The Leap Motion Movement for 2D Pointing Tasks

Characterisation and Comparison to Other Devices

Manuel César Bessa Seixas[1,2], Jorge C. S. Cardoso[1] and Maria Teresa Galvão Dias[3]
[1] *CITAR/School of Arts, Portuguese Catholic University, Porto, Portugal*
[2] *FEUP, University of Porto, Porto, Portugal*
[3] *INESC TEC/FEUP, University of Porto, Porto, Portugal*

Keywords: Interaction Device, Leap Motion, HCI, Pointing Devices, Performance Evaluation, Selection Tasks, Performance Measurement.

Abstract: In this paper we present the results from an experiment designed to characterize the Leap Motion movement in 2D pointing tasks and compare it to a mouse and touchpad. We used the ISO 9241-9 multi-directional tapping test for comparing the devices, and we analyse the results using standard throughput and error rate measures as well as additional accuracy measures such as target re-entry, task axis crossing, movement direction change, orthogonal direction change, movement variability, movement offset, and movement error. We also present the results from the ISO 9241-9 assessment of comfort questionnaire, and our observations from the participant's postures when using the Leap Motion device. Results indicate that the Leap Motion performs poorly in these tasks when compared to a mouse or touchpad.

1 INTRODUCTION

The Leap Motion (LM) controller is a new 3D sensing device for hand gesture interaction with a computer. It is capable of sensing the position and orientation of the fingers of the hands, as well as the palm orientation and curvature. The LM is a small device that sits on top of the computer desk and is operated by positioning the hands over the device. The controller can be used to point to a computer screen with a finger or with a tool (a pen or pencil, for example), or perform other hand gestures. The LM controller is also integrated in the HP ENVY 17t-j100 Leap Motion QE CTO Notebook PC and in the HP Leap Motion keyboard.

It is often depicted as a controller for pervasive and natural user interaction scenarios, allowing new ways to interact with a computer. Games, music controllers, 3d modelling, are examples of applications that have been created with new interaction paradigms to take advantage of this new controller. For example Ethereal (Crispy Driven Pixels Inc., 2014) is a Photoshop add-on that allows users to draw with their fingers, controlling the thickness of the line with the distance of the finger to the screen. Geco MIDI (Uwyn, 2013) is a software that transforms LM gestures into MIDI (Musical Instrument Digital Interface) messages for music composition with any MIDI enabled software. For example, DJ's can use it to add special effects in real-time to the music, by simply waving a hand.

Although not meant to be a replacement of the mouse, many of the interactions with the LM involve pointing and selecting targets on a computer screen.

Many applications in the Leap App Store are meant to give users various degrees of control over the computer, from selection and launching predefined applications and settings to scrolling content on webpages. Some applications even emulate the mouse, allowing cursor control and mouse actions (Lab, 2014; Leap Motion Inc., n.d.; Nu-Tech, 2014; Touchless, 2014).

Many applications that take advantage of the LM device still require users to perform typical WIMP tasks at some point (in many cases giving users the option of using the mouse or the LM device). For example, in many games users still need to select options and activate buttons; some software for surgery rooms also provides cursor control for specific functions (Manolova, 2014);

If we assume that the LM device gains commercial traction and becomes embedded in additional laptop computers and desktop keyboards, we must also assume that it will become an additional alternative to typical WIMP tasks. In a

Seixas M., Cardoso J. and Galvão Dias M..
The Leap Motion Movement for 2D Pointing Tasks - Characterisation and Comparison to Other Devices.
DOI: 10.5220/0005206100150024
In *Proceedings of the 5th International Conference on Pervasive and Embedded Computing and Communication Systems* (PECCS-2015), pages 15-24
ISBN: 978-989-758-084-0

situation where the user is operating the LM device in a specific LM task it may be faster to perform a WIMP task also with the LM, instead of moving the hand to operate the mouse.

However, up until now, there have been no studies about the performance of the LM device for 2D pointing tasks.

The objective of this work is to provide an initial assessment of the LM device for 2D pointing tasks and compare it with a mouse and touchpad. For this, we have performed an experimental evaluation using the ISO 9241-9 multi-directional tapping test (International Organization for Standardization, 2000) for pointing devices and calculated the various accuracy measures proposed by (MacKenzie, Kauppinen, & Silfverberg, 2001). We have also used the ISO 9241-9 assessment of comfort questionnaire to get a subjective device preference.

The contributions of this paper are:

- Characterization of the LM movement for pointing tasks in terms of the accuracy measures proposed by Mackenzie;
- Comparison of the LM movement and performance with the mouse and touchpad devices;
- An assessment of the subjective preferences and comfort of the LM device;
- An analysis of the postures adopted by users of the LM device.

2 RELATED WORK

2.1 The Leap Motion Device

The LM is a small input device controller (7.6 x 3 x 1.3 cm) developed by Leap Motion Inc., which detects and recognizes users' hands posture and gestures (Figure 1).

Figure 1: The Leap Motion device.

Programmers can use the Leap Motion SDK (available for C++, Java, Objective-C, C#, Python, Javascript, and other programming languages) to develop applications that take advantage of the device's capabilities. Currently, the SDK provides high-level functions such as:

- Presence/absence of hands within the range of the LM, and their 3D position in space.
- Orientation of the palms.
- Curvature of the palms.
- Overall scale, rotation, and translation motions calculated from the movement of the hands.
- Orientation of individual fingers (or tools such as pencils), and normalized 2D pointing position on the screen.
- Pre-defined gestures such as a finger tracing a circle, finger swipe, finger tapping movement, and screen tap.

Applications developed for the LM can be distributed via the Airspace store (Leap Motion Inc., 2014), an online store from which users may download applications to use with their device. Several applications are currently available, from games to productivity applications.

The LM driver software does not directly support user interaction with the Operating System (OS), but several applications in the Airspace store provide this capability. Touchless (Leap Motion Inc., n.d.), is an example of such applications, developed by Leap Motion Inc., with versions for Mac and Windows computers. Touchless provides several ways to interact with the OS:

- By pointing with a finger, users can control the position of the mouse cursor on the screen.
- By making a screen tap gesture (i.e., moving the finger towards the screen quickly), users can perform a mouse click.
- By swiping multiple fingers in the air, users can scroll horizontally or vertically.
- By pinching the fingers, users can zoom in and out.

2.2 Performance Evaluation of Input Devices

The most common evaluation measures for input devices are speed, accuracy, and throughput. Speed, or its inverse, movement time (MT), is the time it takes to select a target. Accuracy, usually reported as an error rate, is the number of target selections with the pointer outside the target over the total number of target selections. Throughput is a composite measure, expressed in bits per second, and derived from the movement time, target size, and distance to the target:

$$Throughput = \frac{ID_e}{MT}$$

ID_e is the effective index of difficulty, expressed in bits, and calculated from the distance to the target (D) and the effective width of the target (W_e):

$$ID_e = \log_2\left(\frac{D}{W_e} + 1\right)$$

W_e is calculated from the distance of target selection coordinates over a sequence of trials as $W_e = 4.133 * SD_x$, where SD_x is the standard deviation of the selection coordinates measured along the axis of approach to the target.

To help in testing the efficiency of input devices the ISO standard 9241 part 9, “Ergonomic design for office work with visual display terminals (VDTs) - Requirements for non-keyboard input devices” provides guidelines for testing and comparing pointing input devices. One of the tests proposed in the ISO standard is the multidirectional tapping test used to evaluate pointing movements in many different directions. In this test participants are required to move the cursor across a circle to sequentially numbered targets (see Figure 2). The targets (for example, squares, or circles) are equally spaced around the circumference of the circle and the sequence of targets to select is such that the movements are nearly equal to the diameter of the circle.

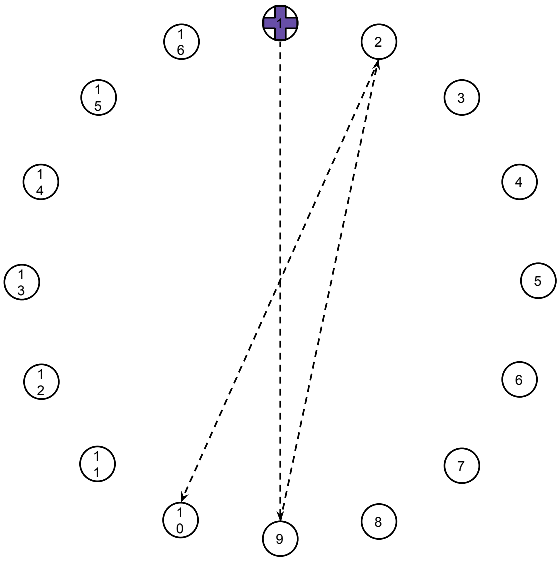

Figure 2: Multidirectional tapping test task.

ISO 9241-9 also provides subjective measures to assess the comfort and effort using the device. One of the questionnaires provided is an independent rating scale with 12 questions that users rate in a 1 to 5 scale:

1. Force required for actuation (very uncomfortable - very comfortable)
2. Smoothness during operation (very rough - very smooth)
3. Effort required for operation (very high - very low)
4. Accuracy(very inaccurate - very accurate)
5. Operation speed (unacceptable – acceptable)
6. General comfort (very uncomfortable - very comfortable)
7. Overall operation of the input device (very difficult (to use) - very easy (to use))
8. Finger fatigue (very high - very low)
9. Wrist fatigue (very high - very low)
10. Arm fatigue (very high - very low)
11. Shoulder fatigue (very high - very low)
12. Neck fatigue (very high - very low)

2.2.1 MacKenzie’s Accuracy Measures

(MacKenzie et al., 2001) proposed a set of seven accuracy measures for pointing devices that can complement the most common measure of throughput. The new measures proposed are Target Re-entry (TRE), Task Axis Crossing (TAC), Movement Direction Change (MDC), Orthogonal Direction Change (ODC), Movement Variability (MV), Movement Error (ME), and Movement Offset (MO). These measures capture aspects of the movement during a trial.

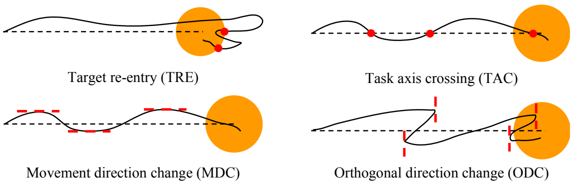

Figure 3: Graphical illustration of TRE, TAC, MDC, and ODC accuracy measures.

TRE measures the number of times the pointer enters the target region (area of the target), leaves it and re-enters again. Figure 3 illustrates a target selection with a TRE of two.

TAC measures the number of times the pointer crosses the task axis (a straight line from the initial pointer's position to the centre of the target).

MDC measures the number of times the pointer’s trajectory changes direction relatively to the task axis. For example, in Figure 3, there are three changes in the movement direction.

ODC is similar to Movement Direction Change. It measures the number of times the pointer’s trajectory changes direction along a perpendicular axis to the task axis. In Figure 3, there are four changes.

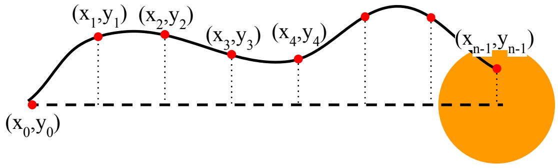

Figure 4: Path sampling.

MV is the standard deviation of the distances of the sample cursor positions to the task axis. It represents the extent to which the cursor positions lie in a straight line along an axis parallel to the task axis. Considering the task axis is transformed so that it is equal to y = 0 (see Figure 4), y_i is the distance between a sample cursor position and the axis, $\bar{y}$ is the mean distance of the sample cursor positions to the axis, and n is the number of sample positions:

$$\mathrm{MV} = \sqrt{\frac{\Sigma(y_i - \bar{y})^2}{n-1}}$$

ME is the average deviation of the sample cursor positions from the task axis, irrespective of whether the points are above or below the axis. If the task axis is y = 0, then:

$$ME = \frac{\Sigma|y_i|}{n}$$

MO is the mean deviation of the sample cursor to the task axis. If the task axis is y = 0, then:

$$MO = \bar{y}$$

2.3 Leap Motion Studies

(Weichert, Bachmann, Rudak, & Fisseler, 2013) analysed the accuracy and robustness of the leap motion controller. They performed an experiment where a robotic arm would hold a pen in its hand and was programmed to place the tip in several real world known positions. These positions would then be compared to the ones acquired by the LM controller, being the difference between each other the precision. These measures were repeated several times in order to find repeatability, for two cases: static and dynamic (with a moving pen). They found the accuracy of the LM to be less than 0.2mm for the static case and less than 1mm for the dynamic case. Weichert et al. focused on the accuracy of device itself; in this paper we focus on the accuracy of the user performing a task with the device.

(Vikram, Li, & Russell, 2013) present a new type of user input for writing, using the LM. Using the finger position data from the LM they are able to identify characters and words written “in the air”. They propose an algorithm that is capable of recognizing gestures without pen down/pen up gestures to mark the beginning and end of a gesture. Although their interaction technique relies on users performing finger gestures, their analysis is concerned with the gesture recognition algorithm. In this paper, we address the issue of the performance of doing the gestures (for simple pointing tasks).

(Nabiyouni, Laha, & Bowman, 2014) performed a usability testing in order to find which of the implemented 3D travel techniques was the most efficient in bare-hand interaction. Five techniques were tested in a set of 3 tasks and the interaction was performed through the use of the LM controller. The techniques developed were based on a “Camera-in-hand” metaphor, where the Leap Motion workspace was directly mapped to the virtual world, and an “Airplane” metaphor, that, similar to driving a vehicle, had the camera always moving straightforward being the user responsible for controlling its velocity and orientation (the orientation was the same as the hand). A 3D virtual scenario, modelled as a city, was used to perform the tests. This is an example of a task that is out of the scope of our evaluation since it uses LM-specific features that are outside of the WIMP paradigm.

(Manolova, 2014) describes a system for touchless interaction with medical images in surgery rooms using the LM device. Surgeons could manipulate image data using the open source Medical Imaging Toolkit (MITO). The system provided several functions such as scaling, zooming, and rotating, but also allowed the operator to manipulate the imaging software with traditional WIMP tasks: “*When the operator pointed one or two fingers towards the screen, the system drew a cursor on the screen so that the operator could point items or buttons in the imaging software, and when the operator moved the finger farther towards the screen, the pointed item was selected (similar to a mouse click)”* (Manolova, 2014, p. 5). This is the type of interaction that is the focus of the current paper: applications that take advantage of the LM’s gesture recognition for non WIMP interactions but that also allow the user to use the LM as a standard mouse, avoiding the use of a separate device (mouse) to control the software’s functions.

3 EXPERIMENT

The experiment was a 3 × 5 × 8 within-subjects factorial design:

- Device {Mouse, Touchpad, LeapMotion}
- Sequence {1,2,3,4,5}
- Block {1,2,3,4,5,6,7,8}

We configured the multi-directional tapping test with 16 circular targets, each with 13mm, in a circular layout with diameter of 180mm. The nominal index of difficulty used was 3.8 bits. The experiment was structured in "sequences" and "blocks." A sequence corresponded to 15 target selections (corresponding to a complete screen in the multidirectional tapping test – the first target did not count as it served only to start the sequence). A block had 5 sequences. Twelve participants were randomly assigned to one of three groups (4 participants/group). Each participant was tested with all devices. The order of devices differed for each group and was fully counter-balanced.

For testing the LM device, we used the Touchless application, which emulates mouse movement and mouse button presses, so the same software was used to collect device movement data for all three devices, at 40 samples per second.

At the beginning of the experiment we explained to participants the purpose of the experiment, the task to be performed, and the devices to be used. We also asked participants to fill in a questionnaire to determine the participant's computer literacy and experience with the devices. Age and gender were also asked.

We asked participants to perform the selection task as fast as possible without exceeding one error per sequence. Participants were allowed to perform practice trials until they felt ready to start the experiment. Participants used their preferred hand to operate the devices. Participants were also informed to take a break between sequences, if they so desired.

During the experiment, we observed and took notes about the participant's posture operating the devices. At the end of each device's trials we asked participants to fill in the 12 item ISO 9241-9 comfort and effort questionnaire. At the end, we asked participants to tell us which device they preferred best and which device they disliked the most. The experiment lasted about 1 hour and 15 minutes.

3.1 Participants

Table 1: Age distribution of participants.

Age interval	*Frequency*
[21-25]	1
[26-30]	3
[31-35]	5
[36-40]	1
[41-45]	1
[46-50]	0
[51-55]	0
[56-60]	1

Twelve non-paid participants (9 male, 3 female) were recruited. Their ages ranged from [21-25] to [56-60] years old (see Table 1). All participants were daily computer users. Most participants used the computer mouse every day (2 used the mouse only often and seldom). 6 participants used the touchpad every day, 3 used it often, 2 seldom, and 1 used didn't use it at all. The Leap Motion was a novel device for 7 participants, but the other 5 had already tried it (but did not use it regularly).

3.2 Apparatus

We used the following hardware and software:

- Apple Mac Mini (2.5GHz Intel Core i5, with 4GB RAM), running Mac OS X 10.8.3;
- HP L1706 LCD Display, with resolution set to 1280 x 1024;
- Genius Xscroll USB mouse, with the tracking speed set to third tick mark in Mac OS X mouse configuration panel in System Preferences;
- Apple Magic Trackpad, with the tracking speed set to the fourth tick mark, in Mac OS X trackpad configuration panel in System Preferences;
- Leap Motion device (commercial version), with tracking priority set to "Balanced", version 1.2.1+10992;
- The Touchless for Mac software (Leap Motion Inc., n.d.), version 1.0.9.8404;
- A software that implements the multi-directional tapping test and collects data (Cardoso, 2014);

4 RESULTS AND DISCUSSION

Raw data from the experiment and R (R Core Team, 2014) analysis scripts are available at (Cardoso & Seixas, 2014).

4.1 Movement Time, Throughput, and Error Rate

Figure 5 shows the movement time (in seconds) as a function of block.

To estimate the learning effect, we ran pairwise t-tests for average throughput per block (considering all devices) with a significance level of 5%.

The results indicate a clear learning effect in blocks 1 to 3, but also indicate a significant different between blocks 6 and 7, suggesting that there was still some learning effect after block 6. However, in our following analysis we discard only blocks 1 to 3, since those represent the most significant learning effect.

It is obvious that the LM device performs poorly in terms of movement time when compared to the mouse or touchpad. Participants needed more than twice as much time to successfully select a target with the LM than with a mouse.

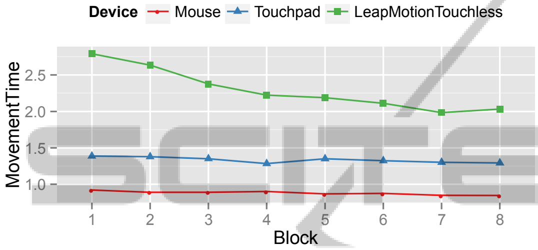

Figure 5: Movement time as a function of block.

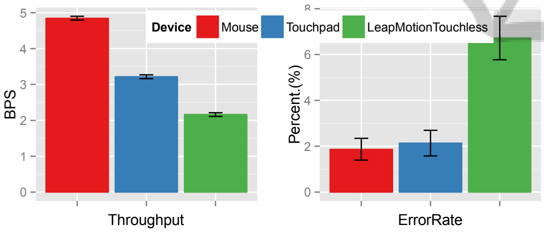

Figure 6: Throughput and error rate.

Throughput for the mouse and touchpad (Figure 6) are consistent with previously reported values (MacKenzie et al., 2001).

Throughput and error rates confirm that, in this experiment, the LM performed poorly. The throughput of the LM is comparable to the throughput of the joystick in (MacKenzie et al., 2001). Error rate for the LM was about 3 times larger than the error rate for the mouse or touchpad, suggesting also that it is more difficult to select a target with the LM than with a mouse, or touchpad.

4.2 Mackenzie's Accuracy Measures

The Mackenzie's accuracy measures (Table 2 and Figure 7) allow us to see the differences between the LM and the mouse/touchpad with greater detail.

Table 2 shows the means, standard deviations, and F statistic for all accuracy measures. Analysis of variance indicates that there are significant differences between devices for all measures except MV and MO.

The most obvious differences are in the TRE, TAC, MDC, and ODC measures, with the LM showing more path events in all measures.

This indicates that the LM movement is more variable than the mouse or touchpad movement in terms of direction changes. However, the overall movement variability, error, and offset are similar for all devices.

In part, these results may be explained by our choice of mechanism for selecting a target with the

Table 2: Means and standard deviations of accuracy measures for each device.

	Mouse		*Touchpad*		*Leap Motion*		
Accuracy measure	*Mean*	*SD*	*Mean*	*SD*	*Mean*	*SD*	*F*
Target re-entry (TRE)	0.10	0.09	0.14	0.12	0.35	0.21	203***
Task axis crossing (TAC)	1.66	0.37	1.26	0.32	2.35	0.62	367***
Movement direction change (MDC)	4.86	0.96	4.48	1.15	8.78	2.43	521***
Orthogonal direction change (ODC)	1.19	0.59	1.04	0.55	4.50	2.18	528***
Movement variability (MV)	20.52	6.88	20.05	6.81	20.47	6.24	0.38
Movement Error (ME)	18.41	5.27	16.55	5.24	16.27	5.33	12.1***
Movement Offset (MO)	-1.68	5.93	-1.58	6.36	-1.49	4.59	0.07

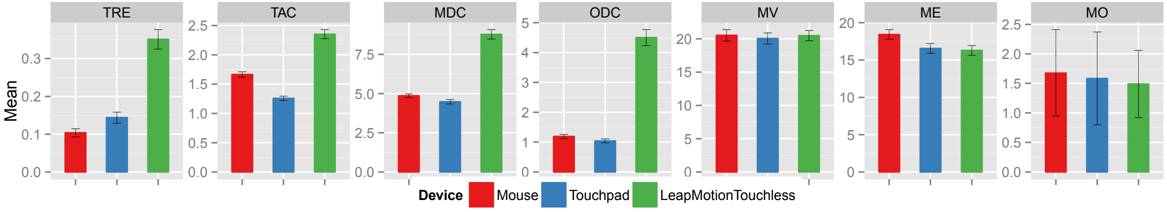

Figure 7: Accuracy measures for the three devices.

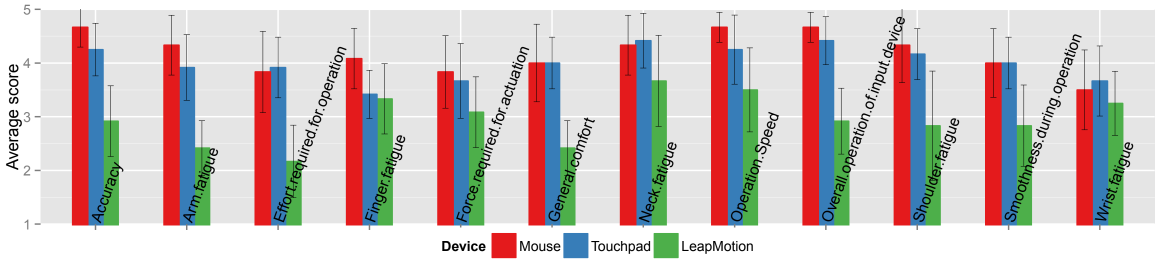

Figure 8: Average scores for the various comfort questions.

LM. With the Touchless application, a mouse click is emulated with a "screen tap" gesture: a quick movement of the pointing finger towards the screen. This movement is not easy to accomplish and may result in a considerable motion of the pointer on the screen. This may explain the large differences in some of the accuracy measures: some of the movement errors may occur during the final screen tap gesture. Further analysis of the data is required to determine in which part of the movement path these differences occur.

4.3 Effort and Comfort

We also collected subjective device preferences and comfort through the ISO 9241-9 assessment of comfort questionnaire. Figure 8 shows the average scores for each question. Again the LM device is rated poorly, having the worst rates in all questions. In seven of the twelve questions, the LM receives a negative average score (below 3), with the worse classifications in the "arm fatigue", "effort required for operation", and "general comfort", with average scores below 2.5.

4.4 Posture Observations

During the experiment, we took notes regarding the postures adopted by the participants while using the LM device.

The common position (see Figure 9) taken by participants was backs against the chair, straight, or slightly curved to the front. The index finger of the dominant hand was responsible for controlling the pointer movement. The other fingers were hidden in order to not to activate other functions of the application (like the scroll function).

Figure 9: Two often observed poses. Left: auxiliary hand resting; right: auxiliary hand supporting the head.

Some participants would leave the thumb showing which sometimes caused the software to perform unintended clicks with the thumb. The auxiliary hand would remain quiescent on the table, over the participant's legs, or supported on the chair's arm. Frequently, the participant would place his/her elbow on the table and support their heads on the auxiliary hand.

4.4.1 Gorilla Arm

One of the problems detected in the posture was that participants had to keep their dominant upper limb suspend in the air. After a while, this caused the participant to start feeling discomfort and the necessity to make a pause.

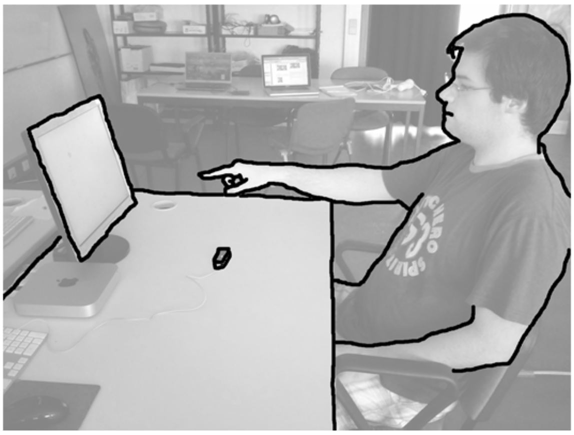

Figure 10: Gorilla arm effect. Dominant arm extended for long periods caused discomfort and a feeling of heavy arm.

In the end, those who perform the experiment with their dominant upper limb suspended in air, no matter how many pauses they took, felt pain in their arm.

This is a similar problem faced by vertical touch screens, where the arm is held in unsupported horizontal position (see Figure 10), leading to fatigue and pain. This is often described as the “gorilla arm” effect because the arm starts to feel very heavy over time. Sometimes this discomfort/pain would extend to the shoulders, neck and back.

4.4.2 Hand Pose

We also observed that participants had to keep their index finger under tension to keep it extended in order to have control over the pointing motion. Over time, stress and fatigue would appear and as the experiment proceeded, this would end up in pain. Even after a short pause, the finger fatigue would not go away and, as a natural response, the finger would start to relax (see Figure 11). This resulted in a loss of precision: when the finger approached the touch zone to perform a click, the pointer would sometimes move outside the target area and trigger an error click.

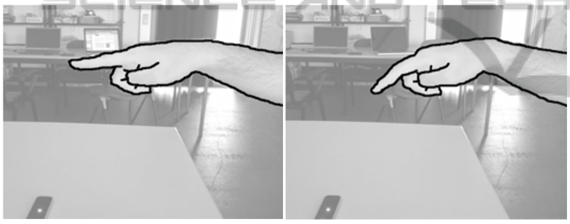

Figure 11: Finger poses. Left: finger completely extended and in tension; right: fatigue causes finger to relax.

Tension was also required when trying to point very precisely at a point and forcing the hand to remain motionless. Long periods of tension would result in the same consequences of physical stress. One other aspect related to the fingers is that most of them (more than three) must remain hidden otherwise a different function of the *Touchless* would be activated (one example is the scroll function). Keeping the fingers hidden is also a demanding effort.

Some participants would prefer to leave their thumb in a relaxed position (hiding it would cause some discomfort and pain after a while). While acceptable, it could be one more cause of an erroneous click because the thumb itself would sometimes register a click. This second click would count as an erroneous selection of the next target.

Also one of the problems of going too far in the touch zone is that the device might lose the tracking of the controlling finger. When this happened, the device would look for another finger. This made the pointer onscreen jump to a different position, sometimes more than once.

One of the participants practiced high performance sports and joked that the LM evaluation seemed like one of the exercises he/she used to do. This participant was the fastest to perform the evaluation, probably due to his/her physical conditions.

4.4.3 Device Position

During the experiment, some participants would attempt to reposition the LM device closer or farther away from them, in an attempt to place their elbows on the table and achieve some comfort - Figure 12.

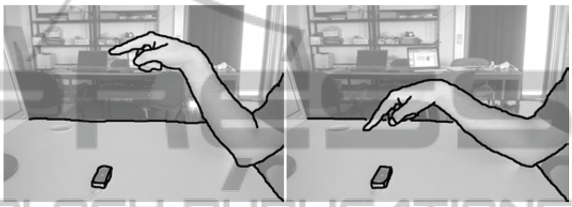

Figure 12: Elbow supported on table. Left: hand at a high position with finger pointing straight at the screen; right: as hand goes down users have tendency to point down.

This led to a reduction in the performance, as participants took longer to perform a successful click, and an increase in the number of times the device lost tracking. The trajectory performed by the hand when the elbow is static is an arc, which caused some participants to end up pointing down at the table. In this situation, the device is sometimes unable to track the finger and the cursor position is lost.

Several participants also commented that if the LM device was in a lower position, the elbows could be more easily supported on the table and the hands wouldn't have to move so high.

4.4.4 Final Remarks

In terms of selecting targets, participants remarked that the lower targets were the hardest to select. One of the causes of this could be that the LM tracking volume is an inverse pyramid, which means that there is a smaller detection range at the bottom.

In some cases, when moving the finger up or down, participants would unintentionally get trigger a click because the movement was not perfectly vertical and crossed the screen tap gesture threshold. This would lead to wrong clicks and frustration. We did not notice this happening on left-to-right or right-to-left movements.

One participant completed all the trials with the LM without reporting feeling any pain. The participant supported her elbows on the table and used her thumb to move the pointer and perform the clicks (the index finger was hidden – see Figure 13). Due to the participants' constitution, she was able to find a comfortable position, able to reach every target without raising his/her elbows. The downside is that the finger tracking failed several times.

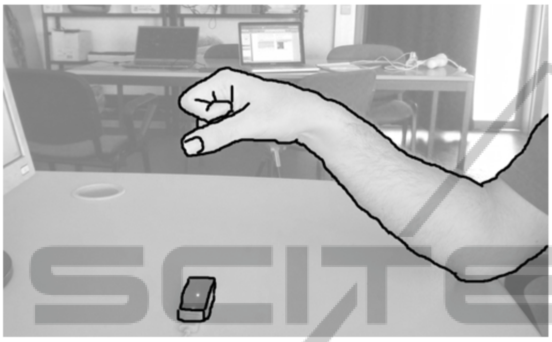

Figure 13: Using the thumb to point.

Finally, some participants with injuries like tendinitis, or with back problems, had greater difficulty performing the experiment and were unable to complete it.

5 CONCLUSIONS

We have presented the results from an experiment designed to characterize the Leap Motion movement on 2D pointing tasks. Results indicate that the Leap Motion performs poorly when compared with a mouse or touchpad, not only in performance but also in user fatigue and comfort.

The results seem to indicate that the LM device is not appropriate for lengthy, or high performance (selection) operations.

However, these results must be interpreted with care. We believe that in a great part, these results reflect the novelty of the LM. One the one hand, the software to make use of the LM may not be fully matured. For example, the Touchless application maps the finger tip position to a screen position, it does not take into account the finger pointing direction, which could potentially result in faster pointing movements. The results in this paper should be taken as a basis to develop and test additional techniques for pointing with the LM, so that more efficient software becomes available.

On the other hand, there are no standard operation posture guidelines for long-term use of the LM device. We noticed that even after some practice trials, users were still trying to determine the best way to position their hands, arms, elbows, etc., to use the device. While posture may not be very important for short-term operation, it impacts efficiency if users have to operate the LM for long periods of time. The results of this study should lead to more study on the ergonomics for the LM device, so that optimal positioning of the device and user's limbs can be determined.

Additionally, the comparison between the LM, the mouse, and the touchpad is not totally fair. Both the mouse and the touchpad use non-linear mapping between device displacement and cursor displacement: faster movements translate to greater cursor displacement. This does not happen with the Touchless application. It may be worth investigating whether similar techniques can be applied to the LM device, even if only in particular situations.

It is also important to note that this experiment only evaluates the LM device for target selection operations; it does not address many of the other types of interaction tasks that the LM allows.

ACKNOWLEDGMENTS

This paper was financially supported by the Foundation for Science and Technology — FCT — in the scope of project PEst-OE/EAT/UI0622/2014. We would like to thank all the participants in this experiment for their time and collaboration.

REFERENCES

Cardoso, J. C. S. (2014). Leap motion study source code. Retrieved from https://github.com/jorgecardoso/leapmotionstudy.

Cardoso, J. C. S., & Seixas, M. (2014). Leap Motion experiment raw data and analysis scripts - UCP. doi:10.6084/m9.figshare.1104376.

Crispy Driven Pixels Inc. (2014). Ethereal. Retrieved June 25, 2014, from https://airspace.leapmotion.com/apps/ethereal/weblink.

International Organization for Standardization. (2000). *Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Requirements for non-keyboard input devices* (Vol. 2000, p. 55). ISO. Retrieved from http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=30030.

Lab, P. (2014). Pointable. Retrieved December 06, 2014, from https://apps.leapmotion.com/apps/pointable/windows.

Leap Motion Inc. (n.d.). Touchless for Mac. Retrieved from https://airspace.leapmotion.com/apps/touchless-for-mac/osx.

Leap Motion Inc. (2014). Airspace store. Retrieved from https://airspace.leapmotion.com/

MacKenzie, I. S., Kauppinen, T., & Silfverberg, M. (2001). Accuracy measures for evaluating computer pointing devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '01* (pp. 9–16). New York, New York, USA: ACM Press. doi:10.1145/365024.365028.

Manolova, A. (2014). System for touchless interaction with medical images in surgery using Leap Motion. In *Proceedings og the 9th INTERNATIONAL CONFERENCE on Communications, Electromagnetics and Medical Applications*. Sofia, Bulgaria.

Nabiyouni, M., Laha, B., & Bowman, D. A. (2014). Poster: Designing Effective Travel Techniques with Bare-hand Interaction. In *EEE Symposium on 3D User Interfaces (3DUI)*. Center for Human - Computer Interaction and Department of Computer Science, Virginia Tech. Retrieved from http://people.cs.vt.edu/blaha/docs/Poster - Designing Effective Travel Techniques with Bare-hand Interaction.pdf.

Nu-Tech. (2014). Mudra Mouse. Retrieved December 06, 2014, from https://apps.leapmotion.com/apps/mudra-mouse/osx.

R Core Team. (2014). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. Retrieved from http://www.r-project.org.

Touchless. (2014). AirInput. Retrieved December 06, 2014, from https://apps.leapmotion.com/apps/airinput/osx.

Uwyn. (2013). GECO MIDI. Retrieved June 25, 2014, from http://uwyn.com/geco/

Vikram, S., Li, L., & Russell, S. (2013). Handwriting and Gestures in the Air, Recognizing on the Fly. *CHI 2013 Extended Abstracts, ACM 978-1-*.

Weichert, F., Bachmann, D., Rudak, B., & Fisseler, D. (2013). Analysis of the accuracy and robustness of the leap motion controller. *Sensors (Basel, Switzerland)*, *13*(5), 6380–93. doi:10.3390/s130506380.