In the Blink of an Eye: Investigating Latency Perception during Stylus Interaction

Albert Ng[1], Michelle Annett[2,3], Paul Dietz[2], Anoop Gupta[2], and Walter F. Bischof[3]

[1]Microsoft Applied Sciences Group
Redmond, WA, USA
v-albeng@microsoft.com

[2]Microsoft Research
Redmond, WA, USA
{paul.dietz, anoop}@microsoft.com

[3]University of Alberta
Edmonton, AB, Canada
{mkannett, wfb}@ualberta.ca

ABSTRACT

While pen computing has become increasingly more popular, device responsiveness, or latency, still plagues such interaction. Although there have been advances in digitizer technology over the last few years, commercial end-to-end latencies are unfortunately similar to those found with touchscreens, i.e., 65 - 120 milliseconds. We report on a prototype stylus-enabled device, the High Performance Stylus System (HPSS), designed to display latencies as low as one millisecond while users ink or perform dragging tasks.

To understand the role of latency while inking with a stylus, psychophysical just-noticeable difference experiments were conducted using the HPSS. While participants performed dragging and scribbling tasks, very low levels of latency could be discriminated, i.e., ~1 versus 2 milliseconds while dragging and ~7 versus 40 milliseconds while scribbling. The HPSS and our experimentation have provided further motivation for the implementation of latency saving measures in pen-based hardware and software systems.

Author Keywords

Latency; stylus; pen; pen computing; perception; psychophysics

ACM Classification Keywords

B.4.2 **[Input / Output Devices]**; H.5.2 **[User Interfaces]**: Input devices and strategies; H.1.2. **[User/Machine Systems]:** Human factors.

INTRODUCTION

Recently, there has been an explosion of stylus-enabled devices available on the market. Styli harness fine motor control to offer increased precision and accuracy compared to a finger [17, 20, 32]. This precision enables users to perform inking tasks such as note taking and sketching with ease. Unlike the mouse and keyboard, styli afford direct interaction with content, which enables users to diagram, work out equations, sketch, create calligraphy, annotate, or sign their name in a more natural and fluid manner than with a mouse or keyboard [5]. When coupled with touch, styli also afford natural bimanual interaction, supporting the transfer of behaviors and interaction techniques found with traditional pen and paper [10, 15, 16].

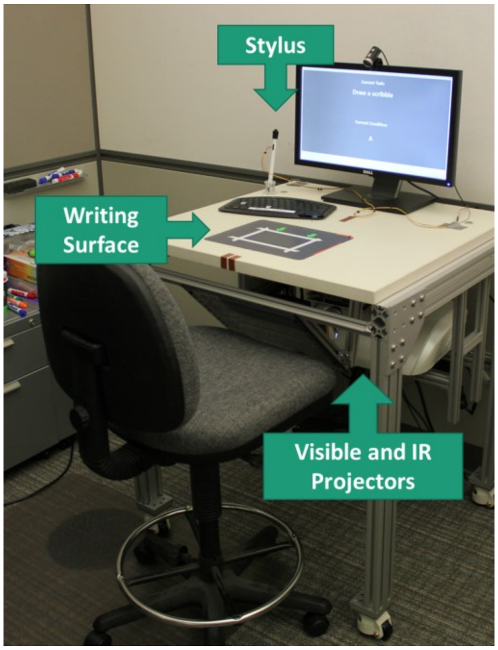

Figure 1. The High Performance Stylus System (HPSS) and experimental setup used for our latency perception studies.

The stylus experience on today's devices is unfortunately far from optimal. One of the foremost grievances regarding stylus-enabled devices involves the responsiveness, or latency, encountered while inking [28]. If digital ink does not appear directly under the stylus nib, users complain that their writing is illegible and their drawings are messy because strokes and letters are disconnected. To accommodate poor latency, users often slow their movements to allow their device to 'catch up'. Others divert their attention away from the screen, thereby sacrificing legibility and precision. If a child is learning to write using a stylus, for example, illegible ink and behavioral accommodations are unacceptable.

Regardless of the technology used to detect the stylus, the end-to-end latency on most commercial tablets is between 65 and 120 milliseconds [25]. This is significantly different from pen and paper, where latency is essentially zero as ink

CHI 2014, April 26–May 1, 2014, Toronto, Ontario, Canada.
Copyright © 2014 ACM 978-1-4503-2473-1/14/04...$15.00
http://dx.doi.org/10.1145/2556288.2557037

instantly flows from the nib onto the page. Latencies are much higher on digital devices because input information and feedback travels through a complex pipeline composed of input sampling from the digitizer, the filtering and reporting of samples to the operating system, the processing samples by the operating system, the reporting and processing of events by an application, and the updating of the display [15]. Each step in this pipeline adds to the lag or delays that users perceive.

Reducing the issues caused by latency will require advances in both hardware and software. As it is unlikely that stylus-enabled devices will ever achieve true zero latency, an achievable goal is to minimize latency such that users have the illusion, or perceive that a device is more responsive than it truly is [27]. To users, the *perceived latency* would thus seem identical to true zero latency.

Although touch and stylus interaction are both direct, there is reason to believe that the minimum perceivable latency would differ. Since a stylus nib occludes much less of the screen than a fingertip, it is likely that less occlusion, i.e., stylus-based interaction, enhances the ability to detect latency. The natural loci of attention and eye gaze required for stylus-based and finger-based tasks may also affect the perception of latency. Perception will likely differ while selecting or moving objects with the finger, where attention is focused on the visual feedback, compared to inking tasks where attention is often elsewhere. Stylus inking tasks may differ from dragging tasks, as a user may focus on the trailing ink as opposed to the stylus nib. While the level of perceivable latency has been explored using direct touch-based interaction [11, 25], the minimum perceivable latency for stylus-based interaction is currently unknown.

This work presents three main contributions. The first is the design and implementation of the High Performance Stylus System (HPSS; Figure 1). The HPSS is a stylus-enabled device capable of variable end-to-end latency from latencies found on devices today, down to a minimum possible latency of one millisecond for dragging tasks and seven milliseconds for inking tasks. Using such a system, we investigated users' perception of latency while performing a number of stylus-based tasks. The results constitute the second contribution - the identification of the minimum latency perceivable while performing stylus-based activities. The ability to test a range of delays, many of which are not achievable in the near future, provided a wealth of information and understanding regarding latency perception. The last contribution is thus an understanding of how latency is perceived and a discussion how lower latencies can be achieved with future the touch and stylus systems.

RELATED WORK

Although latency has been a complaint for years, very little work has assessed its influence on the user. Research on latency outside virtual reality and video games has focused on direct-touch input. Work by Anderson, Doherty and Ganapthy used two touchscreen devices to determine the effects of touch latencies between of 80 to 780 milliseconds during everyday tasks such as reading, photo browsing, and web browsing [3]. They found that delays of less than 580 milliseconds were acceptable to users. In work by Ng and colleagues, a low latency touch device assessed the latency perception while performing a dragging task [25]. The device consisted of an opaque touch sensor interfaced with a TI Discovery 4100 DLP development kit [8]. The kit's FPGA performed all of the processing and directly controlled a DMD at very high frame rates, allowing a baseline latency of one millisecond. It was found that users were unable to perceive latencies below six milliseconds (range 2 - 11 milliseconds). In follow up work by Jota et al., the influence of latency, target location, and width of the target was examined [11]. Performance was found to decrease as target width decreased, latency increased, and target distance increased. The just-noticeable difference while perceiving the 'land-on' portion of the dragging event was found to be 64 milliseconds. These varying results suggest that task may have a great effect on the perception of latency. The device used in this prior work was unfortunately not suitable for a stylus because the sensor resolution was suited for larger input modalities such as fingers and no disambiguation between the hand and a stylus was possible. We thus developed a new low latency prototype specifically targeted towards stylus input.

Although much further work can be done within the touch domain, this work focuses on exploring latency perception while using a stylus. Just as with direct-touch input, few have assessed stylus latency, largely due to technical limitations within digitizers and tablets. In early work with a light pen system, Miller estimated that 100 milliseconds of delay would be acceptable to users while drawing slow deliberate strokes [23]. Unfortunately, few details are available regarding the derivation of this estimate. In recent work, Henzen and colleagues developed a low latency electronic ink display for drawing and animation applications [13, 14]. The display had a minimum latency of 40 milliseconds and exhibited zero parallax. The setup was a prototype and did not undergo experimental evaluation with participants. Unlike Henzen et al.'s work, our system was capable of displaying latencies as low as one millisecond (~1.4).

HIGH PERFORMANCE STYLUS SYSTEM

Many factors in addition to latency, such as resolution and parallax, influence the stylus experience. As these factors are often confounded, the goal of the High Performance Stylus System was to build a system as close to ideal for as many parameters as possible. As latency has been shown to be perceptible down to single digit levels [25], a minimum latency of about one millisecond was targeted. As display and sensing resolution directly influence the aesthetics of the visual response, a resolution of at least 200 dpi was desired. Additionally, parallax error due to separation of the stylus from the display posed a great source of frustration for users [30], so we ensured that the stylus could actuate directly on the display with no parallax.

Guided by these requirements, the HPSS utilized two projectors, a fiber optic stylus, and a first-surface mirror for rear-projection (Figure 2). A rear-projected display, with the diffuser applied directly to the writing surface, allowed the stylus to actuate directly on the display with no parallax. An optical sensing system was used for the stylus, with another rear-projector displaying IR patterns on the diffuser. The IR patterns were detected by the stylus and later decoded to obtain the current stylus location. High resolution was achieved using HD projectors with short throw distances and low latency was achieved via hardware acceleration, as opposed to software approaches.

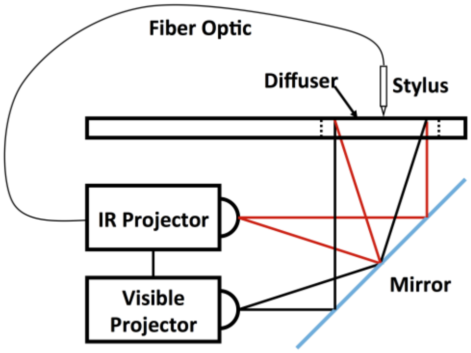

Figure 2. The system setup wherein one projector displayed IR gray-code patterns, a second displayed visible images, and an optically-based stylus detected gray-code patterns.

Given the system design, there were thus three engineering challenges to overcome, i.e. sensing, display, and processing, with each requiring a custom solution to achieve the intended level of performance.

Sensing

None of the commercial or research stylus sensing systems available met the latency, resolution, and parallax requirements that were desired. Although camera-based solutions have worked in the past (i.e., a high-speed imager views the screen from behind, and the pen emits an infrared light [16]), finding a high-speed, high-resolution camera with a low latency interface proved to be very challenging. While the sensing system used in Ng et al.'s low latency work was sufficiently fast, it was unfortunately optimized for input the size of the finger [24, 25]. Additionally, it had an opaque surface, making it unsuitable given our needs. We were thus forced to design our own sensing system utilizing fiber optics and structured IR light.

IR Projector

A Texas Instruments Discovery 4100 DLP projector development kit was used for the IR pattern projector [8]. The DLP projector used a Digital Micromirror Device (DMD), consisting of an array of micromirrors with one mirror per pixel. Each mirror had two orientations - an "on" orientation that directed light from a light source out through the lens, and an "off" orientation that directed the light away from the lens. The mirrors had response times on the order of microseconds, thus allowing for binary frame rates of tens of kHz. Typically, the DMD is multiplexed and duty-cycle-modulated to obtain color and grayscale images.

The Discovery 4100 platform provided a custom interface to a 1920x1080 DMD controlled by an FPGA. This allowed the DMD to be driven at 17,636 binary fps, using the global reset mode by omitting color and grayscale capabilities [8]. Global reset mode updated the DMD mirrors simultaneously after the pixel data has been loaded, as opposed to block reset modes, which incrementally updated portions of the DMD and hid the mirror update and settling times. While block reset modes achieve frame rates as high as 23,148 fps, the use of global reset mode avoided position-dependent latency. At 17,636 fps, with 24 frames required per sample, a sample rate of ~735 Hz was achieved, resulting in an input sampling latency of approximately one millisecond (~1.4).

The DMD was mated to an Optoma HD80 DLP projector that had its light source removed. As a sufficiently bright, non-varying, IR light source was required for the projector, an OSRAM SFH4740 20W, 850nm IR LED array was placed inside the projector's optical integrating tube. A custom heat sink coupled the light source to a commercial liquid cooling system.

Infrared Gray-code Structured Light

Lee et al.'s gray-code structured light technique was used to determine the location of the stylus [18, 19]. A 1920 x 1080 IR pattern was rear-projected onto the diffuse writing surface, measuring 148 x 91 millimeters with a resolution of 323 pixels per inch. Gray-codes were sent in the X and Y dimensions (Figure 3). To encode every pixel on the screen uniquely, $\log_2 Xdim$ and $\log_2 Ydim$ images were required. Two constant images were also included for synchronization purposes. Thus, the use of 24 images ensured that there was a unique sequence of flashes at each pixel. The codes were inverted on every other frame so that the average intensity was constant for all pixels. This allowed the sensor circuitry to obtain an accurate threshold to distinguish between light and dark pixels.

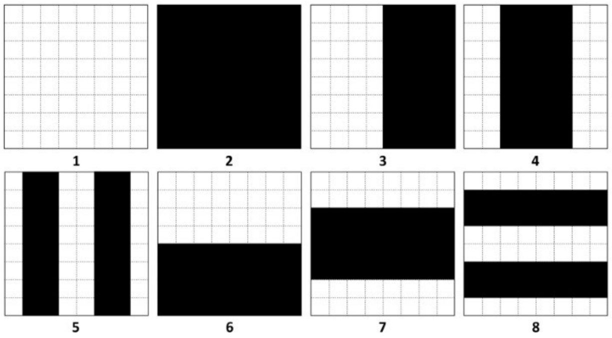

Figure 3. Gray-code patterns for an example 8x8 pixel area. The patterns include two synchronization patterns (1-2), $\log_2 Xdim$ patterns for the X coordinate (3-5), and $\log_2 Ydim$ patterns for the Y coordinate (6-8).

Using gray-codes, the encodings of two adjacent pixels differ by one bit. This property guarantees that samples between pixels will be recovered correctly. In addition, sampling errors due to fast stylus motions are minimized. While in motion, the stylus may change pixel position mid-read. Any bit boundaries that are crossed may thus be incorrectly sensed. This property guarantees that crossing a single bit boundary will generate at most a one least significant bit error. This error generalizes such that crossing bit boundaries at any given level generates a worst-case error of that bit level, independent of location. Given our high sample rate and the limitations of the human motor system, the distance that can be covered on one frame is quite small, resulting in benign errors. A simple binary encoding would not exhibit these behaviors.

Fiber Optic Stylus

As the DMD was not designed for IR efficiency, very sensitive circuitry was required to detect the IR gray-code patterns. To ensure that the stylus was thin and light, an optical fiber transmitted the received light patterns to a Hamamatsu Avalanche Photodetector, Model C5460-01 [1]. The received signal was then amplified, filtered, and quantized into a bit stream. To prevent interference from the visible light projector, an IR-blocking filter with 850 nm cutoff was used.

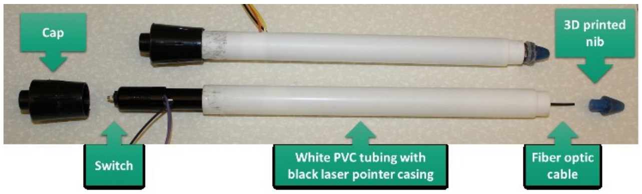

Figure 4. The fiber optic-based stylus used in the experiments.

As users vary greatly in the way they hold a stylus, it was important that the stylus could be robustly actuated at a variety of angles. We thus went through many iterations of the stylus design, finally settling on one that used a one-millimeter flexible fiber optic cable. The cable was bundled within a stylus constructed from PVC tubing, a cap, and a hollowed out laser pointer (Figure 4). The fiber optic cable was placed inside the laser pointer casing, and the casing was nested inside the PVC tubing. Attached to the end of the laser pointer casing was a momentary switch. Whenever the stylus was pressed or removed from the screen, the laser pointer (de)pressed the switch against the cap, switching the ink on or off. This simple construction allowed participants to write with the stylus at any angle comfortable to them.

As the fiber optic cable was very thin, the nib of the stylus was 3D printed from UV cured ABS plastic and a one-millimeter hole was drilled in the tip. The diameter of the nib's tip was 1.2 millimeters. The resulting stylus was 187 millimeters long, had a 13-millimeter barrel diameter, and weighed 19 grams, which is close in weight and size to a typical Wacom stylus.

Limitations and Considerations

The precision of the stylus is dependent upon the width of the optical hole in the stylus nib. Ideally, the stylus nib would be the width of one pixel. However, a one-pixel wide hole received too little light for the optical circuitry to detect, so the width was increased to one millimeter. Due to the gray-code scheme, this results in spatial jitter of about one millimeter. As jitter has been shown to decrease performance in direct pointing tasks [26], it is important to consider its effect within our system. The reduction of jitter involves filtering the samples over a certain window, thereby increasing the latency. In typical devices, increasing the filter window size would drastically increase the latency. However, due to our extremely high sample rate, we can filter over larger windows while only adding a few milliseconds of latency.

Jitter may interfere with one's perception of latency, especially at latencies as low as we explored. However, this jitter is actually constant spatial jitter that is exhibited regardless of the latency. Even with random jitter, higher latencies still yield higher gap distances between a moving stylus and the visual response on average. We thus explored the perception of latency in spite of this jitter. The exploration of jitter and the understanding of its effects at low latencies are left for future work.

Display

Similar to the IR pattern projector, another Texas Instruments Discovery 4100 projector development kit was used to display visible images. The visible projector refreshed at the maximum 23,148 binary fps using a block reset mode [8], with a pixel resolution of 1920 x 1080. This setup was similar to that used previously in the High Performance Touch System by Ng et al. [24, 25].

As the sensing system used infrared light, we needed a light source that was limited to the visible range. Thus, the DMD was mated to the optics of another Optoma HD80 projector, wherein the metal halide light source was replaced with a commercial white LED lighting unit.

Processing

In a commercial system, there is typically a CPU running a non-real-time operating system and associated applications, contributing greatly to the latency pipeline. Sensor control, display control, and stylus sample processing for the HPSS were performed by FPGAs, thereby removing the 'software latency' from the system. Each of the Discovery 4100 kits contained an onboard Xilinx Virtex 5 FPGA. The IR projector FPGA was responsible for providing stylus sample coordinates to the visible projector FPGA, while the visible projector FPGA rendered to the display and communicated with a PC for parameter control.

IR Projector FPGA Processing

The IR projector FPGA processing included five general blocks (Figure 5). A pixel generator cycled through the 24 1920 x 1080 gray-code patterns at 17,636 fps and communicated with the DMD controller to render to the DMD. These patterns were computed on a per pixel basis on the fly, without the use of an image buffer. The pattern was inverted with each scan, simplifying the design of the IR receiver by maintaining a constant average intensity level, independent of position.

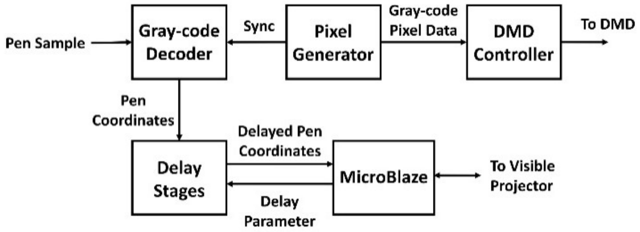

Figure 5. Block diagram of the processing performed on the IR pattern projector's FPGA.

A gray-code decoder received and decoded the pen sample bit stream from the stylus circuitry. The gray-code decoder sent the pen sample coordinates to a 100-stage variable delay buffer that introduced artificial latency into the pipeline, with a granularity of 100, 50 MHz clock periods, or 2 microseconds. This 100-stage buffer allowed for the introduction of up to 100 pen sample periods (approximately 140 milliseconds) of latency without losing any samples.

A Xilinx MicroBlaze soft processor controlled the variable delay, received the pen sample coordinates, and sent the coordinates over a high-speed serial link to the visible projector FPGA. Additionally, the MicroBlaze accepted commands from the visible projector FPGA to set the delay.

Visible Projector FPGA Processing

The processing performed in the visible projector FPGA was organized into four blocks (Figure 6). A MicroBlaze soft processor received pen sample coordinates from the IR projector FPGA, ran simple applications, and rendered images to the frame buffer. The soft processor also communicated with a host PC over serial to allow for the selection of applications and the controlling of various parameters such as latency.

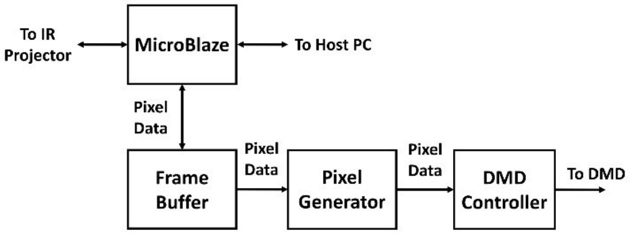

Figure 6. Block diagram of the processing performed on the visible projector's FPGA.

Calibration was necessary to map the coordinates from the IR and visible pattern projectors. A linear mapping between the two projection planes using five coordinate pairs was used to compute a sufficiently accurate mapping. The mapping was applied by the MicroBlaze.

The frame buffer was synthesized in the on-chip BRAM blocks. The Virtex 5 only had enough BRAM for a 1280 x 800 image, so the active area was limited to a portion of the full image. The DMD Controller continuously read out frames at 23,148 fps to render to the DMD.

Applications

Two applications were implemented on the MicroBlaze, a box dragging application and an inking application. Both applications received input coordinates through an interrupt event handler. On each frame of the box dragging application, a box was drawn at the sensed location. The minimum latency of the box dragging application was approximately one millisecond (~1.4), with latency being due to the sensing mechanism and rendering latency.

Within the inking application, the unfiltered ink trail was not smooth due to the jitter described previously, so a moving average filter with a window size of 10 filtered each coordinate and the Bresenham method [6] drew a line to the previous coordinate. Since the ink thickness was no more than a few pixels wide, the latency introduced while rendering was negligible. Due to the 10-frame filter window, the minimum latency of the inking application was five frames, or approximately seven milliseconds.

To minimize latency, rather than double buffering and redrawing the entire frame, the application simply modified the pixels in the frame buffer that changed between frames. Because the images were drawn so quickly, no tearing of the image was apparent, even without the double buffers.

Latency Verification

Accurately measuring the end-to-end latency of such a high performance device is a great challenge. Recently, Berard and Blanch proposed statistical methods to measure the distribution of latencies in the face of high variability in common touch devices [4]. As our system was developed to tightly control latency by design, such methods were not required. All processing and controls were in the form of a clock-cycle granular FPGA design and primitive MicroBlaze RISC processor software that was measurable with timer registers. By design, there was no significant variable latency in the system. The baseline latency was limited by the sample rate of the stylus, which was approximately one millisecond (~1.4) per sample. Image rendering on the MicroBlaze was below 0.1 milliseconds, with the latency from the remainder of the FPGA logic being negligible. The artificially introduced latency was accurately settable in two microsecond increments.

A high-speed camera was also used for further verification [25]. From the camera footage, the gap distance and stylus velocity of a stroke were used to compute the estimated latency. This approach allowed us to verify the accuracy of our designed latencies to within several milliseconds. The high-performance time-driven design, combined with the high-speed camera footage, provided us with sufficient evidence that the system was performing as intended.

USER STUDY

As little is known about the perception of latency while inking or using a stylus, three psychophysics experiments were conducted. Each was designed to determine the lowest latency detectable using a task hypothesized to increase latency perception, i.e., dragging and scribbling. Such tasks ensured ecological validity and comparability to prior work.

Participants

Sixteen naïve individuals (4 female) participated in the study (M = 33, range 24-52 years). All participants were right handed and had normal or corrected-to-normal vision. Participants had varying levels of exposure to tablets and styli, from complete novices to others who worked with stylus-enabled devices each day. Each participant was provided a $10 honorarium for the 30-minute experiment.

Tasks

Three tasks were used to determine the lowest latency participants could perceive: *large box dragging*, *small box dragging*, and *scribbling*.

In the *large box dragging* task, participants placed their stylus on the left hand side of the screen underneath the left arrow, dragged the stylus laterally to the right hand side underneath the right arrow, and then dragged the stylus back underneath the left arrow (Figure 7). While doing this, a 20 millimeter x 20 millimeter white box was continually centered at the location of the nib. To maintain consistency with prior work, the same box dimensions were used [25]. We also verbally encouraged participants to maintain the same dragging speed throughout all trials.

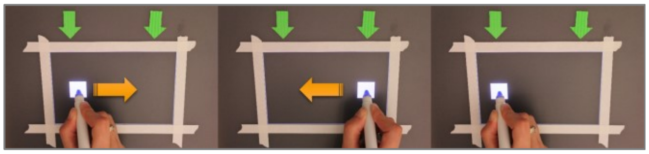

Figure 7. During box dragging, a box appeared around the stylus and participants dragged the stylus from the left to right and back to the left. The orange arrows have been added to indicate the direction of stylus movement.

As the size of the nib was much smaller than the finger, a variation of the box dragging task, *small box dragging,* was also conducted. By closely matching the size of the box to the dimensions of the nib, it should have been much easier for participants to notice if the nib fell outside the box or was not in the center, and hence enhance their ability to perceive latency. To compute the appropriate box dimensions for this condition, we determined the finger-to-box ratio used by Ng et al. using the average width of the index finger at the proximal interphalangeal joint, i.e. 16 millimeters [9], and the large box dimensions of 20 x 20 millimeters. Using this ratio and the size of our nib, a 6.25 x 6.25 millimeter box was appropriate for this condition.

In the *scribbling* task, participants drew a curvilinear line, or scribble, starting in the upper left corner and moving towards the lower right corner (Figure 8). To encourage the same motor motions, participants were instructed not to touch the edge of the interactive area with their scribble nor draw a shallow, wavy line. Feedback was provided in the form of a 1-pixel thick on-screen ink trace. Similar to the dragging tasks, participants were verbally encouraged to maintain the same speed throughout all trials.

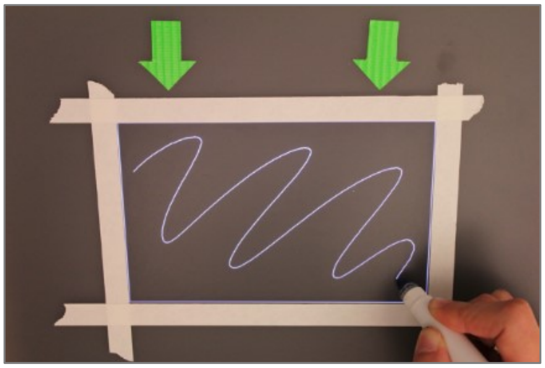

Figure 8. In the scribbling task, participants were instructed to move the stylus from the upper left down to the lower right.

Methodology

Similar to Ng et al. and Jota et al., a psychophysical just-noticeable difference methodology governed the latencies presented to participants and ultimately converged at the minimum perceivable latency [11, 25]. JND paradigms are used to determine perceivable latency because they assume that the converged latency threshold is the result of participants being unable to discriminate between the baseline and all latencies below the converged threshold. We thus assume that participants would not perceive such latency on an actual device because they were unable to discriminate it from the minimum baseline latency.

The baseline latency while dragging was one millisecond and seven milliseconds while scribbling. The testing latency was refined using the Parameter Estimation by Sequential Testing (PEST) adaptive technique [28] because it allowed for maximal efficiency, whilst maintaining comparability to legacy JND methodologies such as staircase methods used in prior work. Such efficiency allowed for maximal engagement and minimal fatigue, which given the repetitive nature of the motor movements required was necessary.

With PEST, the choice of the testing latency was dictated by the history of a participant's prior responses at the given stimulus level. PEST determined the step size using the Wald sequential likelihood-ratio test [31]. For comparability to Ng et al. and Jota et al.'s work, the initial testing latency was 65 milliseconds, the step size was 8 milliseconds, and an expected probability of 75% was used. Our experiment terminated using the Minimal Overshoot and Undershoot Sequential Estimation (MOUSE) technique [21]: once the step size reached one millisecond, the experiment concluded. The JND was the last testing latency participants experienced before the experiment terminated.

Equipment and Apparatus

During the experiment, a C# and WPF program running on an HP Z400 Workstation automatically determined the appropriate latency values to send to the HPSS (over serial) using the PEST algorithm. The program also recorded all participant responses and the final JND values obtained for post-analysis. A Dell 21" monitor provided feedback about the current task and trial condition.

Procedure

During the recruitment process, participants were naive to the purpose of the experiment. This removed any bias or experience they may have had towards latency from pen or touch-enabled systems or video games. At the start of each experiment, participants sat in an adjustable drafting chair in front of the High Performance Stylus System. Latency was explained to participants, and participants were informed that we were measuring the minimum latency they could perceive while performing a number of tasks.

Each trial was composed of two conditions. In the first condition, participants completed the task using one latency, either the baseline or target. Participants then pressed the space bar to go to the next condition where they completed the task again using the other latency value. Once finished, participants again pressed the space bar. A tone played and participants then indicated "which condition had less delay" using the A or B keys. Once PEST had converged at their JND level, participants had a break and the next task began. The presentation order was counterbalanced using a Latin square design.

Although forcing participants to attend to latency intentionally primes them, the use of subjective questions, such as "Which condition did you like more" would have allowed for too much interpretation of what could be focused on or attended to by participants, confounding our results.

RESULTS

As the baseline latencies for the dragging and scribbling tasks were different (i.e., one millisecond versus seven milliseconds), the two tasks could not be statistically compared. As such, two separate analyses were conducted.

Even though the baseline latencies for the dragging tasks were the same, the resulting JND threshold values were not normally distributed (Figure 9, Figure 10). A Wilcoxon signed-rank test compared the two box sizes and revealed that participants were able to discriminate between lower latencies when the smaller box (*Mdn* = 2 milliseconds, range 2 - 16 milliseconds) instead of the larger box (*Mdn* = 6 milliseconds, range 2 - 7 milliseconds) appeared around the nib, $z = -2.769$, $p < 0.01$, $d = 1.03$. Users were able to perceive minute differences caused by latency, similar to results found prior by Ng et al. [25]. The different threshold values suggest that the dimensions of visual feedback in reference to the physical input likely play a role in the perception of latency. Dimensions that complement the relation between the physical reference and digital feedback provide better visual cues and assist in latency perception.

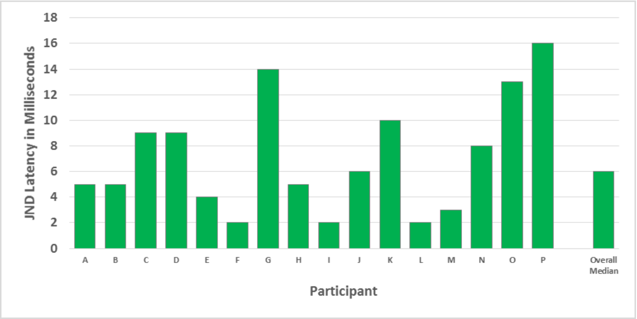

Figure 9. Minimum latency perceived by participants while dragging the large box. Results ranged from 2 to 16 milliseconds, with a median latency of 6 milliseconds.

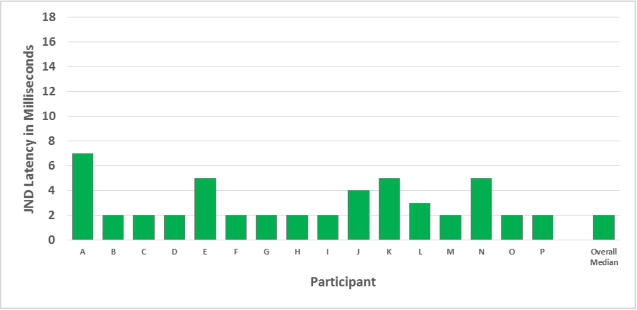

Figure 10. Minimum latency perceived by participants while dragging the small box. Results ranged from 2 to 7 milliseconds, with a median latency of 2 milliseconds.

The results from the scribbling task demonstrated that participants were able to discriminate between the 7-millisecond baseline latency and a median of 40 milliseconds (Figure 11; range 10 - 70 milliseconds). Although not directly comparable to the dragging tasks, such results suggest that task demands may play a role in the perception of latency. The higher perceived latency found while scribbling compared to dragging are likely due to the different visual feedback available, the strategies used to determine latency, or the cognitive loading encountered while scribbling versus dragging the box.

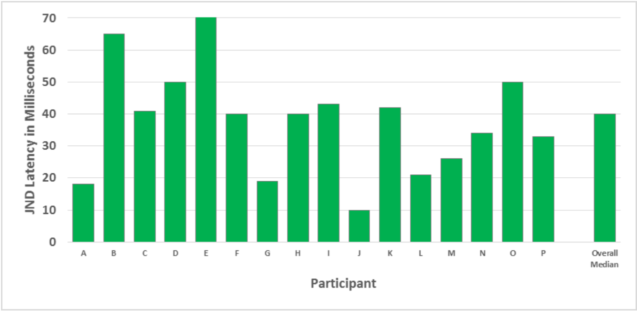

Figure 11. Minimum latency perceived by participants while performing the scribbling task. Results ranged from 10 to 70 milliseconds with a median latency of 40 milliseconds.

DISCUSSION

The results found with the dragging and scribbling tasks provide valuable information about the basic visual process and the ability of participants to detect and notice latency. Given the latencies possible on devices today, i.e. 65 - 120 milliseconds, these results additionally provide support for decreasing latency along all levels of the pipeline.

Understanding the Perception of Latency

Users appear to detect latency by attending to the visual disparity between a visual reference, which may move, and corresponding feedback available from the display. The perceptibility of this disparity appears to depend on a number of factors. The size and type of feedback presented likely influenced participant's comparisons. With the scribbling task, a small, persistent line was visible, whereas with the dragging tasks, a large box was visible. As there was minimal occlusion from the stylus nib, it is likely that participants had an easier time viewing feedback as the box(es) activated a larger area within the foveal or parafoveal region. In the original latency experiments by Ng et al., the finger occluded a larger area of the visual feedback than our stylus nib, so this may explain why our participants found it easier to detect latency, especially when the dimensions of the physical input were appropriately scaled to the feedback. When considering Jota et al.'s tapping task, it is likely that the haptic feedback generated by touching the screen was integrated with the visual feedback from the display to make latency judgments. The combination and synchronization of feedback from other modalities likely decreases the perception of latency, hence the higher thresholds found by Jota et al. Further work is thus needed to truly understand the role of feedback in the perception of latency.

As participants were free to use whatever method necessary to make their latency judgments, it is also possible that the location of focus and subsequent visual cues available influenced perception. When asked how comparisons were made, participants reported different focus loci for the dragging versus scribbling tasks. While dragging, participants reported fixating on the box's edges to determine if the nib was inside, outside, or in the center of the box, likely similar to what occurred in the prior touch work by Ng et al. As there was high contrast on the screen, and little occlusion from the nib, such a determination was easier with the stylus than with touch. The smaller the box, the easier it was to make this judgment. While scribbling, the location of focus was not always on the nib. Some participants reported focusing on the ink, others on the pen or hand, some continually switched between the ink and the pen, while others held a bird's eye view of the whole screen. Such strategies changed the location of the nib and ink on the eye as well as the distance between them. While varied, such attentional patterns made different visual cues available to participants and fall in line with those found in the eye tracking literature [2, 7, 12, 22].

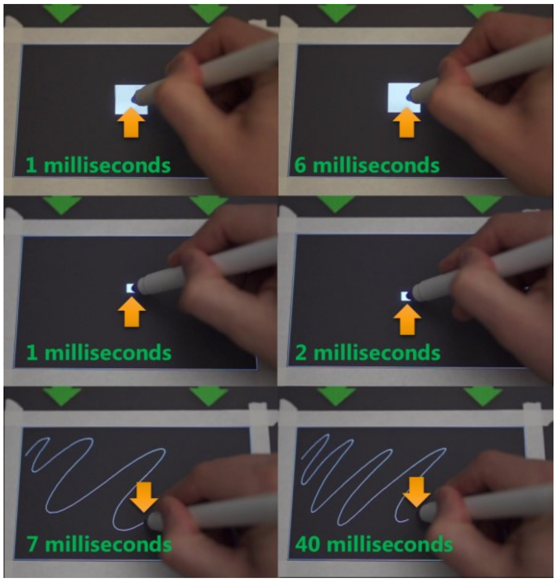

Figure 12. Demonstration of the latencies perceivable by participants, with the yellow arrow indicating the current nib location. From top: The nib was on the right side of the box during the large box dragging task, the nib was slightly off center during the small box dragging task, and there was a visible gap between the nib and ink while scribbling. The images were recorded using a high speed camera at 480 fps.

Cognitive load and attention required to complete a specific task may also influence latency perception. The tapping and dragging tasks used in this and prior work with touch had low levels of cognitive difficulty and did not divert much attention from the latency judgments being made. Repeatedly drawing the same pattern on the screen and trying to maintain its shape and length (scribbling), on the other hand, is more cognitively challenging and attention diverting because participants had to focus on maintaining the same shape and speed of stroke on each trial. Although our scribbling task is a rudimentary form of inking, it is likely that tasks with even more cognitive and attentional demands such as taking notes, solving mathematical equations, or sketching a portrait, will divert more cognitive resources away from latency perception and likely decrease latency perception even more. This, of course, requires further investigation. Based on our initial results, it may be reasonable to target latencies closer to 40 milliseconds for stylus-based experiences, instead of the 10 milliseconds recommended prior for touch-based tasks.

While our results provide many insights into the perception of latency, they are not without their limitations. Approximately 63% of participants could discern between one and two milliseconds of delay. When asked if they were guessing or confident in their comparisons, most indicated that they could still discriminate between the two conditions, but it was difficult to do so. There was a very small difference in the location of the box for one versus two milliseconds, compared to one versus six milliseconds (Figure 12). As the human eye can discern motion or deviations in spatial patterns separated by one arc minute (1/60th of a degree), if the HPSS could display latencies of finer fidelity, e.g., 1.5, 1.6, etc. or the baseline latency could be lowered below one millisecond, it is likely that most participants would be able to discriminate between latencies smaller than one millisecond while performing dragging tasks.

The results demonstrate the minimum latency users can detect, but under conditions where users were explicitly primed to look for latency and do not have attentional competition. In situations where explicit cueing is not available, the minimum noticeable latency will likely be higher. Although the results suggest that developers and device manufacturers have much further work before latency is no longer a complaint from users, they should be considered carefully.

Improvements Necessary for Low Latency in the Future

The High Performance Stylus System was not intended to be a commercially viable technology. Rather, it was a system through which extremely low latency stylus interaction could be experienced and evaluated. Bringing such experiences to future commercial systems will involve substantial development and innovation across *all* sensing, processing, and display subsystems.

The latency of the HPSS was primarily limited by the sensing mechanism. Twenty-four frames were required for the full 1920 x 1080 sensing area, resulting in a sampling latency of approximately one millisecond. To increase the sample rate and reduce the latency, one could take advantage of the fact that a stylus will not move very far between samples, especially at such high sample rates. Instead of scanning the full frame every sample, the system can sample only a small region of interest about the last sample point, at much higher rates. This technique can raise the sample rate by several factors.

The HPSS hardware was successful in producing extremely low latencies due to the extremely high sampling rates across the sensing, display, and processing subsystems. The current systems of today run nowhere near as fast as the HPSS. Often, the sampling frequencies of the stylus digitizers are low (~100 Hz), display frame rates are low (60 Hz or possible 120 Hz if stereo is supported), and applications typically run at 30 Hz, resulting in high latencies within each subsystem. In addition, unlike current commercial systems, the subsystems of the HPSS were synchronized. Upon sampling the pen location, all processing and display updates were performed within one frame, resulting in only one frame of latency. In current systems, subsystems are pipelined for simpler composability, resulting in several frames of latency. Across all of these subsystems, innovations and improvements in the update rate and synchronization processes are required before users can experience true low-latency when using commercial devices.

The Accelerated Touch toolkit has been proposed previously as a method to decrease perceived latency [25]. By combining high performance and current generation hardware, one can render both low and high latency feedback and visualizations. Ng et al. hypothesized that the combination of two such 'layers' of visual information could be used to mitigate many of the issues and complaints about latency. While certainly possible today, such an approach does however require the augmentation of existing hardware with a low latency system.

Outside of high performance hardware approaches, there are a number of software-based pen-specific enhancements that can be implemented today. The use of pen location and stroke prediction algorithms can not only increase the smoothness and beautification of strokes, but can also be used to pre-render strokes before they occur and then adapt them after the stylus has reached or surpasses a given point. The current application context and knowledge about hand postures or grip could also be useful in sampling sub-regions of the input sensor and redrawing targeted sub-regions of the display. Vogel's hand occlusion model [29], for example, could be implemented within an inking application to define the region of interest. As the location of the stylus reveals much about the intentionality of current and future interaction, harnessing it within applications should help decrease the latency perceived by users and ultimately improve the stylus experience both today and in the future.

CONCLUSION

In this work, we explored the perception of latency using a prototype stylus-based system, the High Performance Stylus System. The HPSS was composed of two high-speed DLP projectors and a fiber optic stylus and enabled participants to experience latencies as low as approximately one millisecond while dragging objects and seven milliseconds while inking. Such low latencies were made possible by performing all sensing and processing on FPGAs, thereby removing the latency typically introduced by the operating system and application layers.

Through experimentation, we determined that users can perceived latencies at substantially lower levels than those possible on devices today, i.e., 2 milliseconds while dragging a box and 40 milliseconds while scribbling. Our results corroborate with prior work and suggest that the perception of latency is influenced by the visual feedback available, location of focus, and the cognitive demands of the task. Future experimentation of these factors, in addition to hardware-based factors such as parallax and resolution, should broaden our understanding of latency and its role in the stylus experience.

ACKNOWLEDGEMENTS

We express our sincerest gratitude to Microsoft's Applied Sciences Group and Microsoft Research's Hardware Lab for their assistance in designing and assembling the High Performance Stylus System. We also thank Chris O'Dowd, Patrick Therien, Ken Hinckley, and Ed Burress for their assistance with the HPSS's stylus.

REFERENCES

1. APD module C5460-01 | Hamamatsu. https://www.hamamatsu.com/jp/en/product/alpha/A/4162/C5460-01/index.html. Accessed December 2013.
2. Alamargot, D., Chesnet, D., Dansac, C., and Ros, C. Eye and Pen: A New Device for Studying Reading During Writing. *Behavior Research Methods*, 2006, 38(2), pp. 287 - 299.
3. Anderson, G., Doherty, R., and Ganapathy, S. User Perception of Touch Screen Latency. *Design, User Experience, and Usability Part 1*, 2011, pp. 195 - 202.
4. Bérard, F. and Blanch, R. Two Touch System Latency Estimators: High Accuracy and Low Overhead. In *Proc. of ITS,* 2013*,* pp. 241 - 250.
5. Brandl, P., Forlines, C., Wigdor, D., Haller, M., and Shen, C. Combining and Measuring the Benefits of Bimanual Pen and Direct-Touch Interaction on Horizontal Interfaces. In *Proc. of Advanced Visual Interfaces*, 2008, pp. 154 - 161.
6. Bresenham, J.E. Algorithm for Computer Control of a Digital Plotter. *IBM Systems Journal*, 1965, 4, pp. 25 - 30.
7. Calgi, R.C., Coraggio, P., Napoletano, P., Schwartz, O., Ferraro, M., and Boccignone, G. Visuomotor Characterization of Eye Movements in a Drawing Task. *Vision Research*, 2009, 49(8), pp. 810 - 818.
8. DLP Discovery 4100 Development Kit. http://www.ti.com/lit/er/dlpu008a/dlpu008a.pdf. Accessed December 2013.
9. Dreyfuss, H. and Tilley, A. *The Measure of Man & Woman*. 2002, John Wiley & Sons, Inc., New York.
10. Guiard, Y. Asymmetric Transfer of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. *Journal of Motor Behavior*, 1987, 19, pp. 486 - 517.
11. Jota, R., Ng, A., Dietz, P., and Wigdor, D. How Fast Is Fast Enough? A Study of the Effects of Latency in Direct-Touch Point Tasks. In *Proc. of CHI,* 2013, pp. 2291 - 2300.
12. Gowen, E. and Miall, R.C. Eye-tracking Interactions in Tracing and Drawing tasks. *Human Movement Science*, 2006, 25, pp. 568 - 585.
13. Henzen, A., Ailenci, N., Di Fiore, F., Van Reeth, F., and Patterson, J. Sketching with a Low-Latency Electronic Ink Drawing Tablet. In *Proc. of Graphite,* 2005, pp. 51 - 60.
14. Henzen, A., Ailenci, N., Di Fiore, F., Van Reeth, F., Vansichem, G., Zehner, R.W., and Amundson, K. An Electronic Ink Low Latency Drawing Tablet. In *SID International Symposium Digest of Technical Papers,* 2004, 35(1), pp. 1070 - 1073.
15. Hinckley, K. and Wigdor, D. Input Technologies and Techniques. *The Human Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, 2002.
16. Hinckley, K., Yatani, K., Pahud, M., Coddington, N., Rodenhouse, J., Wilson, A., Benko, H., and Buxton, B. Pen + Touch = New Tools. In *Proc. of UIST*, 2010, pp. 27 - 36.
17. Holzinger, A., Holler, M., Schedlbauer, M., and Urlesberger, B. An Investigation of Finger versus Stylus Input in Medical Scenarios. In *Proc. of Information Technology Interfaces*, 2008, pp. 433 - 438.
18. Lee, J.C., Dietz, P., Maynes-Aminzade, D., Raskar, R., and Hudson S. Automatic Projector Calibration with Embedded Light Sensors. In *Proc. of UIST*, 2004, pp. 123 - 126.
19. Lee, J.C., Hudson, S.E., Summet, J.W., and Dietz, P. Moveable Interactive Projected Displays Using Projector Based Tracking. In *Proc. of UIST*, 2005, pp. 63 - 74.
20. Mackenzie, I.S., Sellen, A., and Buxton, B. A Comparison of Input Devices in Elemental Pointing and Dragging Tasks. In *Proc. of Computer Human Interaction*, 1991, pp. 161 - 166
21. McMillian, N.A. and Creelman, D.C. *Detection Theory: a User's Guide*. 1991, Cambridge University Press, Cambridge.
22. Miall, R.C., Imamizu, H., and Miyauchi, S. Activation of the Cerebellum in Co-Ordinated Eye and Hand Tracking Movements: An fMRI Study. *Experimental Brain Research*, 2000, 135, pp. 22 - 33.
23. Miller, R.B. Response Time in Man-Computer Conversational Transactions. *Fall Joint Computer Conference*, 1968, pp. 267 - 277.
24. Ng, A. and Dietz, P. The Need for Speed in Touch Systems. In *SID International Symposium Digest of Technical Papers,* 2013, 44(1), pp. 547 - 550.
25. Ng, A., Lepinski, J., Wigdor, D., Sanders, S., and Dietz, P. Designing for Low-Latency Direct-Touch Input. In *Proc. of User Interfaces and Software Technologies* 2012, pp. 453 - 462.
26. Pavlovych, A. and Stuerzlinger, W. The Tradeoff between Spatial Jitter and Latency in Pointing Tasks. *ACM SEICS*, 2009, pp. 187 - 196.
27. Seow, S. *Designing and Engineering Time: The Psychology of Time Perception in Software*. 2008, Pearson Education, Boston.
28. Taylor, M.M. and Creelman, C.D. PEST: Efficient Estimates on Probability Functions. *Journal of the Acoustical Society of America*, 1967, 41, pp. 782 - 787.
29. Vogel, D. *Direct Pen Input and Hand Occlusion*. PhD Thesis, University of Toronto, 2010.
30. Vogel, D. and Balakrishnan, R. Direct Pen Interaction with a Conventional Graphic User Interface. *Human Computer Interaction*, 2010, 25, pp. 324 - 388.
31. Wald, A. *Sequential Analysis*. 1947, John Wiley & Sons, Inc. New York.
32. Zabramski, S. Careless Touch: A Comparative Evaluation if Mouse, Pen, and Touch Input in Shape Tracing Task. In *Proc. of the Australian Computer-Human Interaction Conference*, 2011, pp. 329 - 332.