Titre: Title:	Usability of Avionic Touchscreens Under Vibration: Supported versus Freehand Target Selection in Cockpit Conditions
Auteur: Author:	Adam Jonathan Schachner
Date:	2022
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Schachner, A. J. (2022). Usability of Avionic Touchscreens Under Vibration: Supported versus Freehand Target Selection in Cockpit Conditions [Master's thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/10321/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/10321/
Directeurs de recherche: Advisors:	Philippe Doyon-Poulin
Programme: Program:	Maîtrise recherche en génie industriel

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Usability of Avionic Touchscreens Under Vibration: Supported versus Freehand Target Selection in Cockpit Conditions

ADAM JONATHAN SCHACHNER

Département de mathématiques et de génie industriel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

Génie industriel

Mai 2022

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

Usability of Avionic Touchscreens Under Vibration: Supported versus Freehand Target Selection in Cockpit Conditions

présenté par **Adam Jonathan SCHACHNER**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment acceptée par le jury d'examen constitué de :

Daniel IMBEAU, président

Philippe DOYON-POULIN, membre et directeur de recherche

Karyn MOFFAT, membre

ACKNOWLEDGEMENTS

My supervisor, Professor Philippe Doyon-Poulin, has been amazing and provided frequent and invaluable support on all aspects of the project. My mom, dad and brother are always there for me in everything I do. Professor Daniel Imbeau provided advice on aspects related to physical ergonomics, including how to evaluate the impact of whole-body vibration. Alexandre Ferreira Benevides and Marc Charbonneau at the LESIAQ Laboratory provided guidance during the design and fabrication of the test setup, and stayed at the lab longer hours than they needed to, to accompany me, every time I went. Nader Ammari helped troubleshoot the D-Box chair and helped set up the coding environment so that I could program its motion. Norman Fisher provided help with design concepts for the test setup. While designing and building the test setup, I would call Norm every two weeks or so to run ideas by him, and every conversation was illuminating. Arthur Bevier worked with me and Professor Doyon-Poulin during a 3-month remote internship. He wrote a script to organize the raw data collected as part of this study, to make it easier for analysis, as well as started working on future aspects of the project not addressed in this dissertation, such as designing different soft-keyboard options for use in the cockpit. I contacted Maxence Hébert-Lavoie whenever I had questions about statistical analysis tests, and he was always quick to respond with helpful advice.

My research was funded by Mitacs and CMC Electronics Inc., as part of the Mitacs Accelerate program. In addition, Mitacs recognized my work on this project by giving me the 2021 Mitacs Award for Outstanding Innovation – Master's. Research funding was also provided by the Natural Sciences and Engineering Research Council of Canada, as part of their Canada Graduate Scholarships- Master's program.

The two avionic touchscreens used in this study were provided by CMC Electronics. I would like to express my extreme gratitude to CMC Electronics employees Nami Bae, Sophie Duchesne and Pascal Gagnon. Pascal help me properly interface with CMC's touchscreens, and explained about different touchscreen technologies. Sophie and Nami made sure I got the equipment and information I needed, provided details on touchscreen placement in the cockpit, and validated the experimental procedure and setup. They also provided encouragement, throughout the project.

This research would not have been possible without the 24 participants who gave several hours of their time, during a global pandemic, to come and take part in this study.

RÉSUMÉ

(Problème) Dans le poste de pilotage, les écrans tactiles peuvent être éloignés du pilote et utilisés dans des conditions de vibration ou de turbulence. Des travaux précédents ont trouvé que les écrans tactiles offrent un débit (*throughput*; une métrique de performance capturant à la fois la vitesse et la précision) plus élevé dans des conditions statiques par rapport à d'autres dispositifs testés, mais que leur performance et leur taux d'erreur a dégradé plus rapidement sous l'effet des vibrations. L'utilisation d'un support manuel pour atténuer cet effet a été suggérée par la norme SAE ARP60494. Il est nécessaire de quantifier l'impact des vibrations sur la sélection des cibles à l'écran tactile dans le poste de pilotage, ainsi que de mesurer l'utilité du support manuel, en utilisant une méthodologie normalisée. **(Objectif)** Nous avons mesuré les effets de la vibration sur la sélection des cibles de l'écran tactile à l'aide d'une méthode normalisée, ce qui permet de comparer nos résultats avec ceux d'autres études, et comparé le support manuel (utilisant le pouce, en tenant le bord de l'écran, versus utilisant l'index, avec main libre), les écrans tactiles avioniques et commerciaux, et les positions de l'écran tactile. **(Méthode)** 24 participants ont effectué une tâche de sélection multidirectionnelle ISO 9241-411 (une tâche de sélection standardisée de Fitts). Nous avons bâti une plateforme d'essai réglable qui vibre, avec des écrans tactiles positionnés dans une géométrie représentative du poste de pilotage. Les participants ont été exposés à des niveaux de vibration représentatifs du vol en hélicoptère. Nous avons testé quatre écrans tactiles, deux positions d'écran, deux méthodes de support manuel et deux niveaux de vibration. Nous avons mesuré le débit de sélection des cibles, le taux d'erreur et la préférence subjective. **(Résultats et discussion)** Nous avons trouvé des valeurs moyennes de débit de 6,5 bits/sec sans vibration, contre 5,7 bits/sec avec vibration, ce qui est plus élevé que les valeurs de débit pour d'autres dispositifs d'entrée rapportées par les travaux précédents. Les taux d'erreur moyens étaient élevés : 10,3 % sans vibration, contre 16,6 % avec vibration. Comme dans les travaux précédents, nous avons constaté une augmentation exponentielle du taux d'erreur lorsque la taille de la cible diminue, ce qui souligne l'importance d'utiliser des cibles de taille appropriée. Dans des conditions statiques, le support manuel a clairement nui à l'utilisation de l'écran tactile à main levée. Dans des conditions de vibration, le débit était plus faible et le taux d'erreur similaire lors de l'utilisation du support manuel par rapport à l'utilisation à main levée. Nous n'avons donc pas trouvé de preuves de l'avantage d'un support manuel lors de l'exécution d'une tâche de sélection multidirectionnelle sur un écran tactile soumis à des vibrations. Dans des conditions de vibration, la position du piédestal

a été meilleure que la position sur le tableau de bord principal (débit plus élevé et taux d'erreur plus faible). Dans des conditions statiques, les deux positions ont donné des résultats similaires. Les écrans tactiles avioniques ont une performance similaire aux écrans tactiles commerciaux. **(Conclusion)** Les résultats de cette étude sont importants pour établir des tailles cibles minimales pour les interfaces avioniques soumises à des vibrations, ainsi que pour comprendre l'impact des vibrations sur l’utilisabilité des écrans tactiles dans le cockpit, les différences entre les positions des écrans et les limites de l'utilisation d'un support manuel comme méthode d'atténuation des vibrations.

ABSTRACT

(Problem) Touchscreens in the cockpit may be placed far from the pilot, and be used under vibration or turbulence. Prior work found that touchscreens offered higher target selection throughput (a performance score combining both speed and accuracy) in static conditions, compared to other tested input devices, but their performance and error rate degraded more quickly under vibration. SAE ARP60494 has suggested using a hand-support to mitigate this effect. There is a need to quantify the impact of vibration on touchscreen target selection in the flight deck environment, as well as measure hand-support utility, using a standardized methodology. **(Objective)** We measured the impact of vibration on touchscreen target selection using a standardized methodology, allowing our results to be compared against other studies, and compared between hand-support methods (using the thumb, while holding onto the screen's edge, versus using the index finger freehand), avionic versus commercial touchscreens, and touchscreen positions. **(Method)** 24 participants completed an ISO 9241-411 multidirectional selection task (a standardized Fitts' target selection task). We built an adjustable vibration test platform, with the touchscreens placed in representative cockpit positions. The participants were exposed to vibration levels representative of helicopter flight. We tested four touchscreens, two screen positions, two hand-support methods and two vibration levels. We measured target selection throughput, error rate and subjective preference. **(Results and Discussion)** We found average throughput values of 6.5 bits/sec in static conditions, versus 5.7 bits/sec with vibration, which is higher than alternative input device throughput reported in prior work. Average error rates were high: 10.3% in static conditions, versus 16.6% with vibration. Similar to prior work, we found an exponential increase in error rate with decreasing target size. We failed to find evidence of a benefit from using the hand-support, compared to the freehand baseline, for the task and vibration conditions we investigated: the hand-support resulted in lower throughput, and higher or equivalent average error rate, in both static and vibration conditions. Under vibration, the pedestal outperformed the main instrument panel position, with higher throughput and lower error rate. In static conditions, the two positions performed similarly. The avionic touchscreens performed similarly to the consumer touchscreens. **(Conclusion)** The findings of this study are important for establishing minimum target sizes for avionic interfaces under vibration, as well as understanding the impact of vibration on touchscreen usability in the cockpit, the differences between screen positions, and the limits of using a hand-support as a vibration mitigation method.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

This list presents the symbols and abbreviations used in the thesis or dissertation in alphabetical order, along with their meanings.

A	Amplitude
Ae	Effective amplitude
ANOVA	Analysis of variance
Approx.	Approximately
cm	Centimetres
FMS	Flight Management System
G	Used to denote the base of the numbering system, when calculating octave bands
g	Acceleration due to gravity
ID	Index of difficulty
IDe	Effective index of difficulty
in	inches
m	Meters
MCDU	Multi-Function Control and Display Unit
MFD	Multi-Function Display
p	p-value (probability that the result is due to chance)
PFD	Primary Flight Display
RMS, rms	Root mean squared
s	Seconds
sec	Seconds
TP	Throughput
W	Target diameter
We	Effective target diameter
wRMS	Weighted root mean squared
σ	Standard deviation
X^2	Chi squared, a statistical test

Letters used to denote each experimental condition

Screens	Hand Support Method	Vibration Environment	Screen Position
M (**M**edium screen. MFD)	**F** (index **F**inger. Unsupported hand)	**U** (**U**nmoving. Static)	**[blank]** (MIP)
S (**S**mall screen. MCDU)	**T** (**T**humb. Hand supported on bezel edge)	**V** (**V**ibration)	**P** (Pedestal)
L (**L**arge screen. Planar touch monitor)			
i (**i**Pad)			

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

Aircraft are becoming more automated, and the job of the pilot is transitioning towards providing instructions to automated systems, monitoring them to make sure everything is working properly, and troubleshooting if something goes wrong. As controlling an aircraft becomes more like interfacing with any other computer (e.g., data entry, menu selection and map manipulation tasks), it makes sense that the interface becomes more like that of any other computerized device. This means adding interfaces like keyboards and selection devices, as well as larger screen displays. Information can be displayed more clearly on these large screens, which helps for monitoring the state of the aircraft and the increasingly complex systems controlling it. The pilot is thus better kept in the loop, allowing them to discover and address potential issues before they become emergencies.

Touchscreens are already being installed in aircraft cockpits. One prominent example is the Gulfstream G500 and G600, which are private jets whose flight decks were designed with many of the traditional physical controls replaced with touchscreens [1]. Touchscreen systems such as Thales' FlytX [2] and Garmin's G500H TXi [3] are already being sold for use in helicopters, including for retrofit into older helicopters that were not designed for them. During retrofits, touchscreens are often placed in the same position as existing screens, which could make them more difficult for the pilot to use, especially for extended periods, since the physical ergonomics of the flight deck was not adapted for this use case.

Space in the cockpit is at a premium. Because of space limitations, the physical keyboards currently in use in the flight deck are often much smaller that office keyboards, with smaller keys. For example, small-key keyboards have traditionally been built into MCDU's (Multi-Function Control and Display Units) [4]. With physical keyboards, every key that may possibly be needed for any task is always displayed and takes up space. One advantage of touchscreen soft keyboards is that they are capable of having multiple modes. If numeric entry is required, then only numbers need be displayed. This reduces the potential for entry of invalid characters into a given field, and frees up space for larger number keys. In addition, even if an input field accepts both letters and numbers, the keyboard and number pad can still be placed in separate modes, with the user capable of toggling between them. Again, this frees up space in each mode for larger keys. As a result, with the careful design of soft keyboard modes, it is possible that data entry on MCDU touchscreen

soft keyboards may ultimately be faster than with physical MCDU keyboards, although we found no prior work explicitly comparing the two; Wang and Zhao investigated different soft key sizes for use on an MCDU, but did not directly compare it against a physical version [5]. In addition, soft keyboards can appear only when needed, leaving extra screen space to display task-relevant information. For example, the same screen space used for data entry during one part of the flight can subsequently be used to display an interactive map, during another part of the flight.

An advantage of touchscreens is that they are very compact, since the display and control device are combined. Selection speeds on touchscreens are also faster than on alternative selection devices. MacKenzie found a throughput value (a combined measure of speed and accuracy) of 6.95 bits/sec on a smartphone touchscreen, which was 1.4 to 1.9 times higher than previously reported throughput values for mice (3.7-4.9 bits/sec) [6]. Thomas, who compared common flight deck interfaces against each other in static conditions, found that a touchscreen had much higher throughput values than a trackball, trackpad, thumbstick or fingerstick [7]. Lin et al. found that a touchscreen had 1.3 times higher throughput than a mouse and 3 times higher throughput than a trackball in static conditions [8].

A disadvantage of touchscreens is that they have higher error rates at smaller target sizes, as shown by Lin et al. [8]. Lin et al. compared a touchscreen against a mouse and trackball, under both static and ship vibration conditions. For the mouse and trackball, the error rate remained relatively constant, at less than 2%, across the target sizes tested (1-2.5 cm), when averaged across the vibration conditions. However, for the touchscreen, the error rate increased exponentially with decreasing target size, going from 2% error rate at a 2.5 cm target size to 16% error rate at a 1 cm target size, when averaged across the vibration conditions.

Another disadvantage of touchscreens is that they are more impacted by vibration than other input devices, as shown by Lin et al. [8]. Lin et al. showed that touchscreen throughput and error rate degraded at a faster rate than for a mouse and trackball, under ship vibration conditions compared to static conditions.

One problem with touchscreens in the cockpit environment is that they are often placed at a far distance from the pilot. The screen located directly in front of the pilot, the primary flight display (PFD), is positioned at a distance ranging from around 27 to 32 inches (69-81 cm) from the pilot's eye, depending on the aircraft type. The distances can be even larger for multi-function displays

(MFDs), since they are at the same longitudinal distance, but are off to the side. This can require the pilots to lean forward and fully extend their arms to interact with the screens. It may also result in fatigue after extended use, which may further increase error rate. This is in contrast with a trackball or similar input device, which can be located right beside the pilot, within a hand-rest, so that the hand is always ergonomically supported while using the device.

1.1 Aircraft Vibration and Efforts to Mitigate its Impact

The effect of reach distance on both touchscreen throughput and error rate may be exacerbated by vibration. All aircraft can experience turbulence, but helicopter pilots in particular can be exposed to high levels of vibration during normal flight. Chen et al. [9] found that, in a CH-147F helicopter during the 150 knot flight and "normal approach to hover" conditions, the level of vibration on the pilot's seat reached "extremely uncomfortable", as defined by the ISO 2631-1 standard [10]. That is the highest level of discomfort defined in that standard. Wickramasinghe also found that vibration levels measured on the pilot and co-pilot's seat of a Bell-412 helicopter vary from "a little uncomfortable" to "extremely uncomfortable" [11].

As a result, there is an urgent need to quantify the effects of vibration on touchscreen usability in the helicopter cockpit environment, and find solutions to mitigate this effect. One mitigation solution that is suggested by the SAE ARP60494 standard, which addresses touchscreens in aircraft cockpits, is to provide a hand support. This standard refers to "hand grips" and a hand support on the "bezel area" [12].

Cockburn et al. [13] mentions that, under vibration, participants naturally stabilized their hand on the screen edge to select targets, and claims that this is an effective strategy. However, their study did not control for this hand-stabilization variable, and did not compare the results against an unsupported condition.

Cockburn et al. [14] tested three different methods of registering selections, under vibration and static conditions: on double-tap, on dwell, and on reaching a force threshold. The force threshold was deliberately chosen to be relatively high, at 6.9 N. The vibration condition had an RMS of 2.15 m/s^2 and a discomfort level of "very high", as defined by the ISO whole-body vibration standard [15]. They compared using the three selection methods (on double-tap, on dwell and on reaching a force threshold) with one finger, freehand, versus using these same three selection

methods while supporting multiple fingers directly on the screen, under both vibration and static conditions. They found that double-tapping to select outperformed the dwell and force-threshold condition. They found that when comparing the support-finger method against the freehand method, using support-fingers resulted in higher error rate and movement time in static conditions, but lower movement time and error rate in vibration conditions. In the static-freehand condition, they reported error rates between 2-24%, with double-tap having 2% error and force-threshold having 24% error.. In the vibration-freehand condition, they reported error rates between 49-56%, depending on selection registration method. In addition, Cockburn et al. found that the support-finger method had a higher frequency of accidentally selecting the wrong target, when compared to freehand. Under vibration, the support-finger method resulted in a wrong-target selection in 1-3% of selections, depending on selection registration method, whereas freehand resulted in a wrong-target selection in 0-1% of selections. As a result, aircraft manufacturers may be hesitant to implement this type of hand-support strategy, since selecting the wrong target is a worse form of error than missing the target and hitting an inactive area of the interface. In addition, the selection-registration methods tested by Cockburn et al. may add additional selection time and frustration, versus a standard, on-release method. Cockburn et al. did not compare an on-release selection registration method as a baseline.

Coutts et al. [16] measured selection task performance on a vibrating test platform, with touchscreens placed in cockpit positions. They permitted the participants to "anchor" their hands on the edge of the screen, leaving the decision of whether or not to use this hand-support method up to the participants. The authors also claimed that this hand-support strategy was an effective method to mitigate against vibration, but did not control for this variable and did not compare it against an unsupported condition.

Dodd et al. [17] measured task time, vibration and fatigue on a typing and menu selection task under turbulence. They also permitted the participants to either anchor or not anchor their hands on the bezel edge, without explicitly controlling for this variable. They reported that several participants claimed that supporting their hand on the bezel edge helped with the task, under turbulence.

We found one prior study, by Hourlier and Servantie [15], that explicitly tested the utility of using the bezel edge as a hand-support against a freehand baseline condition. Participants were exposed

to a vibration profile with a high mean acceleration of 1.53 m/s^2, which is considered either "uncomfortable" or "very uncomfortable" by the ISO whole-body vibration standard [15]. In this study, three tasks were performed. One of those tasks was to press on a circle, drag it to a target, and release it as precisely as possible, centered on that target. The only comparison shown between the hand-support versus freehand conditions was for the "press" component of this task. The authors found that the error rate was much higher in the freehand condition, compared to when using a hand-support: more than twice the error rate at small target sizes of 7 mm, with the difference between the two conditions becoming smaller with increasing target size. The authors recommended using a hand-support under vibration conditions. However, they did not compare the hand-support for the other tasks and provided no statistical analysis of their data to assess if the difference observed was significant.

Both Hourlier and Servantie [15], and Cockburn et al. [16] used intense turbulence profiles of 1.53 and 2.15 m/s^2, respectively. This could explain the advantage that they found for their respective hand-support methods. Discussions with aerospace industry experts revealed that backup input devices, such as knobs and/or bezel keys are generally provided in case the touchscreen becomes unusable either due to hardware malfunction or environmental factors like excessive vibration and turbulence, such as that investigated by Hourlier and Servantie.

1.2 ISO 9241-411 Multidirectional Selection Test

The present study focuses on target selection tasks specifically, since these are very common tasks that are usually included in even the simplest interfaces. Soukoreff and MacKenzie have suggested that "throughput" is the best measurement to use, when objectively quantifying the performance of input devices [18].

Fitts' law was first described by Paul Fitts, who showed that there is a trade-off between speed and accuracy in target selection tasks [19]. Fitts used information-theory to explain his experimental findings. While this theoretical basis for the speed-accuracy trade-off is disputed [20-22], Fitts' experimental finding of a speed-accuracy trade-off has been replicated many times in different contexts, as discussed by Soukoreff and MacKenzie [18]. In essence, these experimental findings show that the human motor system can naturally switch between prioritizing selection speed or

accuracy. But, above a certain limit, increasing selection speed comes at the expense of decreased accuracy, and increasing selection accuracy comes at the expense of decreased speed.

A Fitts' task involves having participants select between targets as quickly and as accurately as possible [23]. Paul Fitts' original study involved a one dimensional tapping task, where participants used a stylus to move horizontally back and forth, tapping between two metal plates of specified widths [19]. A 2D tapping task is now preferred, in order to control for the impact of movement direction on throughput [18]. The ISO 9241-411 standard [24] provides a specification for how a multidirectional selection task should be conducted. Within a given task trial, targets of a constant width (W) are arranged around a circle at a distance (A); see Figure 1.1. Participants select targets in a predictable pattern, always moving across the circle to select the opposing target. Multiple trials are conducted, with each trial having a different combination of target size and movement distance.

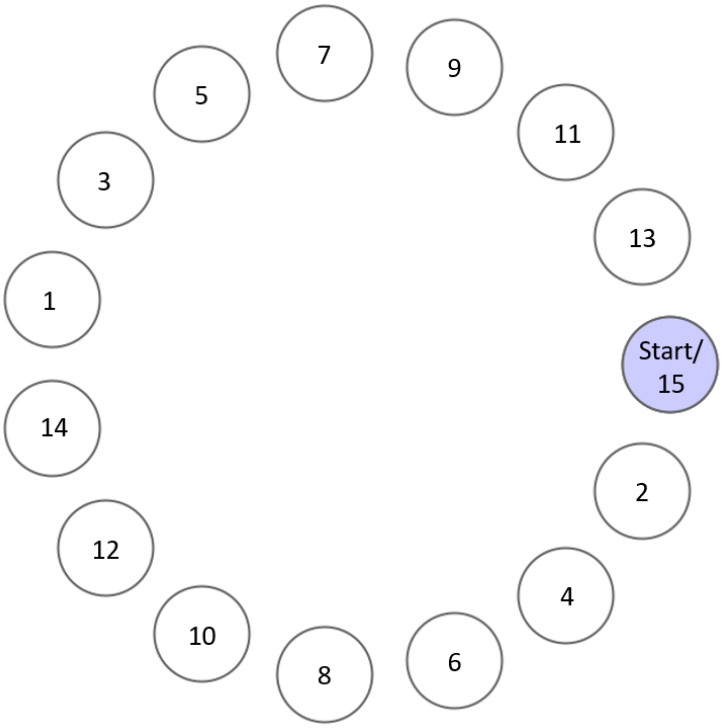


Figure 1.1 – The layout of targets in the multidirectional selection task, with the targets numbered per their order of selection.

A central variable for the calculation of Fitts' law results in the Index of Difficulty, represented by ID. As it's name suggests, it is considered to represent the difficulty of the task, in terms of human

motor performance limits for target selection, and it is also a representation of the accuracy required by the task. Over time, several different methods have been proposed for calculating ID, based on the results of a Fitts' task [25, 26]. The most widely accepted, including by the ISO 9241-411 standard [24], is the "Shannon formulation" [18, 24]:

$$ID = Log_2(\frac{A}{W} + 1)$$

Where ID is the "index of difficulty", A is the distance between targets, and W is the target diameter. ID is calculated per trial, where each trial has a different combination of A and W.

The "speed-accuracy trade-off" [27] of Fitts' law is then represented with the following formula, which relates target selection time to ID [18]:

$$MT = a * ID + b$$

Where MT represents the average time it takes to select a target (movement time), and "a" and "b" are device-specific constants that are calculated based on a linear fit of the ID vs MT data points.

Throughput (TP) is calculated based on the following formula [27]:

$$TP = \frac{ID}{MT}$$

Units for throughput are bits per second (bps), in an analogy to information theory [19, 28], with larger throughput values meaning a higher ratio of accuracy to target selection time (with ID being a representation of accuracy, and MT being the speed of target selection).

However, while ID represents the difficulty of the task presented to the participants, the participants may make the task easier or harder for themselves by being either less or more precise than the task ID implies [29]. To address this phenomenon, it is now commonly accepted to correct the values of A and W based on the participants' observed accuracy by using the adjusted index of difficulty (IDe) [18, 24]:

$$IDe = Log_2(\frac{Ae}{We} + 1)$$

Where Ae is the effective movement distance, and We is the effective target width. This calculation and corrections are done per task (A x W combination), per participant.

Ae is calculated as the average distance between sequential pairs of taps, per task, per participant. The Ae correction uses the participant's actual tap points, as measured when they are projected onto a 1D line that connects the two target centers that those taps were targeting [23, 30].

We is calculated based on the standard deviation of tap point offset distances from the target center, where those distances are projected onto a 1D line connecting the previous and current targets [18, 24]:

$$We = 4.133 * \sigma_{dx}$$

Where σ_{dx} represents the standard deviation of dx. dx is the distance between a given tap and the target's center, projected onto a 1D line connecting that target's center to the previous target's center.

Once IDe is calculated, based on these corrections for accuracy, it replaces the ID value that was used in the movement time and throughput formulas shown previously [27]:

$$MT = a * IDe + b$$

$$TP = \frac{IDe}{MT}$$

Soukoreff and MacKenzie suggest that the MT(IDe) function may be used for prediction purposes, such as when attempting to predict one-handed typing speeds on a soft keyboard [18, 31]. They also suggest that throughput can be used when comparing selection task performance in different conditions using the same device, or between devices as the method to compute throughput is device-independent [18]. In this study, we are mainly comparing conditions against each other, and so have used throughput for this purpose. The throughput value captures both the speed and accuracy of using the input device under a given set of conditions.

1.2.1 Factors That Can Impact Throughput

There are many factors that impact throughput. Throughput varies between input devices: using a mouse (4.5 bps) results in a different throughput than using a trackball (3.0 bps), which does not have the same throughput as a touchscreen (6.8 bps) [8]. In addition, differences between two devices of the same type can affect their throughput values, such as their responsiveness [32]. Use

context can also impact throughput, such as the angle at which the touchscreen is placed [33], or vibration [8].

Throughput may also vary between people. Not everyone has the same hand-eye coordination, and older users may have lower throughput values than younger users, on average [34, 35]. However, Soukoreff and MacKenzie have shown that, for the same device and use context, throughput values are relatively constant for average populations of similar groups of people [18].

This list provides some of the factors that can impact throughput:

- Device used [7, 18]
- Environmental context of use (including screen orientation and vibration) [8]
- Demographics of the participant population (for example: people with higher levels of hand-eye coordination versus people with motor impairments; younger users versus older users) [34-36]
- Body part used (such as the finger used, or even whether the hand is used versus the foot or eye) [37, 38]

Movement angle also impacts throughput [33, 39, 40]. But it is controlled for by the ISO pointing task, by arranging targets around a circle to provide multiple angles, and then calculating the average throughput across all target angles.

One of the goals of using throughput is to have a standardized value (calculated in a standardized way) that can be compared across input devices to untangle how these and other factors impact user performance [18].

1.2.2 Effects of targets orientation

A number of studies have presented the impact of movement direction on target selection performance, while using touchscreens. Trudeau et al. measured index of performance values per selection angle, for single-handed thumb selection on a mobile phone (index of performance is also a combined measure of speed and accuracy, like throughput) [41]. He found that, for right-handed users, their performance was better when their selection ended towards the north-west of the screen

(farther from the hand). Lehmann and Kipp measured throughput values for different sections of a smartphone touchscreen, with participants using either their thumb or index finger [42]. They found that, using the thumb, for right-handed users, moving towards the north-east quadrant had the best performance. Using the index finger, moving towards the north-west had the best performance. Kim and Jo also found that the thumb had better performance when selecting targets towards the north-east, whereas the index finger had better performance towards the north-west, for a smartphone touchscreen [43]. Lee et al. also found that thumb selections ending towards the north-east of the screen had the best throughput, for a smartphone touchscreen [39]. Bachynskyi et al., who tested touchscreens of different sizes (representative of a smartphone, tablet, touch laptop, "tabletop display", and large "public display") found that performance was better when moving along a horizontal line between targets, compared to moving along a vertical line between targets [33].

In summary, it seems as though the consensus between all these articles is that users have better performance when starting at the bottom-half the screen and selected targets towards the top-half of the screen. They have worse performance when starting at the top-half of the screen and selecting targets towards the bottom-half of the screen. These results seem to apply both to thumb and index finger use.

The studies that we found that investigated the impact of movement direction on throughput when using the thumb for target selection on a touchscreen [39, 41-43] were all done on small, smartphone screens. Bachynskyi et al. [33] used cardboard sheets with printed targets to represent larger screens, tracking the participants' finger positions to detect selections, but did not explicitly test index-finger freehand versus thumb for target selection on these larger cardboard sheets.

Lin et al. investigated the impact of movement direction on touch target selection under vibration, for mouse, trackball and touchscreen devices (using a 17" touchscreen device) [8]. They provided throughput values per angle, as a combined average across both device and vibration/static conditions. These combined throughput averages ranged from 4.3 bit/sec at 135°, to 4.6 bits/sec at 0°, measured from the horizontal axis. However, these were combined angles that were mirrored across the origin. So the 0° angle included both right and left horizontal motions. While the 135° angle included both top-left-to-bottom-right as well as bottom-right-to-top-left motions. The

results of the articles mentioned previously imply that direction of movement has a large impact, and that mirrored angles should not be combined.

These notions are of importance to this work as we will investigate the effects of target orientation on selection performance to identify areas of the screens that are easier to access, and for comparing the hand-support against the freehand condition.

1.2.3 Effects of Vibration on Multi-Directional Selection Task Throughput and Error Rate

We found only one prior study, Avsar [44], that provided throughput values for touchscreen use in vibrating aircraft. Avsar measured touchscreen performance during a Fitts' law task in an AW139 helicopter during flight [44, 45]. The participants sat in the cabin of the helicopter, with a touchscreen tablet either affixed to the cabin window, or held in the participants' hands. During flight, with the screen affixed to the cabin window, Avsar reported throughput values of between 4.4 and 4.5 bits per second, depending on flight condition (with hover, transition and cruise flight conditions tested). During flight, with the tablet held in the participants hands, they reported throughput values of between 4.6 and 4.8 bits/sec, depending on the flight conditions. In comparison, when the helicopter was static, on the ground, Avsar reported throughput values of 5.2 bits/sec with the tablet affixed to the cabin window, and 5.6 bits/sec with the tablet hand-held. This study also tested four different target sizes: 0.5 cm, 1 cm, 1.5 cm and 2 cm. Avsar [44] and Avsar et al. [45] showed that the error rate decreased exponentially with increasing target size, across all experimental conditions. They reported error rates of 14.6% and 21.6% during flight, for the hand-held and fixed tablet positions respectively. They reported 6.8% and 7% error rates with the helicopter static, on the ground, for the hand-held and fixed tablet positions, respectively.

However, Avsar did not use the multi-directional selection task proposed by the ISO 9241-411 standard [24]. Instead, both the target size and distance were randomized for each sequential target. Hence, even within a given trial, the layout of the targets, and their sizes, did not follow a predictable pattern. Due to this deviation in task design and resultant throughput calculation method, the study's methodology did not strictly follow that set out in the ISO standard, or proposed by Soukoreff and MacKenzie [18, 24]. Avsar notes that this may make it difficult to

compare their study's throughput values against those found in other studies. In addition, Avsar [44] and Avsar et al. [45] did not control for one-handed or two-handed use, instead allowing participants to choose whether they used one or two hands for the target selection task. Thus, their results may not be fully applicable for one-handed touchscreen use. During discussions with pilots, we were told that helicopter pilots often fly with one hand always on the flight stick, if there is no autopilot or if the autopilot is disengaged. Thus, they would only have one hand free to use the touchscreen. As a result, investigating one-handed use of touchscreens is important in the helicopter context, and was not done by this previous work.

Coutts et al. [16] and Cockburn et al. [14] had participants perform a target selection task similar to the one mentioned in the ISO 9241-411 standard [24], with targets arranged around a circle. However, they reported the selection time and error rate separately, without providing a throughput value. Cockburn et al. [14] was discussed previously, and did not use a standard on-release selection registration method.

In the study by Coutts et al. [16] the next target would only appear if the previous target was successfully selected. If an error was made, the participant needed to go back and try to select the same target again, before being allowed to progress (this is contrary to the methodology we have used in our study, where the next target appears after any tap has been registered, whether successful or not, which is based on the methodology used in the GoFitts software [46, 47]). As a result, Coutts et al. reported the average number of error made, rather than the error rate. Coutts et al. used four levels of vibration: static condition, 0.26 m/s^2 wRMS, 0.37 m/s^2 wRMS, and 0.52 m/s^2 wRMS. wRMS is the weighted RMS value calculated per the ISO 2631-1 standard [10]. Coutts et al. tested large touch displays in three positions: PFD position (centered on main instrument panel), pedestal position, and overhead panel position. They tested five target sizes, of: 1, 1.5, 2, 2.5, and 3 cm. They tested three different touchscreen technologies: capacitive without force sensing, capacitive with force sensing, and infrared. They found that the number of errors decreased exponentially with increasing target size, across all Touchscreen-Technology x Screen-Position x Vibration-Level conditions. They found that the pedestal position resulted in a higher number of errors than the main instrument panel position. However, the authors noted that the screen did not always recognize taps when placed in the pedestal position, which may have influenced the results. In addition, the screen they used was larger than a normal screen that might be found on the pedestal of a flight deck, which may have influenced the ergonomics of reach.

While Avsar [44, 45] conducted his experiments in a helicopter during routine flight operations, the other studies we found, which measured target selection performance on touchscreens in aircraft vibration conditions, used vibration profiles representative of turbulence or light chop [13-17]. It is possible that, especially at high levels of turbulence, the pilots may need to concentrate on other aspects of flying the aircraft, and may avoid or limit their use of the touchscreens, or use backup input systems. In the present study, we have focused on vibration representative of level flight in a helicopter. Since helicopter pilots may be exposed to this level of vibration for large portions of a normal flight, it is a realistic vibration environment in which pilots could be expected to use the touchscreens often.

Outside of the aviation field, we found one study, by Lin et al. [8], that provided throughput values for touchscreens under vibration, with the vibration profile used being based on the motion of a ship at sea. The authors compared throughput values for a touchscreen, mouse and trackball under three levels of ship vibration: none, "not uncomfortable" (average vibration of 0.22 ms^{-2}) and "a little uncomfortable" (average vibration of 0.34 ms^{-2}), as defined by the ISO 2631-1 standard. Participants sat on a chair at a comfortable distance from an office desk to interact with the devices, within a Steward motion platform. The authors found that the touchscreen had a much higher throughput in the static (no-vibration) condition, when compared to the mouse and trackball (8.2 bps for the touchscreen, 6.3 bps for the mouse, and 2.6 bps for the trackball). However, with "not uncomfortable" and "a little uncomfortable" levels of vibration, the touchscreen throughput values dropped significantly, to be equivalent to those of the mouse (5 bits/sec with "not uncomfortable" levels of vibration, and 4 bits/second with "a little uncomfortable" levels of vibration, for both the mouse and touchscreen). The average throughput using the touchscreen remained consistently higher than the trackball, for all vibration conditions. Of note, the error rate on the touchscreen was consistently higher than on the mouse and trackball, in all vibration conditions. The error rate on the touchscreen raged from under 2% in the static condition, up to 14% in the "a little uncomfortable" vibration condition. For the mouse and trackball, the error rate remained below 2.1% across all vibration conditions. Lin et al. found that the error rate on touchscreens decreased along a negative exponential curve with increasing target size, when averaged across vibration conditions, whereas the error rate with the mouse and trackball remained relatively constant across target sizes. On the touchscreen, the error rate was 16% at the smallest target size of 1 cm, before decreasing to 2% error for the largest target size of 2.5 cm. With the mouse and trackball, the error

rate remained below 2% for all target sizes. The authors did not present the combined effects of target size and vibration. It is difficult to tell whether the same drop in throughput and the increase in error rate, due to vibration, occurred for the larger targets as well as the smaller targets. In addition, for their target selection task, Lin et al. did not use the exact same layout as that recommended in the ISO 9241-411 standard [24].

Consistent across the results from Avsar [44, 45], Coutts et al. [16], Lin et al. [8] and Hourlier and Sevantie [15] is that error rate on touchscreens decreased along a negative exponential slope with increasing target size, in the range from 0.7 cm to 3 cm. Avsar et al. [45] reported that there was no significant difference between error rate at a target size of 1.5 cm versus 2 cm, at least for the conditions they tested. This implies that there may be a point of diminishing returns, at which increasing the target size further provides incrementally less benefit, in terms of error rate reduction.

1.3 Synthesis

This literature review presented the state of knowledge on selection performance using touchscreens under vibration. Selection with touchscreens provides higher throughput, when compared to other input devices, but this advantage lessens under vibration. Error rate on touchscreens is more sensitive to target size: with decreasing target size, the error rate on touchscreens increases more rapidly than when using other selection devices. The error rate on touchscreens increases exponentially with decreasing target size. When performing a target selection task, some directions of motion have higher throughput than others, implying advantages for certain areas of the screen (motions towards the top of the screen have better performance than motions towards the bottom of the screen).

Several studies that investigated touchscreens in the context of aircraft cockpits recommended stabilizing the hand to improve selection performance under vibration, but did not explicitly test using the bezel edge as a hand-support method for on-release target selection, against a freehand baseline.

We found previous work that investigated throughput on touchscreens under vibration conditions representative of a helicopter. However, the task did not follow all the recommendations of the ISO 9241-411 standard [24] for multidirectional selection tasks, and did not control for one or two

handed use. As a result, it is difficult to compare the throughput results against other studies, for example those investigating other input devices for use in a cockpit environment. In addition, the touchscreens were not placed in a manner representative of a flight deck layout, and the vibration conditions were not fully controlled.

Conversely, we found previous work investigating the impact of fixed-wing turbulence on touchscreen usability in a representative flight-deck layout, and comparing common flight-deck screen placements, using a controlled vibration environment. However, these studies did not present throughput values, and did not use vibration levels representative of a helicopter in normal flight conditions.

We also observed that previous work used commercial touchscreen products, or did not specify which touchscreens were used, and the evaluation of actual avionic touchscreens installed in the flight deck received less attention.

1.4 Objectives

The present study attempts to fill in research gaps in the following ways:

1) Quantify the impact of vibration on one-handed target selection task performance by providing throughput values measured in controlled, representative helicopter cockpit conditions. Use the "multi-direction pointing task" recommended in the ISO 9241-411 standard and the methods recommended by Soukoreff and MacKenzie to measure and calculate throughput [18, 24].

2) Compare the effectiveness of using the edge of the screen as a hand support, versus using the touchscreen without any support, for a 2D Fitts' selection task. Compare these results against those of Hourlier and Servantie [15].

3) Compare selection task performance on real avionic touchscreens of representative sizes, versus commercial touchscreens.

4) Compare selection task performance for touchscreens located at two positions: Main Instrument Panel (MIP) position (directly in front of the pilot and at a vertical incline) versus the pedestal position (beside the pilot and at a horizontal incline). Compare these

results against Coutts et al., who measured selection task speed and accuracy in different screen positions, but did not provide throughput values [16].

5) Confirm the impact of target size on error rate under vibration and compare the results against those of Avsar [44], Avsar et al. [45], and Coutts et al. [16].

CHAPTER 2 METHODOLOGY

This section presents the method used for data collection and analysis. First, the specifications of the four touchscreens used are presented along with their locations, relative to the eye reference point. The hand support method used for each screen is also presented. Second, the vibration profile to which the participants were exposed is analysed to show that it replicates measurements taken on the pilot's seat in a helicopter. Third, the variables measured and 18 test conditions are presented. Fourth, the software used for Fitts' test data recording and analysis are presented. Fifth, the participants' anthropometric measurements are reviewed, before presenting the statistical method used for the analysis of variance (ANOVA) and pairwise comparisons. The section ends with the experimental procedure.

2.1 Touchscreen Specifications

Four different touchscreens were used (see Figure 2.1). Two were prototype avionic touchscreens provided by CMC Electronics Inc. The other two were consumer touchscreens.

- **Prototype version of the MDU-268** avionic multi-function display unit (MFD), sold by ScioTeq [48]. This touchscreen has the following specifications: 1024 pixels vertical x 768 pixels horizontal; 123.6 pixels per inch with a display area of 21 cm x 16 cm; projected capacitive touch technology. Including the screen's bezel, it has overall dimensions of 10 inches vertical by 8 inches horizontal (25.4 cm x 20.3 cm). This screen is called the "**MFD**" or "medium MFD screen" for most of the rest of this document (since it was of "medium" size, compared to the other screens tested here, and since it is denoted as the letter "M" in the analysis section).
- **Prototype touchscreen MCDU**, made by CMC Electronics. It is similar to the TSCU-5045, sold by ScioTeq [49]. This touchscreen has the following specifications: 1024 pixels vertical x 768 pixels horizontal; 201.6 pixels per inch with a display area of 13 cm x 10 cm; digital resistive touch technology (a resistive touch technology that accepts multi-fingered touch). Including the screen's bezel, it has overall dimensions of 6.7 inches vertical by 5.8 inches horizontal (17.0 cm x 14.6 cm). This screen is called the "**MCDU**" or "small MCDU

screen" for most of the rest of this document (since it is of "small" size, compared to the other screens tested here, and since it is denoted as the letter "S" in the analysis section).

- **Planar Helium PCT2435**, a consumer touchscreen computer monitor [50]. It has the following specifications: 1080 pixels vertical by 1920 pixels horizontal; 92.53 pixels per inch with a display area of 30 cm x 53 cm; projected capacitive touch technology that allows 10-point multi-touch. Including the bezel, it has overall dimensions of 14 inches vertical by 21.3 inches horizontal (35.5 cm x 54.1 cm). This makes it slightly larger than a standard Large Area Display (LAD) avionic touchscreen, which often have display areas of around 8 inches vertical by 20 inches horizontal (20.3 cm x 50.8 cm) [51, 52]. This screen is called the "**large touch monitor**" for most of the rest of this document (since it is of "large" size, compared to the rest of the screens tested here, and since it is denoted by the letter "L" in the analysis section).
- **Apple iPad 6th Generation**, a consumer tablet touchscreen [53]. It has the following specifications: 2048 pixels vertical x 1538 pixels horizontal; 264 pixels per inch with a display area of 20 cm x 15 cm. These dimensions are very close to the MFD screen, allowing comparison between a consumer touchscreen (an iPad) and an avionic touchscreen (the prototype MDU-268). Including the bezel area, the iPad 6th Generation has overall dimensions of 9.4 inches vertical by 6.6 inches horizontal (24 cm x 17 cm). This screen is called the "**iPad**" for most of the rest of this document (it is denoted by the letter "i" in the analysis section).

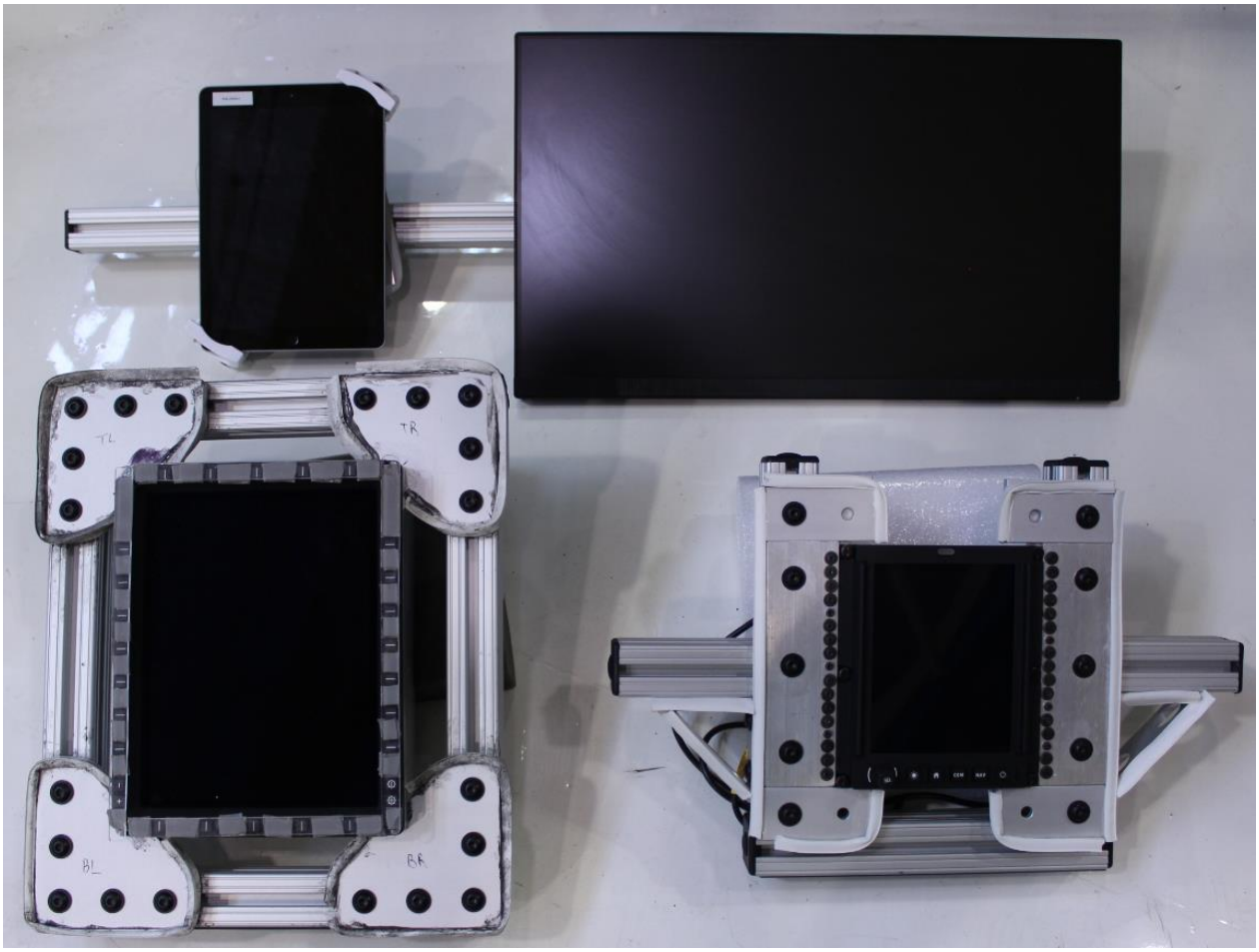


Figure 2.1 – The four screens used in this experiment, inside their frames. From top left to bottom right: iPad, Planar Helium PCT2435, Scioteq MDU-2068, CMC Electronics MCDU.

The iPad was running iPadOS 14.4.2. All other screens were connected to a laptop running Windows 10, and operated as a touch-enabled monitor to that laptop.

The two avionic touchscreens were not originally built to operate as touch monitors for Windows 10. These units were combined computer-touchscreen assemblies that are meant to run their own proprietary operating system. Employees of CMC Electronics programmed a plugin so that these screens could operate as touch monitors, connected to a Windows 10 computer. However, the plugin reduced the touch sampling rate of the screens, and the screens' responsiveness may not have been fully representative of their real use case.

A fast connection cable was installed to the MCDU, which improved the data transfer rate between the screen and the computer, and its responsiveness.

The MFD display was less responsive. There was a slight delay between touching a target and the screen registering that touch. In addition, the screen occasionally missed touches entirely. It is unknown whether this represents the real responsiveness of this screen, or whether it was a result of the connection between the screen and the Windows 10 laptop.

In either case, the results from the MFD are still useful, because they show the impact of a lower touch responsiveness on touchscreen usability in the context of a vibrating helicopter environment. Especially after years of use, an old avionic touchscreen may have a diminished level of responsiveness, similar to that of the MFD used in this study. The SAE ARP60494 standard mentions that touchscreen responsiveness may degrade over time, and that it may be difficult to tell that it has done so until the responsiveness reaches such a low level that it becomes obvious [12].

2.2 Touchscreen Positions

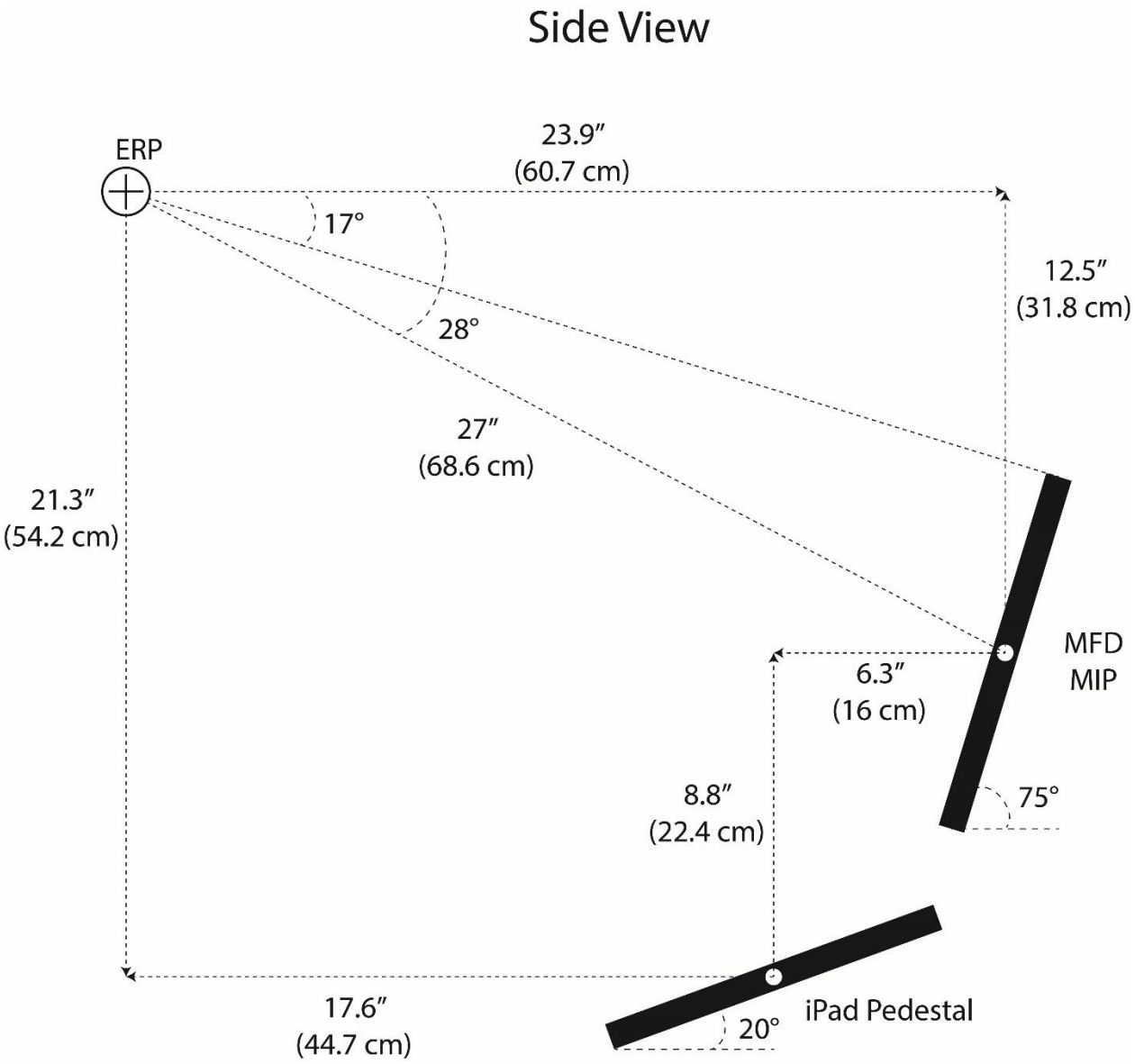


Figure 2.2 – Side view showing the screen positions.

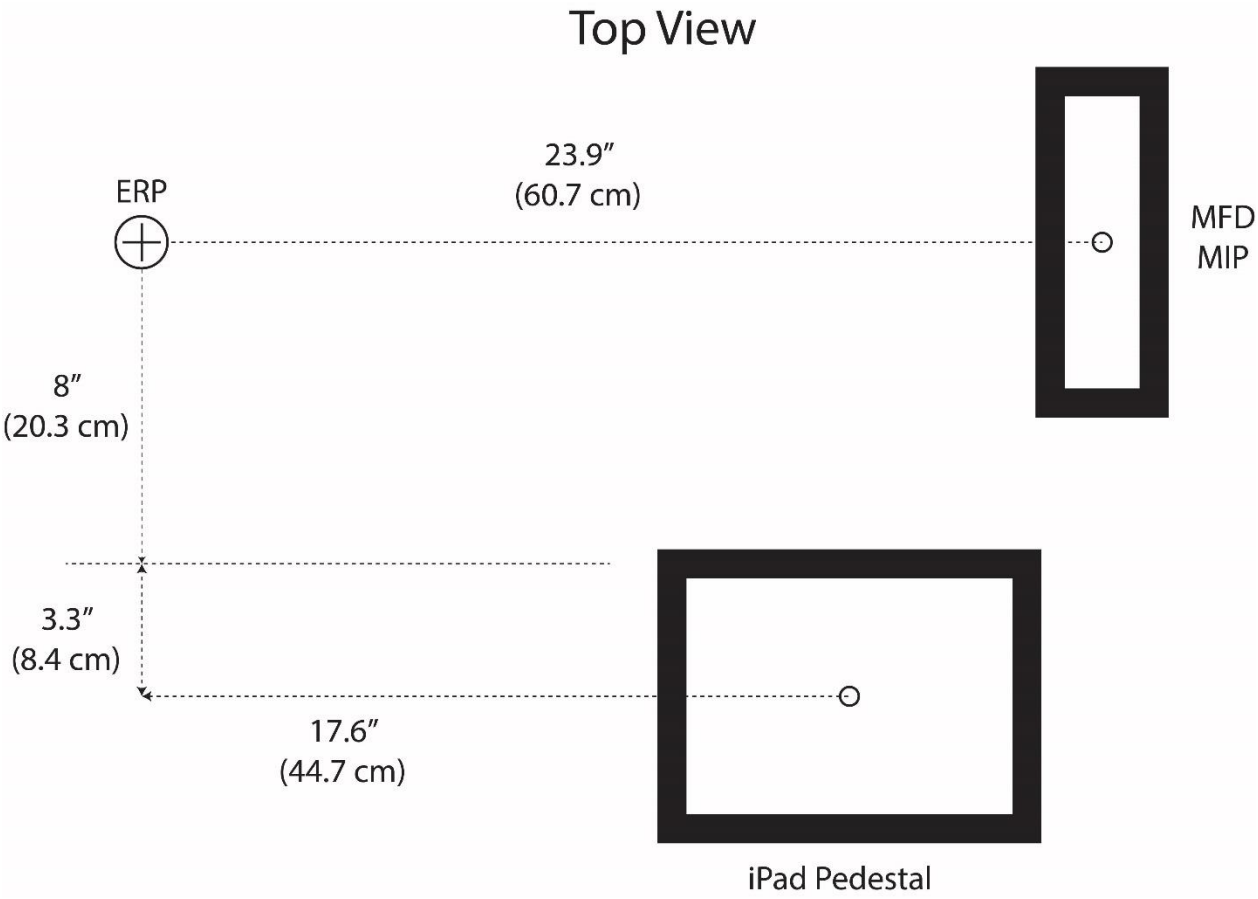


Figure 2.3 – Top view showing the screen positions.

When in the cockpit, the pilot adjusts the seat so that the pilot's eyes are located at the Eye Reference Point (ERP). This ensures that all pilots have the same view of the instruments and outside the window, no matter their stature.

In this study, the height of the screens was adjustable instead of the height of the chair, due to limitations with the test setup. The height of the screens was adjusted based on the sitting eye height of each participant, in order to maintain a consistent ERP to screen center distance across participants of different statures. See Appendix A for more details on this screen height adjustment.

The Federal Aviation Administration (FAA)'s Advisory Circular (AC) 29-2C states that, for helicopters, the "throttle envelope" can extend up to a maximum of 23 inches (58 cm) longitudinally from the ERP [54]. As a result, we ensured that the center of the screen was placed slightly farther than 23 inches (58 cm) longitudinally, to give room for an imaginary throttle.

The FAA AC 25.773-1 states that pilots must have clear view out the window from 17° up to 17° down, measured from the ERP [55]. As a result, in this study, we ensured that the top of the MFD screen was just below 17°, as measured from participants' ERP. We used the top of the MFD screen, since this is the screen that would normally be on the main instrument panel in the cockpit.

The center of the screen was placed at 27 inches (68.6 cm) from the ERP (see Figure 2.2). We selected this distance in consultation with human factors experts at CMC Electronics. This is close to the minimum distance that an MFD screen can be placed in the cockpit. While these screens are often placed farther from the ERP than 27 inches, it was decided that, in the future, as touchscreens become more prevalent in the cockpit, they will likely need to be moved closer to the pilot, for easier interaction. This length was chosen as a reasonable compromise, especially since some current screens are placed at this distance, and it gives a reasonable clearance for the throttle to move.

Screens were positioned in two locations : the Main Instrument Panel (MIP) or the pedestal (see Figure 2.3). The MIP position was representative of the position of a Primary Flight Display (PFD) [56, 57], whereas the pedestal position was representative of the position of an MCDU [5, 58]. In the MIP position, the screen was directly in front of the participants. In the pedestal position, the inner edge of the iPad was in line with the outer edge of the chair.

All four touchscreens were placed in the MIP position, with the centers of each screen placed in the same location, ± 1 inch (2.5 cm) since the screens had slightly different depths. The iPad was placed in both the MIP and pedestal positions. The iPad was the only screen placed in the pedestal position. We determined that testing all screens in both positions would be too time consuming for the participants, since the test already took 2 hours per participant. The method in which the iPad was secured allowed it to be easily adjusted in the lateral axis (Figure 2.4). As a result, the iPad could be quickly slid close to the participant's knee, when placed in the pedestal position, or it could be re-centered, when placed in the MIP position.

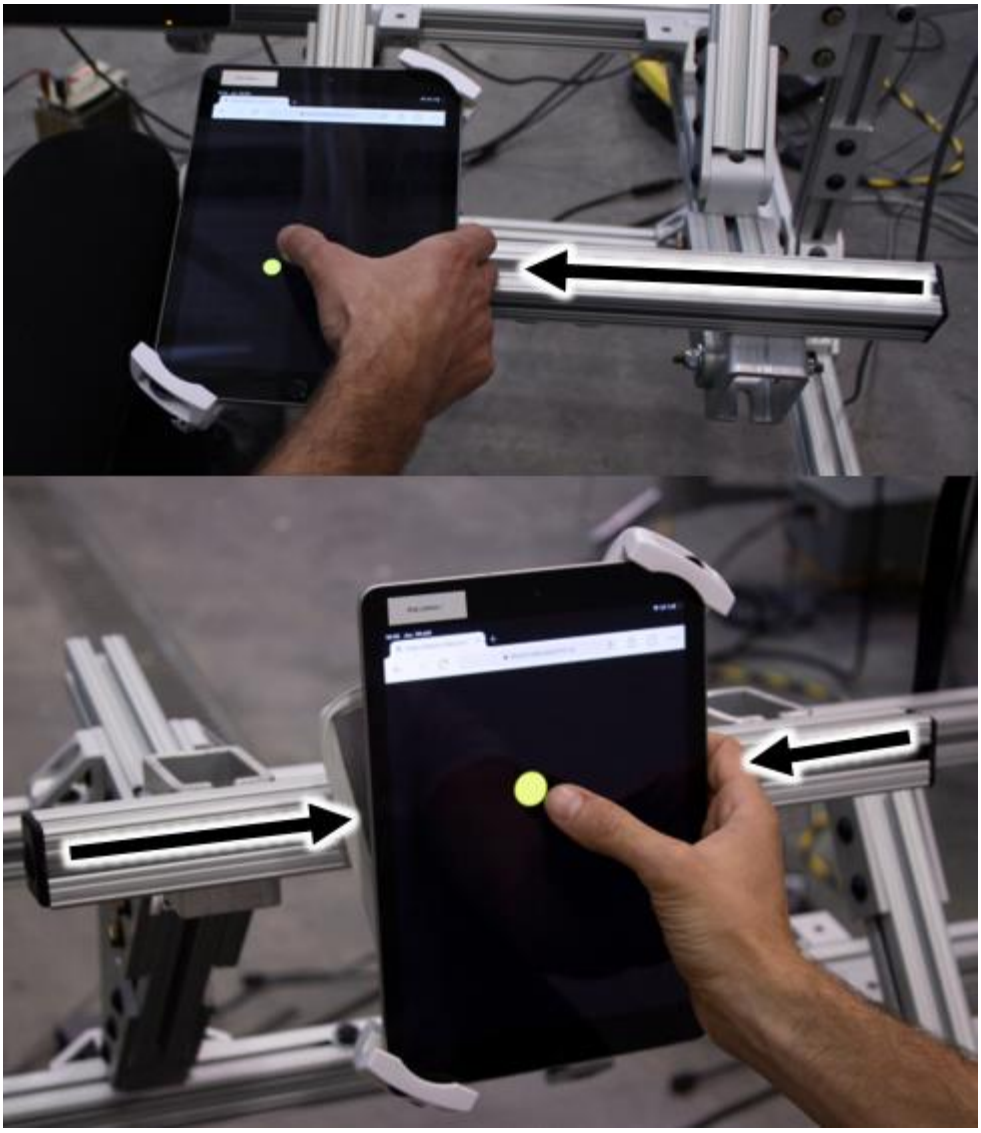


Figure 2.4 – Due to the way in which the iPad was secured to the structure, its lateral position could be easily adjusted.

The iPad's screen is similarly sized to the MFD. As a result, it was considered an acceptable proxy for the MFD screen placed in either the MIP or pedestal position.

In many cockpit layouts, the height of the pedestal screen's top edge is almost in line with the bottom edge of the MIP MFD's bottom edge. Thales' FlytX layout provides one such example [2]. We replicated this layout by placing the iPad's top edge very close, in height and longitudinal position, to the bottom edge of the MFD (Figure 2.5).

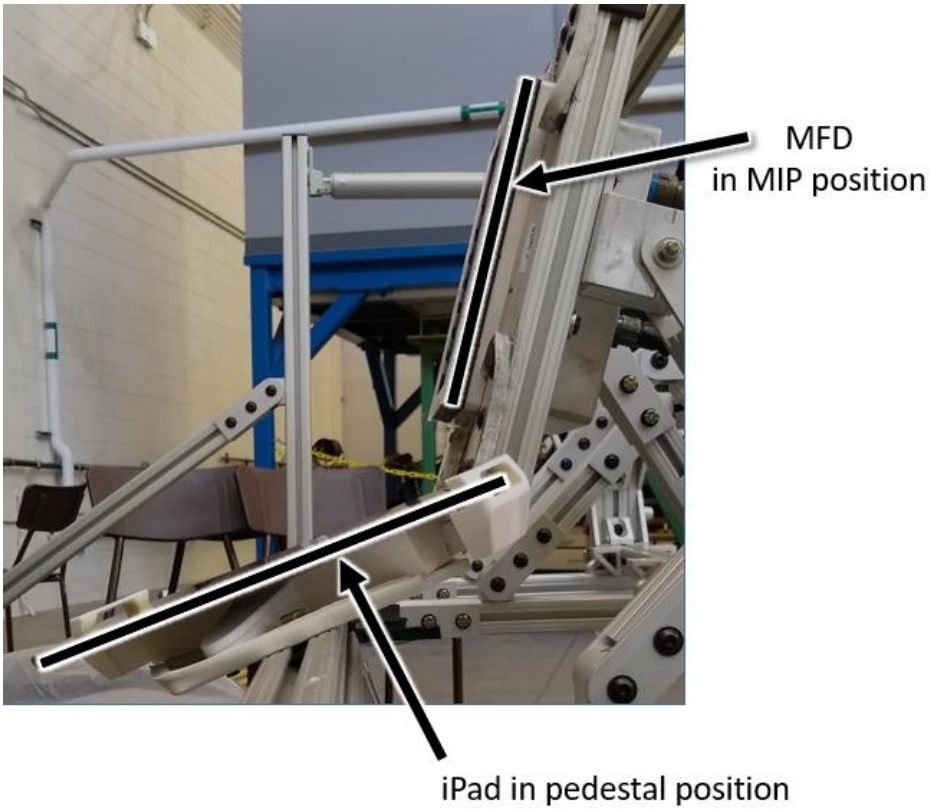


Figure 2.5 - Side view of the iPad in pedestal position and the MFD in MIP position. The glowing black lines denote the screens' edges. The top edge of the iPad is approximately aligned, in height and longitudinal position, with the bottom edge of the MFD.

In the pedestal position, the screen was placed at an angle of 20 degrees from horizontal. This angle was chosen to replicate the ergonomics of typical cockpits. This angle was validated by aerospace human factors specialists. In the pedestal position, the iPad was located at 21.3" vertical by 17.6" longitudinal by 11.3" lateral (54.2 cm x 44.7 cm x 28.7 cm) from the ERP. The Euclidian distance from the ERP to the center of the iPad, in pedestal position, is 29.9 inches (75.9 cm).

2.3 Hand Support

Two different hand support methods were tested:

1. Using the index finger freehand, without any hand, wrist or arm support. This was to test the "**no hand support**" condition.

2. Resting the fingers on the bezel edge of the screen and using the thumb to select the targets. This was to test the "**hand support**" condition, as described in previous academic research and in the SAE ARP60494 standard [12-14, 16].

Both of these conditions were tested on the MFD, MCDU and iPad screens, in both positions. The large Planar touch monitor was too large to both hold onto the side of the screen and reach all the targets. As a result, it was not possible to test the hand support condition on this screen. For the Planar touch monitor, only the freehand index finger condition was tested.

Figure 2.6 and Figure 2.7 show the no-hand-support condition, using the index finger freehand:

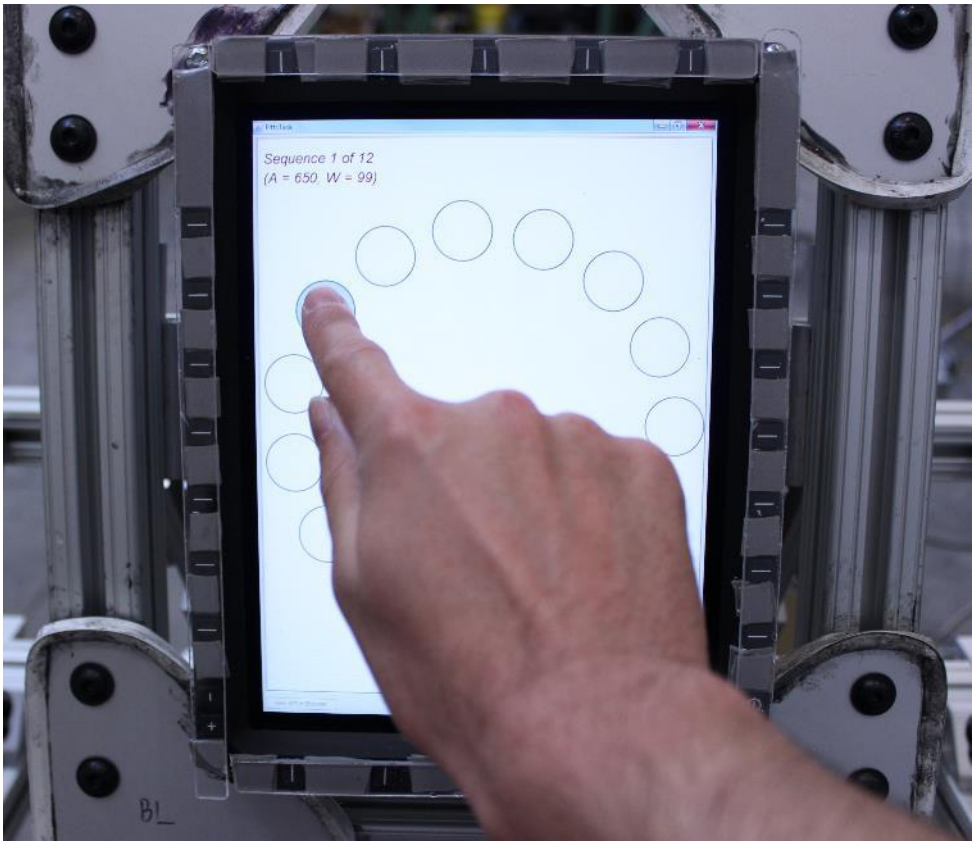


Figure 2.6 - Using the index finger for target selection, without support, on the MFD screen.

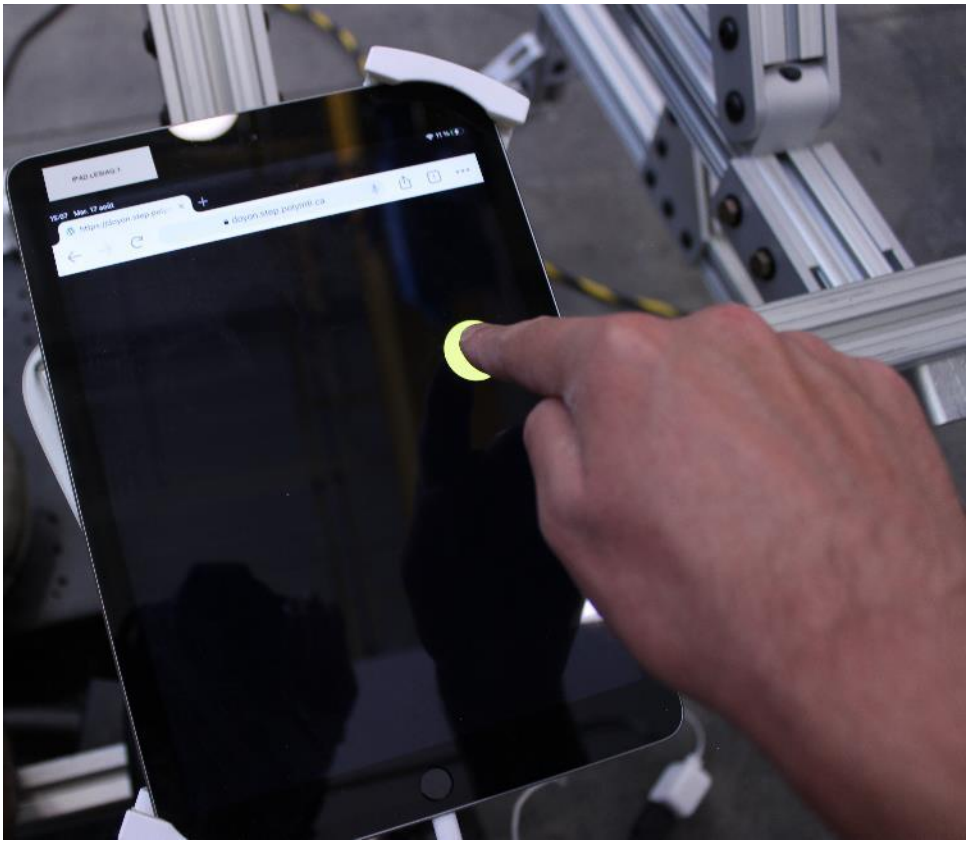


Figure 2.7 – Using the index finger, without support, for target selection on the iPad, in the pedestal position.

Figure 2.8 and Figure 2.9 show the hand support technique, using the thumb for target selection while supporting the fingers on the edge of the screen.

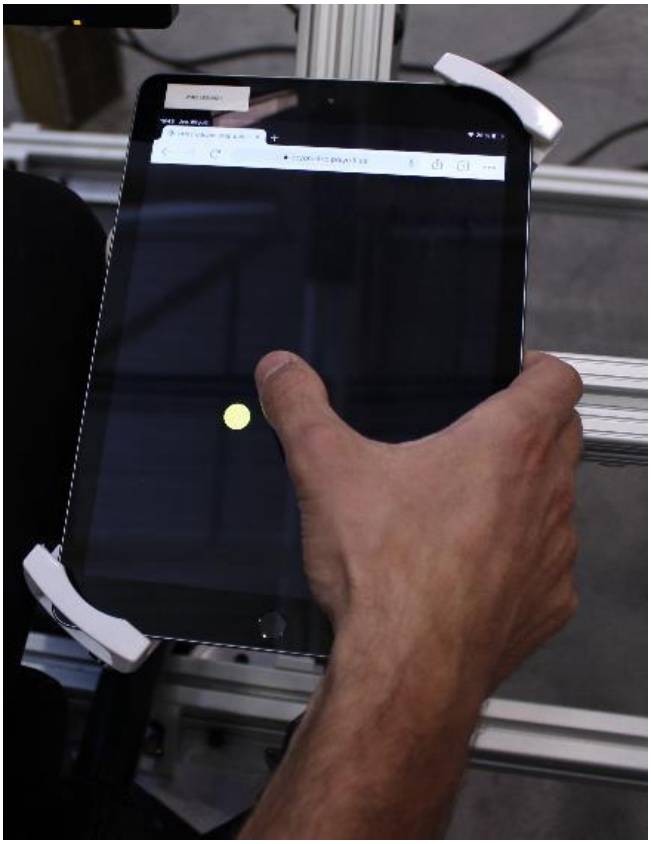


Figure 2.8 – Hand support condition onto the iPad in pedestal position.

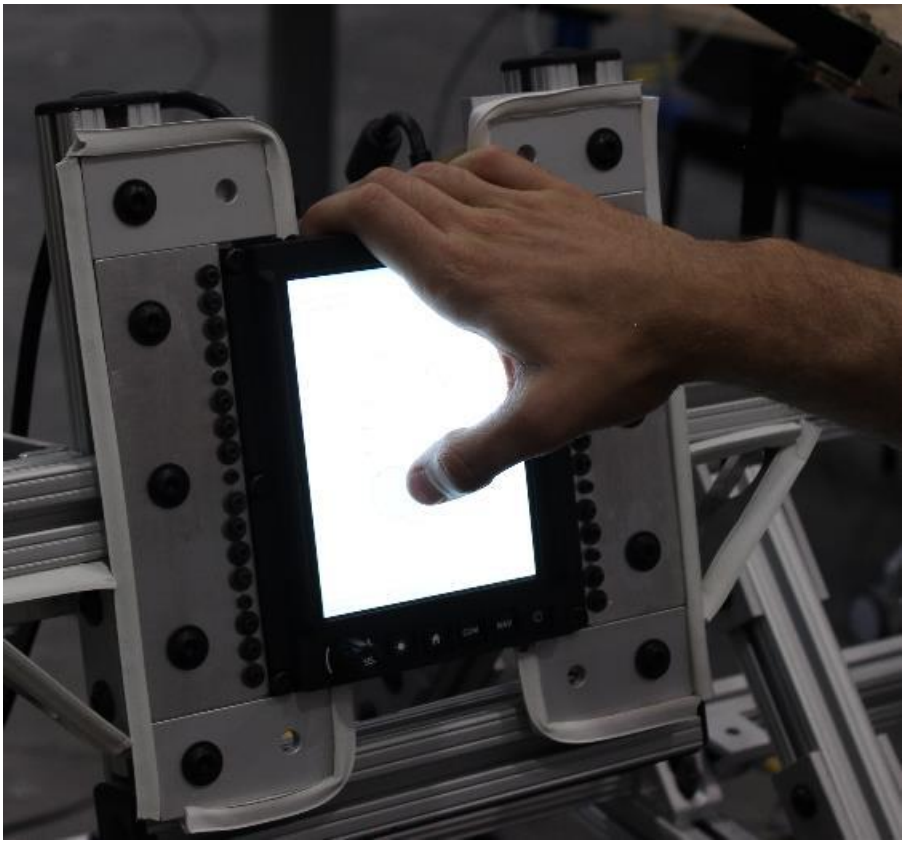


Figure 2.9 – Hand support condition onto the MCDU screen.

When using the MCDU screen in the hand support condition, participants supported their hand on the top of the screen (Figure 2.9). This was because the way this screen connected into its frame made it difficult to grab onto the side of the screen. For all other screens, in the hand support condition, the participants supported their hand on the side of the screen. Participants were allowed to move their hand along the edge of the screens. They were allowed to switch between grabbing onto the top and side of the screen, if they so chose; but they generally did not, since this movement between support positions slowed them down.

For more details on the bezel edges of the screens, and for a description of why these bezel edges were used as hand supports instead of pull-handles, refer to Appendix B.

2.4 Vibration Profile

2.4.1 Target Vibration Profile

The target vibration profile used in this study was provided by the National Research Council of Canada's Flight Research Laboratory (the NRC). They measured the vibration level in the cockpit of a Bell-412 helicopter, using an accelerometer. The NRC took measurements in several different cockpit locations and flight conditions. Members of the NRC team used these accelerometer recordings to drive their vibration platform, for several different research studies [11, 59-61]. The research articles published from these studies provide further detail into the placement of the accelerometers and the methods used to take the measurements.

The NRC provided us with accelerometer recordings for the following flight conditions, measured in a Bell-412 helicopter:

- Level flight at 120 knot flight speed
- Level flight at 60 knot flight speed
- Hover

They provided us with measurements taken on the pilot seat cushion (X, Y and Z-Axis), on the pilot seat pan (X, Y and Z-axis), and on the pilot seat rail (the cockpit floor X and Z-axis). Only X, Y and Z-axis vibration was provided to us, so we did not receive acceleration values in any of the rotational axes.

We decided to use the 120 knot level flight condition, since it had the highest level of vibration out of the three flight conditions that were provided to us, while still being realistic for pilot touchscreen use[1]. The 120 knot level flight condition is also the one chosen for use in an NRC article by Wright-Beatty et al., and so had already been accepted as representative [60].

The NRC used a sampling rate of 2,048 Hz. The 120kt level flight recording contained 248,320 samples, meaning that it was 121.25 seconds long.

[1] If the vibration level is too high, for example under high turbulence, the pilots would likely be concentrating on flying and would avoid using the touchscreen until they reached a calmer zone, or else they might use backup input methods.

In the article by Wright-Beatty et al., they found that vibration in the Z-axis (vertical axis) was much higher than vibration in the other axes. As a result, they concluded that only Z-axis acceleration was necessary for a representative reproduction of vibration in the Bell-412 120 knot level flight condition [60].

We used the methods described in ISO 2631-1[2] to analyze the contribution of each axis to whole-body vibration exposure. Table 2.1 shows the weighted root-mean-square (wRMS) per axis, as well as the total for all axes. The weighting factors used to calculare the wRMS are higher for those frequencies which most impact the human body, and lower for those frequencies which impact it less [63]. Comparing the wRMS for just the Z-axis against the total wRMS from all axes, there is approximately a 10% difference, which was deemed relatively minor. As a result, we concluded that having solely Z-axis (vertical) vibration was acceptable for our study. Other than table Table 2.1, all other graphs and calculations in this document solely use Z-axis vibration, while ignoring the other axes.

Table 2.1 – ISO 2631-1 weighted root-mean-square calculations, per axis, for the 120 knot level flight condition, as measured on the pilot's seat

	X-axis	Y-axis	Z-axis	All Axes
Acceleration wRMS (m/s^2)	0.15	0.16	0.71	0.78

A Fourier transform of the acceleration versus time data was taken to obtain the frequency spectrum; i.e., acceleration versus frequency (see Figure 2.10). It shows that the vibration contains three large peaks at 5.4 Hz, 10.8 Hz, and 21.6 Hz. The NRC articles by Wickramasinghe et al.,

[2] The ISO 2631-1 standard provides weighting factors per 1/3rd octave frequency bands, up to a maximum of 400 Hz. This standard denotes each band by its center frequency, but does not specify the start and end frequency of each band. To calculate the start and end frequency of each band, the formulas provided in the ANSI S1.11 standard were used [62] *ANSI S1.11-2004: Specification for Octave-Band and Fractional-Octave-Band Analog and Digital Filters*, American National Standards Institute, Melleville, NY, February 19 2004. This standard allows two options for the value of the constant G, when preforming the calculations: using G=2 for base 2 calculation or G=10^(3/10) for base 10 calculation. In this study, we used G=10^(3/10).

and Yong et al., explain that these peaks are the result of N/rev harmonic frequencies caused by the helicopters rotors [59, 61].

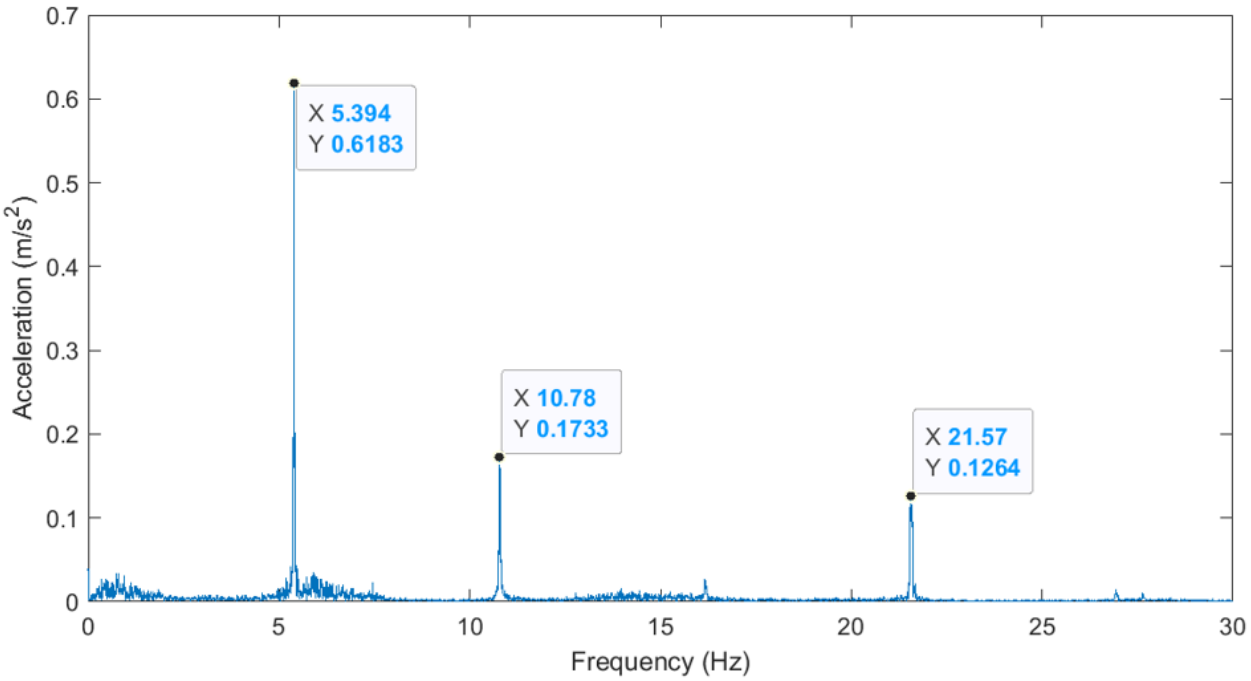


Figure 2.10 – Frequency spectrum of the acceleration in the Z-axis, for the vibration measured on the pilot's seat of a Bell-412 helicopter in 120kt level flight.

Figure 2.11 shows the Z-axis wRMS value for each 1/3rd octave frequency band, denoted by their center frequencies[3]; i.e., each band's impact on human exposure. It shows that the majority of the wRMS is coming from frequencies in the 1 Hz to 30 Hz range. As a result, it is reasonable to only use frequencies in this range, and to remove frequencies outside of this range, when replicating the vibration on the test setup.

[3] The acceleration RMS was calculated per 1/3rd octave frequency band. Each 1/3rd octave frequency band's RMS was then multiplied by the associated ISO 2631-1 weighting factor. To calculate the total RMS, the weighted RMS value from each band were squared, summed together, and then square-rooted.

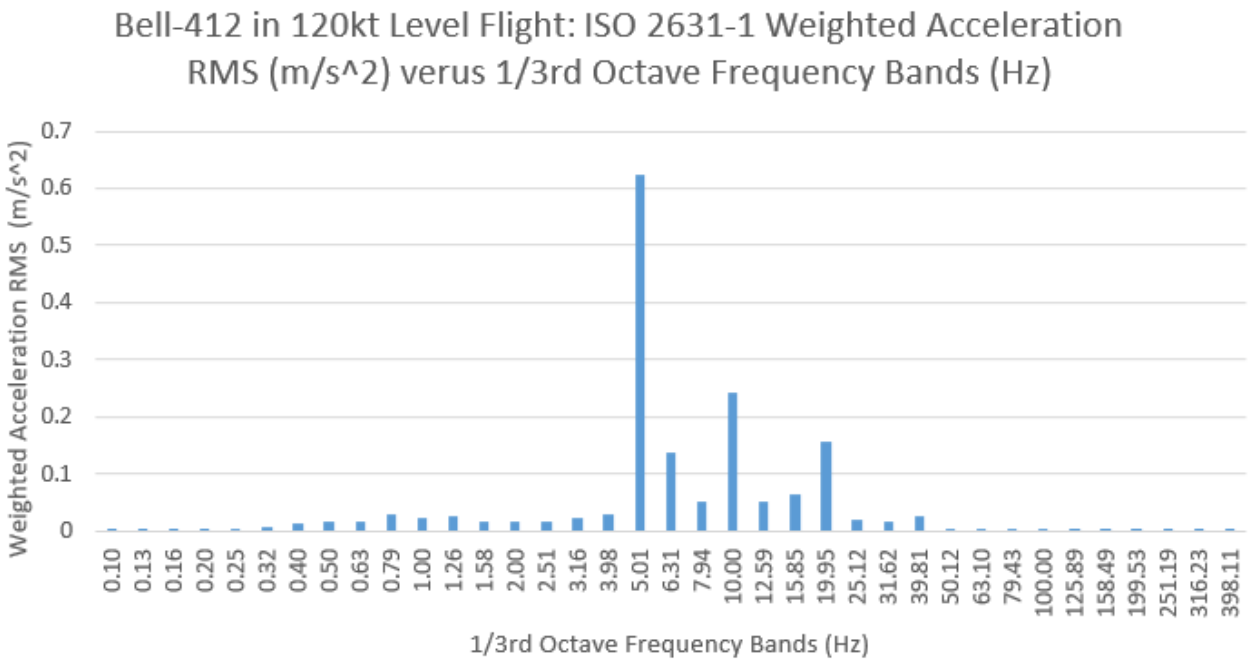


Figure 2.11 – Z-axis wRMS for each 1/3rd octave frequency band, for the vibration on the pilot's seat of a Bell-412 in 120kt flight.

2.4.1.1 D-Box Chair and DevSim Specifications

The GP Pro 500 chair, manufactured and sold by D-Box, was used to generate the vibration in this study. Figure 2.12 shows what the GP Pro 500 chair looked like, before the test structure was built onto it:

Figure 2.12 - Image of the D-Box GP Pro 500, before the test structure was built onto it.

Figure 2.13 shows that the GP Pro 500 chair looked like, after the test structure was built onto it, for this study:

Figure 2.13 - D-Box GP Pro 500 with test structure built onto it.

The D-Box GP Pro 500 has the following specifications [64]:

- 3 electric motors, two in the front, one in the back
- 94,207 motor steps (using optical position sensing)
- 3.4 cm total vertical travel (+/-1.7 cm)
- 400 pound (181 kg) capacity, not including the base chair and its original structure (which weighed 150 lbs/ 68 kg)
- +/- 100 mm/s maximum velocity [65]
- +/- 1 g maximum acceleration [65]

Two of the motors are positioned directly under the front part of the chair, while the back motor is positioned behind the chair:

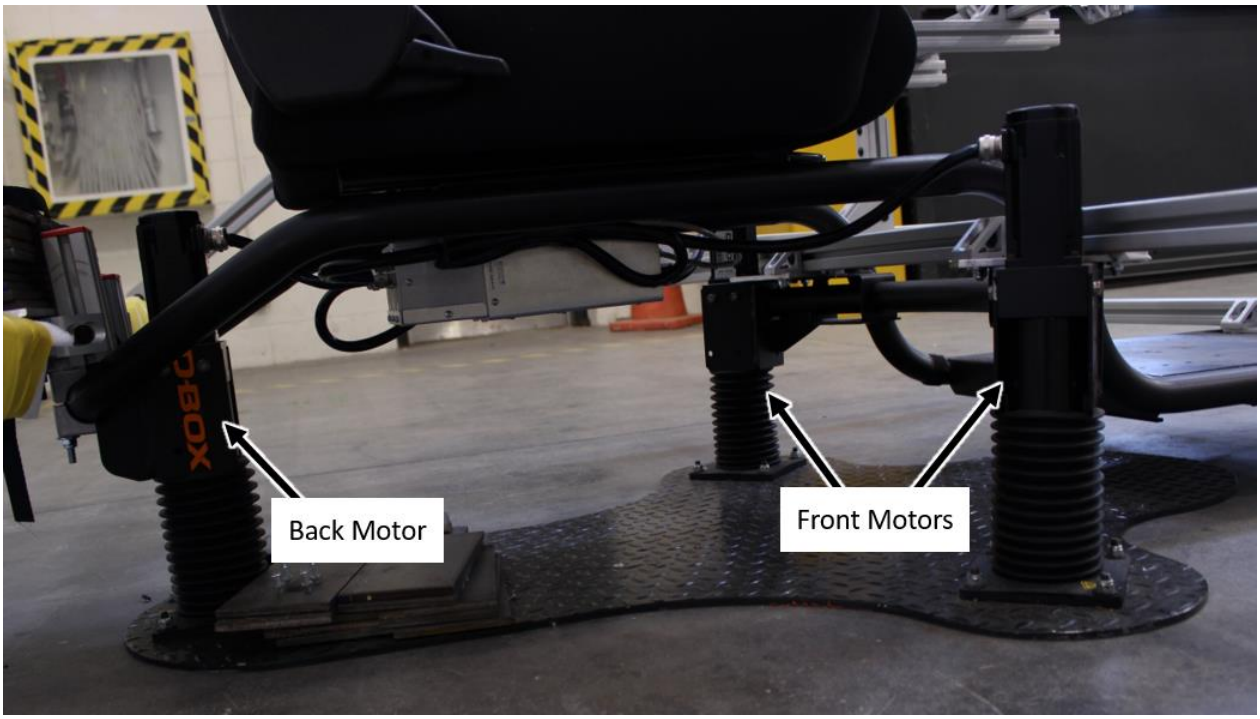


Figure 2.14 – Location of the three motors.

The chair can move up, down, tilt right, tilt left, tilt forward and tilt back, with the motors adjusting their height and angle accordingly.

By pulling on a lever, similar to a car seat, the chair can slide forward or backwards on a rail. When the lever is released, the chair's position is locked. The total travel of the chair along this rail is 7 inches (18 cm).

To program and control the movement of the GP Pro 500, we used D-Box's DevSim motion code, which is a set of C++ libraries. D-Box's DevSim only accepts position values and does not accept acceleration values. Appendix C explains the method used to convert from acceleration to position data to control the chair's motion. D-Box's DevSim contains a set of filters that limit the motion of the chair and added nuisance frequencies (called "mirror frequencies" in the rest of this document). Appendix D presents the frequency limitations of the chair and the corrections implemented to accurately replicate the helicopter vibrations.

2.4.2 Vibration Profile Used

Figure 2.15 presents the Fourier transform of the vibration profile used in this study, as measured on the D-Box chair. As can be seen by comparing it with Figure 2.10, the vibration peaks at 5.4 Hz and 10.8 Hz have been successfully replicated on the D-Box chair. The peak at 21.6 Hz has been lost. An unwanted additional peak at 14.2 Hz has been added, since it is the mirror frequency of 5.4 Hz, caused by DevSim (refer to Appendix D for more details).

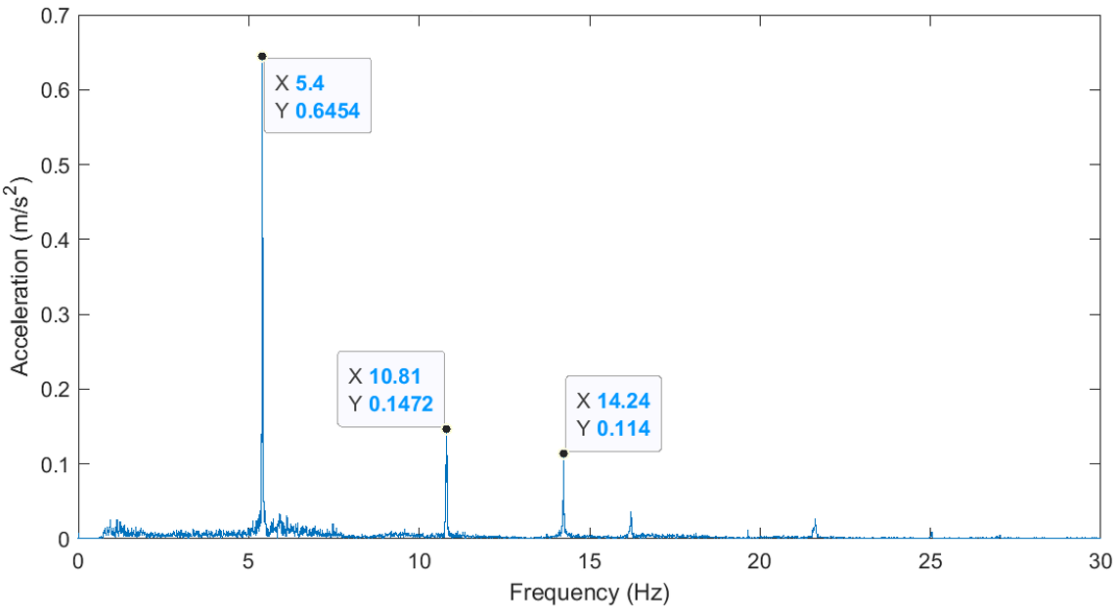


Figure 2.15 - Frequency spectrum of the acceleration used in the experiment, as measured on the D-Box chair.

To evaluate the accuracy of the vibration profile implemented, we compared the ISO 2631-1 wRMS per 1/3rd frequency octave of the original Bell-412 120kt vibration measurement against the final vibration measurement on the D-Box chair[4] (see Figure 2.16). The wRMS in each frequency band is very close, between the original and final vibrations. Only the 19.95 Hz band deviates noticeably. However, the magnitude of this band is much lower than the combined magnitudes of the other bands, and so this difference was considered to be negligible. Furthermore, Table 2.2 shows that the total acceleration wRMS value for the Z-axis measured on the D-Box

[4] wRMS values for the D-Box chair are an average across 7 recordings, as there were slight differences in the values measured in each recording.

chair was within 0.5% of the recording made on the pilot's seat of the Bell-412. In conclusion, the vibration profile used for this study replicated the vibration profile experienced by the pilot.

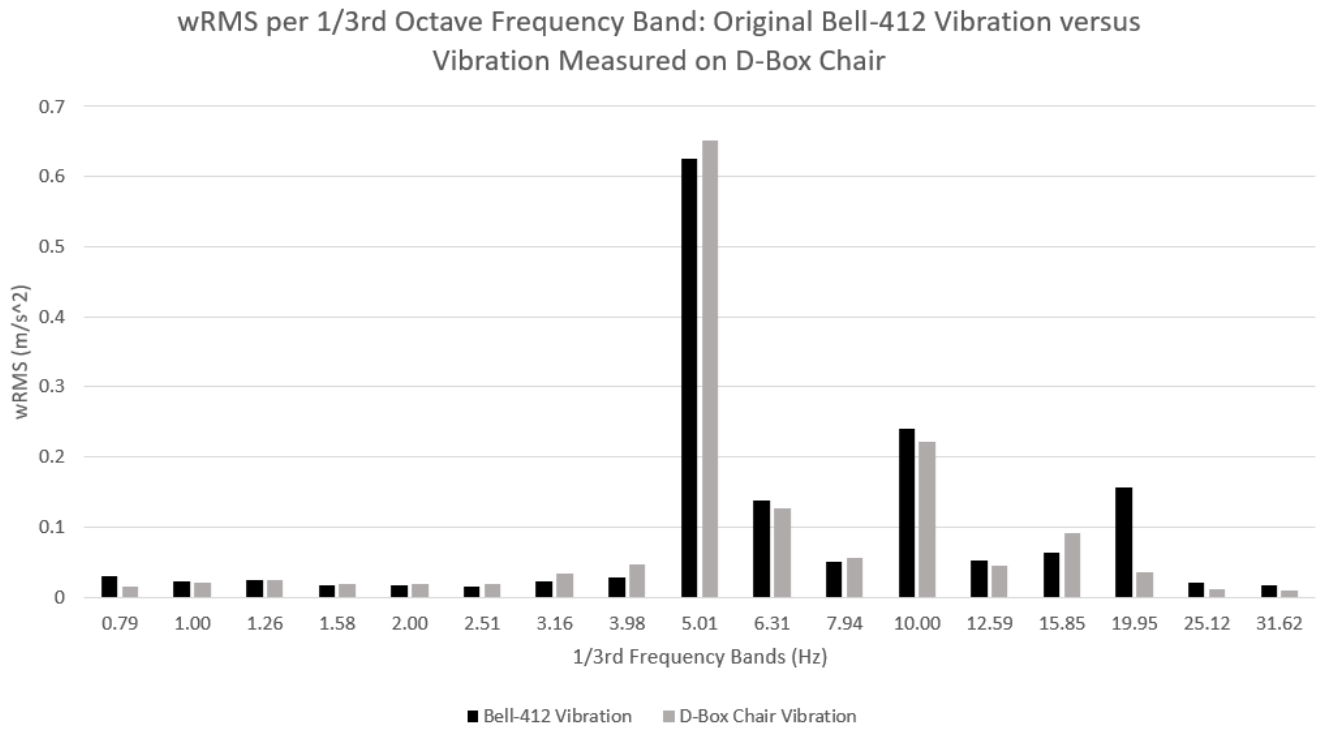


Figure 2.16 - ISO 2631-1 wRMS per 1/3rd frequency octave, comparing the original vibration profile measured in the Bell-412 helicopter against the vibration profile used in this experiment, as measured on the D-Box chair.

Table 2.2 – Total acceleration wRMS values in the Z-axis, original helicopter vibration versus final D-Box vibration

	Original Bell-412 120kt recording	**Vibration profile Measured on the D-Box Chair**
Total wRMS (m/s^2)	0.712	0.715

Cockburn et al. used vibration profiles with a maximum frequency of 5 Hz [13, 14]. Coutts et al. used vibration profiles in the range of 1-6Hz [16]. The vibration profile used in this present study has a broader range of frequencies than in those prior works. In addition, those previous articles used wRMS as their primary means of classifying the vibration and gave relatively little additional

detail about the shape of the vibration profiles: Cockburn et al. reported the mean and maximum frequencies, while Coutts et al. solely provided the frequency range. Therefore, since wRMS was used in prior work as the primary means of classifying vibration, we have considered it the primary means of determining whether the final D-Box chair vibration successfully replicated the original Bell-412 pilot's seat vibration.

2.4.2.1 Vibration Safety

To assess the safe exposure time of participants to the vibration profile, we used the ISO 2631-1 standard's health assessment calculations. It provides a maximum daily exposure limit for long term, whole-body vibration, if a worker were to be exposed to this vibration level every work day over a period of months or years [10]. Our study only exposed participants to vibration for around one hour on a single day.

However, we could not find an appropriate, widely recognized method for determining the impact of vibration on health over shorter exposure periods.

We assumed that using the values from ISO 2631-1 would offer a more stringent safety limit: if the vibration exposure is safe for long-term daily exposure, it is definitely safe for exposure on a single day.

The entire study took around 2 hours per participant. Half the conditions were performed under vibration, and half were performed without vibration. Overall, participants were exposed to the vibration for a maximum of 1 hour.

Figure 2.17 presents the three risk zones from ISO 2631-1. On the y-axis is the acceleration wRMS value. On the x-axis is the exposure time per day (in seconds). We added this study's vibration level of 0.71 m/s^2 wRMS, and found that the time to remain within the minimal risk zone is slightly over 3 hours. With an exposure time of 1 hour, the vibration level used in this study was well within the safety guidelines set out in ISO 2631-1.

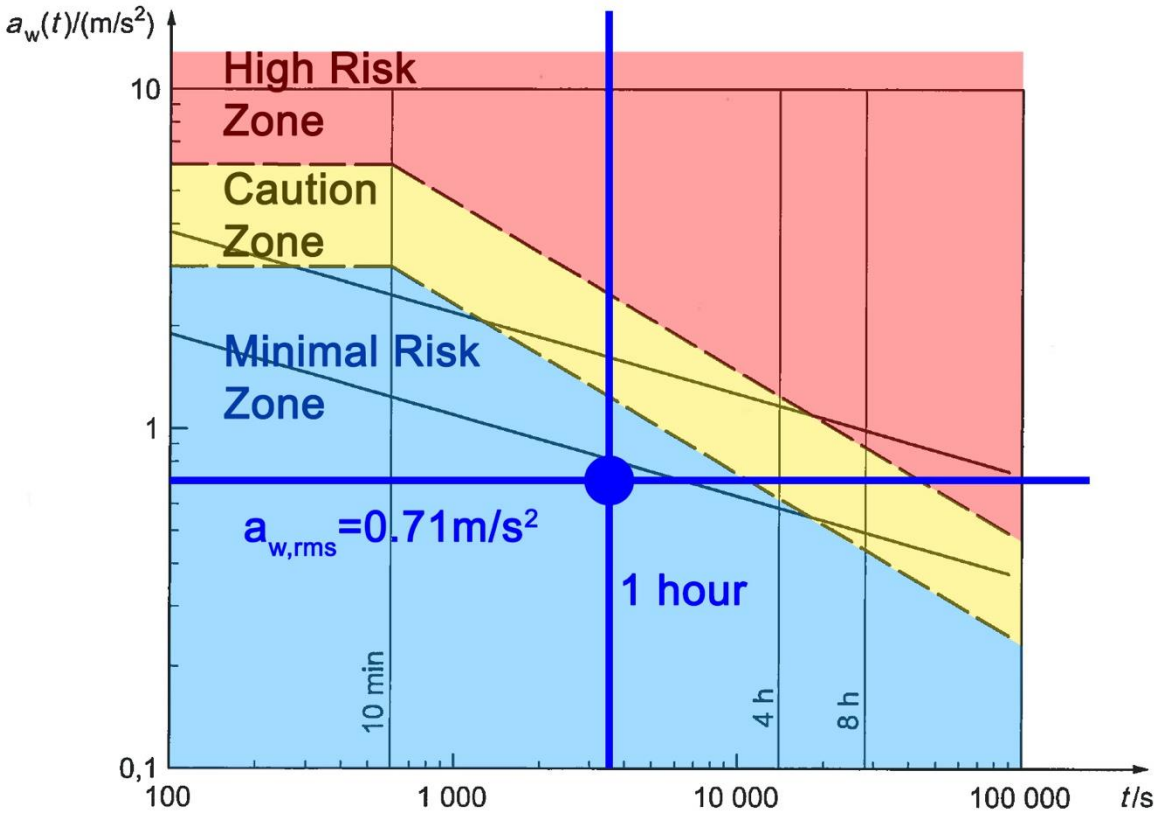


Figure 2.17 – Risk zones for maximum daily exposure to whole-body vibrations, taken from [10]. Blue lines show the wRMS used in this study.

2.4.2.2 Vibration On Seat Versus Screens

Note that, when comparing the vibration on the seat to the vibration on the screens, it makes more sense to use RMS rather than wRMS, since wRMS is a measure of human exposure to whole-body vibration, which does not apply to the screens. As a result, all values shown in this subsection are unweighted RMS values.

The NRC did not provide us with accelerometer data measured on the main instrument panel or pedestal. A detailed analysis of the vibration profile on the main instrument panel in a Bell-412 helicopter is thus impossible without such data. However, the NRC team did record Z-axis vibration on the Bell-412's instrument panel, even if this data was not provided to us. Wickramasinghe [11], published the acceleration RMS values and found that, during 120kt level flight in a Bell-412, Z-axis acceleration RMS on the instrument panel was approximately 3.4 times

higher than on the pilot's seat (0.47 g-RMS on the instrument panel, versus 0.14 g-RMS on the pilot's seat).

Table 2.3 shows the total RMS values, in the Z-axis, as measured on the D-Box chair seat, versus on each screen used in this experiment. Three recordings were taken per screen, with the table showing the average. The acceleration RMS measured on the screens is, on average, 3 times higher than on the D-Box seat. This is relatively close to the relative RMS values reported by [11]. The overall vibration level on the screens is relatively representative of the conditions we are trying to replicate.

Table 2.3 – Z-axis total RMS values, as measured on the D-Box chair, versus on each screen.

Location of Accelerometer	Total RMS (m/s^2)	Ratio of Screen to Seat RMS
D-Box Chair Seat	0.71	-
iPad MIP	2.76	3.88
iPad Pedestal	2.69	3.79
Large Touch Monitor	1.76	2.47
MFD Screen	1.47	2.06
MCDU Screen	1.89	2.67
All MIP Screens Averaged	1.94	2.73
All Screens Averaged (including both iPad MIP and Pedestal)	2.09	2.93

Figure 2.18 shows the RMS, per 1/3rd octave frequency band, as measured on the seat of the D-Box chair versus on the screens in the MIP position. The values for the screens have been averaged across all screens used in this experiment. While it is no longer necessary to use 1/3rd octaves, since the following RMS graph is unweighted, they still provide a useful method to compare vibrational energy per frequency zone. As can be seen, much of the vibrational energy on the screens is concentrated around 10 Hz. The structure holding the screens likely had a harmonic around 10 Hz, which amplified the vibration at this frequency. Refer to appendix E for details on the structure and the features added to mitigate internal vibration.

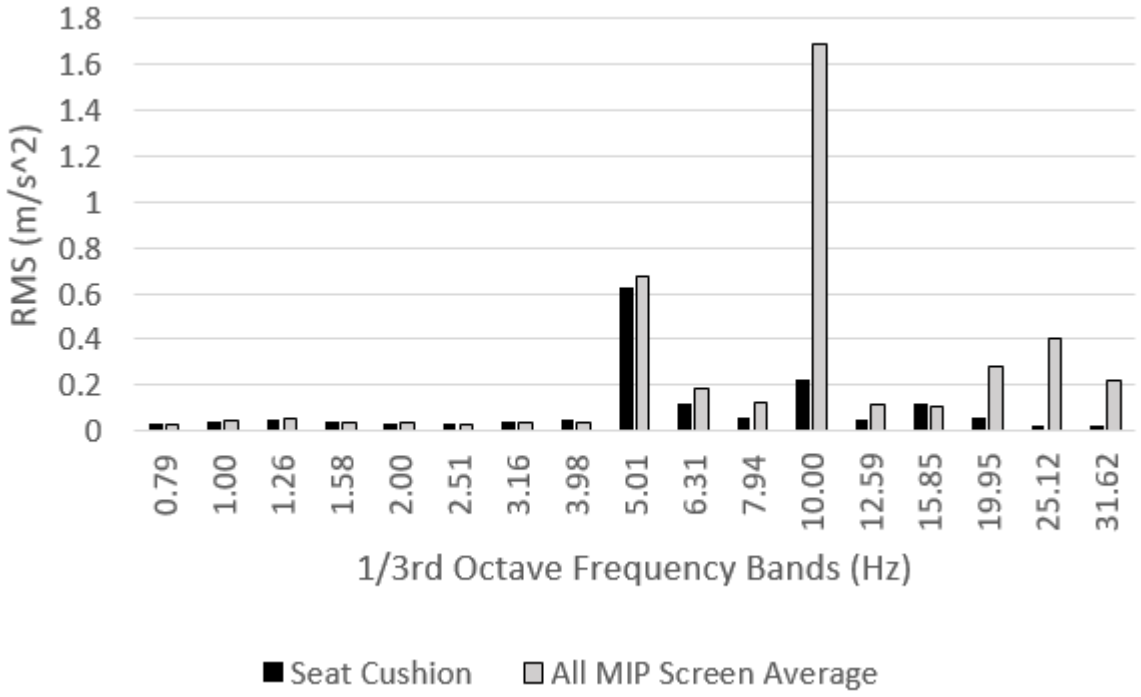


Figure 2.18 –Acceleration RMS per 1/3rd octave frequency bands measured on the seat cushion (in black) and the screens in the MIP position (in grey).

2.5 Experimental Conditions

This section presents the independent variables, test conditions and dependent variables.

2.5.1 Independent variables

This study had seven independent variables:

1. Four screens (MFD, MCDU, large touch monitor, and iPad)
2. Two screen positions (MIP and pedestal)
3. Two hand support methods (using the thumb while supporting the hand on the screen's bezel edge, versus using the index finger with the hand unsupported)

4. Two vibration conditions (with vibration and without vibration)
5. Target size
6. Distance between targets. GoFitts, the software used to run the experiment, uses the word "amplitude" to describe the distance between targets [47], and so we will use the same word for the rest of this document.
7. Selection angle (angular position of the target)

Out of these seven variables, the first four are being considered the primary independent variables: screen, screen position, hand-support method, and vibration. The last three are being considered secondary independent variables: target size, amplitude and selection angle. This distinction is being made because Fitts' law tasks require a range of target sizes, amplitudes and selection angles for a representative throughput calculation [18, 24], with throughput being one the main dependent variables analyzed in this study.

The test conditions described in this document are made up of combinations of the primary independent variables. All combinations of the secondary independent variables were tested in each test condition.

We were unable to conduct a fully factorial test plan because of the following:

- Only the iPad was tested in both the MIP and pedestal positions
- The large Planar touch monitor was too big to select all targets by supporting the hand on the bezel edge. This screen was only tested using the index finger, freehand.

This provides the following 18 test conditions.

$$\begin{aligned} Conditions = {} & (2\ screens) * (1\ position) * (2\ support\ methods) * (2\ vibration\ states) \\ & + (1\ screen) * (2\ positions) * (2\ support\ methods) * (2\ vibration\ states) \\ & + (1\ screen) * (1\ position) * (1\ support\ methods) * (2\ vibration\ states) \\ & = 18 \end{aligned}$$

The participants followed a within-subject plan

Table 2.4 and Table 2.5 present the coding system to identify test conditions in the rest of this document.

Table 2.4 – Coding system to identify test conditions

Screens	Screen Position	Hand Support Method	Vibration Environment
M (**M**edium screen. MFD)	**[blank]** (MIP)	**F** (index **F**inger. Unsupported hand)	**U** (**U**nmoving. Static)
S (**S**mall screen. MCDU)	**P** (Pedestal)	**T** (**T**humb. Hand supported on bezel edge)	**V** (**V**ibration)
L (**L**arge screen. Planar touch monitor)			
i (**i**Pad)			

Table 2.5- Table showing the 18 test conditions, using the system to code for screen, hand support method, vibration environment and screen position.

#	Conditions
1	MFU
2	MFV
3	MTU
4	MTV
5	SFU
6	SFV
7	STU
8	STV
9	LFU
10	LFV
11	iFU
12	iFV
13	iTU
14	iTV
15	iFUP
16	iFVP
17	iTUP
18	iTVP

2.5.1.1 Balancing Conditions Across Participants

Although the test structure was built to be able to swap between screens relatively quickly, this process was still time consuming. In an effort to minimize the amount of time spent switching between screens, participants completed, in a block, all hand support methods and vibration conditions for a given screen, before moving on to the next screen (for the purposes of this discussion, the iPad in the MIP position and the iPad in the pedestal position are considered as though they were two different screens).

Table 2.6 shows this visually, with each block highlighted in a different colour. The conditions within each block always stayed grouped together, although the order of blocks changed and the order of conditions within each block changed, per participant.

Table 2.6 – Table showing each block of conditions, highlighted in a different colour.

Condition Blocks
MFU
MFV
MTU
MTV
SFU
SFV
STU
STV
LFU
LFV
iFU
iFU
iFU
iFU
iFUP
iFUP
iFUP
iFUP

So as not to confuse the participants, the two conditions per hand support method (index finger/thumb) were also grouped together in a sub-block. The following table shows just the M block (for the MFD screen). Note that the two conditions with F are grouped together into a sub-block, whereas the two conditions with T are grouped together into a different sub-block.

Table 2.7 –Table showing the M block, with the two sub-blocks highlighted in different shades.

Condition Blocks
MFU
MFV
MTU
MTV

Again, these two conditions per sub-block always remained together, although their order alternated, and the order of vibration conditions within each sub-block also alternated. This is shown visually in the following table:

Table 2.8 – From one participant to the next, the order of blocks was always changed. In addition, the order of sub-blocks always swapped. Every two participants, the order of vibration conditions within the sub-blocks also swapped.

Participant 1	Participant 2	Participant 3	Participant 4
M-F-U	A-T-U-P	A-F-V	L-F-V
M-F-V	A-T-V-P	A-F-U	L-F-U
M-T-U	A-F-U-P	A-T-V	A-T-V
M-T-V	A-F-V-P	A-T-U	A-T-U
S-F-U	M-T-U	A-F-V-P	A-F-V
S-F-V	M-T-V	A-F-U-P	A-F-U
S-T-U	M-F-U	A-T-V-P	A-T-V-P
S-T-V	M-F-V	A-T-U-P	A-T-U-P
L-F-U	S-T-U	M-F-V	A-F-V-P
L-F-V	S-T-V	M-F-U	A-F-U-P
A-F-U	S-F-U	M-T-V	M-T-V
A-F-V	S-F-V	M-T-U	M-T-U
A-T-U	L-F-U	S-F-V	M-F-V
A-T-V	L-F-V	S-F-U	M-F-U
A-F-U-P	A-T-U	S-T-V	S-T-V
A-F-V-P	A-T-V	S-T-U	S-T-U
A-T-U-P	A-F-U	L-F-V	S-F-V
A-T-V-P	A-F-V	L-F-U	S-F-U

As seen in Table 2.8, the sub-blocks and vibration conditions were alternated for each participant. The sub-block alternates between each sequential participant, while the vibration conditions within each sub-block alternate every two participants. In such a way, each condition was balanced. An equal number of participants were presented with thumb before index finger as index finger before thumb. An equal number of participants were presented with vibration before static as static before vibration.

This study had 24 participants. For the first 12, the last block of the previous participant became the first block for the next participant.

The order of blocks for the first participant was:

$Block_{1,1}$ = M

$Block_{1,2}$ = S

$Block_{1,3}$ = L

$Block_{1,4}$ = A

$Block_{1,5}$= AP

The order of blocks for the second participant was:

$Block_{2,1}$ = AP

$Block_{2,2}$ = M

$Block_{2,3}$ = S

$Block_{2,4}$ = L

$Block_{2,5}$= A

In essence, we denote a variable Block, with subscripts i and n:

$$Block_{i,n}$$

"i" represents the participant number, while "n" represents the position of the block in the sequence.

For the first 12 participants, the following formula was used to calculate the order of each block:

$$Block_{i,n} = \begin{cases} Block_{i-1,5}, & n = 1 \\ Block_{i-1,n-1}, & n > 1 \end{cases}$$

For the second 12 participants, first the same formula was used, but then the order of the blocks was inversed.

In essence, first use the previously shown formula for all 24 participants:

$$Block_{i,n} = \begin{cases} Block_{i-1,5}, & n = 1 \\ Block_{i-1,n-1}, & n > 1 \end{cases}$$

Then apply the following formula just for participants 13-24, to reverse the order:

$$Block_{i,n} = Block_{i,6-n}$$

This reverse order was done to control for any order effects from one screen following after another. In particular, we were concerned about order effects on the iPad, since this screen was placed in two positions. Each screen had slightly different sensitivity and responsiveness. So, when presented with a new screen, users may have adapted to the sensitivity and responsiveness over time. Since the iPad was tested in two positions, users may have been more accustomed to its sensitivity and responsiveness when they tried it in the second position. By reversing the order so that half the participants saw the iPad in MIP position first, and the other half saw it in pedestal position first, we controlled for this order effect.

Overall, the order that was planned, for each participant, was:

Table 2.9 – Table showing the order of conditions for each participant

Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6	Participant 7	Participant 8	Participant 9	Participant 10	Participant 11	Participant 12
MFU	iFUP	iFU	LFV	SFU	MTU	iFUP	iFU	LFU	STU	MFV	iFUP
MFV	iFUP	iFU	LFU	SFV	MTV	iFUP	iFU	LFV	STV	MFU	iFUP
MTU	iFUP	iFU	iFU	STU	MFU	iFUP	iFU	IFU	SFU	MTV	iFUP
MTV	iFUP	iFU	iFU	STV	MFV	iFUP	iFU	iFU	SFV	MTU	iFUP
SFU	MTU	iFUP	iFU	LFU	STU	MFV	iFUP	iFU	LFU	SFV	MTV
SFV	MTV	iFUP	iFU	LFV	STV	MFU	iFUP	iFU	LFV	SFU	MTU
STU	MFU	iFUP	iFUP	iFU	SFU	MTV	iFUP	iFUP	iFU	STV	MFV
STV	MFV	iFUP	iFUP	iFU	SFV	MTU	iFUP	iFUP	iFU	STU	MFU
LFU	STU	MFV	iFUP	iFU	LFU	SFV	MTV	iFUP	iFU	LFV	STV
LFV	STV	MFU	iFUP	iFU	LFV	SFU	MTU	iFUP	iFU	LFU	STU
iFU	SFU	MTV	MTV	iFUP	iFU	STV	MFV	MFU	iFUP	iFU	SFV
iFU	SFV	MTU	MTU	iFUP	iFU	STU	MFU	MFV	iFUP	iFU	SFU
iFU	LFU	SFV	MFV	iFUP	iFU	LFV	STV	MTU	iFUP	iFU	LFV
iFU	LFV	SFU	MFU	iFUP	iFU	LFU	STU	MTV	iFUP	iFU	LFU
iFUP	iFU	STV	STV	MFU	iFUP	iFU	SFV	SFU	MTU	iFUP	iFU
iFUP	iFU	STU	STU	MFV	iFUP	iFU	SFU	SFV	MTV	iFUP	iFU
iFUP	iFU	LFV	SFV	MTU	iFUP	iFU	LFV	STU	MFU	iFUP	iFU
iFUP	iFU	LFU	SFU	MTV	iFUP	iFU	LFU	STV	MFV	iFUP	iFU

Table 2.9 - Table showing the order of conditions for each participant (continued and end)

Participant 13	Participant 14	Participant 15	Participant 16	Participant 17	Participant 18	Participant 19	Participant 20	Participant 21	Participant 22	Participant 23	Participant 24
LFU	STU	MFV	iFUP	iFU	LFU	SFV	MTV	iFUP	iFU	LFV	STV
LFV	STV	MFU	iFUP	iFU	LFV	SFU	MTU	iFUP	iFU	LFU	STU
SFU	SFU	MTV	iFUP	iFU	STU	STV	MFV	iFUP	iFU	SFV	SFV
SFV	SFV	MTU	iFUP	iFU	STV	STU	MFU	iFUP	iFU	SFU	SFU
STU	MTU	iFUP	iFU	LFU	SFU	MFV	iFUP	iFU	LFU	STV	MTV
STV	MTV	iFUP	iFU	LFV	SFV	MFU	iFUP	iFU	LFV	STU	MTU
MFU	MFU	iFUP	iFU	SFU	MTU	MTV	iFUP	iFU	STU	MFV	MFV
MFV	MFV	iFUP	iFU	SFV	MTV	MTU	iFUP	iFU	STV	MFU	MFU
MTU	iFUP	iFU	LFV	STU	MFU	iFUP	iFU	LFU	SFU	MTV	iFUP
MTV	iFUP	iFU	LFU	STV	MFV	iFUP	iFU	LFV	SFV	MTU	iFUP
iFUP	iFUP	iFU	STV	MFU	iFUP	iFUP	iFU	SFU	MTU	iFUP	iFUP
iFUP	iFUP	iFU	STU	MFV	iFUP	iFUP	iFU	SFV	MTV	iFUP	iFUP
iFUP	iFU	LFV	SFV	MTU	iFUP	iFU	LFV	STU	MFU	iFUP	iFU
iFUP	iFU	LFU	SFU	MTV	iFUP	iFU	LFU	STV	MFV	iFUP	iFU
iFU	iFU	SFV	MTV	iFUP	iFU	iFU	STV	MFU	iFUP	iFU	iFU
iFU	iFU	SFU	MTU	iFUP	iFU	iFU	STU	MFV	iFUP	iFU	iFU
iFU	LFU	STV	MFV	iFUP	iFU	LFV	SFV	MTU	iFUP	iFU	LFV
iFU	LFV	STU	MFU	iFUP	iFU	LFU	SFU	MTV	iFUP	iFU	LFU

Recording errors were made for six of the participants, who needed to return to the laboratory, later on, to repeat some of the conditions on some of the screens. As a result, the screen order was different from the ideal for these six participants. However, this different order was included when analyzing whether there were any order effects, as will be described below, and no main order effects were found. This change in screen order for six of the participants did not cause skewed results.

2.5.2 2D Fitts' Selection Task

Participants performed a "multi-directional pointing task", as recommended by the ISO 9241-411 standard [24], which is equivalent to a 2D Fitts' selection task [6]. There are several ways of performing the throughput calculation from the results of a Fitts' law task [25, 26]. For this study,

we have used the recommendations laid out in a widely cited article by Soukoreff and MacKenzie [18], supplemented with additional detailed clarifications and recommendations from MacKenzie [23], and from the GoFitts application documentation [30].

2.5.2.1 GoFitts versus iPad Web Application

The 2D Fitts' selection task was conducted using the GoFitts software application [46], on all devices except for the iPad. The GoFitts software application is a Java script developed by I. Scott MacKenzie [18].

Figure 2.19 shows the MFD screen running the GoFitts software application. The currently active target is highlighted in purple/blue, while the other targets are shown using un-filled circles with a black outline. As a result, when using the GoFitts application, the participant can see the location of the next target, as well as all targets before and after it in the sequence.

Figure 2.19 - An image of the MFD screen running MacKenzie's GoFitts software application.

For the iPad, the Fitts' task was modeled after GoFitts, but was programmed by Philippe Doyon-Poulin and ran on a website using the Google Chrome web browser. We were unable to port the Java script from GoFitts onto the iPad and had to use a web app instead. In comparison to GoFitts, the iPad application uses yellow targets on a black background, and only one target location is shown at a time. The next target in the sequence appears only after the previous target has been selected (whether successfully or unsuccessfully). This is shown in Figure 2.20.

Figure 2.20 – Image of the Fitts' selection task website programmed by Philippe Doyon-Poulin. Only one target location is shown at a time. The active target is shown as a yellow circle on a black background.

The other difference between GoFitts and Doyon-Poulin's web application (used on the iPad) is that it had a different starting target. GoFitts' starting target is located on the positive horizontal axis. Whereas the iPad application's starting target is on the positive vertical axis. This angular distance between targets implies that the two target layouts are slightly offset from each other, by 6°. This is shown in Figure 2.21 and Figure 2.22, which have been creating using a screenshot from GoFitts as a base.

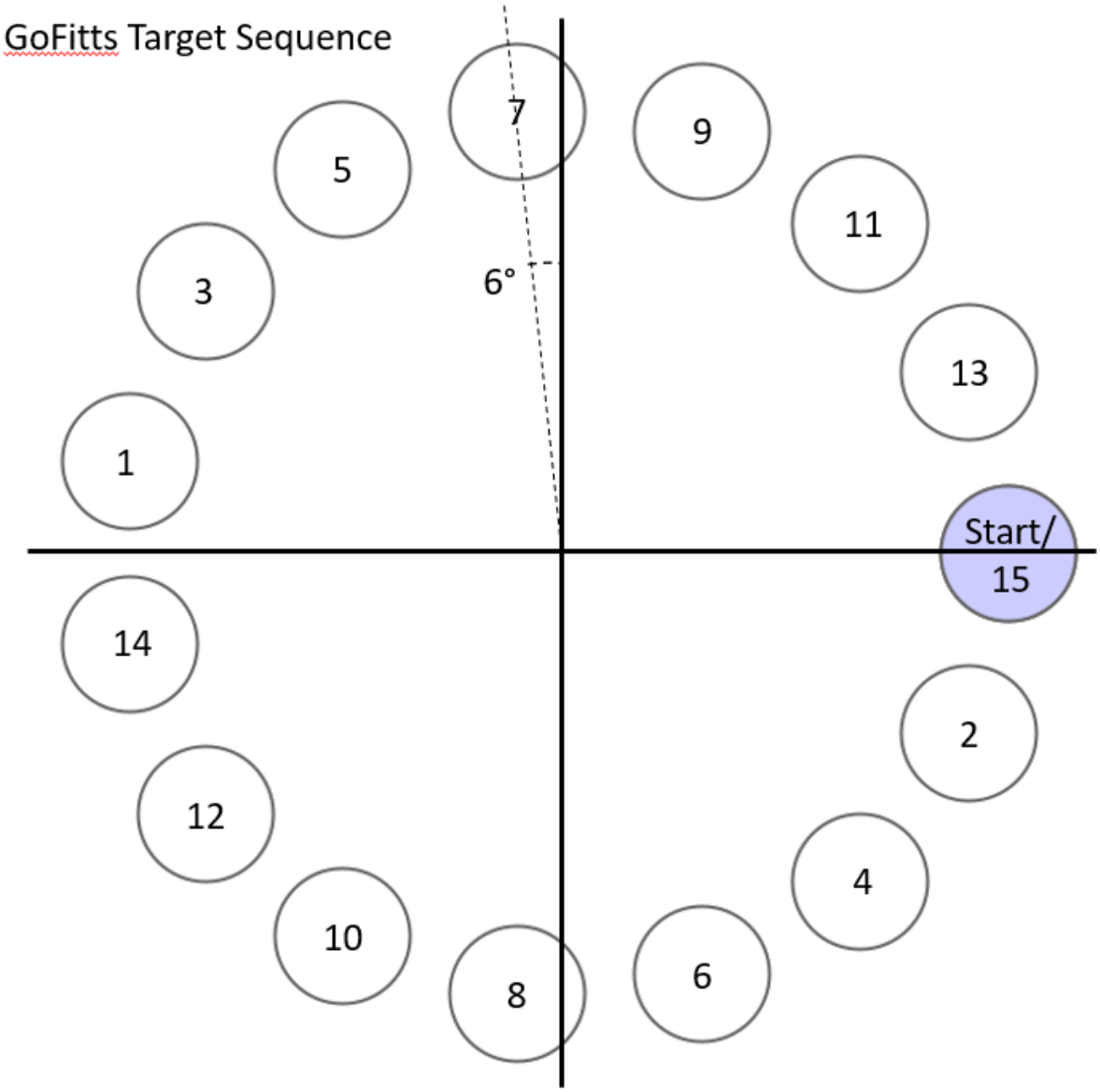


Figure 2.21 – GoFitts target selection sequence, using 15 targets. Note that the start and end target are the same (highlighted in purple), and that it lies on the positive horizontal axis. Note that target 7 is shifted -6° off the vertical axis.

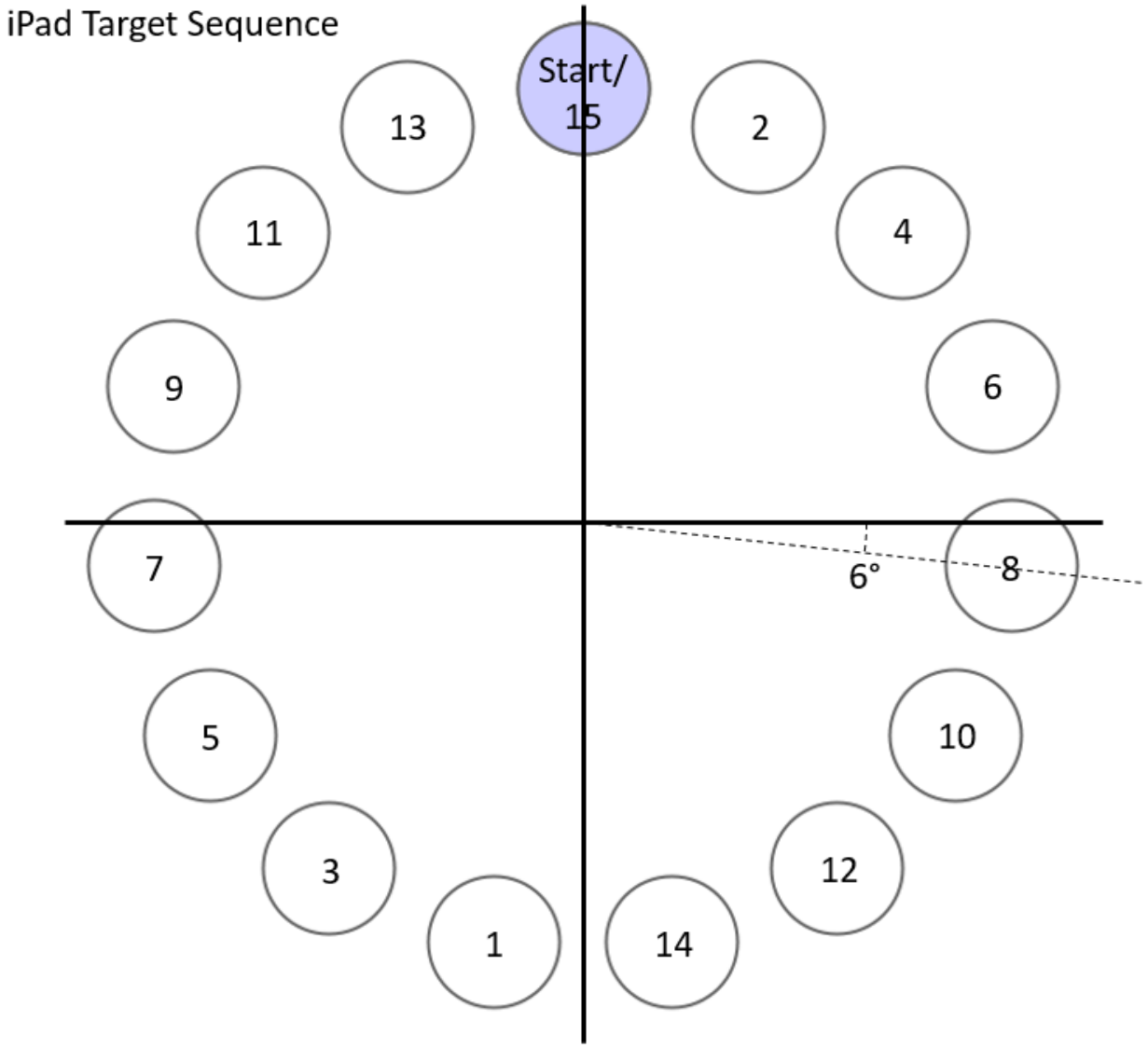


Figure 2.22 –iPad target selection sequence, using 15 targets. Note again that the start and end target are the same (highlighted in purple). However, this time, the start/end target lies on the positive vertical axis. Note that target 8 is shifted -6° off the horizontal axis.

As seen in Figure 2.21 and Figure 2.22, not only is the starting target different, for the iPad compared to the other screens (which ran GoFitts), but the targets are also slightly offset rotationally, by 6°. We considered this 6° offset to be minor, and ignored it in our analysis of the impact of selection angle.

Also note, in Figure 2.21 and Figure 2.22, that the starting and ending target are the same, for a given application. The starting target is selected twice, while all other targets are selected just once.

The reason that the starting target must again be selected at the end is that the Fitts calculation requires both movement distance and selection time [18]. On the first tap in the sequence, the software does not know where the finger is coming from, and when it started its motion. As a result, the first target must be selected again at the end, when the previous target location and selection time are known.

2.5.2.2 Methods used to Calculate Throughput

In this study, we followed the methods used in GoFitts to calculate throughput [30, 46, 47]. These methods follow the recommendations given in Soukoreff and MacKenzie [18], with some additional clarifications and recommendations on the subject of Ae* found in another article by MacKenzie [23]. This section goes into greater detail about the steps taken to calculate throughput, compared to the brief overview given in the introduction.

Throughput was calculated per trial. For each trial, the size of the targets and amplitude (distance between targets) remained constant. Each trial consisted of 15 targets. The following discussion explains how throughput was calculated for a given trial, using the data recorded from these 15 targets selections.

The following variables are relevant to this discussion:

A, W, MT, Ae, dx

"A" represents the distance between targets (for **A**mplitude). "W" represents target size (for **W**idth). MT represents movement time, and is the time difference between the current tap and the previous tap. "Ae" and "dx" require a graphic to clearly explain, since they involve a projection onto a line.

Figure 2.23 shows a representation of dx. It is the distance between the current tap and the current target center, projected onto the line between the previous target center and the current target center.

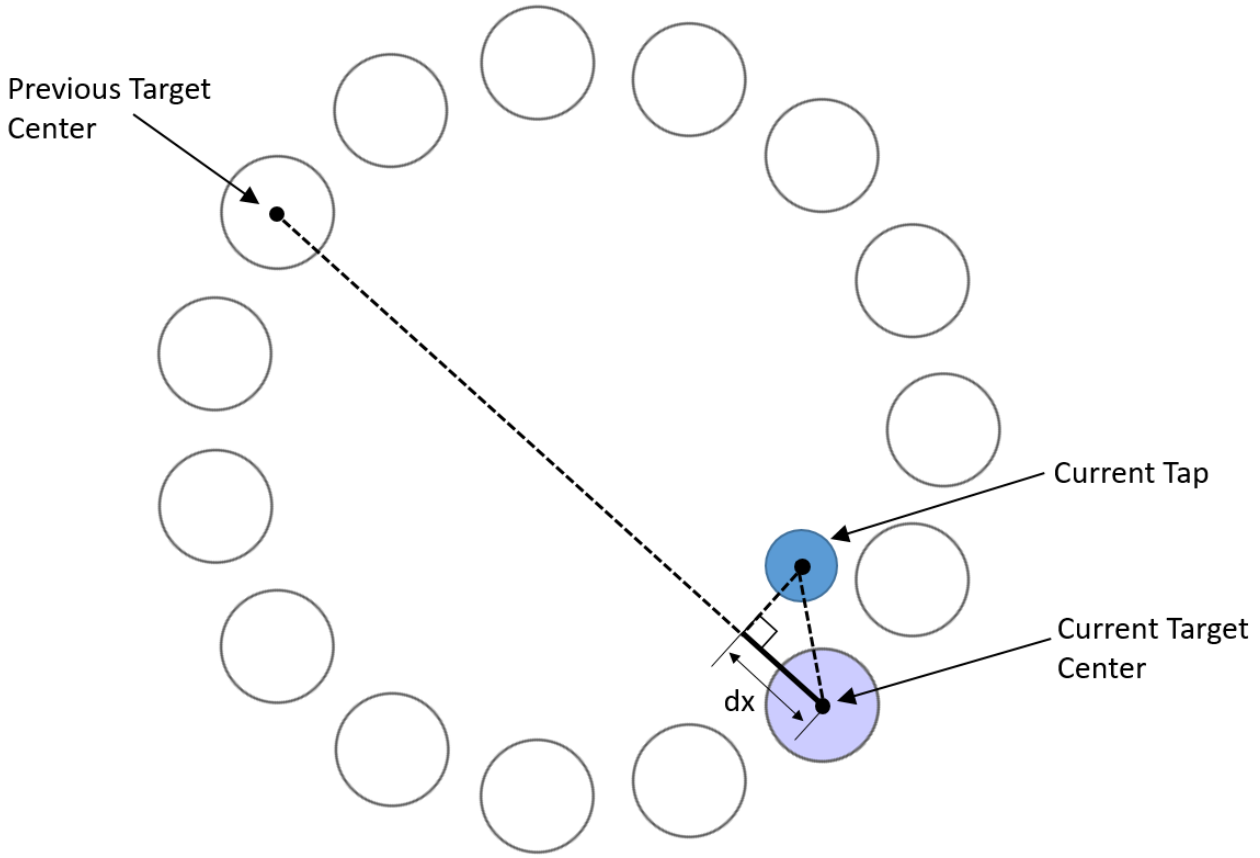


Figure 2.23 - dx is the distance between the current tap and the current target center, projected onto the line between the previous target center and the current target center.

Note that dx can take on either a positive or negative value, depending on whether it is outside or inside the circle of target centers, respectively.

Figure 2.24 shows a representation of Ae. Ae is the distance between the previous target center and the current tap, projected onto the line between the previous target center and the current target center.

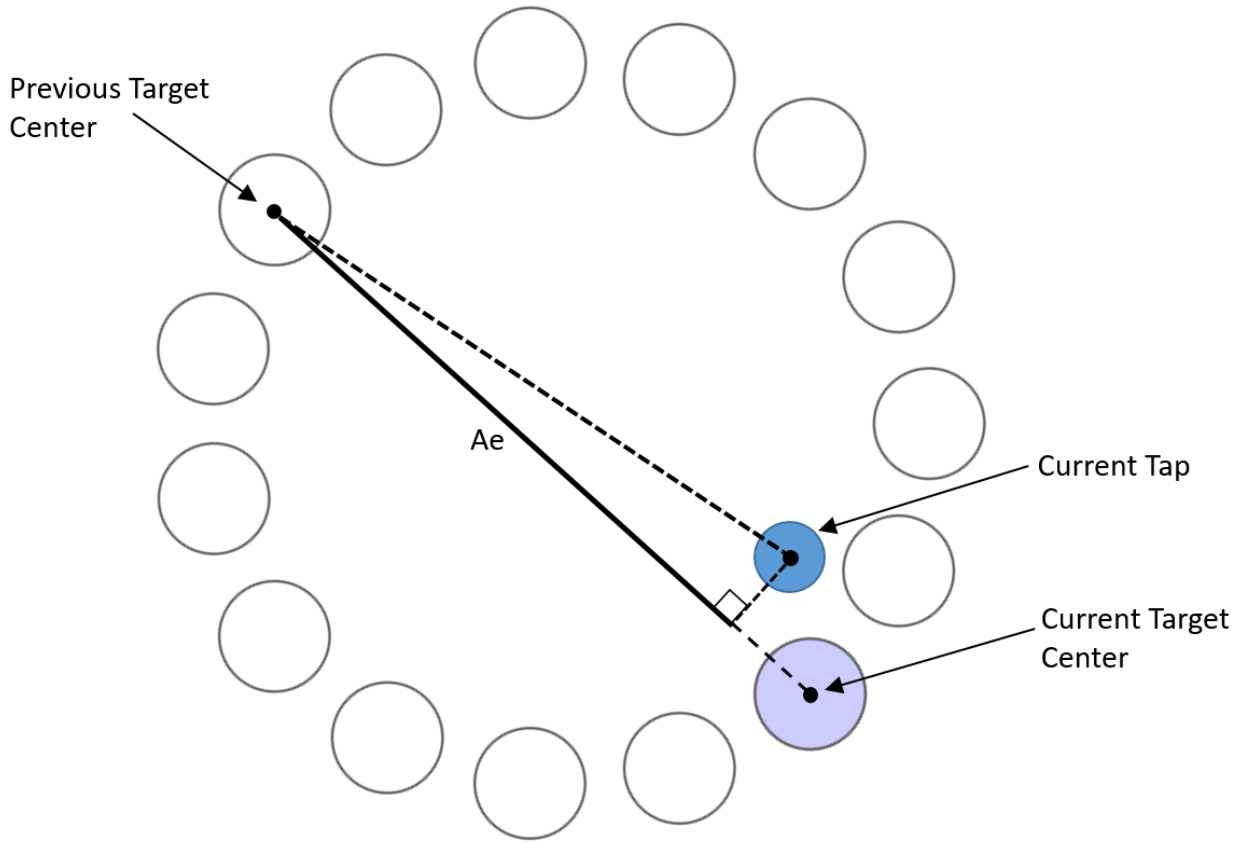


Figure 2.24 - Ae is the distance between the previous target center and the current tap, projected onto the line between the previous target center and the current target center.

MacKenzie recommends performing a correction on Ae [23, 30]. A better representation of movement distance would be the projected distance between the previous tap and the current tap. He proposes the use of Ae* for this purpose (see Figure 2.25).

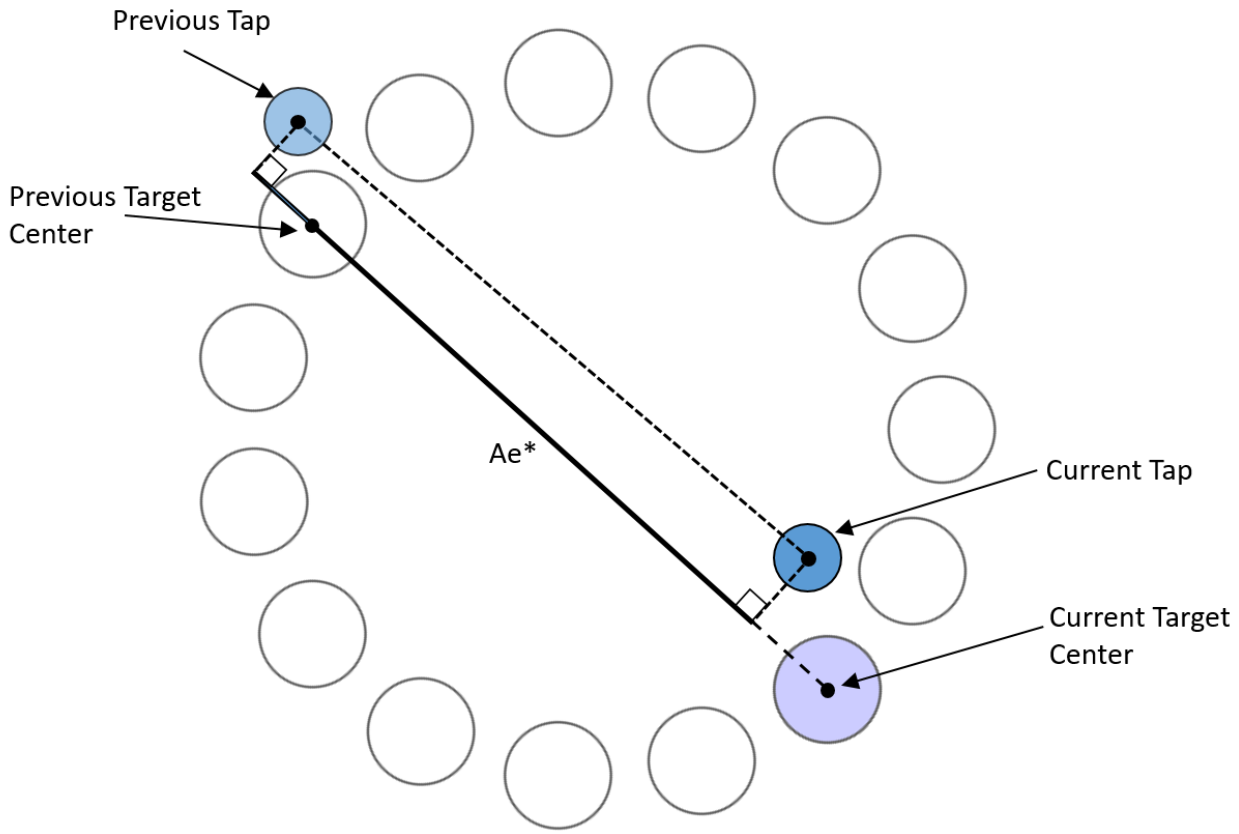


Figure 2.25 – The corrected Ae value, named Ae*, is shown as a solid black line. It is the distance between the two taps, projected onto the line between the two target centers.

To calculate Ae*, MacKenzie recommends adding the previous dx value to Ae [23, 30]:

$$Ae_i^* = Ae_i + dx_{i-1}$$

The throughput calculation was done in several steps, with each step based on the recommendations of Soukoreff and MacKenzie [18], MacKenzie [23] and the GoFitts documentation [30, 47]:

1) Outlier taps that didn't match the assumptions of Fitts' law were removed from the data, when calculating throughput. Soukoreff and MacKenzie explain that Fitts' law only applies for "rapid aimed movements". They explain, in a footnote, that double-clicks and "misfires" violate this assumption, and can be removed [31]. For this study, we have removed double-clicks, as well as the tap immediately following a double-click, when calculating throughput. After doing a double-click, the participant was usually briefly

disorientated, because the target had now moved to a location they didn't expect it to be. There was often a pause, where the participant needed to relocate the target and reorient themselves, before continuing. Since this pause and reorientation process violated the rapid-aimed movement assumption, the double-click and the tap immediately following it were both removed from the throughput calculation. In essence, the tap immediately following the double-click was treated as a re-start point.

To clarify, while outlier taps were discarded when calculating throughput, they were retained when calculating error rate. The logic is that a double-click is still a type of error, and this type of error could still cause serious problems if not addressed. In essence, a double-click does not satisfy the conditions for calculating throughput, but it does still satisfy the conditions for being an error.

2) Using the remaining taps in the given trial (called the "retained taps, for the purposes of this discussion) after the outlier taps have been removed, the standard deviation of dx was calculated. We used Bessel's correction [66] when calculating the standard deviation, since our sample size was relatively small and since using this form of the standard deviation gave throughput values that corresponded to those provided by GoFitts [46]. This means that the formula used to calculate standard deviation of dx per trial was [66]:

$$\sigma_{dx} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(dx_i - \overline{dx})^2}$$

Where "n" is the number of retained taps, per trial (so $15 - number\ of\ discarded\ taps$). $\overline{dx}$ is the average dx of the retained taps, per trial ($\frac{1}{n}\sum_i^n dx_i$). σ_{dx} is the standard deviation of dx, per trial.

Soukoreff and MacKenzie then give the following formula to calculate the corrected target size, We, from the standard deviation of dx [18]:

$$We = 4.133 * \sigma_{dx}$$

3) As mentioned previously, Ae was corrected by adding the previous dx [30]. However, GoFitts does not provide a dx value for the starting target. So the Ae* correction described previously could not be done for the target immediately following the starting target. This

target is denoted as target #1 in Figure 2.21 and Figure 2.22. For target #1, the uncorrected Ae value was used. For all other targets, the corrected Ae value was used. This is described by the following formula:

$$Ae_i^* = \begin{cases} Ae_i, & i = 1 \\ Ae_i + dx_{i-1}, & i > 1 \end{cases}$$

dx_{i-1} can be positive of negative. Ae* can be greater, less than or equal to Ae.

We then took the average of Ae*, across the retained taps per trial:

$$Ae^* = \frac{1}{n}\sum_{i=1}^{n} Ae_i^*$$

Where "n" is once again the number of retained taps, per trial.

4) The We and average Ae* values, as calculated above, are then plugged into the following formula, to calculate IDe, following the Shannon formulation of the IDe calculation [18]:

$$IDe = Log_2(\frac{Ae^*}{We} + 1)$$

This is the IDe value per trial (per 15 targets, minus the removed targets).

5) The throughput per trial was calculated with the following formula:

$$TP_j = \frac{IDe_j}{MT_j}$$

Where j represents the trial.

6) Soukoreff and MacKenzie recommend using the following formula to calculate the mean-of-means throughput, which is an average across all trials and all participants, per condition [18]:

$$TP = \frac{1}{y}\sum_{k=1}^{y}(\frac{1}{x}\sum_{j=1}^{x}\frac{IDe_{kj}}{MT_{kj}})$$

Where "y" is the number of participants (24 in the case of this study), "k" represents each individual participant, "x" represents the number of trials per condition (12, in the case of this study), and j represents each trial.

In addition to the mean-of-means throughput, the throughput per target size and per amplitude were also calculated. For these, steps 1-5 were still followed. But instead of taking the total average across all trials, per condition, the average was instead taken for smaller subgroups of trials. For example, when calculating the throughput per target size, for a given condition, an average throughput was calculated across just those trials that shared the same target size.

2.5.2.3 Target Sizes and Amplitudes Used in This Study

Per experimental condition, there were four targets sizes and three amplitudes, for a total of 12 trials per experimental condition. The four target sizes were consistent across all screens, to within +/- 1 pixel accuracy. The four target sizes that were used in this study are shown in the following graph:

Table 2.10 – Target sizes used for the Fitts' target selection task

Target Sizes in In. (cm.)
0.3 (0.8)
0.4 (1.0)
0.6 (1.5)
0.8 (2.0)

0.3 inches (0.8 cm) was chosen based on discussions with aviation experts about common minimum target sizes currently used on avionic systems, for non-touch use. 0.6 inches (1.5 cm) was chosen because it is the minimum touchscreen target size in the MIL-STD-1472H standard [67].

We made an attempt to keep the three amplitudes as close as possible across all the screens, to be able to more easily compare values between them. The maximum amplitude was set based on the maximum display area of the MFD, since it was a representative avionic touchscreen. Unfortunately, the MCDU screen was much smaller than the MFD, and so required much smaller amplitudes. As a result, while the other three screens used relatively similar amplitudes, the MCDU used amplitudes that were different.

Table 2.11 shows the three amplitude sizes for each screen. Note that the iPad's largest amplitude is slightly smaller then the MFD and large touch monitor. This is because the iPad is slightly smaller than the MFD. This difference was considered negligible for the purposes of the analysis.

Table 2.11 – Amplitude values for each touchscreen.

Screen	Amplitude (qualitative)	Amplitude (inches)	Amplitude (cm)
MFD	Small	2.43	6.17
	Medium	3.64	9.25
	Large	5.26	13.36
iPad	Small	2.41	6.13
	Medium	3.62	9.19
	Large	4.71	11.95
Large Touch Monitor	Small	2.40	6.09
	Medium	3.62	9.20
	Large	5.24	13.31
MCDU	Small	1.98	5.04
	Medium	2.48	6.30
	Large	2.98	7.56

To give a visual to the charts above, Figure 2.26 shows all 12 trials superimposed on top of each other, using the MFD amplitudes and target sizes:

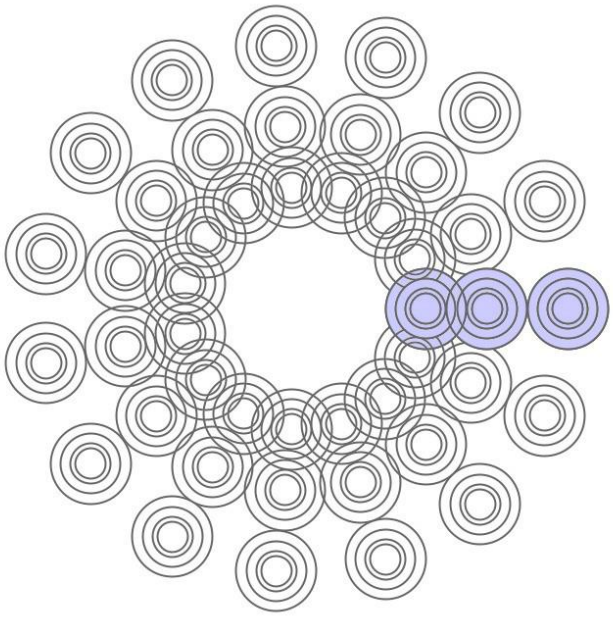


Figure 2.26 – All 12 trials, with each combination of amplitude and target size, superimposed on top of each other.

2.5.2.4 Index of Difficulty Values Tested

In this study, we have followed the recommendations of Soukoreff and MacKenzie and have used IDe (corrected index of difficulty) instead of ID (index of difficulty) [18]. However, it is still useful to note the range of ID values used, per screen.

Soukoreff and MacKenzie recommend testing over a broad range of ID values, between 2 and 8 bits [18]. Unfortunately, on touchscreens, the minimum attainable target size (minimum W) is larger than with a mouse, due to the size of the finger [6]. In addition, the maximum amplitude (maximum A) is limited by the size of the screen. Due to these practical limitations, we had to use a narrower range of ID values, of around 2 to 4 bits.

Table 2.12 shows the different ID values used in this study, per target size and amplitude combination, and per screen. Note that the small MCDU screen has an even narrower range of ID values compared to the other screens, due to its smaller size (and hence smaller maximum amplitude).

Each combination of target size (W) and amplitude (A) gives a different ID value. Since there were four different target sizes, and three different amplitudes tested, per screen, this gives 12 different combinations, and so results in 12 different ID values, per screen.

Table 2.12 – The ID values used for this study, per screen.

Amplitude (large, medium or small amplitude)	Target size (cm)	iPad ID (bits)	Large Touch Monitor ID (bits)	MFD Screen ID (bits)	MCDU Screen ID (bits)
Large	2	2.78	2.92	2.92	2.24
Medium	2	2.47	2.47	2.47	2.04
Small	2	2.01	2.00	2.01	1.80
Large	1.5	3.14	3.28	3.29	2.58
Medium	1.5	2.81	2.81	2.82	2.36
Small	1.5	2.33	2.32	2.33	2.11
Large	1	3.67	3.82	3.82	3.08
Medium	1	3.33	3.33	3.34	2.85
Small	1	2.81	2.81	2.82	2.58
Large	0.8	4.06	4.21	4.21	3.45
Medium	0.8	3.71	3.71	3.72	3.21
Small	0.8	3.18	3.17	3.18	2.93

2.5.2.5 Linear Fit of Movement Time versus IDe

In order to validate the Fitts' law model used, a linear fit of movement time versus IDe is often performed on the data set. Along with the linear fit, the associated linear formula and R^2 values are often given, with the R^2 values used to show how appropriate the model is for the data set [25, 26]. This section describes the methods used to perform this linear fit.

In this study, a linear fit was performed on each of the 18 conditions. As described previously, each of these 18 conditions included 12 trials.

The following steps were used to perform the linear fit, per experimental condition:

1) Per trial, an average IDe and an average movement time (MT) was calculated, across all participants

$$IDe_j = \frac{1}{y}\sum_{k=1}^{y}(IDe_{kj})$$

$$MT_j = \frac{1}{y}\sum_{k=1}^{y}(MT_{kj})$$

Where "y" is the number of participants, "k" represents each individual participant, and j represents each trial.

Since there were 12 trials, this produces 12 IDe_j and MT_j values.

2) Using Microsoft Excel, a scatter plot was made, using the 12 IDe_j and MT_j data points. IDe was plotted on the X-axis, and movement time was plotted on the Y-axis.

3) Microsoft Excel's linear fit tool was used, which provided a linear formula for the scattered data, as well as an R^2 value, per experimental condition. These formulas and R^2 values were then written in a table.

2.5.2.6 Criteria for Discarding Points for Throughput Calculation

Several types of false taps were observed during the study. All of these are being called "double-clicks", even if some of them fall under other categories.

The types of false taps observed where:

1) Accidental double-taps, in rapid succession. This was often done during vibration. The finger accidentally bounced off the screen and hit it again in almost exactly the same spot.

2) Accidental triple-taps, in rapid succession.

3) Confused tap, due to a lack of responsiveness of the screen. A Fitts' selection task, as defined in the ISO standard, is predictable [24, 68]. The sequence of targets is always the same. Over time, the participant gets very used to this pattern, and comes to anticipate the position of the next target (this is a desired part of a Fitts' test, that the task be repetitive). The original Fitts' test involved moving back and forth between two fixed targets [19]. When there was a lack of responsiveness, the screen sometimes did not register a tap, and the target remained in the same location. However, the participant anticipated that the tap had been registered, and was already moving towards the next target in the sequence. The participant then realized that the target did not moved, but became disorientated, and accidentally tapped on the screen anyway, with the hand already close to the next target in

the sequence. This happened most often with the MFD screen, since it was not very responsive and occasionally did not register taps.

4) Accidentally hitting the side of the screen with the hand. This happened most often in the thumb condition, since the participants were holding onto the side of the screen with their finger and reaching their hand over to hit the targets. It was also observed most often on the iPad, since the iPad edge was thin, and the active area of the screen came close to the edge. In essence, with users holding onto the side of the iPad screen for support and reaching across the screen with their hand, there was a chance that they could accidentally tap the side of the screen with their palm. This tap with their palm was sometimes registered as a tap.

Accidental double-taps, in rapid succession (Case 1), can be found by measuring the distance from the tap to the target. The previous target and current target are on opposite ends of the circle. The rapid double-tap occured very close to the previous target, and so is across the circle from the current target. As a result, if the location of a tap is close to a full circle's distance away from the target, it is likely an accidental, rapid double-tap.

To measure this distance, it's easiest to use dx. As shown previously, dx is a representation of the distance between a tap and the target, although it is projected onto a line between the current and previous targets, which can cause some distortion. The following image shows how dx can be used to find accidental, rapid double-taps, as described in Case 1.

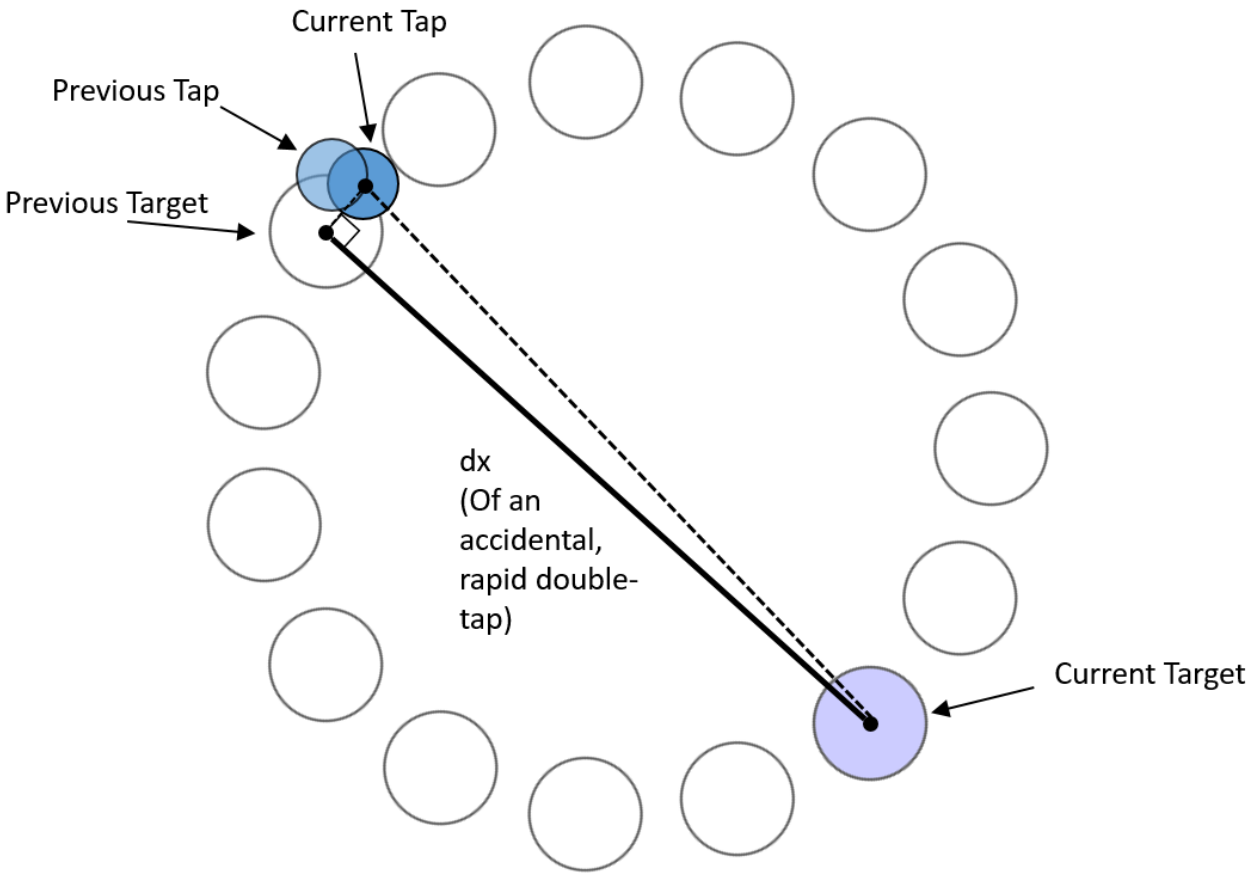


Figure 2.27 – An accidental, rapid double-tap, as described in Case 1. This results in a dx value that is almost equal to the "A" value given by GoFitts.

A confused tap due to a lack of responsiveness (Case 3), can be found in a very similar way. The participant expected the target to be on the opposite side of the circle, and so their hand moves towards that opposite side and taps, even as the circle remained in the same spot. The participant sometimes noticed, midway through, that the target had not moved, but made a tap anyway, out of disorientation. As a result, this tap may not be exactly on the other end of the circle from the target, but could be a large proportion of the way there (the position of the hand when the participant realized that the target hadn't moved). This is shown in the following image.

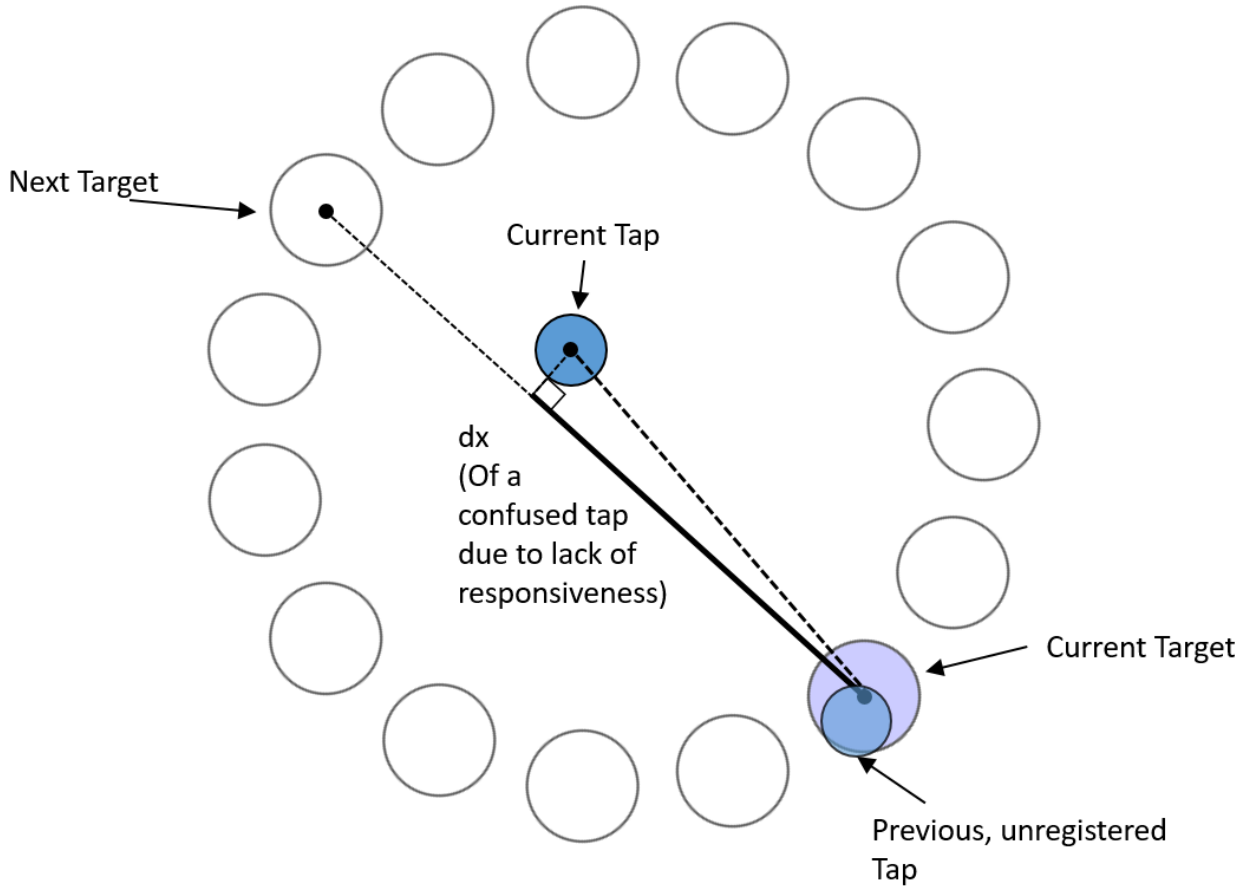


Figure 2.28 – Image showing a confused tap, caused by a lack of screen responsiveness, as described in Case 3.

Since the dx in Case 3 may not exactly equal "A", but may be a large percentage of it, we performed an analysis to determine which percentage to use. The following graph shows the number of double-clicks caught using different percentages of movement distance "A" as a cut-off value for dx. Essentially, the following formula was used to produce the graph:

$$if\,(dx_i > percentage * A)$$

$$Then\ tap_i\ it\ is\ a\ double-click$$

Different values of "percentage" were plugged into this if statement. These percentages are shown on the X-axis of the graph. The number of double-clicks identified, for each percentage value, is shown on the Y-axis.

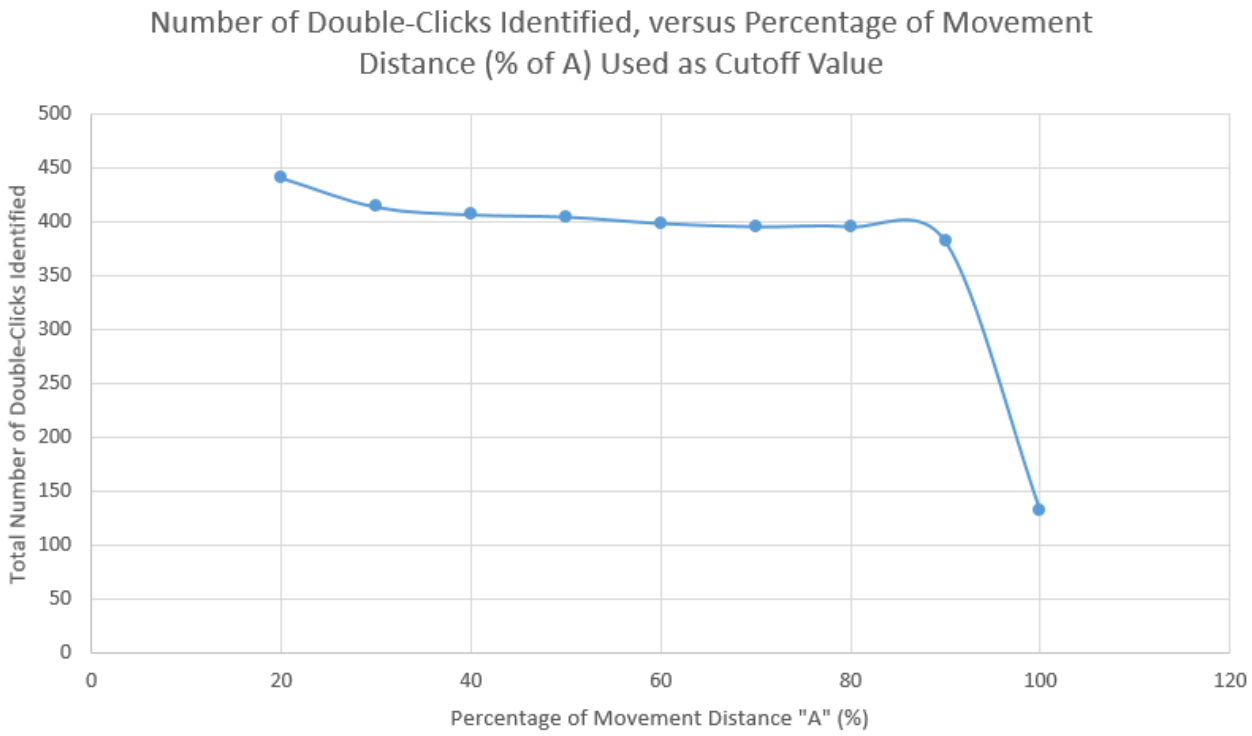


Figure 2.29 - Double-clicks identified versus percentage of movement distance.

We wanted to avoid false positives, while catching most double clicks. We assumed that, when plotting cut-off percentage versus double-clicks, it would follow a normal distribution. We wanted to use the cut-off percentage in the center of the normal distribution (that finds the majority of double-clicks, while avoiding false positives). The slope of a normal distribution is zero at the center. So the slope of the double-click versus cut-off percent graph should be flat, close to the center of the normal distribution. The part of the graph that looks the flattest is from 60-80%. We chose 75%, to be conservative and avoid having false positives.

Accidental, rapid triple-taps, as described in Case 2, were much harder to find, since targets that are two numbers away from each other in the selection sequence are quite close together. This is especially true when using dx as a representation of distance, since the projected distance between the two targets is very close. In essence, dx cannot be reliably used to identify accidental, rapid triple-taps. This is shown in the following image.

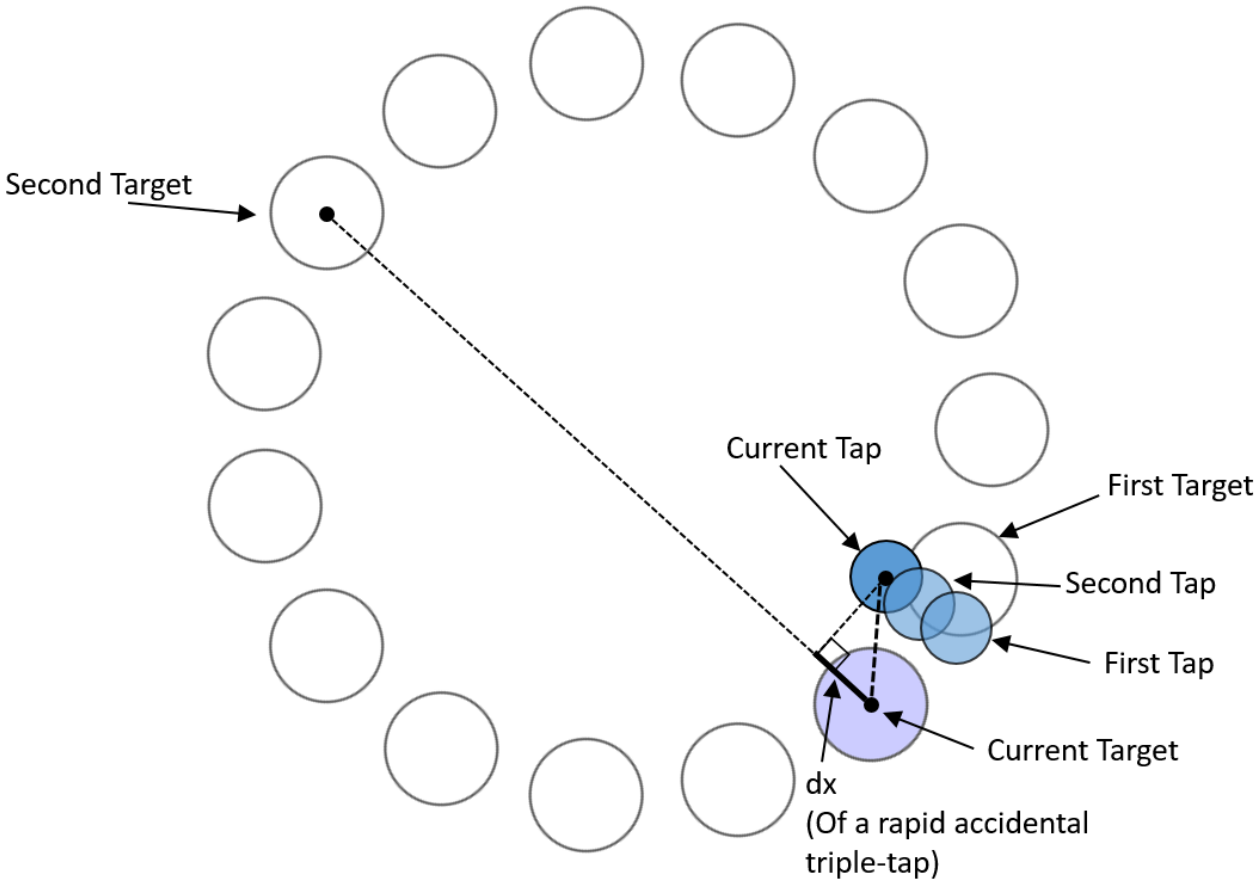


Figure 2.30 – Accidental, rapid triple tap. The third tap is very close to the current target.

The dx value can be used to identify most accidental palm-taps on the edge of the screen, as described in Case 4. Most targets are far from the edge of the screen, and the participant would often tap the edge of the screen while stretching their hand to reach a far target. The dx value was often quite large during these errors. Still, the dx method cannot be used to find all errors of this type, if the target is close to the edge of the screen on which the hand is resting. This scenario, where the palm touches the edge of the screen, while the target is close to that edge, is shown in the following image.

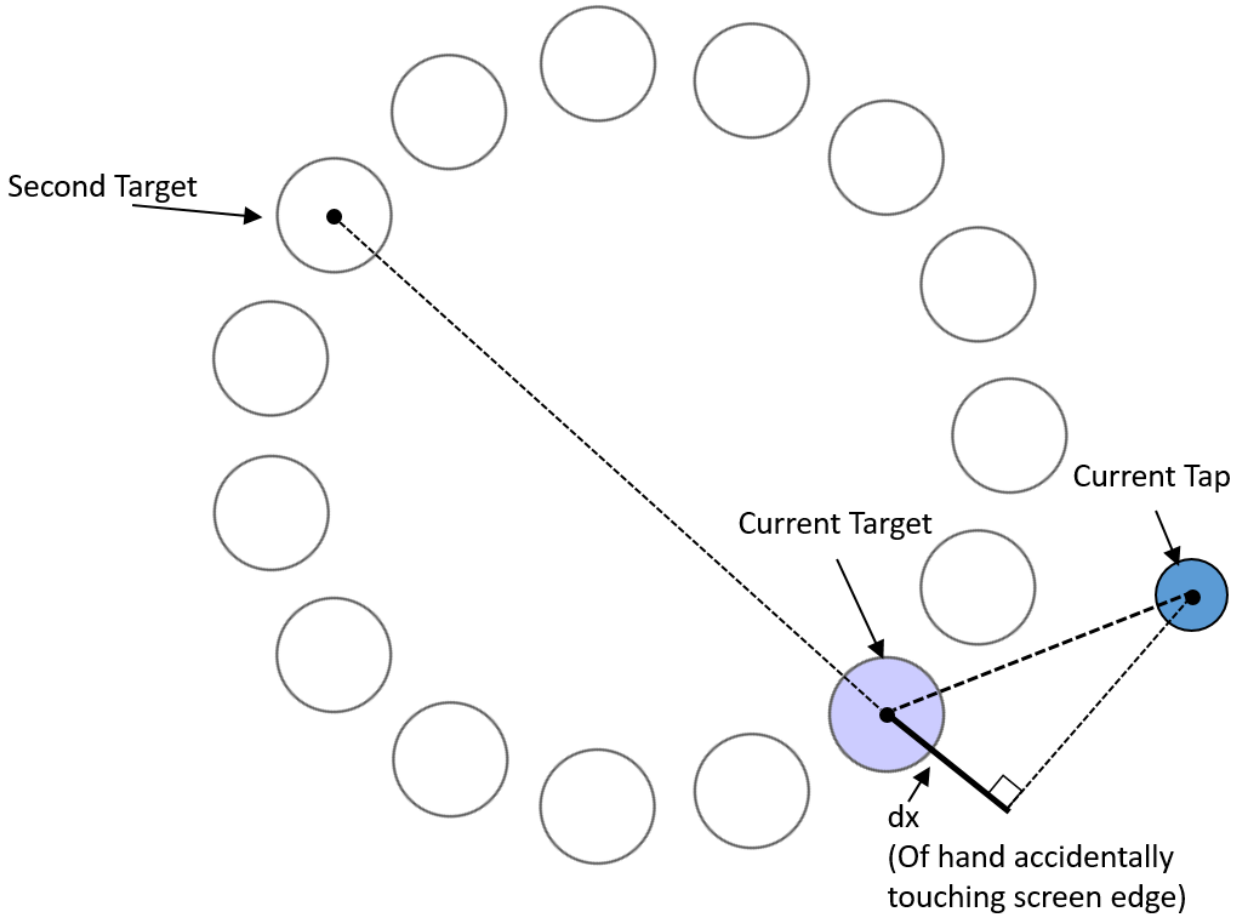


Figure 2.31 – The hand accidentally touches the edge of the screen. This scenario is a subset of Case 4.

In order to find double-clicks (accidental taps) of Case 2 and Case 4, an additional criterion was added to the if statement, to take into account selection time. In essence, if the selection time was much faster than expected, it was likely an accidental tap. A selection time of 175 milliseconds was chosen, since no successful taps were accomplished at this speed (no targets were successfully selected at a speed of 175 milliseconds).

This additional criteria, that any taps executed in 175 milliseconds or less were double-clicks, only resulted in 21 additional clicks being discovered. Since the number of additional accidental taps found was relatively small, it was assumed that this additional criterion was not resulting in too many false negatives.

Overall, the following criteria were used to identify double-clicks. For this study, a double-click is a tap for which:

- the dx value is larger than 75% of movement distance "A":

$$dx > 0.75 * A$$

- the movement time is less than 175 milliseconds:

$$MT < 175\ milliseconds$$

Using these criteria, 417 double-clicks (of all four types) were discovered, across all 24 participants. There were a total of 77,760 taps, across all 24 participants, and 10,468 errors. The double-clicks made up around 4% of all errors, and just 0.5% of all taps. Adding in the taps immediately following the double-clicks, which were also discarded, a total of 814 taps were discarded (some double-clicks were the last tap in the sequence, and so where not immediately followed by any other tap. As a result, the number of discarded taps was slightly less than double the number of double-clicks). This made up just 1% of all taps, and was a relatively small amount of discarded data-points.

2.5.2.7 Error Measurement

Errors were any taps registered outside of the current target circle. In the case of accidental double-clicks, as discussed previously, these taps may have been inside of the previous target circle (or inside the next target circle, if the screen lacked enough responsiveness to cause confusion).

However, double-clicks, as discussed previously, made up a small percentage of the total errors. Most errors were slight misses of the target, occurring in the region around the target. Most of these error taps occurred in the dead zone, outside of any of the target circles.

The participants were notified of an error through auditory feedback. No explicit visual feedback was provided for an error, but an error sound-effect was played. In the case of GoFitts, the error sound was synchronized with the error, and played through a set of speakers. In the case of the iPad, due to slower response rate from the browser, the error sound was slightly delayed from the error, especially when the participant made many errors back-to-back.

GoFitts allows an error threshold to be set, per trial, to avoid having outlier trials with very high error rates [47]. By default, GoFitts sets this threshold at 50%. We used this default 50% error threshold per trial, for this study. This meant that, if a participant made more than 7 errors, out of a total of 15 taps per trial, they were automatically asked to redo the trial over again (7/15 targets missed means that the maximum allowable error rate per trial was 46.7%). The 50% error threshold was used, in this study, because it encouraged participants to balance speed and accuracy. Without any error threshold, participants may have had a tendency to over-prioritize speed, since there would have been no penalty for a lack of accuracy. However, using a lower error threshold may have caused the participants to over-prioritize accuracy, especially under vibration and with small target sizes, where there is a natural tendency towards more errors. 50% was considered a good middle ground. However, there were a few participants, with small target sizes under vibration, who had some difficulty successfully completing certain trials, due to this error threshold, and needed to redo certain trials more than once.

Conversely, the iPad application did not have the ability to set an error threshold. As a result, for the iPad, all error rates were allowed, per trial. However, if the total error rate seemed particularly high, and seemed like an outlier, participants were asked to redo that condition.

2.5.3 Participants

24 participants took part in this study. Note that this study measured human performance for target selections and no prior knowledge of aircraft systems was required for the task, nor would such knowledge offer performance advantage. As a result, participants were not required to be pilots.

The participants' average age was 38 (minimum 23 to maximum 64 years of age). Their average height was 173.6 cm (minimum 157.5 cm to maximum 183 cm), with an average sitting eye height of 120.1 cm (minimum 110.5 cm to maximum 128.3 cm). We measured the participants' height and sitting eye height using markers on a wall, in order to keep a 2 meter distance from the participants at all times (refer to Appendix A). Note that to respect aviation requirements for pilot statures, participants' height had to be within 5'2'' and 6'3'' (14 CFR 25.777 [69]).

The participants had an average index finger width of 1.6 cm (minimum 1.3 cm to maximum 2 cm). They had a mean hand size of 18.9 cm (minimum 17 cm to maximum 20.5 cm).

Index finger width was measured using a 3D printed block with holes of different sizes (see Figure 2.32). The sizes went from 11 to 22 mm, augmenting by 1 mm at a time. Participants were asked to find the hole in which their index finger fit snuggly, without being able to pass fully through. The participants were asked to measure their hand size, from the base of their palm to the tip of their middle finger. This was done using a paper with lines printed on it at 1 cm increments, from 12 to 22 cm.

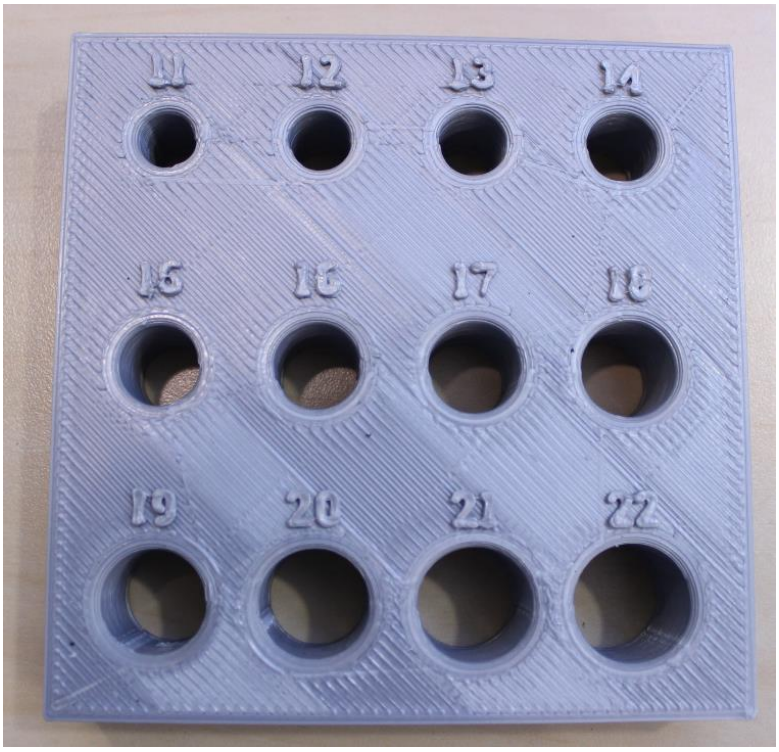


Figure 2.32 – The 3D printed block used to measure participants' index finger widths.

20 of the participants were right-handed, 2 were left-handed and 2 were ambidextrous. One of the ambidextrous participants always used their right hand, while the other alternated between using both hands for the MIP positioned screens. For both ambidextrous participants, the pedestal screen was placed to the right-hand side, and they used their right hands to interact with it.

All participants signed the informed consent form (CER-1920-48-D, see appendix F).

2.5.4 Statistical Analysis

To validate the results, various within-subjects, factorial analysis of variance (ANOVA) tests were used, as well as various pairwise comparison tests. The ANOVA, pairwise comparison tests and

assumption checks used for this study followed the recommendations given in “Designing, Running and Analyzing Experiments”, by Scott Klemmer and Jacob Wobbrock [70].

2.5.4.1 ANOVA

There were several complicating factors, when performing the ANOVA tests. The ANOVA tests used require a full-factorial data set. In this experiment, there were six independent factors:

1. Vibration level (no-vibration or with-vibration)
2. Finger/support method used (index finger or thumb)
3. Screen used (iPad, small MCDU screen, medium MFD screen, or large touch monitor)
4. Screen position (MIP or pedestal)
5. Target size: 0.8 inches (2 cm), 0.6 inches (1.5 cm), 0.4 inches (1 cm), and 0.3 inches (0.8 cm)
6. Amplitude (distance between targets). This amplitude varied slightly between some of the screens, since the screens were different sizes.

The test was not fully factorial for several reasons:

- **Only the iPad was tested in both screen positions (all other screens were solely tested in the MIP position).**

  To address this issue, the iPad in MIP position and the iPad in pedestal position were considered as two separate screens for the analysis. In essence, for the purposes of the ANOVA, there was no separate “Screen Position” factor. Screen position was instead combined into the “Screen” factor, with the “Screen” factor including the following: iPad in MIP position, iPad in pedestal position, small MCDU screen (in MIP position), medium MFD screen (in MIP position), and large touch monitor (in MIP position).

  The iPad in MIP position and iPad in pedestal position were considered separate screens both for the purposes of the ANOVA, as well as for any graphs that show averages across all screens.

- **The large touch monitor was only tested in the index finger (no-support) conditions. It was too large to permit the thumb (with-support) conditions.**

  To address this issue, each ANOVA analysis was split in two: one ANOVA excluded the large touch monitor but included the thumb conditions, while the other ANOVA included the large touch monitor but excluded the thumb conditions. This technique was also used for graphs showing average conditions across all screens. For many of the graphs shown in the results section, the large screen has been excluded from the averages shown, in order to be able to include the thumb conditions. In some of the graphs, the thumb conditions have been excluded from the average, in order to include the large screen. If either the thumb or large touch monitor conditions has been excluded from the average, it is explicitly mentioned in the description for each graph.

- **The amplitude values used for the small MCDU screen were different from the other screens.**

  Table 2.13 shows the amplitude values, per screen. The largest amplitude size, for each screen except the large touch monitor, was chosen so that the largest targets (of 2 cm) remained fully onscreen, with a buffer of a few pixels from the screen edge. As can be seen, the amplitudes on the MCDU were smaller than on the other screens, due to its smaller size.

Table 2.13 – The three amplitude conditions that were tested, per screen

Screen	Smallest Amplitude (cm)	Medium Amplitude (cm)	Largest Amplitude (cm)
iPad (in both positions)	6.1	9.2	12
Large Touch Monitor	6.1	9.2	13.3
Medium MFD Screen	6.2	9.2	13.4
Small MCDU Screen	**5.0**	**6.3**	**7.6**

For the purposes of this study, it was considered that the three amplitudes used on the iPad, large touch monitor and medium MFD screen were close enough to each other that they could be averaged together and used together in an ANOVA, when analysing the impact of amplitude. When creating graphs, the largest amplitude has been denoted as the average largest amplitude, between all the screens being average. However, the amplitude values on the small MCDU screen were considered too different from the others, in a way that might skew the averages and impact the ANOVA, when analysing the impact of amplitude.

To address this issue, the small MCDU screen was removed from the ANOVAs that included amplitude (this screen was included in all other ANOVAs). It was also removed from graphs showing the average impact of amplitude, across screens.

Soukoreff and MacKenzie suggest that the "mean of means" throughput is the most accurate and correct way of analysing throughput [18]. This mean-of-means throughput is an average throughput across all target size and amplitude combinations tested. Since the "mean of means" throughput is considered the most "correct" form of throughput, it was desirable to do a top-level ANOVA using it, before going deeper into additional ANOVAs using the average throughput per target size, or per amplitude.

In total, six ANOVAs were conducted, as described in Table 2.14.

Table 2.14 – Due to the constraints described previously, the ANOVA analysis needed to be split up into several separate ANOVAs. This chart shows the six ANOVAs that were conducted, per dependent variable.

ANOVA Test 1: Vibration x Finger x Screen [the dependent variable has been averaged across all target size and amplitude combinations. The large touch monitor conditions have been excluded]		
Vibration	Finger	Screen
No-Vibration	Index	iPad MIP
Vibration	Thumb	iPad Pedestal
		Medium MFD Screen
		Small MCDU Screen

ANOVA Test 2: Vibration x Screen [the dependent variable has been averaged across all target size and amplitude conditions. The thumb conditions have been excluded]		
Vibration	Finger	Screen
No-Vibration	Index	iPad MIP
Vibration		iPad Pedestal
		Medium MFD Screen
		Small MCDU Screen
		Large Touch Monitor

ANOVA Test 3: Vibration x Finger x Screen x Target Size [the dependent variable has been averaged across all amplitude levels. The large touch monitor conditions have been excluded]			
Vibration	Finger	Screen	Target Size
No-Vibration	Index	iPad MIP	0.8 inch (2 cm)
Vibration	Thumb	iPad Pedestal	0.6 inch (1.5 cm)
		Medium MFD Screen	0.4 inch (1 cm)
		Small MCDU Screen	0.3 inch (0.8 cm)

ANOVA Test 4: Vibration x Screen x Target Size [the dependent variable has been averaged across all amplitude conditions. The thumb conditions have been excluded]			
Vibration	Finger	Screen	Target Size
No-Vibration	Index	iPad MIP	0.8 inch (2 cm)
Vibration		iPad Pedestal	0.6 inch (1.5 cm)
		Medium MFD Screen	0.4 inch (1 cm)
		Small MCDU Screen	0.3 inch (0.8 cm)
		Large Touch Monitor	

Table 2.14 - Due to the constraints described previously, the ANOVA analysis needed to be split up into several separate ANOVAs. This chart shows the six ANOVAs that were conducted, per dependent variable (continued and end)

ANOVA Test 5: Vibration x Finger x Screen x Amplitude [the dependent variable has been averaged across all target size levels. The large touch monitor conditions have been excluded]			
Vibration	Finger	Screen	Amplitude (cm)
No-Vibration	Index	iPad MIP	6.1
Vibration	Thumb	iPad Pedestal	9.2
		Medium MFD Screen	12.5 approx.
ANOVA Test 6: Vibration x Screen x Amplitude [the dependent variable has been averaged across all target size conditions. The thumb conditions have been excluded. The small MCDU screen has been excluded.]			
Vibration	Finger	Screen	Amplitude (cm)
No-Vibration	Index	iPad MIP	6.1
Vibration		iPad Pedestal	9.2
		Medium MFD Screen	12.5 approx.
		Large Touch Monitor	

There was some overlap between factors and interactions effects tested in the six ANOVAs. To avoid having duplicates, some of the results were discarded. Table 2.15 shows the rows of results that were retained from each of the six ANOVAs:

Table 2.15 – This table shows the rows of each ANOVA test which were retained, when creating the final ANOVA table, for each dependent variable

ANOVA Test 1: Vibration x Finger x Screen [the dependent variable has been averaged across all target size and amplitude combinations. The large touch monitor conditions have been excluded]
Vibration
Finger
Screen
Vibration x Finger
Vibration x Screen
Finger x Screen
Vibration x Finger x Screen
ANOVA Test 2: Vibration x Screen [the dependent variable has been averaged across all target size and amplitude conditions. The thumb conditions have been excluded]
Screen
Vibration x Screen
ANOVA Test 3: Vibration x Finger x Screen x Target Size [the dependent variable has been averaged across all amplitude levels. The large touch monitor conditions have been excluded]
Target Size
Vibration x Target Size
Finger x Target Size
Screen x Target Size
Vibration x Finger x Target Size
Vibration x Screen x Target Size
Finger x Screen x Target Size
Vibration x Finger x Screen x Target Size
ANOVA Test 4: Vibration x Screen x Target Size [the dependent variable has been averaged across all amplitude conditions. The thumb conditions have been excluded]
Screen x Target Size
Vibration x Screen x Target Size
Finger x Screen x Target Size
Vibration x Finger x Screen x Target Size

Table 2.15 This table shows the rows of each ANOVA test which were retained, when creating the final ANOVA table, for each dependent variable (continued and end)

ANOVA Test 5: Vibration x Finger x Screen x Amplitude [the dependent variable has been averaged across all target size levels. The large touch monitor conditions have been excluded]
Amplitude
Vibration x Amplitude
Finger x Amplitude
Screen x Amplitude
Vibration x Finger x Amplitude
Vibration x Screen x Amplitude
Finger x Screen x Amplitude
Vibration x Finger x Screen x Amplitude
ANOVA Test 6: Vibration x Screen x Amplitude [the dependent variable has been averaged across all target size conditions. The thumb conditions have been excluded. The small MCDU screen has been excluded.]
Screen x Amplitude
Vibration x Screen x Amplitude
Finger x Screen x Amplitude
Vibration x Finger x Screen x Amplitude

As seen in the Table 2.15 above, ANOVA Test 1, which uses the mean-of-means throughput or error and includes both finger conditions (but excludes the large touch monitor), is used as the basis for the table, and is retained in its entirety. For the other ANOVA tests, just the relevant rows have been retained, those which add new information that was not included in ANOVA test 1. For those tests that include the large screen, but exclude the thumb conditions, just the rows that include "Screen" and its interaction effects have been retained, while all other rows have been discarded.

For each ANOVA, a parametric and non-parametric version of the ANOVA test was conducted. For the parametric ANOVA, the ezANOVA function in R was used, which is part of the ez library [71]. For the non-parametric ANOVA, the art function was used, which is part of the ARTool library, and which uses the aligned rank transform method [72, 73]. In both cases, a within-subjects, factorial ANOVA was performed.

Each row of the ANOVA (each independent variable effect and interaction effect) was tested for normality and homoscedasticity. Normality was checked on the residuals of each ANOVA row, using Shapiro tests. The homoscedasticity of each row was checked using Brown-Forsythe tests.

If either the Shapiro test or Brown-Forsythe test failed, on a given row of the ANOVA, the non-parametric ANOVA result was used, for that specific row. If the row passed the Shapiro and Brown-Forsythe tests, then the parametric ANOVA result was used, for that specific row. If the parametric result was used, an additional Mauchly's test of sphericity was conducted. If the row failed the Mauchly's test, the Greenhouse-Geisser corrected result was used, for that specific row.

In the ANOVA results tables, in the results section of this study, a column has been added to specify which method was used: non-parametric, parametric, or parametric with Greenhouse-Geisser correction. An additional column has been added to specify any ANOVA assumption tests that failed: Shapiro, Brown-Forsythe or Mauchly's tests.

2.5.4.2 Pairwise Comparison Tests

An effort was made to reduce the number of pairwise comparison tests conducted, in order to reduce the number of false positive and false negatives (since a Holm-Bonferroni correction was used). Towards this end, pairwise comparisons were only conducted if the ANOVA first showed that the associated factor and interaction effect were significant.

In addition, pairwise comparisons were only conducted if the comparison lead to further insight into a result or aided in interpreting a graph. As a result, some cross-comparisons were avoided, since the result would not have provided further insight.

For pairwise comparisons done across screens:

- If the comparison was between the large touch monitor and any other screen, then the thumb conditions were excluded from the averages for that other screen.
- If the comparison was between any other two screens, then the thumb conditions were included in both screen averages.

Three groups of pairwise comparisons were conducted, per dependent variable (per throughput and per error rate). A separate Holm-Bonferroni correction was applied to each group.

These three groups are as follows:

Table 2.16 – Three groups of pairwise comparison p-value results. A Holm-Bonferroni correction was applied to each group.

Group Number	Method of Averaging the Dependent Variable, per Condition and per Participant
1	Mean-of-Means
2	Average per W
3	Average per A

A parametric and non-parametric version of each of these three groups was made, giving a total of six pairwise results tables. For the parametric tests, t-tests were used. For the non-parametric tests, Wilcoxon signed-rank tests were used. For each pairwise comparison, a Shapiro test was conducted on the differences between the two conditions. If the Shapiro test failed, the p-value from the non-parametric, Wilcoxon signed-rank test table was used. Otherwise, the p-value from the parametric, t-test table was used.

In order to save space and ensure that the tables are clearly readable, it is not specified in this document whether a parametric or non-parametric test was used to obtain the pairwise p-values presented.

2.5.5 Experimental Procedure

Data collection occurred from April 30 to August 19, 2021.

On arrival, the participant was greeted and instructed on the main activities for the study and asked if he or she had any questions. Then the participant signed the informed consent form.

The participant's anthropometric characteristics were noted, as described previously. The height of the screens was adjusted based on the sitting eye height of the participant. Refer to Appendix A for more details on how the screen height was adjusted. Then the participant was asked to sit on the D-Box GP Pro 500 chair.

The person running the experiment explained to the participant about Fitts' law and the selection task, and answered any of the participant's questions. It was explained that Fitts' law states that there is a trade-off between speed and accuracy, and that we were measuring a value that combines

both speed and accuracy. It was also explained that the order in which targets appear in a Fitts law task is predictable. The participant was instructed to select the targets as quickly and as accurately as possible.

Vibration was then played on the motion platform, to give the participant a chance to get used to the vibration used in this study. The participant was told to conduct a practice Fitts' selection task, while being exposed to vibration. For the purposes of this practice task, the participant used both hand support methods (index finger freehand and thumb supported), to get a sense of both techniques.

The participant was asked if they would like to repeat the practice run, with or without vibration. None of the participants wanted to conduct another practice run.

The participant was then asked to perform the multidirectional selection task in each of the 18 conditions, in the sequence that was planned. The participant took a rest between each condition to prevent fatigue and discomfort and resumed the experiment once the participant felt ready. Once all conditions were completed for a given screen, the participant was asked to stand up so that the screen could be replaced with the next one in the sequence. Once the next screen was installed, the participant was asked to sit back down and continue the study by completing the multidirectional selection task on the new screen. The participant was not required to complete a practice run for every new screen.

We used the comfort questionnaire for device interaction from ISO 9241-411 Table C.1 [24] and replicated it on a webpage that was easily accessible from an iPad (refer to Appendix G). The participant filled out the comfort questionnaire on the iPad immediately after having completed each of the following four test conditions: iPad MIP index finger without vibration, iPad MIP index finger with vibration, iPad pedestal index finger without vibration, iPad pedestal index finger with vibration. The comfort questionnaire was not conducted for the other conditions, since it would have been too time consuming and burdensome on the participants. The iPad was chosen since it was used in both positions. The index finger was chosen since it was considered the baseline selection method.

Once all conditions had been completed, a debriefing questionnaire was administered (see Appendix H).

All questionnaires used in this study (including the anthropometric questionnaire, comfort questionnaire and debriefing questionnaire) were administered in either English or French, depending on the preference of the participant.

CHAPTER 3 RESULTS

This chapter presents the results collected during our experiment. It begins by verifying if there is an order effect on the dependant variables and if the data follows Fitts' law. Then, it presents results of interest for each dependent variable: throughput, error rate, and subjective questionnaires. As a reminder, this section will label the test conditions using the code presented in Table 2.4.

3.1 Data Verification

3.1.1 Order of presentation

An ANOVA was performed on the order of the primary independent variables of Vibration x Finger x Screen. It was not performed for the order of the secondary independent variables of target size or amplitude, as their presentation was randomly varied by the software programs used to run the test [47], and are likely to be well balanced as a result.

The results showed no effect of the order of presentation on throughput (Table 3.1) or error rate (Table 3.2). Hence order was successfully balanced.

Table 3.1 – ANOVA on the effect of condition order on the mean-of-means throughput.

Mean-of-Means Throughput - Order Effects - Excludes Large Screen						
	DFn	DFd	F	p	Method Used	Reason for Method Used
VibrationOrder	1	345	1.83	p>.05	Non-Parametric	Shapiro test failure
FingerOrder	1	345	3.15	p>.05	Non-Parametric	Shapiro test failure
ScreenOrder	3	69	0.89	ns	Parametric	
VibrationOrder x FingerOrder	1	23	0.28	ns	Parametric	
VibrationOrder x ScreenOrder	3	69	1.47	p>.05	Parametric	
FingerOrder x ScreenOrder	3	69	0.15	ns	Parametric	
VibrationOrder x FingerOrder x ScreenOrder	2.31	53.18	2.10	p>.05	Parametric, Greenhouse-Geisser	Mauchly's test failure
Mean-of-Means Throughput - Order Effects – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
ScreenOrder	4	92	0.36	ns	Parametric	
VibrationOrder x ScreenOrder	4	207	0.09	ns	Non-Parametric	Shapiro test failure

Table 3.2 - ANOVA on the effect of condition order on the mean-of-means error rate.

Mean-of-Means Error Rate - Order Effects - Excludes Large Screen						
	DFn	DFd	F	p	Method Used	Reason for Method Used
VibrationOrder	1	345	0.02	n.s	Non-Parametric	Shapiro test failure
FingerOrder	1	23	0.26	n.s	Parametric	
ScreenOrder	3	345	0.74	n.s	Non-Parametric	Brown-Forsythe test failure
VibrationOrder x FingerOrder	1	23	0.59	n.s	Parametric	
VibrationOrder x ScreenOrder	3	69	0.98	n.s	Parametric	
FingerOrder x ScreenOrder	3	69	0.99	n.s	Parametric	
VibrationOrder x FingerOrder x ScreenOrder	3	345	0.09	n.s	Non-Parametric	Shapiro test failure
Mean-of-Means Error Rate - Order Effects – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
ScreenOrder	1	207	0.00	n.s	Non-Parametric	Shapiro test failure
VibrationOrder x ScreenOrder	2.84	65.42	0.39	n.s	Parametric, Greenhouse-Geisser	

3.1.2 Verifying Fit of Fitts' Law to Data Set

The following chart shows the linear fit formulas for movement time versus IDe, per condition tested, compared against movement time versus ID, per condition tested.

The formula for calculating IDe and ID has been discussed previously. For IDe, the effective (corrected) target size and amplitude values are used. For ID, the actual target size and amplitude values are used. MT represents "movement time", or target selection time, in milliseconds. For the purposes of this discussion, movement time as a function of IDe is denoted as MT(IDe), while movement time as a function of ID is denoted as MT(ID).

Table 3.3 – Movement time versus IDe formula, per condition, with associated R^2 value, compared against movement time versus ID formula, per condition, with associated R^2 value. Movement time is in units of milliseconds, while ID and IDe are in bits.

Condition	Formula MT(IDe)	R^2	Formula MT(ID)	R^2
iFU	MT = 163.4*IDe - 40.94	0.734	MT = 104.17*ID + 148.21	0.974
iFV	MT = 195.82*IDe - 28.28	0.435	MT = 141.24*ID + 97.67	0.932
iTU	MT = 194.57*IDe - 82.47	0.876	MT = 121.45*ID + 139.4	0.972
iTV	MT = 220.13*IDe - 72.30	0.698	MT = 145.29*ID + 107	0.972
iFUP	MT = 159.93*IDe - 24.80	0.751	MT = 108.36*ID + 154.28	0.971
iFVP	MT = 163.61*IDe + 19.29	0.509	MT = 124.71*ID + 133.56	0.967
iTUP	MT = 176.53*IDe - 52.25	0.724	MT = 116.14*ID + 142.44	0.973
iTVP	MT = 190.11*IDe - 42.56	0.536	MT = 139.53*ID + 95.59	0.979
LFU	MT = 116.62*IDe + 83.12	0.678	MT = 81.079*ID + 221.25	0.961
LFV	MT = 139.91*IDe + 91.95	0.465	MT = 108.52*ID + 189.47	0.888
MFU	MT = 143.77*IDe + 59.55	0.868	MT = 82.075*ID + 296.71	0.892
MFV	MT = 177.64*IDe + 54.57	0.549	MT = 130.76*ID + 212.51	0.913
MTU	MT = 209.31*IDe - 89.94	0.907	MT = 123.43*ID + 214.98	0.905
MTV	MT = 204.03*IDe - 1.08	0.763	MT = 149.77*ID + 181.67	0.939
SFU	MT = 116.72*IDe + 81.73	0.541	MT = 71.147*ID + 234.45	0.910
SFV	MT = 214.98*IDe - 102.1	0.352	MT = 138.98*ID + 123.75	0.927
STU	MT = 199.57*IDe - 72.21	0.572	MT = 115.41*ID + 218.2	0.959
STV	MT = 152.5*IDe + 101.76	0.337	MT = 116.66*ID + 234.7	0.882

As seen in Table 3.3, the R^2 values were high in nearly all conditions using the MT(ID) formula (> 0.88). When using the MT(IDe) formula, the R^2 values were lower for the small MCDU screen (0.34-0.57), and for the vibration conditions on the other screens (0.43-0.76). The correlation between movement time and IDe was weaker under vibration.

To save space, the 36 graphs associated with Table 3.3 are not shown here. However, for the MT(IDe) graphs, the data points were noticeably noisier (more scattered, in cases with particularly low R^2 values they appeared more like elliptical blobs of points) in the vibration conditions, compared to those in the no-vibration conditions. However, the relationship between movement time and IDe still appeared to be broadly linear, if noisy. This indicates that there likely continues to be is a linear relationship between movement and IDe, even for cases where the R^2 value is low, but it becomes weaker under vibration.

In comparison, the MT(ID) graphs appeared very linear in nearly all conditions, as evidenced by the high R^2 values. This implies that the movement time versus ID relationship may be more robust to vibration conditions, compared to the movement time versus IDe relationship.

In general, Soukoreff and MacKenzie have recommended using the mean-of-means throughput for comparing conditions (as is being done in this study), whereas they have recommended using linear fit for predictive purposes, such as predicting typing speed on a soft-keyboard [18].

The R^2 results imply that using the MT(ID) formula likely provides a better prediction of movement time, when compared to the MT(IDe) formula, especially under vibration conditions.

In this study, we are attempting to compare test conditions using throughput. Since a single throughput value is being used to represent a test condition, its value should ideally be reasonably constant across all trials of that condition.

In the following discussion, TP represents throughput. TP(IDe) represents throughput calculated using IDe, while TP(ID) represents throughput calculated using ID. We can compare TP(IDe) versus TP(ID), to verify which offers a more consistent throughput value across trials, per condition. Towards this end, the following methodology has been used:

$$TP(IDe) = \frac{IDe}{MT}; TP(ID) = \frac{ID}{MT}$$

The standard deviation of each type of throughput was then calculated across the 12 trials (Target Size x Amplitude combinations), per condition. Standard deviation, per condition, was calculated with the following formula, which uses Bessel's correction as discussed previously [66]:

$$\sigma_{TP} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(TP_i - \overline{TP})^2}$$

σ_{TP} represents the standard deviation of throughput for a given test condition. TP_i represents the throughput for a given trial (Target Size x Amplitude combination), averaged across all participants. $\overline{TP}$ represents the mean throughput for a given condition, averaged across all participants (see Table 3.4).

Table 3.4 – Standard deviation values for TP(IDe) versus TP(ID), per condition.

Condition	$\sigma(TP(IDe))$ (bits/sec)	$\sigma(TP(ID))$ (bits/sec)
iFU	0.475	0.480
iFV	0.670	0.370
iTU	0.335	0.389
iTV	0.483	0.297
iFUP	0.442	0.459
iFVP	0.623	0.385
iTUP	0.448	0.420
iTVP	0.659	0.291
LFU	0.500	0.692
LFV	0.647	0.568
MFU	0.240	0.685
MFV	0.546	0.455
MTU	0.250	0.490
MTV	0.380	0.375
SFU	0.455	0.716
SFV	0.697	0.371
STU	0.429	0.446
STV	0.524	0.470
Average, across all conditions	0.489	0.464

Table 3.4 shows that the standard deviation of throughput per condition was relatively similar, when comparing between TP(IDe) and TP(ID). TP(IDe) appeared to be larger in the vibration conditions, and smaller in the no-vibration conditions. When taking the average standard deviation across all conditions, the two methods of calculating throughput yielded similar values: 0.489

bits/sec for TP(IDe) versus 0.464 bits/sec for TP(ID). As a result, both methods of calculating throughput appeared equally appropriate, when comparing between conditions in this study.

For the purposes of this study, we will use IDe and TP(IDe). This is to follow the recommendations of Soukoreff and MacKenzie [18], so that the results reported here can be compared against other studies that use this method.

3.1.3 Validation of Relationship Between Effective Target Size (We) and Error Rate

Soukoreff and MacKenzie [18] mentioned that the effective target size (We) is meant to correct the target size (W) for an error rate of approximately 4%, at least for 1D Fitts' tasks. In essence, We is meant to calculate the "effective" target size used by the participants, rather than the ideal target size presented to the participants, since the participants may choose to make the task either easier or more difficult on themselves by being more or less precise than the task demands [29]. One of the goals of the present study is help establish a minimum touch target size under vibration. The effective target size, We, appears to be a potential tool to aid in this task, if it is representative of the real target size used by the participants. Prior studies have compared target size (W) against effective target size (We), and used this comparison to indicate the relative difference between the distribution of participant tap locations and the target size presented to the participants [25, 74].

However, before we can be confident in using We as a representation of the real target size used by the participants, it is necessary to validate whether We really does correspond to a 4% error rate.

Soukoreff and MacKenzie [18] provide two different formulas for calculating We. One formula uses the standard deviation of 1D touch positions, with distances relative to the current target center, and projected along the line connecting the previous target and the current target. This formula has been discussed previously, and is the one used in this study to calculate IDe and throughput. It is also the We formula favoured by Soukoreff and MacKenzie [18]. The other formula uses the error rate directly, to calculate a correction factor for the target size. Soukoreff and MacKenzie [18] recommend that this error rate version of the We formula only be used when the raw selection point data is unavailable.

To validate whether either of these We formulas correspond to a circle containing 96% of tap points (with 4% of taps being outside the circle), we used the following methods:

1. We looked at the number of trials where We<W, while Error Rate>4%. In essence, if We corrects for a 4% error rate, then we would expect it to increase the target size when the error rate is higher than 4%, and decrease the target size when the error rate is less than 4%.

2. We gathered the X,Y position data of each tap. We aggregated all the tap points for a given trial (a given W x A x Experiment Condition combination), across all participants. Since there were 24 participants and 15 target selections per trial, that gave 24*15=360 tap points per trial. We calculated the circle size, centered at the same location as the target, that encapsulated 346/360=96% of those tap points. This is described mathematically by:

$$R_i = \sqrt{(Tap_{x,i} - Target_{x,i})^2 + (Tap_{y,i} - Target_{y,i})^2}$$

Where "i" represents each tap in a given trial, "R" represents the Euclidian distance between the tap and the target center, Tap_x represents the x-axis coordinate of the tap point, Tap_y represents the y-axis coordinate of the tap point, $Target_x$ represents the x-axis coordinate of the center of the target, and $Target_y$ represents the y-axis coordinate of the center of the target.

For each trial, we sorted all the tap points by their "R" values, from smallest to largest. Since there were 360 total taps per trial, across all participants, the tap with the 346th largest distance was used to calculate the radius of a circle encapsulating 96% of tap points:

$$W_{96\%} = 2 * R_{346}$$

Where $W_{96\%}$ is the diameter of the circle containing 96% of tap points for that trial, and R_{346} is the distance of the 346th furthest tap from the target center, for a given trial (out of 360 total taps per trial).

The $W_{96\%}$ value, per trial, was then compared against the We values per trial, calculated using the two different formulas mentioned by Soukoreff and MacKenzie [18].

3.1.3.1 Standard Deviation Formula for We

The standard deviation formula for calculating We, presented previously, is shown again here for clarity [18]:

$$We_{kj} = 4.133 * \sigma_{dx,kj}$$

Where,

$$\sigma_{dx,kj} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(dx_{i,kj} - \overline{dx}_{kj})^2}$$

"k" represents each participant, while "j" represents each trial. σ_{dx} and We values are thus calculated per trial, per participant.

Note that dx is not the Euclidian distance from the tap to the center of the target, but is rather a 1D projection of that distance, onto a line connecting the current target to the previous target. As a result, this formula does not take into account 2D offsets from the target center, only 1D offsets. Also note that the standard deviation is calculated around $\overline{dx}_{kj}$, which is the mean of dx. As a result, even in a 1D reference plane, We is not centered in the same place as W: We is centered at dx=$\overline{dx}_{kj}$, whereas W is centered at dx=0. Since $\overline{dx}_{kj}$ is calculated per trial and per participant, different trials and participants may have different $\overline{dx}_{kj}$ values (these values can even switch signs, from one participant to another). As a result, We is not necessarily centered at the same location,

across different participants. When taking an average of We across participants and/or trials, this offset between centers is inherently ignored.

In summary, the standard deviation formula of We has the following limitations:

1. It is based on a 1D selection distribution. It does not take into account errors caused or exacerbated by an offset perpendicular to the expected path of motion (the line connecting the previous target to the current target).
2. The center of We is not necessarily at the center of the target. It is thus possible to imagine a scenario where all the taps were close together, resulting in a small We, but were all far from the target, resulting in a high error rate.
3. The center of We is variable between trials and participants, for the same W and amplitude conditions. As a result, it may not be possible to take an average of We across trials and participants, since We is not being measured from a consistent reference coordinate.

Note that the method of calculating IDe and throughput takes into account limitations 2 and 3, by using a ratio of Ae/We. Since Ae is the distance to the effective target center, IDe takes into account any shifts in target center that may occur. IDe uses a corrected target width at a corrected target center location, and is calculated on a per trial, per participant basis. As a result, the validation analysis presented in this section is not to validate whether We, calculated using either formula, is appropriate for a throughput calculation. The validation analysis performed in this section is rather to validate whether We can be used as a tool for recommending a minimum target width, for the purposes of reducing error to an acceptable level.

To highlight this, before comparing We against a 4% error rate, we checked that the formula works, when validated along a 1D line and when centered around $\overline{dx}_{kj}$. We followed the following procedure for this initial validation:

A given tap was considered to be outside the We zone if one of the following statements was true:

$$dx_{i,kj} > \overline{dx_{kj}} + We_{kj}$$

$$dx_{i,kj} < \overline{dx_{kj}} - We_{kj}$$

Where “i” represents the number of each individual tap, within a given trial.

Applying this formula across all taps, and averaging across all trials and all participants in this study provided the following result: An average of 3.1% of taps fell outside of the 1D We zone, across all trials in this study

In summary, when using a corrected center position, for each trial and each participant, and when applied along a 1D line, the standard deviation formula of We works as expected. However, that does not mean that it necessarily corresponds to a trial error rate of 4%, as will be discussed below.

3.1.3.1.1 Trials where Error Rate>4%, but We<W

To check whether We really corresponds to a 4% error rate, we looked at the number of trials where We<W, while Error Rate>4%. In essence, if We corrects for a 4% error rate, then we would expect it to increase the target size when the error rate is higher than 4%, and decrease the target size when the error rate is less than 4%.

5,184 total trials were conducted across all participants. 293 of those trials contained double-clicks, from which data points were removed when calculating We but not when calculating error rate.

This process was described previously. For the purposes of this discussion, we are excluding the 293 trials that contained double-clicks. That leaves 4,891 trials.

3,048 of those trials (or 62%) had error rates above 4%. Of the 3,048 trials with error rates above 4%, 1,125 of them had We values that were smaller than the W value for that trial. In other words, in 37% of trials where Error Rate>4%, We<W.

Table 3.5 shows a breakdown of the number of trials where We<W and Error Rate>4%, for each target size tested in this study. As can be seen, this discrepancy is especially pronounced at larger target sizes.

Table 3.5 – Number of trials where Error Rate was greater than 4% but We was smaller than target size, for each target size. This is compared against the total number of trials where Error Rate is greater than 4%, for each target size.

W (cm)	Error Rate >4%	Error Rate >4% and We<W	
	Number of trials	Number of trials	Percentage
0.8	1,129	149	13%
1	988	322	33%
1.5	589	384	65%
2	342	270	79%

Table 3.6 shows the percentage of Error Rate>4% trials where We<W, for the vibration, no vibration, index finger and thumb conditions. Of note, this discrepancy does not appear to have been caused by vibration. The no-vibration conditions actually had a higher percentage of discrepancies.

Table 3.6 - Number of trials where Error Rate was greater than 4% but We was smaller than target size, for vibration, no vibration, index finger and thumb conditions.

Conditions	Number of Trials where Error Rate >4%	Number of Trials where Error Rate >4% and We<W	Percentage of Trials where Error Rate >4% and We<W
No-Vibration	1,412	600	42%
Vibration	1,636	525	32%
Index Finger (Excluding Large Touch Monitor)	1,375	507	37%
Thumb	1,340	503	38%

Overall, in our study, the standard deviation formula of We did not correspond to a 4% error rate.

3.1.3.1.2 Comparing We against $W_{96\%}$

As mentioned previously, per trial, we calculated the diameter of a circle centered at the same location as the target and encapsulating 96% of tap points, across all participants. We named the diameter of this circle W96%.

There were 216 trials. As a result, it is impractical to show all the We and W96% values per trial.

However, when averaged across all 216 trials, the average We result was 1.2 cm. The average W96% result was 1.6 cm. Hence, We was smaller than W96% by 0.4 cm, on average. The average percent difference between We and W96% across trials was 30.8%.

On a per trial basis (averaged across participants, for the same screen, vibration, hand-support method, target size and amplitude conditions), the different between We and W96% ranged from −0.11 to −1.5 cm (with the negative sign meaning that We was smaller than W96%). The percent difference between We and W96%, per trial, ranged from -10.8% to -73.2%. As a result, in all of trials, the absolute value percent difference was above 10%. In half of all trials, the absolute value percent difference was above 29.6%.

Thus, the standard deviation formula of We does not represent a circle containing 96% of tap points and centered at the target location.

3.1.3.2 Error Rate Formula for We

Soukoreff and MacKenzie [18] provide the following formula for We, using the error rate of a given experimental condition or trial:

$$We = \begin{cases} W * \dfrac{2.066}{z(1 - \dfrac{Error\ Rate}{2})}, & Error\ Rate > 0.0049\% \\ W * 0.5089, & Error\ Rate < 0.0049\% \end{cases}$$

We have applied this formula to the average error rate per trial, across all participants.

Since this formula uses error rate directly in the calculation of We, We was always larger than W if the trial's average error rate was above 4%, and smaller than W if the trial's average error rate was less than 4%.

When averaged across all 216 trials, the average We result, calculated using this error rate formula, was 1.6 cm. The average W96% result was 1.6 cm. The average percent difference between We and W96% across trials was 0.6%. Thus, the difference between We and W96% was less than one pixel, when averaged across trials.

There was more of a difference between values when looking on a per trial basis, using the We error rate formula. The percent difference between We and W96% ranged from −41.3% to 32.2% difference (with a positive value meaning that We was larger than W96%). In 86% of trials, the absolute value percent difference was less than 10%. In 57% of trials, the absolute value percent different was less than 5%. The differences were well balanced around zero, with 47% of trials having We>W96%, and 53% of trials having We<W96%.

In terms of the centimeter differences between We and W96%, the differences ranged from −0.70 cm to 0.94 cm. The absolute value difference was less than 0.1 cm in 68% of trials.

On a per condition basis (per Finger x Vibration x Screen condition), the percent difference between We and W96% ranged from -3.8% to 8.1% (with a positive value meaning that We was larger than W96%). The absolute value percent difference was less than 5% in 89% of conditions, and less than 2% in 50% of conditions.

On a per condition basis, the difference between We and W96% (We minus W96%) ranged from 0.16 cm to -0.08 cm. The absolute value difference was less than 0.1 cm in 89% of conditions.

Overall, We, as calculated with the error rate formula, correlated closely with W96%.

3.1.3.3 Summary of We Formulae Validation

The standard deviation formula to calculate We proposed by Soukoreff and MacKenzie does not correspond well to an error rate of 4%. It also does not correspond well with a circle encapsulating 96% of tap points and centered on the target. As a result, we will avoid using this We formula for the purposes of determining minimum attainable target sizes. However, this does not invalidate its

utility for the purposes of calculating throughput, as discussed previously. As a result, we will continue to use this formula when calculating IDe and throughput, since it is preferred by Soukoreff and MacKenzie [18].

The error rate formula to calculate We proposed by Soukoreff and MacKenzie [18] matches extremely closely with a directly measured circle encapsulating 96% of tap points and centered at the same location as the target. This implies that it is likely a good formula to use, for the purposes of correcting target size to an overall 96% error rate across participants.

Since W96% corresponds well with the error rate formula of We, and since it is the most direct form of corrected target size, we will use W96% as a tool, when recommending minimum target sizes.

3.2 Throughput

Throughput results per condition, per target size and condition, and per amplitude and condition.

3.2.1 Throughput per Condition (Mean-of-Means Throughput)

Average throughput results per condition.

3.2.1.1 Statistical Analysis for Throughput per Condition

Table 3.7 shows the ANOVA results for the impact of each factor and cross-factor on mean-of-means throughputs.

Table 3.8 shows the p-value results for the pairwise throughput comparisons, comparing the difference between specific conditions in terms of their mean-of-means throughput.

Table 3.7 – ANOVA results for whether each factor and cross-factor had a statistically significant impact on the mean-of-means throughput.

Mean-of-Means Throughput x Condition x Participant - Excludes Large Screen						
	DFn	DFd	F	p-value	Method Used	Reason for Method Used
Vibration	1	23	216.61	p<.0001*	Parametric	
Finger	1	23	68.52	p<.0001*	Parametric	
Screen	2.09	48.00	9.15	p<.0005*	Parametric, Greenhouse-Geisser	Mauchly's test failure
Vibration x Finger	1	23	28.47	p<.0001*	Parametric	
Vibration x Screen	3	345	2.33	p>.05	Non-Parametric	Brown-Forsythe test failure
Finger x Screen	3	345	3.64	p<.05*	Non-Parametric	Shapiro test failure
Vibration x Finger x Screen	3	69	7.46	p<.0005*	Parametric	
Mean-of-Means Throughput x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen	2.70	62.21	9.19	p<.0001*	Parametric, Greenhouse-Geisser	Mauchly's test failure
Vibration x Screen	4	207	1.52	p>.05	Non-Parametric	Shapiro test failure

Table 3.8 – p-value results from pairwise comparisons on the mean-of-means throughput. All pairwise comparisons are a result of parametric t-tests.

Mean-of-Means Throughput T-Test P-Values with Holm-Bonferroni Correction				
Screen Pairwise Comparisons				
	iPad Pedestal	Large Screen	Medium Screen	Small Screen
iPad MIP	p<.05*	p<.01*	p>.05	p>.05
iPad Pedestal		p>.05	p<.01*	p>.05
Large Screen			p<.001*	p>.05
Medium Screen				p<.005*

Finger x Vibration Pairwise Comparisons			
	Index x Vibration	Thumb x No Vibration	Thumb x Vibration
Index x No Vibration	p<.0001*	p<.0001*	
Index x Vibration			p<.005*
Thumb x No Vibration			p<.0001*

Screen Position x Vibration Pairwise Comparisons		
	iPad Pedestal x No Vibration	iPad Pedestal x Vibration
iPad MIP x No Vibration	p>.05	
iPad MIP x Vibration		p<.001*

3.2.1.2 Throughput Results per Condition

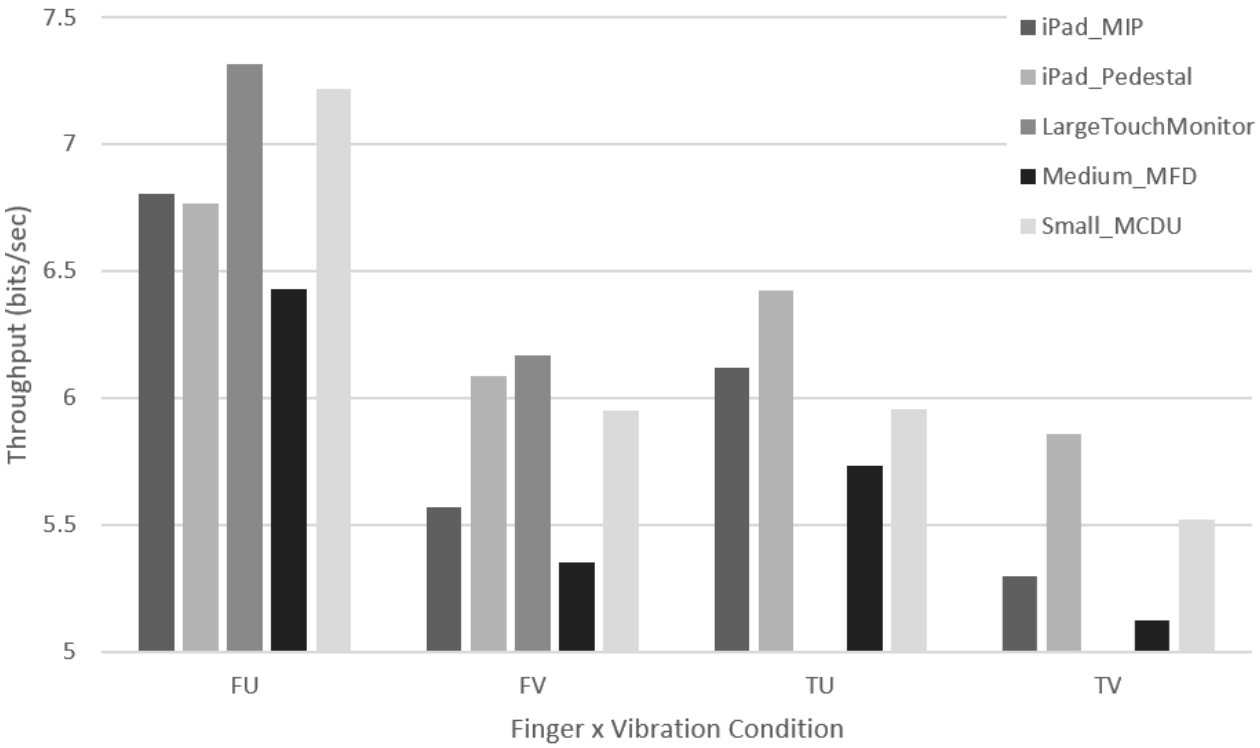


Figure 3.1 - Mean-of-means throughput result for every condition tested: Vibration x Finger x Screen. FU represents the index-no-vibration condition. FV represents the index-vibration condition. TU represents the thumb-no-vibration condition. TV represents the thumb-vibration condition.

Note, in the graph above, that the index-vibration condition had higher throughput than the thumb-vibration condition, across all screens. In addition, the throughput in the index-no-vibration condition is higher than the thumb-no-vibration condition, across all screens.

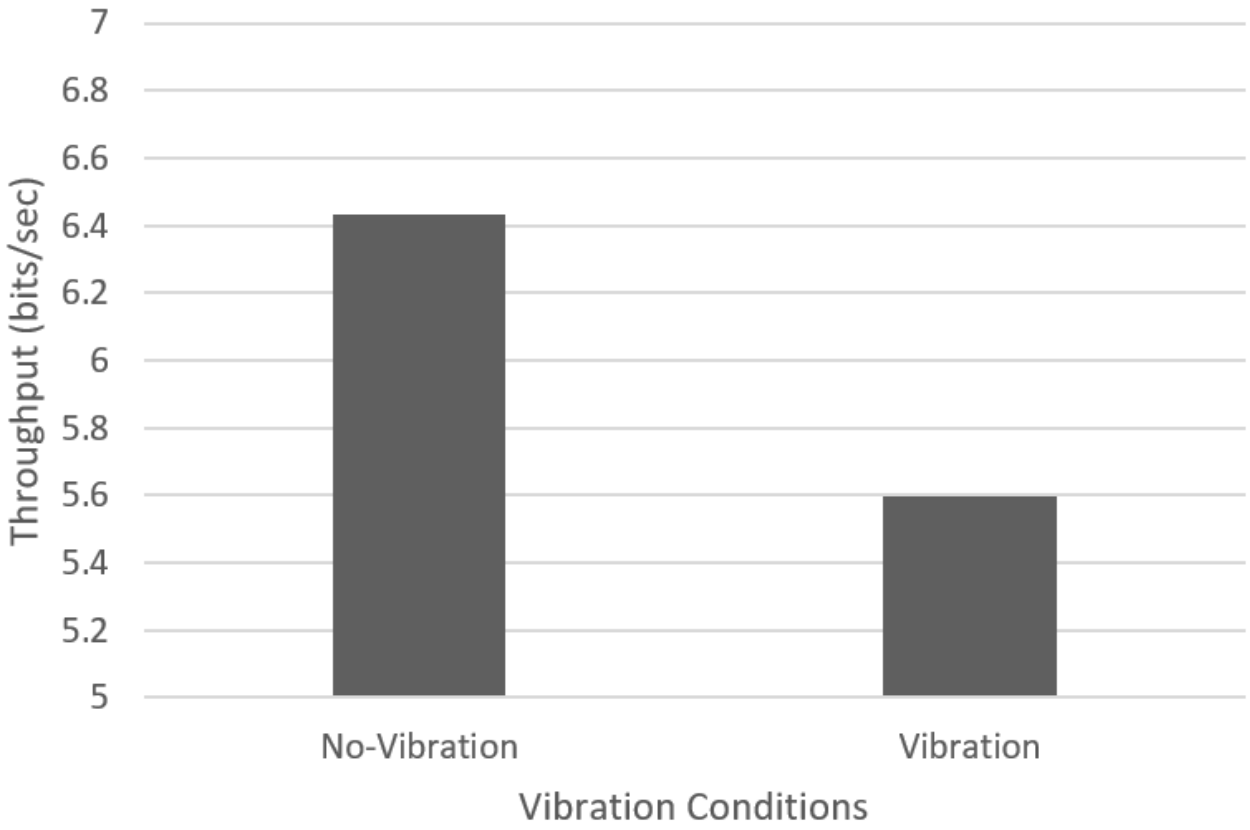


Figure 3.2 - Average throughput in the no-vibration versus vibration conditions. This average is across all screens, excluding the large touch monitor.

As seen in the graph above, vibration has a significant negative impact on throughput, ($F_{1,23}$=216.6, p<.0001).

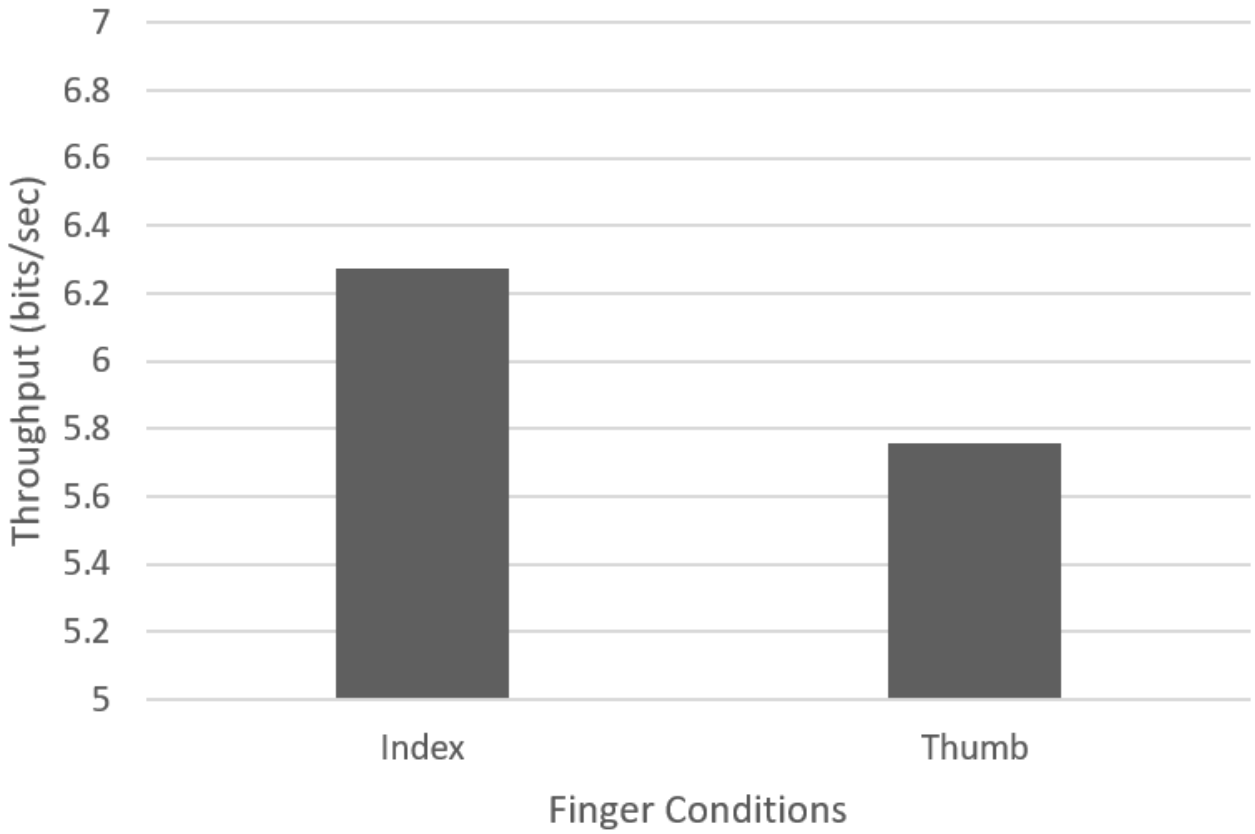


Figure 3.3 - Average throughput in the index (no-support) versus thumb (with-support) conditions. This average is across all screens, excluding the large touch monitor.

As seen in the graph above, using the index finger, without any hand support, resulted in significantly higher throughput than using the thumb while holding onto the side of the screen as a hand support ($F_{1,23}$=68.52, p<.0001).

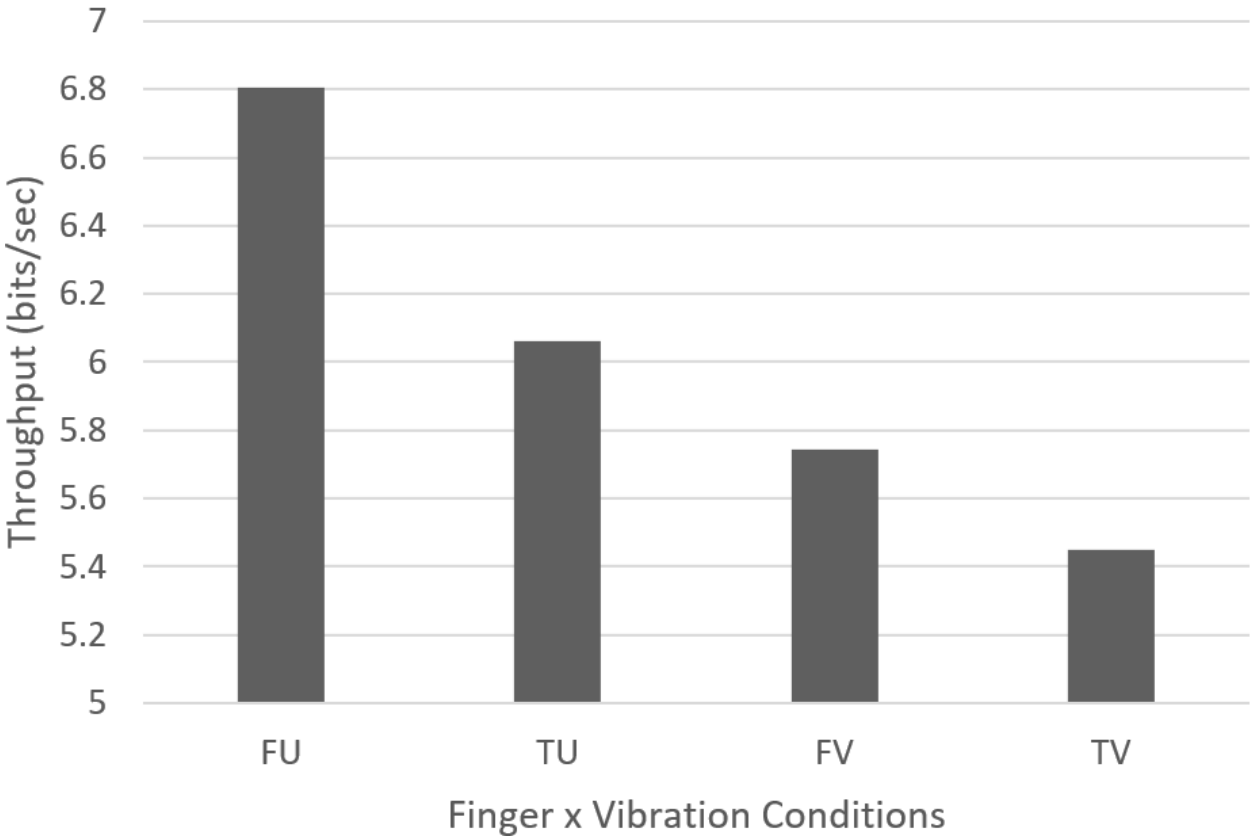


Figure 3.4 – Average Vibration x Finger conditions. This average is across all screens, excluding the large touch monitor.

As can be seen in the above graph, the index finger (without hand support) condition resulted in higher throughput in both no-vibration and vibration conditions, when compared against the thumb (with hand support). The difference was statistically significant, when comparing index-no-vibration against thumb-no-vibration ($p<.0001$). The difference was also statistically significant when comparing index-vibration against thumb-vibration ($p<.005$).

However, note that the difference between the two finger conditions is much larger in the no-vibration condition, compared to the vibration condition. There was a statistically significant interaction effect between Finger x Vibration ($F1,23=28.47$, $p<.0001$). It is possible that, if we had used an even higher vibration level, there may have eventually been a crossing-over point, where the thumb (with support) may have resulted in higher throughput than the index finger (without support). However, the vibration level used in this study was quite high and uncomfortable. In vibration levels much higher than the one used here, the pilot would likely need to focus their

attention on flying, and stop using the touchscreen. This implies that, for most practical scenarios, the index finger (without support), will likely result in a higher throughput, for this type of task.

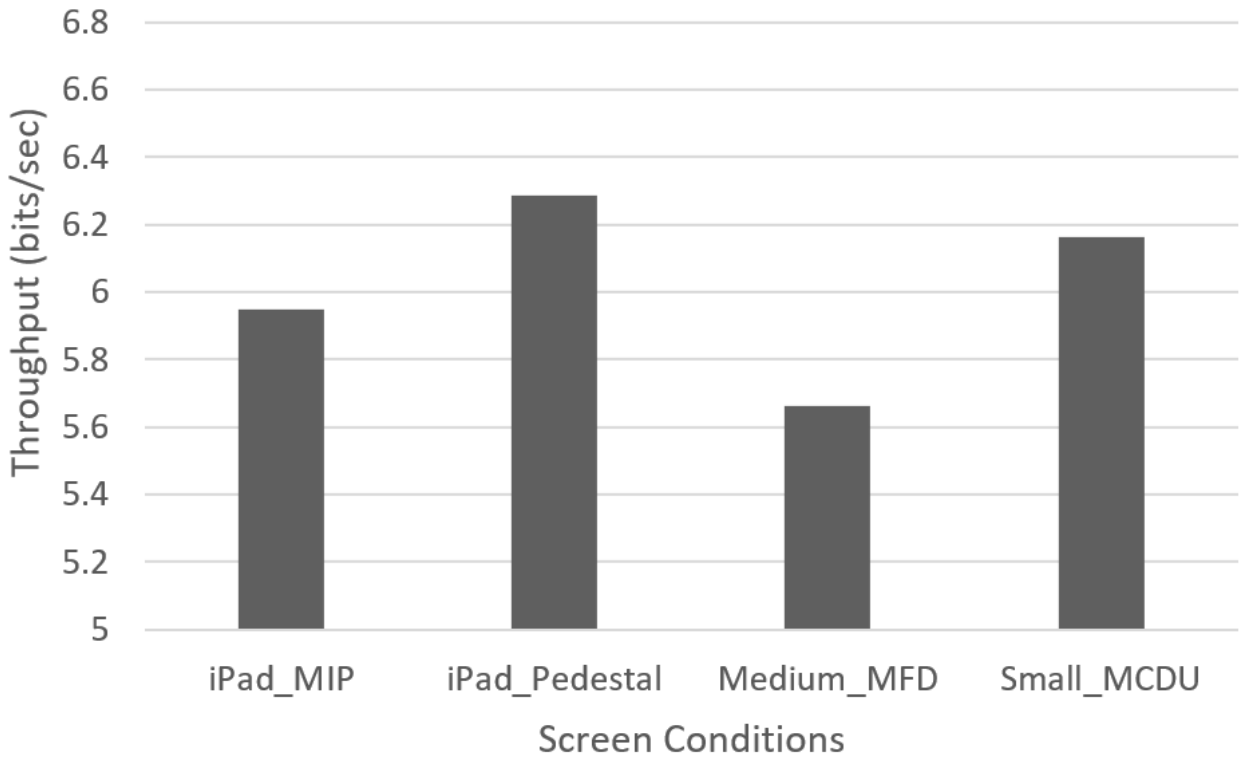


Figure 3.5 – Average throughput per screen, across all four conditions done on each screen. The large touch monitor has been excluded from this graph.

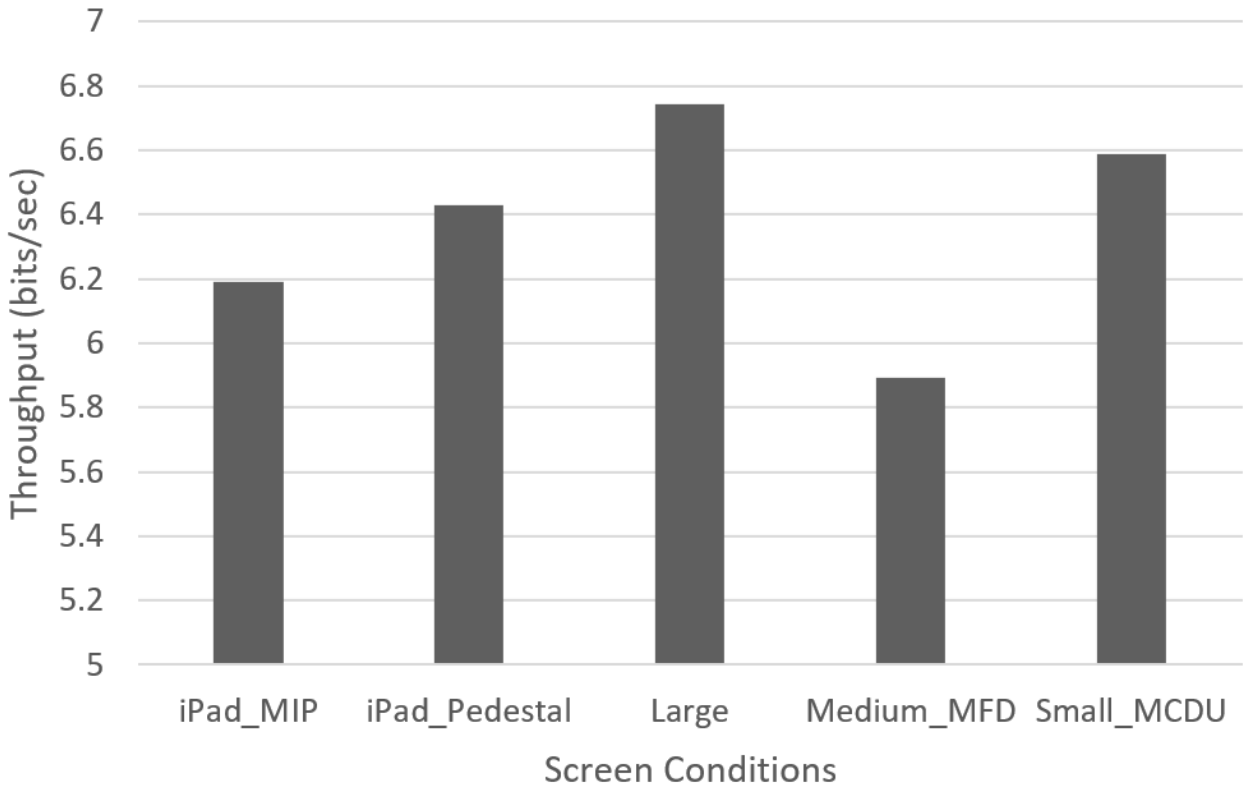


Figure 3.6 – Throughput per screen, using just the index finger conditions for the average. The thumb conditions have been excluded from the averaged for each screen, in order to be able to include the large touch monitor (with is denoted as "Large", in the graph).

The two graphs above show the average throughput per screen. The first graph excludes the large touch monitor, in order to include the thumb conditions in the average. The second graph excludes the thumb conditions from the average, in order to include the large touch monitor.

In both cases, the "Screen" condition had a statistically significant result (without the large touch monitor: $F_{2,1, 48}=9.2$, $p<.0005$. With the large touch monitor, $F_{2.7, 62.2}=9.2$, $p<.0001$).

Table 3.8, above, shows which specific screen differences were statistically significant.

From the pairwise comparisons seen in Figure 3.6, it appears as though the large touch monitor, the small MCDU screen, and the iPad in the pedestal position all had the best average throughput. There were no statistically significant differences between the throughput on these three screens. The medium MFD screen and the iPad in the MIP position had the lowest average throughput. The difference in average throughput between these two screens was not statistically significant.

Of note, the iPad in the pedestal position had higher throughput than the iPad in the MIP position. The following graph compares the throughput in these two positions in more detail.

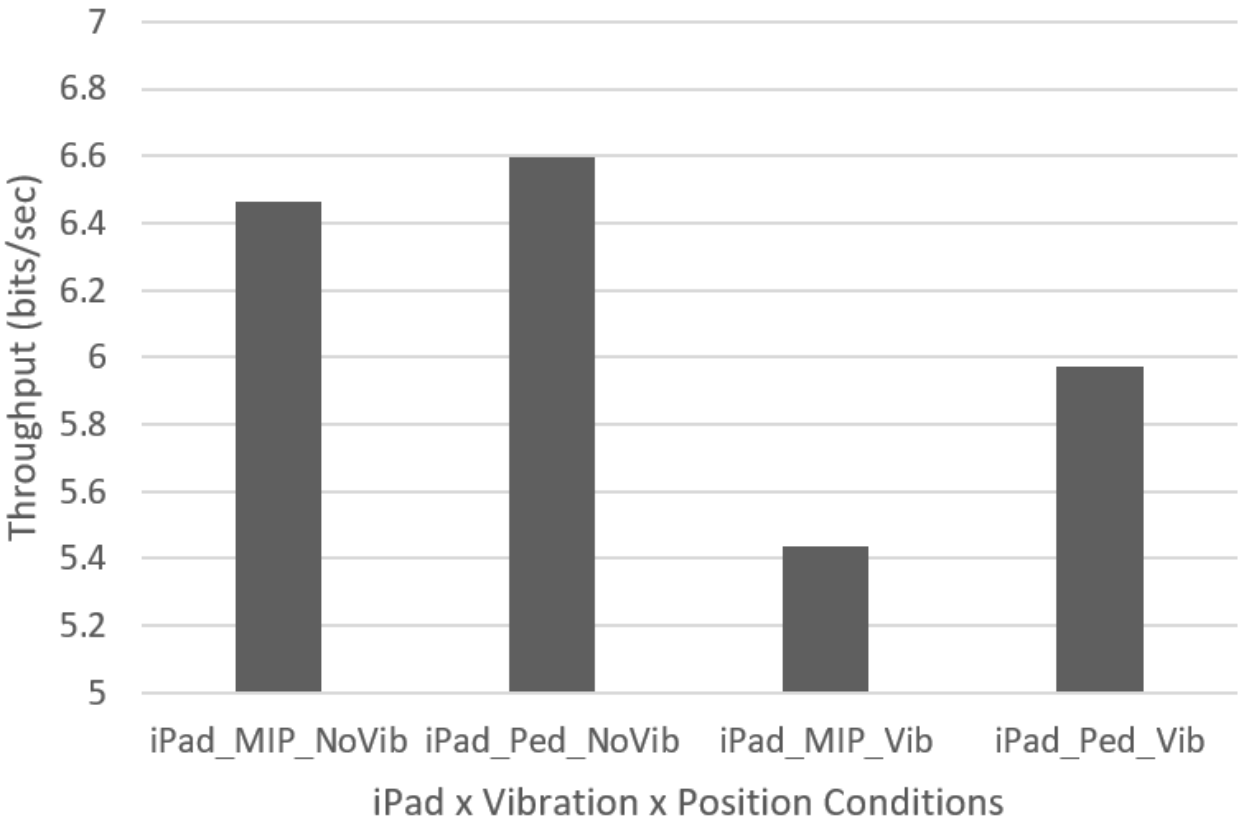


Figure 3.7 - Average throughput on the iPad in both positions, with and without vibration

As seen in the graph above, without vibration, there was no significant difference between the two screen positions ($p>.05$). However, with vibration, the difference became larger and significant ($p<.001$). This implies that any performance differences between two screen positions may be amplified by vibration.

3.2.2 Throughput per Target Size and Condition

Average throughput results per target size and condition.

3.2.2.1 Statistical Analysis for Throughput per Target Size and Condition

Table 3.9 shows the ANOVA results for whether each factor and combined factors crossed with target size had a statistically significant impact on throughput. Table 3.10 shows pairwise

throughput comparisons, comparing specific target sizes against each other, and comparing specific combinations of Condition x Target Size.

Table 3.9 - ANOVA results for whether target size, and each combination of factors crossed with target size, had a statistically significant impact on throughput.

Throughput Averaged per W (Target Size) x Condition x Participant- Excludes Large Screen						
	DFn	DFd	F	P	Method Used	Reason for Method Used
W	1.63	37.45	145.04	p<.0001*	Parametric, Greenhouse-Geisser	Mauchly's test failure
Vibration x W	3	1449	11.95	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Finger x W	2.19	50.44	1.01	p>.05	Parametric, Greenhouse-Geisser	Mauchly's test failure
Screen x W	9	1449	3.01	p<.005*	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x W	3	69	3.20	p<.05*	Parametric	
Vibration x Screen x W	9	1449	0.19	ns	Non-Parametric	Shapiro and Brown-Forsythe test failure
Finger x Screen x W	9	1449	0.62	ns	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Finger x Screen x W	9	207	1.42	p>.05	Parametric	
Throughput Averaged per W (Target Size) x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen x W	12	897	1.45	p>.05	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Screen x W	12	897	0.51	ns	Non-Parametric	Brown-Forsythe test failure

Table 3.10 - p-value results from pairwise comparisons on throughput per target size (W) averages.

Throughput Averaged per W (Target Size) x Condition Being Compared x Participant T-Test or Wilcoxon Signed-Rank Test P-Values with Holm-Bonferroni Correction							
Per W Comparisons- Excluding Large Screen							
	W=1.5	W=1	W=0.8				
W=2	p<.01*	p<.0001*	p<.0001*				
W=1.5	-	p<.0001*	p<.0001*				
W=1		-	p<.0001*				
W=0.8			-				
Per W Comparisons- Just Index (Includes Large Screen)							
	W=1.5	W=1	W=0.8				
W=2	p<.05*	p<.0001*	p<.0001*				
W=1.5	-	p<.0001*	p<.0001*				
W=1		-	p<.0001*				
W=0.8			-				
Vibration x W Comparisons (Vib=Vibration, No-Vib=No-Vibration)							
	Vib x W=1.5	Vib x W=1	Vib x W=0.8	No-Vib x W=2	No-Vib x W=1.5	No-Vib x W=1	No-Vib x W=0.8
Vib x W=2	p<.0001*	p<.0001*	p<.0001*	p<.0001*			
Vib x W=1.5	-	p<.0001*	p<.0001*		p<.0001*		
Vib x W=1		-	p<.0001*			p<.0001*	
Vib x W=0.8			-				p<.0001*
No-Vib x W=2				-	p>.05	p<.0001*	p<.0001*
No-Vib x W=1.5					-	p<.0001*	p<.0001*
No-Vib x W=1						-	p<.0001*

Table 3.10 - p-value results from pairwise comparisons on throughput per target size (W) averages (continued and end)

Screen x W Comparisons (i=iPad MIP, iP=iPad Pedestal, M=Medium Screen, S=Small Screen)												
	iPad Pedestal				Medium MFD Screen				Small MCDU Screen			
	W=2	W=1.5	W=1	W=0.8	W=2	W=1.5	W=1	W=0.8	W=2	W=1.5	W=1	W=0.8
ixW=2	p<.05*				p>.05				p>.05			
ixW=1.5		p<.05*				p>.05				p>.05		
i xW=1			p>.05				p>.05				p>.05	
ixW=0.8				p>.05				p>.05				p>.05
iPxW=2					p<.005*				p>.05			
iPxW=1.5						p<.01*				p>.05		
iPxW=1							p>.05				p>.05	
iPxW=0.8								p>.05				p>.05
MxW=2									p<.005*			
MxW=1.5										p<.0005*		
MxW=1											p>.05	
MxW=0.8												p>.05

3.2.2.2 Throughput Results per Target Size and Condition

There was a statistically significant impact of target size on throughput ($F_{1.6,37.5}$=145.0, p<.0001). Overall, from the following graphs, throughput appears to increase with target size, before reaching a plateau. There was a statistically significant interaction of Vibration x W on throughput ($F_{3,1449}$=12.0, p<.0001).

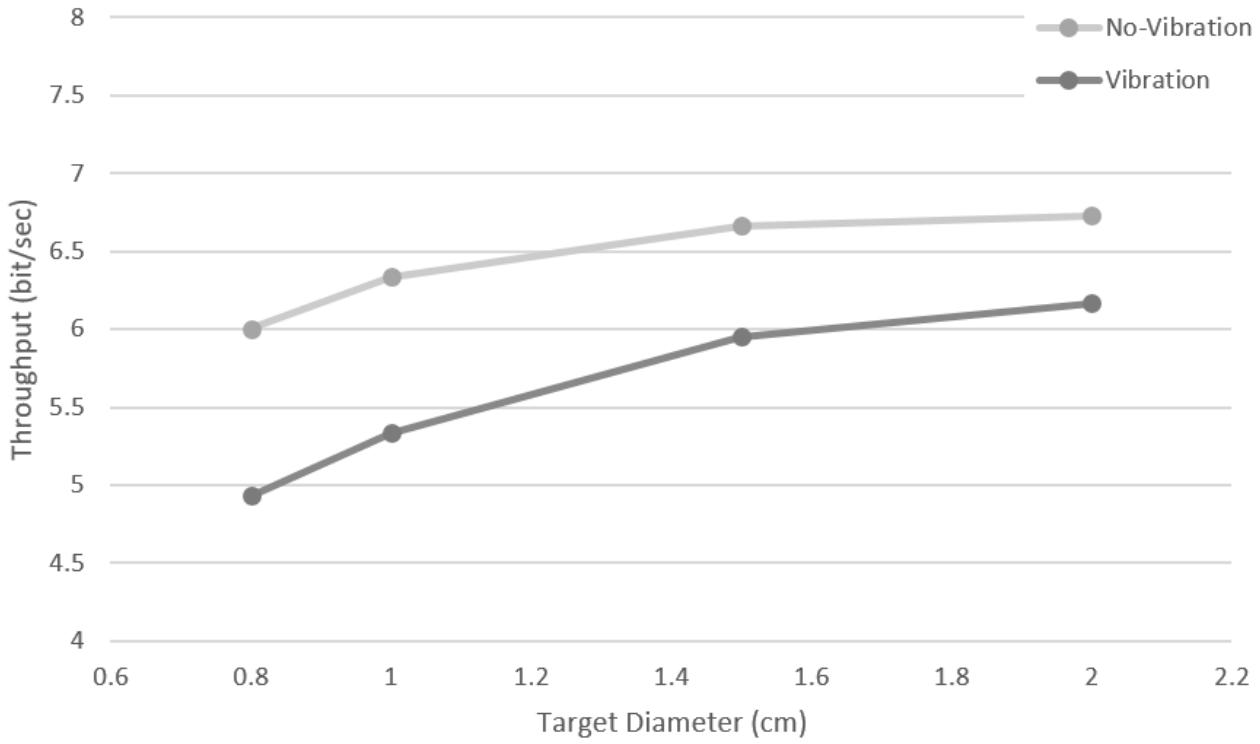


Figure 3.8 – Average results in vibration and non-vibration conditions, plotted on a graph of throughput versus target size. The large touch monitor has been excluded from the average.

Looking at the graph above, it appears as though the throughput in the no-vibration condition flattened out at larger target sizes. In the vibration condition it was slower to flatten out (the slope is slightly steeper at larger target sizes). Overall, the combined effect of Vibration x W appears, visually, to be smaller than the individual effects of vibration and target size, on their own.

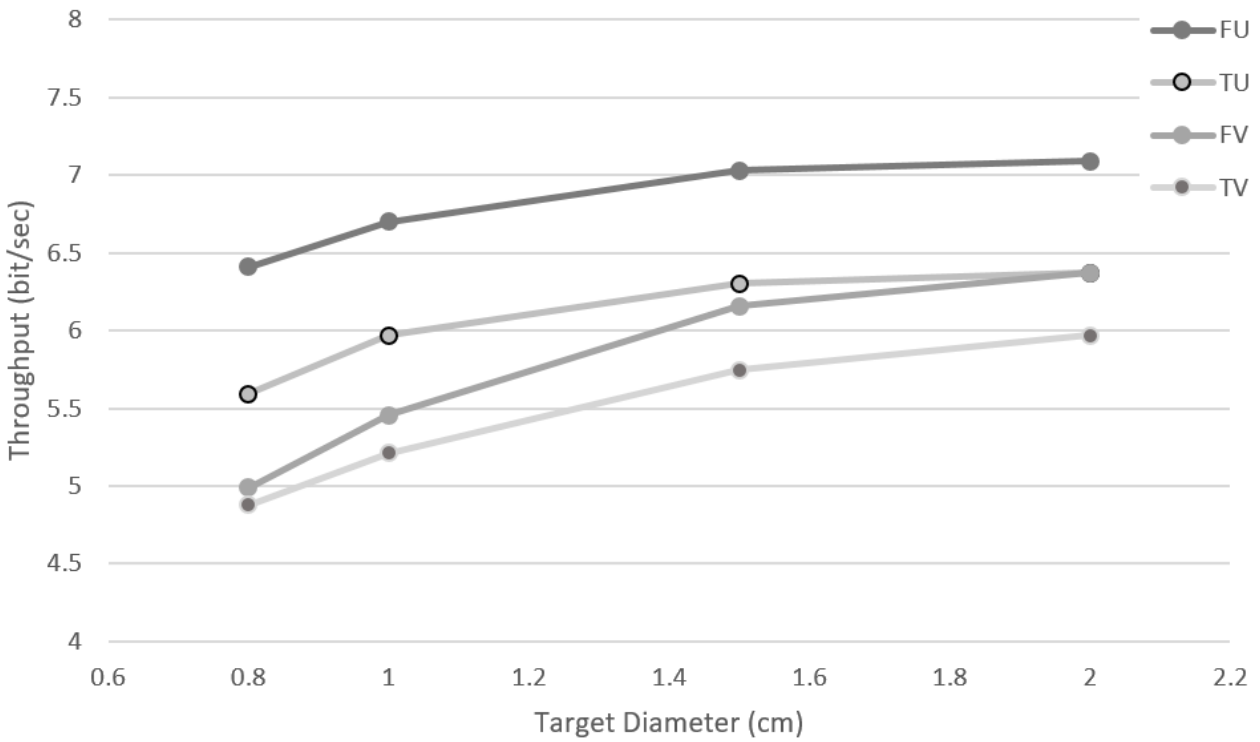


Figure 3.9 - Average results for each Vibration x Finger condition, plotted on a graph of throughput versus target size.

Vibration x Finger had a statistically significant impact on throughput ($F_{3,69}$=3.2, p<.05). Looking at the graph above, it appears that the index-vibration condition was the most impacted by target size, compared to the other conditions. The throughput for the index-vibration condition appears to rise more rapidly, with increasing target sizes.

Surprisingly, the interaction between hand support and target size was not statistically significant ($F_{2.2,50.4}$=1.01, p>.05), as shown in the graph below, despite the previous observation about the index-vibration condition.

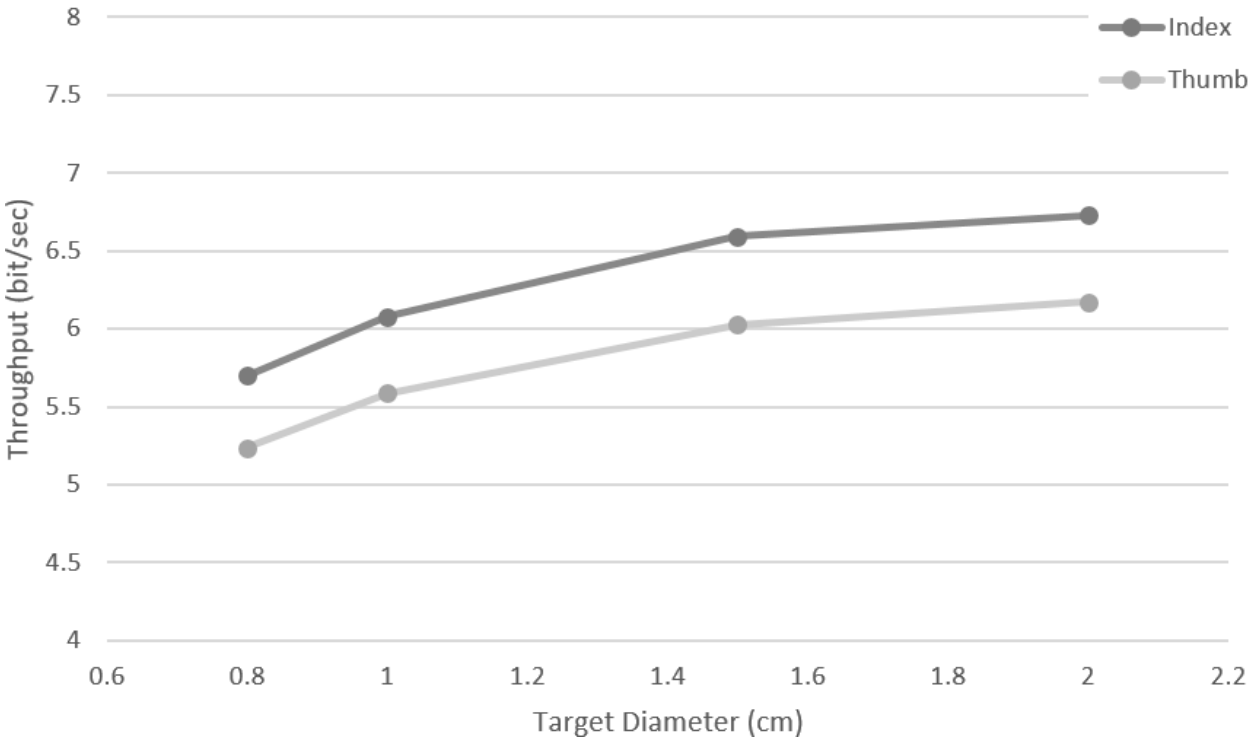


Figure 3.10 - Average results in the index (no hand support) and thumb (with hand support) conditions, plotted on a graph of throughput versus target size. The large touch monitor has been excluded from the average.

3.2.3 Throughput per Amplitude and Condition

Average throughput results per amplitude and condition.

3.2.3.1 Statistical Analysis for Throughput per Amplitude and Condition

Table 3.11Table 3.9 shows the ANOVA results for whether each factor and combined factors crossed with amplitude had a statistically significant impact on throughput. Table 3.12 shows pairwise throughput comparisons, comparing specific amplitudes against each other, and comparing specific combinations of Condition x Amplitude.

Table 3.11- ANOVA results for whether amplitude, and each combination of factors crossed with amplitude, had a statistically significant impact on throughput.

Throughput Averaged per A (Amplitude) x Condition x Participant - Excludes Large and Small Screens						
	DFn	DFd	F	p	Method Used	Reason for Method Used
A	1.45	33.28	26.38	p<.0001*	Parametric, Greenhouse-Geisser	Mauchly's test failure
Vibration x A	2	46	3.89	p<.05*	Parametric	
Finger x A	2	46	8.12	p<.001*	Parametric	
Screen x A	4	805	0.63	ns	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x A	2	46	1.44	p>.05	Parametric	
Vibration x Screen x A	4	92	0.58	ns	Parametric	
Finger x Screen x A	4	92	0.70	ns	Parametric	
Vibration x Finger x Screen x A	4	92	0.28	ns	Parametric	
Throughput Averaged per A (Amplitude) x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen) – Excludes Small Screen						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen x A	6	138	2.51	p<0.5*	Parametric	
Vibration x Screen x A	6	138	0.27	ns	Parametric	

Table 3.12 - p-value results from pairwise comparisons on throughput per Amplitude (A) averages.

Throughput Averaged per A (Amplitude) x Condition Being Compared x Participant T-Test P-Values with Holm-Bonferroni Correction			
Per Amplitude (A) Comparisons – Excluding the Large and Small Screens from the Average			
	A=9 cm	A=6 cm	
A=12.5 cm	p>.05	p<.001*	
A=9cm		p<.0001*	
Per Amplitude (A) Comparisons - Just Index Finger – Excluding the Small Screen			
	A=9 cm	A=6 cm	
A=12.5 cm	p>.05	p<.0001*	
A=9cm		p<.0001*	
Finger x A Comparisons – Excluding the Large and Small Screens			
	Thumb A=12.5 cm	Thumb A=9 cm	Thumb A=6 cm
Index A=12.5 cm	p<.0001*		
Index A=9 cm		p<.005*	
Index A=6 cm			p<.005*

3.2.3.2 Throughput Results per Amplitude and Condition

Many of the participants mentioned that, when using their thumb and supporting their hand on the edge of the screen, it was difficult to reach the furthest targets, in the largest amplitude conditions. As a result, it is of particular interest to see how amplitude might impact throughput when using the thumb, versus index finger. The results from the iPad have been combined with the results from the medium MFD screen, as the screen size and amplitude conditions were close enough[5]. These are the results presented below.

As mentioned previously, the amplitudes on the small MCDU screen were different from on the other screens, due to its smaller size. As a result, the results from the small MCDU screen could

[5] for the purpose of the graphs, the largest amplitude has been assigned a value of 12.5 cm (the average value between the iPad, in both positions, and the medium MFD screen).

not be combined with the results from the other screens, to form averages. Thus, in the graphs shown below, the small MCDU screen results have been excluded.

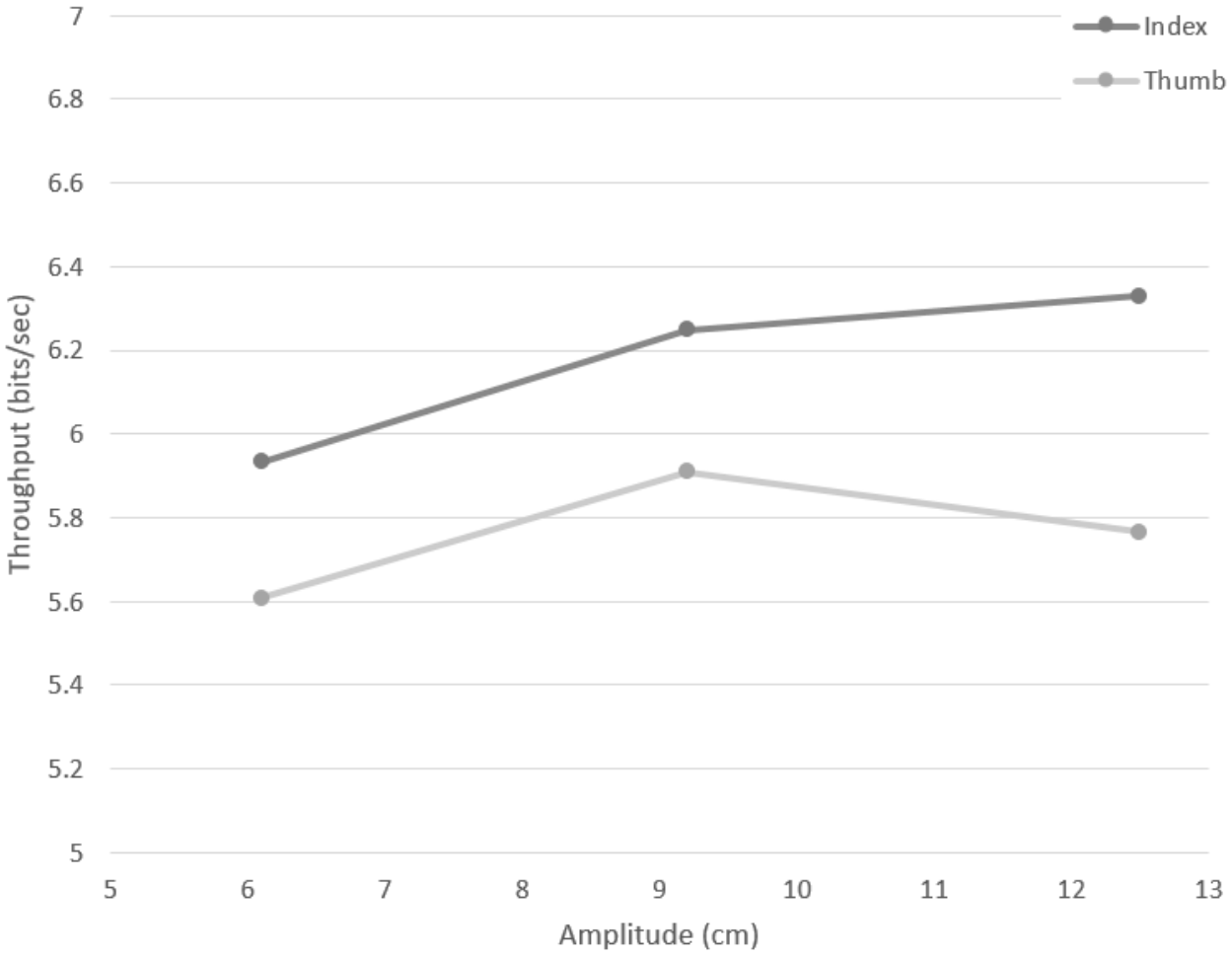


Figure 3.11 - Average results in the index (no hand support) and thumb (with hand support) conditions, plotted on a graph of throughput versus amplitude.

There was a statistically significant interaction effect of Finger x Amplitude on throughput ($F_{2,46}$=8.1, p<.001). In the graph above, the slope between the first and second amplitudes appears similar, between the two finger conditions. However, the slope between the second and third amplitude values is different between the two finger conditions. The difference between index and thumb is most pronounced at the largest amplitude, as would be predicted from the participant's feedback.

It is noteworthy that the throughput in the index (no support) condition was consistently higher than in the thumb (with support) condition, across all three amplitude values. For each amplitude

value, the difference between index and thumb was statistically significant, as shown in Table 3.12, above.

Vibration x A also had a statistically significant impact on throughput ($F_{2,46}$=3.89, p<.05). However, looking at the following graph, the interaction effect appears to be less than the individual impacts of vibration and amplitude, separately.

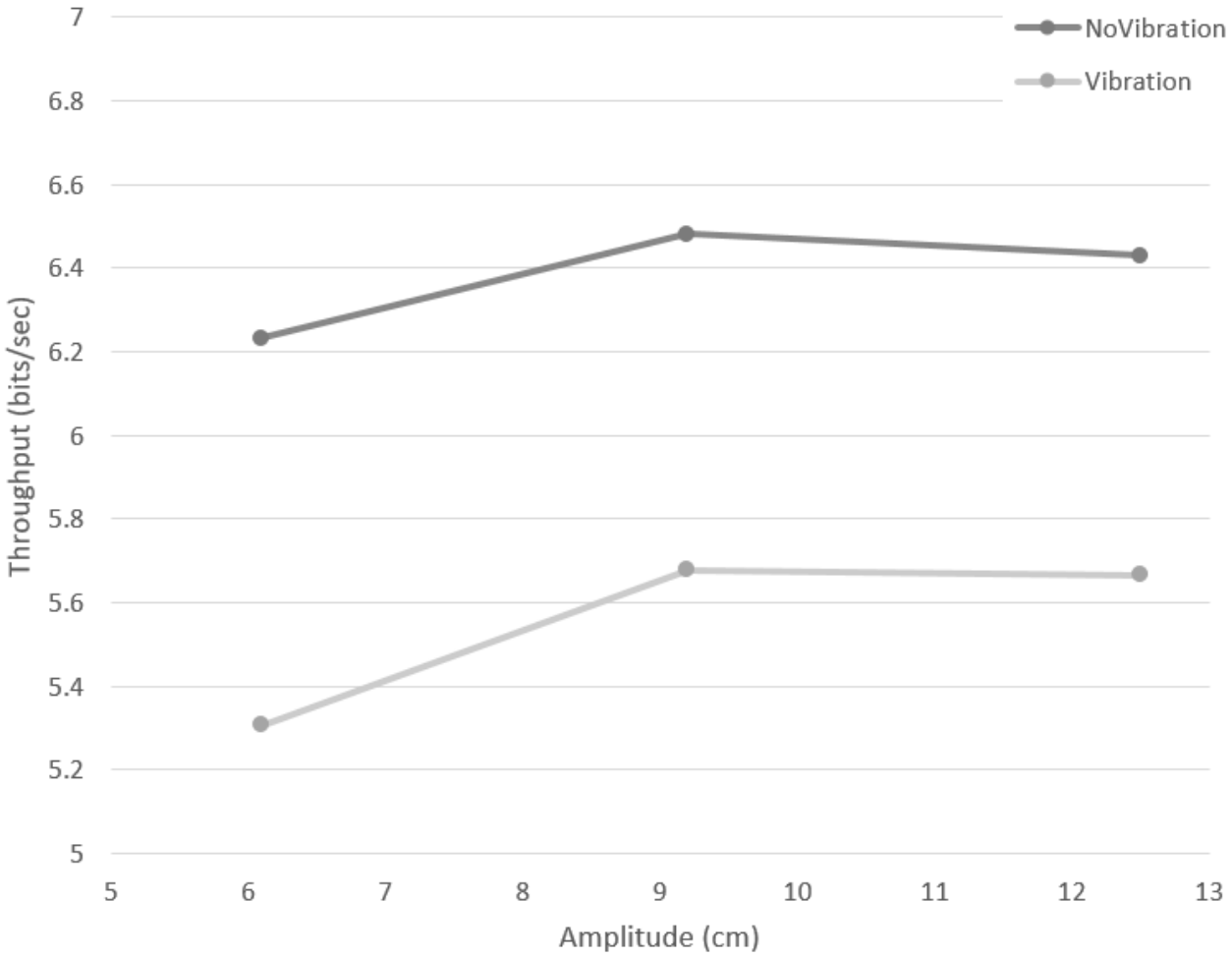


Figure 3.12 - Average results in vibration and non-vibration conditions, plotted on a graph of throughput versus amplitude. Both the large touch monitor, as well as the small MCDU screen have been excluded from the average.

There was no significant interaction effect on throughput from Vibration x Finger x A ($F_{4,46}$=1.44, p>.05). However, it is still useful to show the graph, since it is of interest whether the index finger continues to have higher throughput than the thumb, under vibration, across amplitude conditions.

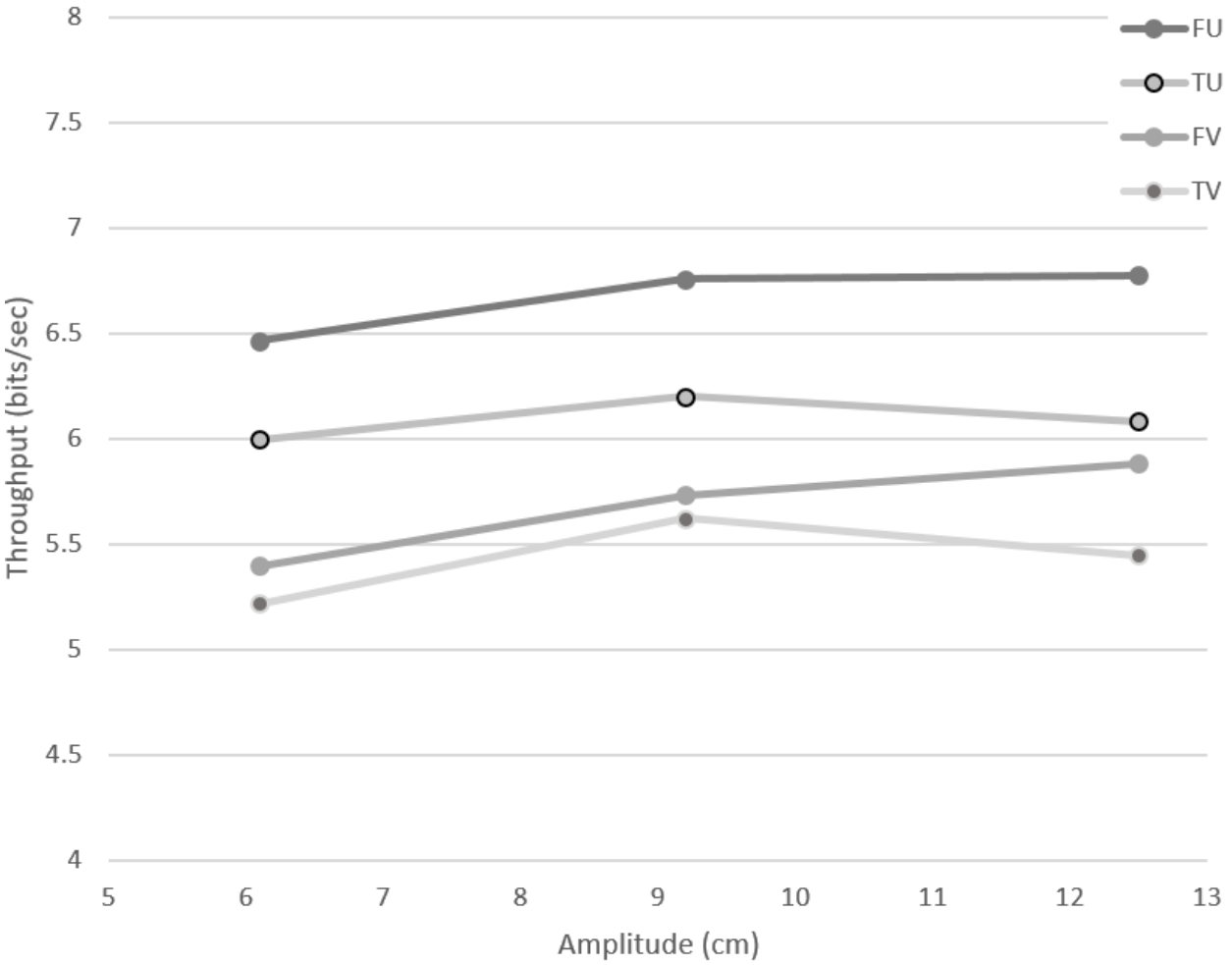


Figure 3.13 - Average results for each Vibration x Finger condition, plotted on a graph of throughput versus amplitude

3.3 Error Rate

Error rate results per condition, per target size and condition, and per amplitude and condition.

3.3.1 Error Rate per Condition

Average error rate results per condition.

3.3.1.1 Statistical Analysis for Error Rate per Condition

Table 3.13 shows the ANOVA results for the impact of each factor and cross-factor on error rate. Table 3.14 shows the p-value results for the pairwise error rate comparisons, comparing the difference between specific conditions in terms of their error rates.

Table 3.13 - ANOVA results for whether each factor and cross-factor has a statistically significant impact on error rate.

Error Rate x Condition x Participant - Excludes Large Screen						
	DFn	DFd	F	p-value	Method Used	Reason for Method Used
Vibration	1	23	177.98	p<.0001*	Parametric	
Finger	1	23	8.79	p<.01*	Parametric	
Screen	3	345	127.33	p<.0001*	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Finger	1	345	13.62	p<.0005*	Non-Parametric	Brown-Forsythe test failure
Vibration x Screen	3	345	10.85	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Finger x Screen	3	345	5.51	p<.005*	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x Screen	3	345	1.08	p>.05	Non-Parametric	Brown-Forsythe test failure
Error Rate x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen	4	207	32.58	p<.0001*	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Screen	4	207	4.92	p<.001*	Non-Parametric	Brown-Forsythe test failure

Table 3.14 - p-value results from pairwise comparisons on error rate per condition.

Error Rate per Condition T-Test or Wilcoxon Signed-Rank Test P-Values with Holm-Bonferroni Correction				
Screen Pairwise Comparisons				
	iPad Pedestal	Large Screen	Medium Screen	Small Screen
iPad MIP	p<.0001*	p<.0005*	p<.0001*	p<.0005*
iPad Pedestal		p<.05*	p<.0005*	p<.001*
Large Screen			p>.05	p>.05
Medium Screen				p>.05
Finger x Vibration Pairwise Comparisons				
	Index x Vibration	Thumb x No Vibration	Thumb x Vibration	
Index x No Vibration	p<.0001*	p<.0005*		
Index x Vibration			p>.05	
Thumb x No Vibration			p<.0001*	
Screen Position x Vibration Pairwise Comparisons				
	iPad Pedestal x No Vibration	iPad Pedestal x Vibration		
iPad MIP x No Vibration	p>.05			
iPad MIP x Vibration		p<.0001*		

3.3.1.2 Error Rate Results per Condition

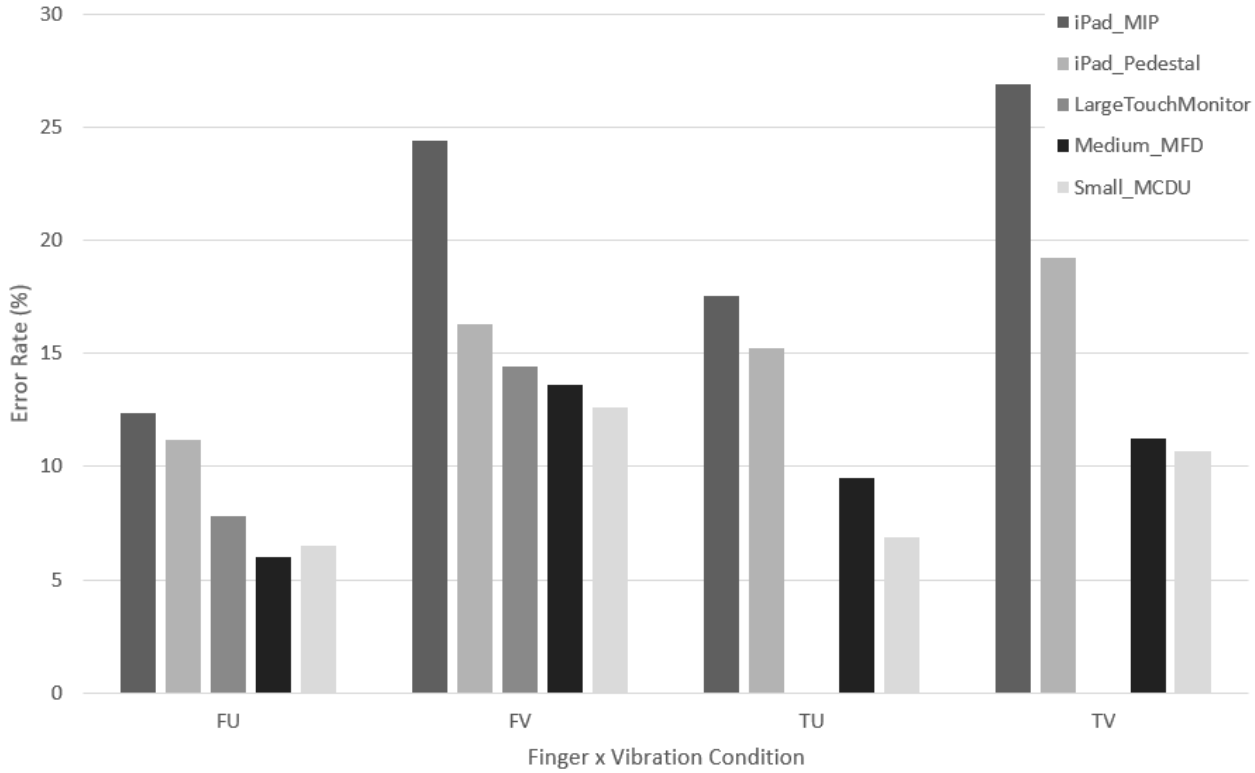


Figure 3.14 - Average error rate, across all participants, for every Vibration x Finger x Screen condition. FU represents the index-no-vibration condition. FV represents the index-vibration condition. TU represents the thumb-no-vibration condition. TV represents the thumb-vibration condition.

Of note, in the above graph, are the differences between the iPad and the other screens. The iPad had higher error rates of 20% on average in the MIP position (15% in the pedestal position), versus 9-11% for the other screens. As was mentioned previously, the software used for the target selection task on the iPad did not put a limit on error rate, per trial, per participant. Whereas, on the other screens, a limit of 50% was placed, per trial, per participant. If the error rate was over 50%, the participant was asked to redo that trial (that target size x amplitude condition). This was not done for the iPad. In addition, on all screens, only audio feedback was given when an error was made. There was no visual feedback to denote an error. On the iPad, this audio feedback was noticeably delayed and offset from the error, due to issues with the software used. As a result, it is unclear whether the iPad was truly more error prone, or whether the increased error on the iPad was caused by these differences.

Across all screens, the thumb-vibration condition caused more error than the index-no-vibration condition (for the small MCDU screen, the two were essentially equivalent). However, this was not the case when comparing the thumb-vibration condition against the index-vibration condition. On the iPad, the thumb-vibration caused more error than the index vibration condition. On the medium MFD screen and the small MCDU screen it was the inverse, with the index-vibration causing more error than the thumb-vibration condition.

Overall, when performing an average across screens, this discrepancy between the iPad and the other two screens canceled out:

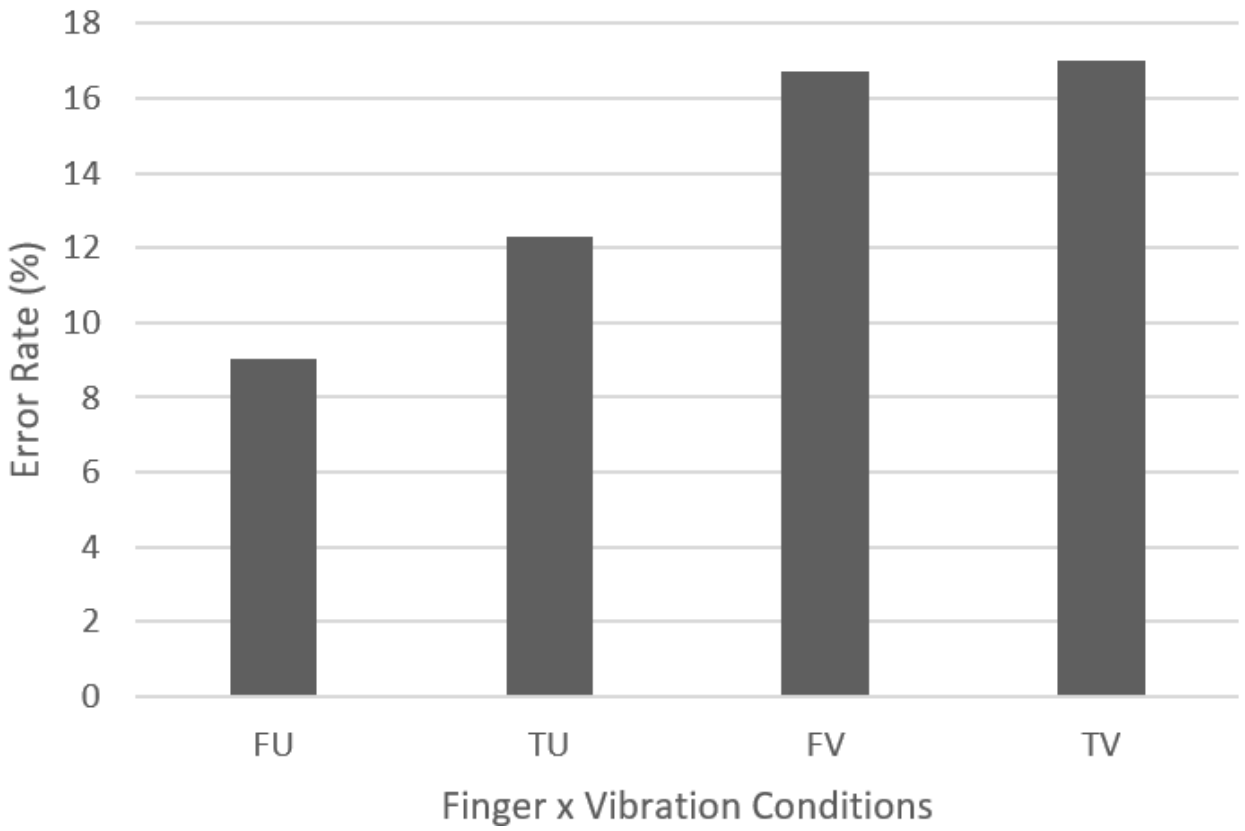


Figure 3.15 – Error rate in each Finger x Vibration condition, averaged across screens. The large touch monitor has been excluded from the average.

There was a significant interaction effect of Vibration and Finger on error rate ($F_{1,345}$=13.62, p<.0005). Vibration impacted the index finger more than the thumb in terms of error rate (the difference between the two vibration conditions was higher for the index finger, versus the thumb). However, the error rate of the index finger remained either equivalent or higher than the thumb, in both conditions. The index finger had significantly less error in the no-vibration condition, as

shown in Table 3.14, above. There was no statistically significant difference observed between the two finger conditions under vibration.

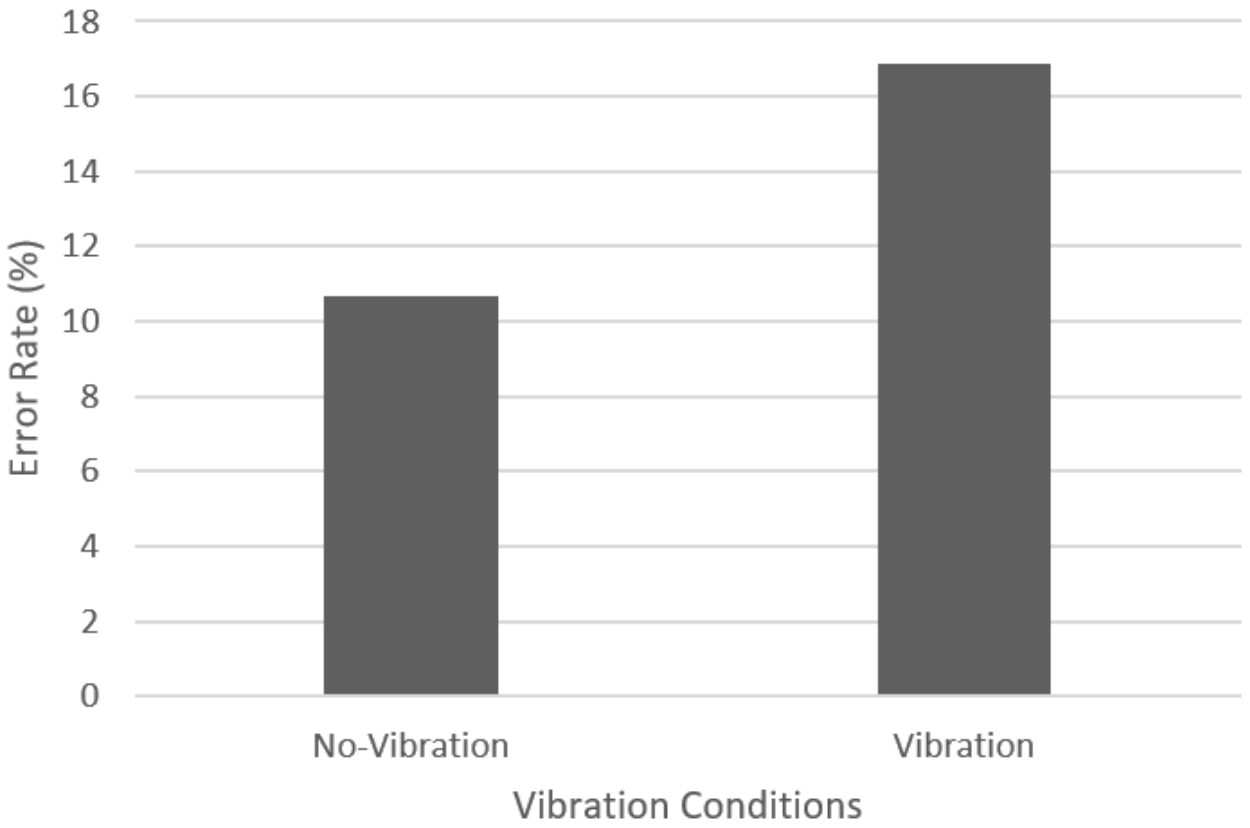


Figure 3.16 – Impact of vibration on error rate, compared to the no-vibration condition. The large screen was excluded from this average.

As would be expected, there was a significant difference in error rate caused by vibration, versus no-vibration ($F_{1,23}$=177.98, p<.0001).

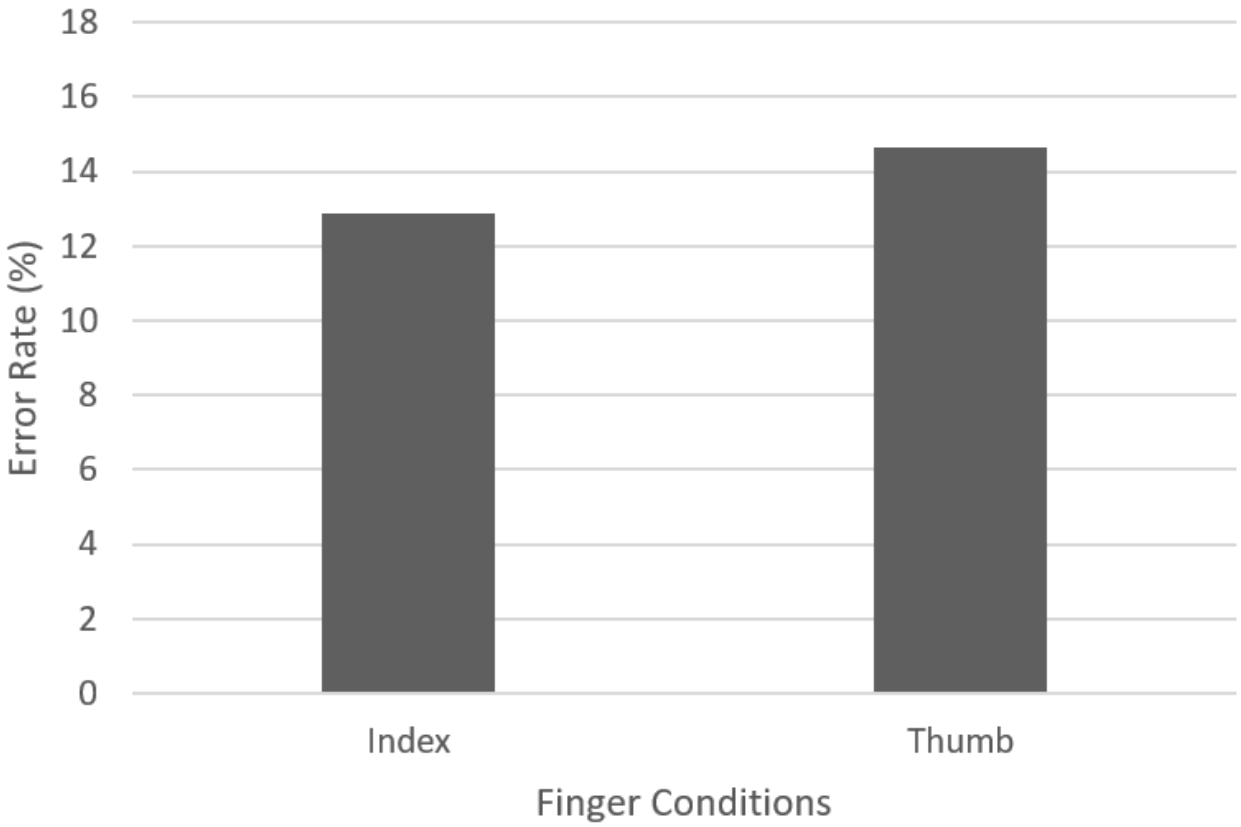


Figure 3.17 – The error rate when using the index finger (no hand-support) versus using the thumb (with hand-support). The large screen was excluded from this average.

There was a significant impact of finger on error rate ($F_{1,23}$=8.79, p<.01). On average, using the index finger, without a hand-support, caused less errors than using the thumb, with a hand-support.

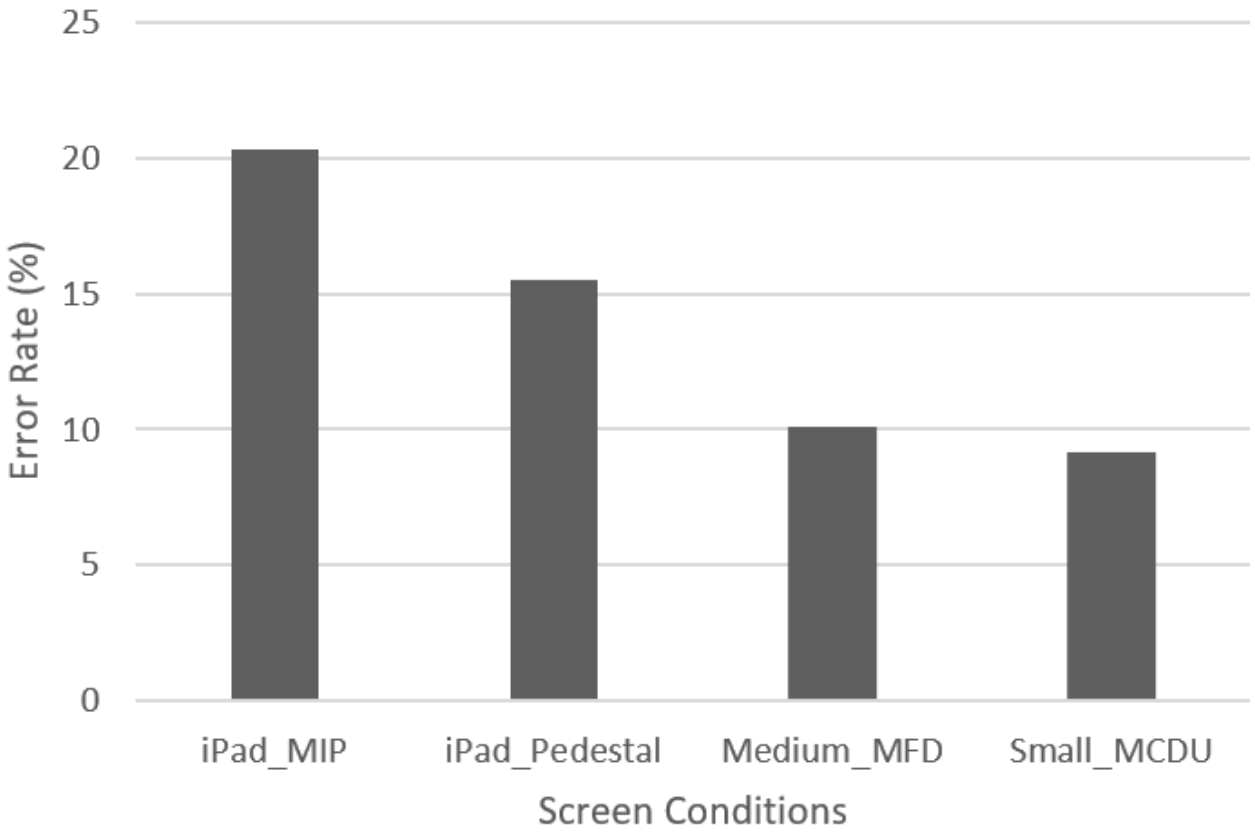


Figure 3.18 – Average error rate per screen. The large screen was excluded from this graph.

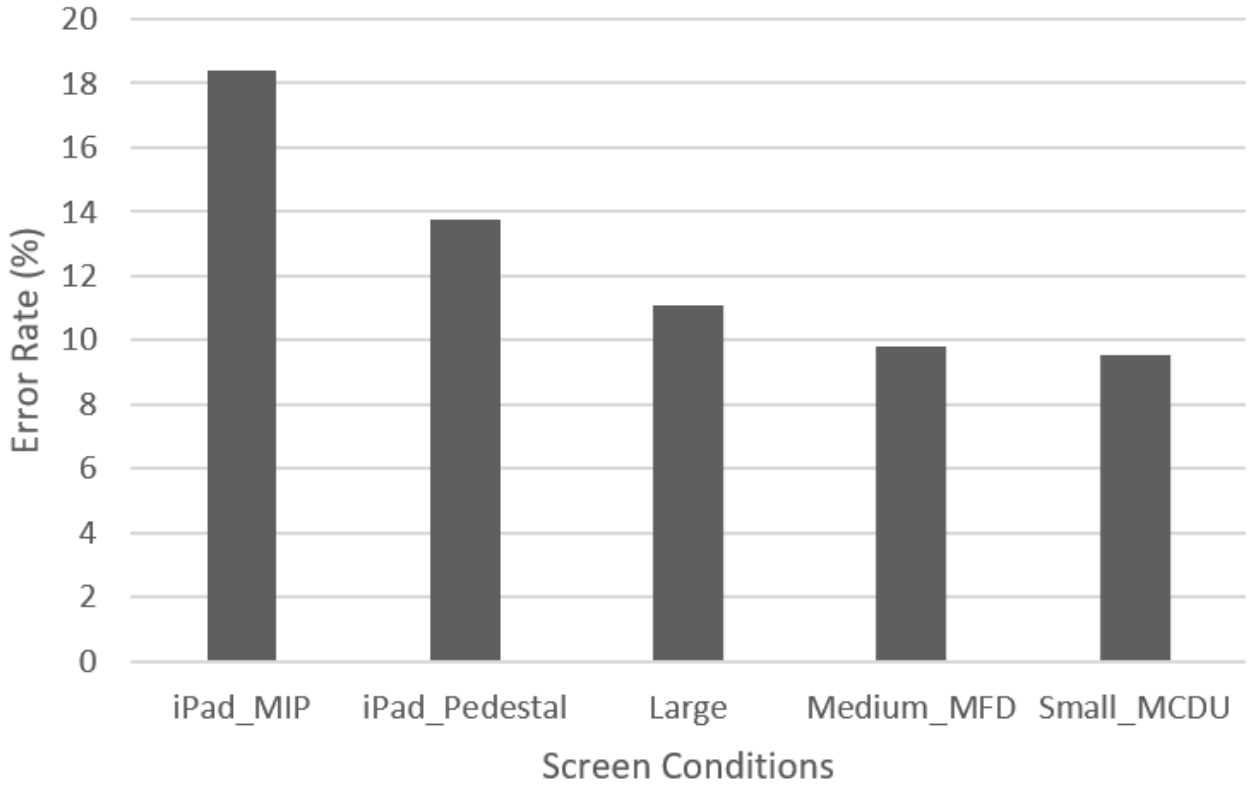


Figure 3.19 – Average error rate in the index finger conditions, per screen. The thumb conditions have been excluded from this graph.

When the large screen is excluded, and the thumb conditions included in the average, there is a significant impact of screen on error rate ($F_{3,345}$=127.33, p<.0001). When the large touch monitor is included, but the thumb conditions excluded from the average, there is also a significant impact of screen on error rate ($F_{4,207}$=32.58, p<.0001).

Table 3.14, above, shows which differences between each pair of screens were significant, for error rate. Overall, the iPad, in both positions, appears to have had worse error rates than any other screen. The large, medium and small screens had comparable error rates, with no significant differences observed between them. The iPad in the pedestal position had significantly less error than the iPad in the MIP position.

These results can now be combined with the throughput results shown previously, to determine which screen was best, in terms of both throughput and error rate. The small and large screens had the highest throughputs and lowest error rates. The iPad in the pedestal position had high throughputs but also high error rates. The medium MFD screen had among the lowest throughputs

but also low error rates. The iPad in the MIP position had among the lowest throughputs and the highest error rate.

Of particular note, the iPad in the pedestal position also had both significantly lower error rate and higher throughput than the iPad in the MIP position. There was no significant difference between error rate in the two iPad positions without vibration ($p<.05$). However, there was a significant difference between error rate in the two iPad positions under vibration ($p<.0001$). Remember that this was true for throughput as well. Once again, it appears that the difference between the two positions has been amplified by vibration.

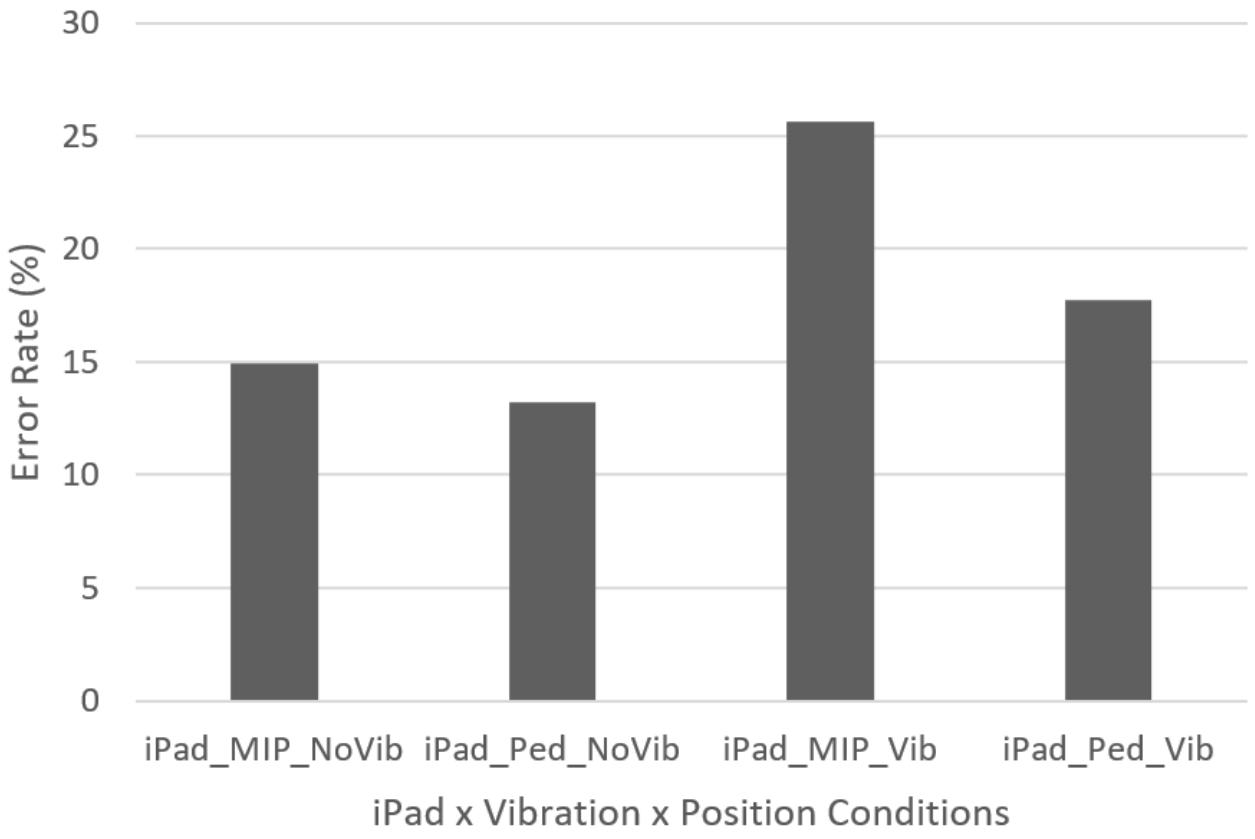


Figure 3.20 – Error rate for the iPad in each Position x Vibration condition.

3.3.2 Error Rate per Target Size and Condition

Average error rate results per target size and condition.

3.3.2.1 Statistical Analysis for Error Rate per Target Size and Condition

Table 3.15 shows the ANOVA results for whether each factor and combined factors crossed with target size had a statistically significant impact on error rate. Table 3.16 shows pairwise error rate comparisons, comparing specific target sizes against each other, and comparing specific combinations of Condition x Target Size.

Table 3.15 - ANOVA results for whether target size, and each combination of factors crossed with target size, had a statistically significant impact on error rate.

Error Rate Averaged per W (Target Size) x Condition x Participant- Excludes Large Screen						
	DFn	DFd	F	P	Method Used	Reason for Method Used
W	3	1449	1146.7	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Vibration x W	3	1449	94.42	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Finger x W	3	1449	3.01	p<.05*	Non-Parametric	Brown-Forsythe test failure
Screen x W	9	1449	29.44	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x W	3	1449	8.98	p<.0001*	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Screen x W	9	1449	1.63	p>.05	Non-Parametric	Shapiro and Brown-Forsythe test failure
Finger x Screen x W	9	1449	1.78	p>.05	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x Screen x W	9	1449	0.47	ns	Non-Parametric	Shapiro and Brown-Forsythe test failure
Error Rate Averaged per W (Target Size) x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen)						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen x W	12	897	11.27	p<.0001*	Non-Parametric	Brown-Forsythe test failure
Vibration x Screen x W	12	897	1.42	p>.05	Non-Parametric	Shapiro and Brown-Forsythe test failure

Table 3.16 - p-value results from pairwise comparisons on error rate per target size (W) averages.

Error Rate Averaged per W (Target Size) x Condition Being Compared x Participant T-Test or Wilcoxon Signed-Rank Test P-Values with Holm-Bonferroni Correction							
Per W Comparisons- Excluding Large Screen							
	W=1.5		W=1		W=0.8		
W=2	p<.0001*		p<.0001*		p<.0001*		
W=1.5			p<.0001*		p<.0001*		
W=1					p<.0001*		
W=0.8							
Per W Comparisons- Just Index (Includes Large Screen)							
	W=1.5		W=1		W=0.8		
W=2	p<.0001*		p<.0001*		p<.0001*		
W=1.5			p<.0001*		p<.0001*		
W=1					p<.0001*		
W=0.8					-		
Vibration x W Comparisons (Vib=Vibration, No-Vib=No-Vibration)							
	Vib x W=1.5	Vib x W=1	Vib x W=0.8	No-Vib x W=2	No-Vib x W=1.5	No-Vib x W=1	No-Vib x W=0.8
Vib x W=2	p<.0001*	p<.0001*	p<.0001*	p<.05*			
Vib x W=1.5		p<.0001*	p<.0001*		p<.05*		
Vib x W=1			p<.0001*			p<.0001*	
Vib x W=0.8							p<.0001*
No-Vib x W=2					p<.005*	p<.0001*	p<.0001*
No-Vib x W=1.5						p<.0001*	p<.0001*
No-Vib x W=1							p<.0001*

Table 3.16 - p-value results from pairwise comparisons on error rate per target size (W) averages (continued and end)

	Screen x W Comparisons (i=iPad MIP, iP=iPad Pedestal, L=Large Screen, M=Medium Screen, S=Small Screen)															
	iPad Pedestal				Large Screen				Medium MFD Screen				Small MCDU Screen			
	W=2	W=1.5	W=1	W=0.8	W=2	W=1.5	W=1	W=0.8	W=2	W=1.5	W=1	W=0.8	W=2	W=1.5	W=1	W=0.8
ixW=2	p>.05				p>.05				p<.05*				p<.0001*			
ixW=1.5		p<.005*				p<.0001*				p<.0005*				p<.001*		
i xW=1			p<.001*				p<.001*				p<.0001*				p<.0001*	
ixW=0.8				p<.005*				p<.001*				p<.0001*				p<.0001*
iPxW=2					p>.05				p>.05				p<.01*			
iPxW=1.5						p>.05				p>.05				p<.001*		
iPxW=1							p>.01				p<.01				p<.005*	
iPxW=0.8								p>.05				p<.001*				p<.01*
LxW=2									p>.05				p>.05			
LxW=1.5										p>.05				p>.05		
LxW=1											p>.05				p>.05	
LxW=0.8												p>.05				p>.05
MxW=2													p>.05			
MxW=1.5														p>.05		
MxW=1															p>.05	
MxW=0.8																p>.05

3.3.2.2 Error Rate Results per Target Size and Condition

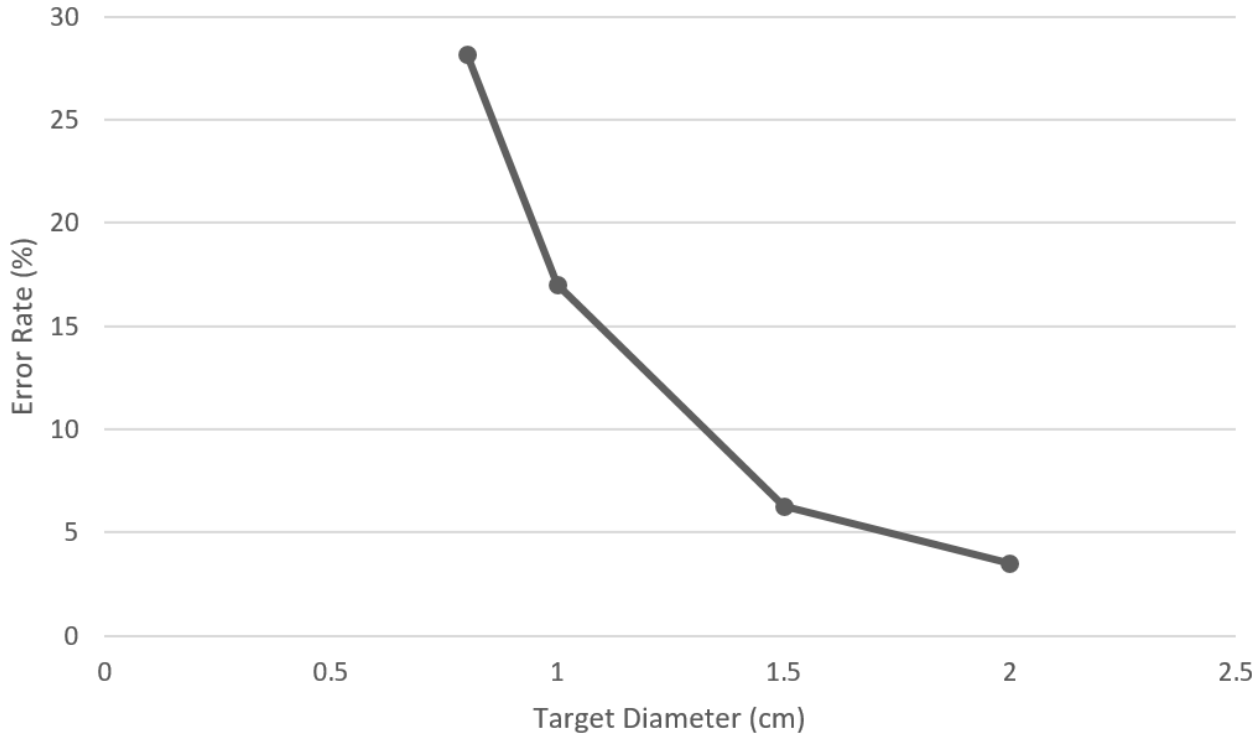


Figure 3.21 – Error rate per target size, averaged across all conditions and screens, excluding the large screen.

As can be clearly seen in the graph above, there was a significant impact of target size on error rate ($F_{3,1449}$=1146.7, p<.0001). Of particular interest, the impact of target size on error rate appears to be exponential, for the range of target sizes tested, with the error rate decreasing much more rapidly between the two smallest target sizes, compared to between the two largest ones. However, there was still a significant difference in error rate between the two largest target sizes, W=1.5 cm and W=2 cm (p<.0001). As a result, we cannot say for sure at which target size the error rate flattens out.

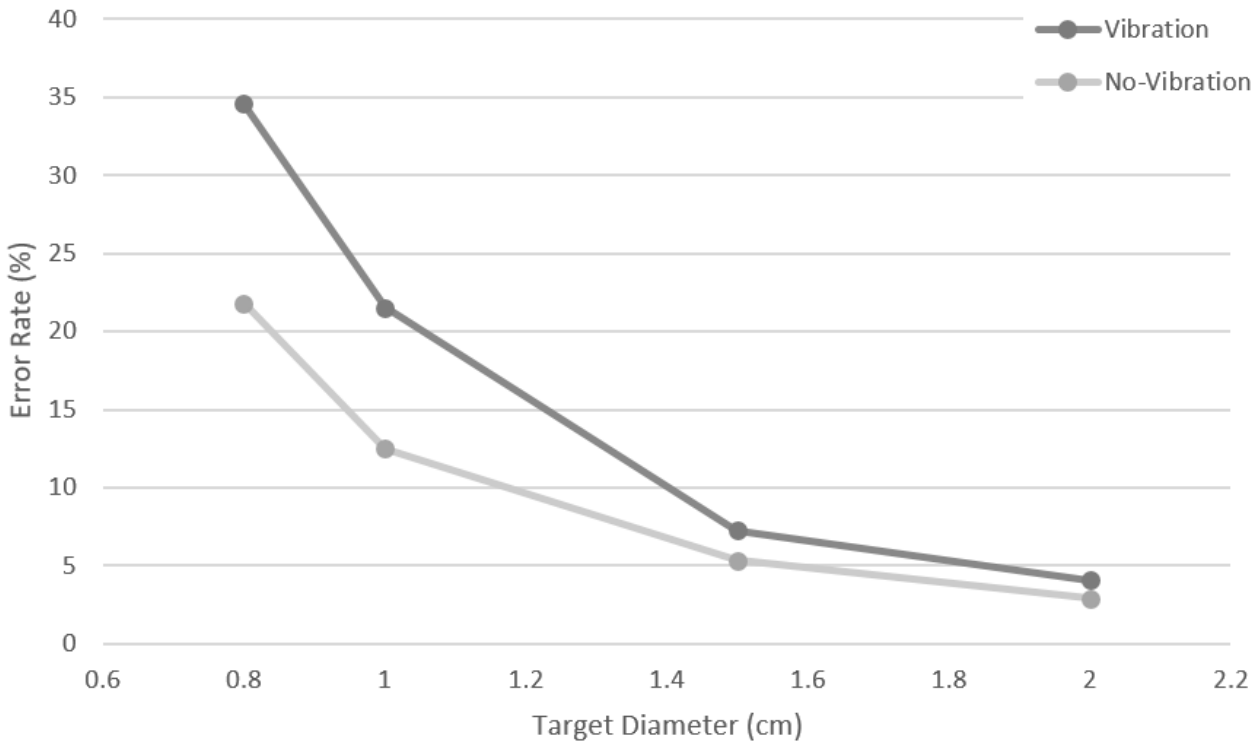


Figure 3.22 – Error rate per target size for the no-vibration condition average, compared against the vibration condition average. The large screen was excluded from these averages.

There was a significant interaction effect between Vibration and Target Size ($F_{3,1449}$=94.4, $p<.0001$). The difference between vibration and no-vibration appears more pronounced at smaller target sizes, with a smaller difference at larger target sizes. However, there was still a significant difference in error rate at W=2, between the vibration and no-vibration averages ($p<.05$). Looking at the graph above, the error rate, in both vibration and no-vibration conditions appears to follow an exponential decrease with increasing target size. However, the slope is higher in the vibration condition (error rate starts higher, but decreases faster, with increasing target size).

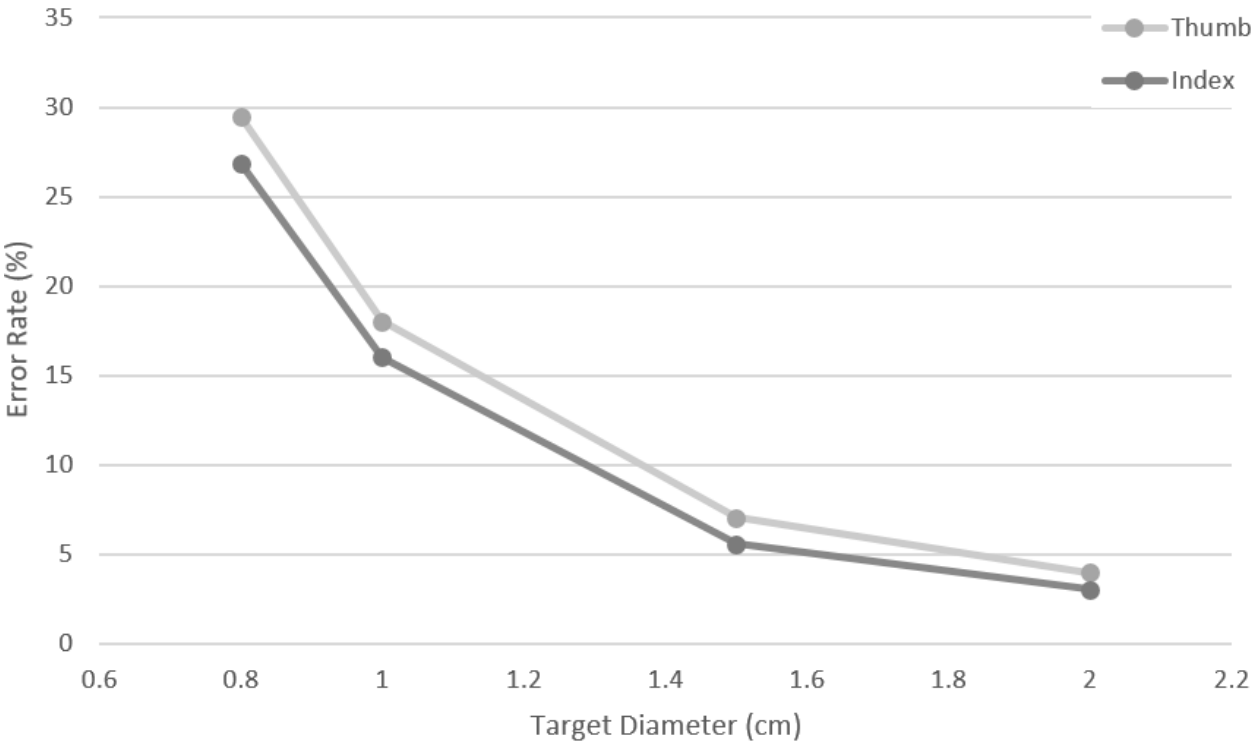


Figure 3.23 – Error rate per target size for the thumb (with hand-support) condition average, versus the index finger (without hand-support) average. The large touch monitor has been excluded from these averages.

There was a significant interaction effect between Finger and Target Size on error rate ($F_{3,1449}$=3.01, $p<.05$). Looking at the graph above, the interaction effect appears to be small, when compared to the individual impacts of finger and target size. There is a slightly larger difference at the smaller target sizes, when compared to the larger target sizes. Since the thumb is a larger finger than the index, it is to be expected that it would have a higher error rate at very small target sizes. However, the difference between the two finger conditions is remarkably consistent, overall, across target sizes, when considering error rate.

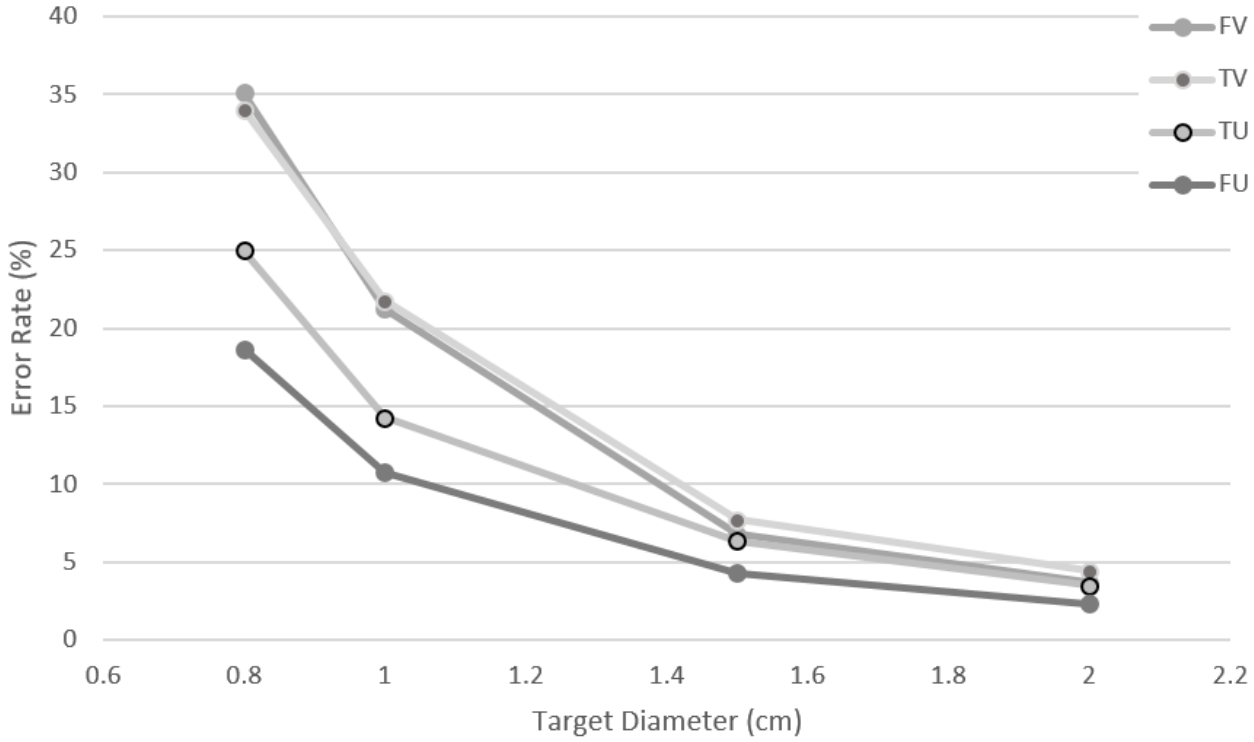


Figure 3.24 – Error rate per target size for each Finger x Vibration condition.

There was a significant interaction effect on error rate between Finger x Vibration x W ($F_{3,1449}$=8.98, p<.0001). Looking at the graph above, the error rate of the index finger under vibration appears to decrease slightly faster than the thumb under vibration.

3.3.3 Error Rate per Amplitude and Condition

Average error rate results per amplitude and condition.

3.3.3.1 Statistical Analysis of Error Rate per Amplitude and Condition

Table 3.17Table 3.9 shows the ANOVA results for whether each factor and combined factors crossed with amplitude had a statistically significant impact on error rate. Table 3.18 shows pairwise error rate comparisons, comparing specific amplitudes against each other, and comparing specific combinations of Condition x Amplitude.

Table 3.17 - ANOVA results for whether amplitude, and each combination of factors crossed with amplitude, had a statistically significant impact on error rate.

Error Rate Averaged per A (Amplitude) x Condition x Participant - Excludes Large and Small Screens						
	DFn	DFd	F	p	Method Used	Reason for Method Used
A	2	46	29.32	p<.0001*	Parametric	Brown-Forsythe test failure
Vibration x A	2	805	0.87	ns	Non-Parametric	Brown-Forsythe test failure
Finger x A	2	46	0.41	ns	Parametric	Brown-Forsythe test failure
Screen x A	4	805	0.95	ns	Non-Parametric	Shapiro and Brown-Forsythe test failure
Vibration x Finger x A	2	805	0.81	ns	Non-Parametric	Brown-Forsythe test failure
Vibration x Screen x A	4	805	0.23	ns	Non-Parametric	Brown-Forsythe test failure
Finger x Screen x A	4	805	0.16	ns	Non-Parametric	Brown-Forsythe test failure
Vibration x Finger x Screen x A	4	805	0.62	ns	Non-Parametric	Brown-Forsythe test failure
Error Rate Averaged per A (Amplitude) x Condition x Participant – Excludes Thumb Conditions (Includes Large Screen) – Excludes Small Screen						
	DFn	DFd	F	p	Method Used	Reason for Method Used
Screen x A	6	529	0.77	ns	Non-Parametric	Brown-Forsythe test failure
Vibration x Screen x A	6	529	0.48	ns	Non-Parametric	Brown-Forsythe test failure

Table 3.18 - p-value results from pairwise comparisons on error rate per Amplitude (A) averages.

Error Rate Averaged per A (Amplitude) x Condition Being Compared x Participant T-Test or Wilcoxon Signed-Rank Test P-Values with Holm-Bonferroni Correction		
Per Amplitude (A) Comparisons – Excluding the Large and Small Screens from the Average		
	A=9 cm	A=6 cm
A=12.5 cm	p<.001*	p<.0001*
A=9cm		p<.005*
Per Amplitude (A) Comparisons - Just Index Finger – Excluding the Small Screen		
	A=9 cm	A=6 cm
A=12.5 cm	p<.005*	p<.0001*
A=9cm		p<.001*

3.3.3.2 Error Rate Results per Amplitude and Condition

There was a significant impact of amplitude on error rate ($F_{2,46}$=29.32, p<.0001). However, none of the interaction effects between amplitude and vibration, finger, and/or screen were significant (p=ns)

Remember that the participants reported having more trouble reaching the furthest targets, in the largest amplitude condition. Also remember that there was a significant interaction effect on throughput between Finger and Amplitude. It is interesting to note that, while the furthest targets on the largest amplitude may have been more difficult to reach, this did not have a significant impact on error rate, only on throughput.

The graphs below still show various interaction conditions, in order to provide additional insight, even if the interaction effects were not significant. Once again, note that the small MCDU screen could not be included with the other screens, when evaluating amplitude, for the purposes of both the statistical analysis and graphs.

As seen in the graphs below, it appears as though error rate increases with amplitude.

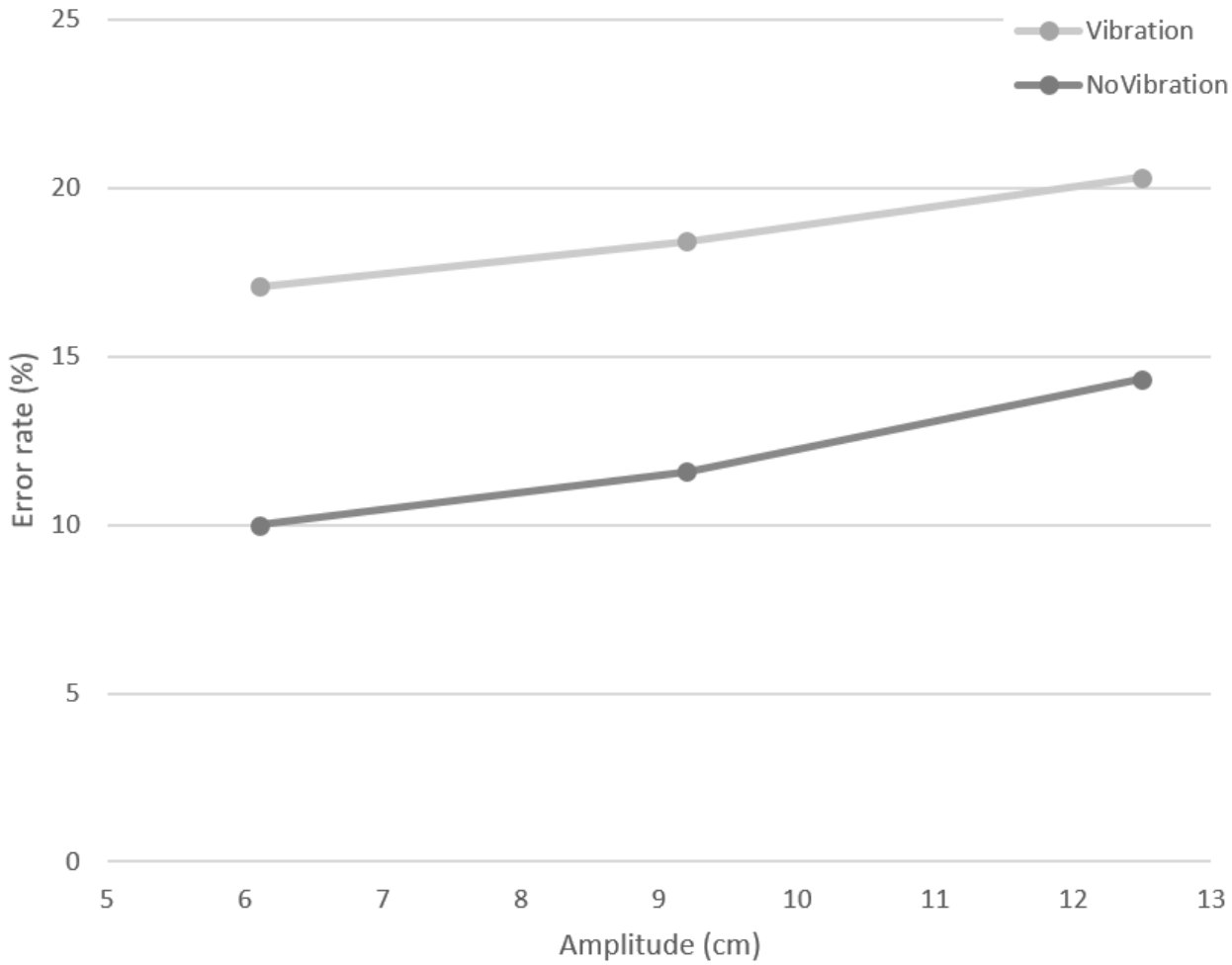


Figure 3.25 - Error rate versus amplitude in the thumb (with hand-support) versus index finger (without hand-support) conditions. The large touch monitor and small MCDU screen were excluded from this average.

While there was no significant interaction effect between Finger x Vibration, it does appear that the error rate slightly increased at the largest amplitude for the thumb condition, compared to the index condition. Of note, the index finger has lower error rate than the thumb, for all amplitude conditions.

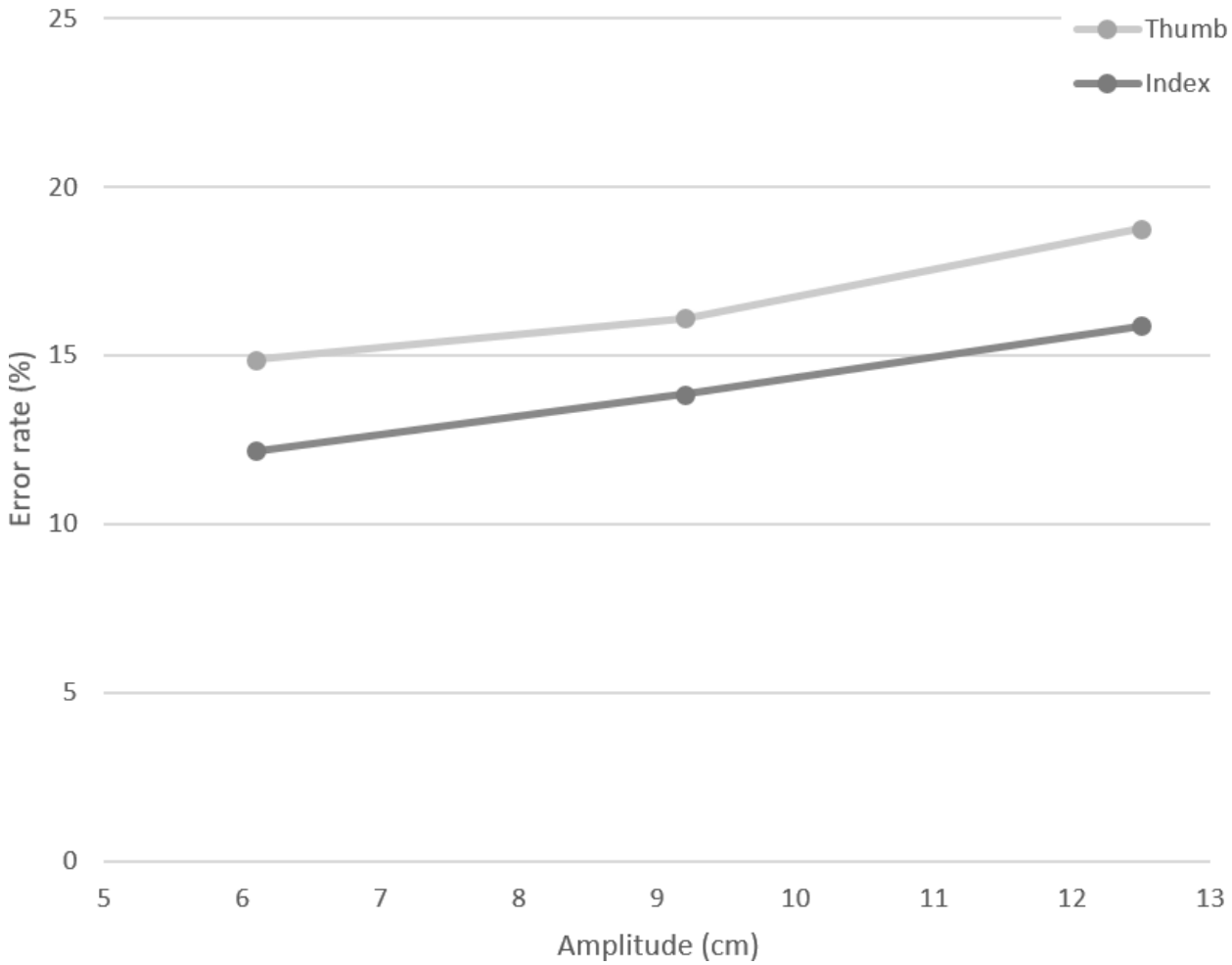


Figure 3.26 - Error rate versus amplitude for the vibration condition average, versus the no-vibration condition average. The large touch monitor and small MCDU screens have been excluded from this average.

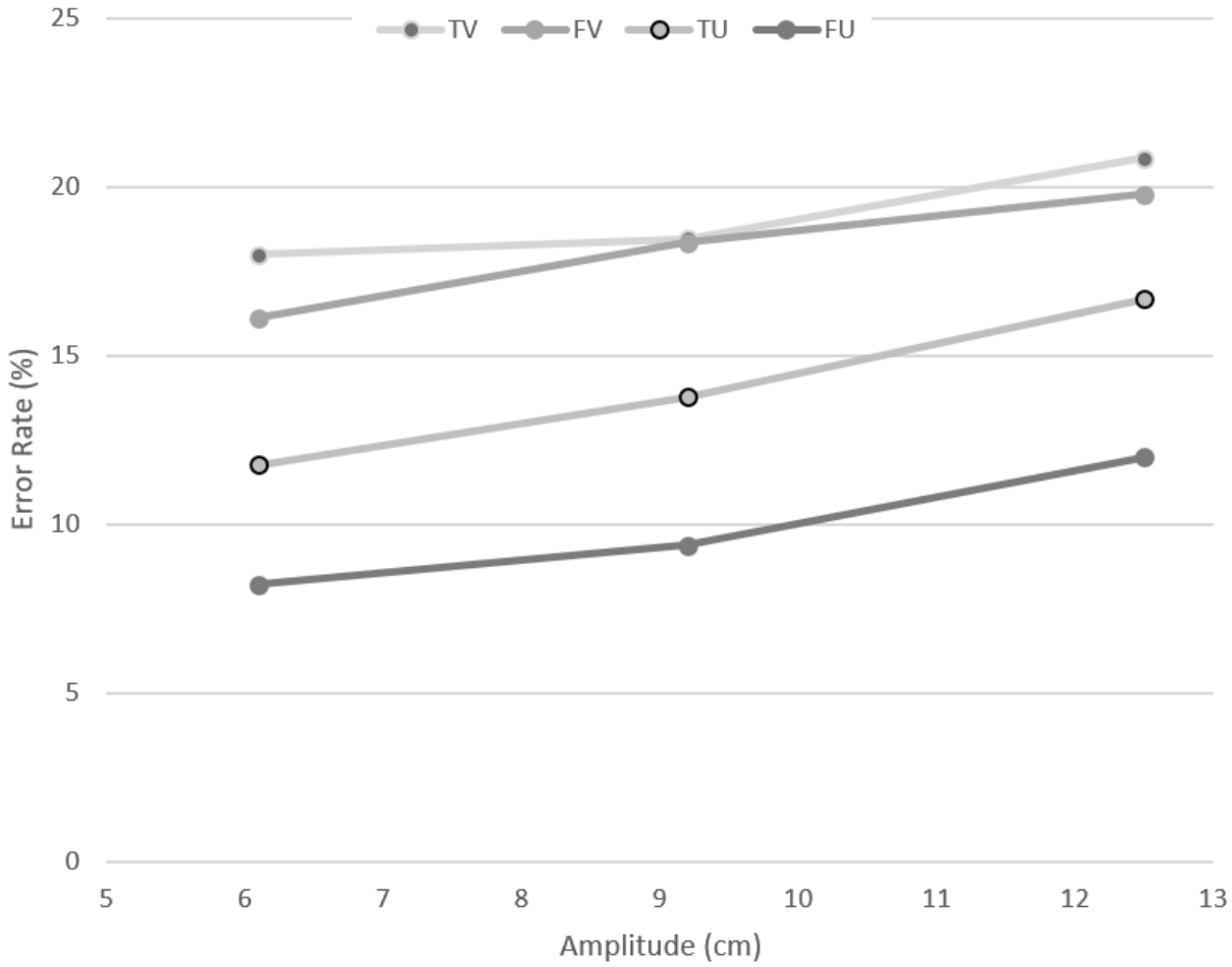


Figure 3.27 – Error rate versus amplitude for each Finger x Vibration condition.

Of note, in the graph above, the index-vibration condition has lower or comparable error rate than the thumb-vibration condition, across all amplitudes tested.

3.4 Double-Clicks

As described previously, the participants occasionally made accidental double-clicks. These double-clicks were excluded from the throughput calculations, since they violated Fitts' law assumptions. However, they were included in the error rate and are important for error considerations. In particular, it is useful to highlight these double-clicks separately from other forms of error because they may be more likely to go unnoticed.

The methods used to classify an error as a double-click were described previously.

There were several distinct types of double-clicks observed:

1. Accidentally tapping twice (or more times) in the same spot. This was most prevalent under vibration.
2. Confused tapping caused by a lack of responsiveness. In essence, the finger is moving towards the next target, but the previous target has not been selected due to a lack of screen responsiveness, causing a confused tap. This was most prevalent on the medium MFD screen, due to its lower responsiveness.
3. Accidentally brushing up against an active area of the screen, causing an unintentional tap. This was most prevalent in the thumb conditions, since the hand was always in close proximity to the screen edge.

It is difficult to disentangle these distinct types of double-taps, when looking at the data. As a result, they are presented together here. However, the third type of double tap, wherein the hand accidentally brushes up against part of the screen, may be particularly likely to go unnoticed, since the tap may occur in an area of the screen far from where the users are focusing their visual attention.

There were 417 double-clicks out of a total of 10,468 errors, across all trials and participants. Thus, double-clicks represent just 4% of errors. Table 3.19 shows the number of double-clicks observed for different test conditions. In particular, note that:

1) There was a greater number of double-clicks under vibration versus without vibration.
2) There was a greater number of double-clicks when using the thumb, versus the index finger. This is likely due to the hand and thumb hovering closer to the screen in the thumb condition, during all parts of the selection motion.
3) In particular, the thumb under vibration had a higher number of double-clicks. This is likely due to the combined effects of the hand and thumb hovering near the screen at all times, and vibration making it more likely that the hand or thumb will accidentally contact the screen.
4) The pedestal position had a higher number of double clicks than the MIP position.

5) The MFD had a higher number of double-clicks. This is likely due to its lower responsiveness.

Table 3.19 – Double-clicks associated with various conditions.

Condition	Double clicks	
	Number	(%)
Overall	417	100,0%
Vibration	281	67,4%
No Vibration	136	32,6%
Index Finger (excluding the large screen)	141	33,8%
Thumb (excluding the large screen)	261	62,6%
Index Finger x No Vibration (excluding the large screen)	42	10,1%
Index Finger x Vibration (excluding the large screen)	99	23,7%
Thumb x No Vibration (excluding the large screen)	92	22,1%
Thumb x Vibration (excluding the large screen)	169	40,5%
iPad MIP	84	20,1%
iPad Pedestal	102	24,5%
Large Touch Monitor	15	3,6%
Medium MFD	130	31,2%
Small MCDU	86	20,6%

3.5 96% Adjusted Target Size ($W_{96\%}$) per Target Size (W)

As mentioned previously, we calculated a circle size that encapsulated 96% of all taps, across all participants for a given trial. We called this circle size $W_{96\%}$, and considered it a more direct measurement of the We formulas proposed by Soukoreff and MacKenzie [18]. We proposed it as a potential tool to aid in the search for a minimum recommended target size under vibration conditions.

In this section we present $W_{96\%}$ versus W. Of particular interest is the $W_{96\%}$ value at the smallest target size, W=0.8 cm, since it may give additional insight into the minimum target size that is reasonably attainable at a decent speed for this type of task, under the tested conditions.

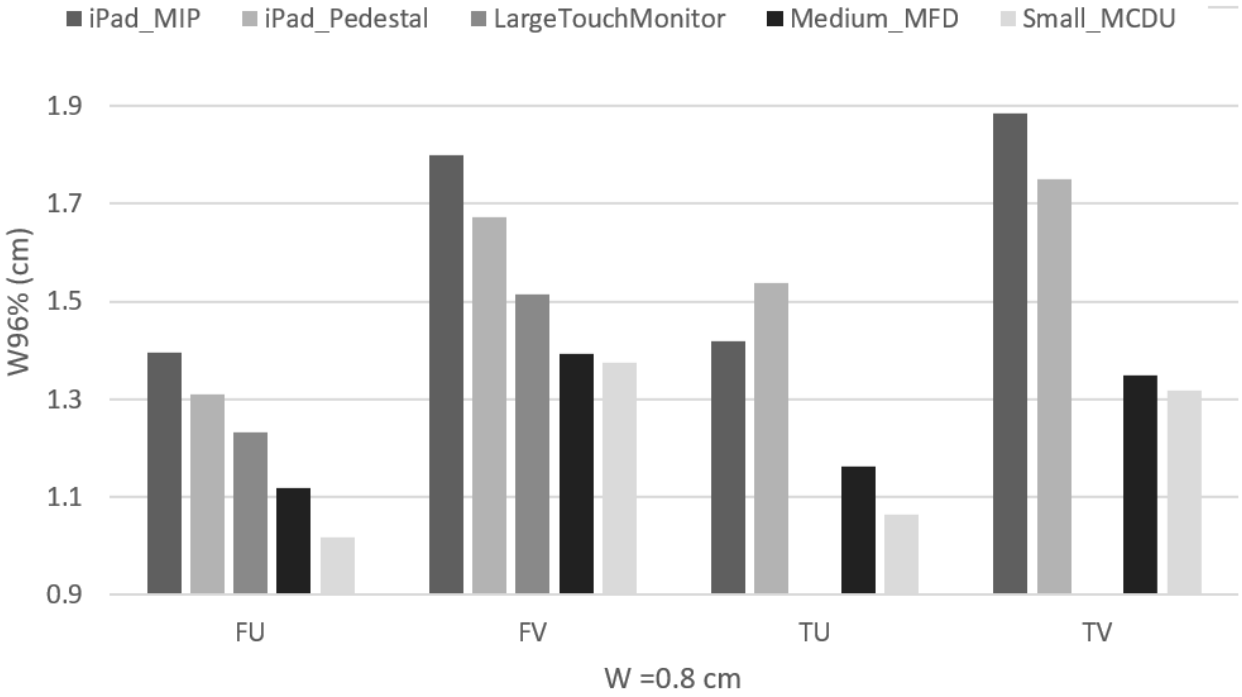


Figure 3.28 - $W_{96\%}$ for each Vibration x Finger x Screen condition, at W=0.8 cm. FU represents the index-no-vibration condition. FV represents the index-vibration condition. TU represents the thumb-no-vibration condition. TV represents the thumb-vibration condition.

Figure 3.28 shows that there are large difference in $W_{96\%}$ across different screens, when W=0.8 cm. In particular, note that the medium and small screens have smaller $W_{96\%}$ values in the thumb-vibration condition, versus the index-vibration condition, whereas it was the opposite for the iPad. This mirrors the error rate results shown previously, in Figure 3.14.

If we treat these results as potentially representative of the minimum reasonably target size on each device, under each condition, the iPad in the MIP position has larger minimum target size, of nearly 1.9 cm under vibration, versus the medium or small screens, which have a minimum target size of less than 1.4 cm under vibration.

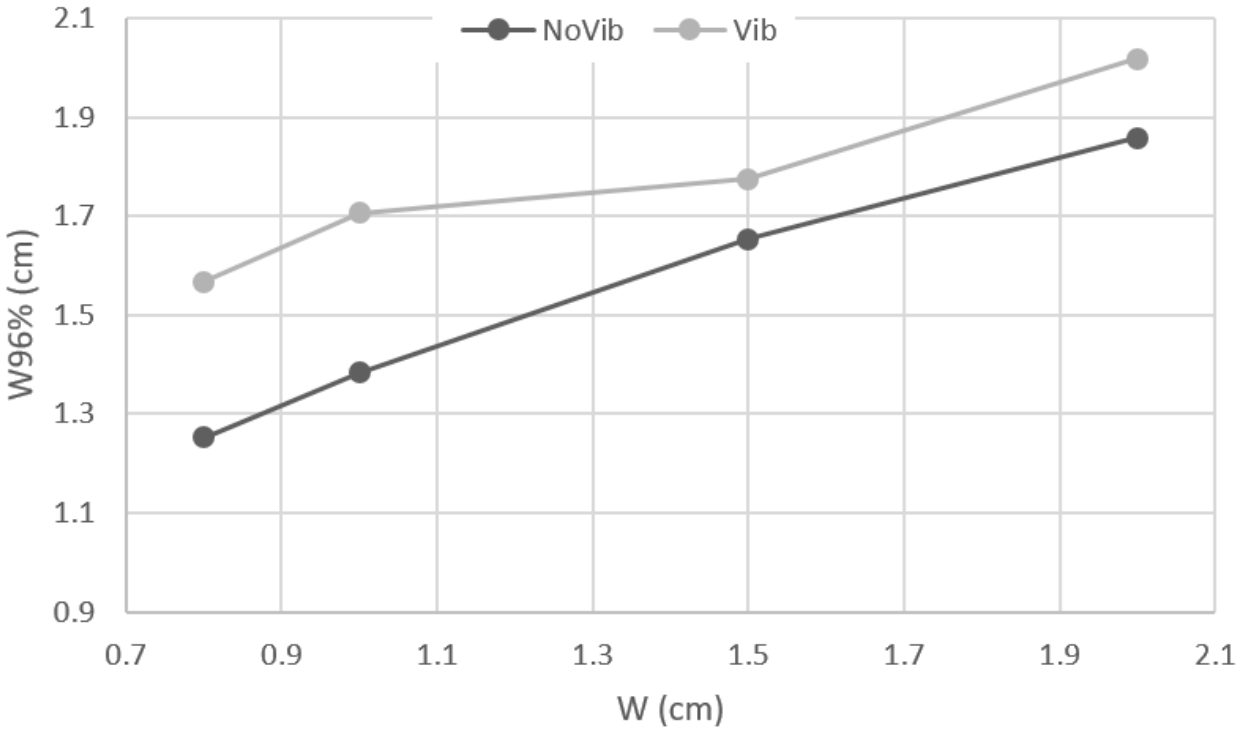


Figure 3.29 - Average results in vibration and non-vibration conditions, plotted on a graph of throughput versus target size. The large touch monitor has been excluded from the average.

If the end-point selections made by users were distributed within the target, then we would expect $W_{96\%}$ <= W. Figure 3.29 shows that this was not the case. When comparing vibration condition against no-vibration conditions, the difference in $W_{96\%}$ was larger at smaller target sizes, compared to larger target sizes. This was expected, since vibration makes it more difficult to attain smaller targets compared to larger targets. However, it is noteworthy that the two lines do not converge at the larger target sizes measured, and remain offset from each other. It is also noteworthy that the lines appear to be broadly linear, and have not flattened out, for the range of target sizes tested in this study.

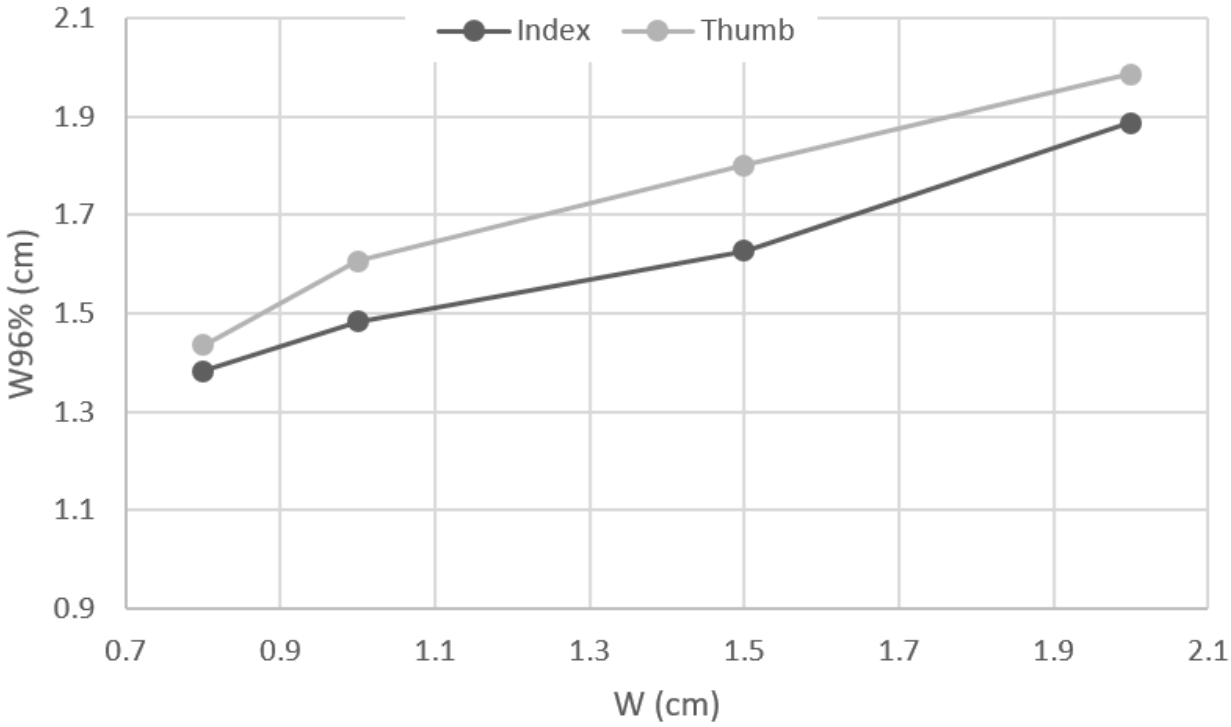


Figure 3.30 - Average results in the index (no hand support) and thumb (with hand support) conditions, plotted on a graph of throughput versus target size. The large touch monitor has been excluded from the average.

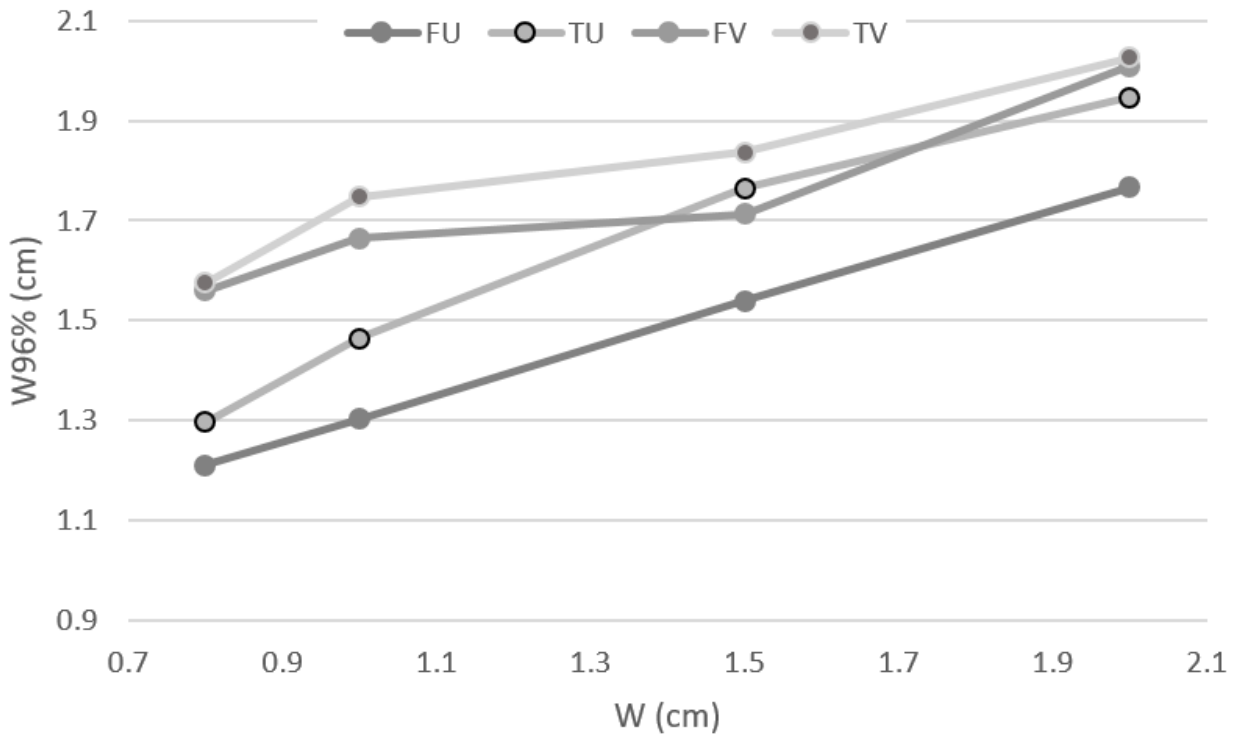


Figure 3.31 - Average results for each Vibration x Finger condition, plotted on a graph of throughput versus target size.

Figure 3.30 and Figure 3.31 show that the index finger had a smaller or equivalent $W_{96\%}$ size, compared to the thumb, across all target sizes and vibration conditions, when looking at the average across all screens. However, as previously noted, there may be some divergence between the medium MFD screen and small MCDU screen, versus the iPad, when looking at thumb-vibration versus index-vibration.

3.6 Effect of Selection Angle

Several participants mentioned that, in the hand-support condition, the targets furthest from the hand were difficult to reach. As a result, it is of interest to see whether target angle may have impacted the thumb condition more than the index finger condition.

In this section, movement time, error rate, and dx values were averaged per target angle, per condition. This average is across target sizes, amplitudes, participants and screens, with the large screen and small MCDU screen excluded from the averages.

This analysis per selection angle has several limitations:

1. Other studies that analyzed the impact of selection angle on touchscreen used a different target layout and selection pattern [8, 33] than the one used in this study.

2. We did not balance for selection angle; the participants always selected the targets in the same order. For the iPad, the participants always started with the target on the positive Y-axis. For the other screens, the participants always started with the target along the positive X-axis.

   Looking qualitatively at the results, the order effects may have adversely effected the first couple of targets selected more than the other targets, since the participants may have needed a slight ramp up before getting into the flow of the task. The first two recorded targets in the sequence on the iPad were at 192° and 288°. The first two recorded targets in the sequence on the other screens were at 24° and 120°. Therefore, particular caution is advised when interpreting the results at these angles.

3. The small MCDU screen was excluded because, when using the small MCDU screen in the thumb condition, participants were asked to hold the screen from the top, rather than the side, when supporting their hand. As a result, for the thumb condition, different targets/angles were furthest when using the small MCDU screen.

4. Only those participants that exclusively used their right hands were included in the averages and graphs shown in this sections (three participants out of 24 were excluded from the following averages). This was because, in the thumb condition, participants held onto the side of the screen corresponding to their handedness, which meant that different targets were closest and farthest, between right-handed versus left-handed participants.

5. The targets on the iPad were offset from the targets on the other screens by an angle of 6°. This difference was assumed to be negligible, and the iPad values were averaged with the medium MFD values in this section, as though this 6° offset did not exist. The iPad angles were used in the graphs in this section, since 0° was at the top for the iPad. In essence, the medium MFD screen values have been shifted by +6° and then averaged with the iPad, in the graphs shown in this section.

While the polar graphs shown in this section are interesting, it is difficult to draw broader conclusions from them, since there are so many points to compare against each other. It is easier to compare averages for motions across quadrants of the screen. Unfortunately, 15 targets were selected per trial, and 15 does not divide evenly by four. As a result, one of the angles was used for two cross-quadrant motions. The results at 0° were used both for the bottom-left-to-top-right quadrant average, as well as the bottom-right-to-top-left quadrant average. Table 3.20 shows the angles that correspond to each cross-quadrant motion.

Table 3.20 – The movement angles that were averaged together to show the value for each cross-quadrant motion.

Cross-Quadrant Motion	Movement Angles Averaged for Each Cross Quadrant Motion, in Degrees
Bottom-Right-To-Top-Left	0°, 288°, 312°, 336°
Bottom-Left-To-Top-Right	0°, 24°, 48°, 72°
Top-Right-To-Bottom-Left	192°, 216°, 240°, 264°
Top-Left-To-Bottom-Right	96°, 120°, 144°, 168°

3.6.1 Movement Time Versus Selection Angle

The following graphs show movement time, in milliseconds, versus selection angle. These are average values across target sizes, amplitudes, participants that used their right hands, and the iPad MIP iPad pedestal and medium MFD screens.

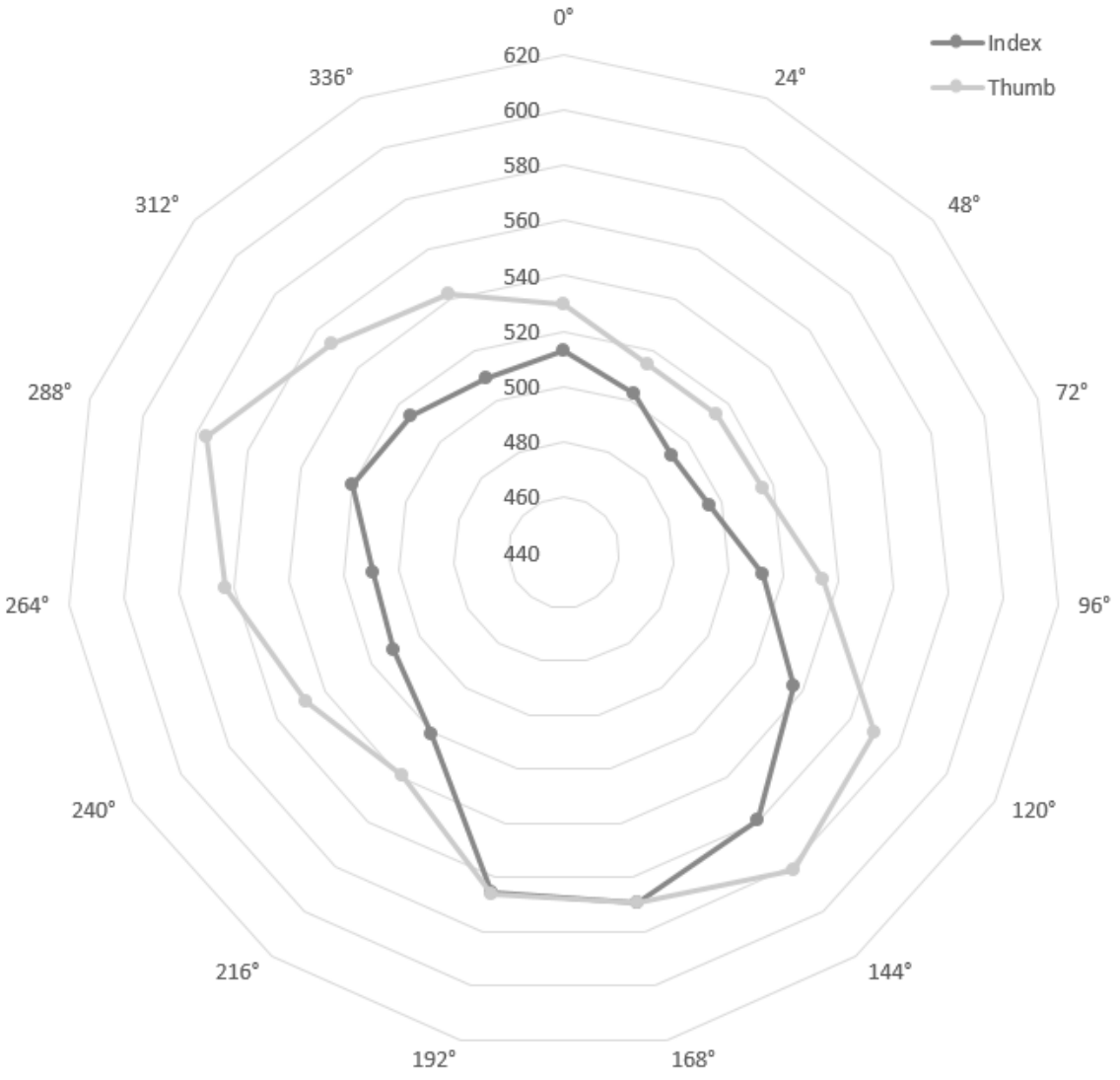


Figure 3.32 - Movement time (in milliseconds) versus selection angle, for the index finger and thumb conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands. Note that any increase in movement time at 192°, 288°, 24° and/or 120 may be partially due to order effects.

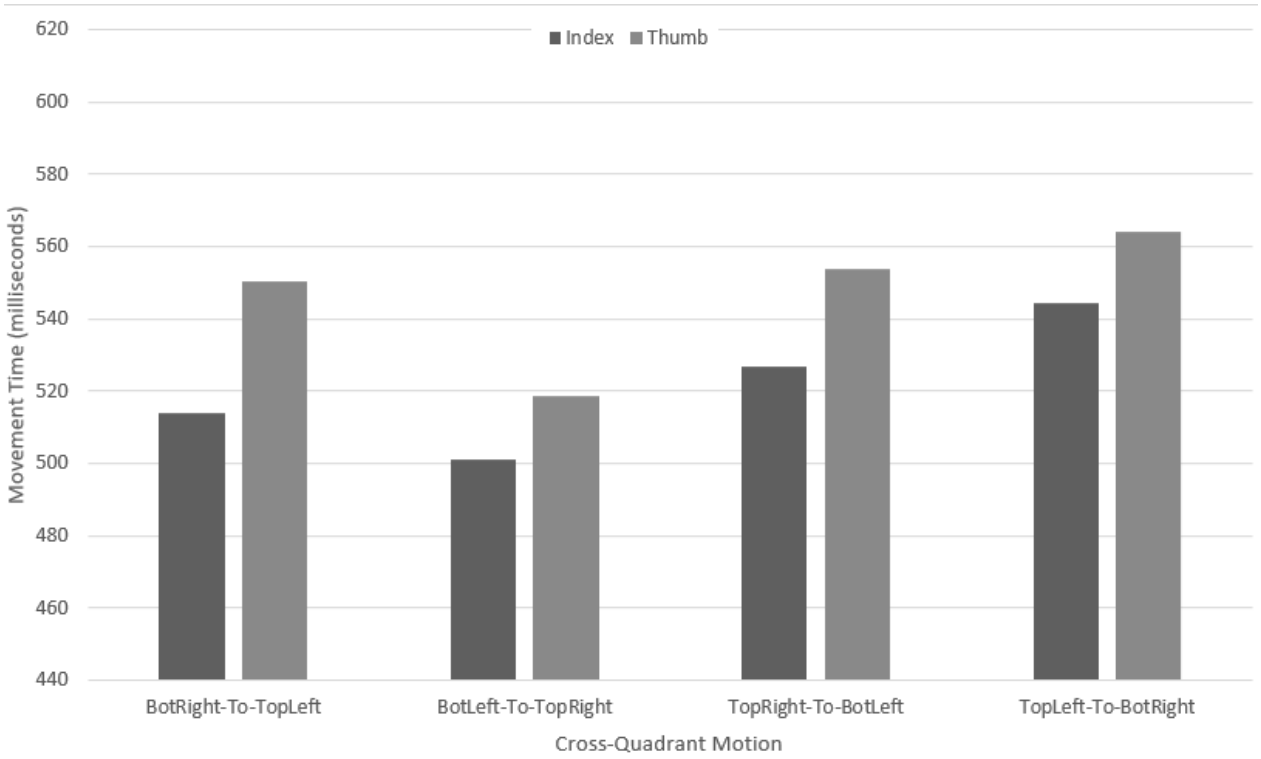


Figure 3.33 – Movement time in milliseconds per cross-quadrant motion, when using either the index finger or thumb. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.32 and Figure 3.33 show that the index finger position had faster or equivalent movement speeds to the thumb, on average, across all selection angles. The thumb did have slower movement times towards the furthest targets at 264°, 288°, and 312°, compared to the other movement angles, as expected (with 288° likely also being influenced by order effects).

For the thumb, going from the bottom-right to the top left of the screen, and vice versa, resulted in slower movement times. The thumb condition had noticeably faster movement times for the bottom-left-to-top-right cross-quadrant, compared to the other quadrants. This may be due to reach.

For the index finger, going from the top of the screen to the bottom of the screen had noticeably slower movement times, compared to going from the bottom to the top of the screen.

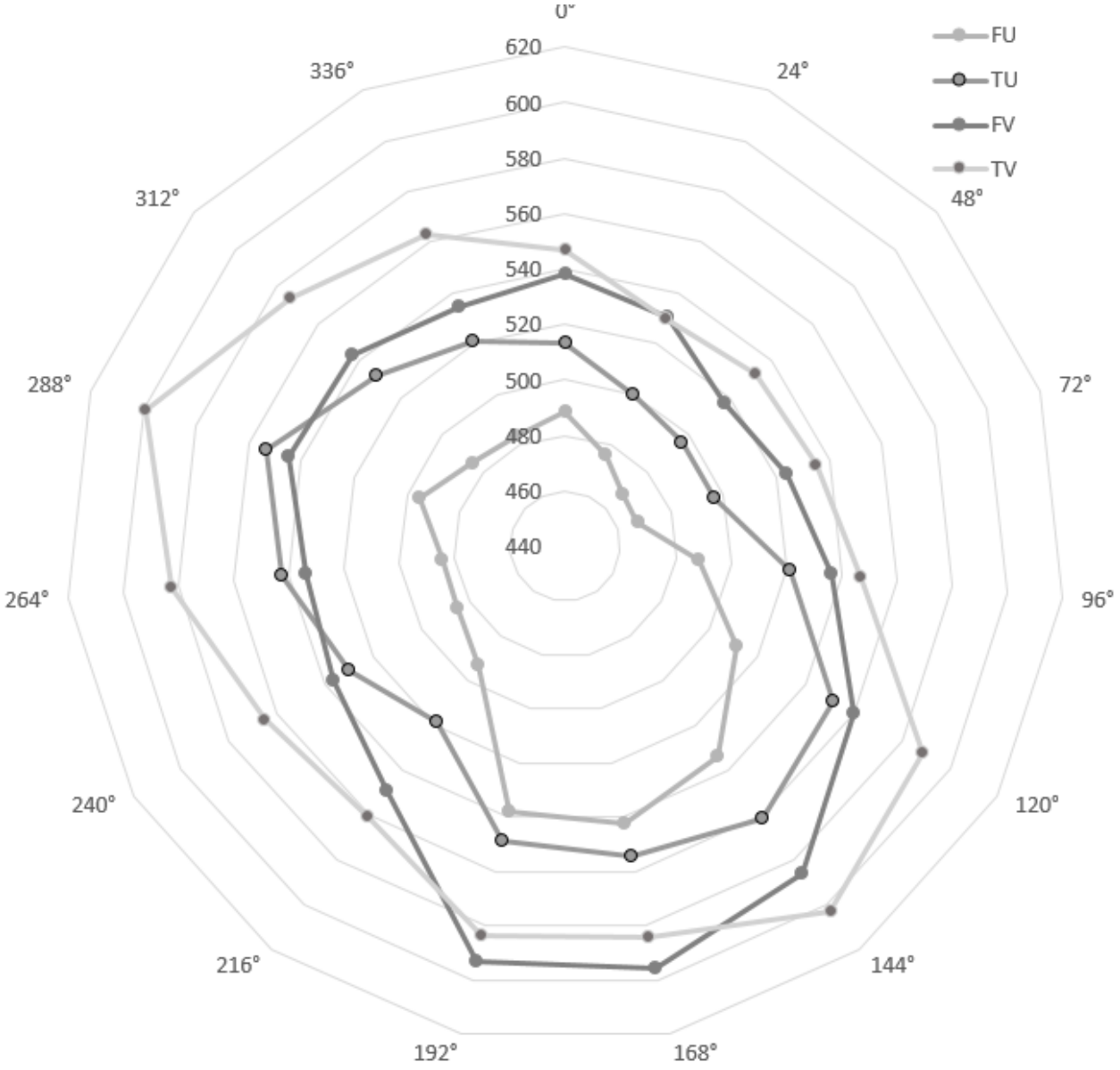


Figure 3.34 – Movement time (in milliseconds) versus selection angle, for the no-vibration and vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands. Note that any increase in movement time at 192°, 288°, 24° and/or 120 may be partially due to order effects.

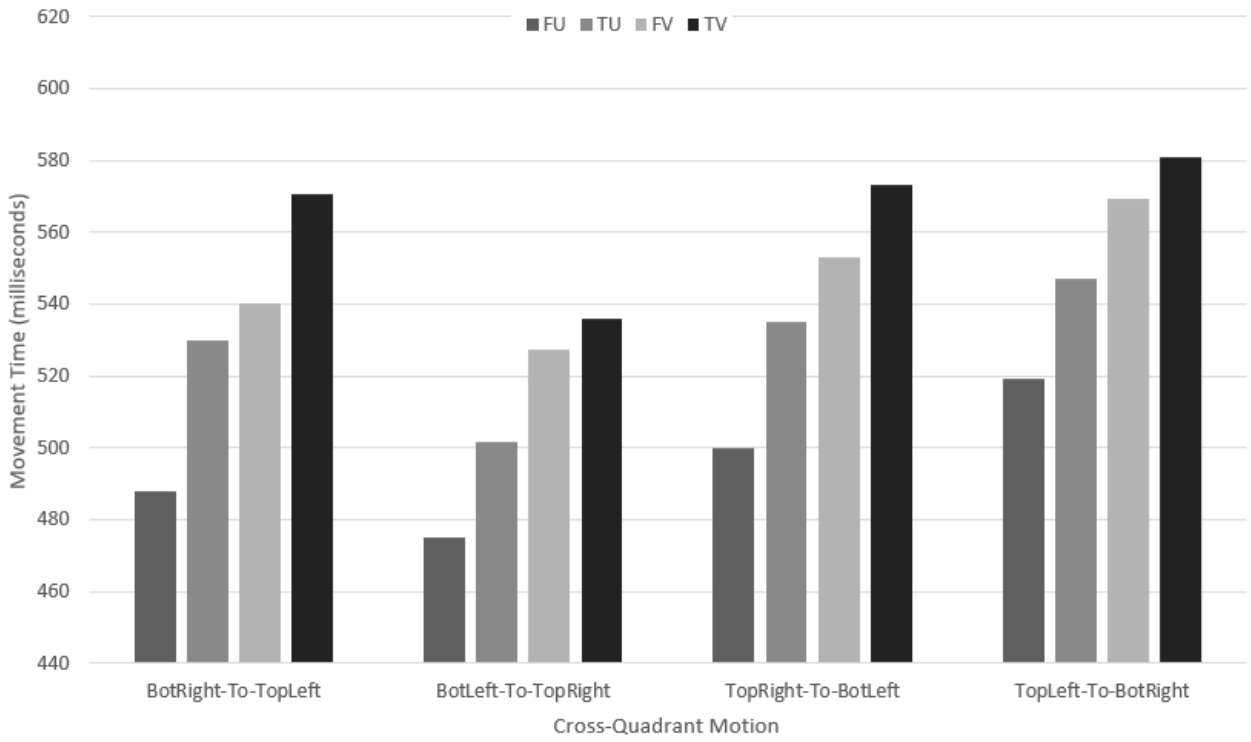


Figure 3.35 – Movement time in milliseconds per cross-quadrant motion, for the Finger x Vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.34 and Figure 3.35 show that the index finger position had faster or nearly equivalent movement speeds to the thumb, on average, across all selection angles, even under vibration.Figure 3.35

3.6.2 Error Rate per Selection Angle

The following graphs show error rate, in percent, versus selection angle. These are average values across target sizes, amplitudes, participants that used their right hands, and the iPad MIP iPad pedestal and medium MFD screens.

Figure 3.36 – Percent error rate versus selection angle, for the index finger and thumb conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands. Note that any increase in error rate at 192°, 288°, 24° and/or 120 may be partially due to order effects.

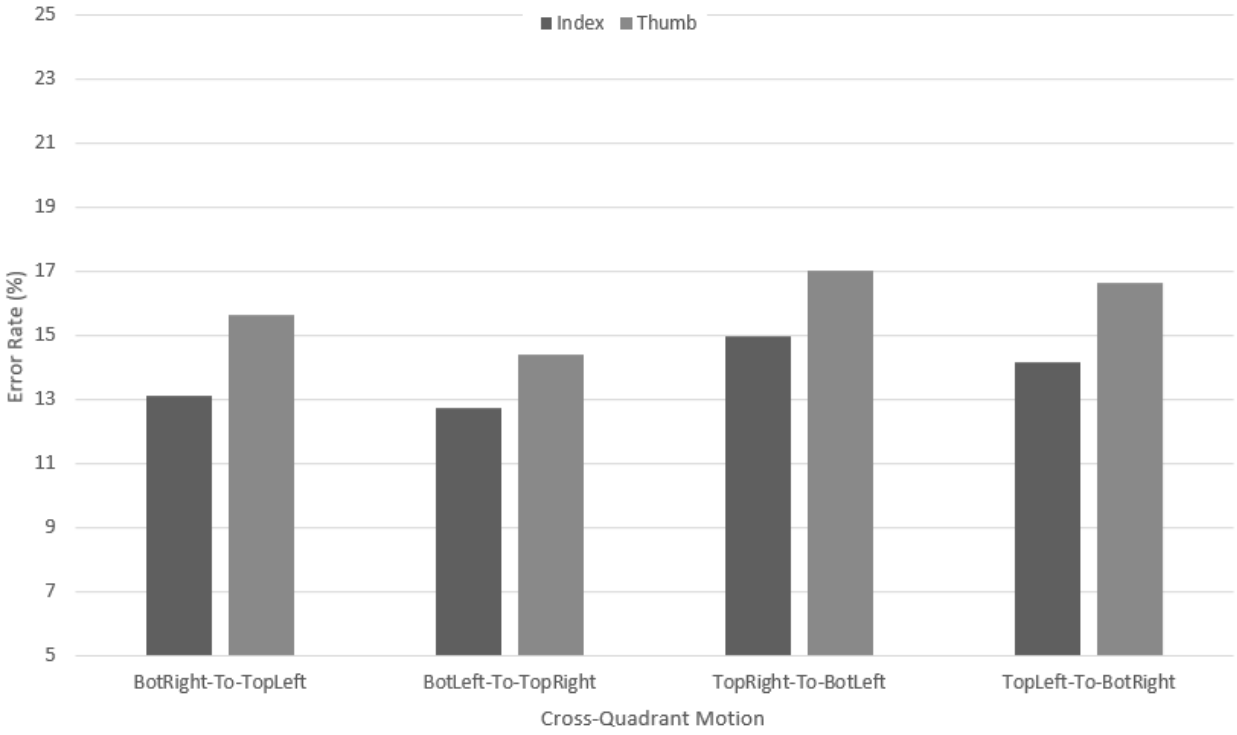


Figure 3.37 – Percent error rate per cross-quadrant motion, when using either the index finger or thumb. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.36 and Figure 3.37 shows that the index finger had a lower error rate than the thumb across all target angles, on average.

The graphs for the index finger and thumb appear similarly shaped.

The index finger had higher error rates when moving from the top of the screen to the bottom of the screen, compared to the inverse motion.

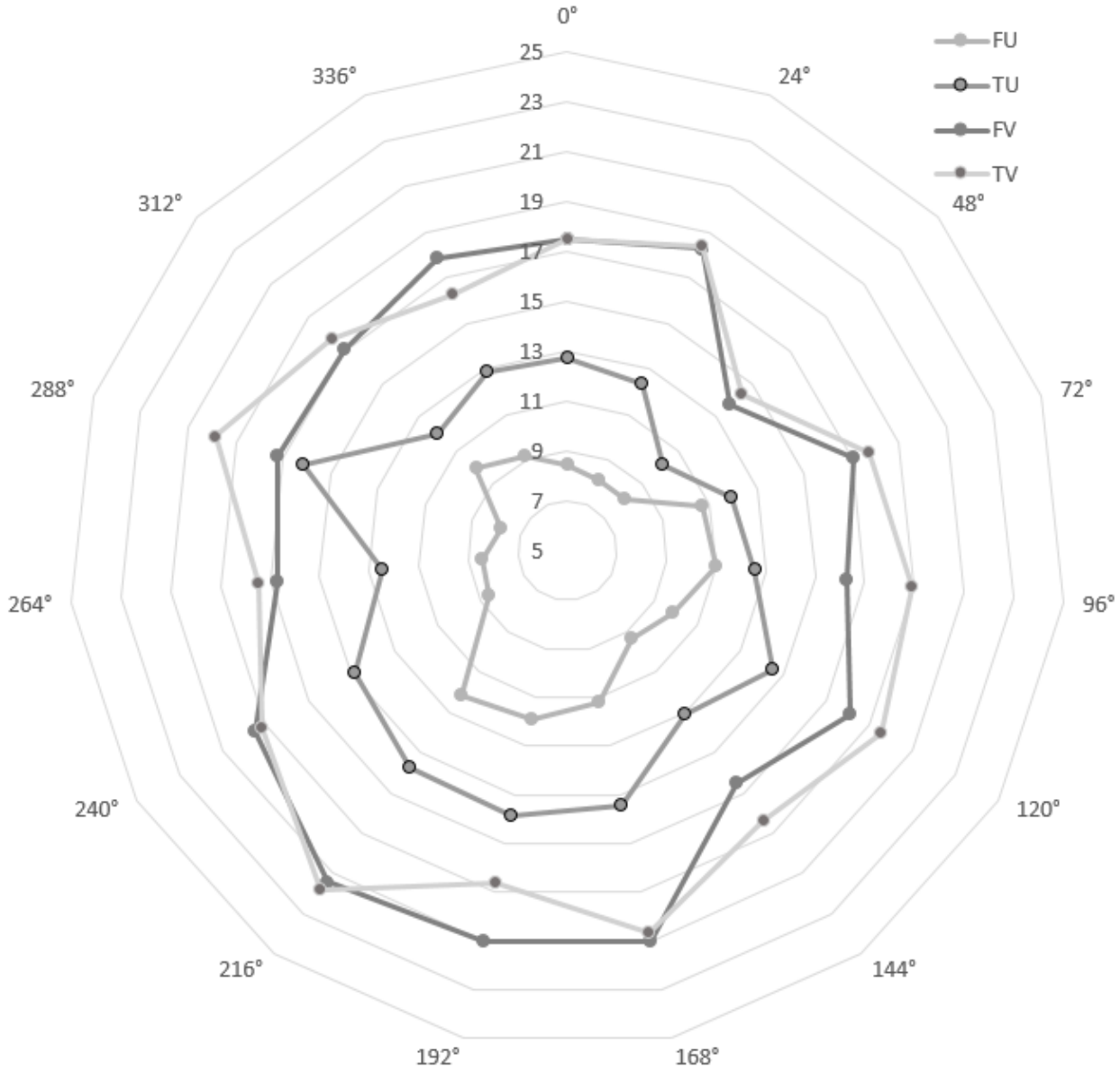


Figure 3.38 – Error rate versus selection angle, for the no-vibration and vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands. Note that any increase in error rate at 192°, 288°, 24° and/or 120 may be partially due to order effects.

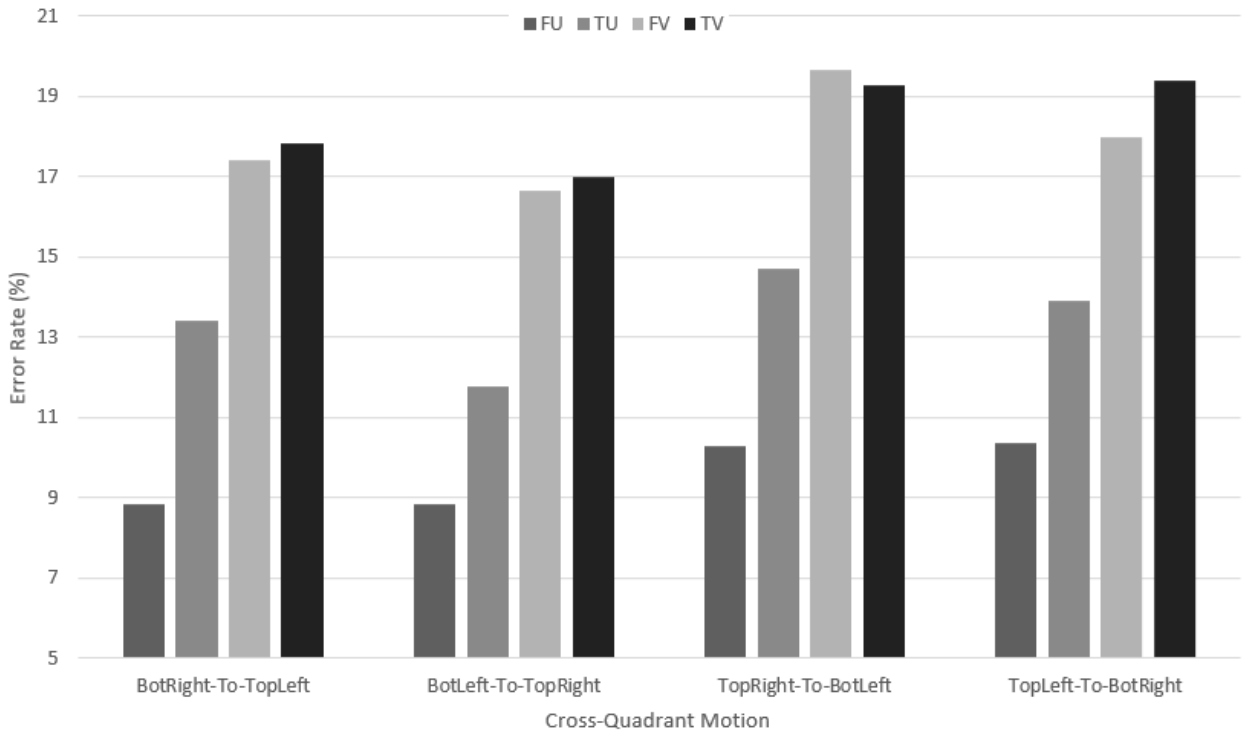


Figure 3.39 – Error rate per cross-quadrant motion, for the Finger x Vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.38 and Figure 3.39 show that, under vibration, the index finger and thumb had similar error rates across selection angles.

3.6.3 dx per Target Angle

The following graphs show dx, in centimeters, versus selection angle. These are average values across target sizes, amplitudes, participants that used their right hands, and the iPad MIP iPad pedestal and medium MFD screens.

These graphs help reveal if participants consistently offset their taps in one direction or the other, from the target. A negative dx value means that the taps were offset towards the center of the screen whereas a positive dx value means that the taps were offset away from the center of the screen.

Note, for the graphs in this subsection, that the negative axis is longer than the positive axes (the graphs are not symmetrical around zero).

Figure 3.40 - dx (in cm) versus target angle, for the index finger and thumb conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

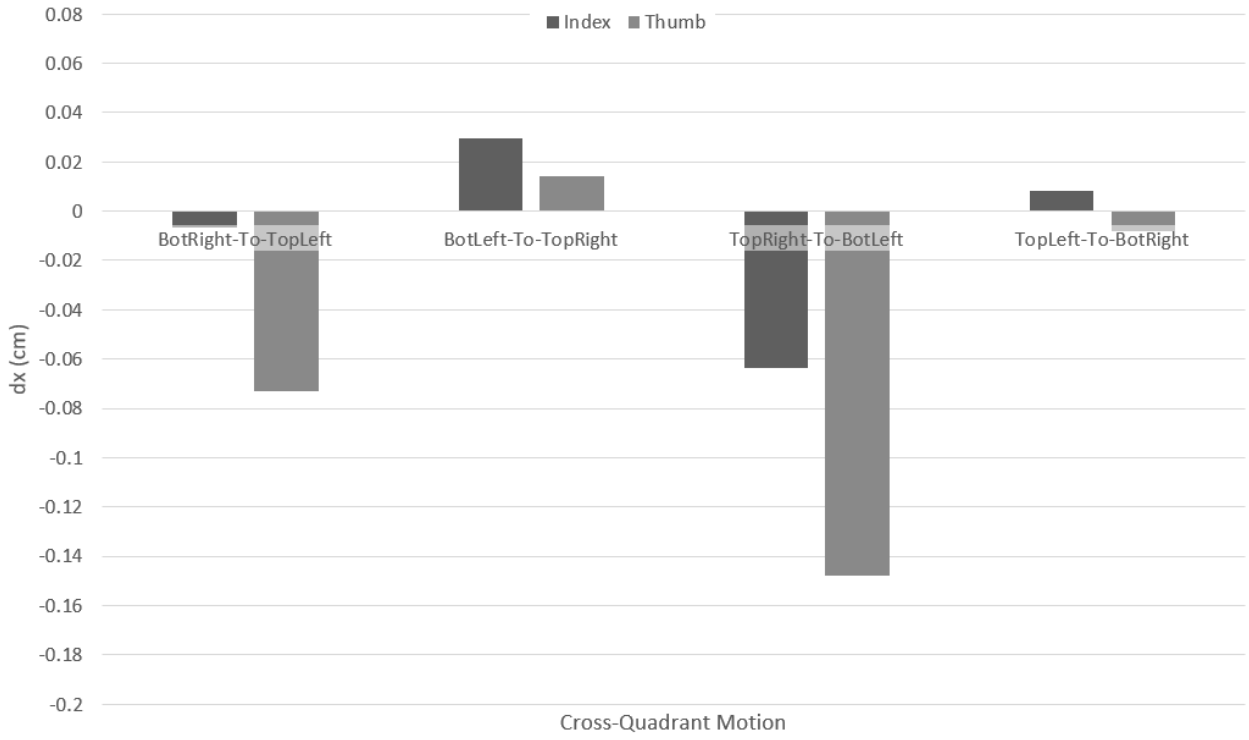


Figure 3.41 – dx (in cm) per cross-quadrant motion, when using either the index finger or thumb. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.40 and Figure 3.41 show that the dx values are as close or closer to zero (less offset) for the index finger, compared to the thumb, across all target angles.

The shape of the graphs for the index finger and thumb appear similarly shaped. There was an average negative dx offset (closer to the center of the screen) when selecting targets towards the left side of the screen, especially targets towards the bottom-left, for both the thumb and index finger. There was an average positive dx offset for targets towards the right side of the screen, for both the index finger and thumb.

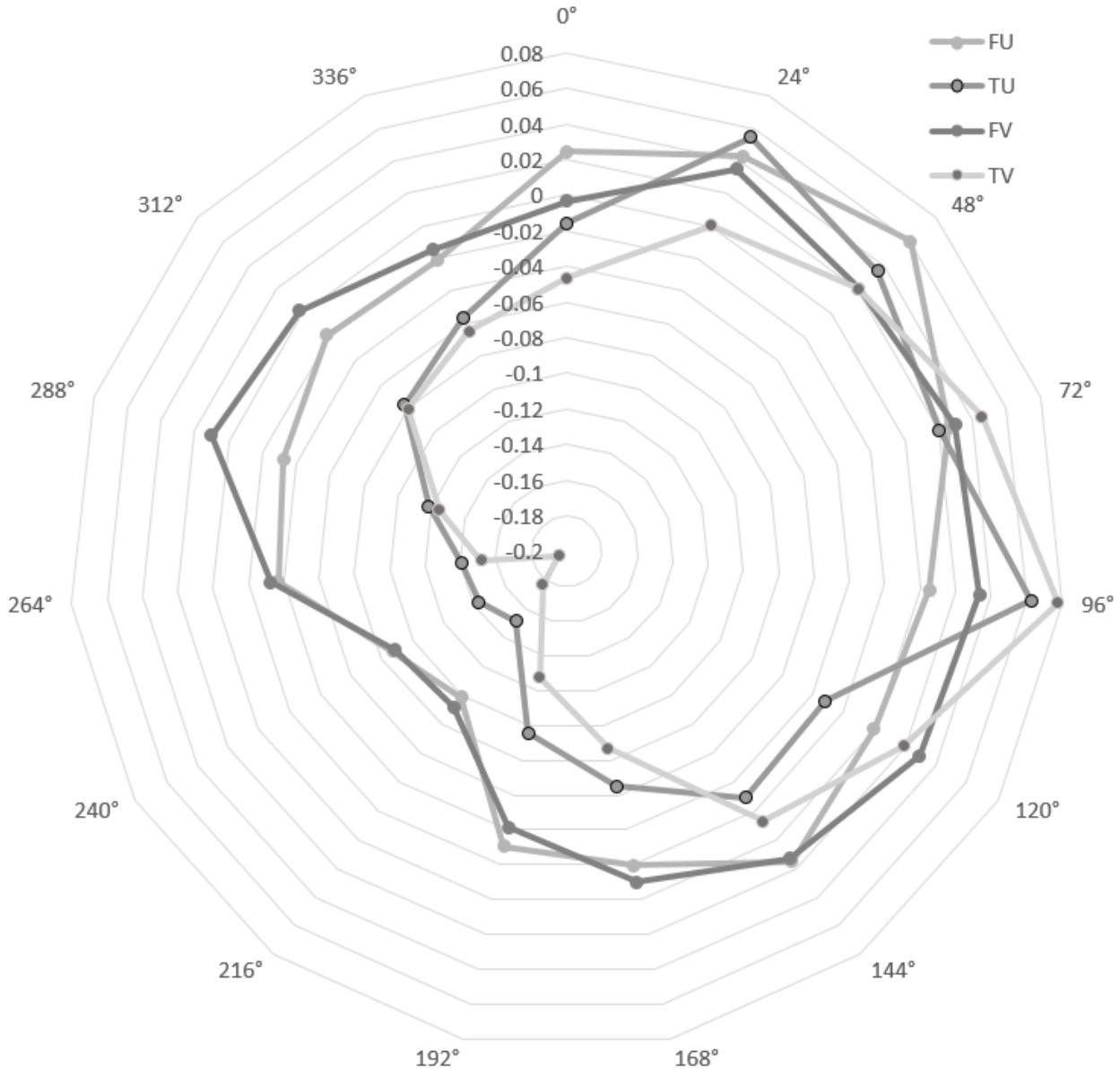


Figure 3.42 - dx (in cm) versus target angle, for the Finger x Vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

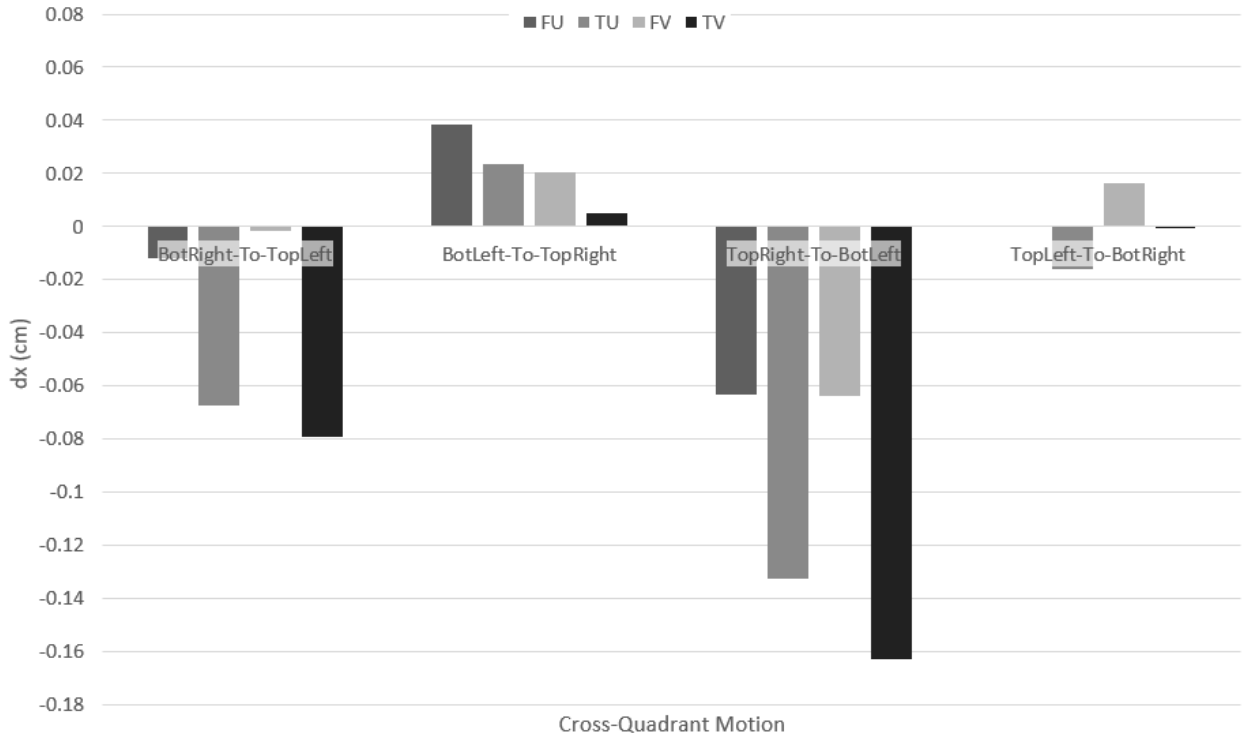


Figure 3.43 – dx (in cm) per cross-quadrant motion, for the Finger x Vibration conditions. The large screen and small MCDU screen have been excluded from this average, as well as any participants that used their left-hands.

Figure 3.42 and Figure 3.43 shows that, under vibration, the dx values are as close or closer to zero (less offset) for the index finger, compared to the thumb, across all target angles.

3.6.4 Summary of Selection Angle Results

Under vibration, the index finger had similar error rates and dx offsets to the thumb across selection angles. It also had similar or faster movement times to the thumb across selection angles, with faster movement times for motions from the bottom-left to top-right, as well as from the top-right to bottom-left of the screen.

The index finger had faster movement time and lower error rates when moving from the bottom to the top of the screen, compared to the inverse motion.

The thumb had faster movement times and lower error rates when moving from the bottom left to top right of the screen, compared to the other cross-quadrants.

For both the index finger and the thumb, there was an average negative dx offset (closer to the center of the screen) when selecting targets towards the left side of the screen, especially targets towards the bottom-left. For both the index finger and the thumb, there was an average positive dx offset for targets towards the right side of the screen.

In addition to reach and biomechanics, the visibility of targets may have played a role in these results. When using the index finger, the bottom targets were more obscured than the top targets, as shown in Figure 3.44. When the index finger was selecting the top targets, the hand tended to obscure the next target, which was towards the bottom of the screen. When the index finger was selecting the bottom targets, the next target, which was towards the top, was clearly visible. As a result, in addition to biomechanical and reach considerations, visibility may have adversely effected target selections towards the bottom of the screen, compared to the top, when using the index finger.

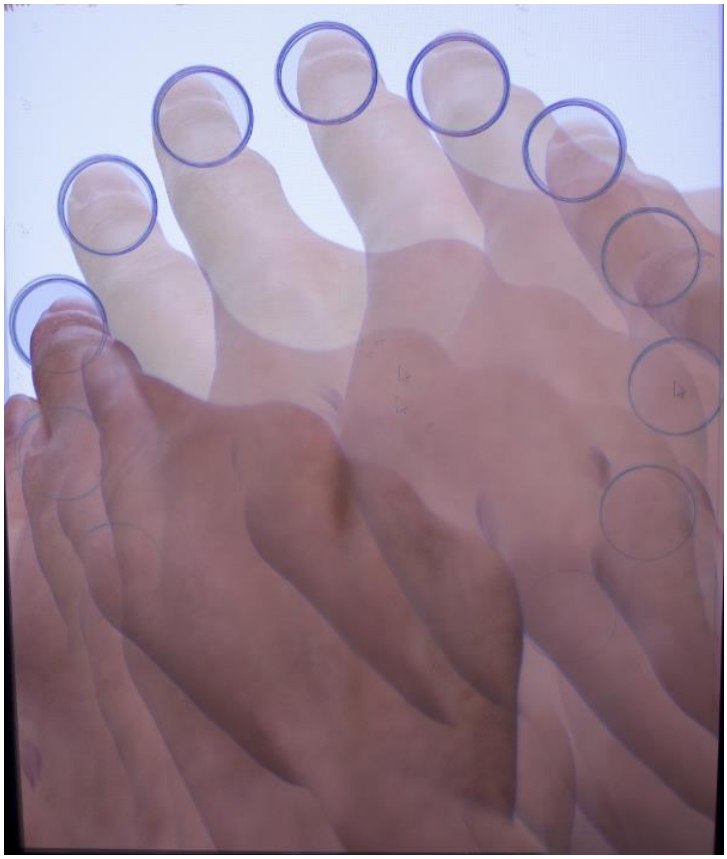


Figure 3.44 – Composite image made by combining 15 images with the index finger touching each of the 15 targets. As can be seen, the bottom targets tend to be obscured, while the top targets are more visible.

When using the thumb, the targets closer to the hand (towards the right, for a right-handed participant) were obscured when selecting targets farther from the hand, as seen in Figure 3.45. However, while selecting targets closer to the hand, the targets far from the hand (towards the left, for a right-handed participant) were clearly visible. As a result, in addition to biomechanical considerations, it is possible that reach adversely effected target selections towards the left side of the screen, while visibility adversely effected some of the target selections towards the right side of the screen (for right-handed participants).

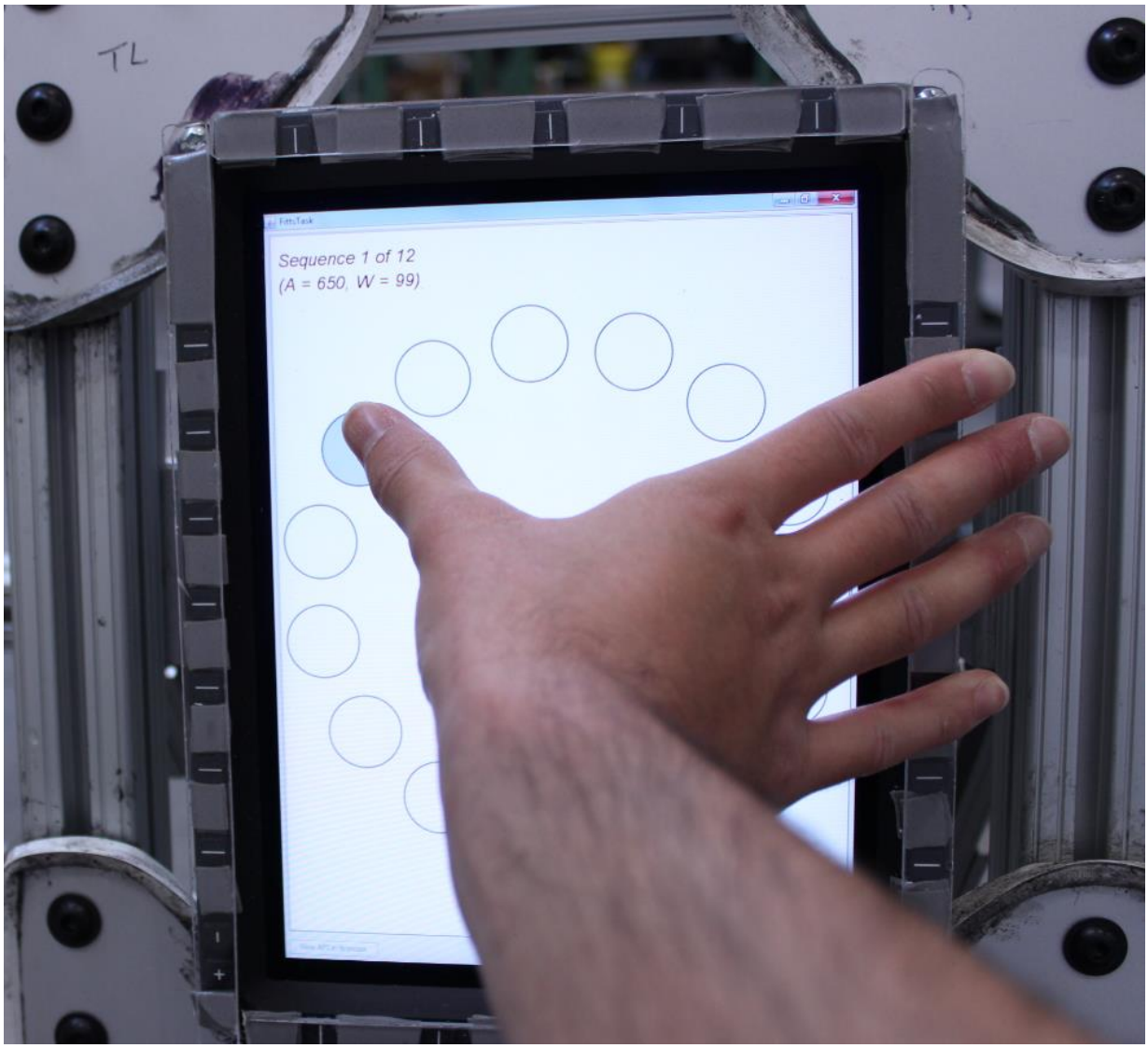


Figure 3.45 – In the hand-support (thumb) condition, when reaching for a far target, the hand often hides the next target, which is located closer to the hand.

3.7 Questionnaires

3.7.1 Debriefing Questionnaire

None of the questionnaires answers provided statistically significant results, as shown in the table of χ^2 results below. Despite this, the questionnaire results are presented here for completeness, and the additional insight they may bring.

Table 3.21 - χ^2 results for the questions on the debriefing questionnaire.

Question	χ^2 Result
Difficulty impact of vibration (slightly more, moderately more, much more)	$\chi^2_{(2,N=22)}=3.36$, $p>.05$
Did hand support help or hinder you? (Helped/Hindered/Sometimes Helped)	$\chi^2_{(2,N=22)}=0.64$, $p>.05$
Most liked screen (iPad, large touch monitor, medium MFD screen, small MCDU screen)	$\chi^2_{(3,N=21)}=2.05$, $p>.05$
Least liked screen (iPad, large touch monitor, medium MFD screen, small MCDU screen)	$\chi^2_{(3,N=21)}=4.71$, $p>.05$
Preferred position (MIP or Pedestal)	$\chi^2_{(1,N=17)}=.06$, $p>.05$

Two participants did not answer any questions. One participant did not specify which screen they most liked. Another participant did not specify which screen they most disliked.

The first five participants originally tried the medium MFD screen in both MIP and pedestal positions, and answered the questions based on that experience. The medium MFD screen did not fit properly in the pedestal position, and was too far to the side. Based on their experience with the medium MFD screen, which was too far, all five of these participants preferred the MIP position to the pedestal position. These five participants then came back to the lab, to redo the pedestal conditions on the iPad. However, they were not asked whether their preference for screen position had changed, based on using the iPad instead of the medium MFD screen. As a result, the preferences of these five participants have been excluded from the preferred position analysis.

Five of the participants had a pre-existing bias towards preferring the hand-support (thumb) condition. Four out of the five claimed that it helped, with one out of the five claiming that it

did not help. Their preferences have been included. However, note that the results for hand-support preference may be skewed by this bias.

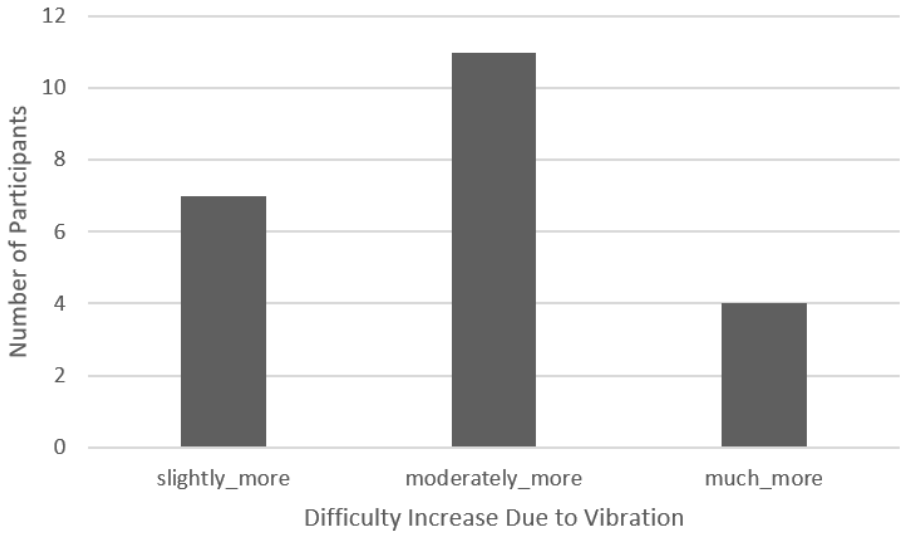


Figure 3.46 - The number of participants who answered either slightly more, moderately more or much more to the question: “How did vibration impact your ability to complete the task?”

The interpretation of “slightly more difficult”, “moderately more difficult” and “much more difficult” appeared to be particularly subjective, with different participants having differing interpretations. For those participants who explicitly asked, they were told that “no more difficult” and “less difficult” were also acceptable responses, but none of the participants who asked ended up selecting these options.

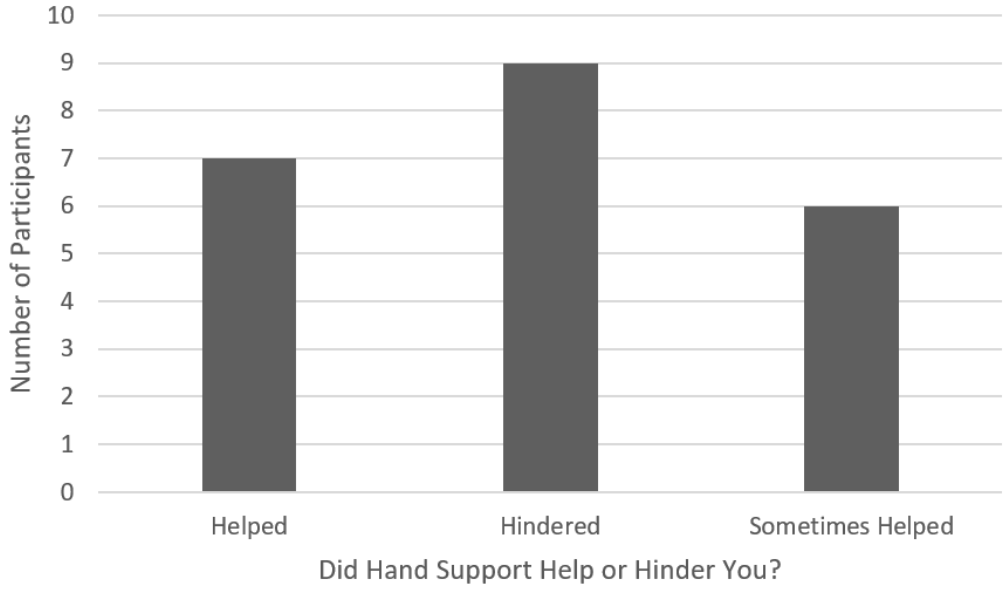


Figure 3.47 – The number of participants who answered “helped”, “hindered”, or “sometimes helped” to the question of whether the hand support (thumb condition) helped or hindered them.

Those who replied that the hand-support helped felt that it helped on some screens, but not others. Of the participants who said that the hand support sometimes helped, many of them mentioned that it did not help on the iPad. Some people found it helped on the small MCDU screen only. Some people found that it helped on the medium MFD screen only. Some people found that it helped on both the small MCDU screen and medium MFD screen.

Some people preferred holding the screen from the top, as was done on the small MCDU screen. Some people preferred holding the screen from the side, as was done on all the other screens. Many people mentioned that, when holding the screen from the top, their vision of the targets was obscured, and they needed to shift their head to the side, to be able to see around their hand.

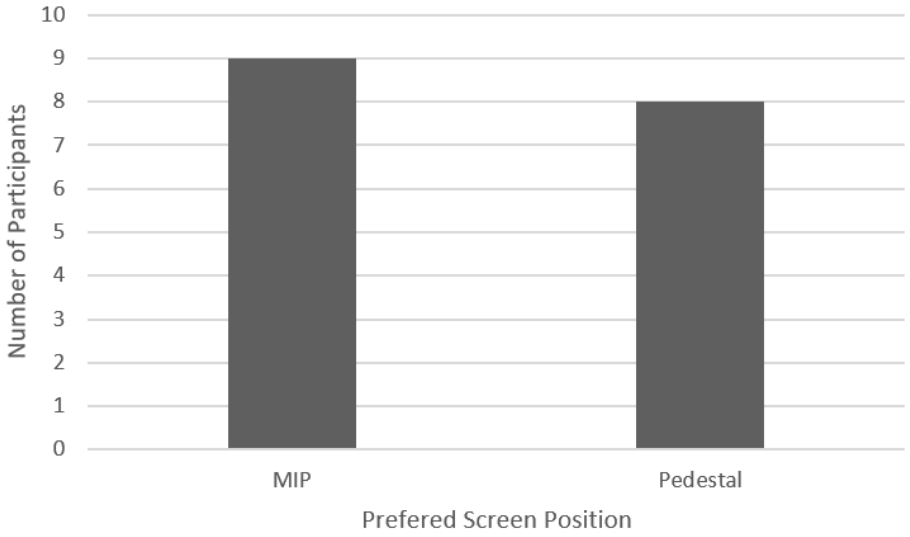


Figure 3.48 – The number of participants who responded “MIP” or “pedestal” to the question: “Which screen position was the easiest to use?”

Many of the participants who preferred the pedestal position mentioned that the MIP position as more fatiguing on their arm. Many of the participants who preferred the MIP position liked that it was directly in front of them. They found the pedestal position more fatiguing on their neck, and didn’t like having to look over to the side and down.

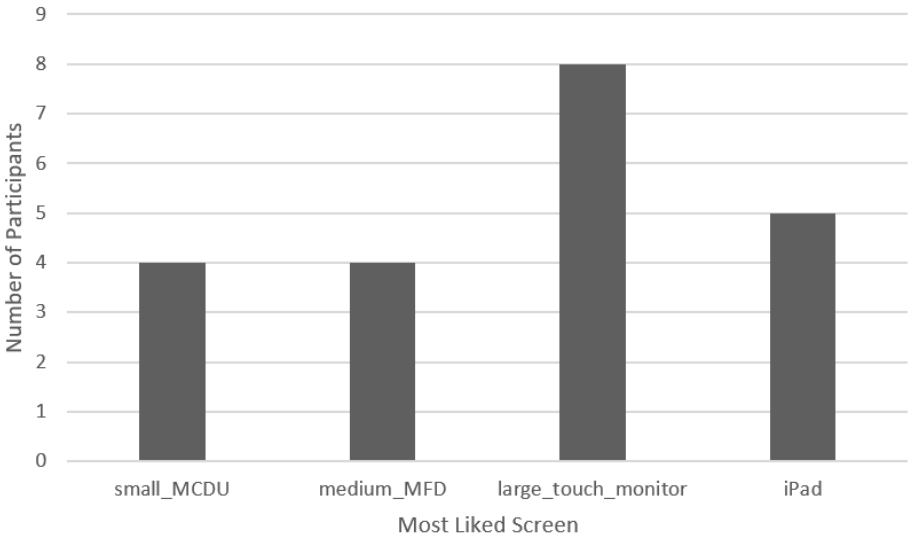


Figure 3.49 – The number of participants who responded with the “small MCDU screen”, “medium MFD screen”, “large touch monitor”, or “iPad” to the question: “Which of the screens did you like using the most?”

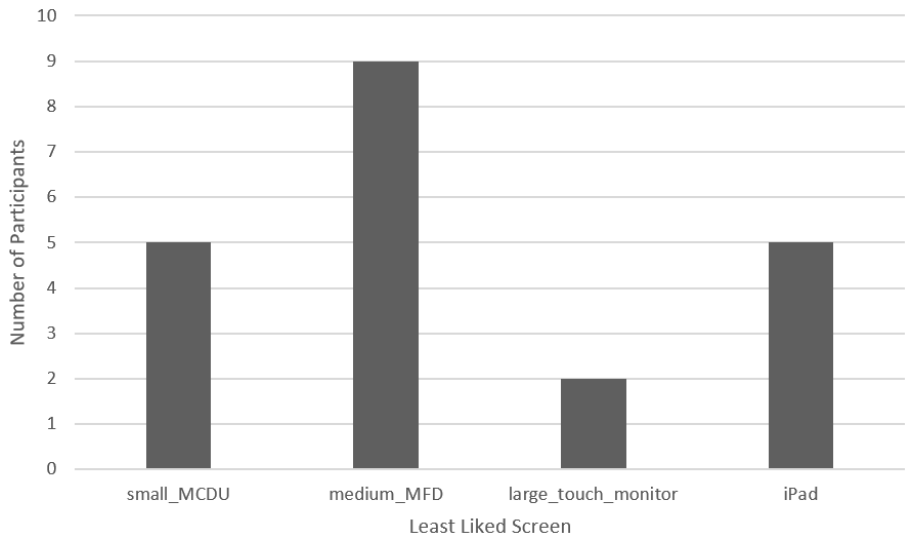


Figure 3.50 - The number of participants who responded with the "small MCDU screen", "medium MFD screen", "large touch monitor", or "iPad" to the question: "Which of the screens did you most dislike using?"

For the question of "Which of the screens did you most dislike using?", many of the participants who chose the medium MFD screen spoke about its lack of responsiveness, and the fact that it occasionally did not register their clicks, which they found frustrating. For those who chose the iPad, some participants mentioned that they did not like the way the selection task was presented on the iPad as much, versus how it was presented on the other screens. Other participants mentioned that the iPad seemed too sensitive, for them. Some participants did not like having to grab behind the screen, for the hand-support condition. Those who disliked the large touch monitor found it too large. Several of those who disliked the small MCDU screen found it too small, and were frustrated by the very small size of the "OK" button, to continue to the next trial on GoFitts, which was very difficult to click.

For the question of "Which of the screens did you like using the most?", some of the participants who chose the iPad cited its familiarity to them and its quick responsiveness for why they liked it. Several of those who liked the medium screen felt that it provided a better hand-support in the thumb condition. Other liked it because it was a real avionic touchscreen. Those who liked the large touch monitor found it very responsive. Some people noted that they liked the tactile way it felt, when clicking it (absorbed taps a bit differently than the other screens. It had a bit of give to it, when clicked). Those who liked the small MCDU screen also found it very

responsive, and liked its small size. Some of those who liked the small MCDU screen or the medium MFD screen liked that they were not too sensitive.

3.7.2 Comfort Questionnaire

The first five participants have been excluded from these questionnaires, since they answered the questions while using the medium MFD screen in both conditions, rather than the iPad (these participants then came back to the lab, to use the iPad in the pedestal position, but were not asked to redo the comfort questionnaire). For two of the participants, their questionnaire on one of the four conditions was missing. As a result, the comfort questionnaire results from these two participants were also excluded. In summary, the comfort questionnaire results from only 17 of the 24 participants were retained.

Due to the relatively low number of participants that could be included for the comfort questionnaire results and the relatively minor differences observed in the graphs below, an attempt has not been made to perform a statistical analysis on the results. The results are presented here for completeness. For the following graphs, the median rating for each condition is shown. Note that a high value denotes higher comfort (better) and a low value denotes lower comfort (worse). All ratings were measured on a Likert scale from 1 to 7.

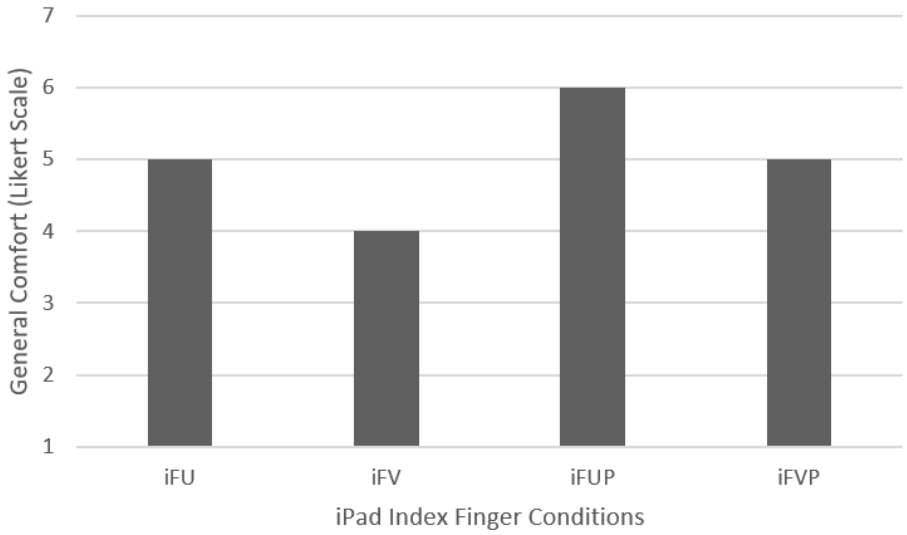


Figure 3.51 – The median general comfort Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition. iFU denotes the iPad in the MIP position, without vibration. iFV denotes the iPad in the MIP position, with

vibration. iFUP denotes the iPad in the pedestal position, without vibration. iFVP denotes the iPad in the pedestal position, with vibration.

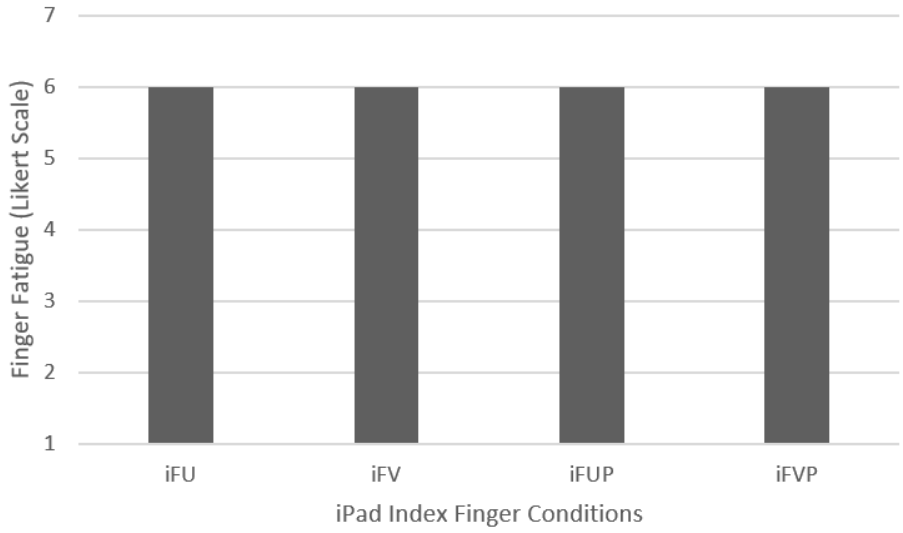


Figure 3.52 - The median finger fatigue Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition.

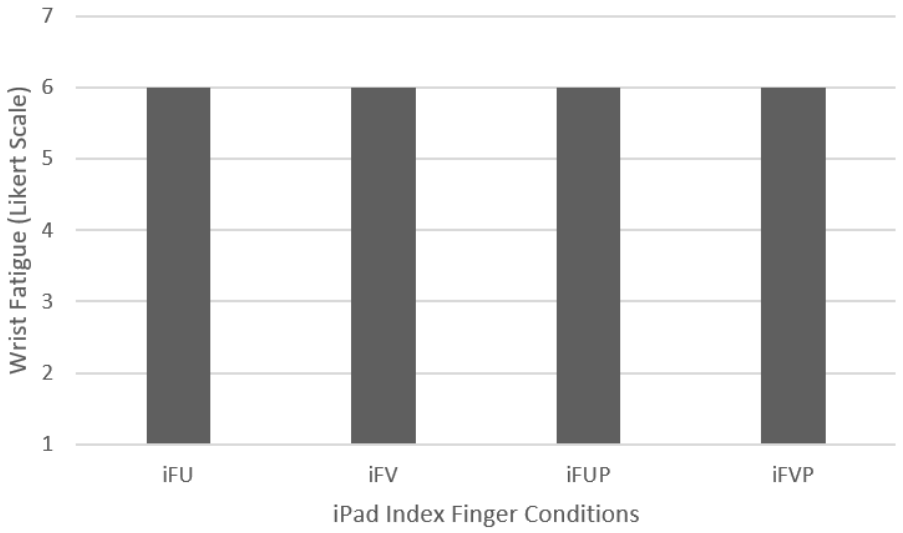


Figure 3.53 - The median wrist fatigue Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition.

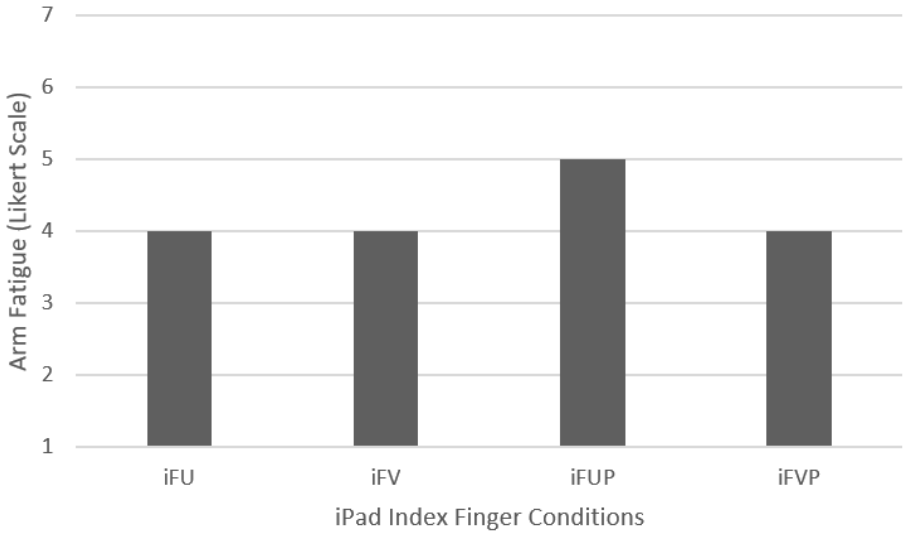


Figure 3.54 - The median arm fatigue Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition.

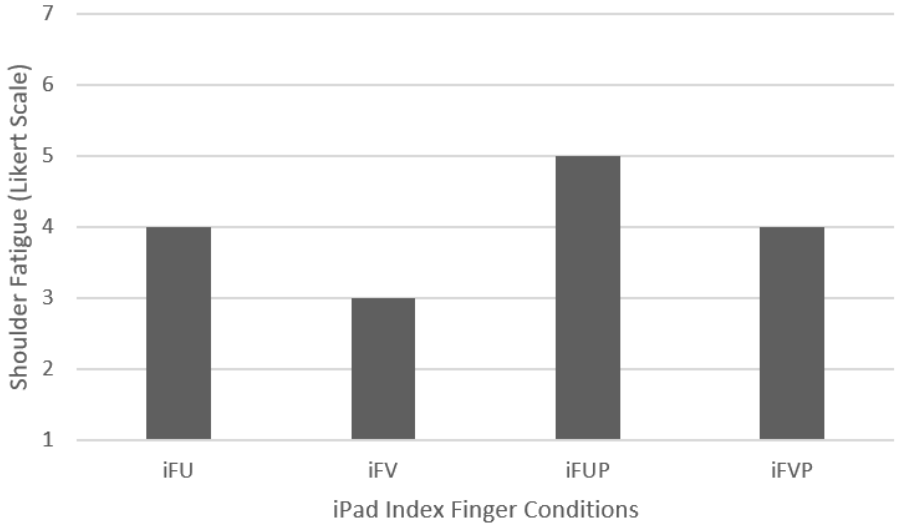


Figure 3.55 - The median shoulder fatigue Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition.

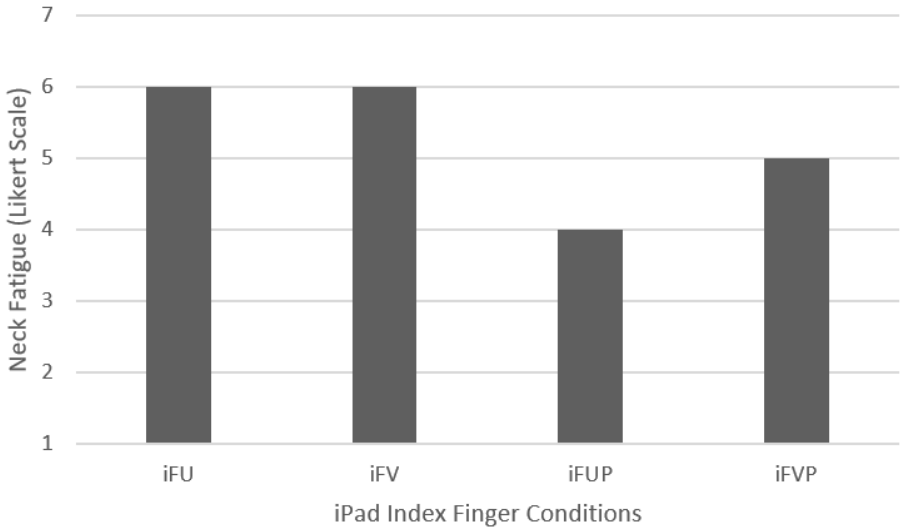


Figure 3.56 - The median neck fatigue Likert rating (a higher value denotes higher comfort, while a lower value denotes lower comfort) in each iPad index finger condition.

As mentioned previously, during the debriefing questionnaire, several participants mentioned that the MIP position caused more arm and shoulder fatigue, whereas the pedestal position was a bit more fatiguing on the neck and required a slightly worse posture, since it required looking down and to the side.

3.8 Additional Observations

While using the hand support, the participants tended to rest two of their fingers on the side of the screen, while extending their thumb to reach far targets. As a result, having a deeper flat edge, like on the small MCDU and medium MFD screen seemed to provide a better hand support, versus the rounded, thin edge of the iPad.

Like many avionic touchscreens, the MFD screen used in this study had bezel keys. In the case of this MFD screen, the keys protrude outwards from the bezel, as shown in Figure 3.57. We ran a pre-test, just among the research investigators, to go through the experimental procedure before inviting participants. During this pretest, in the hand-support (thumb) condition, we accidentally clicked in and activated these bezel keys several times, while reaching for targets. Some of these bezel keys swapped video input to the display. Hence, when they were pressed, the screen's connection to the computer running GoFitts was broken, and we could not continue

the task until the connection was re-established. After this happened several times in a short period of time, we decided that we needed to attach a barrier over the keys to prevent them from being accidentally clicked. This barrier, covering the bezel keys, is shown in Figure 3.58.

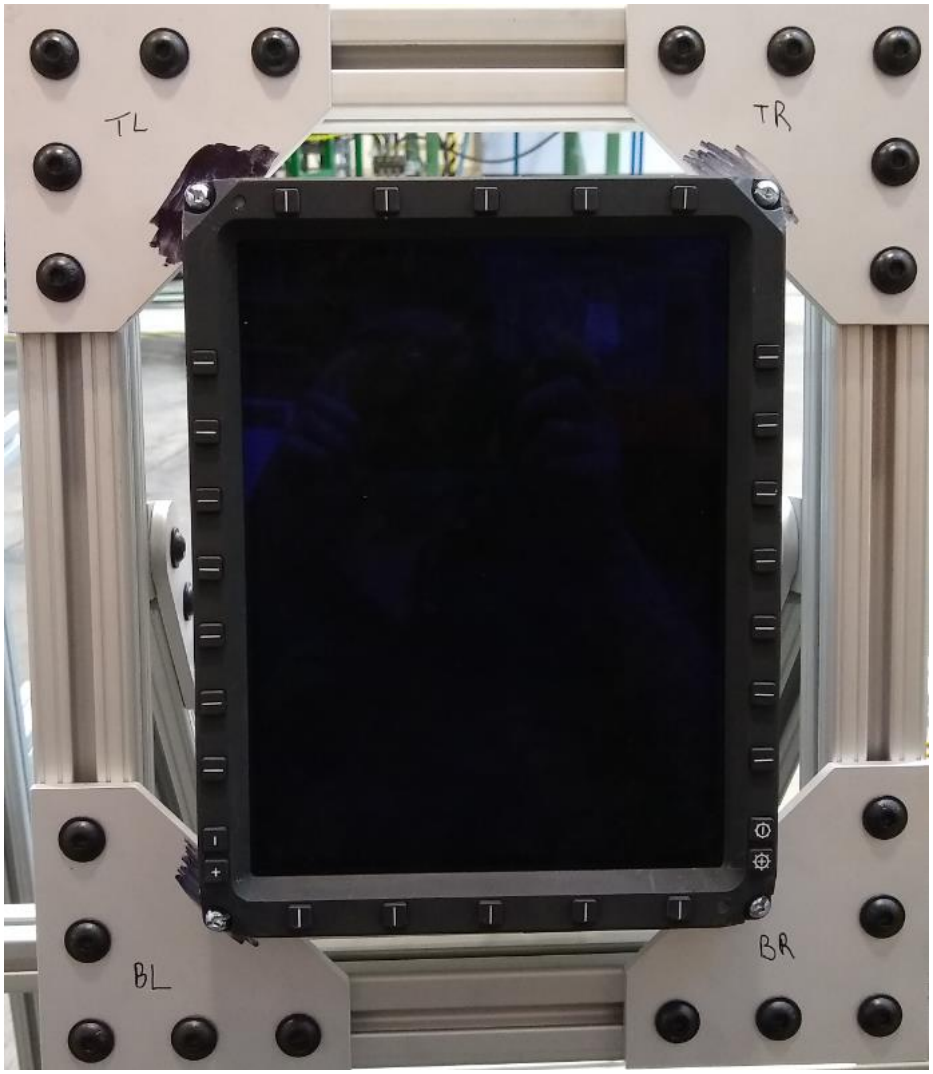


Figure 3.57 – When using the bezel edge as a hand support, the protruding bezel keys of the MFD screen were accidentally clicked, on several occasions, while reaching for a target.

Figure 3.58 – A barrier was affixed to the bezel, in order to cover the bezel keys, to prevent them from being accidentally activated in the hand-support condition.

As mentioned previously, participants had trouble reaching the furthest targets, in the thumb (hand-support condition).

Some participants naturally tried to use their knees as an arm support, especially in the index-vibration condition and especially as they became more tired over time. There was a structural support bar directly under the chair, and some participants naturally wanted to rest their feet on it and bring their knees higher. They then tried to use their knees to support their arm, in the index-vibration condition. They were discouraged from doing this, and told that their arm should not be supported. However, this may be a viable strategy on some aircraft. In some aircraft, the layout of the chair (which can sometimes be low and tilted back) and the screens (which are

sometimes much lower than in this study) could allow pilots to rest their arm on their knee, while using the touchscreen.

Some participants tried to support their arm with their other hand, especially in the index-vibration condition, and especially as they became more tired over time.

The posture of the participants tended to change over time, especially as they became more tired. Some participants tended to slouch more when using the pedestal screen.

Participants tended to lean forward to use the screens, especially under vibration. This was especially true for participants that had trouble selecting the targets, some of whom leaned forward to a significant degree.

Some participants reported feeling fatigued after prolonged use of the touchscreens, especially under vibration. They reported that both the index finger (freehand) and thumb (hand-support) conditions were fatiguing over time.

Participants tended to become much more familiar with the task and target sequence over time. They tended to select targets more quickly by the end of the experiment, and were slower towards the beginning. The participants who started off selecting targets quickly tended to maintain a fast pace throughout. The participants who started off selecting targets slowly tended to speed up much more over time (this is a general impression; we have not objectively quantified this to confirm it).

As mentioned previously, none of the screens had visual feedback to denote an error. However, the small MCDU screen, medium MFD screen and large touch monitor were hooked up to a computer as touch monitors, with a finger tap being treated the same way as a left-mouse click. As a result, the mouse cursor moved to the position of each finger tap, providing feedback for where the tap was registered. In addition, the large touch monitor showed a subtle ripple effect, for each tap: a transparent circle appeared, centered on the finger, and expanded in size while decreasing in opacity. This transparent circle was larger than the finger, and so was not hidden by it.

When using the small MCDU screens, participants were asked to hold the screen from the top, rather than the sides, due to limitation in how the frame was built, for this screen. Some participants found this more comfortable, while other participants found it less comfortable and

more awkward. In general, holding the screen from the top seemed to result in poorer visibility of the targets. The participants needed to shift their head to the side, to look around their hand.

CHAPTER 4 DISCUSSION AND LIMITATIONS

This section reviews the objectives of this study, then it summarizes the main findings per objective and compares them with previous studies. It highlights the implications for the design of avionic interfaces and concludes with the limitations and future research questions.

The goals of this study, as specified in the introduction, were:

1) Quantify the impact of vibration on one-handed target selection task performance by providing throughput values measured in controlled, representative helicopter cockpit conditions. Use the "multi-direction pointing task" recommended in the ISO 9241-411 standard and the methods recommended by Soukoreff and MacKenzie to measure and calculate throughput [18, 24].

2) Compare the effectiveness of using the edge of the screen as a hand support, versus using the touchscreen without any support, for a 2D Fitts' selection task. Compare these results against those of Hourlier and Servantie [15].

3) Compare selection task performance on real avionic touchscreens of representative sizes, versus commercial touchscreens.

4) Compare selection task performance for touchscreens located at two positions: Main Instrument Panel (MIP) position (directly in front of the pilot and at a vertical incline) versus the pedestal position (beside the pilot and at a horizontal incline). Compare these results against Coutts et al., who measured selection task speed and accuracy in different screen positions, but did not provide throughput values [16].

5) Confirm the impact of target size on error rate under vibration and compare the results against those of Avsar [44], Avsar et al. [45], and Coutts et al. [16].

4.1 Quantify the Effects of Vibration on Selection Task Performance, Using a Standardized Methodology

Vibration had a significant impact on task throughput. On average, across all screens and hand-support (finger) conditions, throughput was 6.5 bits/second in static conditions, versus 5.7

bits/second under vibration. As a result, on average, throughput was around 13% lower under the vibration condition tested here, when compared against the static condition.

Looking at the error rate for these same two averaged conditions, there was an error rate of 10.3% in static conditions, versus 16.6% with vibration. Hence the error rate went up by a factor of 1.6 times, due to vibration.

When considering just the index finger (no-hand-support condition, excluding the large touch monitor), the average throughput was 6.8 bits/second in static conditions, versus 5.7 bits/second under vibration. As a result, there was around a 16% decrease in performance under vibration, on average.

When looking at the error rate for these same two averaged conditions, there was an error rate of 9.0% in static conditions, versus 16.7% with vibration, when using the index finger unsupported. This is a 1.9 times increase in error rate.

When considering just the thumb (hand-support condition, excluding the large touch monitor), the average throughput was 6.1 bits/second in static conditions, versus 5.5 bits/second under vibration. As a result, there was around a 10% decrease in performance under vibration, on average.

When looking at the error rate for these same two averaged conditions, there was an error rate of 12.3% without vibration, versus 17.0% with vibration, when using the thumb with the hand supported on the screen's bezel edge. This is a 1.4 times increase in error rate.

Since we have used a standardized methodology and combined speed-accuracy value for our study, our results can be compared to those in other studies that used the same or similar methodology.

The touchscreen throughput value we found under vibration, of 5.7 bits/second, is higher than the throughput values for other types of common input devices, measured in static conditions, given by Soukoreff and MacKenzie [18]. Soukoreff and MacKenize gave throughput values of between 3.7-4.9 bits/seconds for a computer mouse in static conditions. In the context of aviation, Doyon-Poulin and Routhier [74] found a throughput value of 1.9 bits/second for a cursor control device in static conditions, positioned in a representative cockpit layout. Hence their throughput value for a cursor control device, under static conditions, was 3 times less than

the throughput value we have found in this study for a touchscreen under vibration conditions. Letsu-Dake et al. [75] reported throughput values of 1.3 bits/second when using a cursor control device under turbulent conditions (so around 4.5 times less than the throughput value we have reported for touchscreens under vibration), although their vibration level appears to have been higher than the one used in our study.

We found average error rates of 10.3% in static conditions, versus 16.6% under vibration. Lin et al. [8] reported an average error rate of 7.3% for touchscreens, averaged across both static and vibration conditions. However, they used lower vibration levels, and larger average target sizes (they used a target size range of 1-2.5 cm, whereas we used a range from 0.8-2 cm). In addition, their screens were likely closer to the participants, since they were using an office desk layout, and they had younger participants, on average. For a mouse and trackball, Lin et al. reported average error rates of less than 2%, across all vibration levels. Lin et al. found that error rate increased faster for the touchscreens, compared to the mouse and trackball, with increasing vibration and decreasing target sizes.

Letsu-Dake et al. [75] found error rates of 1.7% to 5.6% when using a cursor control device under vibration levels higher than those used in our study, with this error rate range including conditions where participants used their non-dominant hands. Although Letsu-Dake et al. do not specify their target size directly, they appear to have used a target size that was in the range of 0.8-1 cm. In our study, we found error rates, under vibration and averaged across all screen and finger conditions, of 34.2% and 21.2%, when using target sizes of 0.8 cm and 1 cm respectively. As a result, it appears as though the error rate on touchscreens may be much higher than when using a cursor control device, especially under vibration conditions, and especially for small target sizes.

4.2 Compare the Effectiveness of Using the Bezel Edge as a Hand Support, Versus a Freehand Baseline

We found no clear benefit to using a hand-support under vibration for the ISO standard multi-directional pointing task [24], while it provided a clear hindrance in the static condition. This applied for both throughput and error rate. It applied across all target sizes and amplitudes.

Our experiment was not designed to investigate the impact of selection angle. Nevertheless, we performed a validation analysis to compare average movement time, error rate and one-dimensional offset from the target center (dx) per movement angle. This analysis showed no clear benefit from the hand-support, across all movement angles.

Under vibration, the throughput was an average of 5.7 bits/second when using the index finger unsupported (excluding the large touch monitor, since it could not be tested in both hand-support conditions). It was an average of 5.5 bits/second when using the thumb (supported). Under vibration, the error rate was an average of 16.7% with the index finger unsupported. It was an average of 17.0% with the thumb (hand supported on the bezel edge).

This result is in contrast to the SAE ARP60494 standard, which recommends using the bezel edge as a hand support [12]. It is in contrast to the qualitative observations of Coutts et al. [16], who also looked at a multi-directional pointing task but did not control for hand-support and did not explicitly test it against a freehand baseline.

Cockburn et al. [13] also qualitatively observed that the hand-support was helpful to the participants, in contrast to our results, without controlling for hand-support and without explicitly testing it against a freehand baseline. Cockburn et al. conducted a 2D target selection task, but their task design did not follow the ISO standard multi-directional pointing task [24].

Dodd et al. [17] also qualitatively observed that the hand-support was helpful to the participants, in contrast to our results, without controlling for hand-support and without explicitly testing it against a freehand baseline. Dodd et al. conducted a menu navigation, data entry and map panning task, in contrast to the ISO standard multi-directional pointing task used in our study [24].

Hourlier and Servantie [15] did explicitly test a hand-support condition against a freehand baseline, and found that the hand-support reduced error rate. The task they had participants perform was not a Fitts' target selection task (where, generally, participants select a series of targets, in a predictable pattern, in rapid succession). They had participants press a circle, drag it to a target, and release it. They only reported a comparison between the hand-support and freehand condition for the "press" component of this task. The difference between their task and ours may explain the differences between our results.

There are several other factors that could explain why Hourlier and Servantie found a benefit to the hand-support, while we did not:

1. It is possible that tasks focused on slow, precise selection may benefit more from having a hand support than those requiring quick selection of a sequence of targets. For a task that requires the participant to carefully align their hand over a target and then subsequently select it, it's possible that a hand-support may be beneficial. The task used in our study is more representative of typing on a soft-keyboard [31]. However, it may not be representative of a task where someone deliberately selects one menu option at a time, searches through a list and then selects an option, or selects a button multiple times to move forward or backwards through a list. In some of these cases, a hand-support may allow the user to deliberately align their hand with a button or section of the screen, before tapping downwards. It may also help keep their hand aligned, if selecting a button multiple times. As a result, it is possible that the "press" section of the task used by Hourlier and Servantie was one of these slower, more deliberate selection tasks.

2. Hourlier and Servantie used nearly double the magnitude of vibration compared to our test (1.5 m/s2 vs. 0.7 m/s2). It is possible that having a hand-support may become beneficial at even higher vibration levels than those used in this study. In our study, the freehand condition performed much better than the hand support condition in static conditions, but the difference became smaller under vibration . As a result, it is possible there is a crossing-point at which the hand-support eventually becomes beneficial, under higher levels of vibration. However, note that the vibration levels used in our study are quite high (rated as "fairly uncomfortable" in the ISO whole body vibration standard [10]), and pilots may need to concentrate on other aspects of flying and may avoid using the touchscreens at very high levels of vibration.

3. Hourlier and Servantie measured error rate on just the "press" component of a larger task. It is unclear whether participants knew their accuracy was being measured on this component of the task. It is possible that participants naturally tended to move faster when their hand was unsupported, and tended to move slower when their hand was supported. Since, by Fitts' law, movement time is related to accuracy, this could have

caused more errors in the unsupported condition, if participants did not know that their accuracy was being measured.

4. The screen placement in our study was different from that in Hourlier and Servantie's, which could have had an impact.

Given that the task used in this study to compare between hand-support conditions is quite different than that used by Hourlier and Servantie [15], there is currently no evidence to support using a hand-support under vibration for tasks that are well represented by the ISO 9241-411 multi-directional selection task [24].

4.2.1 Hand Support Reach

The hand-support limited the range of movement of the hand and made it difficult to acquire targets farther from the edge. This prompted the participants to adopt certain hand-support strategies, to increase their reach and flexibility when moving between closer and further targets.

When using the MFD, participants found that resting the tips of their fingers on the side of the screen offered the most flexible strategy to reach most of the targets (see Figure 4.1).

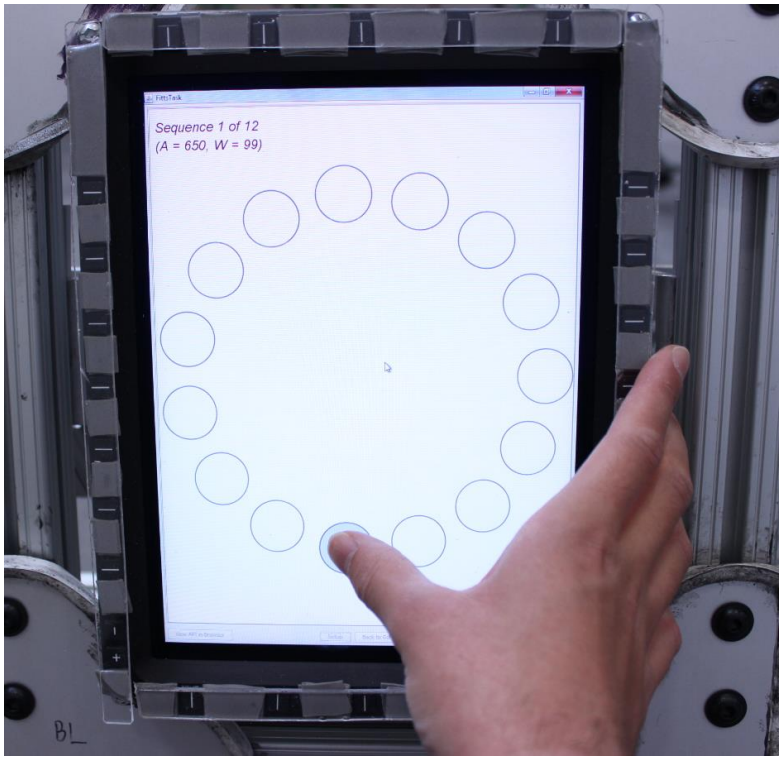


Figure 4.1 – A participant holding onto the MFD screen, during the hand support condition.

Reaching the farthest targets of the MFD and iPad screens was difficult in the hand support condition (see Figure 4.2). For many participants, the farthest targets were at the edge of their reach. They extended their hand fully while resting their fingers flat on the side of the screen.

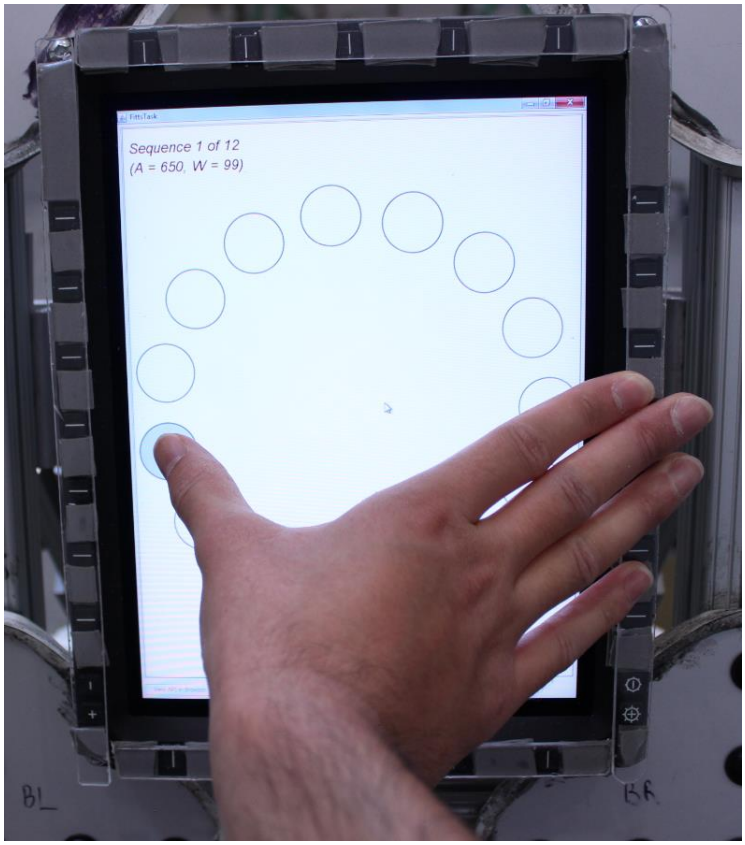


Figure 4.2 – For many participants, the farthest targets were at the edge of reach, in the "with support" condition, for both the MFD screen and iPad.

With the MCDU screen, since it was small, it was much easier to reach fully across the screen to the farthest targets, while maintaining a firmer grip on the bezel edge (see Figure 4.3).

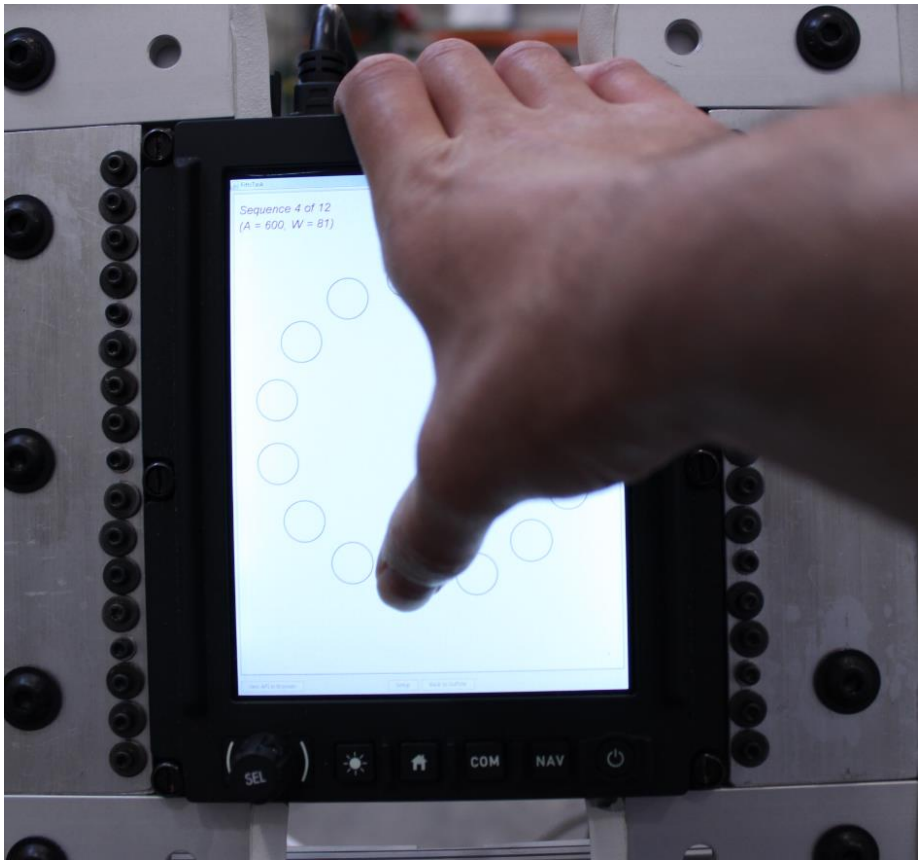


Figure 4.3 – Hand support with the small MCDU screen.

4.3 Compare the Performance on Avionic Touchscreens Versus Commercial Touchscreens

When looking at both throughput and error rate, in combination, the commercial iPad performed the worst, among all the screens. This may have been due to differences in the 2D target selection task interface on the iPad versus the other screens. In particular, the audio feedback for errors lagged behind the error event, for the iPad.

The medium MFD screen performed second worst, considering both throughput and error rate. This was to be expected, since it occasionally did not register taps. This lack of responsiveness may have been due to various workarounds that were done to be able to connect it as a touch monitor to a Windows computer.

The small MCDU screen and large touch monitor performed the best, considering both throughput and error rate.

Overall, with the previously mentioned caveats for the iPad and medium MFD screen, the avionic touchscreens appeared to perform reasonably well against the commercial touchscreens.

Most importantly, the range of throughput values measured for avionics touch screens (5.8 - 6.6 bps) were twice as fast as the throughput of traditional input devices used in the flight deck: track ball (3.0 bps), joystick (1.6 - 2.5 bps), or touchpad (1.0 - 2.9 bps) [8]. Note that these values were measured in static (non-vibrating) conditions. Our findings support the use of touch screens as an efficient input device in the flight deck, as it outperforms other input devices even under the vibrating conditions tested in this study, in terms of throughput. However, note that the error rate on touchscreens was very high for small target sizes, in both vibration and static conditions. As a result, appropriately sized targets are necessary when integrating touchscreens into the flight deck. In addition, the vibration profile used in this study corresponded to normal 120 knot flight conditions in a Bell-412 helicopter. We made the assumption that backup input devices, other than a touchscreen, would be provided for conditions with higher levels of vibration, such as under turbulence or in abnormal flight conditions. In addition, some participants reported feeling fatigued after prolonged use of the touchscreens, both with and without the hand-support. As a result, prolonged touchscreen use in the flight deck may cause fatigue. In addition, in our study, the participants were allowed to fully focus on the target selection task. In a real cockpit, pilots may split their attention between multiple tasks and distractions.

4.4 Compare Task Performance with the Touchscreen in the Main Instrument Panel Versus Pedestal Positions

There was a significant difference between throughput and error rate on the iPad in the main instrument panel versus pedestal positions. This difference was not significant without vibration, but became significant with vibration.

In static conditions, the average iPad throughput (averaged across both hand-support conditions) in the main instrument panel position was 6.5 bits/second, versus 6.6 bits/second for the iPad in

the pedestal position. The average error rate was 14.9% on the main instrument panel, versus 13.2% on the pedestal.

With vibration, the average iPad throughput (averaged across both hand-support conditions) on the main instrument panel was 5.4 bits/second, versus 6.0 bits/second for the iPad on the pedestal. This was a 10% decrease in performance when using the iPad on the MIP compared to the pedestal in vibration conditions. The average error rate was 25.6% on the main instrument panel, versus 17.8% on the pedestal.

We observed an average decrease in throughput of 16% due to vibration on the main instrument panel. Whereas, on the pedestal, the average decrease in throughput due to vibration was 9%.

This points to the importance of screen positioning when attempting to reduce the impact of vibration.

The results of Coutts et al., who measured error rate and movement time, did not show a clear improvement in the pedestal position, compared against the main instrument panel [16]. Coutts et al. found a higher number of errors in the pedestal position, compared to the main instrument panel position, especially for smaller target sizes. They did not that in pedestal position certain taps were not registered, which did not happen in the main instrument panel position. This might explain the discrepancy between our results and theirs. Another possibility is that their screen was larger, and its position may have been slightly farther away, towards the side. In our pre-study, we observed, qualitatively, that placing the screen too far to the side had a large impact on the usability of the pedestal screen (this was based on feedback from five participants, who noted that the screen was initially too far, and based on our own observations). The pedestal screen in particular may be quite sensitive to where it is placed, with just a few inches further to one side making a large difference.

When using the screen in the MIP position, we observed that the participants' hands were outstretched and elevated. Under vibration, this likely put more torque on the shoulder, since the weight of the arm was at a further distance. In the pedestal position, the arm was extended more downwards and less outwards. As a result, the vibration likely acted on less of a lever arm, especially since the vibration profile used in this study was solely in the vertical axis. It may be less true for vibration profiles that include higher levels of vibration in the other axes. Participants who liked the pedestal position over the MIP position tended to find it less fatiguing

on the shoulder and arm. Those who liked the MIP position over the pedestal position liked that it was directly in front of them, requiring them to turn their head less, which put less strain on their neck and gave them a clearer view of the screen.

4.5 Effect of Target Size on Error Rate

The independent variable that most impacted error rate was target size. The impact of target size on error rate appeared to be exponential, for the target sizes tested in this study. Under vibration, the error rates per target size, when averaged across all finger and screen conditions, were: 34.2% for a target size of 0.8 cm (0.3 inches), 21.2% for a target size of 1 cm (0.4 inches), 7.0% for a target size of 1.5 cm (0.6 inches), and 4.0% for a target size of 2 cm (0.8 inches). The error rate varied significantly between screens, with some screens have lower error rates than this average, and some having higher error rates. Since there were no consequences to making errors and participants were not asked to prioritise accuracy, these error rates may be higher than in a real task. However, on the other hand, the participants could focus their attention on this one task, and where not asked to split their attention between multiple tasks and distractions, as a pilot might need to. Hence, the error rate results reported here could alternatively be lower than in a real cockpit environment during flight. As a result, the absolute error rates shown here should be regarded with some caution. However, the relative error rates do provide insight and show an exponential relationship.

This exponential relationship between error rate and target size matches the findings of Avsar et al. [45], and Coutts et al. [16]. Avsar et al. reported lower average error rates than in our study, with the screen in a fixed position, of 12% for 1 cm targets, 5.3% for 1.5 cm targets, and 2% for 2 cm targets. Coutts et al. reported number of errors, rather than error rate, and had the participants attempt to re-select targets if they missed them, making it difficult to compare our results against theirs.

Our study found a significant interaction effect on error rate between target size and screen, and target size and vibration. As a result, it may be difficult to come up with an error rate per target size graph that applies for all scenarios, cockpit layouts and interfaces. Further testing may be required for each specific context, in determining an appropriate target size and acceptable error rate.

We looked at the size of the circle containing 96% of taps points (W96%) versus target width (W). In static conditions, averaged across all screen and finger conditions, a target size of: W=0.8 cm resulted in W96%=1.3 cm; W=1 cm resulted in W96%=1.4 cm; W=1.5 cm resulted in W96%=1.6 cm; W=2 cm resulted in 1.8 cm.

Under vibration, averaged across all screen and finger conditions, a target size of: W=0.8 cm resulted in W96%=1.6 cm; W=1 cm resulted in W96%=1.7 cm; W=1.5 cm resulted in W96%=1.8 cm; W=2 cm resulted in 2.0 cm.

When looking just at the W=0.8 cm target size, we compared the W96% size across different Finger x Vibration x Screen conditions. When using the index finger under vibration at a target size of W=0.8: the iPad in the MIP position had a W96% of 1.8 cm; the iPad in the pedestal position had a W96% of 1.6 cm; the large touch monitor had a W96% of 1.5 cm; the MFD screen had a W96% of 1.4 cm; and the small MCDU screen had a W96% of 1.4 cm.

This means that, even when presented with a target size of 0.8 cm, the participant's tap points tended to be distributed over a zone, centered on the target, of 1.4-1.8 cm (with 4% of taps being outside this zone), depending on the screen being considered.

4.6 Effect of Target Distance on Error Rate

In addition to the impact of target size on error rate, we also looked at the impact of target distance on error rate.

For the freehand condition, the error rate appeared to increase linearly with increasing distance between targets, over the amplitude conditions tested in this study.

For the hand-support condition, the error rate increased more between the medium and largest amplitude conditions, compared to between the smallest and medium amplitude conditions. This was likely because the largest amplitude condition was at the limit of the participants' reach.

4.7 Implications for design

This study has implications for the design of avionic touchscreen interfaces.

The multi-directional selection task [24] used in this study has been shown to be representative of one-handed typing on a soft-keyboard [18, 31]. Thus, the results of our study are most applicable to the task of one-handed typing on a touchscreen soft-keyboard that spans the screen.

Based on our findings, the two smallest target sizes of 0.8 and 1 cm should be avoided in touchscreen interfaces for the flight deck, given their very high error rates. This finding is important as these are the smallest widget sizes present on avionic interfaces that use pointing devices (ex. Trackball), as mentioned to us by aviation experts, and as implied by Letsu-Dake et al. [75]. Larger targets are thus required for touchscreen interaction. When shown a small target size of 0.8 cm, participant tap points resulted in a W96% size of 1.4-1.8 cm under the freehand-vibration condition, depending on the screen being considered. This means that, even when presented with a target size of 0.8 cm, the participant's tap points tended to be distributed over a zone, centered on the target, of 1.4-1.8 cm (with 4% of taps being outside this zone), depending on the screen being considered. When presented with larger targets, the participant tap points spread out more, with an average W96% value of 2 cm when the participants were presented with a 2 cm target, under freehand-vibration conditions. In comparison, the MIL-STD-1472H standard [67] recommends a minimum target size of 1.5 cm for touch targets (except those on touch keyboards), and suggests a target size of 1.6 cm, without providing a minimum, for touch keyboards. At a 1.5 cm target size, we found an average W96% size of 1.8 cm, under vibration conditions.

We found better performance and reduced error rate when participants used the screen on the pedestal compared to the MIP under vibration. Data entry-intensive formats, such as the Flight Management System (FMS) and checklist application, could be presented on a touch screen located on the pedestal to make them more robust to error under vibration. This follows the recent trend of aircraft manufacturers that favour interactive formats on the pedestal or side console whereas displays on the MIP are mostly for presentation of non-interactive information, ex. Primary Flight Display (PFD).

We found no clear benefit to using a hand-support under the levels of vibration tested, and there was a clear detriment to using it without vibration, when performing a multi-directional selection task. Under vibration, increasing the target size provided a much larger benefit, compared to the hand-support, in terms of reducing error rate and increasing throughput.

On one of the avionic touchscreens that was used in this study, there were physical buttons located on the top and side bezels. In a pretest, conducted amongst the investigators, we found that there was a tendency to accidentally hit and activate these bezel keys, when using the bezel edge as a hand support. When using the bezel edge as a hand support, the hand is in close proximity to the bezel key buttons and can easily come into contact with them, especially when reaching for far targets. If this type of error were to occur during flight, especially if it were to go unnoticed, it could have negative safety implications, depending on the function assigned to the bezel key. Thus, for avionic touchscreens that include bezel keys, the bezel keys should be designed in such a way to prevent accidental activation, especially if the bezel edge is intended to be used as a hand support.

4.8 Limitations and Recommendations for Further Research

This study did not find that using a hand support provided a benefit under vibration, for the task that was asked of the participants. However, it might provide a benefit for different types of tasks, not tested here, or it may provide a benefit under higher levels of vibration. The limits of where a hand support can provide a benefit and where it does not should be determined by future research, especially since it is currently recommended in the SAE ARP60494 standard [12]. Other hand-support strategies, not investigated here, should also be tested.

This study used one vibration profile, which was solely in the vertical axis. Future research should investigate the impact on throughput from different levels of vibration in different axes and in combinations of axes, as well as the impact of having different vibration levels on the touchscreen versus on the pilot's seat.

Some participants naturally wanted to rest their arm on their knee, as an arm-support strategy. This may be a practical strategy that pilots could use in certain cockpit layouts. As a result, future research could evaluate the effectiveness of resting the arm on the knee, while using touchscreens under vibration.

The chair used for this study did not have an armrest. As a result, when using the touchscreen in the pedestal position, the arm was not supported. However, in certain cockpit layouts, the

pilot may be able to rest part of their arm on a pedestal support, while using the touchscreen. This arm-support strategy should be tested under vibration.

The participants were instructed to perform the task as quickly and as accurately as possible. The participants were not penalized for making errors. The study took around 2 hours per participant, and the participants may have sped up over time, in order to get through the tasks faster. Hence, the error rate results reported here may have been higher than if there were clearer downsides to making an error, as there would be in a cockpit. On the other hand, pilots must sometimes split their attention between multiple tasks, and can have additional distractions. In this study, we only asked the participants to perform one task, on which they could focus their full attention. As a result, performing a target selection task under stressful, high workload conditions, as one of several tasks, could result in higher error rates and lower performance values than those reported here. Future research should measure error rate and performance at different target sizes for different tasks, use contexts and environmental conditions.

In this study, we had participants use their dominant hand. However, pilots often use their non-dominant hand to interact with controls as well. Future research should look into quantifying the impact of vibration on non-dominant hand use, while using touchscreens.

Helicopter pilots often wear gloves. Future research should measure the impact of gloves on error rate and effective target size.

In this study, the targets were arranged around a circle, centered in the screen. Future research should look into task performance and error rate in different areas of the screen, with different hand-support strategies.

In this study, we did not prevent the participants from leaning forward, nor did we measure the amount that each participant leaned. Future studies should measure the impact of leaning forward on touchscreen performance under vibration.

In this study, we measured two screen positions, one for the main instrument panel and one for the pedestal. Each screen position was tested at only one distance, as measured relative to the participant's eye. However, different aircraft cockpits can have different layouts, with screens placed at different distances, and screens can be placed in different positions than those tested

here. Future research should look into the impact of different aspects of screen placement, including screen distance, height, lateral offset and orientation, on touchscreen usability.

This study provided average throughput and error rate values across participants of different ages and anthropometric measurements. Future research should investigate the impact of age and different anthropometric measurements on touchscreen usability in a vibrating cockpit environment.

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS

Touchscreens are beginning to be installed in many different types of aircraft cockpits [1-3]. These touchscreens may be located at a distance that is difficult for the pilot to reach. Helicopter cockpits in particular can undergo high levels of vibration during normal flight [9, 11]. This combination of reach distance and vibration could make touchscreen use in the flight deck more difficult than in other contexts. There is thus a need to quantify the impact of vibration on touchscreen use in the flight deck environment. There is also a need to evaluate mitigation methods that could improve touchscreen use in this environment.

We used the multi-directional selection task recommended by the ISO 9241-411 standard [24]. We calculated and reported the throughput value, per independent variable, according to the recommendations in that standard, as well as those of Soukoreff and MacKenzie [18], and Mackenzie [23]. We also reported the error rate results, per independent variable, for this type of task. Since we used a standardized task and throughput formula, our results can be compared against others in the literature [18].

The multi-directional selection task used in this study is most representative of a one-handed typing task on a soft-keyboard [18, 31]. Hence, the results of this study are most applicable for a one-handed typing task on a touchscreen soft-keyboard that spans the screen.

The screen positions were representative of a flight deck layout [54, 55]. We used a vibration profile representative of a Bell-412 helicopter during 120 knot flight [11, 60].

This study had 24 participants, who had a range of ages and anthropometrics.

The independent variables were: 1) Hand-support method (thumb with hand-support versus index finger freehand); 2) Vibration (no vibration, with vibration); 3) Screen (avionic MFD touchscreen, avionic MCDU touchscreen, consumer touch monitor, iPad); 4) Screen position (MIP position, pedestal position); 5) Target size; 6) Distance between targets; 7) Selection angle.

The main dependent variables where: throughput and error rate.

We found that throughput values on touchscreens, even under vibration and even in a flight deck layout, were higher than those reported for other common input devices tested in static environments [7, 8, 18, 74]. This implies that touchscreens could be very efficient input devices

for the flight deck, in terms of allowing more rapid selections compared to other input devices. However, touchscreens have several limitations that need to be considered:

- In this study, the error rate when using touchscreens under vibration was high, particularly for small target sizes. We recommend avoiding target sizes of 0.8 cm and 1 cm, since these caused very high error rates. We have confirmed, along with prior work [16, 44, 45], that error rate increases exponentially with decreasing target sizes, with the slope becoming steeper under vibration. Hence careful consideration of target size is recommended, when designing touchscreen interfaces for the vibrating cockpit environment.
- Prior work has shown that vibration has a larger impact on touchscreen throughput and error rate, when compared against other selection devices [8]. Hence, we expect that, at vibration levels above those tested in this study, it is possible that task performance may degrade and error rate may increase to unacceptable levels, requiring a backup input device that is less impacted by vibration.
- Participants reported fatigue after prolonged use of the touchscreens, even when using a hand-support. As a result, prolonged touchscreen use in the cockpit may cause fatigue, especially under vibration and especially for farther reach distances.

We failed to find a clear benefit of using the bezel edge as a hand-support while using the thumb for selection, compared to using the index finger freehand, when performing a multi-directional selection task. The index finger freehand outperformed the hand-support method, in terms of higher throughput and lower error rate, in static conditions. Under the vibration conditions tested here, the index finger freehand showed similar performance to the hand-support method, in terms of throughput and error rate.

We found that having the touchscreen in the pedestal position rather than the main instrument panel position resulted in better performance (higher throughput and lower error rate) in vibration conditions. However, the two positions were equivalent, in terms of throughput and error rate, in static conditions.

Finally, we built a vibrating platform that reproduces the physical geometry of a flight deck and we validated that the vibration profile used was representative of helicopter flight. This test setup

is easily reconfigurable and adjustable to accommodate different sized participants and touchscreens. It offers novel and much needed test capabilities to the Québec aviation community. One aerospace company has plans to use the test setup as part of an interface certification process with an aviation regulatory body. The setup makes it much faster and cheaper to test, modify and improve technology under representative vibration conditions, compared to running flight tests, which thus expands the range and quantity of tests that can be conducted.

Since the test setup we built is configurable and programmable, it could also be used to conduct usability testing for other types of devices in a range of different vibrating environments. These vibrating environments could be representative of different types of aircraft, but also other types of vehicles, such as cars, trucks, trains, etc.

This work will contribute to improving the usability of touchscreen interaction in aviation, as it provided design recommendations on factors impacting selection performance and accuracy. Our findings can also apply to other transportation environments, as touchscreens are making their way into cars (ex. GPS navigation, music selection), trains (ex. railroad navigation) and marine ships. We hope that our findings will contribute to improving the overall safety of operations that use touchscreens in vibrating environments.

REFERENCES

[1] C. B. Watkins, C. Nilson, S. Taylor, K. B. Medin, I. Kuljanin, and H. B. Nguyen, "Development of touchscreen displays for the gulfstream g500 and g600 symmetry™ flight deck," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 23-27 Sept. 2018 2018, pp. 1-10, doi: 10.1109/DASC.2018.8569532.

[2] Thales Group. "FlytX tactile large display flight deck for aircraft & rotorcraft." Thales Group. https://www.thalesgroup.com/en/markets/aerospace/flight-deck-avionics-equipment-functions/flytx-tactile-large-display-flight-deck (accessed September 20, 2021).

[3] Garmin. "G500H TXi." Garmin. https://buy.garmin.com/en-CA/CA/p/625427 (accessed September 20, 2021).

[4] CMC Electronics. *CMA-9000 FMS/RMS: Flight Management System for Commercial, Paramilitary and Military Helicopters*. https://cmcelectronics.ca: CMC Electronics Inc. Accessed: September 26, 2021. [Online]. Available: https://cmcelectronics.ca/Portals/17/Documents/en-us/CMC-CMA9000-FMS-RMS-19-003.pdf

[5] L. Wang, Q. Cao, J. Chang, and C. Zhao, "The Effect of Touch-key Size and Shape on the Usability of Flight Deck MCDU," in *ACHI 2015*, 2015.

[6] I. S. MacKenzie, "Fitts' throughput and the remarkable case of touch-based target selection," in *International conference on human-computer interaction*, 2015: Springer, pp. 238-249.

[7] P. R. Thomas, "Performance, characteristics, and error rates of cursor control devices for aircraft cockpit interaction," *International Journal of Human-Computer Studies*, vol. 109, pp. 41-53, 2018/01/01/ 2018, doi: https://doi.org/10.1016/j.ijhcs.2017.08.003.

[8] C. J. Lin, C. N. Liu, C. J. Chao, and H. J. Chen, "The performance of computer input devices in a vibration environment," *Ergonomics*, vol. 53, no. 4, pp. 478-490, 2010/04/01 2010, doi: 10.1080/00140130903528186.

[9] Y. Chen, S. Ghinet, A. Price, V. Wickramasinghe, and A. Grewal, "Evaluation of aircrew whole-body vibration exposure on a Canadian CH-147F Chinook Helicopter," (in eng), *Journal of the American Helicopter Society*, 2017/03/07 2017, doi: 10.4050/JAHS.62.022004.

[10] *ISO 2631-1:1997 - Mechanical vibration and shock - Evaluation of human exposure to whole-body vibration - Part 1 : General requirements*, International Organization for Standardization, [Geneva], 1997.

[11] V. K. Wickramasinghe, "Dynamics Control Approaches to Improve Vibratory Environment of the Helicopter Aircrew," Library and Archives Canada, Ottawa, 2012.

[12] *ARP60494: Touch Interactive Display Systems: Human Factors Considerations*, SAE International, 2019.

[13] A. Cockburn *et al.*, "Turbulent Touch: Touchscreen Input for Cockpit Flight Displays," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*: Association for Computing Machinery, 2017, pp. 6742–6753.

[14] A. Cockburn *et al.*, "Design and evaluation of braced touch for touchscreen input stabilisation," *International Journal of Human-Computer Studies*, vol. 122, pp. 21-37, 2019/02/01/ 2019, doi: https://doi.org/10.1016/j.ijhcs.2018.08.005.

[15] S. Hourlier and X. Servantie, "Avionics Touch Screen in Turbulence: Simulation for Design," in *19th International Symposium on Aviation Psychology*, 2017. [Online]. Available: https://corescholar.libraries.wright.edu/cgi/viewcontent.cgi?article=1034&context=isap_2017.

[16] L. V. Coutts *et al.*, "Future technology on the flight deck: assessing the use of touchscreens in vibration environments," *Ergonomics*, vol. 62, no. 2, pp. 286-304, 2019/02/01 2019, doi: 10.1080/00140139.2018.1552013.

[17] S. R. Dodd, J. Lancaster, S. Grothe, B. DeMers, B. Rogers, and A. Miranda, "Touch on the flight deck: The impact of display location, size, touch technology & turbulence on pilot performance," in *2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC)*, 5-9 Oct. 2014 2014, pp. 2C3-1-2C3-13, doi: 10.1109/DASC.2014.6979428.

[18] R. W. Soukoreff and I. S. MacKenzie, "Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI," *International journal of human-computer studies*, vol. 61, no. 6, pp. 751–789, 2004, doi: 10.1016/j.ijhcs.2004.09.001.

[19] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *Journal of experimental psychology*, vol. 47, no. 6, p. 381, 1954.

[20] S. Kelso, "Theoretical Concepts and Strategies for Understanding Perceptual-Motor Skill: From Information Capacity in Closed Systems to Self-Organization in Open, Nonequilibrium Systems," *Journal of experimental psychology. General*, vol. 121, pp. 260-1, 10/01 1992, doi: 10.1037/0096-3445.121.3.260.

[21] M. Takeda, T. Sato, H. Saito, H. Iwasaki, I. Nambu, and Y. Wada, "Explanation of Fitts law in Reaching Movement based on Human Arm Dynamics," *Scientific Reports*, vol. 9, 2019.

[22] J. Gori, O. Rioul, and Y. Guiard, "Speed-Accuracy Tradeoff: A Formal Information-Theoretic Transmission Scheme (FITTS)," *ACM Trans. Comput.-Hum. Interact.*, vol. 25, no. 5, p. Article 27, 2018, doi: 10.1145/3231595.

[23] I. S. MacKenzie, "Fitts' law," *Handbook of human-computer interaction*, vol. 1, pp. 349-370, 2018.

[24] *ISO 9241-411 - Ergonomics of human-system interaction. Part 411, Evaluation methods for the design of physical input devices*, International Organization for Standardization, Geneva, [Suisse], 2012.

[25] X. Bi, Y. Li, and S. Zhai, "FFitts law: modeling finger touch with fitts' law," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2013, pp. 1363-1372.

[26] I. S. MacKenzie, "A Note on the Validity of the Shannon Formulation for Fitts' Index of Difficulty," *Open Journal of Applied Sciences*, October 2013. [Online]. Available: https://www.yorku.ca/mack/ojas2013.html.

[27] I. S. MacKenzie and P. Isokoski, "Fitts' throughput and the speed-accuracy tradeoff," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, 2008. [Online]. Available: https://doi.org/10.1145/1357054.1357308.

[28] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379-423, 1948.

[29] S. Zhai, J. Kong, and X. Ren, "Speed–accuracy tradeoff in Fitts' law tasks—on the equivalency of actual and nominal pointing precision," *International journal of human-computer studies*, vol. 61, no. 6, pp. 823-856, 2004.

[30] S. MacKenzie and W. Soukoreff. "GoFitts Throughput Webpage." York University. https://www.yorku.ca/mack/FittsLawSoftware/doc/GoFitts.html (accessed November 1, 2021).

[31] R. William Soukoreff and I. Scott Mackenzie, "Theoretical upper and lower bounds on typing speed using a stylus and a soft keyboard," *Behaviour & Information Technology*, vol. 14, no. 6, pp. 370-379, 1995/11/01 1995, doi: 10.1080/01449299508914656.

[32] A. Pavlovych and W. Stuerzlinger, "The tradeoff between spatial jitter and latency in pointing tasks," presented at the Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems, Pittsburgh, PA, USA, 2009. [Online]. Available: https://doi.org/10.1145/1570433.1570469.

[33] M. Bachynskyi, G. Palmas, A. Oulasvirta, J. Steimle, and T. Weinkauf, "Performance and Ergonomics of Touch Surfaces: A Comparative Study using Biomechanical Simulation," presented at the Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Republic of Korea, 2015. [Online]. Available: https://doi.org/10.1145/2702123.2702607.

[34] S. Vetter, J. Bützler, N. Jochems, and C. M. Schlick, "Fitts' law in bivariate pointing on large touch screens: Age-differentiated analysis of motion angle effects on movement times and error rates," in *International Conference on Universal Access in Human-Computer Interaction*, 2011: Springer, pp. 620-628.

[35] C. J. Lin and S.-H. Ho, "Prediction of the use of mobile device interfaces in the progressive aging process with the model of Fitts' law," *Journal of Biomedical Informatics*, vol. 107, p. 103457, 2020.

[36] E. J. Frett and K. E. Barner, "Accuracy and frequency analysis of MultiTouch interfaces for individuals with Parkinsonian and essential hand tremor," in *Proceedings of the 7th International ACM SIGACCESS Conference on Computers and Accessibility*, 2005, pp. 60-67.

[37] R. Plamondon and A. M. Alimi, "Speed/accuracy trade-offs in target-directed movements," *Behavioral and brain sciences,* vol. 20, no. 2, pp. 279-303, 1997.

[38] A. Goguey, M. Nancel, G. Casiez, and D. Vogel, "The Performance and Preference of Different Fingers and Chords for Pointing, Dragging, and Object Transformation," presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San Jose, California, USA, 2016. [Online]. Available: https://doi.org/10.1145/2858036.2858194.

[39] S. C. Lee, M. C. Cha, and Y. G. Ji, "Investigating Smartphone Touch Area with One-Handed Interaction: Effects of Target Distance and Direction on Touch Behaviors," *International Journal of Human–Computer Interaction,* vol. 35, no. 16, pp. 1532-1543, 2019/10/02 2019, doi: 10.1080/10447318.2018.1554320.

[40] Q. Nguyen and M. Kipp, *Where to Start? Exploring the Efficiency of Translation Movements on Multitouch Devices*. 2015.

[41] M. B. Trudeau, T. Udtamadilok, A. K. Karlson, and J. T. Dennerlein, "Thumb Motor Performance Varies by Movement Orientation, Direction, and Device Size During Single-Handed Mobile Phone Use," *Human Factors,* vol. 54, no. 1, pp. 52-59, 2012, doi: 10.1177/0018720811423660.

[42] F. Lehmann and M. Kipp, "How to Hold Your Phone When Tapping: A Comparative Study of Performance, Precision, and Errors," presented at the Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces, Tokyo, Japan, 2018. [Online]. Available: https://doi.org/10.1145/3279778.3279791.

[43] I. Kim and J. H. Jo, "Performance Comparisons Between Thumb-Based and Finger-Based Input on a Small Touch-Screen Under Realistic Variability," *International Journal of Human–Computer Interaction,* vol. 31, no. 11, pp. 746-760, 2015/11/02 2015, doi: 10.1080/10447318.2015.1045241.

[44] H. Avsar, "Exploring potential benefits and challenges of touch screens on the flight deck," University of Nottingham, 2017.

[45] H. Avsar, J. Fischer, and T. Rodden, "Target size guidelines for interactive displays on the flight deck," in *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, 13-17 Sept. 2015 2015, pp. 3C4-1-3C4-15, doi: 10.1109/DASC.2015.7311400.

[46] *GoFitts*. (2017-2020). I. Scott MacKenzie, https://www.yorku.ca/. Accessed: September 10, 2020. [Online]. Available: https://www.yorku.ca/mack/FittsLawSoftware/doc/GoFitts.html

[47] I. S. MacKenzie. "GoFitts Webpage." York University. https://www.yorku.ca/mack/FittsLawSoftware/doc/GoFitts.html (accessed October 18, 2021).

[48] ScioTeq. "MDU-268v2." ScioTeq. https://www.scioteq.com/en/avionics/video-displays/mdu-268v2 (accessed November 7, 2021).

[49] ScioTeq. "TSCU-5045 " ScioTeq. https://www.scioteq.com/en/avionics/smart-displays/tscu-5045 (accessed November 7, 2021).

[50] Planar. *Planar Helium PCT2435 - 24" Touch Screen Monitor*. (2021). www.planar.com: Planar. Accessed: October 2, 2021. [Online]. Available: https://www.planar.com/products/desktop-touch-screen-monitors/touch-screen-monitors/24-27-inch/pct2435/planar-helium-pct2435/?f=Planar%20Helium%20PCT2435.pdf

[51] Collins Aerospace. "MFD-4820 Large Area Display." Collins Aerospace. https://www.collinsaerospace.com/what-we-do/Military-And-Defense/Displays-And-Controls/Airborne/Head-Down-Displays/Mfd-4820-Large-Area-Display (accessed November 7, 2021).

[52] ScioTeq. "RDU-4208." ScioTeq. https://www.scioteq.com/en/avionics/video-displays/rdu-4208 (accessed November 7, 2021).

[53] Apple. "iPad (6th generation) - Technical Specifications." Apple Inc. https://support.apple.com/kb/sp774?locale=en_CA (accessed October 2, 2021).

[54] *AC 29-2C - Certification of Transport Category Rotorcraft*, Federal Aviation Administration, 2016. [Online]. Available: https://www.faa.gov/documentLibrary/media/Advisory_Circular/AC_29-2C_Change_1-7.pdf

[55] *AC 25.773-1 - Pilot Compartment View Design Considerations*, Federal Aviation Administration, 1993. [Online]. Available: https://www.faa.gov/documentLibrary/media/Advisory_Circular/AC_25_773-1.pdf

[56] A. Stefanidis and M. Macik, "Simulation of Electronic Flight Instrument System of Boeing 787 aircraft," in *Proceedings of the 16th Central European Seminar on Computer Graphics, CESCG*, 2012.

[57] R. A. Wynne, K. J. Parnell, M. A. Smith, K. L. Plant, and N. A. Stanton, "Can't Touch This: Hammer Time on Touchscreen Task Performance Variability under Simulated Turbulent Flight Conditions," *International Journal of Human–Computer Interaction*, vol. 37, no. 7, pp. 666-679, 2021/04/21 2021, doi: 10.1080/10447318.2021.1890492.

[58] N. C. M. van Zon, C. Borst, D. M. Pool, and M. M. van Paassen, "Touchscreens for Aircraft Navigation Tasks: Comparing Accuracy and Throughput of Three Flight Deck Interfaces Using Fitts' Law," *Human Factors*, vol. 62, no. 6, pp. 897-908, 2020, doi: 10.1177/0018720819862146.

[59] V. Wickramasinghe, Y. Chen, and D. Zimcik, "Development of an active suspension system for adaptive vibration control of helicopter seats," presented at the Active Suspension Technologies for Military Vehicles and Platforms, 2011/05/01, 2011.

[60] H. Wright Beatty *et al.*, "Effects of vibration level and a vibration mitigating cushion on neck strain using a human rated shaker facility," in "Laboratory Technical Report (National Research Council of Canada. Flight Research Laboratory); no. LTR-FRL-2016-0069," National Research Council of Canada. Aerospace, 0384-3157, 2016/06/30 2016.

[61] C. Yong, V. Wickramasinghe, and D. Zimcik, "Development of Adaptive Helicopter Seat Systems for Aircrew Vibration Mitigation," *Proc SPIE*, vol. 6928, 05/01 2008, doi: 10.1117/12.773336.

[62] *ANSI S1.11-2004: Specification for Octave-Band and Fractional-Octave-Band Analog and Digital Filters*, American National Standards Institute, Melleville, NY, February 19 2004.

[63] N. J. Mansfield, *Human response to vibration*, Boca Raton, FL: CRC Press, 2005. [Online]. Available: https://www.taylorfrancis.com/books/e/9780203487228.

[64] D-Box. *GP PRO-500 GUIDE D'INSTALLATION*. D-Box.

[65] J.-M. Lizotte, "MFX-130 SEAT ACCELERATION : 185-985-0099-EN1," D-Box, March 27 2019.

[66] D. J. Biau, "In Brief: Standard Deviation and Standard Error," *Clinical Orthopaedics and Related Research®*, vol. 469, no. 9, pp. 2661-2664, 2011/09/01 2011, doi: 10.1007/s11999-011-1908-9.

[67] *MIL-STD-1472H: Department of Defense Design Criteria Standard - Human Engineering*, United States of America Department of Defense, September 15 2020.

[68] *ISO 9241-9 - Ergonomic requirements for office work with visual display terminals (VDTs) — Part 9: Requirements for non-keyboard input devices*, International Organization for Standardization, Geneva, [Suisse], 2000.

[69] *14 CFR 25.777 - Cockpit controls*, Office of the Federal Register - National Archives and Records Administration, www.govinfo.gov, January 1 2002. [Online]. Available: https://www.govinfo.gov/app/details/CFR-2002-title14-vol1/CFR-2002-title14-vol1-sec25-777

[70] S. Klemmer and J. O. Wobbrock, "Designing, Running, and Analyzing Experiments," ed. https://www.coursera.org: Coursera.

[71] *Easy Analysis and Visualization of Factorial Experiments*. (2016). Michael A. Lawrence, http://github.com/mike-lawrence/ez. [Online]. Available: http://github.com/mike-lawrence/ez

[72] *Aligned Rank Transform*. (2021). M. Kay, L. A. Elkin, J. J. Higgins, J. O. Wobbrock, GitHub, https://github.com/mjskay/ARTool/. [Online]. Available: https://github.com/mjskay/ARTool/

[73] J. O. Wobbrock, L. Findlater, D. Gergle, and J. J. Higgins, "The aligned rank transform for nonparametric factorial analyses using only anova procedures," in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2011, pp. 143-146.

[74] P. Doyon-Poulin and N. Routhier, "Use of throughput to evaluate a Cursor Control Device (CCD) performance," presented at the Canadian Aerospace Institute Conference - AERO, Montreal, Canada, 2011.

[75] E. Letsu-Dake, E. Alves, R. Khatwa, and S. Dodd, "Motion Simulator Evaluation of a Flight Deck Cursor Control Device (CCD)," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 2018: IEEE, pp. 1-10.

[76] J. O. Smith, "Mathematics of the discrete fourier transform (dft)," *Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California*, 2002.

[77] D-Box. *Motion Code Creation with D-BOX*. (2019). D-Box.

APPENDIX A SCREEN HEIGHT ADJUSTMENT

The height of the touchscreens was adjusted for each participant, based on their sitting eye height. This was done to ensure that the center of the screen was always placed at the same Euclidian distance from the eye, for all participants.

Participants were asked to sit sideways beside a wall with height markings on it (Figure A. 1). This allowed us to take anthropometric measurements from a safe distance, due to COVID-19 concerns. The height of the chair on which the participants sat, for this measurement, was the same height off the ground as the D-Box chair.

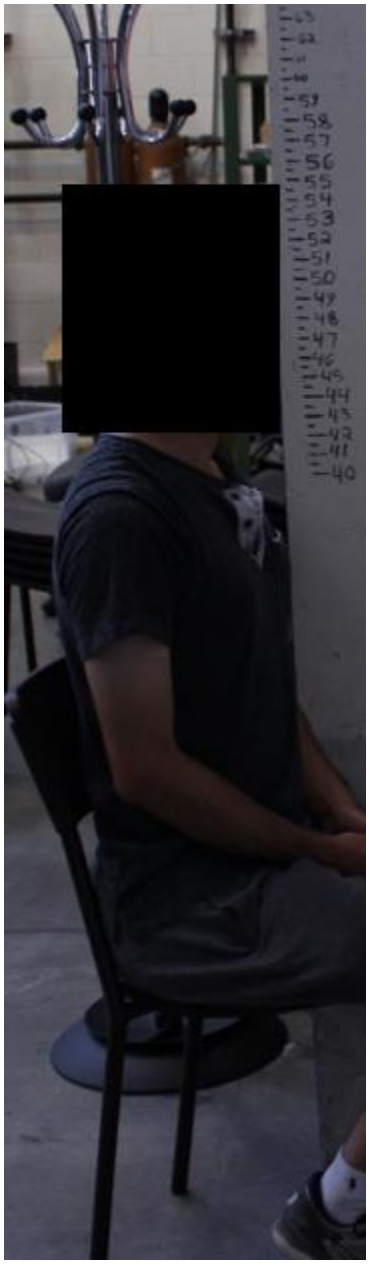


Figure A.1 – A person modeling how the sitting eye height was measured. The markings are in inches.

The participant's sitting eye height was then inserted into the following formula, to determine the vertical placement of the center of the MIP screen off of the ground:

(Center of MIP screen off the ground) = (Measured sitting eye height in inches) – (12.5 inches)

Where 12.5 inches (21.8 cm) was the vertical distance between the ERP and center of the MFD in the MIP position, shown in the side view (see Figure 2.2). A tape measure was then used to ensure that the screen's center was the proper vertical distance off the ground.

The pedestal screen was attached to the same adjustable support beams as the MIP screen. Hence adjusting the MIP screen automatically also adjusted the height of the pedestal screen.

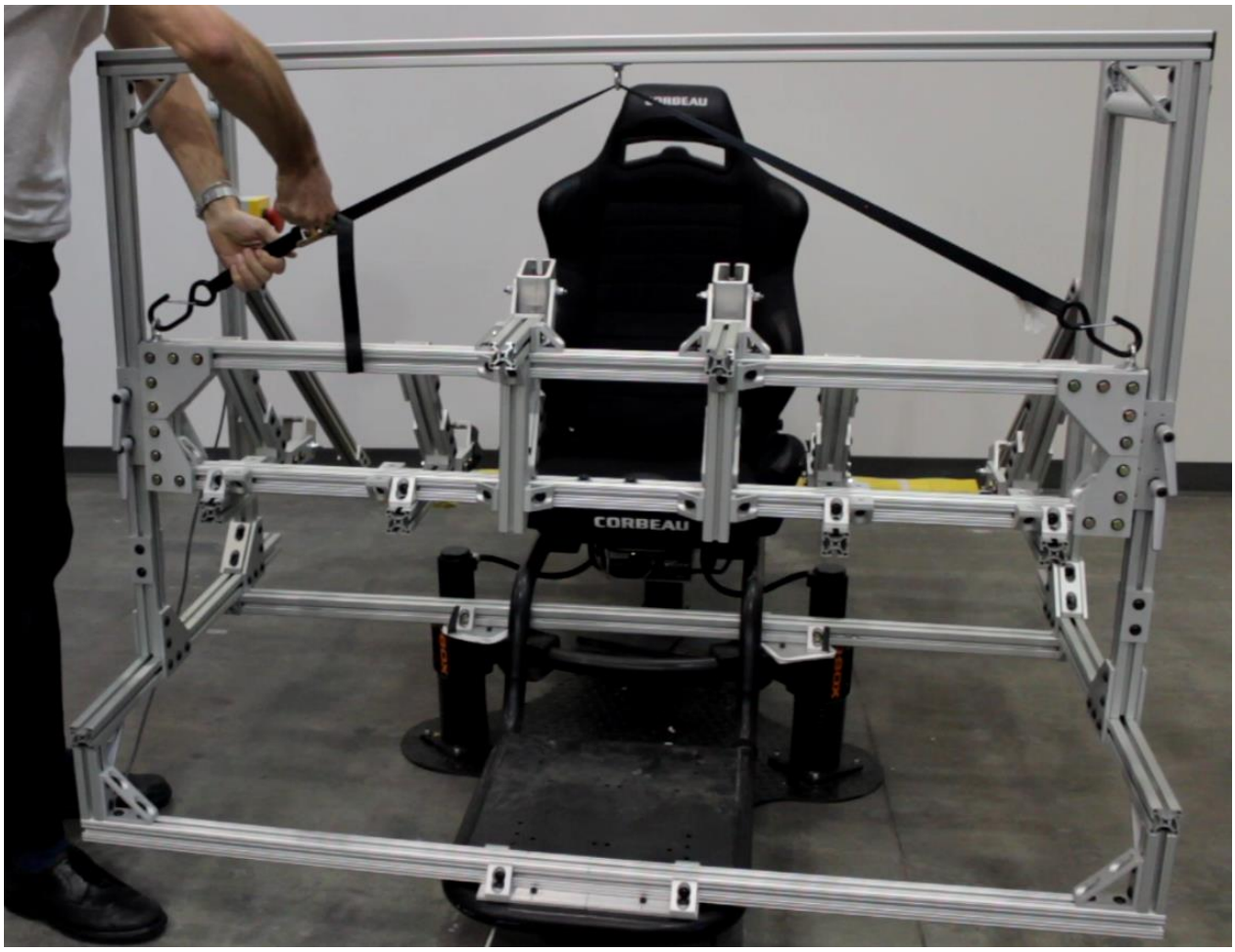


Figure A.2 – Screen support beams being adjusted to the correct height.

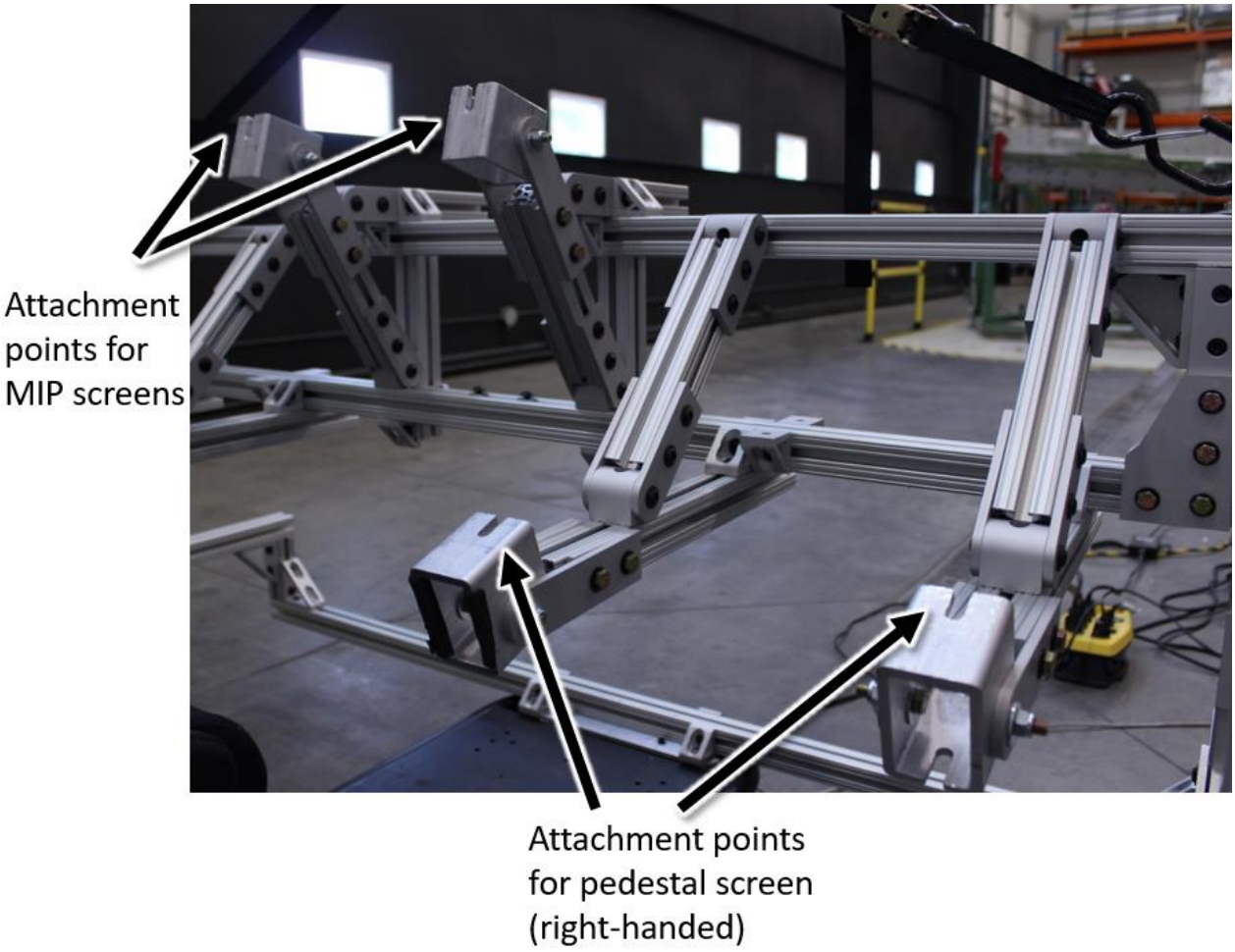


Figure A.3 – The attachment points for the MIP screens and pedestal screen are fixed to the same adjustable support beams.

The MIP screens were centered, so no change was required to accommodate righties or lefties. However, the pedestal screen was placed to the right of the participant, if they were a righty. Alternatively, it was placed to the left of the participant, if they were a lefty. For ambidextrous participants, the pedestal screen was placed to the right.

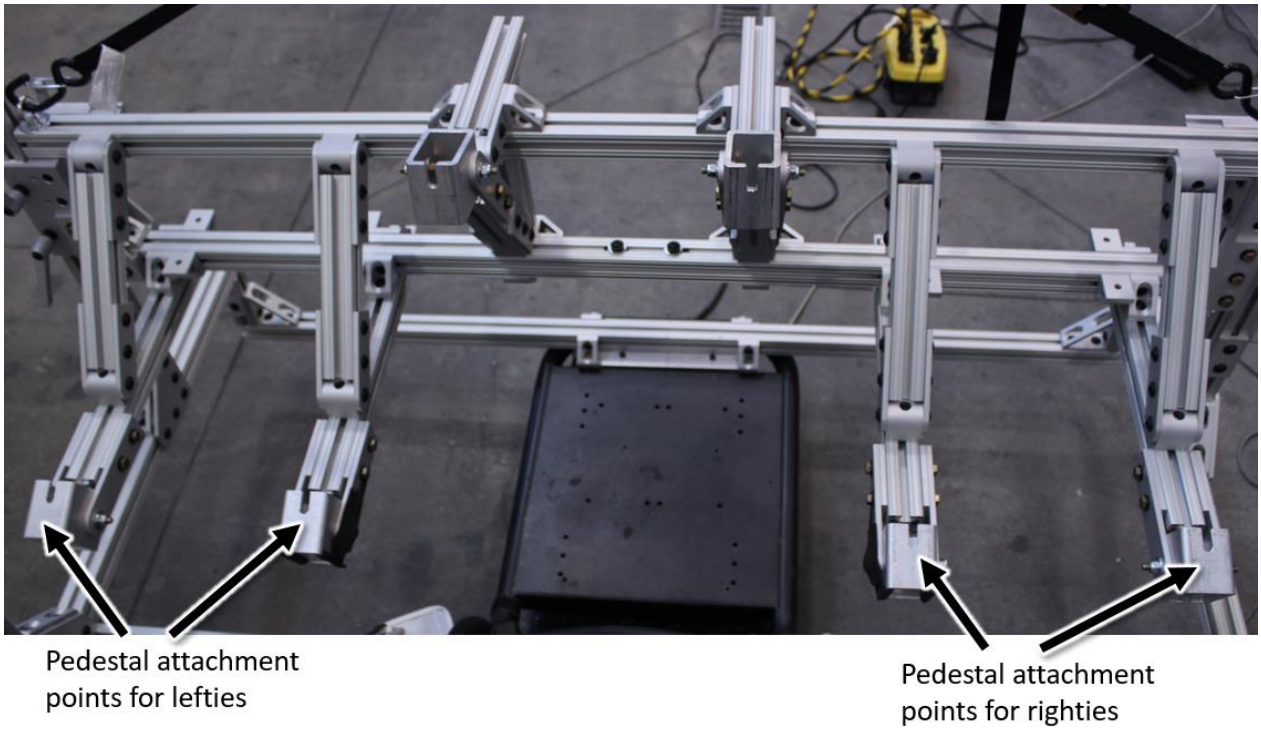


Figure A.4 – For righty and ambidextrous participants, the pedestal screen was placed to their right. For lefty participants, the pedestal screen was placed to their left.

It was assumed that the eye-to-back distance remained relatively constant between participants. The longitudinal (forward-backwards) position of the chair was kept constant between participants.

To set the longitudinal position of the chair, one of the investigators sat on the chair, with the screens adjusted to the appropriate vertical height. They held a 27 inch (68.6 cm) stick, with one end positioned beside their eye, and the other end pointed towards the center of the MIP screen. They then adjusted the chair backwards and forwards until the far end of the stick touched the center of the MIP screen (with the other end still positioned directly beside their eye). It was also verified that the angle of the stick was at 28°. This longitudinal chair position was then maintained for all participants.

APPENDIX B CHOOSING HAND SUPPORT METHOD

We originally planned to use cylindrical drawer pull handles as hand supports, around each screen. These are displayed in the following images:

Figure B.1 – Drawer pull handles surrounding the large touch monitor. We initially planned to use these as hand supports.

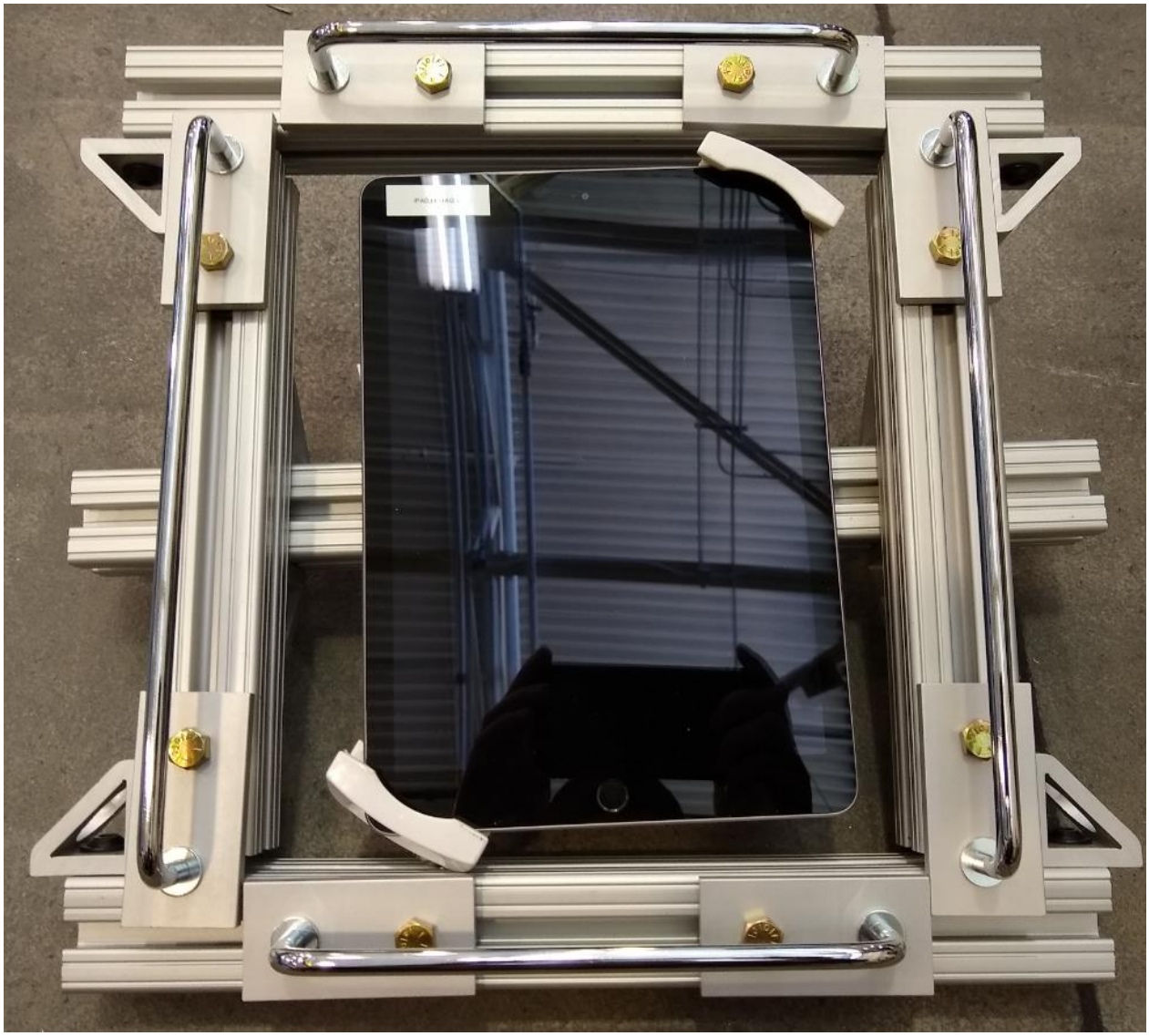


Figure B.2 - Drawer pull handles surrounding the iPad. We initially planned to use these as hand supports.

Figure B.3 – Proposed placement of the drawer pull handles around the small screen. This image just shows two handles, placed to the left and bottom of the screen, but there would have also been handles placed to the right and top, as well. None of these handles were ever installed.

However, due to the shape of the screens and frames, it was difficult to install these handles in a proper ergonomic position. In the above images, the base of each handle is mounted nearly flush with the screen, while the handle tops (which support the hands) are higher than the screen. This meant that the fingers of the hand were above the screen. In order for the thumb to come into

contact with the screen, the thumb needed to reach down below the fingers. This proved to be uncomfortable.

It was more comfortable to have the fingers grab behind the screen surface, so that the thumb could naturally be positioned at screen-height (Figure B. 4).

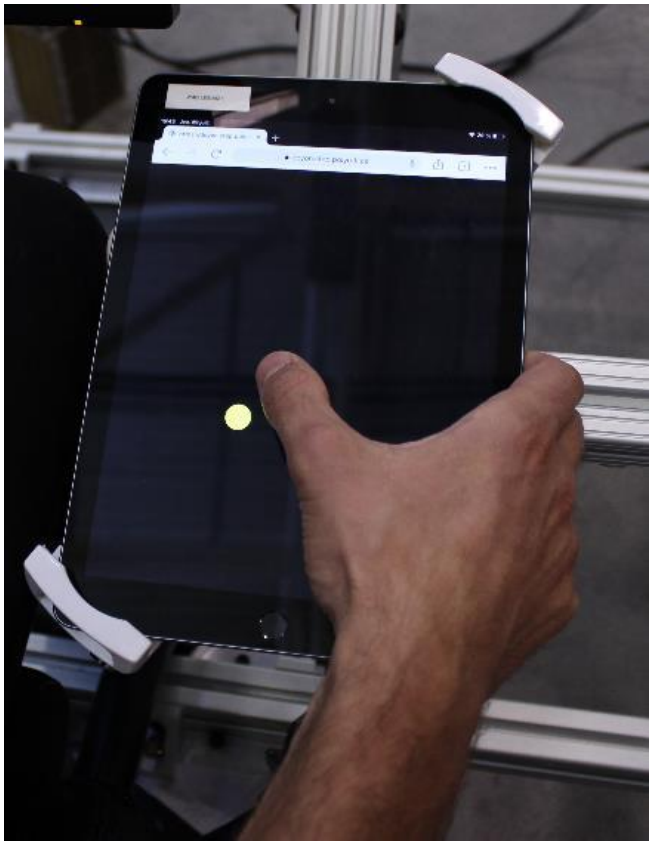


Figure B.4 – Fingers of the hand hold onto the screen from behind.

In a real aircraft, the avionic screens would normally be mounted into a panel. As such, the black, plastic bezel would stick outwards from the panel, while the metallic silver area would be hidden. This is shown in the following images.

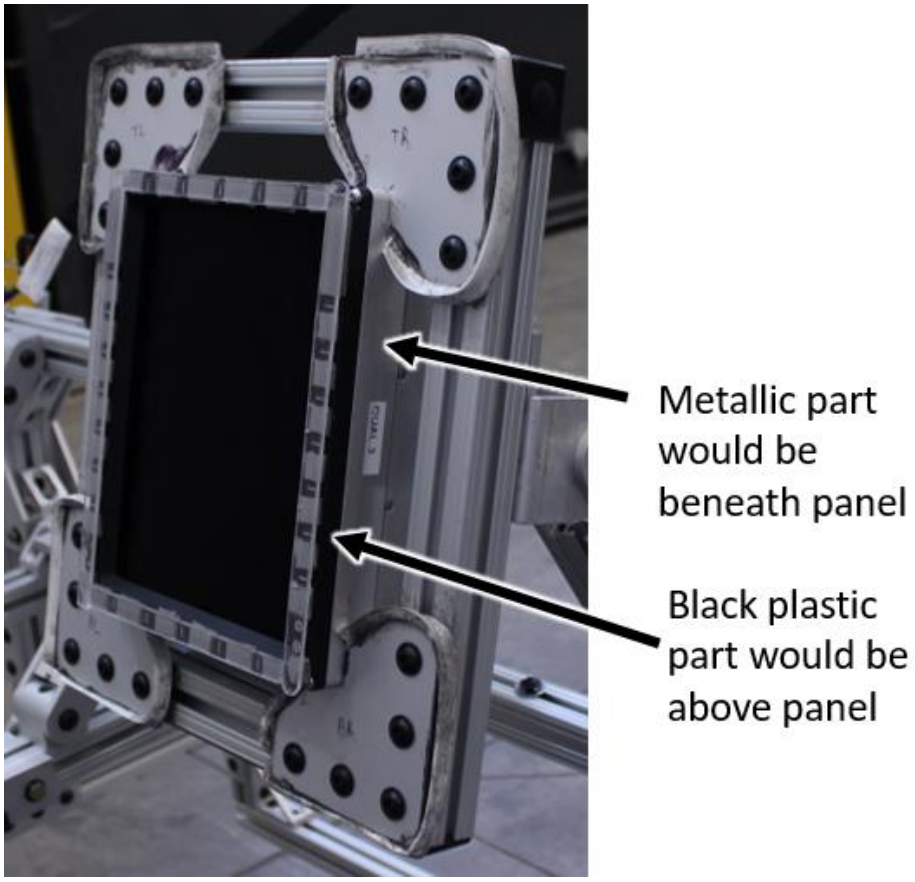


Figure B.5 – In a real aircraft, the metallic part would be hidden behind the panel, while the black plastic part would jut out from the panel.

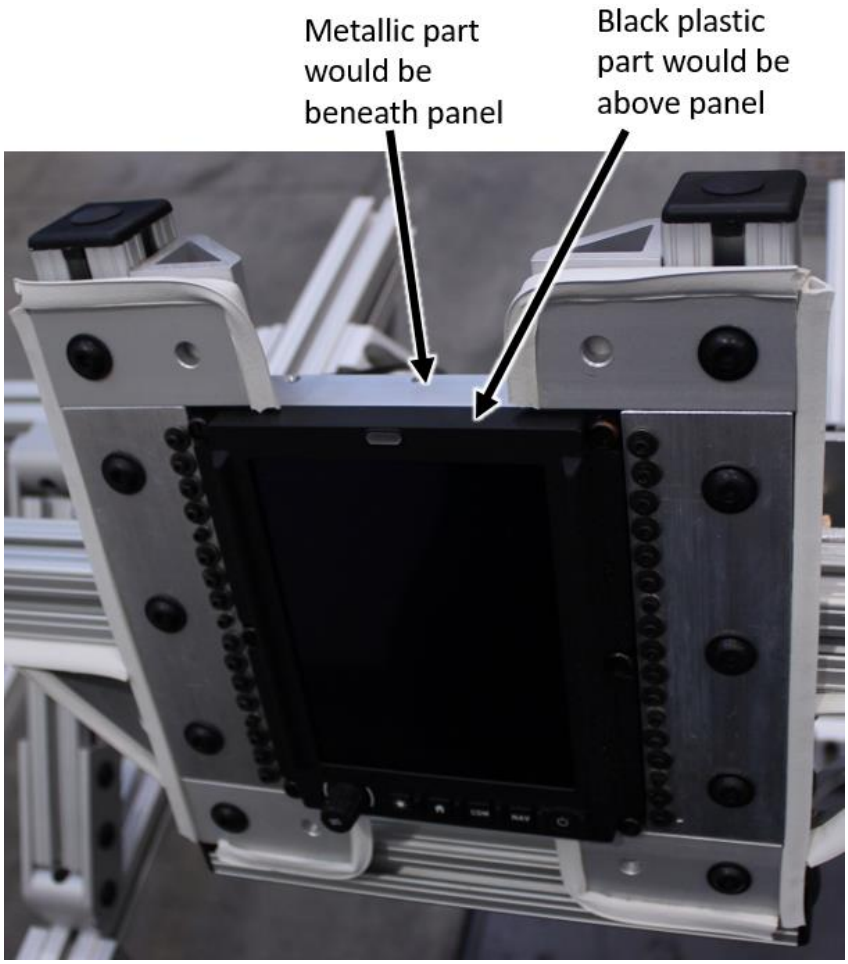


Figure B.6 - In a real aircraft, the metallic part would be hidden behind the panel, while the black plastic part would jut out from the panel.

In this study, participants were allowed to hold onto both the metallic and black plastic parts if they so chose. The justification for allowing participants to do this is that, if the aircraft manufacturer so chose, we believe that an inset handle could be added to the panel, allowing pilots to reach into it.

In either case, participants tended not to grab too deeply onto the metallic area of the screen, since this reduced their range of thumb motion. This was especially true for the MFD screen, since the furthest targets were 6 in from the screen edge. In order to be able to reach far targets participants tended to support their two middle fingers on the edge of the screen, and did not grab beneath it too deeply.

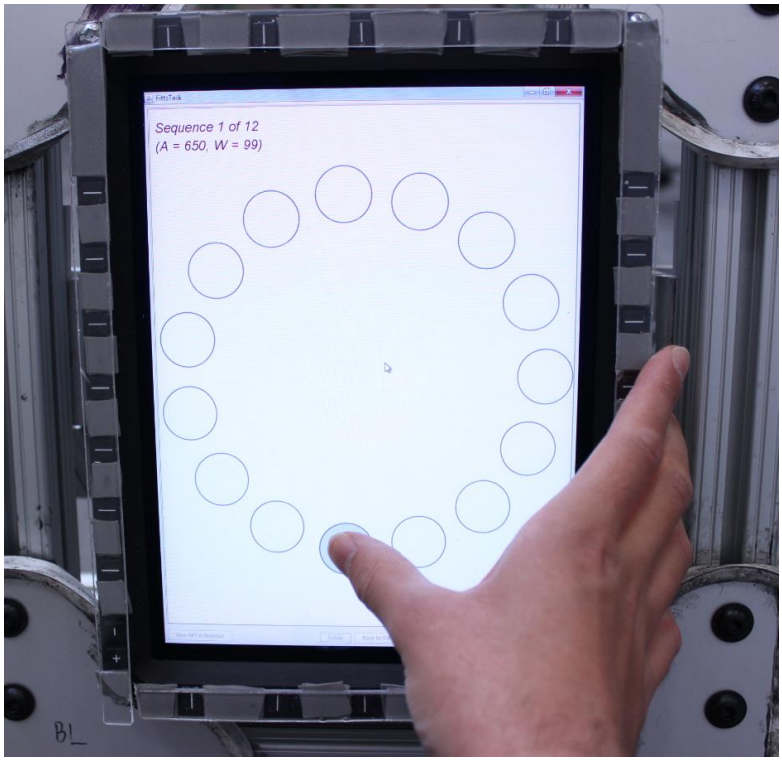


Figure B.7 – On the MFD screen, participants tended to rest their two middle fingers on the edge of the screen, without grabbing too deeply, to allow more range of thumb motion.

On the MCDU screen, participants were able to grab more deeply, since the farthest targets were 4 in from the screen edge (due to the screen being much smaller). However, they still tended not to reach too deeply into the metallic area, in order to allow for an appropriate range of thumb motion.

Figure B.8 - Participants were able to grab more on the MCDU screen.

APPENDIX C CONVERTING FROM ACCELERATION TO POSITION

D-Box DevSim, which was used to program the motion of the chair, only accepts position points as input. This meant converting the accelerometer data into position data. By taking the discrete Fourier transform of the accelerometer data, it is possible to convert the data into a sum of thousands of cosine waves. The discrete Fourier transform provides a series of complex numbers, associated with a series of frequencies. Taking the magnitude of the complex number, and multiplying it by 2, provides the amplitude of each cosine wave. Taking the phase of each complex number provides the phase offset of each cosine wave. The frequency associated with the data point is multiplied by 2π to become the angular frequency of the cosine wave [76].

The acceleration data is now represented in a function with the following form:

$$a = A_1 cos_1(2\pi f_1 t + \varphi_{1)} + A_2 cos_2(2\pi f_2 t + \varphi_2) + \cdots + A_n cos_n(2\pi f_n t + \varphi_{n)}$$

Where a represents acceleration, A_i represents the amplitude of each cosine wave, f_i represents the frequency of each cosine wave, t represents time, φ_i represents the phase offset of each cosine wave, and n represents the number of cosine waves in the equation.

Once the acceleration data has been converted into a sum of cosine waves, it is relatively easy to convert it to position data. The cosine waves can be integrated twice, with respect to time, to go from acceleration to position. In essence, each cosine wave takes a negative, and the amplitude of each cosine wave is divided by $(2\pi f_i)^2$.

Once this has been done, position is now represented as a sum of cosine waves:

$$D = \frac{A_1}{(2\pi f_1)^2} cos_1(2\pi f_1 t + \varphi_{1)} + \frac{A_2}{(2\pi f_2)^2} cos_2(2\pi f_2 t + \varphi_2) + \cdots + \frac{A_n}{(2\pi f_n)^2} cos_n(2\pi f_n t + \varphi_{n)}$$

Where D represents position (for "displacement").

The above formulas are for continuous functions, but DevSim accepts discrete position points at a maximum rate of 400 samples per second [77]. To output a series of discrete points, the time variable, in the above formula, can be replaced by:

$$t = \frac{k}{400\ Hz}$$

Where k is simply a sequence of integers, and 400 Hz is the sampling frequency of DevSim.

APPENDIX D FREQUENCY LIMITATIONS OF THE CHAIR AND CORRECTIONS IMPLEMENTED

D-Box DevSim was used to program the motion of the chair. DevSim's technical documentation presents the following specifications [77]:

- Can replicate frequencies up to a maximum of 30 Hz
- Sampling frequency of 400 Hz (maximum frequency at which position points can be sent to the chair)

In practice, however, D-Box's DevSim had a much lower maximum frequency. The following shows a frequency response graph of the chair, without the test structure (the frequency response graph with the test structure will be shown later). This graph was made by using DevSim to program the chair to vibrate at 1 m/s^2 at each integer value frequency, from 1 – 22 Hz. An accelerometer reading was taken on the cushion of the chair, with someone sitting on top of it, for each integer value frequency. We used an enDAQ Slam Stick X 100g accelerometer, which has a built-in 16g MEMS sensor for low-g accelerations, with a sampling rate of 800 Hz. We then performed the Fourier transform of each accelerometer reading. The RMS of the acceleration was then calculated, in a range +/- 0.25 Hz around the target frequency. The output RMS, as measured using the enDAQ accelerometer, was then divided by the RMS of the input frequency, as programmed into DevSim.

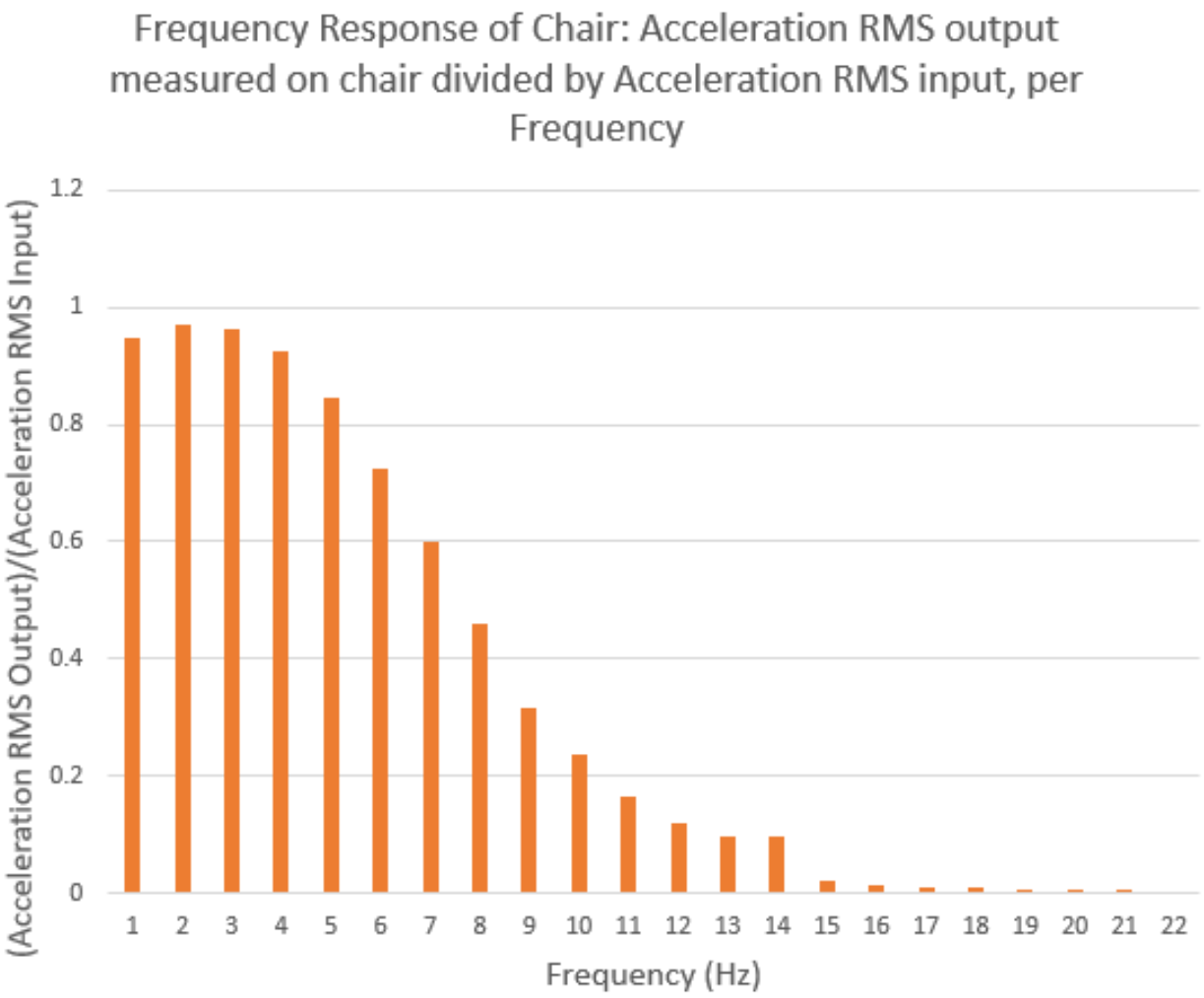


Figure D.1 - Frequency response graph of the GP Pro 500, without the test structure, using DevSim to program its motion.

The above graph shows that DevSim starts filtering frequencies higher than 5-6 Hz. By 12 Hz, the input frequency is almost entirely filtered out.

One way to counteract this filtering effect is to attempt to amplify the input signal by the inverse of the frequency response graph i.e.,amplify the vibration based on their frequencies.

However, the following graphs show the Fourier transform of the measured acceleration on the D-Box chair, at a sampling of different input frequencies As shown in these graphs, there is consistently an undesired frequency peak that appears. This “mirror frequency” peak is always mirrored around 12.5 Hz. If the desired frequency and the mirror frequency are added together, they always add up to 25 Hz, and they are always centered around 12.5 Hz. For frequencies

above 12.5 Hz, the mirror frequency peak is higher than the desired frequency. This means that, if amplification is used to counteract the filtering effect shown previously, the mirror frequencies will be similarly amplified, adding undesired vibration frequencies. This means that, in practical terms, even if amplification is used to counteract the filtering effect, D-Box DevSim has an upper limit of around 12 Hz.

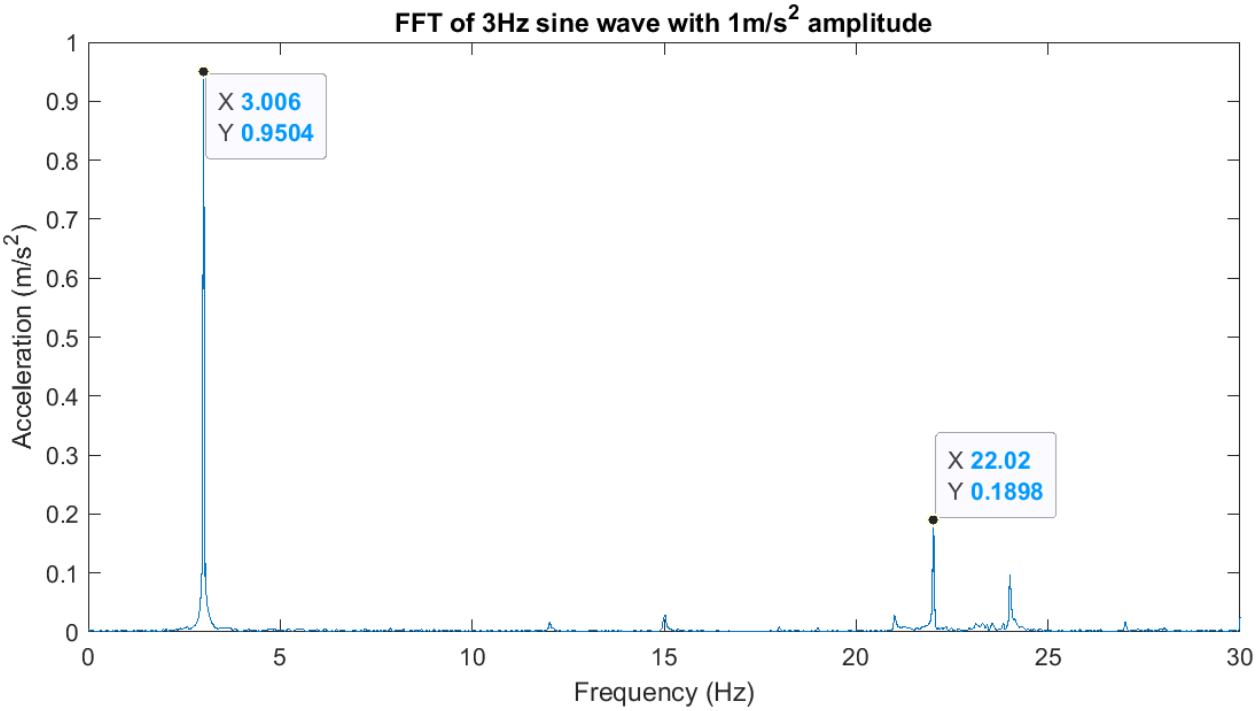


Figure D.2 – The desired frequency (frequency input into DevSim) is 3 Hz, in this graph. The mirror frequency is 22 Hz.

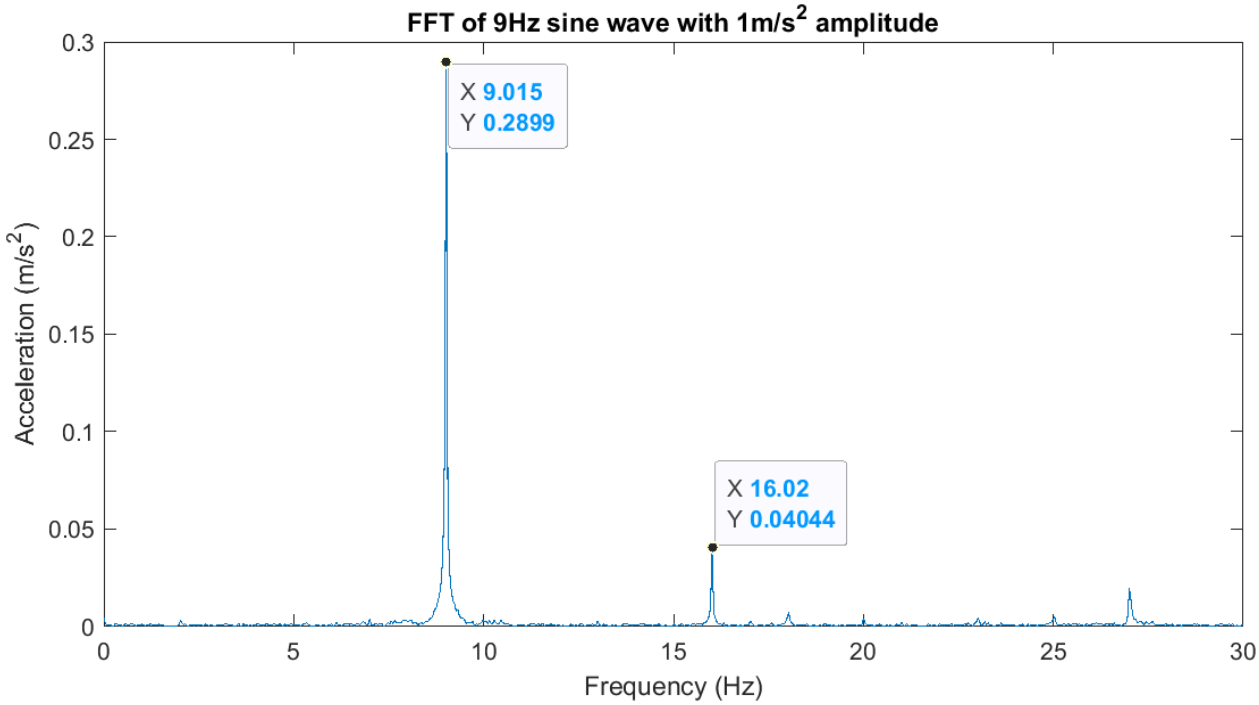


Figure D.3 - The desired frequency (frequency input into DevSim) is 9 Hz, in this graph. The mirror frequency is 16 Hz.

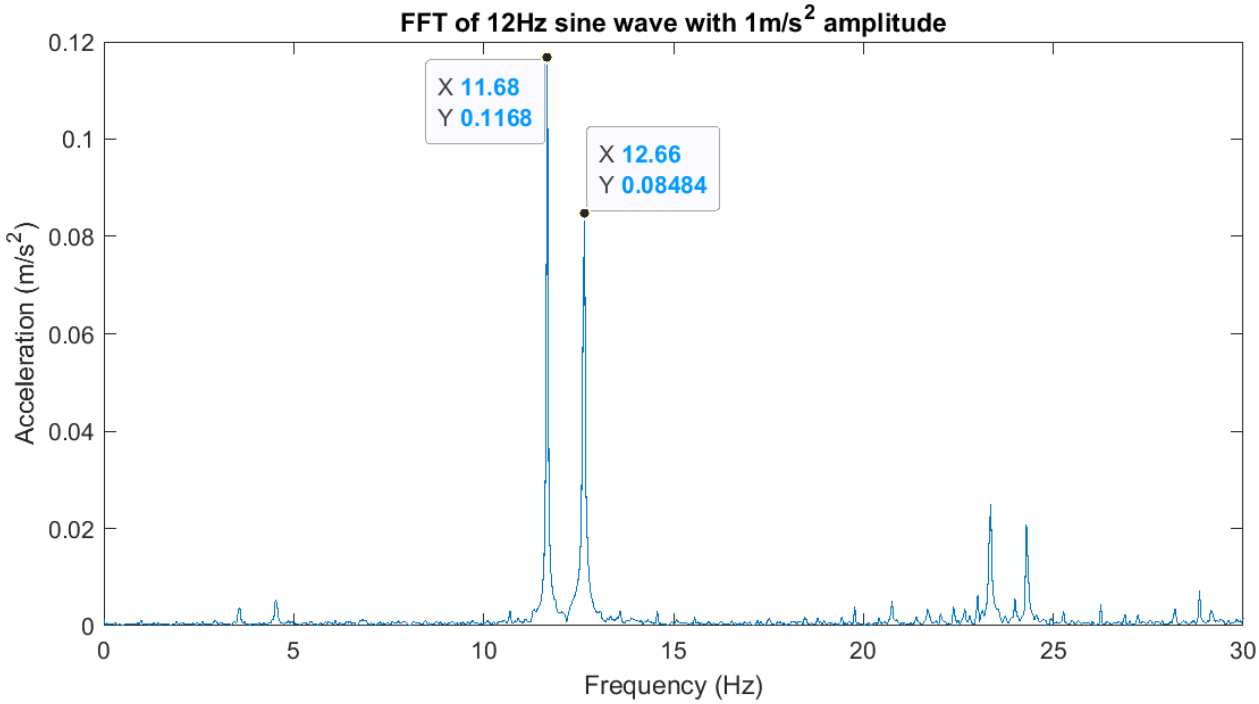


Figure D.4 - The desired frequency (frequency input into DevSim) is 12 Hz, in this graph. The mirror frequency is 13 Hz. Note that the magnitude of the desired and mirror frequencies are very close.

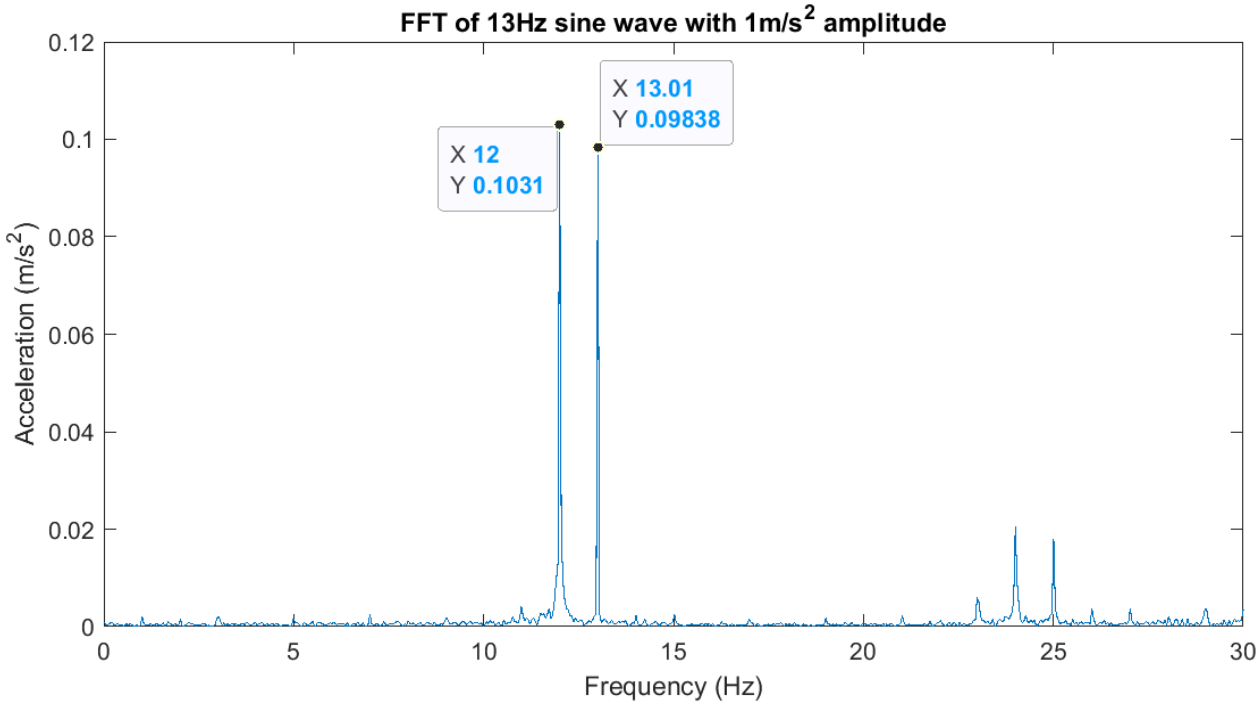


Figure D.5 - The desired frequency (frequency input into DevSim) is 13 Hz, in this graph. The mirror frequency is 12 Hz. Note that the magnitude mirror frequency has become higher than the desired frequency.

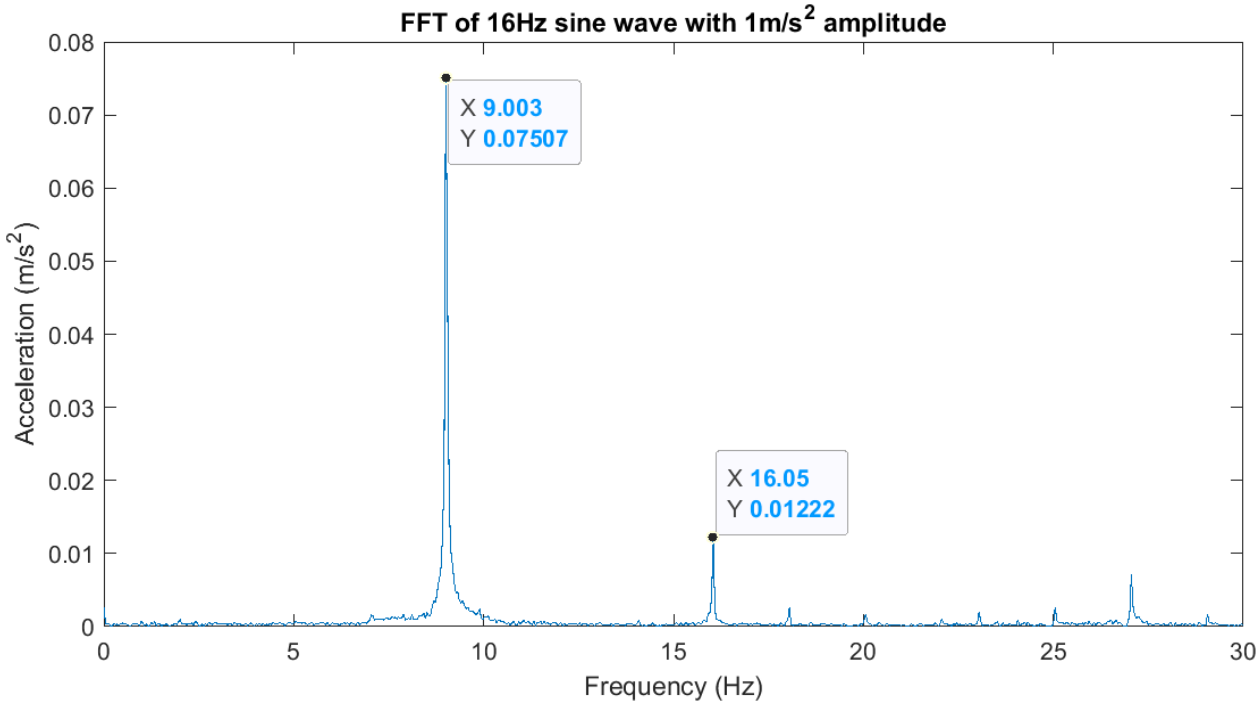


Figure D.6 - The desired frequency (frequency input into DevSim) is 16 Hz, in this graph. The mirror frequency is 9 Hz. Note that the magnitude of the mirror frequency is now higher than the desired frequency.

To validate these accelerometer readings, the output of the chair was checked by sitting on it. It was clear, by feel, that the magnitude of the vibration was greatly reduced at higher frequencies, being nearly imperceptible at 22 Hz. In addition, for higher frequency inputs, the output vibration clearly felt low frequency, validating the low frequency mirror peak identified by the accelerometer.

The accelerometer was also validated by measuring a high-amplitude, high-frequency test input, using D-Box's test software. The accelerometer correctly measured high-amplitude and high-frequency, when measuring this test vibration.

Compensating for DevSim Filters

As shown previously, D-Box DevSim filters frequencies starting around 6 Hz. In addition, DevSim creates undesired mirror frequencies. After 12.5 Hz, the amplitude of the mirror frequency peak becomes higher than the amplitude of the input frequency.

Ultimately, this means two constraints:

- Undesired mirror frequencies are inevitable and could not be fully avoided. But they can be minimized, by choosing the narrowest acceptable frequency range for the vibration profile. The narrowest acceptable frequency range, in this case, was 0.85 to 11 Hz (this range will be justified below).
- Frequencies from 6Hz to 11 Hz need to be amplified, to counteract DevSim's filtering. This was done by multiplying the amplitude, at each frequency, by a linear interpolation of the measured frequency response graph of the chair with the test structure on it.

The frequency response graph of the base D-Box chair, using the DevSim control system, was shown previously. However, that previous graph showed the frequency without the test structure built on top of it. The test structure added certain harmonics, which changed the frequency response of the chair. Here is the frequency response of the D-Box chair, running DevSim, with the test structure built on top of it:

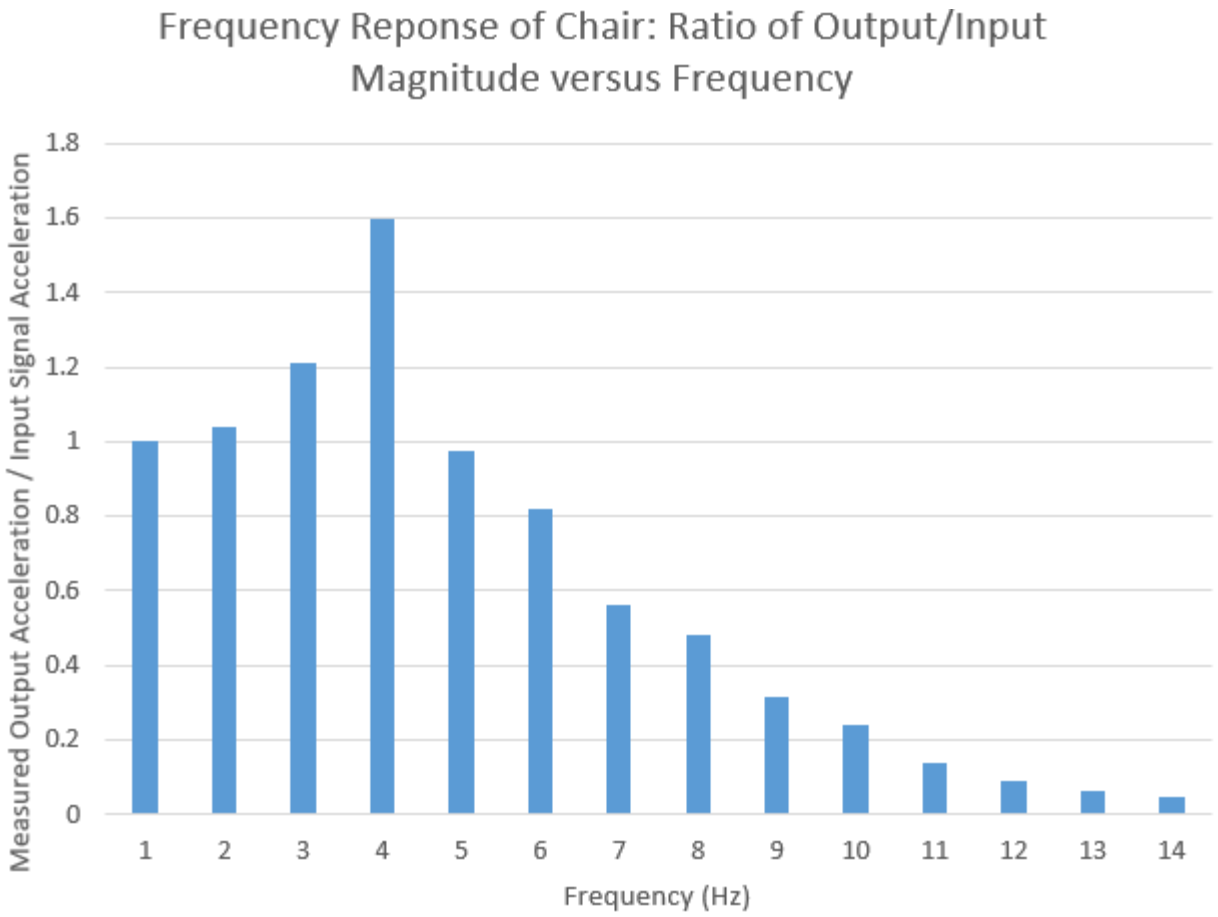


Figure D.7 – Frequency response of the D-Box chair with the test structure.

This graph was used to generate amplification factors to counteract the filtering effects of DevSim. While it shows that there is a large amplification effect at 4 Hz (likely due to it being close to one of the harmonics of the structure), this amplification effect was not as pronounced when playing the entire frequency profile. Through trial-and-error, it was found that using an amplification factor of 1 (so no amplification factor) worked best for frequencies between 0-4 Hz. In addition, frequencies higher than 12 Hz were filtered out of the final profile, to reduce the amount of undesired mirror peaks. Frequencies higher than 12 Hz were not amplified. Therefore, amplification factors were only applied to frequencies from 4 Hz-12 Hz. Linear interpolation was used to calculate the amplification factor between points (see Table D. 1).

Table D.1 – Amplification factors applied to the input signal to compensate the filtering.

Frequency (Hz)	Ratio of measured output acceleration over input signal acceleration (A.U.)	Amplification Factor
1	1.002	1
2	1.041	1
3	1.213	1
4	1.597	1
5	0.975	1.025
6	0.817	1.223
7	0.561	1.782
8	0.482	2.075
9	0.316	3.165
10	0.240	4.170
11	0.137	7.306
12	0.093	1
13	0.062	1
14	0.046	1

An 8^{th} order low-pass Butterworth filter, with a cut-off frequency of 13 Hz, was applied to the accelerometer data to remove high frequency from the signal. An 8^{th} order high-pass Butterworth filter, with a cut-off frequency of 0.85 Hz was also applied to the data, to remove very low frequency movements that might exceed the displacement limits of the chair. In addition, a brick wall filter was applied for frequencies lower than 0.01 Hz to ensure that frequencies close to 0 Hz were eliminated, to avoid running up against the chair's displacement limit. A brick wall filter was applied for frequencies above 100 Hz to ensure that frequencies higher than the chair's vibration limit were also zeroed out.

APPENDIX E MODIFICATIONS DONE TO THE STRUCTURE TO CONTROL FOR INTERNAL VIBRATION

The structure that holds the screens is built onto a cantilever, which sticks out in front of the motors. This cantilevered section of the structure flexes, during vibration. It is likely that this flexing structure has a harmonic around 10 Hz, as explained previously.

Figure E.1 – The front section of the structure is cantilevered out beyond the motors, and flexes during vibration.

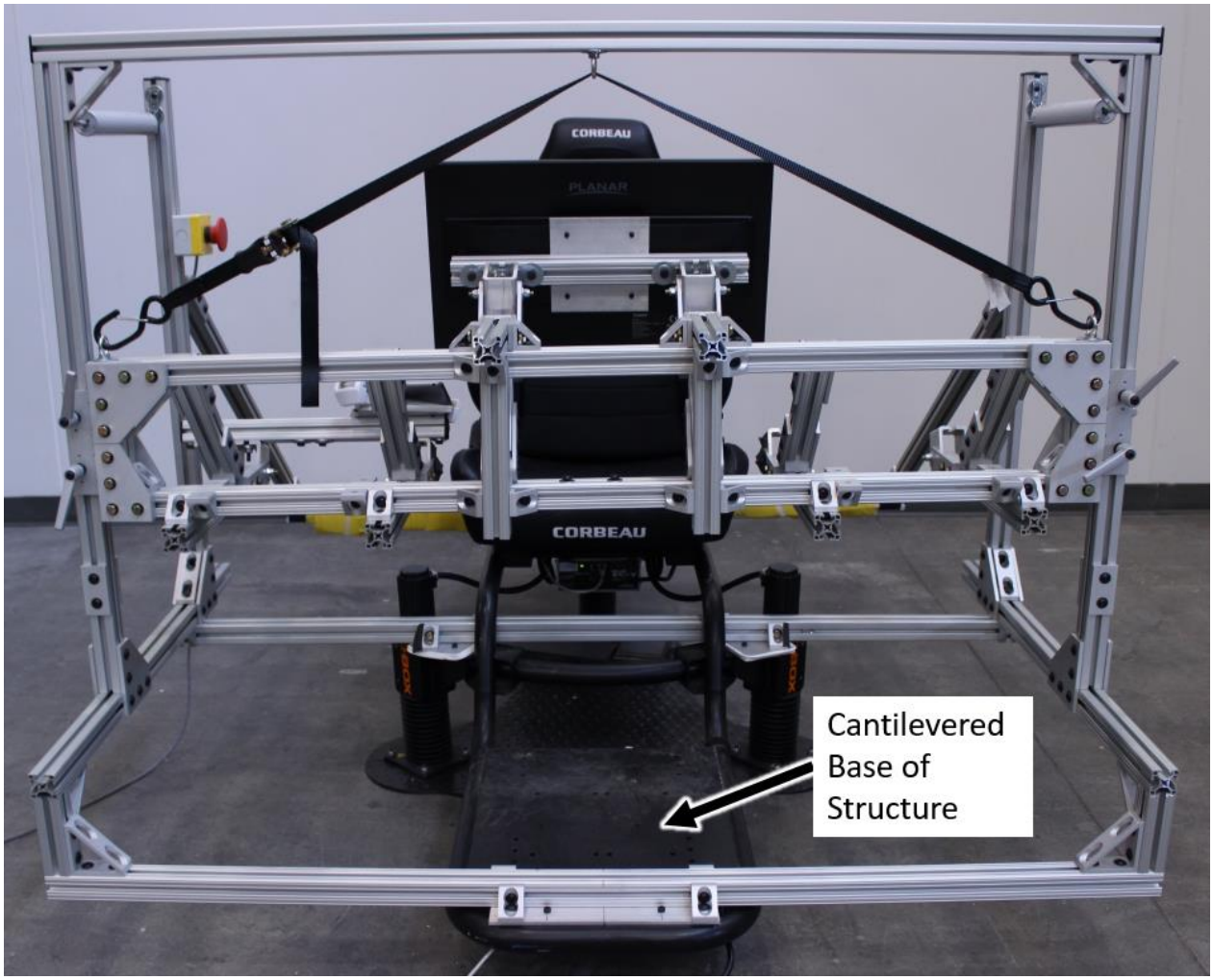


Figure E.2 – The cantilevered base on the structure, which flexes during vibration, after the structure was built up onto it.

The first version of the structure had the screen support columns connected rigidly to the rest of the structure, as shown in this image:

Figure E.3 - The first version of the structure, which had the screen support columns connected rigidly, on top, with the rest of the structure.

However, since the base still flexed, the screen support structure still caused a harmonic peak around 10 Hz. This harmonic peak was then translated to the rest of the D-Box chair, which caused a large harmonic peak to appear on the seat as well.

To dampen this 10 Hz peak and ensure that it was not transferred to the seat, vibration dampers were added to the final version of the structure. The following image shows these vibration dampers, which are made from pneumatic door closers:

Figure E.4 – Final version of the structure, with vibration dampers that dampen the 10 Hz harmonic peak and ensure that it is not transmitted to the seat.

Since most of the structure's weight wave positioned in front of the motors, counterweights were added, behind the back motor, to ensure that the structure was balanced. By balancing the structure, this ensured that it moved up and down uniformly, rather than rocking back and forth.

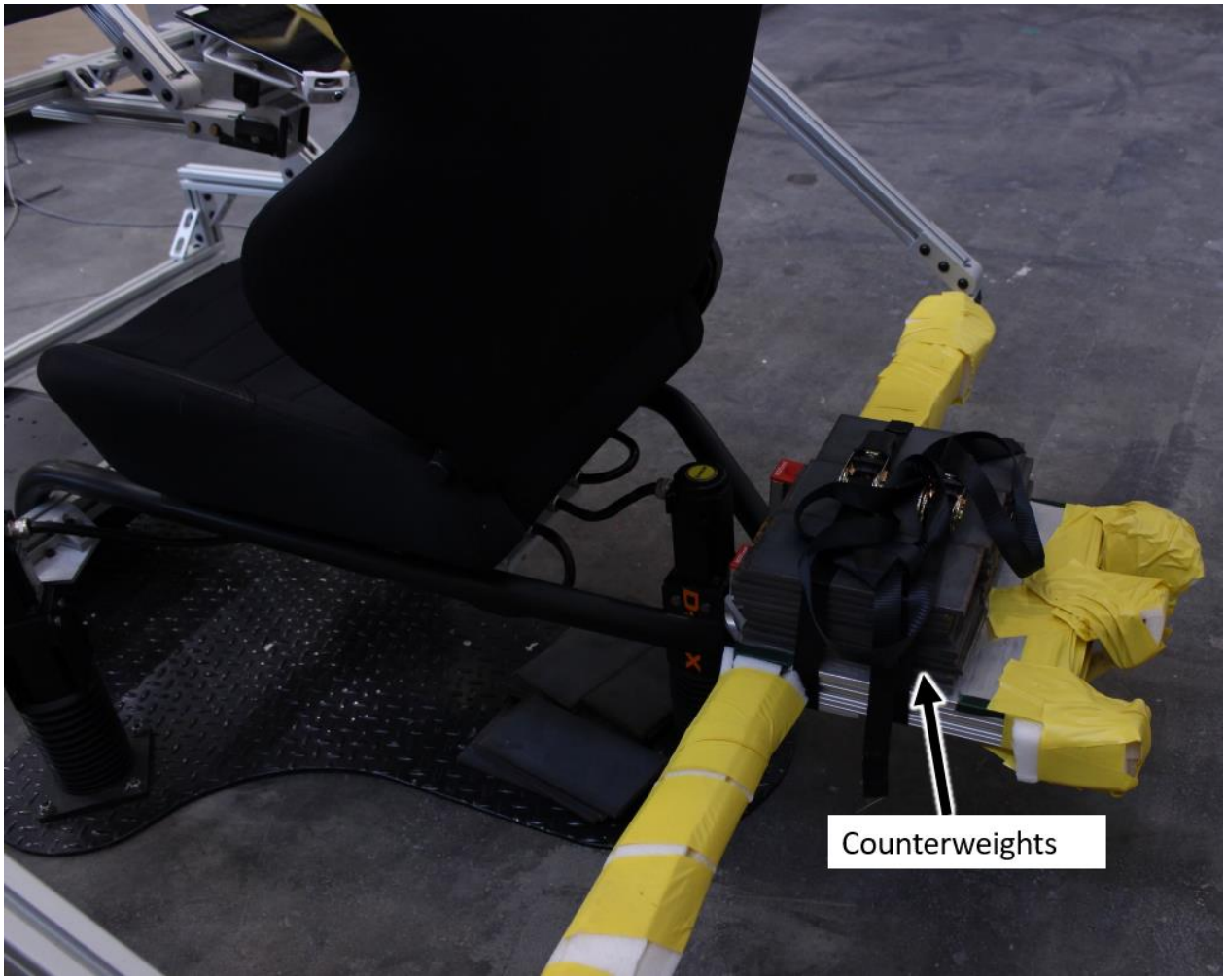


Figure E.5 – A stack of steel plates was added as counterweights behind the back motor, to ensure that the structure is balanced.

The front part of the final structure was also redesigned to reduce weight, and bring the center of mass closer to the motors, compared to earlier iterations, in an effort to further balance the structure and reduce the amount of counterweights needed.

APPENDIX F ETHICS COMMITEE APPROVAL - CER-1920-48-D

Montréal, le 2 avril 2020

Objet: Approbation éthique – « Preliminary test on the usability of avionic touchscreens under vibration » - Projet CER-1920-48-D

M. Adam Schachner,

J'ai le plaisir de vous informer le Comité d'éthique de la recherche, selon les procédures en vigueur, en vertu des documents qui lui ont été fournis, a examiné le projet de recherche susmentionné et conclu que ce dernier répond aux normes en vigueur au chapitre de l'éthique de la recherche énoncées dans la Politique en matière d'éthique de la recherche avec des êtres humains de Polytechnique Montréal.

Veuillez noter que le présent certificat est valable pour une durée d'un an, soit du 2 avril 2020 au 3 avril 2021, pour le projet tel qu'approuvé au Comité d'éthique de la recherche avec des êtres humains.

Veuillez noter que conformément aux exigences auxquelles l'institution et son personnel sont assujettis afin d'être admissibles aux fonds des organismes subventionnaires, il est de votre responsabilité de déposer au CÉR un rapport annuel ou un rapport final avant l'expiration de la présente approbation éthique afin de l'informer de l'avancement de vos travaux. Le formulaire à remplir est disponible à l'adresse suivante : (http://www.polymtl.ca/recherche/formulaires-et-guides).

De plus, il est de votre responsabilité d'informer le CER de toute modification importante qui pourrait être apportée au protocole expérimental avant sa mise en œuvre, de même que de tout élément ou évènement imprévu pouvant avoir une incidence sur le bien-être ou l'intégrité des participant(e)s impliqué(e)s dans le projet de recherche. Nous vous invitons aussi à nous signaler tout problème susceptible d'avoir une incidence sur les membres de l'équipe de recherche.

Je vous souhaite bonne chance dans la poursuite de vos travaux.

Nous vous prions d’agréer, Monsieur, l’expression de nos sentiments les meilleurs,

Farida Cheriet, présidente
Comité d’éthique de la recherche
Polytechnique Montréal

c.c. Direction de la formation et de la recherche; Service des Finances
Philippe Doyon-Poulin, professeur adjoint, Département de mathématiques et génie industriel
Daniel Imbeau, professeur titulaire, Département de mathématiques et génie industriel
Cochercheurs
p.j. Certificat # CER-1920-48-D

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

CER-1920-48-D
CERTIFICAT D'APPROBATION ÉTHIQUE

Le Comité d'éthique de la recherche de Polytechnique Montréal, selon les procédures en vigueur, en vertu des documents qui lui ont été fournis, a examiné le projet de recherche suivant et conclu qu'il respecte les règles d'éthique énoncées dans sa Politique en matière d'éthique de la recherche avec des êtres humains.

Projet

Titre du projet

Preliminary test on the usability of avionic touchscreens under vibration CER-1920-48-D

Étudiant requérant **Adam Schachner**, Candidat à la maîtrise, Département de mathématiques et génie industriel

Sous la direction de: **Philippe Doyon-Poulin, professeur adjoint**, Département de mathématiques et génie industriel, Polytechnique Montréal &
Daniel Imbeau, professeur titulaire, Département de mathématiques et génie industriel, Polytechnique Montréal.

Avec la collaboration de: **Nami Bae (CMC Electronics)**

Coordination du projet: Alexandre Ferreira Benevides (LESIAQ)

Financement	
Organisme	MITACS / CMC Electronics
No de UBR	3320456

Programme	Bourse Accelaration
No d'octroi:	IT15812
Titre original de l'octroi:	Evaluation of human performance using touch screen displays in cockpit environment
Chercheur principal:	Philippe Doyon-Poulin

MODALITÉS D'APPLICATION

Toute modification importante qui pourrait être apportée au protocole expérimental doit être transmise au Comité avant sa mise en œuvre.

L'équipe de recherche doit informer le Comité de tout élément ou évènement imprévu pouvant avoir une incidence sur le bien-être ou l'intégrité des participant(e)s impliqué(e)s dans le projet de recherche ainsi que tout problème susceptible d'avoir une incidence sur les membres de l'équipe de recherche.

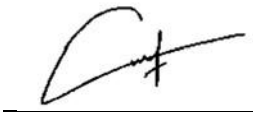
Selon les règles universitaires en vigueur, un suivi annuel est minimalement exigé pour maintenir la validité de la présente approbation éthique, et ce, jusqu'à la fin du projet. Le questionnaire de suivi est disponible sur la page web du Comité.

Date de délivrance **:** avril 2020
Date de fin de validité : 1er mai 2021

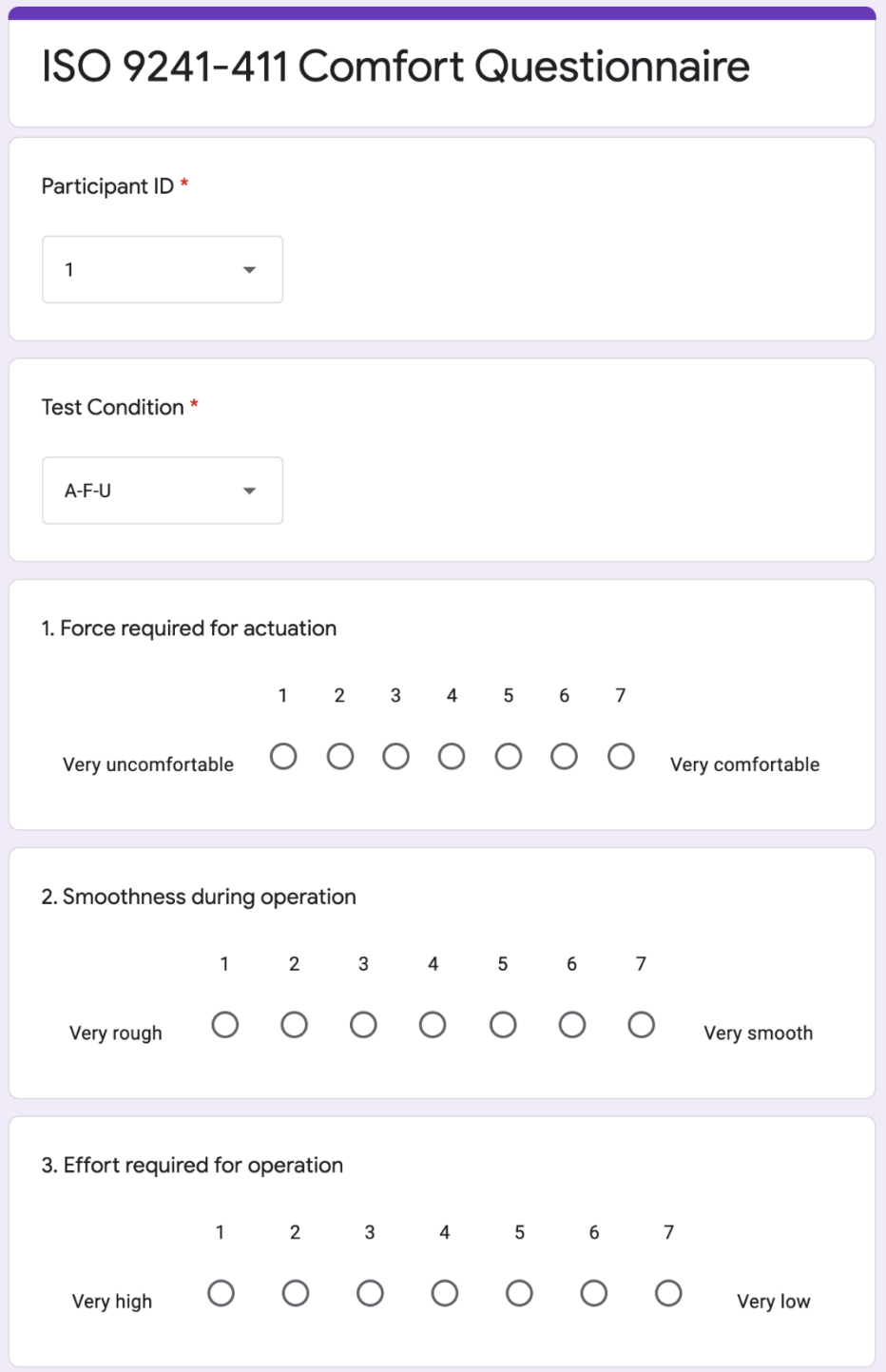
Farida Cheriet, présidente
Comité d'éthique de la recherche
Polytechnique Montréal

Date du prochain suivi **:**
3 avril 2021

Comité d'éthique de la recherche
Campus de l'Université de Montréal
avec des êtres humains Adresse postale 2900, boul. Édouard-Montpetit
Tél.: 514 340-4711 poste : 3755
C.P. 6079, succ. Centre-Ville 2500, chemin de Polytechnique
Fax : 514 340-4992 Montréal (Québec)
Canada H3C 3A7 Montréal (Québec)
Canada H3T1J4
Courriel : ethique@polymtl.ca



APPENDIX G ISO 9241-411 COMFORT QUESTIONNAIRE



ISO 9241-411 Comfort Questionnaire

Participant ID *

1

Test Condition *

A-F-U

1. Force required for actuation

	1	2	3	4	5	6	7	
Very uncomfortable	○	○	○	○	○	○	○	Very comfortable

2. Smoothness during operation

	1	2	3	4	5	6	7	
Very rough	○	○	○	○	○	○	○	Very smooth

3. Effort required for operation

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	Very low

4. Accuracy

	1	2	3	4	5	6	7	
Very inaccurate	○	○	○	○	○	○	○	Very accurate

5. Operation speed

	1	2	3	4	5	6	7	
Unacceptable	○	○	○	○	○	○	○	Acceptable

6. General comfort

	1	2	3	4	5	6	7	
Very uncomfortable	○	○	○	○	○	○	○	Very comfortable

7. Overall operation of input device

	1	2	3	4	5	6	7	
Very difficult to use	○	○	○	○	○	○	○	Very easy to use

8. Finger fatigue

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	None

9. Wrist fatigue

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	None

10. Arm fatigue

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	None

11. Shoulder fatigue

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	None

12. Neck fatigue

	1	2	3	4	5	6	7	
Very high	○	○	○	○	○	○	○	None

Submit

Clear form

APPENDIX H DEBRIEFING QUESTIONNAIRE

1) How did vibration impact your ability to complete the task?

1.a) How much more effort did it require to perform the selection task (Fitts) under vibration compared to no vibration? i.e. slightly more, moderately more, much more?

Did having a hand support help or hinder you?

Did you find that there was an optimal method for holding onto the screen or hand support, during periods of vibration, to help you complete the selection task?

2) Was there anything you liked/disliked about the current design of the hand support? Would you recommend any changes to it?

3) Which of the screens did you like using the most? Why?

4) Which of the screens did you most dislike using? Why?

5) Which screen position was the easiest to use? (Main instrument panel, pedestal) Why?

6) Which screen position did you find the most difficult to use? Why?

7) Did you have any difficulty reaching any of the screens to interact with them?

8) I've finished asking all my questions. Do you have any other comments that you would like to share?