Sloppy Selection: Providing an Accurate Interpretation of Imprecise Selection Gestures

Edward Lank[1], Eric Saund[2], and Luping May[3]

[1]Computer Science Department, San Francisco State University, 1600 Holloway Ave., San Francisco, CA, 94132
[2]Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA, 94304
[3]Computer Science Department, Stanford University, 353 Serra Mall, Stanford, CA, 94305
lank@cs.sfsu.edu, saund@parc.com

Abstract

This paper describes on-going work in the analysis of motion dynamics in pen-based interaction. The overall goal is the creation of a model of user motion in pen gestures where constraint and curvature vary over the length of the path. In particular, speed/curvature models of motion are used to analyze pen trajectories and infer target constraints obeyed by a user performing selection gestures. We aim to use this information to calculate an effective local spatial selection tolerance associated with each gesture. This can be used to perform selection according to user intent instead of their literal stroke. Here, we describe our early analysis of constrained user selection gestures, and outline a prototype application that infers a tolerance for one type of selection gesture. The application selectively splits pen strokes based on an analysis of user motion.

Introduction

This paper describes on-going work in the analysis of motion dynamics of pen-based interaction. The particular problem at hand is the analysis of selection gestures in pen computing. Pen-based selection strategies include two common selection options, tap-to-select, and encircling. We focus on the latter, that is, selection by drawing a freehand closed shape around a target object.

Different applications treat selection by encircling according to domain-specific or application-specific criteria. Most paint programs, for example, view selection gestures as definitive and cut image material precisely on the gesture's path. Advanced selection techniques can adjust a selection stroke to fit the borders of salient visual objects.

In pen-based note taking applications, selection gestures are typically interpreted in light of underlying pen strokes. Digital ink strokes themselves, and sometimes groups of ink strokes forming words, are viewed as immutable objects, and the selection gesture selects among strokes or words. A problem faced by these programs is, which objects should become selected when the selection gesture in fact intersects immutable objects?

In our work, we seek to interpret selection strokes based on inference of user intention. We hypothesize that significant and useful aspects of intent can be estimated from measurable characteristics of the gesture.

This work is currently in its early stages. The purpose of this paper is to present initial work in selection gesture analysis under varying curvature and target constraints. In addition, we outline a prototype proof-of-concept application which uses this analysis to make an intelligent judgment when executing one type of selection gesture.

Problem Formulation

Reliable assertions about user intention with respect to their actions would allow designers of pen-based interfaces to take into account users' likely goals when choosing among program responses. Consider, for example, the user gesture in Figure 1. Here, the user has drawn a circle around a line of text. Note that at the extreme endpoints of the line the user has "clipped" a number of letters from the gesture.

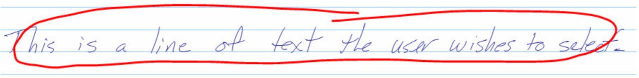

Figure 1: A user selection gesture with inaccuracies.

Given the spatial and semantic coherence of the sentence itself, a reasonable assumption can be made that the user intended to select the whole. Under some circumstances, however, a user may in fact wish to lop off one or more characters, or otherwise select a less salient set of markings. We suggest that in these cases users will gesture more carefully, deliberately, and hence generally speaking, more slowly than they otherwise would.

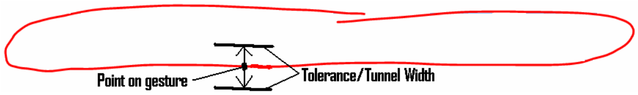

Figure 2: Inferring tolerance at a point on the gesture.

Our goal is therefore analysis of the deliberateness, or carefulness, of user gesture. More accurately, we formulate our problem as follows. Given a user gesture, such as shown in Figure 2, can we infer at any point on that gesture the intended carefulness of the user at that point? Inferring this accuracy allows us to create a "tunnel" around the gesture. Objects located inside this tunnel may be excluded or included in the selection region based on their "attachment" to objects within or outside the selection region. Based on this attachment and the tolerance in the gesture, we can develop an interpretation of the selection gesture that allows a certain degree of inaccuracy, or sloppiness, in a user's expression of his or her intention.

Speed alone is not an indicator of carefulness because any gesture normally varies in speed along its path as a function of its starting and ending points, and shape. Our analysis must effectively factor apart baseline properties of movement trajectories executed under casual conditions from properties governed by intentional constraints due to task-specific targets.

Related Work

The most successful analysis of human motion is undoubtedly Fitts' Law [Fitts 1954], relating the time taken to acquire a target with the distance from and size of the target. However, work exists on the analysis of trajectories, both in HCI and in Neuroscience. In this section, we first detail related work in trajectory analysis, before going on to detail specific work in intelligent selection gestures.

Trajectory Analysis

Trajectories have been analyzed in Neuroscience and in HCI. In Neuroscience, Flash, Hogan, and Viviani [Flash and Hogan 1985, Viviani and Flash 1995] have analyzed the characteristics of trajectories of motion by analyzing pen gestures. This research led to the development of the 2/3 Power Law and the Minimum Jerk Law, two laws of human motion that describe the instantaneous velocity of human movement during trajectories. In HCI, work has focused on the analysis of straight line motion under constraint, and resulted in the development of the Steering Law, describing the movement characteristics of users when traversing nested menu structures.

The Minimum Jerk and 2/3 Power Laws. The Minimum Jerk Law [Flash and Hogan 1985] describes the acceleration of users over a trajectory using the time derivative of acceleration, known as "jerk". This law describes the characteristic of users to prefer smooth, as opposed to "jerky", motion over trajectories. When analyzing user motion, it was determined that people typically create paths that minimize jerk.

An extension to the minimum jerk law involves the traversal of paths of varying curvatures [Viviani and Flash 1995]. X and y components of motion can be factored and jerk found to be minimized in x and y independently, leading to a relationship between the curvature of a path and the instantaneous speed of motion of a person tracing a path. Mathematically, this relationship is expressed as:

$$a(t) = k\left(c(t)\right)^{\frac{2}{3}}$$

In this equation, a(t) represents the angular velocity at time t, c(t) is the curvature, and k is a constant, typically called the "velocity gain factor".

The equation for the 2/3 Power Law can be rewritten in terms of tangential velocity via a simple mathematical manipulation, and based on the fact that v(t) = r(t) a(t) and c(t) = 1/r(t), specifically:

$$v(t) = k\left(r(t)\right)^{\frac{1}{3}}$$

In the work that follows, we plot speed vs. radius of curvature raised to the 1/3 power. Where the range of the x-axis becomes too large, we may revert to examining speed versus curvature raised to the 2/3 power.

The Steering Law. While traversing constrained paths, a person's trajectory's velocity tends to be governed by the level of constraint. More highly constrained paths tend to be traversed more slowly than less constrained paths. The "Steering Law" relates path length and path width to the time taken to traverse a path, much as Fitts' Law relates size and distance from target to time taken to acquire a target [Accot and Zhai 1997]. Specifically:

$$T_c = a + b(A/W)$$

Here, we see that T_c, the time taken, is proportional to the length of the "tunnel" whose boundaries constrain the path, and inversely proportional to the width of the tunnel. In other words, the narrower the tunnel and the longer the tunnel the more time it takes to traverse the tunnel. This predicts an inverse linear relationship between width of the tunnel and time spent in the tunnel.

Other Related Research

In overall focus, our work bears some relationship to work on Intelligent Scissors [Mortensen and Barrett 1998]. Their work involved boundary detection, with the goal of extracting relevant objects from an image. In this work, given a gesture a user draws with a mouse, the segment snaps to an appropriate object in the image. They call this technique the live wire technique, where drawing a gesture around an object ends up creating an encircling gesture for the object.

More generally, our approach is based on inferring user intention from action and context. In the InkScribe system, Saund and Lank describe a modeless interaction mechanism, called the Inferred Mode Protocol, which examines user action and the context of the action with the goal of inferring likely user intention from the action [Saund and Lank, 2003]. The idea of inferring user intention has also been explored in the domain of architecture drawings, where Do has used domain constraint, specifically the fact that a user is engaged in architecture design, to perform analyses of the designer's intention [Do 2002].

Analyzing User Trajectory

When a user draws a selection gesture in an interface, the whitespace—or lack of it---around an object or group they intend to select represents a basic level of constraint on the gesture. Other forms of constraint may also apply. For example, a user might intend to carve up image material by cutting digital ink strokes into pieces in specific places.

Our hypothesis is that the motion dynamics of a user includes information on the relative deliberateness of selection gestures. Analysis of a gesture's instantaneous speed is confounded by the varying curvature of the trajectory. To use motion dynamics in the analysis of user gestures, an understanding of the motion characteristics of users under constraint is essential.

The Minimum Jerk Law and the 2/3 Power Law deal with unconstrained motion. As well, the Steering Law was formulated in terms of straight or circular paths, rather than paths of arbitrary curvature and constraint. These laws must be extended to develop a more complete picture of instantaneous human motion under varying curvatures and constraints.

To study human motion dynamics of selection gestures, we implemented an application that presented users with targets of varying visible path constraints and curvatures. We conducted user trials, asking users to draw a series of gestures. We then analyzed the trajectories to determine the relationship between velocity, curvature, and target path constraint.

In this section, we first describe our experimental set-up to capture user data. Next, we describe our analysis of that data. Finally, we describe the characteristics of mid-path constrained motion in pen-based interfaces.

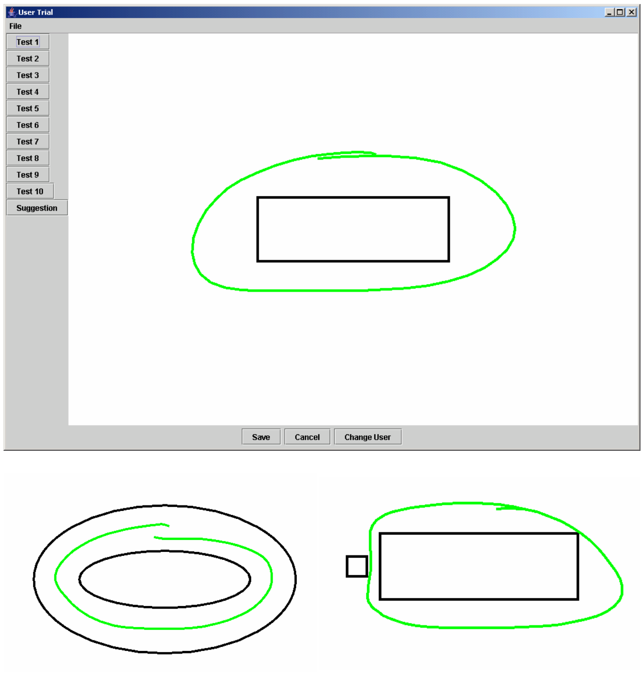

Figure 3: Sample drawing tasks given to users.

Experimental Data Capture

To develop a model of user constraint, we designed an application that generates a series of drawing tasks for the user with varying constraints. The application captured user drawing strokes, and allowed analysis of the strokes.

Figure 3 depicts the gesture selection tasks users were asked to perform. We collected data from ten users, nine right handed and one left handed. Each of the users was asked to draw a circular gesture between two objects or, in the unconstrained case, around an object. They could "cancel" a stroke until they were satisfied, and then, by pressing the "save" button, save the path they drew.

Figure 3 shows the interface used to capture the data. Users would cycle between the eleven trails, saving one gesture for each trial. Test 1, shown at the top, allowed a path unconstrained in its outer extent. Tests 2 through 6 constrained paths to lie between ovals of varying widths. Finally, tests 8 through 10 were a series of paths around a larger block and through a channel between a larger and smaller block. The smaller block was placed, in tests 8 and 9, along the longer side of the larger block and in tests 9, 10, and 11 along the shorter edge. Test 10 is shown at the bottom right in Figure 3.

Data Analysis

To analyze the characteristics of motion during selection, we examined speed and velocity profiles of pen selection gestures. To capture speed and velocity accurately, we performed a linear least squares fit of a third order polynomial to the data path, assuming a polynomial of the form:

$$x(t) = a_o + a_1 t + a_2 t^2 + a_3 t^3$$

$$y(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3$$

As shown, we fit x and y independently as a function of time (t). To obtain x and y velocity directly from our coefficients a_1 and b_1, we set the time at any point along our path to "0" and fit our curve relative to this time using points in either direction. To create a data set for our fitting function, we used a number of points before and after our current point that represented at least 10% of the total curve length. The result is that, at each point, we fit polynomials in x and y to a segment of a user's gesture totaling approximately 20% of the total gesture length. The long curve length for our fitting function minimized discontinuities from pixelization and sampling rate.

Curvature was calculated using the standard curvature formula:

$$\kappa = \frac{|v_x a_y - v_y a_x|}{(v_x{}^2 + v_y{}^2)^{\frac{3}{2}}}$$

We note that when users initially start to draw curves, their strokes follow velocity profiles that correspond to the

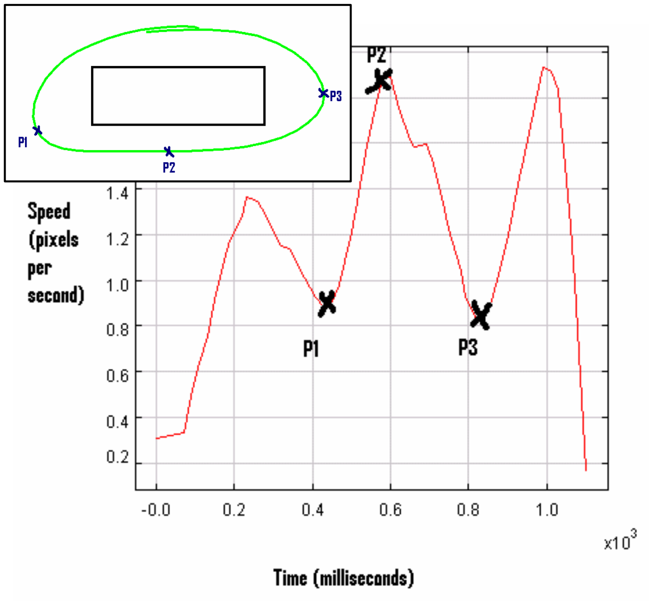

Figure 4: Speed versus time of an unconstrained path. The user starts the stroke and speeds up, then slows down to go around the higher area of curvature (P1). The user then reaccelerates through P2, at low curvature slows down to go around the end (P3), then briefly speeds up before slowing down to stop.

minimum jerk law. As shown in Figure 4, the pen gradually accelerates to a maximum velocity. As the trajectory enters an area of high curvature, the 2/3 Power Law takes over, and the pen decelerates. Figure 4 depicts a typical velocity profile for a test subject in our user trial. The horizontal axis is milliseconds, the vertical axis is pixels per millisecond.

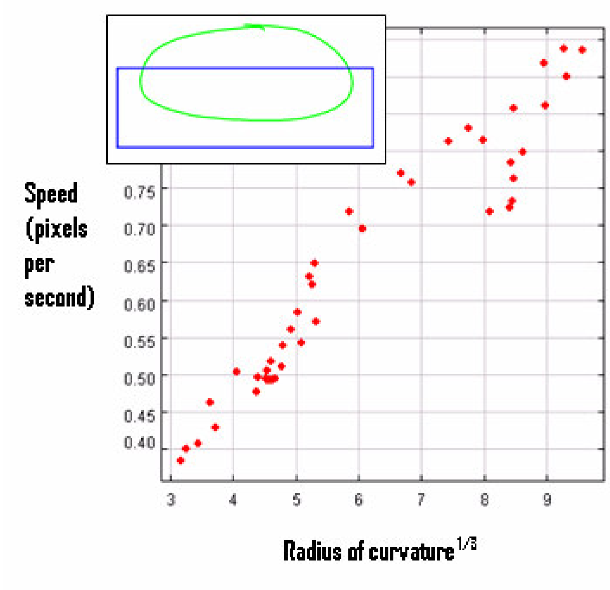

Figure 5: Speed vs. Radius of Curvature$^{1/3}$ of an unconstrained gesture.

Effects of Curvature under Constant Constraint

When drawing a closed loop gesture, users typically speed up according to minimum jerk characteristics. As shown in Figure 4, they reach an initial maximum speed and then slow down as the path curves.

To simplify our analysis, we focused only on the effects of curvature under varying degrees of constraint. For our first set of tests, tests 2 through 6, we constrained the entire user gesture, and held constraint constant over the entire gesture. We selected a portion of the gesture away from the endpoints and analyzed that portion of the gesture with respect to speed and curvature.

For unconstrained gestures, we expect speed vs. curvature plots to obey the 2/3 Power Law. Figure 5 demonstrates the linear relationship between Speed (vertical axis) and Radius of Curvature$^{1/3}$ (horizontal axis).

In Figure 6, we see the results of successively more constrained paths. The paths are constrained by widths, with path widths of 80 pixels, 60 pixels, 40 pixels, 20 pixels, and 10 pixels for trials 2 through 6. Linear correlation (r) varies between 0.88 and 0.97 for these values.

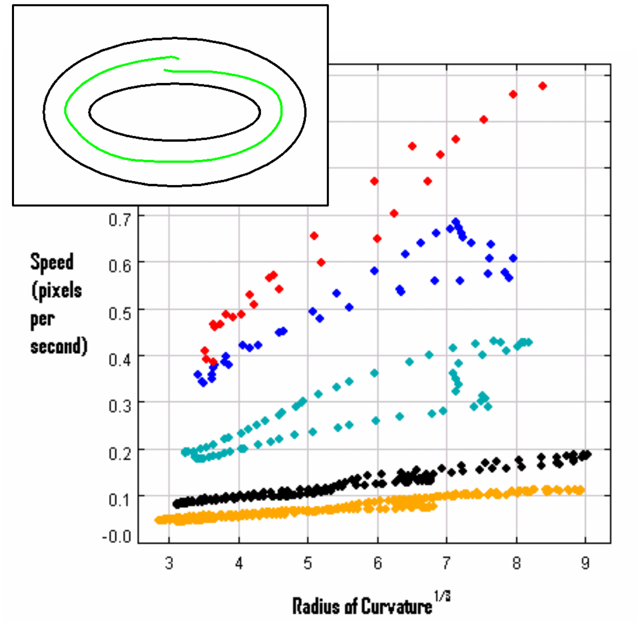

Figure 6: Speed vs. Radius of Curvature$^{1/3}$ under constraints of 80, 60, 40, 20, and 10 pixel tunnels for a single user.

When we examine the most constrained path, of 10 pixels, for five of our users, we see in Figure 7 that the linear relationship between the values continues to hold. Linear correlation remains between 0.86 and 0.97 for these values, with all but one r value over 0.95.

While the linear relationship between speed and radius of curvature$^{1/3}$ was well-known for unconstrained paths, the fact that this relationship is maintained under target constraint appears to be a novel observation.

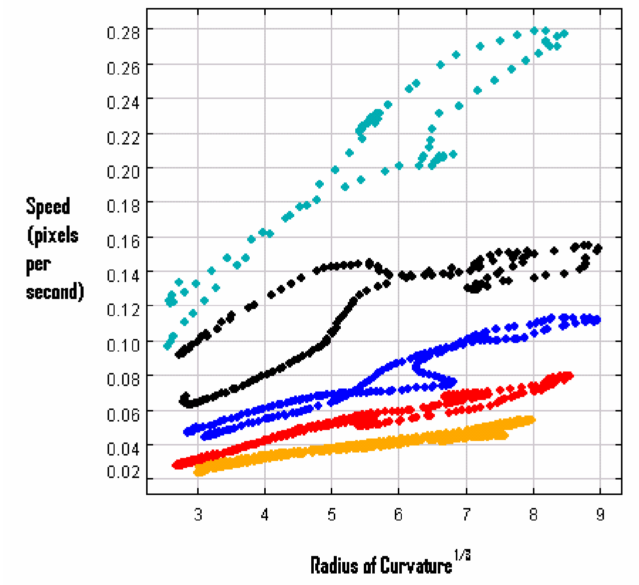

Figure 7: Speed vs. Radius of Curvature$^{1/3}$ for five users with path width of 10 pixels. Note the wide variability of user drawing speeds in spit of the identical constraint.

One other aspect of our observations is the relatively wide range of drawing speeds for various users, as shown in Figure 7. Here we see that for a 10-pixel path, the fastest user drew a gesture more than four times faster (0.28 pixels per second) than the slowest user. This could in itself be a result of variable baseline deliberateness on the part of different users. We noted that some users drew slowly and were very careful to traverse only the middle of the path when drawing, while others were more tolerant to contact with the edges of the path.

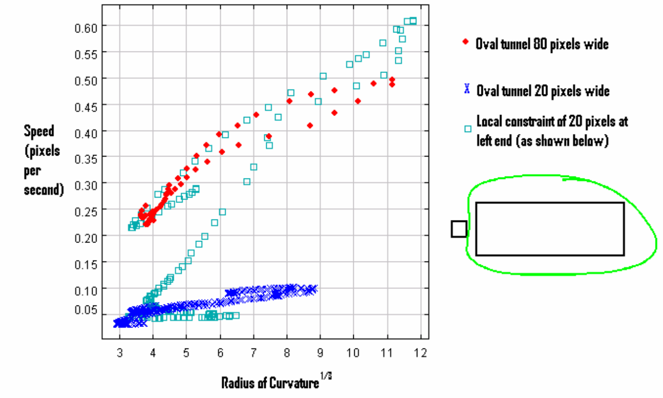

Figure 8: Speed vs Radius of Curvature$^{1/3}$. The plot in red diamond shapes at the top depicts a speed profile for a slightly constrained path (80 pixels in width). The dark blue "X" plot at the bottom is a more constrained 20 pixel wide path. Finally, the light blue "box" shaped points represent the curve drawn over a locally constrained path pictured to the right.

Varying Curvature and Constraint

Tests 7 through 11 require users to draw a gesture around a large rectangle, and between it and another smaller square (see call-out in Figure 6). There were two path constraints, a 40 pixel constraint and a 20 pixel constraint, placed either along the length of a rectangle (tests 7 and 8) or along the width of a rectangle (tests 9 and 10).

Figure 8 examines data points along a locally constrained curve. The call-out depicts the constraint path. We've superimposed three plots in this graph. The first, in diamond points at the top, depicts a user drawing under a weak path constraint, namely, an oval of 80 pixels in width. The second plot, in "X" points at the bottom, depicts a user drawing in a constrained oval path of 20 pixels in width. Finally, a plot drawn in small squares is displayed on the graph. This plot is the locally constrained path shown in the call-out. All these strokes were drawn by the same user performing different tests.

The square-point plot of the locally constrained path is revealing. It coincides with both the dark blue plot and the red plot at different points over its length.

Our analysis of what occurs is as follows. The path plotted in "X" points at the bottom of the image represents the effect of the Steering Law placing an upper bound on user speed. We see, in this plot, that curvature is still a factor in user speed, with areas of high curvature exhibiting slower speeds than areas of low constraint over the course of the stroke. However, the speed of the plot against curvature is consistently slower than that of the diamond point plot, due to path constraint.

Where the plot with square points is locally constrained to a 20 pixel tunnel, it coincides in speed to the plot with "X" points. Where it is unconstrained, it coincides, instead with the plot of the slightly constrained path drawn using red diamonds.

Analyzing User Behavior

The observations we have made are based on early pilot studies from a relatively small data set of ten users. While caution should be taken in drawing strong conclusions from the modest data set, some characteristics of user motion seem to be evident.

First, all our data leads us to believe that the relationship between speed and radius of curvature described by the 2/3 Power Law for unconstrained gestures is preserved under path width constraint. With knowledge of curvature, we can predict the expected speed at different points given constant constraint on the path.

Second, the Steering Law describes an inverse linear relationship between speed and tunnel width. In our results, we demonstrate that the effects due to tunneling and curvature are preserved for individual users.

Designing for Intelligent Selection

We are in the process of designing a prototype application that performs more intelligent selection based on the dynamics of user motion. In this section, we outline some of the design decisions, and describe the current status of our prototype application.

Speed and Constraint from Context

One challenge in the design of intelligent selection is in the creation of a usable model of path motion and constraint. From the basic model of dynamics developed in the previous section, we know that, given the path characteristics at several points with weak constraint (i.e. wider tunnel widths), we can calculate expected behavior at constrained points.

Our goal, however, is to infer likely tunnel width from motion characteristics, rather than predict speed given tunnel widths over the path. By analyzing the image content the selection gesture acts upon, we can begin to develop a partial model of motion characteristics. Consider, for example, Figure 9, where a user has drawn a selection gesture around the operand in an integral. It was the user's intention to exclude the "x" in the term "dx" from the selected region.

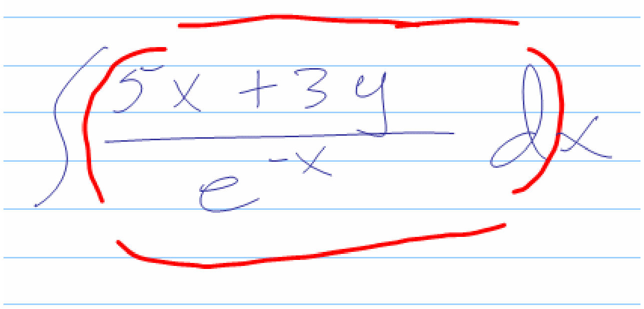

Figure 9: A selection gesture segmented into constrained and unconstrained components.

Saund and Lank note in their work on inferred mode that, given an action and the context of an action, a significant amount of information exists that can be mined by applications [Saund and Lank 2003]. In the contrived example in Figure 9, we have segmented the selection gesture into two unconstrained portions (above and below the integral) and two constrained portions (to the left and the right). Given an understanding of the underlying image material, constraints on a user's gesture can be developed for each individual segment of the gesture dynamically.

If a user draws a gesture where speed varies only with curvature, we assume that the user is operating without taking into account constraint. We may therefore estimate the cognitively unconstrained regions of the selection gesture to extend +/- 40 pixels (i.e. a variability of up to 80 pixels). This value is based on results from our earlier user trials, where tunnels of width between 60 and 80 pixels on a 15 inch Wacom Cintiq data tablet in 1024X768 resolution appeared to exhibit similar behavior to unconstrained paths.

Inferring Tolerance in Gestures

Given that we have inferred a tunnel width for unconstrained portions of the gesture, when a user acts with constraint, we need to infer the level of the constraint (i.e. the perceived tunnel width from the user's perspective). This problem is the one we are currently exploring.

Fortunately, with the assumptions in the previous section, this problem is relatively easy to solve. At any point, given the characteristics of the unconstrained portions of the gesture, we can simply divide the current velocity by the expected velocity to get a tunnel width, i.e.:

$$W = Min\left(T_0\left(\frac{v_i}{v_p}\right), T_0 \right)$$

In this equation, W is the tolerance we should allow in the user's selection gesture, v_i is the observed velocity at the current point, and v_p is the predicted velocity. The Min function accounts for outliers.

Conclusions and Future Work

We have outlined on-going work in the analysis of deliberateness versus sloppiness in selection gestures. We describe an analysis of gesture trajectory and speed under varying curvature and constraint. Evidence is provided for a number of conclusions, including the extension of the 2/3 Power Law to speed profiles for constrained gestures and a validation of the principle of the Steering Law, specifically a linear relationship between tunnel width and speed, for paths of varying curvature. We have built a proof-of-concept prototype application demonstrating tolerance to sloppy selection gestures.

To make intelligent decisions about inclusion or exclusion of image material in a selection region, relevant groupings of the underlying image material would be useful. Understanding the strength of groups in the material being selected would allow the incorporation of prior probabilities into our decision to segment or include or exclude a specific component of a group. We have designed a clustering algorithm for digital ink based on the Earth Mover's Distance. Our current work seeks to use these groupings in our inclusion/exclusion/segmentation decisions.

References

J. Accot and S. Zhai, "Beyond Fitts' law: models for trajectory-based HCI tasks", *Proceedings of the Conference on Human Factors in Computing Systems, CHI 1997*, pp. 295 – 302

Ellen Do, “Drawing Marks, Acts and Reacts: Toward a Computational Sketching Interface for Architectural Design”, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Volume 16 (2002), pp. 149 – 171.

P. M. Fitts, “The information capacity of the human motor system in controlling the amplitude of movements”, *Journal of Experimental Psychology*, 47 (1954), pp. 381-391.

T. Flash and N. Hogan, “The coordination of arm movements: An experimentally confirmed mathematical model”. *Journal of Neuroscience*, Vol. 5 (1985), pp. 1688-1703.

E. N. Mortensen and W. A. Barrett, “Interactive Segmentation with Intelligent Scissors”, *Graphical Models and Image Processing,* Vol 60, No. 5, September 1998, p.349-384.

E. Saund and E. Lank, “Stylus-Based Input and Editing Without Prior Selection of Mode”, *Symposium on User Interface Systems and Technology, UIST 2003*, Vancouver, BC, Canada, pp. 213-216.

P. Viviani and T. Flash, “Minimum Jerk, Two-Thirds Power Law and Isochrony: Converging Approaches to Movement Planning”, *Journal of Experimental Human Perception and Performance*, Vol. 21 (1995), pp. 32 - 53.