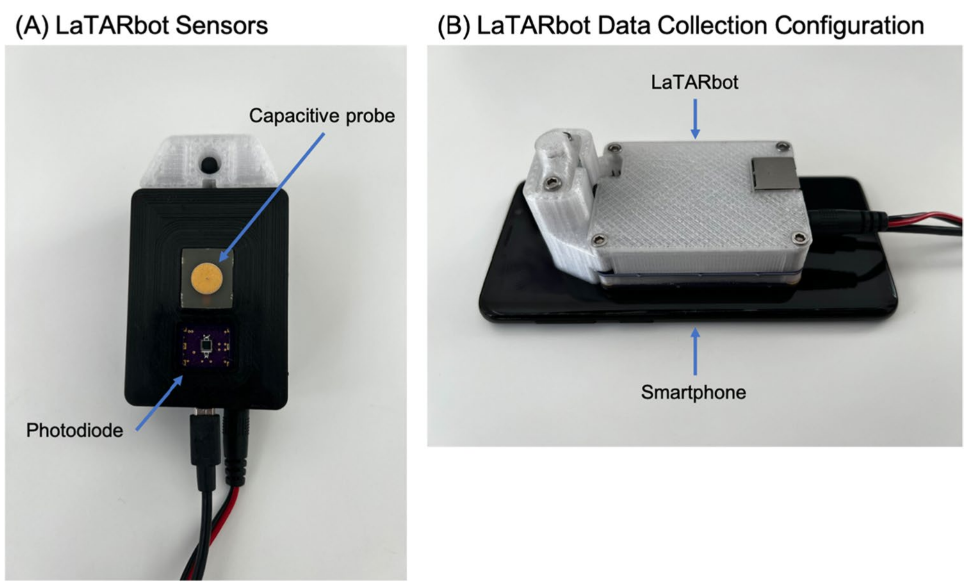

Fig. 1 Latency and Timing Assessment Robot (LaTARbot)

Methods

Apparatus The apparatus setup consisted of three distinct hardware components (see Figs. 1 and 2a): (1) a computer acting as the server and operator interface, (2) the custom-built Latency and Timing Assessment Robot (LaTARbot) capable of simulating user touch and reading screen brightness, and (3) the smartphone under test. The objective of the setup was to measure the latency between the touchscreen and the application running on the operating system (OS) in both directions (i.e., time from a touch event until input was registered or time from an application draw command until the display was updated). Each sample consisted of a pair of timestamps—a stimulus timestamp from the source device and a response timestamp from the destination device. Depending on the latency being measured (display or touch), the smartphone and LaTARbot would switch roles between being the source or destination device. The server laptop collected the samples and stored them for later analysis. A single test consists of a set of samples and its accompanying metadata for each run on each device. The metadata includes the phone information, clock sync results, test type (display or touch), number of samples, and interval between

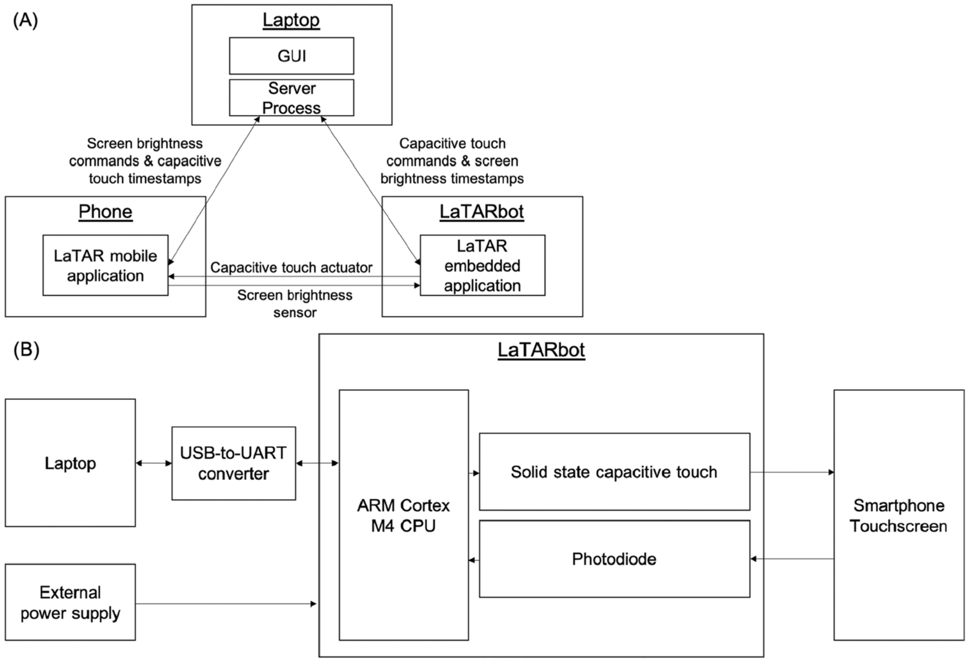

Fig. 2 System block diagram

samples. Clock sync and sampling interval are described in more detail at the end of this section.

The server provided a user interface to configure the test and collect the sample data. The computer used for the server was a Lenovo ThinkPad with an Intel Core i5-6300U CPU, 8 GB RAM, and built-in Wi-Fi running Ubuntu 20.04.3 LTS. The server software consisted of two parts: (1) a server process in the background and (2) a graphical user interface (GUI) in the foreground. Both applications were developed in C++ and the GUI used Qt software for the user interface. The background server process hosted an ad hoc Wi-Fi network for communication with the smartphone. The robot built to interact with the smartphone (LaTARbot) simulated user touch input and measured display output. LaTARbot communicated with the server via a wired USB connection and the smartphone communicated with the server via a wireless connection to the server's ad hoc Wi-Fi network (see Fig. 2).

For display latency testing, the smartphone application toggled the display between all-black and all-white for the LaTARbot to observe. The LaTARbot used a photodiode to sense changes in the brightness of the smartphone touchscreen (see Fig. 2a)[1]. The LaTARbot logged the timestamp at which each step change in brightness was detected. The smartphone logged two timestamps for each display transition. As shown in Fig. 3, the first timestamp was when the application called the drawing function to change the screen color (display "callback time") and the second timestamp was when the drawing function returned control to the application (display "action time").

For touch latency testing, the LaTARbot had a capacitive probe on the surface of an exposed, printed circuit board on the bottom of the robot which was held against the smartphone touchscreen. By electrically charging the capacitive probe, the LaTARbot emulated "human touch" with zero moving parts and precise timing accuracy (i.e., solid-state capacitive touch). Taps were triggered according to the test configuration sent from the server, and the timestamp at which the LaTARbot initiated the charging of the probe was recorded as the time of the tap. Similar to display latency, the smartphone application logged two timestamps during touch latency trials. First, the capacitive touch "action time" was the time at which the OS first registered the touch input (but the application had not yet received the data). Second, the capacitive touch "callback time" was when the application's callback function was called and the application registered the touch.

We examine both display *and* touch latencies such that the sum of the two is necessary to understand the total device latency contributed to response times. Specifically, without further optimization, response times are currently recorded as the time from when a stimulus is displayed on

[1] All response times reported in this paper also contain latency from LaTARbot itself. However, these are likely negligible in the overall context of measuring smartphone latency. For additional details, see hardware and firmware design repositories (https://github.com/CTR-Lab-WashU/latar_hardware; https://github.com/CTRLab-WashU/latar_firmware).

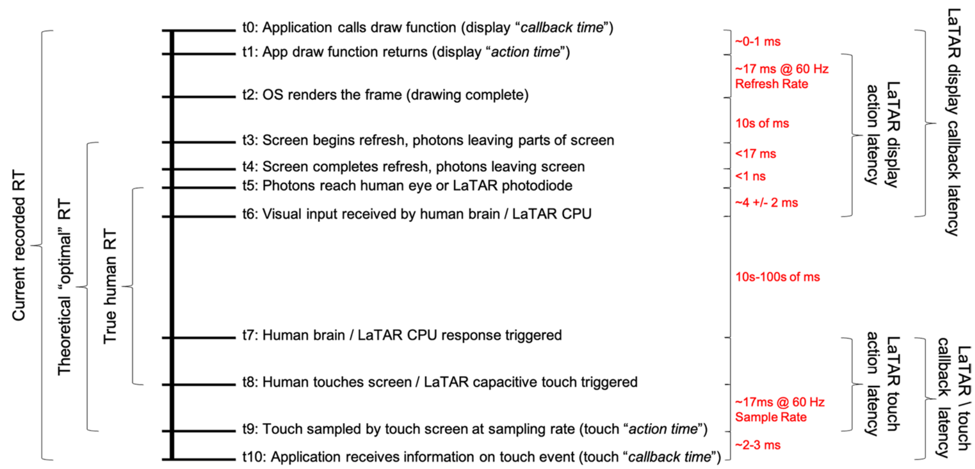

Fig. 3 Measured response time event timeline. *Note.* Timeline of events included in device-recorded response times

the screen (which includes the device's display latency) to when the participant taps the screen (which includes the device's tap latency), as shown in Fig. 3. Therefore, although display and tap latencies are presented separately for a more fine-grained analysis, discussion focuses on the total latency contributed by the device (i.e., the sum of each device's display and tap latency) to provide readers with a wholistic picture of the contribution of device latencies on response time data. Additionally, both action and callback times (again, for both display and touch latencies) are explored to address the hypothesis that manufacturers, OS types, and devices may influence callback latencies more so than action latencies since, as mentioned above, callback times are subject to additional OS delays whereas the action times are not.

Because the timestamps are collected on separate devices, there were differences in both (1) the network latency between the server and each device (i.e., server-to-LaTARbot and server-to-smartphone) and (2) the onboard clock time of all three devices. To account for these latencies and clock differences, a clock sync procedure was run at the start of each test to identify an offset that should be applied to every timestamp from each device. Clock sync is described in further detail in the Procedure section that follows.

Procedure First, the LaTARbot application was installed on each smartphone and all other applications were closed out. Then, the phone was connected to the server's ad hoc Wi-Fi network. Using the server GUI, the testing procedure was configured to run 100 display and 100 capacitive tap samples at 199 ms intervals each. We used a prime number interval between samples (199 ms; for both tap and display samples) to minimize the effect of aliasing (see "aliasing" from signal processing theory for more information) and more accurately reflect the true behavior of the device. The automated test was run after a clock sync procedure was used to get all three components running on the same clock.

During the clock sync, offset values were calculated for the mobile device and LaTARbot, which were then applied to the collected data to bring the timestamps from both devices into the same frame of reference. The offset values were calculated based on an implementation of the Network Time Protocol clock synchronization algorithm (Mills et al., 2010). Thirty cycles of the synchronization algorithm were run between the server and the LaTARbot and 255 synchronization cycles were run between the server and the smartphone. More cycles were needed for the smartphone due to the higher variability in latency over Wi-Fi versus over a hardwired USB connection. Assuming symmetric latency from server-to-device and device-to-server, this procedure synchronized the clocks between the server and phone to within 0.6 ms, and between the server and LaTARbot to within 0.6 ms. This gives us a worst-case error margin of 1.2 ms on each sample. The output of the clock sync (i.e., the average offset) was used to adjust the timestamps from the phone and LaTARbot. Phrased differently, because multiple devices were necessary to collect the data of interest, each running their own clock, knowing the offset between each device's clock was critical to correctly synchronize timestamps from all three devices and extract the relevant latencies of interest. The output of the Network Time Protocol sync was an offset for each device's clock relative to the server (laptop) clock. For example, if the LaTARbot clock indicated it was 12:01:20.20 while the smartphone clock reads the same instance in time as 12:01:23.70, a 3.5 second offset would be applied to "synchronize" the data point timestamps. While critical for the present study given the dependent variables of interest (namely, device latencies), other BYOD studies need not worry about this issue such that this procedure would not be relevant or plausible for any study involving human participants interacting with a device.

Data processing and analysis All code for data collection and processing are available on GitHub (https://github.com/CTRLab-WashU/latar_firmware; https://github.com/CTRLab-WashU/latar_hardware; https://github.com/CTRLab-WashU/latar_android; https://github.com/CTRLab-WashU/latar_ios; https://github.com/jnicosia/latarprocessing) and OSF (https://osf.io/ncjta/). Data were preprocessed using a Python script which corrected the timestamps, aligned corresponding stimuli and responses, and converted the JSON files to CSVs to be read into R for analysis. The script corrected the timestamps from each device (smartphone and LaTARbot) for the offset time between the mobile device's time and the server's time. To get the offset time, we found the average and standard deviation from each clock sync trip and got the average after discarding values exceeding ±2 standard deviations due to the complexities of network communications.

All data analysis was performed in R (R Core Team, 2012). We examined the influence of various device characteristics on their display and capacitive touch latencies. Because many of the phone characteristic variables (i.e., Geekbench 5 Performance Score[2], Touch/Display Refresh Rate, phone age, phone price, etc.) were highly correlated with one another, we used simple correlations, *t*-tests, and fixed-effect analyses of covariance (ANCOVAs; rather than multiple regression models) to investigate which variables had the greatest influence on the device latency metrics and whether device latencies differed between Android and iOS. Dependent variables of interest included action and callback latencies for both display and capacitive touch (see Fig. 3 for definitions of each).

To maintain data quality, we excluded response latencies that may have resulted from technical problems. Specifically, latencies outside ±1.5 standard deviations from the device's mean latency for that specific condition were removed. This procedure removed 1.42% of the capacitive action latencies, 1.23% of the capacitive callback latencies, 1.97% of the display action latencies, and 1.97% of the display callback latencies. Removal of these outliers produced data consistent with the devices' purported refresh rate (this is discussed further in the Results section).

[2] Geekbench 5 measures the performance of a device by performing tests that are representative of real-world tasks and applications. Higher scores are better, with double the score indicating double the performance. See https://www.geekbench.com/doc/geekbench5-cpu-workloads.pdf for more information.

Results

First, we present evidence supporting the validity of the LaTARbot apparatus and setup. Second, we examine the device display and capacitive touch latencies for both action and callback times addressing the hypothesis that manufacturers, OS types, and devices may influence callback latencies more so than action latencies because the callback times are subject to additional delays (due to complexities of non-real-time operating systems which are outside the scope of this paper) whereas the action times are not. More broadly, however, we sought to investigate the influence of the phone characteristic variables on these device latencies to provide researchers with useful data and recommendations to optimize their digital cognitive assessment protocols. Thus, we examined the relationships between device characteristics (i.e., Geekbench 5 performance score, touch/display refresh rate, phone age, and phone price) and device latency metrics. Finally, we examined whether device characteristics and latency differed by OS.

Devices Devices to be included were based on responses from a previous technology survey conducted by our laboratory (Nicosia et al., 2021a, b). We included as many of the most popular phones in the US, ranging in price, as possible based on the survey results and purchasing availability. See Table 2 for the characteristics of each device included.

Mean device latencies Figure 4 shows the display (a) and capacitive touch action (b) latencies for the iPhone 11 as an example of the raw, sample-level data that was collected for each phone. As shown, the actual difference between the maximum and minimum touch latencies elicited from the LaTARbot setup closely approximated what would be expected based on the device's refresh rates—providing some validation data for the LaTARbot setup. For example, the iPhone 11 has a touch sampling rate of 120 Hz. Therefore, one period is equal to 1/120 Hz or 0.008333 seconds (8333 microseconds; µs) per cycle. In addition to providing a preliminary validity check of the system and data, Fig. 4a and b also include a solid grey line indicating the mean device latency to demonstrate how the average latencies were derived for the other analyses described in the paper.

Figure 5 shows the mean display (a) and capacitive (b) latencies for each device tested in the present study. As shown, devices differed significantly in both their display action, $M = 59{,}620$ µs (equivalent to 59.62 ms), confidence interval $(CI) = [55{,}459,\ 63{,}781]$, $F(25,\ 1265) = 68.39$, $p < 0.001$, $\eta^2 = 0.57$, and capacitive touch action latencies, $M = 14{,}586$ µs (equivalent to 14.59 ms), $CI = [12{,}064,\ 12{,}108]$, $F(25,\ 2537) = 380.10$, $p < 0.001$, $\eta^2 = 0.79$. Altogether, when we look at the total amount of time contributed

Table 2 Device characteristics

Device	OS	Geekbench 5 performance score	Touch sample rate (Hz)	Display refresh rate (Hz)	Cost (MSRP; USD)	Release year
LG Journey	Android	107	90	60	89	2019
Motorola moto g stylus	Android	544	120	60	299	2021
Google Pixel 3a	Android	342	120	60	399	2019
Google Pixel 5a	Android	585	180	60	449	2021
Google Pixel 6	Android	979	180	90	599	2021
Google Pixel 6 Pro	Android	1025	240	120	899	2021
Samsung Galaxy A01	Android	145	120	60	149	2020
Samsung Galaxy A30	Android	269	120	60	178	2019
Samsung Galaxy A32	Android	346	180	60	278	2021
Samsung Galaxy A51	Android	342	120	60	349	2019
Samsung Galaxy A52	Android	538	180	60	499	2021
Samsung Galaxy S9	Android	510	120	60	719	2018
Samsung Galaxy S10	Android	676	120	60	233	2019
Samsung Galaxy S20	Android	901	120	60	490	2020
Samsung Galaxy S21	Android	1088	120	120	499	2021
Samsung Galaxy Note9	Android	509	120	60	400	2018
Samsung Galaxy Note10	Android	725	120	60	550	2019
Samsung Galaxy Note20	Android	910	120	60	608	2020
iPhone 7	iOS	778	60	60	649	2015
iPhone X	iOS	932	120	60	999	2017
iPhone XS	iOS	1120	120	60	999	2018
iPhone XR	iOS	1118	120	60	749	2018
iPhone 11	iOS	1334	120	60	699	2019
iPhone 12	iOS	1595	---	---	699	2020
iPhone SE (2020)	iOS	1333	---	---	399	2020
iPhone 13	iOS	1749	---	---	799	2021

Apple stopped publishing sampling and display refresh rates after the iPhone 11; see Apple Documentation Archive here: https://developer.apple.com/library/archive/documentation/DeviceInformation/Reference/iOSDeviceCompatibility/Displays/Displays.html

by the device (i.e., display and touch latencies combined), this could range anywhere from 35 to 140 ms (minimum and maximum of combined display and touch latencies) in device latency[3] and, more practically, could range from 60 to 90 ms (first and third quartiles of combined display and touch latencies). Thus, in the context of experimental reaction time paradigms, which generally elicit response times around 200 to 300 ms, it is likely that within- and between-person differences in device latencies could significantly influence response time estimates (see Table 1). The context in which both device display and tap latencies play into measured response times is illustrated in Fig. 3 and is further described in the Discussion section.

In addition to the action latencies, the callback latencies are also presented in Fig. 5 in the lighter-colored bars. As mentioned above, because callback latencies (but not action latencies) are subject to additional delays in the OS (for both display and touch), we sought to test for differences in the influence of latency type (i.e., action vs. callback) on the observed latencies across devices. As shown, devices differed significantly in both their display callback, $M = 60{,}383$ µs (equivalent to 60.38 ms), $CI = [56{,}021,\ 64{,}745]$, $F(25, 1265) = 74.68$, $p < 0.001$, $\eta^2 = 0.60$, and capacitive touch callback latencies, $M = 17{,}870$ µs (equivalent to 17.87 ms), $CI = [15{,}410,\ 20{,}330]$, $F(25, 2542) = 286.80$, $p < 0.001$, $\eta^2 = 0.74$. Importantly, for capacitive touch, latency type (action vs. callback) interacted with phone model, $F(25, 5079) = 40.87$, $p < 0.001$, $\eta^2 = 0.05$, indicating that some

[3] To arrive at the reported 35 ms minimum, the minimum display and minimum touch latencies were summed for each phone and the minimum across all phones was calculated and reported. To arrive at the reported 140 ms maximum, the maximum display and maximum touch latencies were summed for each phone and the maximum across all phones was calculated and reported. This represents the widest range one might anticipate based on the data collected here.

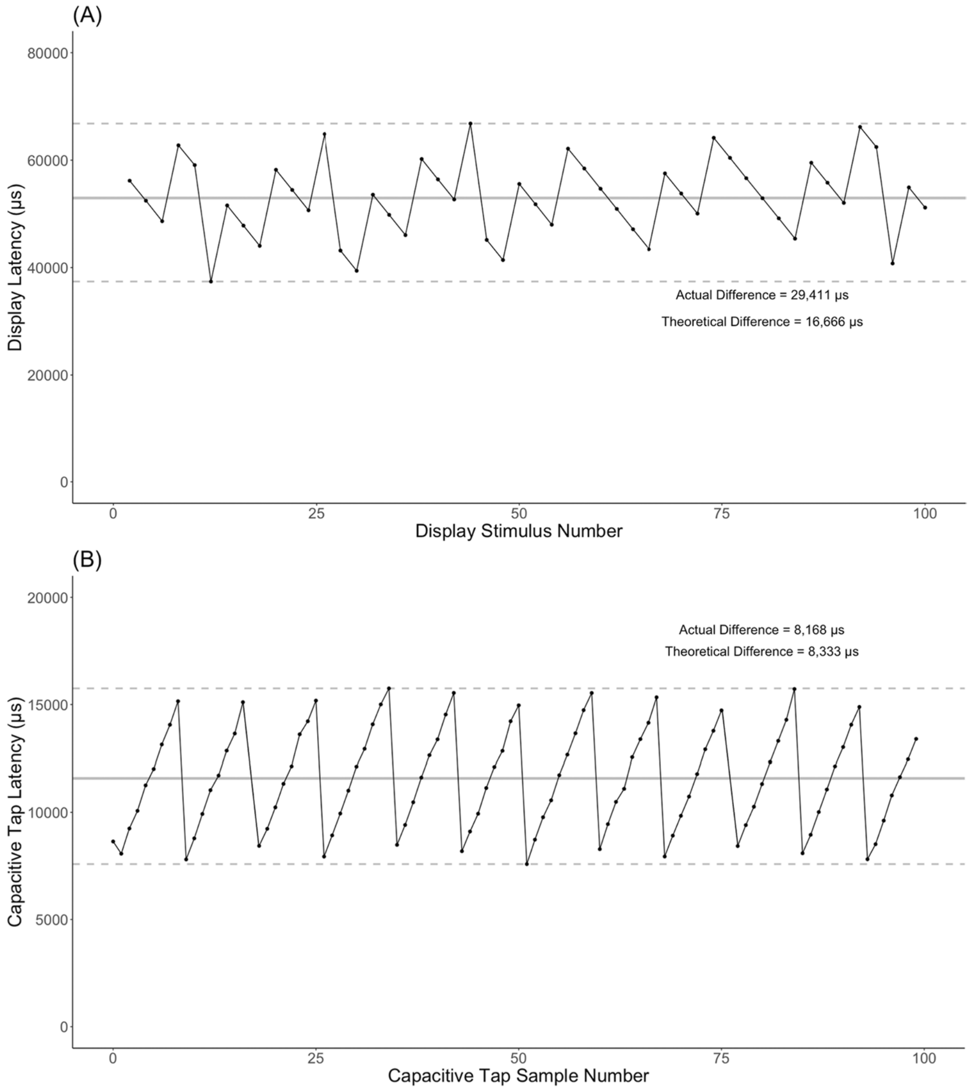

Fig. 4 Example sample-level data. Example sample-level (display and action tap latency) data from the iPhone 11. The sawtooth patterns appear as a result of the action, drawing or tapping, happening at various timepoints within each frame or sampling period, respectively. (**A**) iPhone 11 (60 Hz Display Refresh Rate). (**B**) iPhone 11 (120 Hz Touch Sample Rate)

devices had larger differences in action and callback latencies than others (this interaction did not approach significance for the display latencies, $p = 0.99$). Consistent with our hypothesis, it is important to ensure that the application used to collect data records the action times (for *both* display and touch) rather than callback times to minimize OS- and device-related latency differences.

Device characteristic and latency correlations In order to investigate how device characteristics may influence latencies, we examined the relationships amongst several device characteristic variables and our latency metrics. The device characteristics we explored here included each device's (1) Geekbench 5 performance score, which serves as a measure of CPU performance with higher scores indicating better performance, (2) "age" or years since its initial release date, and (3) cost (MSRP where published, otherwise price on Amazon in September 2021) in US dollars (USD). We examined the correlations amongst these device characteristic variables and several latency measures (i.e., display action latency, display callback latency, display callback-action latency difference, capacitive touch action latency, capacitive touch callback latency, and capacitive touch callback-action latency difference). As shown in Fig. 6a and b, device characteristics had a greater influence on display latencies (and the difference between display callback and action latencies) than on the capacitive touch latencies. Specifically, there were significant negative correlations between the Geekbench 5 performance score and the phone cost and display latency indicating that the more expensive phones

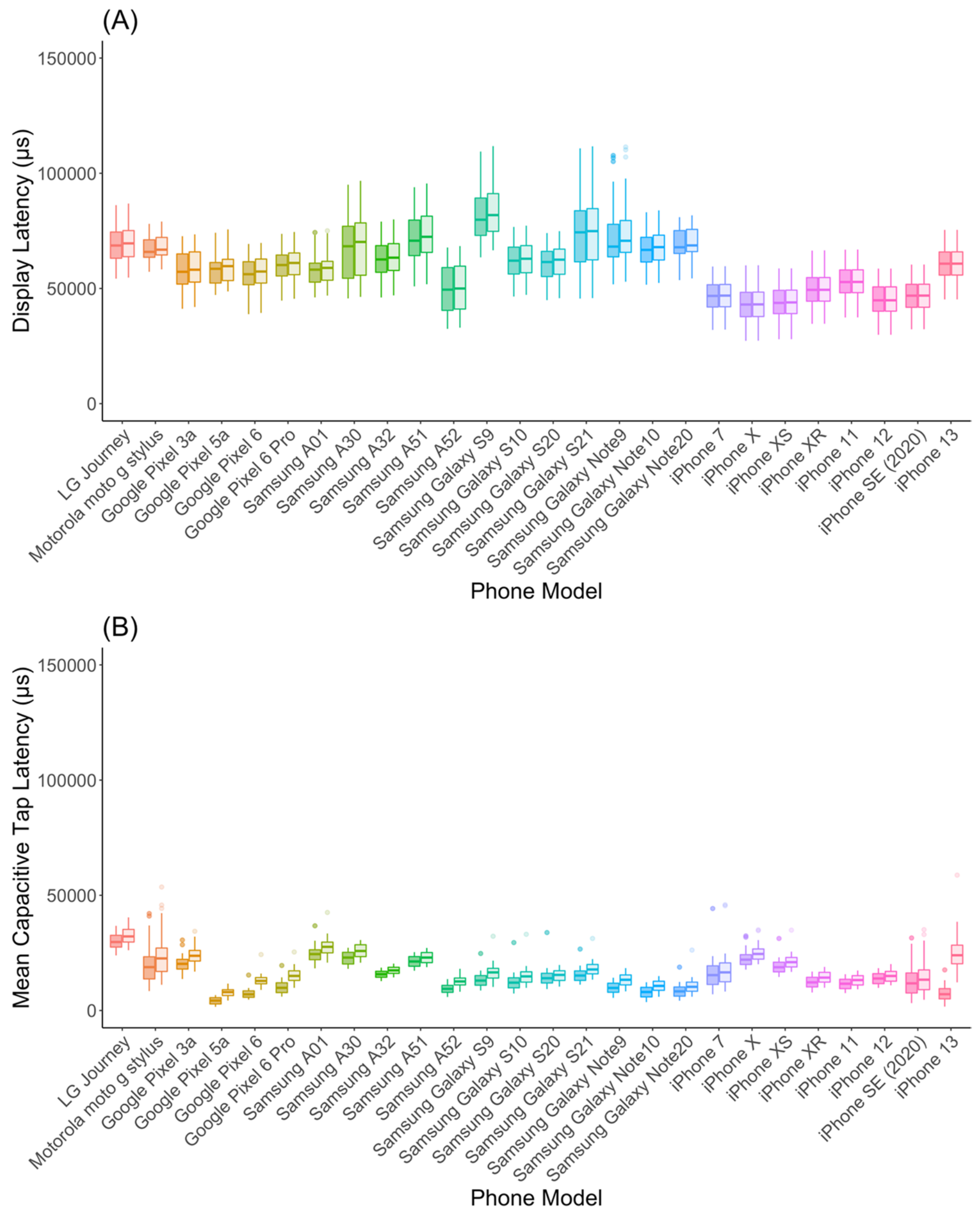

Fig. 5 Device latencies. Box plots of device display and capacitive touch latencies. Lighter shaded bars indicate callback latencies while darker shaded bars indicate action latencies. (**A**) Mean device display latencies. (**B**) Mean device tap latencies

with better-performing CPUs had smaller display latencies, $rs = -0.47$ and -0.44, respectively, $ps < 0.05$. Additionally, the Geekbench 5 performance score and phone cost were negatively correlated with the callback-minus-display latency difference score, $rs = -0.65$ and -0.43, respectively, $ps < 0.01$, suggesting that more expensive phones with better-performing CPUs had smaller differences between their callback and action latency times. Interestingly, however, these correlations were not present for the capacitive touch latency measures (see Fig. 6b).

OS differences Next, we sought to investigate the influence of OS on our latency metrics. Table 3 displays OS differences in device characteristics and latencies. As shown, Android and iOS devices differed significantly on all the display latency measures, $ps < 0.001$, $ds > 2.23$, and phone cost, $p = 0.002$, $d = 1.58$, although OS did not influence capacitive touch latencies. Indeed, both the cost and display latency differences across OS types were anticipated such that iPhones (which run iOS) were consistently more expensive than Android devices and also have different

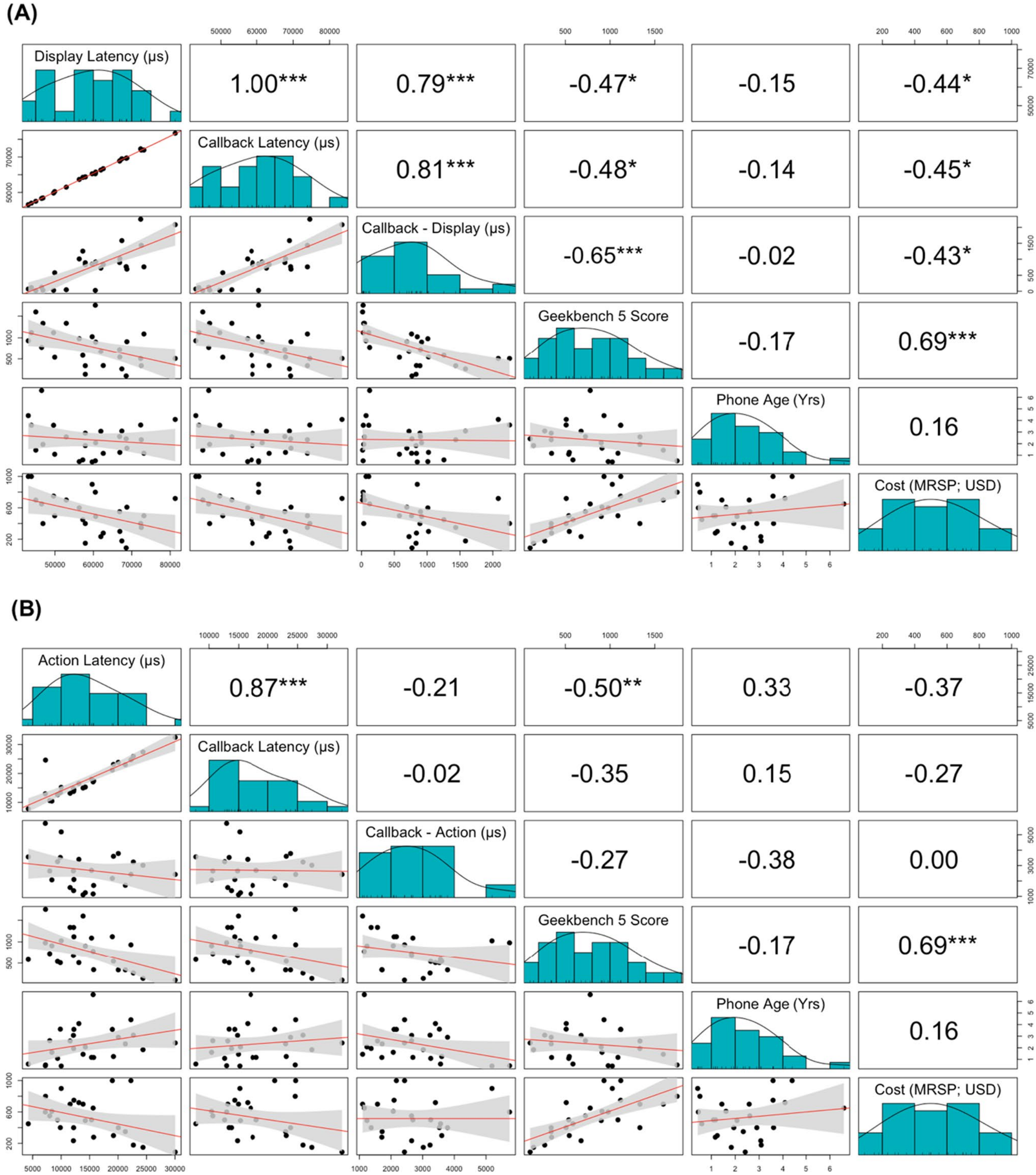

Fig. 6 Device characteristics and latency correlations. Correlations amongst device characteristics and display and touch latencies. Pearson's r for the variable below and to the left are presented in the top diagonal with * indicating $p < 0.05$, ** indicating $p < 0.01$, and *** indicating $p < 0.001$. Scatterplots for the variable above and to the right are presented in the bottom diagonal. (**A**) Device characteristics and display latency correlations. (**B**) Device characteristics and capacitive tap latency correlations

Table 3 OS differences

	Android, $N=18^a$	iOS, $N=8^a$	p-value[b]
Action latency (µs)	14,735 (6948)	14,250 (4676)	0.84
Tap callback latency (µs)	17,824 (6705)	17,973 (4829)	0.95
Tap callback - action latency difference (µs)	3082 (1137)	3,579 (5336)	0.80
Display latency (µs)	64,539 (7679)	48,552 (5795)	<0.001
Display callback latency (µs)	65,616 (7970)	48,609 (5777)	<0.001
Display callback - display latency difference (µs)	1077 (473)	57 (41)	<0.001
Geekbench 5 performance score (higher = better)	586 (302)	1245 (325)	<0.001
Touch refresh rate (Hz; higher = better)	138 (37)	108 (27)	0.072
Display refresh rate (Hz; higher = better)	68 (20)	60 (0)	0.10
Phone age (years; since release date)	1.99 (1.10)	3.08 (1.90)	0.16
Phone cost (MSRP; USD)	427 (208)	749 (195)	0.002

a Mean (standard deviation)

b Welch two-sample *t*-test

implementations when it comes to calling user interface draw functions. Additionally, there was a significant difference in OS type in the (display) callback-minus-display latency difference score, $p < 0.001$, $d = 2.56$, which reflects the nature of how different OS types handle recording action and callback latencies and, again, highlights the importance for researchers to use action rather than callback times for touch input and callback rather than display times for the display to acquire the most precise response time.

Discussion

In the present manuscript, we investigated display and touch latencies across a series of popular smartphones and operating systems with the goals of (1) better understanding device-driven sources of variability that could affect smartphone-based cognitive assessments and (2) providing researchers with a set of recommendations to increase the accuracy of their data. Ultimately, we found that there is considerable variability across smartphone devices in display and capacitive touch latencies which, if unaccounted for, could be misattributed to individual differences in response times. Second, there were significant relationships between device latencies and CPU performance and cost, suggesting (as expected) that higher-performing and more expensive phones have smaller latencies. Finally, there were several differences in display latencies across OS types. Our results suggest that, despite the advantages offered by smartphone-based cognitive assessments in cognitive research, investigators employing the BYOD model should collect additional measures about participants' phones and consider adjusting response times by including device characteristic covariates in higher-level analyses. Additionally, there are at least three experimental design strategies that can be implemented to reduce the influence of device latencies (some recommendations shown in Table 1). The first and most restrictive option is to provide all study participants with a single device type running the same OS and OS version. The second option is to restrict recruitment to participants with a specific set of devices and OS. However, it should be noted that restricting recruitment to participants with specific devices and/or OSs may render the study vulnerable to potential recruitment biases. Third, researchers may consider (1) designing and using tasks that don't rely on high-precision response time measurements, (2) employing tasks that include a "baseline" condition (see Pronk et al., 2020), or (3) avoiding response time metrics altogether and instead focusing on throughput measures like accuracy.

Our results extend upon Passell et al. (2021) in several important ways. First, the present study directly examined device-related latencies and corroborates the notion that differences in mobile cognitive test performance could represent both the effects of the devices themselves and differences introduced by users. Because we used an automated device and removed the human component (and variance) from the equation, our results directly show the magnitude and variability of the devices' latencies. Second, in addition to examining touch latencies, we also presented display latencies. Display latencies are critical for any experiment involving response time analyses given that delays in presentation of stimuli on the screen are lumped into the recorded response time, hence adding more noise to the data. As shown in Fig. 3, any measured response time consists of device display latency, the true human response time, and device input latency. Of these three components, the display end of the timeline appears to have the most room to improve upon with respect to honing response time precision.

Recommendations for researchers There are several main takeaways from the present study for researchers looking to optimize their digital assessments and increase the precision of their data. First, although both in-lab and mobile experiments contain substantial error imparted from the collection-device, when it comes to assessing participants' "true response time" (see Fig. 3), there are several ways to increase data collection precision. Rather than calculating response times as the time from when the application draws the image on the screen to the touch callback time, the time from display action time to touch action time should be used. Better yet, investigators may want to work with developers to acquire t2 or t3 from Fig. 3 to further close the gap (by tens of milliseconds) between the recorded response time and the "true" response time.

Perhaps a more practical suggestion for researchers conducting digital cognitive assessments is to simply understand the proportion of each recorded response time which may be due to device-driven latencies. Specifically, because it's possible that latencies contributed by different phones could add up to around 100 ms of variation in response times, based on the present data, then any main effects or interactions with an absolute difference less than 150 ms should be carefully considered. If this magnitude of an effect is expected, then investigators may want to supply participants with a single, specified device type and OS rather than employing a BYOD design (see Table 1). Regardless, it is critical that researchers allowing participants to use their own devices collect device characteristic data (e.g., make/model, OS version, etc.) and include this information as covariates when reaction times (RTs) are primary outcome variables.

Echoing some of the suggestions put forth by Passell et al. (2021), if it is possible to use outcome measures other than response times (such as accuracy, Euclidian distance, etc.), this would help to avoid many of the device-related sources of variation. When response times are necessary, z-score transforming each individual's data based on their own overall mean response latency and standard deviation is recommended (Faust et al., 1999; Nicosia et al., 2021a, b). This z-score transformation places all participants on the same scale so that one can then use standard ANOVAs and regressions on the z-transformed response times to determine if individual and group differences in any manipulation are due to general slowing, device-related effects, and/or group- or age-specific deficits. Some potential benefits of this approach include the ability to investigate higher-order effects after removing the influences of processing speed and device-related effects. Additionally, Pronk et al. (2020) recommend employing within-participant designs where possible to avoid having to make comparisons between participants with different devices, operating systems, and browsers.

Limitations The findings of this study should be considered in light of a number of broader considerations and limitations which may be addressed in future studies. First, although we know that the latencies introduced by the LaTARbot are relatively small compared to the latencies we see in the devices, the exact values presented here should not be taken as a constant offset to simply subtract from one's response times. Second, the results in this paper are specifically for iOS applications using UIkit and Android applications using Android Views and thus does not fully apply to experiments which may have been run in a web browser or applications that use lower-level graphics (such as OpenGL) or game development engines (like Unity). Third, our display task was extremely basic (i.e., simply switching between an all-black and an all-white background) and the current results do not illustrate potential effects of dropped frames which may occur in more graphics-intensive programs. Fourth, the application had a relatively constant and minimal CPU load and thus cannot generalize findings to programs which may be more demanding. Fifth, the devices tested here did not have any additional user applications installed or running in the background (e.g., sharing location, streaming music, etc.) whereas this may be an additional factor in BYOD studies. Finally, we did not test every device available on the market today, though we aimed to test as many presently popular devices as possible. With such a quickly evolving technology market, these results are most pertinent to researchers aiming to conduct smartphone studies using similar versions of operating systems and hardware configurations.

Conclusions Ultimately, our findings suggest that there is considerable latency included in device-recorded response times and that there is a substantial amount of variability across devices that should be accounted for. Although digital cognitive assessments are advantageous compared to in-lab assessments for many reasons (including reduced recall bias, higher ecological validity, increased accessibility, reduced recruitment barriers, increased engagement, etc.), it is critical that investigators seriously weigh their options when it comes to the methodological details of their study. If main effects or interactions could possibly be due to response time differences less than 150 ms, then a preselected device should be considered and provided to participants. All investigators conducting digital assessments should collect device information and include (at the very least) phone make and model as a covariate in statistical models to account for some of the variance introduced by device-driven latencies. By carefully considering whether device variability could impact their results and taking steps to mitigate these effects, researchers can take advantage of bring-your-own-device digital assessments to increase research participation and engagement.

Acknowledgements This work was funded by the following grants from the National Institutes of Health: U2C AG060408, P01 AG003991, and R01 AG057840. Additional funding was provided by the BrightFocus Foundation grant A2018202S. We would like to thank the dedicated software engineers at happyMedium (www.thinkhappymedium.com), specifically Steven Huff and Michael Votaw for their creativity, ingenuity, and dedication to this project.

Open Practices Statement All code for data collection and processing are available on GitHub (https://github.com/CTRLab-WashU/latar_firmware; https://github.com/CTRLab-WashU/latar_hardware; https://github.com/CTRLab-WashU/latar_android; https://github.com/CTRLab-WashU/latar_ios; https://github.com/jnicosia/latarprocessing) and OSF (https://osf.io/ncjta/).

References

de Bruijne, M., & Wijnant, A. (2013). Comparing survey results obtained via mobile devices and computers: an experiment with a mobile web survey on a heterogeneous group of mobile devices versus a computer-assisted web survey. *Social Science Computer Review, 31*(4), 482–504.

de Liaño, B. G.-G., León, O. G., & Pascual-Ezama, D. (2012). Research participation improves student's exam performance. *The Spanish Journal of Psychology, 15*(2), 544–550.

Fahrenberg, J. (2006). Assessment in daily life. A review of computer assisted methodologies and applications in psychology and psychophysiology, years 2000–2005.

Faust, M. E., Balota, D. A., Spieler, D. H., & Ferraro, F. R. (1999). Individual differences in information-processing rate and amount: Implications for group differences in response latency. *Psychological Bulletin, 125*(6), 777–799.

Foxlin, E. (2002). Motion tracking requirements and technologies. In *Handbook of virtual environments: Design, implementation and applications* (pp. 163–210). Lawrence Erlbaum.

Germine, L., Nakayama, K., Duchaine, B. C., Chabris, C. F., Chatterjee, G., & Wilmer, J. B. (2012). Is the Web as good as the lab? Comparable performance from Web and lab in cognitive/perceptual experiments. *Psychonomic Bulletin & Review, 19*(5), 847–857.

Germine, L., Reinecke, K., & Chaytor, N. S. (2019). Digital neuropsychology: Challenges and opportunities at the intersection of science and software. *The Clinical Neuropsychologist, 33*(2), 271–286.

Gosling, S. D., & Mason, W. (2015). Internet research in psychology. *Annual Review of Psychology, 66*(1), 877–902.

Hamaker, E. L., & Wichers, M. (2017). No time like the present: Discovering the hidden dynamics in intensive longitudinal data. *Current Directions in Psychological Science, 26*(1), 10–15.

Henrich, J., Heine, S. J., & Norenzayan, A. (2010). The weirdest people in the world? *The Behavioral and Brain Sciences, 33*, 61–135.

Henze, N., Funk, M., & Shirazi, A. S. (2016). Software-reduced touchscreen latency. Paper presented at the Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services, Florence, Italy.

Jain, A., Bansal, R., Kumar, A., & Singh, K. D. (2015). A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students. *International Journal of Applied & Basic Medical Research, 5*(2), 124–127.

Kraut, R., Olson, J., Banaji, M., Bruckman, A., Cohen, J., & Couper, M. (2004). Psychological research online: report of Board of Scientific Affairs' Advisory Group on the conduct of research on the internet. *American Psychologist, 59*(2), 105–117.

Mehl, M. R., & Conner, T. S. (Eds.). (2014). *Handbook of research methods for studying daily life* (Paperback ed.). Guilford.

Mills, D., Martin, J., Burbank, J., & Kasch, W. (2010). Network time protocol version 4: Protocol and algorithms specification.

Nicosia, J., Aschenbrenner, A. J., Adams, S., Tahan, M., Stout, S. H., Wilks, H., … Hassenstab, J. (2021a, December 1). Bridging the technological divide: Stigmas and challenges with technology in clinical studies of older adults. https://doi.org/10.31234/osf.io/yeufs

Nicosia, J., Cohen-Shikora, E. R., & Balota, D. A. (2021b). Re-examining age differences in the Stroop effect: The importance of the trees in the forest (plot). *Psychology and Aging, 36*(2), 214–231.

Nosek, B. A., Banaji, M. R., & Greenwald, A. G. (2002). E-Research: ethics, security, design, and control in psychological research on the internet. *Journal of Social Issues, 58*(1), 161–176.

Passell, E., Strong, R. W., Rutter, L. A., Kim, H., Scheuer, L., Martini, P., et al. (2021). Cognitive test scores vary with choice of personal digital device. *Behavior Research Methods, 53*(6), 2544–2557.

Pavlovych, A., & Gutwin, C. (2012). Assessing target acquisition and tracking performance for complex moving targets in the presence of latency and jitter. Paper presented at the Proceedings of Graphics Interface 2012, Toronto, Ontario, Canada.

Pronk, T., Wiers, R. W., Molenkamp, B., & Murre, J. (2020). Mental chronometry in the pocket? Timing accuracy of web applications on touchscreen and keyboard devices. *Behavior Research Methods, 52*(3), 1371–1382.

R Core Team. (2012). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. http://www.R-project.org/. Accessed March 2021.

Reips, U.-D. (2000). Chapter 4 - The web experiment method: advantages, disadvantages, and solutions. In M. H. Birnbaum (Ed.), *Psychological experiments on the internet* (pp. 89–117). Academic Press.

Sliwinski, M. J. (2008). Measurement-burst designs for social health research. *Social and Personality Psychology Compass, 2*(1), 245–261.

Sliwinski, M. J., Mogle, J. A., Hyun, J., Munoz, E., Smyth, J. M., & Lipton, R. B. (2018). Reliability and validity of ambulatory cognitive assessments. *Assessment, 25*(1), 14–30.

Trull, T. J., & Ebner-Priemer, U. (2014). The role of ambulatory assessment in psychological science. *Current Directions in Psychological Science, 23*(6), 466–470.

Wilkinson, R. T., & Allison, S. (1989). Age and simple reaction time: decade differences for 5,325 subjects. *Journal of Gerontology, 44*(2), 29–35.

Woods, A. T., Velasco, C., Levitan, C. A., Wan, X., & Spence, C. (2015). Conducting perception research over the internet: a tutorial review. *PeerJ, 3*, e1058.