Master M2 MOSIG

Augmented and virtual reality: innovative interaction techniques

Evaluation of Hybrid Interaction for Laptop using Finger Tracking and Touchpad

Author:
Jit Chatterjee
Nairit Bandyopadhyay
Zakary Bouchema

Supervisor:
Céline COUTRIX
Alix GOGUEY

Table of Contents:

1) Description of the context:

The idea arises with the initial thought of utilising the 3D space in front of the laptop screen. With the advancement of new technologies like webcam, Infrared Camera, Depth camera it's pretty easy to detect gestures, pose estimation, etc. Moreover, modern Computer Vision and Deep Learning algorithms make all these tasks real-time. These novel technologies reduce barriers to interaction and increase the input bandwidth between the user and the computer, without requiring the user to wear or acquire a tracked object [1].

2) Explanation of the addressed problem:

The touchpad of laptops has a small area in general. So for pointing targets which are at far distances the user has to lift his finger and slide multiple times (a phenomenon called clutching). This can waste some time so instead of doing clutching for selecting distant points,we can use finger tracking by webcam for that and then we can use the touchpad with the alternate hand for doing the precise minor adjustments. If the pointing task for accurately selecting small objects placed at a large distance has similar performance with our proposed hybrid method (webcam + touchpad) as well as touchpad, then we can make the size of the touchpad smaller in future laptops.

3) State of the art:

Researchers have proposed many methods like Mid Air Hand Gestures [3], AirMouse [4], Fich [5], CameraMouse [11] to use hand gesture movements for mouse pointing and came up with some interesting results where some limitations like Midas Touch [6], Illumination [7] and Fatigue [8] became major drawbacks.

HCI Problem:

- With Touchpad Interaction:
    - Clutching - For making large movements + accurate pointing task on the screen
    - Depends on many factors like size of trackpad, sensitivity of trackpad
    - Clutching is bad [9]
    - Clutching is sometimes necessary [10]

- Hands Free Interaction:
    - Erratic cursor control resulting in lack of throughput and accuracy [2]

- Not finger tracking, instead nose tracking (limited movements)
- Less accuracy (inherent problem of hands free)
- Less Speed (Discussed in next slide)
- Surprisingly users had neutral or positive feedback (corresponds to AirMouse)

To have a different look to the problem and evaluate the performance, we proposed our hybrid method of interaction.

4) Description of our approach to the problem:

We proposed a hybrid method where a person can use his dominant or non-dominant hand by finger tracking to perform the larger displacements in the screen and the other hand to use the touchpad for minor and precise adjustments of the mouse pointer on the laptop screen.

- Pointing task : Small points, placed far apart (not along the boundary of the circle as most experiments have done [4] [2]).
- Evaluating the task with touchpad vs webcam+touchpad (dominant hand) vs webcam+touchpad (non-dominant hand).
- Creating a prototype combining webcam (for making large displacements quickly) and touchpad (for doing minor accurate adjustments).
- Evaluating its performance metrics:
  - Quantitative: Throughput, MT-ID plot (with 1D Fitts' Law for targets placed far apart and of different width)
  - Qualitative: User Satisfaction

Fitts' Law:

For predicting, Fitts' law is an equation giving the time to acquire and select a target based on the distance moved and the size of the target. For measuring, Fitts' law provides a method to quantify human performance in a single measure, "throughput" [12].

- It is a model of human-motor performance.
- Human-motor performance -> Various human activities (specially hand/finger movement)
- Target acquisition task.
- Assumption ->Very fast, towards the target, without miss
- Notion of task difficulty -> Amplitude and width

Evaluation Using Fitts' Law:

$\mathbf{ID = \log_2(A/W + 1)}$ where, ID = index of difficulty (Difficulty of each trail)

A = Amplitude (Diameter of the Layout Circle)

W = Width of the target circles

$TP = ID_e / MT$ where, TP = throughput (bits/second)

ID_e = effective index of difficulty

MT = movement time (speed)

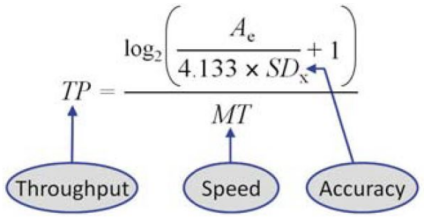
$ID_e = \log_2(A_e / W_e + 1)$

$W_e = 4.133 \times SD_x$

where, SD_x = standard deviation in the selection coordinates computed over a sequence of trials

A_e = The effective amplitude is the actual distance travelled along the task axis

The factor 4.133 adjusts the target width for a nominal error rate of 4% under the assumption that the selection coordinates are normally distributed



5) Implementation:

GitHub Repository: https://github.com/Jit-INP/arvr

Finger Tracking Using:

- Python
- OpenCV (Computer Vision)
- MediaPipe (For Hand Pose Estimation and Keypoint Detection)
- Autopy (For controlling Mouse Pointer and keyboard)

Data Acquisition and Analysis Using:

- GoFitts [2] Software
- MS Excel
- Google Forms

Our prototype works real-time and the Finger Tracking with mouse pointing is pretty smooth. The implemented code runs in a simple CPU.

Video of the Prototype: https://github.com/Jit-INP/arvr/tree/main/videos

6) Experimental Protocol:

- We selected 2 Amplitudes and 2 Widths for the experiment (4 combinations).
- The experimentation was conducted partially onsite (8 participants) and partially remote (4 participants).
- For all the experiments, the screen size was kept the same. However for the remote experiments the lighting condition and the distance from the screen varied somewhat.
- Participants were instructed to select targets as quickly as possible, once by using their dominant hand and then by using their non-dominant hand.
- Before starting the experiment, the participants were instructed to have 5 trial sessions of the Finger Mouse to get accustomed to the new technique.
- After the experiment, we took their feedback for qualitative analysis.

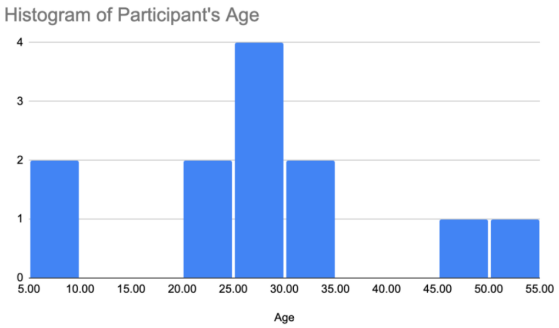
7) Experimental Design:

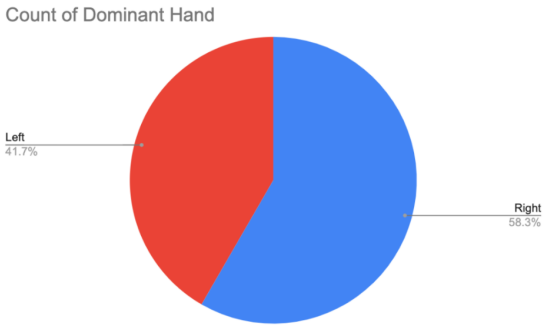
The experiment will be a 3 × 1 × 2 × 2 within-subjects design. The independent variables and levels were as follows:

- Pointing method [Touchpad, Finger Mouse (Dominant) + Touchpad, Finger Mouse (Non-Dominant) + Touchpad]
- Selection method (tap)
- Amplitude (800, 1600 pixels)
- Width (20, 40 pixels)

The result will be 4 test conditions with IDs ranging from $\log_2(1600/20 + 1) = 6.33$ bits to $\log_2(800/40 + 1) = 4.39$ bits.
We had 9 trials for each test combination.
The total number of trials we had (3 × 1 × 2 × 2 × 9 × 12) = 1296

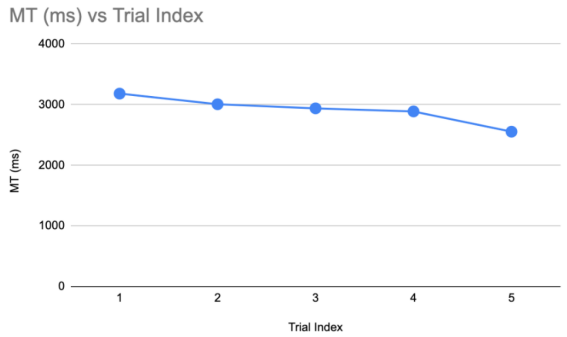
Details of Participants:

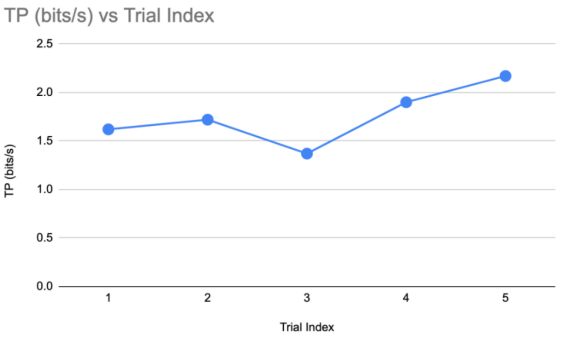




8) Quantitative Results:

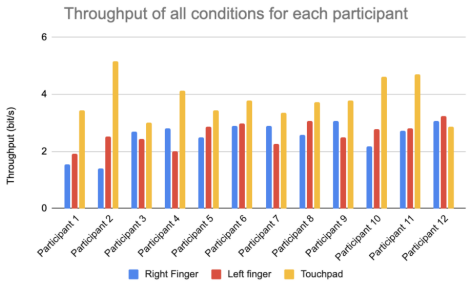
Results of Finger Mouse Trails (Hybrid Method):

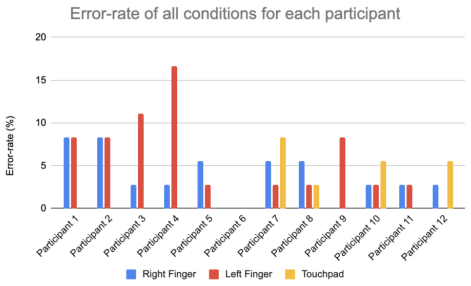


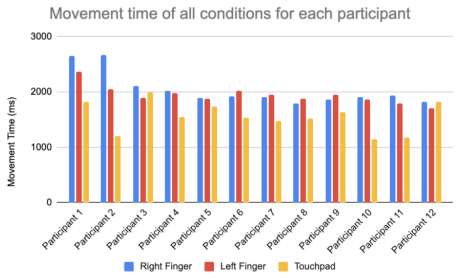


We can see from the trails done by the participants that with time and practice, the throughput of our hybrid method increases.

Overall Results:







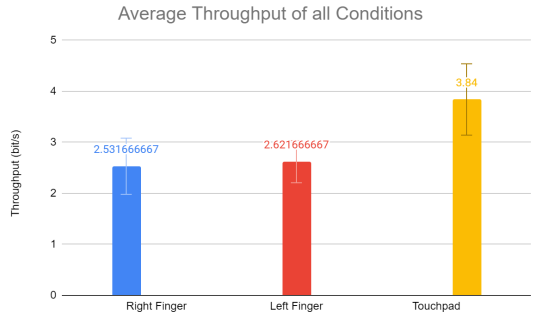
Results and Discussion:

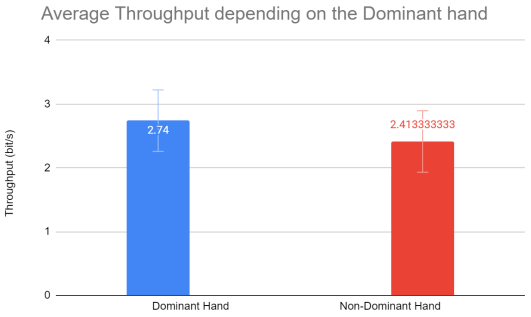
Here we will discuss the results that we've gathered from our experiments.
We analyse the following values, Throughput, Error-Rate and Movement Time.

In the following graphs:
The label 'Right Finger' refers to the use of both the webcam tracking method, with the right hand doing the broad strokes and the use of the touchpad with the left hand for the small adjustments.

The label ‘Left Finger’ refers to the use of both the webcam tracking method, with the left hand doing the broad strokes and the use of the touchpad with the right hand for the small adjustments.
The label ‘Touchpad’ refers simply to the use of the touchpad.

Throughput:



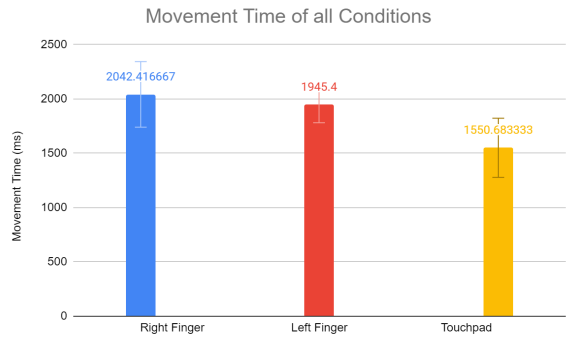


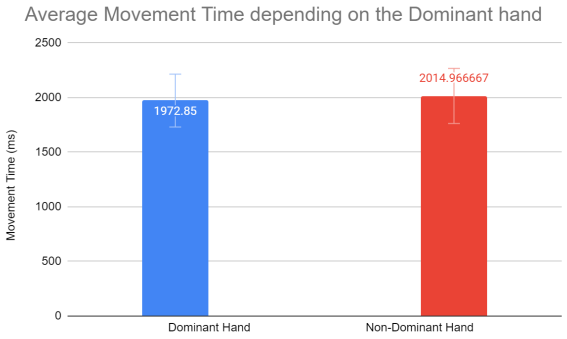
The Right Finger, Left Finger and Touchpad had mean throughputs of 2.53 bit/s, 2.62 bit/s and 3.84bit/s, respectively.
As expected, doing the experiment webcam tracking was more difficult than with the touchpad.
When doing a T-test to compare the Right Finger and Left Finger throughput, the two-tailed P value equals 0.6551 By conventional criteria, this difference is considered to be not statistically significant.

Now let’s have a look at the webcam tracking conditions throughputs but based on the participants' dominant hands.

On average, participants have a throughput of 2.74bit/s when using their dominant hand and a throughput of 2.41bit/s when using their non-dominant hand.
We were expecting the participants to fare better when using their non-dominant hand for the larger strokes and their dominant hand for the small adjustments. However, it ended up being the exact opposites with the using the dominant hand for broad strokes being slightly better.
When doing a T-test to compare the throughput based on the dominant hand, the two-tailed P value equals to 0.0948. By conventional criteria, this difference is considered to be not quite statistically significant.

Movement Time:



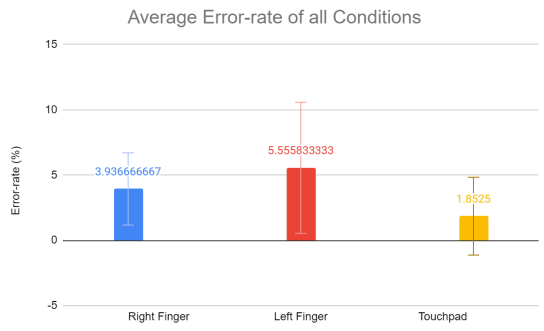


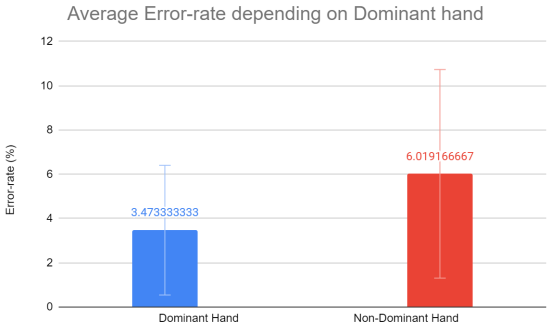
The Right Finger, Left Finger and Touchpad had mean movement time of 2042ms, 1945ms and 1551ms, respectively. As expected, participants were usually faster when using the touchpad, it being the method they're the most accustomed to.
When doing a T-test to compare the movement time of the Right Finger and Left Finger methods, the two-tailed P value equals 0.3386 By conventional criteria, this difference is considered to be not statistically significant.

Now let's have a look at the webcam tracking conditions movement times but based on the participants' dominant hands.

On average, participants have a movement time of 1973ms when using their dominant hand and a throughput of 2015ms when using their non-dominant hand. So participants seem to be faster with their dominant hand.
When doing a T-test to compare the movement time based on the dominant hand, the two-tailed P value equals 0.6804. By conventional criteria, this difference is considered to be not statistically significant.

Error-rate:





The Right Finger, Left Finger and Touchpad had mean error-rates of 3.94%, 5.56% and 1.85%, respectively. As expected, the touchpad is prone to less error than the other two.
When doing a T-test to compare the error-rates of the Right Finger and Left Finger methods,the two-tailed P value equals 0.3388 By conventional criteria, this difference is considered to be not statistically significant.

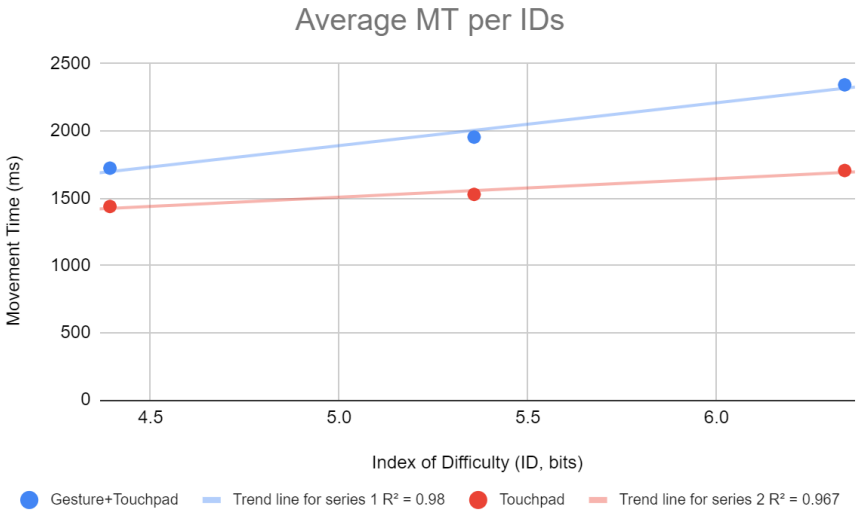
Now let's have a look at the webcam tracking conditions error-rates but based on the participants' dominant hands.

On average, participants have an error-rate of 3.47% when using their dominant hand and a throughput of 6.02% when using their non-dominant hand. So participants seem to be less prone to make errors when using their dominant hands. However the high standard deviation makes it hard to compare.
When doing a T-test to compare the error-rates on the dominant hand,the two-tailed P value equals 0.1263 By conventional criteria, this difference is considered to be not statistically significant.

Fitts's law models:

Fitts' law suggests that MT and ID are directly proportional, hence follows a straight line.
To test this conformance to Fitts' law, we built least-squares prediction equations for each test condition. The general form is : MT = m*ID + c

For the Amplitude and Width, we had 3 different IDs and we plot the corresponding average MT. For both laptop and touchpad+webcam we see that ID and MT follows a linear relation. Here's the scatter plot with the regression line.



Condition	Intercept (c)	Slope (m)
Touchpad	821.3 ms	137.4 ms/bit
Webcam tracking	300.1 ms	318.3 ms/bit

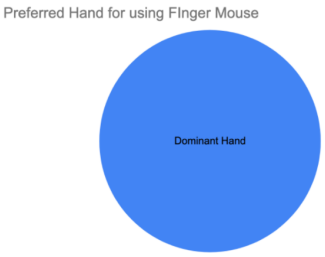
At zero difficulty, webcam+touchpad works better but due to high slope, with slight increase in ID, the movement time increases rapidly. The most notable observation is the intercept of our webcam tracking method being lower than that of the touchpad since intercepts c.

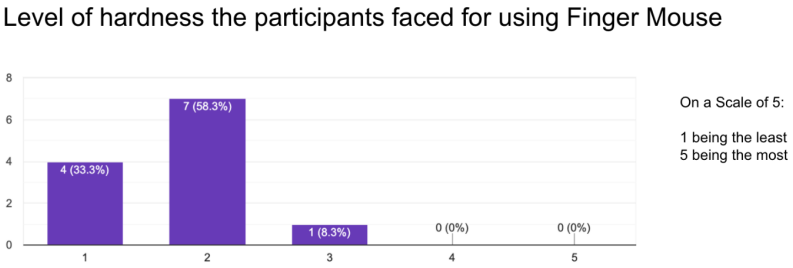
Results conclusion:

As we expected, the touchpad method still outperforms all of our methods based on webcam finger tracking.
However, our webcam tracking still provides interesting results.
When studying our webcam tracking method for conformation to Fitts' law, the low intercepts of our method implies that with less difficult tasks our method could outperform the touchpad and prove handy. Making the use of gestures for targeting acquisition techniques is still very much worth studying.

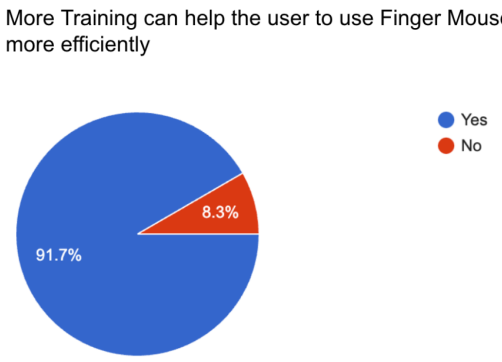
9) Qualitative Analysis:

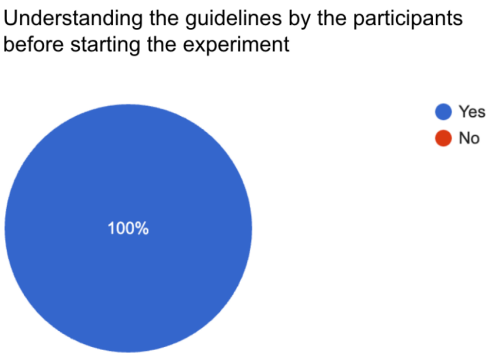
Based on the feedback of the 12 participants via Google Forms we have done the Qualitative Analysis.





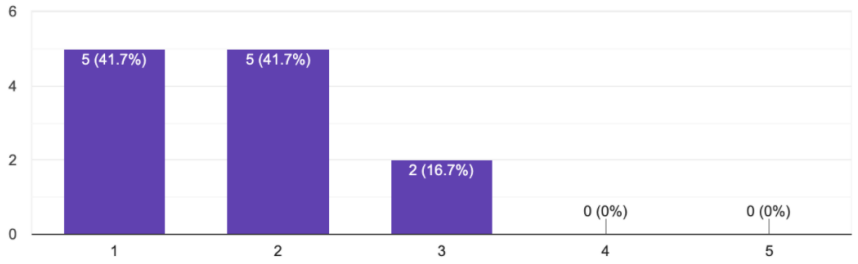
Based on the above figures we can conclude that The Finger Mouse (Hybrid Method) is not that hard for the participants to use.





Moreover the participants tried to convey via their feedback that the throughput of the hybrid method (Finger Mouse) can be increased efficiently with the help of more training.

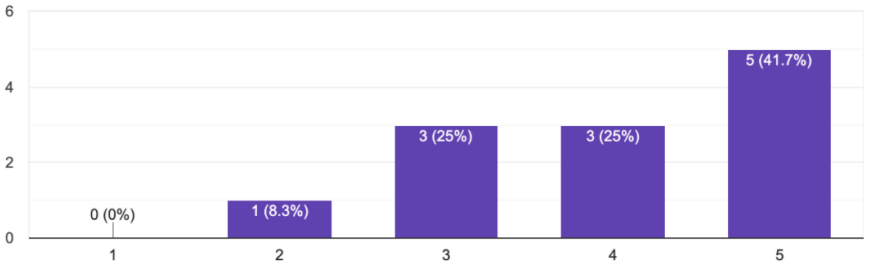
Level of Fatigue using the Dominant Hand for Finger Mouse



On a Scale of 5:

1 being the least
5 being the most

Level of Fatigue using the Non-Dominant Hand for Finger Mouse



Finally based on the Qualitative Analysis we can conclude that the Level of Fatigueness is more using the Non-Dominant Hand as Finger-Mouse.

10) Conclusion:

- Touchpad alone is better than our hybrid interaction method (Quantitatively) because
  - Slope $_{touchpad}$ = 137.4 ms/bit < Slope $_{webcam+touch}$ = 318.3 ms/bit which means for slight increase in ID, the MT for webcam+touch increases by 2.3 times.
  - Generally, the Throughput $_{touchpad}$ > Throughput $_{webcam+touch}$ by approx 1.5 times.

- But on the Qualitative Analysis based on Participants, majority gave feedback saying,
  - Finger-Mouse is fun, an exciting prospect and probably an interesting future interaction technique.
  - Level of Fatigueness is more using the Non-Dominant Hand as Finger-Mouse.

- Some other interesting facts like
    - More Throughput, Less Movement Time and Less Error using the Dominant Hand as Finger-Mouse.
    - Participants having younger age have similar or more Throughput using the Finger-Mouse + Touchpad compared to Touchpad.
    - Based on the trials we have understood more practice by the participants increases the Throughput of the Hybrid method (Finger Mouse + Touchpad).

Future Works:

- Running our implemented code in GPU and optimising the code to make it more smooth.
- To use different lighting conditions (Different Illuminance levels measured in Lux).
- To use different distances between the participant and the webcam.

References:

[1] Roland Aigner, Daniel Wigdor, Hrvoje Benko, Michael Haller, David Lindbauer, Alexandra Ion, Shengdong Zhao, Jeffrey Tzu Kwan Valino Koh | MSR-TR-2012-111 | November 2012 | Understanding Mid-Air Hand Gestures: A Study of Human Preferences in Usage of Gesture Types for HCI

[2] Hassan, M., Magee, J., & MacKenzie, I. S. (2019). A Fitts' law evaluation of hands-free and hands-on input on a laptop computer. Proceedings of the 21st International Conference on Human-Computer Interaction – HCII 2019 (LNCS 11572), pp. 234-249. Berlin: Springer. doi:10.1007/978-3-030-23560-4

[3] Roland Aigner, Daniel Wigdor, Hrvoje Benko, Michael Haller, David Lindbauer, Alexandra Ion, Shengdong Zhao, Jeffrey Tzu Kwan Valino Koh | MSR-TR-2012-111 | November 2012 | Understanding Mid-Air Hand Gestures: A Study of Human Preferences in Usage of Gesture Types for HCI

[4] Michael Ortega, Laurence Nigay. AirMouse: Finger Gesture for 2D and 3D Interaction. Conference Proceedings of INTERACT '09, the Twelfth IFIP TC13 International Conference on Human-Computer Interaction, Springer LNCS (Lecture Notes in Computer Science), 2009, Uppsala, Sweden. pp.214-227, ⟨10.1007/978-3-642-03658-3_28⟩. ⟨hal-00953293⟩

[5] Müller, Jonas & Rieger, Lea & Aslan, Ilhan & Anneser, Christoph & Sandstede, Malte & Schwarzmeier, Felix & Petrak, Björn & Andre, Elisabeth. (2019). Mouse, touch, or fich:

comparing traditional input modalities to a novel pre-touch technique. MUM '19: Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia. 1-7. 10.1145/3365610.3365622.

[6] Wu, Huiyue & Wang, Jianmin. (2015). A visual attention-based method to address the midas touch problem existing in gesture-based interaction. The Visual Computer. 32. 10.1007/s00371-014-1060-0.

[7] Horatiu-Stefan Grif, Cornel Cristian Farcas, Mouse Cursor Control System Based on Hand Gesture, Procedia Technology, Volume 22, 2016, Pages 657-661, ISSN 2212-0173, https://doi.org/10.1016/j.protcy.2016.01.137.

[8] Hincapié-Ramos, J.D., Guo, X., Moghadasian, P. and Irani. P. 2014. Consumed Endurance: A Metric to Quantify Arm Fatigue of Mid-Air Interactions. In Proc. CHI '14, ACM.

[9] Mathieu Nancel, Daniel Vogel, Edward Lank. Clutching Is Not (Necessarily) the Enemy. ACM CHI, 2015, Apr 2015, Seoul, South Korea. ⟨10.1145/2702123.2702134⟩. ⟨hal-01523428⟩

[10] Michel Beaudouin-Lafon, Stéphane Huot, Halla Olafsdottir, Pierre Dragicevic. GlideCursor: Pointing with an Inertial Cursor. International Working Conference on Advanced Visual Interfaces (AVI'14), May 2014, Como, Italy. ⟨hal-00989252⟩

[11] Betke, M., Gips, J., Fleming, P.: The camera mouse: Visual tracking of body features to provide computer access for people with severe disabilities. IEEE Trans. Neural Syst. Rehabil. Eng. 10(1), 1-10 (2002). https://doi.org/10.1109/TNSRE. 2002.1021581

[12] MacKenzie, I. S. (2018). Fitts' law. In K. L. Norman & J. Kirakowski (Eds.), Handbook of human-computer interaction, pp. 349-370. Hoboken, NJ: Wiley. doi:10.1002/9781118976005