Digitizer Technology: Performance Characteristics and the Effects on the User Interface

Jean Renard Ward and Mark J. Phillips
Pencept, Inc.

Tablet digitizers are common devices for getting graphical data into a computer. For many applications, they are the only practical device. Because tablet digitizers have been in existence for many years, they are often regarded as "old hat" technology—nothing new can be said about them and no significant technical developments other than price decreases can be expected.

This is a misconception. We have found several misunderstood performance characteristics, characteristics that can have a major effect on the user's ability to use a tablet digitizer in many interactive applications. The risk to an application designer is that he or she may base the interface design on incorrect assumptions. As a consequence, the interface may be awkward to use.

This article shows what some of these characteristics are and how to evaluate their effect in different classes of applications. Specific examples are taken from a hand-printing application for character recognition.

In the commercial world, tablets are usually described with the following performance specifications:

- active drawing area
- data rate (time resolution in points per second)
- spatial resolution (in physical distance)
- worst case position accuracy (from absolute position)
- pointer type (pen stylus or puck)

The first four are usually measured with a puck pointer held stationary. For a pen stylus, sometimes the worst case position error from tilting the stylus will be given, but frequently this is ignored.

These specifications can fail to show how well (or badly) the tablet performs for a given user action for two reasons. First, the specifications have been measured for one type of action, but perhaps not for the action that is important in a given application. Second, the specifications do not describe the nature of the error, only its magnitude. The nature of the error may be important in a specific user action, since it may determine whether the user or the application can overcome it by some means of correction.

Mutability of tablet performance

Most designers incorporating a digitizing tablet into a system are not so familiar with signal processing techniques as with computer graphics technology. They often overlook the possibility of trading performance in one dimension for performance in another. We have found that common digitizers typically have in one or more domains excess quality (relative to the needs of the application) that can be manipulated to improve quality markedly in domains where the digitizer is deficient.

For example, if a tablet samples the position of the pointer faster than is actually needed, averaging several input values together will substantially reduce the magnitude of any random positional errors. Different filtering or averaging algorithms can correct for Gaussian noise, wild data, and slew rate errors. The effective result will be a digitizer with much better accuracy, but a somewhat lower sampling rate.

Conversely, an application might require a higher sampling rate, but not involve the tracking of sudden, small changes in position. An example is human-generated animation input, which needs smooth, nonjerking motion more than the ability to read sudden changes in

0272-1716/87/0400-0031$01.00 © 1987 IEEE

position. Digitizer data could be interpolated between points by different methods to "improve" the sampling rate: Straight-line interpolation is a very simple method, and produces points only in integer multiples of the input data rate. Other algorithms have much better "tracking," and can simulate any desired data rate. The result is "smoother" positioning from more intermediate points, with a loss of sudden-movement detail.

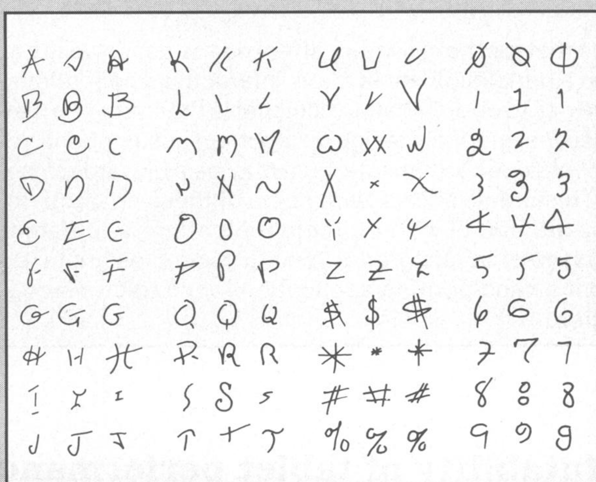
In general, there are methods to overcome any specific deficiency of a digitizing system. The questions for the applications designer are what characteristics are important, and how different characteristics should be traded off against each other.

Example: a performance-critical application for digitizing tablets

Some of the illustrations that we use in this article are based on one use of digitizers: on-line recognition of handwritten text. This application makes perhaps the most severe demands on digitizer performance for spatial resolution, position accuracy, and time resolution.

The requirements that make dynamic character recognition (DCR) so demanding are as follows:

- the need to capture small features
- the need to keep up with a high continuous writing speed
- the requirement to write with a pen stylus, not the larger, more accurate puck
- the need for very good absolute positioning accuracy, for applications where the user is writing on a preprinted data-entry form.



Acceptable handwritten characters for Pencept DCR.

Background: tablet sensing technologies

Many performance problems of digitizer tablets result from the particular design of the tablet. Most tablets use electromagnetic/electrostatic or resistive sheet sensing technology. Other methods are acoustic sensing in air,[1,2] surface acoustic wave sensing,[3] and mechanical sensing.[4,5] Each design is prone to certain performance problems. Before we discuss particular performance characteristics, we will review the common technologies.

Electromagnetic/electrostatic tablets

Electronic sensing tablets typically have an x/y grid of conductors under the tablet surface, spaced from 0.1″ to 0.5″ apart, and a loop of wire within the pointer. The position of the pointer is determined by exciting either the grid or the loop with an electromagnetic pulse, and sensing the induced voltage, current, or spatially dependent phase[6] of a sinusoidal signal in the other. The tablet conductors are scanned to find the conductor closest to the loop (rough position), and the sensed pulse measured to interpolate the precise position between conductors. Usually several pulses are sensed and averaged to give a better final value.[7]

Either the loop can be the transmitter and the tablet grid the receiver,[8] or vice versa.[9] If the pointer is the receiver, it is harder to shield and is more likely to suffer when put near sources of electromagnetic noise (such as a color VDT), causing it to report spurious or inaccurate position data.

Resistive sheet tablets

Another group of technologies is based on measuring the voltage gradient across a resistive sheet. One design uses layers of conductive and resistive material with a spacing between the layers.[10,11] A voltage gradient is applied across one of the layers in one coordinate direction. When a pen tip or other object presses on the layers, the conductive layer gets the voltage at that point in the resistive layer. The voltage can be measured to determine the point of contact along that ordinate. The design has the advantage that it can use an ordinary pen or fingertip. One disadvantage is that a "light touch" can give bad position data due to contact resistance; a "broad touch" (such as a whole finger tip) will give some variable centroid value.

A similar technique is to vacuum deposit a resistive layer on a hard surface and induce a voltage gradient by applying a voltage with the tip of the writing stylus. At least one design uses capacitive coupling to a signal in the stylus[12] instead of dc voltage.

The most notable feature of these designs is that transparent materials can be used to make a "see-through" digitizer. We have found that the performance can be limited by the manufacturing uniformity of a resistive layer of material, whether it is rolled and pressed, or vacuum deposited on a hard surface.

One note: Several applications in recent years involve mounting a digitizer directly on a display, resulting in much better resolution than that of either a light pen or most capacitive or infrared touch screens. But this can introduce new problems. The application involves pointing to features or objects on the display. The display position can change with line voltage or the display can be stretched in portions as much as 10 percent, introducing errors much worse than those of the lowest accuracy digitizer we have ever seen. LCD and plasma panel displays are "flatter," but some generate strong electrical noise and hurt digitizer performance.

Acoustic tablets

Acoustic tablets use either the travel time or the phase of a standing wave for a sound pulse from a transducer to a sensing microphone to compute the position of a pointer.[2,13] This is perhaps the easiest technology for digitizing in three-space, since each transmitter/microphone link can be designed to work well in any direction. Aside from the obvious environmental problems in nonbenign settings (two digitizers operating next to each other, for example), the nonuniformity of air as a medium can also cause substantial performance variation.

For high accuracy, these tablets have to be calibrated for local air temperature and altitude pressure. Local variances, such as a draft[2] or the heat from a cigarette or electrical equipment, can affect accuracy. A 5° C change in temperature near 20° C changes the speed of sound in air by 0.8 percent.[14] If applied to an 11″ sensing distance, this produces an error of 0.08″, which is larger than the nominal error of most tablets.

True characteristics of digitizer performance

Since the usual performance measures used by vendors on their tablets are not adequate, in this section we show what the true characteristics are. We also describe the parts of a digitizer design that could cause problems, the applications that they might affect, and ways the application designer can correct them. We think that the most important performance measures are the following:

- missing coordinates
- monotonicity
- output continuity
- slew rate
- rectilinear displacement
- scaling error
- orthogonality
- differential error

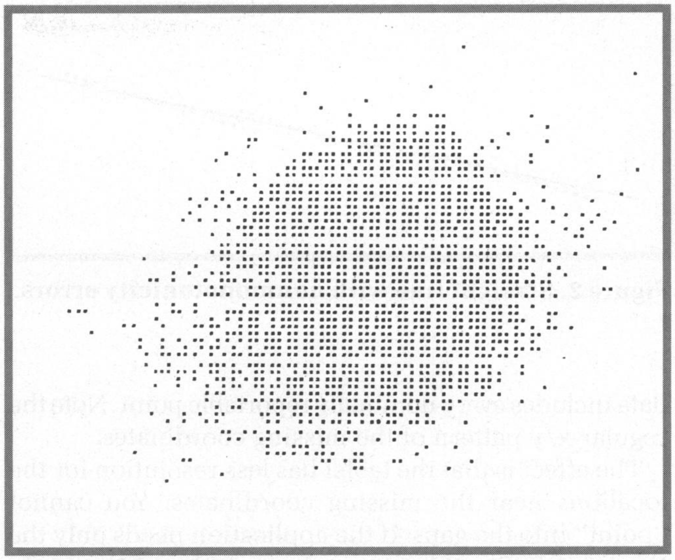


Figure 1. Density map of the coordinates reported by a commercial tablet: actual coordinates reported with "missing" coordinates.

- static error (periodic and nonperiodic)
- hysteresis
- noise and repeatability
- proximity range
- tilt error
- stylus transducer eccentricity
- accuracy at drawing pressure

Each of these deserves a brief discussion.

Missing coordinates

Some tablets that we have examined do not produce every coordinate in their active area. For example, a tablet with an 11″×11″ active area at a resolution of 200 points per inch should produce every coordinate value from zero to 2199 in both the x- and y-axes. A tablet may fail to produce every coordinate because the fine-position interpolation method used does not interpolate far enough between rough-position sensing points. The resulting "under-interpolation" leaves an apparent gap between the reference points.

This does not mean there is an actual gap in the physical positions that can be sensed. The reported coordinates might be uniformly spaced, but the controlling firmware reports only nine distinct points for every 10 physical points on the tablet.

If the application requires fine positioning of a display cross hair, and the pixel range of the display is approximately the same as the coordinate range of the tablet, there may be display pixels corresponding to the missing tablet coordinates that cannot be "pointed to."

Figure 1 shows a "density map" of the coordinates actually reported by one commercial tablet. The input

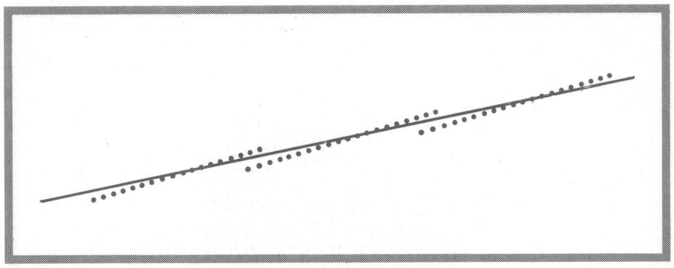


Figure 2. Straight line with nonmonotonicity errors.

data includes every nominally reportable point. Note the regular x/y pattern of the missing coordinates.

The effect is that the tablet has less resolution for the locations near the missing coordinates: You cannot "point" into the gaps. If the application needs only the lower resolution, this may not be a problem. Most applications for screen-oriented positioning fall into this category, since a typical high-resolution display (that is, 1000×1200 pixels) is much grainier than even a low-resolution tablet (200 points per inch on an 11″×11″ area).

If the application emphasizes dynamic entry, such as marking the path an image should move in an animation system, the slight displacement of one point in a set of widely spaced points is not likely to affect the application. Low-pass filtering of the digitizer data could be used to interpolate data into the gaps as the pointer is moved over them. For applications involving fine detail, a more serious problem is the loss of resolution for small features.

Monotonicity

Some tablets will occasionally report a slight jump backward as the pointer is moved across the tablet. In electronic tablets, this is usually caused by overcompensation in the interpolation between sensed positions from the conductor grid. As the pointer moves from a position near one conductor to the next, the interpolation relative to the new conductor results in a coordinate too far away from the new conductor. In resistive sheet tablets this can be caused by nonuniformities in the electrical properties of the materials. When the user must trace small features, the resulting "noise"—which can be seen only when the pointer is moving across the transition—distorts the digitized image.

For the electromagnetic tablets we have examined, these errors are consistent from one tablet to another of the same design with the same firmware. The error cannot be detected without knowledge of the pointer's complete dynamic path. Therefore, the only straightforward correction is to scale the coordinates to reduce the resolution so that it becomes coarser than the magnitude of the errors.

Figure 2 shows the the effects of a simple, periodic nonmonotonicity on a straight diagonal line. The effect, which cannot be corrected, is visible when watching the positions for a moving stylus, because the stylus is not likely actually to jerk discontinuously. However, if the stylus is not moving, there is no indication of which side of the error it is on. The areas where the error occurs are less accurate than the rest of the tablet.

Output continuity

Most digitizing tablets support a "stream" mode of input, where the position of the pointer is reported continuously. The state of the pointer's contact switch (touching/closed or up/open) is included with each set of coordinates, either explicitly or implicitly by not transmitting data when the pointer is "off tablet." We have observed several tablets that send spurious "off" points in the middle of a data stream.

For electronic tablets, one possible source of the problem is the scanning method used to locate the pointer loop. Scanning the entire grid can take longer than the available time between points. For example, the control-

Tasks performed by digitizing tablets

Not a lot has been written about the limits of digitizer performance and how the characteristics affect different applications. A limited amount of material is available describing aspects of digitizer design, mostly tutorial information on functional design (transparent versus opaque, electromagnetic versus electrostatic, etc.), with some discussion of what characteristics are most commonly given in vendor specifications.[1-4] These presentations do not discuss applications in detail.

Foley, Wallace, and Chan give a comprehensive overview of the interaction tasks in graphics applications using digitizers and other pointing devices, but say very little about the devices' performance characteristics.[5] They do list six kinds of graphical interaction tasks for the user: select, position, orient, path, quantify, and text input. They describe the possible methods for each interaction for many kinds of devices.

For digitizer tablets, we find it useful to divide the physical acts required for the interaction tasks into three basic categories: coarse selection, fine positioning, and dynamic graphical entry. Some short definitions follow.

Coarse selection—Commands or icons are selected by pointing and touching the tablet surface to make a selection. The feedback for this process can be either screen- or tablet-oriented, with slightly different requirements for each.

Screen-oriented selection refers to selection of digitized points using feedback from a cursor on the screen. In these applications, the resolution of the screen is often much lower than the resolution of the digitizer.

An example of tablet-oriented selection is "function boxes" on a tablet overlay. The requirements for such a system are minimal, since the selection targets are large, if only to make them readable and easily accessible to the user. Most selections are made close to the center of the function box, so that absolute accuracy usually is not necessary.

Fine positioning—The user must point precisely to a specific point, either relative to the screen, or to a drawing on the tablet.

An example is the manual task of digitizing points from an existing blueprint or engineering drawing to enter the drawing into a computer database. Each point must be accurate, but the data rate (number of digitized points entered per minute) is low, and the digitizer stylus is held stationary when entering a point.

Dynamic graphical entry—The user traces out a (complicated) curved path with the digitizer in real time.

Typical uses of this technique include on-line character recognition, signature verification, and graphical entry for human-generated animation. In these applications, the pen, in general, will not be stationary. To capture the path written in a signature, for example, the digitize rate must be continuous and fast (over 100 points per second). The details to captured are small (some less than 0.05″).

References

1. D.J. Grover, "Graphics Tablets—A Review," *Displays,* July 1979, pp. 83-93.
2. "Digitizer Terminology and Comparability," Science Accessories Corp., Southpark, Conn., 1982.
3. Ivaschentzo et al., "Inductive Transducers for Graphical Input Devices," *Soviet J. of Instrumentation and Control,* Aug. 1969, pp. 22-25.
4. K. Dunn, "Choose Digitizer Technology and Features to Suit Applications," *Computer Technology Review,* Fall/Winter 1981, pp. 171-175.
5. J.D. Foley, V.L. Wallace, and P. Chan, "The Human Factors of Computer Graphics Interaction Techniques," *IEEE CG&A,* Nov. 1984, pp. 13-48.

ler on a tablet with a 10″ square active area and conductors on a 0.1″ grid may need to scan 200 conductors (100 each in x and y) to locate the pointer. If the controller averages five measurements to get good accuracy and must support a digitizing rate of 100 points per second, this represents a scanning rate of 200∗5∗100, or 100 kHz, a very fast rate for the type of circuitry commonly used in low-cost digitizers.

A common method used to reduce the required scanning speed is to scan only the small area of the tablet nearest the last known position of the pointer.[15] A complete scan is necessary only when the pointer is first brought into sensing range. This may cause a delay in reporting the first point after the pointer is in range.

The problem here is that with its restricted scan the tablet may not find the pointer if it has moved rapidly away from the last reported position. When the pointer cannot be found, the tablet electronics cannot distinguish this case from that of the pointer not being on the tablet at all.

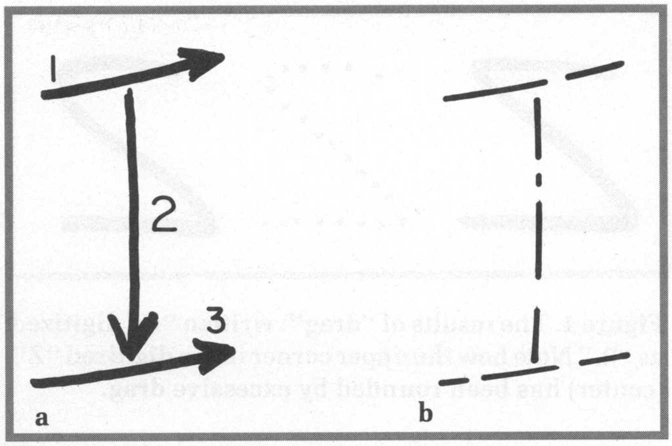


Figure 3. Discontinuities in a resistive sheet tablet: (a) "I" as written, (b) "I" as digitized.

The problem can be reduced, but not eliminated, by extrapolating the next likely position of the pointer based on its travel, not just its previous position. This change must be made in the tablet design itself. The application can reduce the error somewhat by filtering out short "pointer-up" periods from the tablet's data stream. This helps only if the pointer does not continue to move too fast for the tablet to catch up.

In one design for a resistive sheet tablet, a sheet of conductive material is separated from a resistive sheet by small spacing "bumps" at regular intervals.[16] The tablet is effectively "pressure sensitive," since it reads the position where the two sheets get pressed together. When the stylus tip crosses a spacing bump, the two sheets lose contact with each other, producing a spurious "pen-up" point.

In our application (dynamic character recognition), the effect of these discontinuities is to make quickly written strokes in a character look like a series of shorter strokes. Since the number of strokes in a character is one feature used in the recognition process, the character is made "unrecognizable," even though a plotted image of the data might still be legible to a human reader (see Figure 3).

Some applications use a digitizer to control the position of a mechanical object, such as a robotic arm, or to "drag" a graphical object, such as an image in a display. If the application uses the stylus switch to "drop" the dragged object at a final position, the discontinuities can cause the object to be dropped partway along the intended path.

Slew rate

We define slew rate as the maximum travel rate of the stylus that the tablet can report with a specified positional accuracy. Slew rate accuracy can be limited by low-pass filtering, time delays in measuring the x and y ordinates, and wild data.

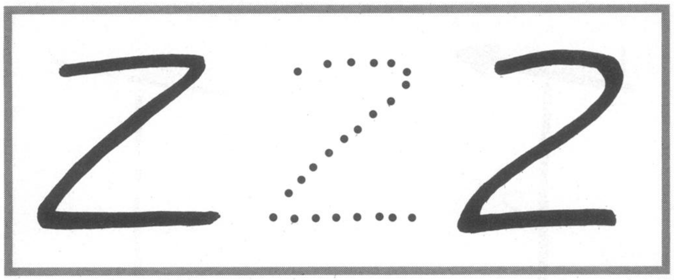


Figure 4. The results of "drag": written "Z" digitized as "2." Note how the upper corner in the digitized "Z" (center) has been rounded by excessive drag.

If the tablet firmware averages several measurements to get good accuracy, it will send a series of interpolated points that "drag behind" the motion of a quickly moving pointer. The result is that rapidly drawn, angular features in dynamic graphical entry are rounded out, and small, quickly drawn loops may be collapsed into single points. For our character recognition application, the results can change the appearance of one character shape to another, as shown in Figure 4. This phenomenon would also affect applications such as signature capture and artistic drawing.

Some digitizer designs perform separate cycles of processing for each coordinate pair: first a measurement in x, then in y. If the time for each cycle is significant, the apparent position will "bow" in one ordinate direction if the stylus is moved at varying speeds along a diagonal line.[17]

If the time between the x and y measurements is known, the error can be substantially corrected by geometric projection, such as the algorithm given by Carau.[18]

Rectilinear displacement

Rectilinear displacement is the difference between the nominal coordinates of a baseline home position on the tablet, and the actual coordinates transmitted for that location.

Digitizer applications in which the user inputs exact points from a fixed menu (possibly permanently mounted on the tablet), or from photographs, may need the precise physical location of the data, not just the relative positions of the different input locations. Some tablet designs have a raised border or other hardware guide for positioning a sheet of paper or a photograph on the tablet.

A fixed offset in the input coordinates for all tablets of one model can be seen as a discrepancy in the manufacturing of the tablet. The error should be fixed in the tablet firmware, since it is the same on all tablets. A varying displacement error among tablets of the same design occurs if the mounting of the sensing grid in the tablet "skin" is not machined to be precise. Imprecise mounting can also lead to errors in orthogonality (see below).

The varying displacement error can be corrected by computing the displacements in x and in y from the baseline value for a reference point on the image and adding these to offset the coordinates of every input point. Some manufacturers include this one-point calibration procedure in the firmware for their tablets.[19] The procedure can just as easily be added to the applications code by having the user point to a specified reference point whose correct position is known.

Any application where the user must digitally "transcribe" an image requires accurate absolute physical

Who wrote the specifications?

We have found that many vendors play a game of "specs-manship" when quoting performance on their digitizers. We give several examples:

- One vendor quoted an 11″×11″ active area, but quoted accuracy only for a region bounded one inch from the edges of the active area, or 9″×9″. Testing showed large sloping errors near the right and bottom edges of the 11″×11″ active area.
- We were shipped one tablet that, according to the vendor, had a resolution of 0.001″ (1000 points per inch) and an accuracy of 0.01″. The tablet turned out to be the vendor's standard 200-points-per-inch tablet with the firmware modified to multiply all coordinates by five to simulate 1000 points per inch. The discrete values of 0.005″, 0.01″, 0.015″, 0.02″, etc., were within the "accuracy" specification of 0.01″.
- All vendors that we have checked offering a choice of a pen stylus or a puck quote accuracy only for the puck. A puck on an electromagnetic tablet is inherently more accurate, since a puck generally has a larger sensing/transmitting loop, is always perfectly horizontal at a constant height above the tablet surface, and is usually held stationary.
- Some vendors quote accuracy relative to an absolute physical point on the tablet, while others quote it relative to the "first" reference point digitized, covering up any offset errors.
- The quoted data rate for one tablet was "up to 100 points/second," but the tablet used an ASCII format that took up to 14 characters per point, on a 9600-baud line. For most values (more than two digits for *x* and for *y*), the extra characters limited the maximum possible reported data rate to about 63 points per second.

All vendors want to show the best side of their product. The figures given in each case are correct and true. The point is that different characteristics are critical depending on the application of the digitizer. For example, many digitizers are sold based on one high-performance characteristic—resolution—but for many applications resolution is not critical.

positioning. Common cases are the digitization of existing engineering drawings that predate the direct use of CAD in an organization, and medical analysis of X-ray images.

Scaling error

For large displacements, scaling error is the ratio between the measured physical distance between two points parallel to either the x- or the y-axis, and the actual distance. The most likely cause in electronic tablets is an improper or inaccurate scaling operation on the sensed position to a "normalized" scale (such as converting 0.1" grid spacing to metric units). In resistive sheet tablets, the most likely cause is variations in component properties in the analog electronics.

The error can be introduced in the application itself, as when the user is digitizing an image that itself has scaling errors. Images from office copiers, for example, typically have up to ±3 percent scaling distortion separately in x and in y. This distortion is enough to shift text the height of an entire text line. Scaling error on photographs, too, caused severe problems when published curves for a control system were transcribed from photocopies.[20]

If the error is simple, it can be corrected by adding an "aspect ratio calibration" to the application.[21]

Orthogonality

We consider two separate characteristics for orthogonality in a digitizer: the relative angle in physical space between the digitized x- and y-axes, and the absolute angle to the physical baseline of the tablet. Some digitizers have a raised edge or frame to make sure that the paper on the tablet is laid down square to prevent misalignment.

For electronic digitizers using an x/y grid of conductors, an absolute error in the angle of an axis is probably a result of improper or imprecise mounting of the x/y grid relative to the tablet skins. At least one commercial tablet has this sort of design.[19]

An error in the relative angle of the two axes is a result of mechanical distortion in the x/y grid. The same apparent phenomenon results when the tablet is nominally "precise," but the image being digitized is itself subject to orthogonality errors. We have observed errors of as much as 3 degrees in nominally "square" alignment on a printed form produced with commercial offset presses or office copiers.

Absolute error in one axis can be measured using two reference points along the "physical" axis. Measuring relative error requires at least three-point calibration,[22] where one point is the vertex of a fixed angle formed by the three points. The trigonometric calculations for the correction have been described.[23] To reduce error in the measured angle from small changes in the placement of one or more of the points, all other positioning errors must be corrected first, and the reference points should be spaced widely apart in the active area of the tablet.

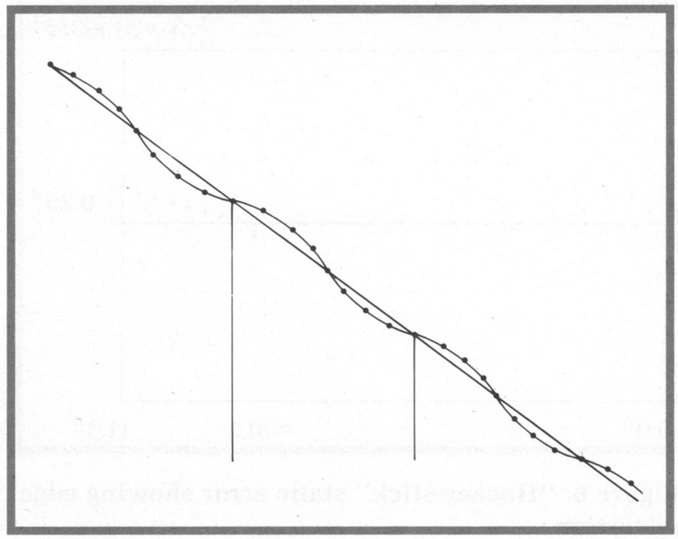


Figure 5. Nonrandom differential error reflects underlying tablet structure. The space between the two vertical lines reflects the spacing of the tablet sensing grid.

Differential error

The positioning error for a tablet can also be measured differentially; that is, the maximum error for a small relative change in position is measured, regardless of the absolute error. An example would be a small-scale sawtooth error in one axis as the pointer is moved slowly along the axis (see Figure 5).

An application requiring the correct "imaging" of small features may be able to tolerate a large cumulative error across the total area of the tablet, but may not be able to tolerate "noisy" data around the area of the pointer tip. For example, a handwritten signature may still be acceptable with a large but well-behaved distortion to the image. The features that make the signature uniquely recognizable are the small hooks and corners of each individual character.

Differential errors can be caused by transition between grid lines on an electromagnetic tablet, local nonlinearities in the material of a resistive sheet tablet, or threshold crossings in the A/D measurement circuitry. Visually, the effects of this type of error may be difficult to distinguish from random noise. However, true Gaussian noise can be compensated for by trading performance in temporal sampling rate for spatial accuracy. Fixed differential errors cannot be compensated for in this way.

Differential error can be an especially pernicious form of static error in the tablet. Confusing its symptoms with noise is easy, but differential error is much harder to correct.

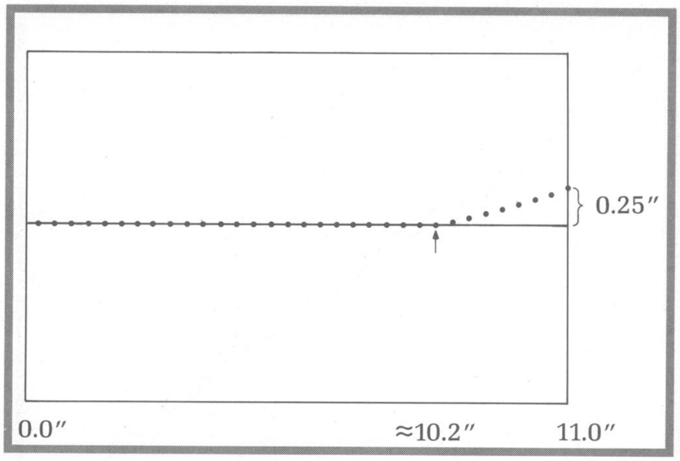


Figure 6. "Hockey-stick" static error showing edge distortion.

Static error

Static error is the fixed error remaining in the reported tablet coordinates after all corrections for scaling, rectilinear displacement, and other errors have been made. If the errors are identical for all tablets of a given model, we say the error is predictable, regardless of the nature or the source of the error. If the errors are not identical for different tablets of the same model, we say the errors are unpredictable. With many applications, the predictable errors can be compensated for.

Different mechanisms can produce a spatially periodic error, or a well-behaved but nonperiodic error. Some tablets exhibit a geometrically regular but varying position error as the pointer loop is moved across the tablet. The most frequent cause is that the interpolation used to compute the position between grid conductors does not correctly model the change in the sensed pulse as the pointer loop is at different distances from the grid conductors.

A nonperiodic error is typical near the boundaries of the active area of a digitizer. It may be caused by the proximity of the pointer to electromagnetic fields from the tablet electronics, poor algorithms for the transition from interpolation to extrapolation, or merely the lack of a suitable "infinite" plane for the tablet grid. Figure 6 shows the actual error we measured for one commercial tablet with an 11″×11″ active area.

Most resistive sheet digitizer designs are very sensitive to nonuniformities in the resistive sheet. Here the errors in *x* and *y* tend to be interrelated in different spots on the tablet. A region that is electrically thicker or thinner than the rest of the sheet affects the resistance paths to the edges for the entire sheet, not just in one direction.

The most general way to fix static, continuous errors is to measure the actual position versus the reported position for every point on the tablet, put the results in a correction matrix, and look up the correcting offset every time a position is read from the tablet. In practice, the size of the correcting matrix has to be limited to avoid taking up too much memory. At least one vendor uses this technique on a resistive sheet tablet, where each tablet has to be "calibrated" individually to correct for the variations in resistance in the sheet.

Note that piecewise continuous factors should be used: You have to interpolate between the points in the correction matrix, not just add or subtract and offset in each small correction patch. One vendor did not interpolate, and the result was discontinuities at the edges of the corrections patches.

Hysteresis

A broad class of characteristics can be described as "memory errors": The reported value for the position of the pointer depends on the spatial and temporal path taken to put it there. The most well-known property in this class is hysteresis.

Many digitizers have an explicit hysteresis incorporated into their design to give the false impression of high stability and lack of noise. If a given position is approached from the left, the coordinates reported are different from those reported if the position is approached from the right. No change is reported until the pointer has been moved by some minimum amount or some minimum time has passed. The rationale is to filter out wiggling and jumping noise when the pointer is actually stationary.

The apparent lack of noise is a false sense of quality: The size of the hysteresis window is merely a large upper bound on the uncertainty in the pointer's position. One detrimental effect of simple hysteresis algorithms is that they make more sophisticated filtering methods impossible later on in processing (for example, the application of low-pass filtering to data with inherent Gaussian noise).

Noise and repeatability

All electronic systems have some inherent random processes in them. Minimally, there is the thermal electron noise in an analog circuit along with drifts in component characteristics caused by internal changes in temperature. External sources such as computer monitors, electric motors, or another nearby tablet also contribute.

All these factors may produce a time variation in the output of a digitizing tablet. If the changes come quickly, occurring on a sample-to-sample basis in the data, they are called *noise*. If they occur slowly, over the course of several seconds or minutes, they are called *drift*.

For rapid noise, various averaging or low-pass methods can improve the results for most applications. Drift cannot be corrected so easily. The cause must be corrected directly, for example, by holding the tablet at constant temperature or regulating the power supply voltage in the face of environmental changes. The only other alternative is to model the physical causes and effects for the drift, measure the environmental factor (for example, temperature or air pressure), and correct for the effects algorithmically. The procedure is generally impractical unless the measurements and correction are designed into the tablet electronics.

The simplest solution for rapid noise that is not too big is to average several data points together; the averaging process acts as a low-pass filter on the noise. This can be done without reducing the number of points you get per second: Simply keep the last n "real" points in a ring buffer at all times, and average each point with the preceding $n-1$ points to get the improved point.

Please note that the presence of digitization noise can help in some applications. Truly Gaussian noise can be passed through a low-pass filter to interpolate position accurately between the LSBs of the input data.[24]

Proximity range

Pressure-sensitive tablets by definition sense position only when the pointer is in contact. Electromagnetic tablets generally sense the position of the pointer when it is near the tablet. A separate mechanical pointer switch is used to detect when the pointer is in contact.

Although some vendors quote a "proximity range,"[25] no vendor we know of measures accuracy as a function of proximity range. Usually the accuracy of the sensed position decreases as the signal weakens with distance from the grid. A few electromagnetic tablets report proximity height as position data.[26]

Proximity sensing is useful for "airbrush" applications, where it provides a rough value for the height of the pen.

Tilt error

Many tablet technologies were originally designed for a puck pointing element. For a stylus pointing element, the original design was minimally modified for the different electrical characteristics of the pointer coil. The performance suffers from the change for the following reasons.

The most dramatic difference in the designs is that the puck contains a large coil held at a fixed height above the digitizer surface and in a rigid coplanar orientation with the tablet. A pen stylus can be tilted and is unlikely to have the coplanar orientation. Further, for our application of writing input, the user may be placing a pad of paper on the tablet, and changing the height of the stylus separately from the tilt.

The sensed position can change substantially due to the parallax between the actual position of the stylus tip and the position of the coil mounted partway up the stylus body.[27] Some digitizer technologies are also sensitive to the change in the sensing field when the coil is at an angle to the tablet (see Figure 7). At least one vendor "solved" this problem by stating that the stylus must be kept vertical.[28]

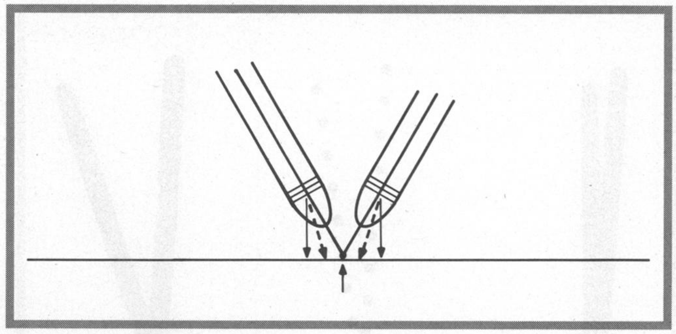


Figure 7. Parallax change for sensing coil in pen stylus. The physical position of the tip is in the center. The solid lines with arrows show the actual position of the coil over the tablet; dotted lines show nonlinear displacement of reported position due to magnetic field effects.

One tablet we examined produced a 0.1″ (2.5 mm) change in position for a 45-degree change in the angle of the stylus. In normal pen usage, the tilt magnitude and orientation both vary. The user cannot compensate for tilt errors when moving the stylus since there is no output manifestation that can be observed.

The "cone of usable stylus angle" is a measure of tilt sensitivity that is infrequently included in vendors' specifications. This is the maximum angle of tilt that the manufacturer believes will still result in usable performance. If specified, it may be quoted separately from "overall accuracy," without a statement of the relationship between the two.[29]

Many technologies show a nonlinear relationship between stylus tilt and parallax error. For example, in one tablet design we examined[30] the error in position for a 20-degree tilt was 0.01″ (0.25 mm), but rapidly became 0.08″ (2.0 mm) between 20 and 40 degrees of tilt.

An example of the detrimental effects of tilt error is shown in Figure 8. In handwriting there is a tendency to retrace vertical and horizontal marks.[24] Since the user is unlikely to maintain the same tilt angle while writing, the position on the retrace will appear to be different, possibly changing the digitized image enough to make it be recognized as a different character.

Tilt error cannot be corrected unless the digitizer accurately reports the angle of the pen in x and y with each point. If the orientation angles are known, and the error is a simple continuous function of the angles, you can make up a correction table for the error and add the

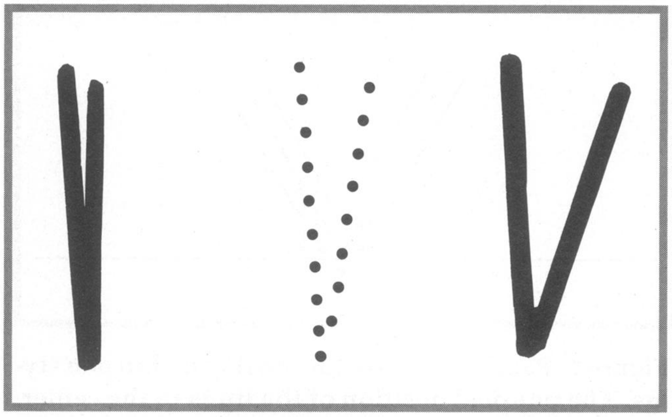


Figure 8. Tilt error: Retraced "1" is digitized (center) as written "V" (right) would be digitized.

appropriate offset to each point. Most tablets do not report the orientation of the stylus.

Stylus transducer eccentricity

Many pointers for digitizers are designed to appear rotationally symmetric. A pen stylus, for example, usually has a rounded housing. In any case, all digitizer manufacturers imply that the pointer can be used in any rotation around the sensing point.

Nevertheless, the internal contents of the pointer, particularly pen stylus pointers, can make the electrical or acoustic fields used for sensing position nonsymmetric. The most common components causing this are secondary switches and buttons mounted to the side of the sensing coil. The rotation of the pointer as the user moves his or her hand changes the apparent position of the sensing point.

If nonsymmetric sensing is a problem in an application, the only solution is to hold the stylus the same way all the time.

Accuracy at drawing pressure

For certain digitizer technologies, such as resistive sheet digitizers, there is also a potential problem with light drawing pressure, typical of "electronic ink" use in drawing and signature capture. The data is unreliable when the stylus is only lightly in contact (less than 50 grams pressure) with the digitizer surface. This light pressure is typical of the start and end of written strokes, even if the user consciously maintains a heavier marking pressure.

There are two typical causes of this error. First, if the design involves mechanically spaced layers of conductive and resistive material, the pressure required to force contact at locations directly over the spacing points may be much more than on the rest of the surface. The effect shows up as locations that give occasional "gaps" in the digitizer data (the layers were not in contact when the spacer was passed over), or that give varying offsets of the reported location (the effective point of contact was

Pencept's digitizer development

One question the reviewers for this article asked is how Pencept corrected the performance problems in its application.

Pencept's main business is dynamic character recognition, not digitizers. We did not start out building any *x*/*y* digitizer tablets ourselves—we used commercial off-the-shelf tablets, the usual course of action for manufacturers of character recognition products. Dynamic character recognition is a very demanding application for digitizer performance. In the past five years, we have tried out many different tablets, and have used modified versions of two different vendors' tablets in volume in our products.

For both the off-the-shelf tablets, we had to make several correcting design changes. An immediate change was a new stylus pen. A "light touch" pen with no detectable travel in the pen-tip switch is necessary for handwriting. Our stylus has a maximum pressure-to-close of 40 grams, and a maximum travel-to-close of 0.8 mm, using a special switch and barrel mechanism design. The pen has been retrofitted to several vendors' tablets for other applications.

Both tablets used in our products had several deficiencies that we fixed using techniques described in this article. To reduce random noise in one of the tablet designs, we applied two stages of low-pass filtering to all incoming *x*/*y* data, with special code to "anchor" the first and last points of a stroke so they would not be shifted by being averaged with other points.

This tended to interpolate points and raise the apparent data rate variably. To avoid getting excess data, we added a resampling step in the middle to prevent points from "piling up" on top of each other. All this processing was added purely by instinct, and was our first lesson in correcting tablet performance.

The tablet circuitry turned out to have a one-shot latching circuit between the pen-tip switch and the rest of the controller electronics—this was to correct for switch bounce when the pen tip was first put down or lifted off the tablet surface. The result was to delay the effect of the pen-tip switch signal for about 25 milliseconds.

With our more accurate pen design, this "correction" added noticeable hooks to the ends of every stroke in a character. In software, we arbitrarily removed the last two points from every stroke, but this was only partially successful: The pen switch bounce effect the vendor had tried to correct was intermittent.

The particular tablet design used magnetostrictive wires strung in a set of notches inside the tablet skin.

on one side of the spacer).

A second effect has to do with contact resistance. At some intermediate pressure between contact and no contact, there is enough additional resistance so that the conductive sheet is not brought fully to the voltage. For example, one digitizer using a layer of indium tin oxide and an induced voltage from the stylus gave wildly erratic errors of as much as 1.2 cm toward the (0, 0) corner on the digitizer when marking was done with the light contact pressure typical of writing motion.

Pen stylus mechanical properties

Many current applications use a puck pointer rather than a pen stylus. Pucks usually give more accurate position data, but several applications, such as artistic drawing or signature capture, work much better with a stylus.

Applications needing a pen stylus impose special requirements on the mechanical design of the pen. We discuss them in the following sections. While these performance problems can exist in puck pointers, they are more pronounced with a pen stylus, and a pen stylus is more likely to be used in exactly those applications where they are more critical.

Pen switch mechanical latency

Stylus designs that use a pressure-sensitive switch to detect when the user is writing place tight requirements on the time latency of the opening and closing actions of the switch. A switch that is slow to close will miss the initial segment of a drawn line. A switch that is slightly sticky will add some amount of proximity data to the end of each stroke.

A spring-loaded mechanical switch, especially if the switch travel distance is very small, can work like a pressure-sensitive switch. We have also found digitizers whose design intentionally simulates a "sticky" pen switch in firmware to cover up bounce in the actual mechanical switch. There has been some work on algorithmic means to correct for these problems.[31]

For small features, the missing and extra data points can substantially change the image of what the user has drawn. In our application of handwriting, for example, a small "hook" added to a character by the switch opening just one or two data points late can change one character into the image of another (see Figure 9).

One fix is always to ignore the last *n* points before the stylus pen switch opened. This requires that the application buffer all the data from the tablet so it can tell which previous points to ignore when the tablet sends a "pen-up" signal. How well this fix works depends on whether the latency time is always exactly the same, or varies with how hard the stylus was pushed down, friction of the stylus ink cartridge against the inside of the barrel, etc.

These tended to have a *mechanical* failure mode where the wires would stretch and slip out of the notches. This problem was not correctable.

We gradually figured out what these problems were over the first year of shipping our product. At first we had thought the problem lay in inadequacies in our application code. We then spent about six months looking for a different tablet. The one we settled on had problems of its own, which also took us several months of field experience to figure out.

As a result of what we had figured out while evaluating new vendors' products, we had the vendor of the second tablet that we used modify the tablet firmware to measure *x* and *y* position alternately several times before averaging them into a final reading, rather than measuring *x* several times before measuring *y*.

This change was in the vendor's firmware for the tablet controller. Without the change, the tablet had slew errors during rapid writing. We also modified portions of the analog circuitry on incoming tablets to add filter capacitors and remove some jitter due to electronic disturbance.

The tablet had a fixed "hockey-stick" error at the right and bottom edges. We made measurements on several of the tablets from different shipments using a laboratory *x*/*y* positioning table accurate to 0.0005". After we determined that the errors were static, continuous, and exactly the same on all the tablets, we computed a piecewise continuous correction matrix for the tablets and added it to our applications code. In later shipments the vendor introduced corrections of its own without telling us, and, with the double correction, we ended up with the same kind of error in the opposite direction. We got the vendor to agree to ship us only the older firmware.

The newer tablet also needed low-pass filtering of the data, but with the parameters adjusted experimentally to match its different characteristics.

Very recently we shifted to a tablet of our own design. We were able to get several patents on some of the tricks that we used to correct the problems mentioned in this article. Our tablet was designed specifically for handwriting capture, and is not, in the current product line, available in the wide range of sizes available from other digitizer vendors.

Since we designed the internal workings of the tablet, we have had much more flexibility than when we had to try to correct some of the problems in the applications software. Most of the techniques mentioned in this article were used in the design.

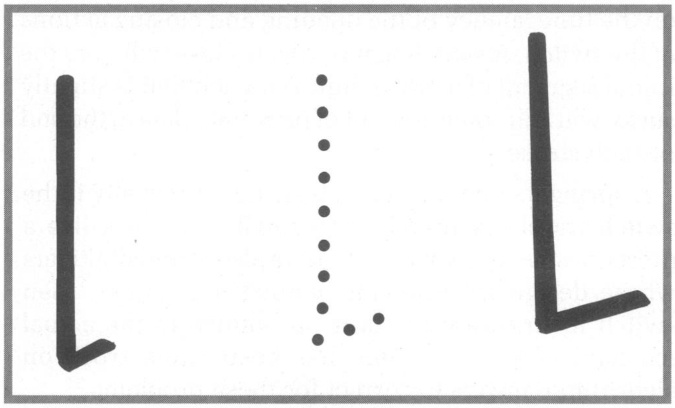


Figure 9. Pen switch mechanical latency: Written "1" is digitized (center) as written "L" (right) would be digitized.

Figure 10. Pen switch pressure sensitivity: Gaps in data due to light writing pressure change a written "B" into "13."

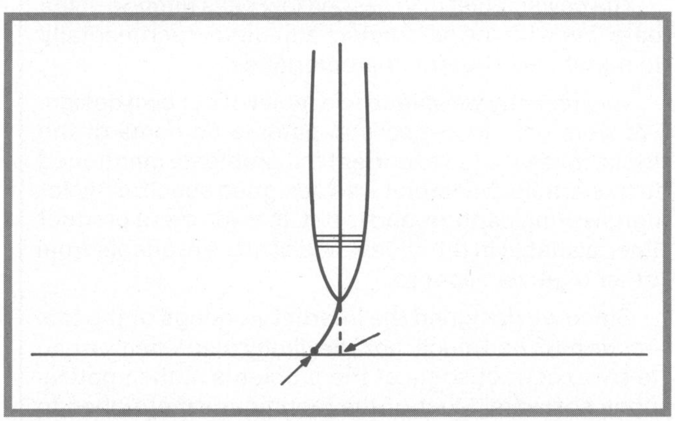


Figure 11. Stylus mechanical slop.

Pen switch pressure sensitivity

Most pen stylus switches can be modeled as pressure sensitive. If the pressure required to close the switch is too great, a drawn line may have irregular gaps where the switch opened during the line. In any application where the user must track fluid movements with the pen, such as signature capture or animation, much of the drawing will be done with very light stylus pressure (25 to 50 grams at the pen tip). These gaps cannot be reliably distinguished algorithmically from intentional liftings of the pen. We give one example of possible effects on handwritten input in Figure 10.

Conversely, if the pressure threshold of the switch is too low, the switch may stick "on," perhaps indefinitely, because the restoring force of the switch spring is lower than the force needed to overcome internal friction in the mechanical parts of the stylus and its housing.

Pen switch travel distance

Another characteristic, related to the stylus pressure threshold, is the travel distance required by an inking or noninking cartridge in the stylus to open and close the switch. Many inexpensive small switches used in digitizer styli have a relatively large travel distance, up to 0.1" (2.5 mm) or more. Some of these styli can leave ink on paper, even for a complete figure or written signature, without pressing the cartridge far enough to close the switch.

Some switch designs have very long travel distances and can be used only for "selection," not drawing. For "drawing," many applications require the user to draw using proximity data only and press the pen down once for a "click" at the start and end of each line. We have found that the large travel distance is a very unnatural characteristic, one that is difficult for a user to adjust to when writing or drawing.

Hand/environmental artifacts

Since the user's fingers enclose a pen stylus and the coil loop in the stylus is generally smaller than the coil in a puck pointer, rings and bracelets have an influence on the electromagnetic fields used in most stylus digitizer technologies. This is particularly true if they are made with conductive metals such as gold and copper, or ferromagnetic materials such as iron.

We have also found that much "stainless steel" used in ink cartridges has strong magnetic properties that can vary unpredictably. Substantial changes in the tilt and proximity behavior of a pen stylus can result because of variations in the steel used in nominally identical ink cartridges from the same manufacturer.

Stylus mechanical slop

To prevent binding and excessive friction on a stylus cartridge in its housing, there must be some mechanical tolerance in the two structures. For some styli, the mechanism can bend enough to put the coil in a distorted position relative to the pen tip. We have observed "slop" in styli such that the coil position relative to the stylus tip can be changed by more than 0.01" (0.25 mm), depending on how the stylus is held.

Figure 11 illustrates a movement of 0.04" we have measured for commercial digitizer styli coils. The movement results from slight side pressure on the housing that does not move the tip of the stylus.

Obscured visual tip

There is a direct connection between the diameter of the coil loop in the pointer and the base accuracy of the digitizer. For this reason, many styli are designed with a nontapered or flared front end to hold a larger coil. The front blocks the user's view of the stylus tip and what he or she is drawing.

Conclusion

Many applications require superior digitizer performance; examples are signature capture, handwriting input, and precise drawing. Even low-tech applications often require some particular aspect of digitizer performance that may not be met by most commercially available pointing devices. The need for low cost and high performance in a tablet digitizer was the driving force behind our design for a tablet digitizer.

However, there is no accepted standard, or set of standards, for tablet performance. Many "high-performance" digitizers do very well in an area that is not important in a particular application—especially "raw" physical resolution—but do very poorly in an area that counts, such as accuracy. Some digitizers seriously degrade performance in one area, such as slew rate accuracy, to make other more "obvious" problems, such as jitter, go away. Most vendors, for understandable reasons, quote the performance characteristics that make their products look good, rather than those that are actually important for the application—and they certainly do not quote the characteristics that show performance limitations.

On the other hand, many apparent problems can be handled easily in the application software, using techniques described in this article. Most digitizers work well for many applications.

The importance of understanding and testing for the correct kinds of digitizer characteristics cannot be stressed enough. We must understand the weaknesses that make a difference and those that do not before we can intelligently design a useful, realistic human/computer interface that uses a pointing device. ■

Acknowledgments

Special thanks must be given to the following people for their help in preparing this manuscript: Barry Blesser, Andy Nilssen, Judy O'Brien, and Patrick Philbin. In addition, Tom Prentice helped in carrying out the engineering studies of actual tablet performance.

References

1. J.J. Romein, US Patent 4,246,439, "Acoustic Writing Combination, Comprising a Stylus with an Associated Writing Tablet," Jan. 20, 1981, assigned to US Philips Corp., New York.
2. Pieter de Bruyne, "Compact Large-Area Graphic Digitizer for Personal Computers," *IEEE CG&A,* Dec. 1986, pp. 49-53.
3. A.M. Hlady, US Patent 3,916,099, "Touch Sensitive Position Encoder Using a Layered Sheet," Oct. 28, 1975, assigned to Canadian Patent and Development Ltd., Ottawa, Canada.
4. "DigiBit Product Description," Numonics, Inc., Lansdale, Pa., 1982.
5. "GeoGraphics Drafting Board Digitizer," GeoGraphics, Inc., Champaign, Ill., 1986.
6. W. Pepper, US Patent 4,071,691, "Human-Machine Interface Apparatus," Jan. 31, 1978, assigned to Peptek, Inc., Bethesda, Md.
7. B. Blesser, US Patent 4,375,081, "Multistage Digital Filtering Utilizing Several Criteria," Feb. 22, 1983, assigned to Pencept, Inc., Waltham, Mass.
8. R.T. Rocheleau, US Patent 4,242,843, "Coarse Position Digitizer," Jan. 6, 1981, assigned to Summagraphics Corp., Fairfield, Conn.
9. R.W. Prugh and B.J. Fadden, US Patent 4,206,314, "Graphic Digitizer," June 3, 1980, assigned to GTCO Corp., Rockville, Md.
10. G. Hurst and W. Colwell, US Patent 3,911,215, "Discriminating Contact Sensor," Oct. 7, 1975, assigned to Elographics, Inc., Oak Ridge, Tenn.
11. L.J. Lukis and G.P. Duhig, US Patent 4,493,104, "Character Recognition Device," Jan. 8, 1985, assigned to Moore Business Forms, Inc., Grand Island, N.Y.
12. "SPC-Series Technical Report," Scriptel Corp., Columbus, Ohio, 1986.
13. "GP-8-3D Grafbar Technical Bulletin," Science Accessories Corp., Southpark, Conn., 1984.
14. L.E. Kinsler and A.R. Frey, *Fundamentals of Acoustics,* 2nd ed., John Wiley, New York, 1962, p. 503.

15. P.C. Carvey, US Patent 3,975,592, "Electrographic System," Aug. 17, 1976, assigned to Applicon, Inc., Burlington, Mass.

16. W. Gibson and J. Talmage, US Patent 4,220,815, "Nonplanar Transparent Electrographic Sensor," Sept. 2, 1980, assigned to Elographics, Inc., Oak Ridge, Tenn.

17. F. Carau, H. Hetzel, and M. Tremblay, US Patent 4,255,617, "Travelling Wave Digitizer," assigned to Hewlett-Packard Co., Palo Alto, Calif.

18. F.P. Carau, "Easy-to-Use, High-Resolution Digitizer Increases Operator Efficiency," *Hewlett-Packard Journal*, Dec. 1978, pp. 2-13.

19. "CalComp 2500 Series Digitizer Operator's Manual," CalComp, Inc., Anaheim, Calif., 1985.

20. Cris Whetton, quoted by W. Richards Adrion, "A Glitch in Spline Shaved Line," *ACM Software Engineering Notes*, Oct. 1985, pp. 9-10.

21. "AutoCAD 2 Drafting Package User Guide," No. 106-006, Autodesk, Inc., Sausalito, Calif., 1984.

22. "Pencept Penpad Manual," Pencept, Inc., Waltham, Mass., 1984.

23. J.D. Foley and A. van Dam, *Fundamentals of Interactive Computer Graphics*, Addison-Wesley, Reading, Mass., 1982.

24. B. Blesser and J. Ward, "Human Factors Affecting the Problem of Machine Recognition of Hand-Printed Text," *Proc. Seventh Ann. Conf. and Exposition NCGA*, NCGA, Fairfax, Va., 1986, pp. 498-514.

25. "Summagrid User's Guide," Summagraphics Corp., Fairfield, Conn., 1982, p. A-3.

26. "3-Space Digitizer Technical Bulletin," Polhemus Navigation Sciences Div. of McDonnell Douglas Electronics Co., Colchester, Vt., 1984,

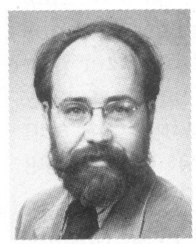
27. J.T. Iannou, US Patent 3,873,770, "Digital Position Measurement System with Stylus Tilt Error Compensation," Mar. 25, 1975, assigned to Bendix Corp., Southfield, Mich.

28. "Calcomp 2000 Series Digitizer Operator's Manual," CalComp, Inc., Anaheim, Calif., 1983, p. 15.

29. "Digi-Pad 5 Family Data Sheet," GTCO Corp., Rockville, Md., 1982.

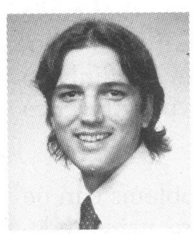
30. "Numonics #2200 Pad with Low Profile and High Resolution," Numonics, Inc., Lansdale, Pa., 1985.

31. J. Ward, US Patent 4,534,060, "Method and Apparatus for Removing Noise at the Ends of a Stroke," Aug. 6, 1985, assigned to Pencept, Inc., Waltham, Mass.



Jean Renard Ward has been with Pencept since its founding and leads the development team for the design of the company's character recognition products. His primary activities currently focus on the development of and research into handwriting and pattern recognition techniques and applications. Prior to Pencept's founding, Ward was involved in interactive compiler design and real-time hardware and software design for graphics and digital signal processing systems.

Ward received his SBEE degree from the Massachusetts Institute of Technology and has studied at the Philipps-Universitaet in Marburg, West Germany.



Mark J. Phillips joined Pencept in 1983. His responsibilities include software system design and project management and development. He has developed a language for character recognition and designed front-end user interfaces, based on artificial intelligence, for CAD and graphics packages. Most recently he has done algorithmic design and performance verification for advanced digitizing systems.

Before coming to Pencept, Phillips was hardware engineer at Computervision. In 1980 he received a BS in electrical engineering and computer science from MIT.

The authors can be reached at Pencept, Inc., 39 Green St., Waltham, Mass., 02154.