

The Tradeoff between Spatial Jitter and Latency in Pointing Tasks

Andriy Pavlovych

Wolfgang Stuerzlinger

York University
4700 Keele Street
Toronto, ON M3J 1P3, Canada
{andriyp|wolfgang}@cse.yorku.ca

ABSTRACT

Interactive computing systems frequently use pointing as an input modality, while also supporting other forms of input such as alphanumeric, voice, gesture, and force.

We focus on pointing and investigate the effects of input device latency and spatial jitter on 2D pointing speed and accuracy. First, we characterize the latency and jitter of several common input devices. Then we present an experiment, based on ISO 9241-9, where we systematically explore combinations of latency and jitter on a desktop mouse to measure how these factors affect human performance. The results indicate that, while latency has a stronger effect on human performance compared to low amounts of spatial jitter, jitter dramatically increases the error rate, roughly inversely proportional to the target size.

The findings can be used in the design of pointing devices for interactive systems, by providing a guideline for choosing parameters of spatial filtering to compensate for jitter, since stronger filtering typically also increases lag. We also describe target sizes at which error rates start to increase notably, as this is relevant for user interfaces where hand tremor or similar factors play a major role.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces – *benchmarking; ergonomics; evaluation/methodology.*

General Terms

Measurement, Performance, Human Factors.

Keywords

Latency, jitter, Fitts' law, pointing.

1. INTRODUCTION

While a mouse is the most common pointing device in human-computer interaction, there is a large variety of other devices that have appeared over the years. One of the most recent additions is

the Nintendo *WiiMote*. Although virtually all of these devices can be used in interactive computing systems, most of them exhibit significantly more latency and/or jitter than the mouse.

Latency, or lag, is the delay in device position updates [7]. Lag has been previously demonstrated to significantly impact human performance in both 2D and 3D tasks [14, 17, 20]. Spatial jitter, due to either noise in the device signal or hand tremor, may also affect performance. These two factors together often affect the choice of an input device. For high-precision tasks, system designers may have to choose between devices with low jitter, or with low latency. However, since there is no published evidence on the relative performance impact of latency vs. jitter, designers have little guidance for this tradeoff.

We present a study that systematically investigates the effects of latency and jitter on human performance. The study employs Fitts' law, a well-established model of pointing device performance. In our experiments, we used a mouse as an exemplary low-latency, low-jitter device, and artificially added latency and jitter to match the range of latency and jitter present in other commonly used devices. The goal of the experiment was to determine, all else being equal, the effects of latency and jitter on device performance. In other words, which has a stronger impact on human performance: latency or jitter? As one can often trade some latency for a decrease in jitter, typically through time-domain filtering, knowing the interrelationship between the two allows a designer to make an informed decision in choosing a filter and about its parameters.

2. BACKGROUND

This section briefly discusses relevant work in object manipulation, tracking and pointing technology, and Fitts' law.

A number of pointing technologies exist today that can be utilized in interactive systems. In addition to a common computer mouse, the major alternatives are:

- Touch pads (used in portable computers)
- Laser pointers (used for distant pointing, e.g. [16])
- Video-based pointer tracking (distant pointing, e.g. [3])
- Video-based hand gesture tracking e.g. [8]
- Accelerometer-enhanced devices (gyro-mouse, gaming devices, tilt-based interaction, e.g. [13])
- Touch screen-based technologies:
- Optics-based (below the screen, e.g. [10])
- Computer vision-based (above screen, e.g. [2])

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EICS'09, July 15–17, 2009, Pittsburgh, Pennsylvania, USA.

Copyright 2009 ACM 978-1-60558-600-7/09/07...\$5.00.

Our current study uses the mouse as an input device. The main reason for that is that it is a well-studied and established device. Another rationale is that previous studies comparing mouse-based and tracker-based manipulation techniques found that the differences between mice and trackers, when used in the same controlled conditions, can be explained to a large degree by the disparities in latency and jitter of the employed devices [19]. Thus, we believe that many classes of devices can be emulated with a mouse, if one artificially adds latency and jitter to an “ideal” mouse. Ultimately, no performance difference between the devices should be detectable when the latency and jitter of the devices match.

2.1 Pointing Systems, Lag and Jitter

Latency is the time from when the device is physically moved, to the time the corresponding update appears on the screen. It is well known that latency adversely affects human performance in both 2D pointing tasks [14] as well as in 3D pointing [4, 20].

Spatial jitter is caused by a combination of hand tremor and noise in the device signal. One way to observe this is to immobilize a tracking device while observing the reported positions; even when the device remains stationary, its reported positions may fluctuate. However, some devices also exhibit additional noise during movements. Hand jitter only exacerbates this problem in free-space tracking devices.

Temporal jitter, or latency jitter, refers to changes in lag with respect to time. Ellis et al. [5] report that people can detect very small fluctuations in lag, likely as low as 16 ms. Hence, when examining system lag, one must also ensure that latency jitter is minimized, or at least known.

For 3D tracking devices, Foxlin [7] provides a thorough overview of the available types of tracking technologies. Although it is argued that one should choose a specific tracking technology based on needs [7], most tracking technologies have shortcomings that affect performance. Specifically, they tend to suffer from high latency and/or jitter.

2.2 Fitts’ Law

Fitts’ law [6] is a model for serial fast, aimed movements. It is expressed as:

$$MT = a + b \cdot \log_2(A/W + 1) \quad (1)$$

where MT is movement time, A is the amplitude of the movement (i.e., the distance between two targets), and W is the width of a target. The log term in the equation is called the Index of Difficulty (ID), which is commonly assigned a unit of bits:

$$MT = a + b \cdot ID \quad (2)$$

The coefficients a and b are usually determined empirically for a given device and interaction style, such as a stylus on a tablet, a finger on an interactive tabletop, etc.

The interpretation of the equation is that movement tasks are more “difficult” when the targets are smaller or farther away. Fitts’ law has been used to characterize the performance of pointing devices and is one of the components of the standard evaluation in accordance with ISO 9241-9 [11]. Indeed, if the movement time

and ID are known, then the ratio gives the throughput BW of the input device in bits per second (bps)¹:

$$BW = ID / MT \quad (3)$$

2.2.1 Effective Width and Effective Distance

During a traditional Fitts’ task, participants are asked to click on targets of various sizes, spaced at various distances. Usually they hit larger targets with fewer misses and relatively closer to their centers. Smaller targets are missed more often and clicks may occur farther away from their centers. Thus, it is beneficial to take this variation of accuracy into account. As an illustration, Figure 1 shows the distribution of hits when a task is performed repeatedly.

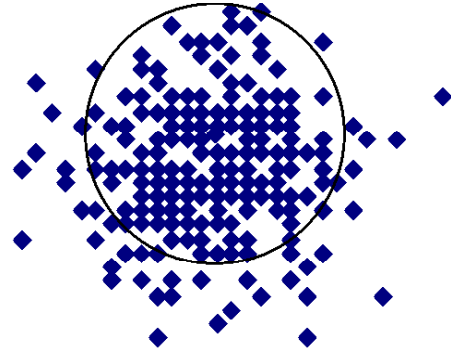


Figure 1. Distribution of clicks on a circular target.

MacKenzie argues for using a sub-range of the hit data, corresponding to about 96%, as the *effective width* of the target [12]. This range corresponds to approximately 4.133 standard deviations of the observed coordinates of hits, relative to the intended target center:

$$W_e = 4.133 \cdot \sigma \quad (4)$$

MacKenzie points out that this corresponds better to the task that the user *actually* performed, compared to the ideal Fitts task.

To calculate the effective parameters in our study, a projection of the actual movement vector onto the intended vector is computed and the difference of the vector lengths is used as the deviation from the intended center – allowing one to later compute the *effective width* using the equation (4) above. A similar approach is used for the distance: the actual movement distances are measured, and then averaged over all repetitions, thus forming the *effective distance*. Figure 2 illustrates both notions. Finally, both effective distance and effective width, in combination with movement time, are used to determine the *effective throughput* of a device, a measure that now takes not only the performance but also the accuracy of target acquisitions into account.

¹ In the absence of noticeable jitter and latency and assuming accurate pointing, throughput does not depend on the index of difficulty. This makes it a convenient metric for comparing pointing devices.

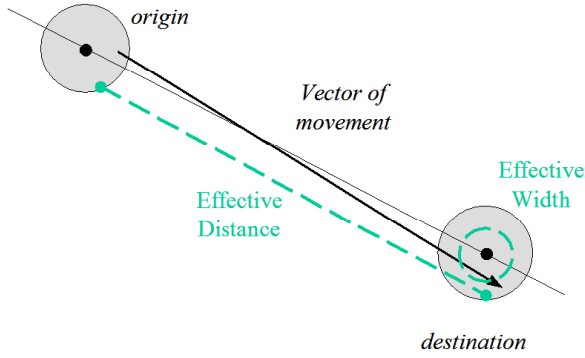


Figure 2. Illustration of Effective Width and Effective Distance. Note that these are averaged over multiple movement vectors.

We use these effective measures in place of the true target widths and amplitudes to seamlessly incorporate the potential effect of differing participant strategies, which favour either speed or accuracy [12]. In essence, the approach treats more accurate clicks (i.e., clicks closer to the centre of the targets) as clicks on smaller targets, while the clicks outside of the intended targets are treated as “successful” clicks of the virtual targets that are larger in size. This makes the measure independent of participant strategy, which is one reason why it is recommended by ISO 9241-9 for pointing devices [11].

3. CHARACTERIZING SYSTEM LATENCY AND JITTER

Before commencing any experiments, we quantified the end-to-end system latency of our setup to establish a baseline condition. We used a mouse on a plain table surface, as optical mice are generally very accurate and smooth in sensing a motion. The friction induced by the surface also effectively dampens any exterior oscillations and we were not able to observe noticeable spatial jitter when the mouse was stationary or moving in straight line. We just note that, if one were to use a different kind of device, which is affected either by external factors (e.g., hand tremor in a laser pointer), or the inaccuracy of the apparatus (e.g., video-based motion tracking), one would have to measure and characterize baseline jitter before further investigation.

3.1 Latency Jitter

For the mouse we also measured latency jitter, i.e., the amount of *change* in latency from one point in time to another. To measure this, we looked at the mouse update intervals. Our mouse reported updates at 125 Hz, a value typical for most USB mice. A histogram of these times showed that more than 99.5% of the updates happened within 8–11 ms of the previous sample. Almost all of the remaining samples followed within 5–8 ms. Consequently, we do not believe latency jitter to be an issue in our experiments.

3.2 Characterizing Latency

A variation of Mine’s method was used to characterize the lag of both the mouse and the tracker [15].

3.2.1 Equipment Setup

A Microsoft optical mouse was moved along the top bezel of CRT display. The area where the mouse moved was covered with a

textured material to ensure reliable tracking. When the mouse was moved, the cursor on the screen moved correspondingly. For consistency, we used the same software that we subsequently employ for latency/jitter measurements, with both factors set to zero. To complete the setup, a video camera simultaneously filmed the motion of both the mouse and the cursor at a frame rate of 60 Hz (in progressive scan mode). We used a 21” CRT² display at a resolution of 1280 × 1024 pixels with a 100 Hz screen refresh rate.

3.2.2 Procedure

The mouse was repeatedly moved by hand sideways along the top edge of the display bezel at a rate of about 1 Hz. Movement of both the mouse and the computer generated cursor was recorded with a digital video camera. The end-to-end tracking latency of the two devices equaled the differences in the frame times of their corresponding phases of motion.

3.2.3 Analysis and Results

Approximately two minutes of video were recorded with the digital camera. This video was analyzed manually after the experiment to derive the end-to-end latency for both devices.

Peaks of mouse and cursor movement were examined. When the mouse reached the peak position of its movement in one direction or the other, the frame number and its time were noted. When it began to move back the other way, the mouse cursor on the screen would move back as well, but after a short delay due to tracking latency. These delays were also recorded.

Because the camera was only recording at 60 Hz, we averaged a total of 10 measurements to remove any potential sampling artefacts. Ultimately, the average delay of the mouse cursor motion relative to motion of the mouse was 33.2 ± 2.8 ms.

As a comparison, here is the summary of the lag measured in other devices:

- The same mouse, on the same machine, but via the PS/2 port: 53.1 ± 3.32 ms.
- The same mouse, with 25 ms software delay added: 59.8 ± 3.7 ms (similar correspondence for higher latencies).
- Same mouse on an LCD display (1280 × 1024 @ 60 Hz): 43.2 ± 2.7 ms.
- Laser pointer on a DLP projection screen (with a 120 Hz camera tracking the laser spot): 102.9 ± 2.2 ms.
- PS/2 wireless mouse on the same screen: 102.9 ± 3.3 ms.
- Wii remote on a DLP projected screen: 106.3 ± 6.2 ms.

This range of values was used as a guideline for choosing the set of latencies to investigate in our experiment.

Some of the numbers in the list are large due to considerable latency in the display itself. E.g., many projectors and LCD monitors buffer frames before displaying them, and this can contribute 50 ms or more to the measured end-to-end latency.

² Cathode Ray Tube (CRT) monitors are well-characterized devices; their latency is predictable, and is among the lowest possible for a desktop output device.

Despite that, the choice of displays is not always independent, as some pointing devices imply the use of a specific display technology. For example, a stylus-based Tablet-PC assumes an LCD display underneath, and a laser pointer-based system often implies a large front or back projected screen.

3.3 Characterizing Jitter

Although the optical sensing method employed by the mouse may generate some jitter, this appears to be filtered in the mouse hardware. While the technical details in each specific implementation may differ, typical optical mice sensors are in essence low-resolution miniature video cameras taking images at a rate of several thousand per second [1]. Since a desktop pointing device only requires about a hundred updates per second, the 10:1 or greater excess of frames is apparently used to smooth the device reports via averaging or some other filtering technique.

Likewise, hand jitter, or hand tremor, does not appear to be an issue in our experiments, as resting the mouse on a physical surface largely eliminates it. This is because tremor, like any other mechanical oscillation, depends on friction, as well as mass, rigidity, and external disturbances. Friction *dampens*, or reduces the magnitude of the oscillations. We assume the mouse we used to have no significant jitter of either kind.

Nevertheless, with other devices or under other conditions, significant jitter may be present, and may be large enough to have a noticeable effect on interaction. To have some guideline as to what range of jitters to use as independent variables in our experiment, we measured the jitter present in several situations:

- Laser pointer, held with extended arm: 0.20–0.25 degrees³ mean-to-peak;
- Same, held with both hands: 0.10–0.15 degrees mean-to-peak;
- Optitrack optical tracking device, 1 m away from the cameras: 0.4 mm mean-to-peak [19].

The given values correspond to the maximum mean-to-peak range of about 6–8 pixels – assuming a user standing 2 m away from a 1.5 m wide screen, with horizontal resolution of 1024 pixels. Depending on the distance to the screen and motion amplification ratio, as well as other factors, such as the emotional state of the person using the device, ambient temperature, human fatigue and so on, this jitter may be even larger.

The device jitter for the *Optitrack* system mostly resembled white noise. It was approximately 0.4 mm mean-to-peak in all axes, with a RMS value of 0.30 mm; and with occasional spikes reaching ± 0.6 mm. Note that there is some increase of noise at the lowest frequencies, thus it is not *strictly* white noise. This is visualized in Figure 3.

In contrast, hand tremor is characterized by an emphasis of lower frequencies with a substantial decay in amplitude above 10–15 Hz [9].

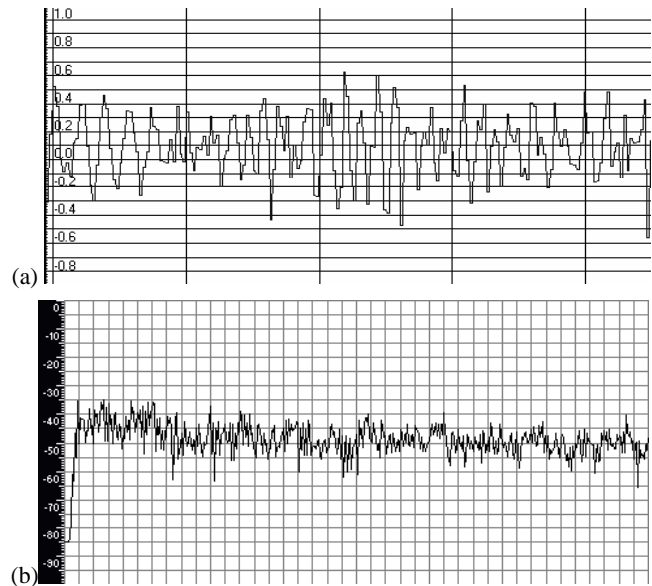


Figure 3. Spatial jitter of the *Optitrack* tracker. (a): jitter during smooth motion in mm, 4.5 s fragment; (b): FFT of the recorded data, logarithmic response, frequencies (linearly) from 0 to 60 Hz, low frequency contribution of regular motion filtered out with a digital filter.

4. EXPERIMENT

This experiment used the procedure specified in the ISO 9241-9 standard to compare the throughput under various magnitudes of lag and spatial jitter. Based on Fitts' law, this standard measures performance of devices in 2D pointing tasks.

4.1 Participants

Twelve students from the local university participated in the experiment, with ages ranging from 19 to 31 (mean age 23 years). Eight were male. All were right handed, or otherwise used the mouse with their right hand. The study lasted 30–40 minutes.

4.2 Apparatus

The computer was an Intel Pentium 4-based desktop, running at 2.4 GHz, with 1 GB of RAM. A Microsoft *Wheel Optical* mouse was used. The software, written in C#, implemented a standard Fitts' 2D task, as described in ISO 9241-9 [11], see Figure 4. The application presented 13 targets in a circle. Upon clicking the first highlighted target (the top one) the timer would start and the opposite (bottom-left) target would be highlighted, directing the participant to select it. The next target was on the opposite side, to the immediate right of the initial target, and so on until all targets were selected. The software automatically logged target sizes, distances between targets, the times to click between targets, errors, and screen coordinates of click events. It also performed the effective width calculation as described above.

4.3 Procedure

After signing informed consent forms, participants were seated in front of the computer display at a distance of about 0.6 m (2 feet).

Participants were given a brief introduction to the system, and were allowed to try the system and find the most comfortable seating position. After that, they were directed to proceed with the

³ Hand tremor depends on many factors. Our numbers were measured for a 25–30 year old person, working in a typical office, and should only be viewed as a guideline estimate. Certain medical conditions, as well as age, can substantially worsen these values.

task, in which they were instructed to click on the highlighted targets as quickly and accurately as possible.

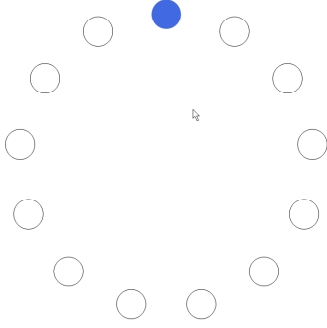


Figure 4. Task for the experiment. Participants would click the highlighted target, which then highlighted the next target, etc. The width of the targets and the distance between targets varied for each “round” of targets.

4.4 Design

This experiment had four independent variables in a $5 \times 5 \times 3 \times 2$ arrangement (150 combinations):

- Latency: 33, 58, 83, 108, and 133 ms;
- Spatial jitter: 0, ± 4 , ± 8 , ± 12 , and ± 16 pixels, implemented as uniformly distributed noise with a maximum offset of 0, 4, 8, 12, and 16 pixels;
- Target widths (diameter): 14, 35, and 91 pixels;
- Target amplitudes⁴: 416 and 728 pixels.

The dependent variables were:

- Device throughput (in bits per second), calculated as described earlier;
- Miss ratio, the percentage of targets not acquired in every round;

The range of latencies and jitters covers the extent of these factors observed in various devices, including the latency of a gaming console remote control used on a projection screen, as well as laser pointer jitter at a distance of several meters. They approximately correspond to the values measured in the previous section. The combination of the target widths and amplitudes forms a uniformly spaced range of indices of difficulty from 2.5 to 5.7 bits, which covers the span encountered in typical desktop tasks, as well as in other forms of interactions. A target 14 pixels large is about the size of a “window close” button in common window managers. Acquiring targets of smaller sizes was observed to be very difficult under the high jitter conditions during a small pilot experiment, and hence we decided to restrict the design to values that avoided excessive participant frustration. As the task is highly repetitive, we choose the total number of combinations to keep the total participation time to well less than one hour to keep the fatigue effect as small as possible.

The experiment was *within subjects*, and the order in which the 150 combinations of the factors were presented was randomized

(without replacement), to compensate for asymmetric transfer of learning effects.

Each participant completed a set of 150 rounds with different latencies, jitters, widths, and amplitudes. As it is not possible to meaningfully measure the click time of the first target, which started each trial, there were only 12 clicks recorded per round. Given that there were 12 participants and 12 recorded target clicks per round, this gave a total of $150 \times 12 \times 12 = 21,600$ trials.

5. RESULTS

5.1 Throughput

Results were analyzed using ANOVA. There were significant main effects for latency, jitter, and target width on throughput. The throughput was computed according to equation 3, using the effective width and amplitude parameters, derived from each set of 12 measurements for every experimental condition.

5.1.1 Latency

The effect of latency on the throughput was significant ($F_{4,44} = 96.77$, $p < .0001$). The interaction between the latency and the width on the throughput was also significant ($F_{8,88} = 4.97$, $p < .0001$). Figure 5 shows a graph of the results.

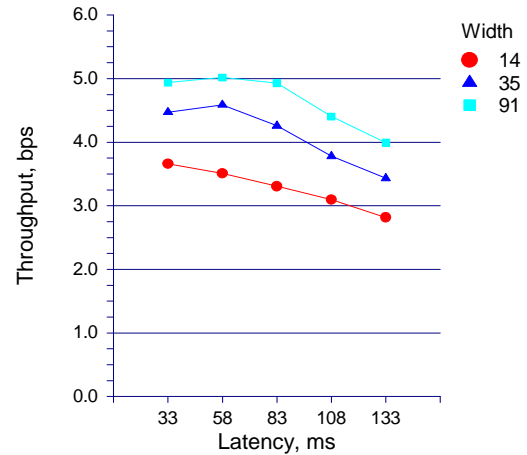


Figure 5. Throughput for varying levels of width and latency.

5.1.2 Jitter

The effect of jitter on the throughput was significant ($F_{4,44} = 82.83$, $p < .0001$). The interaction between the jitter and the width was also significant ($F_{8,88} = 8.20$, $p < .0001$). Figure 6 illustrates the results.

⁴ Defined as the diameter of a large circle, along which the targets are placed. See Figure 4.

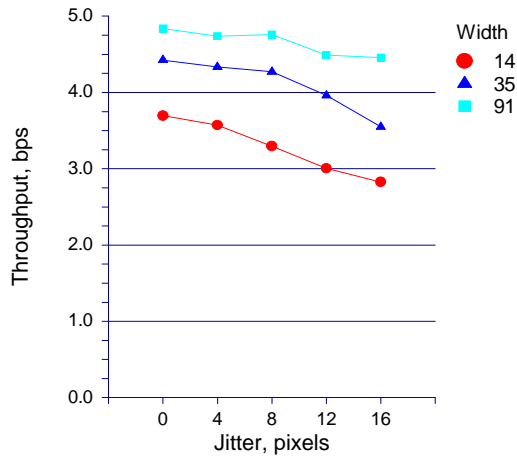


Figure 6. Throughput for varying levels of width and jitter.

5.1.3 Width

The effect of target width on the throughput was significant ($F_{2,22} = 147.58, p < .0001$). See Figure 7 for details.

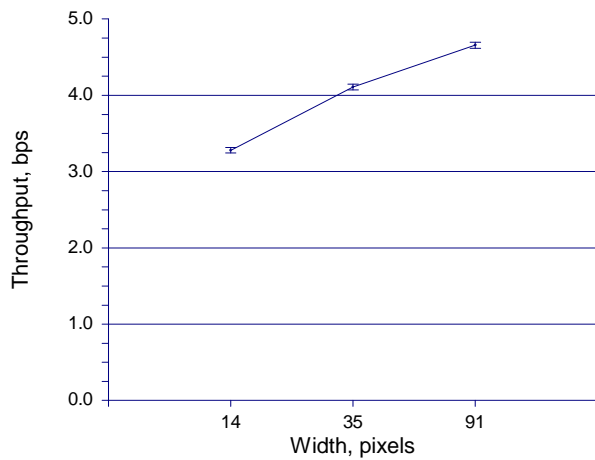


Figure 7. Throughput for varying levels of width. Error bars represent standard error.

5.1.4 Amplitude

The effect of target amplitude on throughput was significant ($F_{1,11} = 42.08, p < .001$), albeit the difference was only about 4 %; 3.93 bps for 416 pixels vs. 4.09 for 728 pixels.

5.1.5 Index of Difficulty

The effect of index of difficulty on the throughput was significant ($F_{5,55} = 119.11, p < .0001$). See Figure 8. IDs were computed from the widths and amplitudes of the targets as they were displayed on the screen.

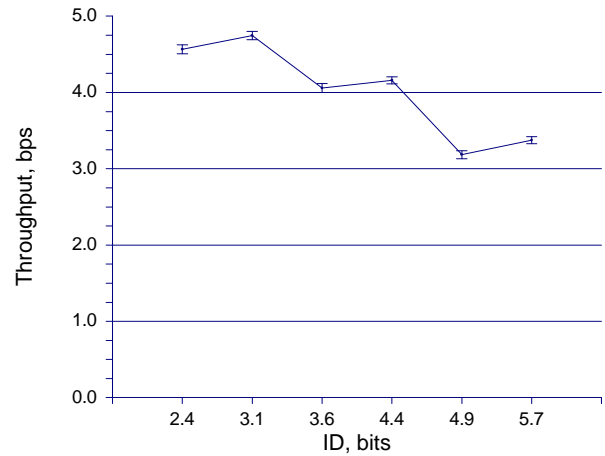


Figure 8. Throughput for varying levels of ID.

5.2 Error Rate

5.2.1 Latency

The effect of latency on the error rate was significant ($F_{4,44} = 8.51, p < .0001$). The interaction between the latency and the width was also significant ($F_{8,88} = 2.13, p < .05$). Figure 9 shows the results.

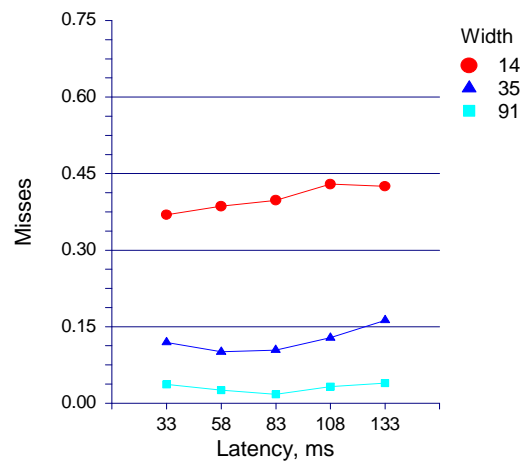


Figure 9. Error rate for varying levels of width and latency.

5.2.2 Jitter

The effect of jitter on the error rate was significant ($F_{4,44} = 239.38, p < .0001$). The interaction between the jitter and the width was also significant ($F_{8,88} = 99.95, p < .0001$). Please refer to Figure 10.

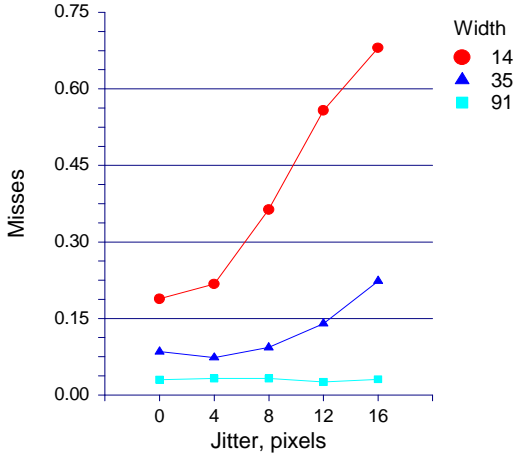


Figure 10. Error rate for varying levels of width and jitter.

5.2.3 Width

The effect of target width on the error rate was significant ($F_{2,22} = 553.40, p < .0001$). See Figure 11 for details.

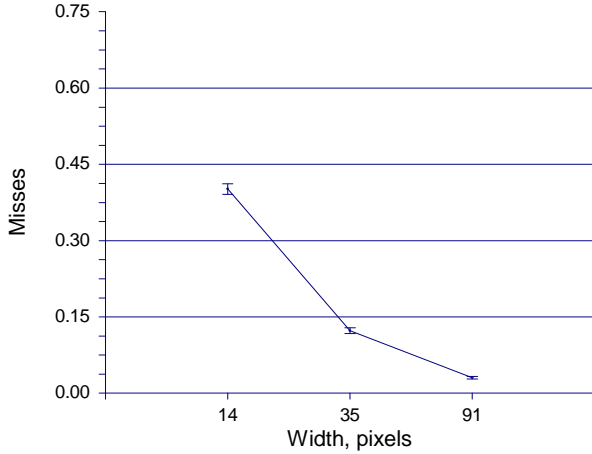


Figure 11. Error rate for varying levels of width.

5.2.4 Amplitude

No evidence of statistical significance of the effect of target distance on the error rate was found ($F_{1,11} = 0.20, ns$).

5.3 Movement Time

5.3.1 Latency

The effect of latency on the movement time was significant ($F_{4,44} = 58.99, p < .0001$). The interaction between the latency and the width was also significant ($F_{8,88} = 7.05, p < .0001$). Figure 12 shows the results.

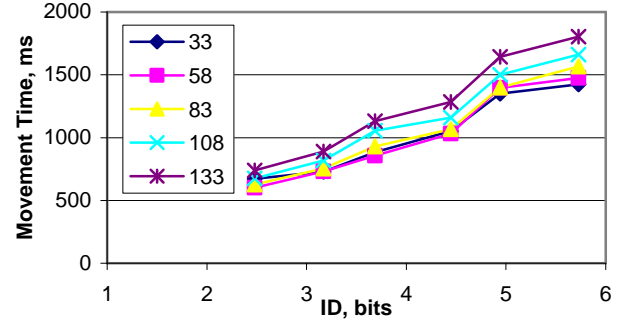


Figure 12. Movement time, for all conditions.

5.3.2 Amplitude, Width

Both amplitude and width had a significant main effect on the movement time: ($F_{1,11} = 109.92, p < .0001$) and ($F_{2,22} = 110.40, p < .0001$) respectively. See Figure 13.

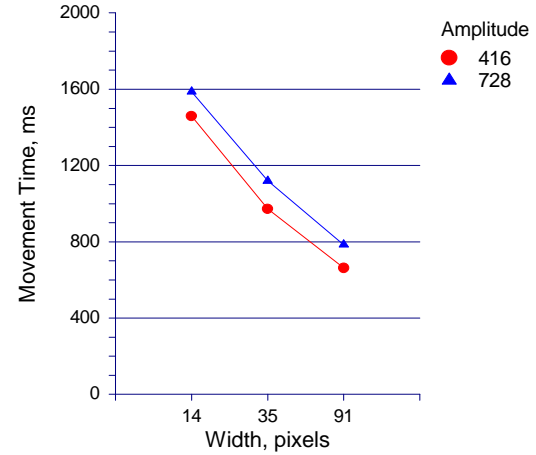


Figure 13. Movement Time as a function of amplitude and width

6. DISCUSSION

The throughput of the baseline mouse condition (i.e., no added lag, nor jitter) is similar to that reported in previous work [18]; and we take this as validation of our experimental design. Overall, we can observe that the performance drops with increased latency and jitter. Also, the error rate increases.

The jagged appearance of Figure 8 is due to the fact that our study used only six combinations of amplitudes and widths, and that these combinations did not overlap in terms of ID intervals. Many other studies, including [14], employ overlapping combinations, which smoothes the result and also hides the effect of the individual parameters.

6.1 Relationship between Jitter and Lag

For small levels of lag, e.g., up to 58 ms, there is no significant performance impact. However, when the drop in throughput begins, it is with a rate of about 0.8 bps⁵ for every 50 ms of added

⁵ 0.8 bps corresponds to about 20% of a typical mouse throughput.

latency. The throughput is affected more strongly by latency, when the targets are small, compared to larger targets. That is, the drop in performance with increased latency begins at higher levels of latency for larger targets. This might be caused by the strategy employed for target acquisition. For example, smaller targets are susceptible to overshoots and undershoots more than larger ones. Furthermore, latency had relatively low impact on error rate. The increase of latency from 33 to 133 ms caused only a moderate, increase in error rate, approximately 10–15%; see Figure 9.

Similarly, jitter has a more pronounced impact on small targets, with the smallest targets experiencing an immediate detrimental effect at the smallest increases of jitter, at a rate of ~ 0.4 bps for every 4 pixels of added jitter. However, unlike latency, jitter has a dramatic effect on the error rate: an increase of jitter from 4 to 8 pixels leads to an almost two-fold increase in errors for the smallest targets (see Figure 10). This was already observed during the experiment, as participants strongly voiced dissatisfaction in high jitter trials with smaller targets, whereas very few comments were made regarding latency. Nevertheless, the effect diminishes for larger targets and for smaller jitter levels.

Another observation is that the effective throughput depends on the index of difficulty of the pointing task. More precisely, it strongly depends on the error rate, and the error rate ultimately depends on a target size. Thus, although it might appear that only the ratio between target sizes and their spacing plays a role, the notion of effective throughput illustrates that smaller targets are disproportionately harder to hit, even if the distances are proportionately smaller. Ultimately, it suggests that Fitts' law does not hold when error rates vary greatly and target sizes vary simultaneously.

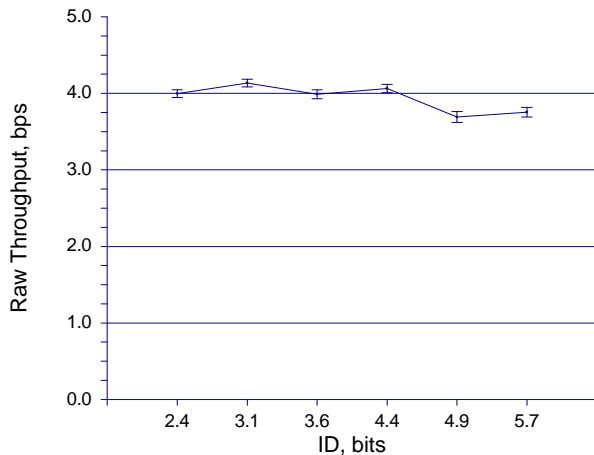


Figure 14. Raw Throughput for varying levels of ID. Note much smaller changes in throughput.

To illustrate this, we recomputed the “raw” throughput (i.e. the throughput without using effective width and distance) for the data shown in Figure 8. This new plot is shown in Figure 14. The drop at higher levels of ID, readily visible in Figure 8, is much less pronounced in Figure 14. This illustrates why the throughput metric was originally selected to assess pointing devices: when errors are rare, and when *any* hit on the target is equally good (as opposed to valuing hits closer to the center more), then the throughput is independent of the target configuration. As error rates increase, this metric loses its appeal in its original

formulation. This implies that one cannot scale interactive elements down infinitely, without expecting any effect on throughput.

6.2 Comparison with MacKenzie and Ware’s Work

It is also interesting to compare results shown in Figure 12 with a graph from a work by MacKenzie and Ware [14]. In that work the authors investigated the effects of varying levels of latency on pointing performance. In Figure 15, we overlay our data points for the zero-jitter conditions on their graph. The data from [14] is smoother than the data from our current study: this is due to overlapping pairs of amplitude-widths being used for obtaining the same indices of difficulty, as explained above. Otherwise, we believe that the data is directly comparable. We can observe the following:

- For small IDs, even their 8.33 ms condition is about as fast as our current 108 ms;
- For medium (and more typical) IDs, our 83 ms condition is slightly faster than the “old” 25 ms;
- For medium IDs, the old 75 ms condition is about as fast as the present 133 ms;
- For large IDs, the data points become reasonably close.

Unfortunately, we could not determine the exact characteristics of the system that was used in the system used by Ware and MacKenzie. As the hardware for that experiment is over 15 years old, there was no practical way to recreate the setup. Therefore, we can only speculate as to the cause of the discrepancies. Given that the number of participants is similarly small (8 vs. 12), it is *possible* that individual differences are the cause. However, the standard error for the data points in our measurements ranges from 25 ms to 50 ms for all but the two largest IDs. This is substantially less than the difference between similar (e.g., 75 ms and 83 ms; 25 ms and 33 ms) latency data points in the graphs for the two studies, which range from 100 ms to over 200 ms in movement time. Most probably there was no jitter in their system. Hence, the only viable hypothesis is that the old system contained an additional ~ 60 ms of latency. Thus, their 8.33 ms condition likely had approximately 68 ms end-to-end latency. The source of this delay is hard to assess post-hoc.

6.3 Trading Jitter for Lag

One of the original motivations for this study was to make informed choices about the tradeoff between latency and jitter, especially when smoothing. A different way is to ask this is how much filtering to apply in systems that rely on it.

From the graphs above, we can estimate that the decrease of jitter for small or medium targets from 12 to 4 pixels costs about as much in terms of throughput as a decrease in latency of 50 ms. As another example, consider that a simple averaging filter reduces jitter by a factor of 3 when averaging 9 samples, assuming that noise is random. But at a sampling rate of 125 Hz, 9 samples means 72 ms of additional delay! Thus, in this example, removing jitter will also have an associated cost in terms of performance. On the other hand, this will also afford more accuracy, which may be desirable for small target sizes (see Figure 10).

Alternatively, assume that, in a device with low latency but moderate jitter, the peak device throughput is adequate, but the

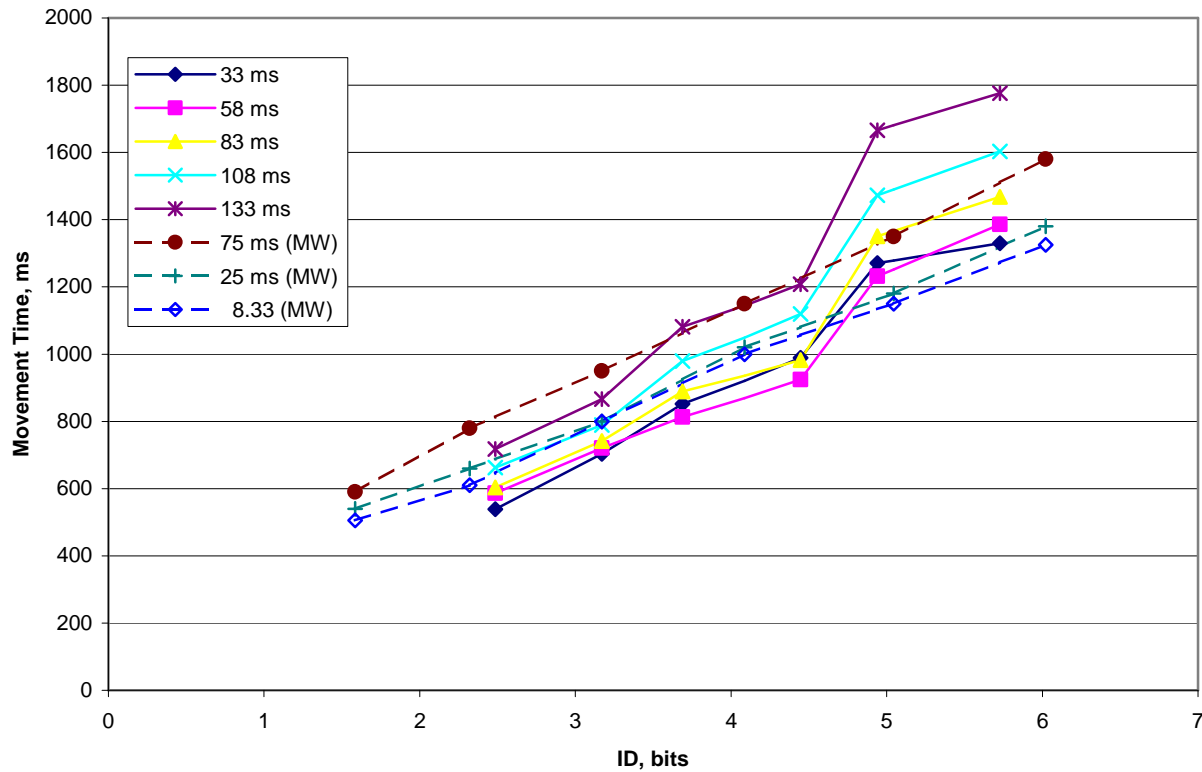


Figure 15. Movement time: Conditions from current study without jitter, compared with results from [14].

error rate is approaching 15%. Depending on how time-costly the errors are, it may be sensible to reduce the jitter through filtering. This will reduce the peak throughput, but introduce the benefit of reducing the overhead of correcting errors. Thus, this may improve interaction speed overall, depending on the cost of error correction.

7. FUTURE WORK

We presented a user studies examining the effects of device characteristics on 2D pointing tasks. In particular, we examined the effect of latency and spatial jitter. While both latency and jitter have detrimental effect on pointing performance, we are not aware of theoretical models successfully incorporating both of these factors. Using a factor proportional to lag as an additive component in Fitts' law, or as a multiplicative part (with ID) has been explored previously [14]. One of the future possibilities is to supplement such models with an additive factor, inversely proportional to jitter.

Another potential path for future work is to characterize jitter in real-world systems better. Most importantly, the jitter is not perfectly uniform and often exhibits occasional spurs that are substantially larger than can be accounted by simple white noise-based models.

Finally, since jitter may well have its strongest effect during the final stages of a movement, applying varying degrees of filtering throughout the targeting action may be worth exploring, too.

8. ACKNOWLEDGEMENTS

Thanks to Rob Teather for his help in writing the software for our experiment. Thanks also to Natural Sciences and Engineering Research Council of Canada for supporting this research.

9. REFERENCES

- [1] Agilent ADNS-3080 Optical Mouse Sensor datasheet, Agilent Technologies, 2005. Accessed April 15, 2009. <http://www.alldatasheet.com/datasheet-pdf/pdf/193358/HP/ADNS-3080.html>.
- [2] Agarwal, A., Izadi, S., Chandraker, M., Blake, A. 2007, High Precision Multi-touch Sensing on Surfaces using Overhead Cameras. In *IEEE Conference on Horizontal Interactive Human-Computer Systems* (Rhode Island, USA, October 2007). Tabletop 2007.
- [3] Cao, X. and Balakrishnan, R. 2003. VisionWand: interaction techniques for large displays using a passive wand tracked in 3D. In *Proceedings of the 16th Annual ACM Symposium on User interface Software and Technology* (Vancouver, Canada, November 02 - 05, 2003). UIST '03. ACM, New York, NY, 173-182.
- [4] Ellis, S. R., Breant, F., Manges, B., Jacoby, R., and Adelstein, B. D. Factors influencing operator interaction with virtual objects viewed via head-mounted see-through displays: viewing conditions and rendering latency. In *Virtual Reality Annual International Symposium*, 1997. IEEE 1997, 138-145.

- [5] Ellis, S. R., Young, M. J., Adelstein, B. D., and Ehrlich, S. M., Discrimination of changes of latency during voluntary hand movements of virtual objects. In *Proceedings of the Human Factors and Ergonomics Society*. Houston, Texas, 1999, 1182-1186.
- [6] Fitts, P. M., The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, vol. 47, 381-391, 1954.
- [7] Foxlin, E., Motion tracking requirements and technologies. In *Handbook of virtual environments: Design, implementation and applications*, K. M. Stanney, Ed. Mahwah, New Jersey: Lawrence Erlbaum, 2002, 163- 210.
- [8] Fukumoto, M., Mase, K. and Suenaga, Y. Real-time Detection of Pointing Actions for a Glove-free Interface. *Proceedings of MYA '92*, 473-476.
- [9] Golik, B., Wueller, D., Measurement Method for Image Stabilizing Systems, *IS+T, SPIE Electronic Imaging Conference 2007*, 51-60.
- [10] Han, J. Y., Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, UIST 2005*.
- [11] ISO/DIS 9241-9 Ergonomic requirements for office work with visual display terminals (VDTs) - Part 9: Requirements for non-keyboard input devices. International Standard, International Organization for Standardization, 2000.
- [12] MacKenzie, I. S., Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction*, vol. 7, 1992, 91-139.
- [13] MacKenzie, I. S., and Jusoh, S. An evaluation of two input devices for remote pointing. *Proceedings of the Eighth IFIP International Conference on Engineering for Human-Computer Interaction – EHCI 2001*, Heidelberg, Germany: Springer-Verlag, 235-249.
- [14] MacKenzie, I. S., and Ware, C., Lag as a determinant of human performance in interactive systems. In *Proceedings of the CHI '93 conference on Human factors in computing systems*. Amsterdam, The Netherlands: ACM, 1993, 488-493.
- [15] Mine, M. R., Characterization of End-to-End Delays in Head-Mounted Display Systems. University of North Carolina at Chapel Hill, 1993.
- [16] Myers, B. A., Bhatnagar, R., Nichols, J., Peck, C. H., Kong, D., Miller, R., and Long, A. C. 2002. Interacting at a distance: measuring the performance of laser pointers and other devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves* (Minneapolis, Minnesota, USA, April 20 - 25, 2002). CHI '02. ACM, New York, NY, 33-40.
- [17] So, R. H. Y., and Chung, G. K. M., Sensory Motor Responses in Virtual Environments: Studying the Effects of Image Latencies for Target-directed Hand Movement. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, 2005, 5006-5008.
- [18] Soukoreff, R. W., and MacKenzie, I. S., 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, 2004, 751-789.
- [19] Teather, R., Pavlovych, A., Stuerzlinger, W., and MacKenzie, S., Effects of tracking technology, latency, and spatial jitter on object movement, *IEEE Symposium on 3D User Interfaces 2009*, IBSN 978-142443965-2, 43-50.
- [20] Ware C., and Balakrishnan, R., Reaching for objects in VR displays: lag and frame rate. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 1, 4, 1994, 331-356.