

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
21 April 2011 (21.04.2011)

(10) International Publication Number  
**WO 2011/044677 A1**

- (51) **International Patent Classification:**  
*G06F 3/03* (2006.01)      *G06F 3/14* (2006.01)
- (21) **International Application Number:**  
PCT/CA2010/001589
- (22) **International Filing Date:**  
13 October 2010 (13.10.2010)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**  
61/252,064      15 October 2009 (15.10.2009)      US
- (71) **Applicant (for all designated States except US):** SMART TECHNOLOGIES ULC [CA/CA]; 3636 Research Road N.W., Calgary, Alberta T2L 1Y1 (CA).
- (72) **Inventors; and**
- (75) **Inventors/Applicants (for US only):** ANTONYUK, Viktor [UA/CA]; 3417 Cedarille Drive S.W., Calgary, Alberta T2W 3J2 (CA). GOETHALS, Joseph Michael [CA/CA]; 20 42 Street S.W., Calgary, Alberta T3C 1Y1 (CA). BANERJEE, Shymmon [CA/CA]; 175 Dalcastle Close N.W., Calgary, Alberta T3A 1Z4 (CA).
- (74) **Agents:** RUSTON, David, A. et al.; Sim & McBurney, 330 University Avenue 6th Floor, Toronto, Ontario M5G 1R7 (CA).

- (81) **Designated States (unless otherwise indicated, for every kind of national protection available):** AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States (unless otherwise indicated, for every kind of regional protection available):** ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

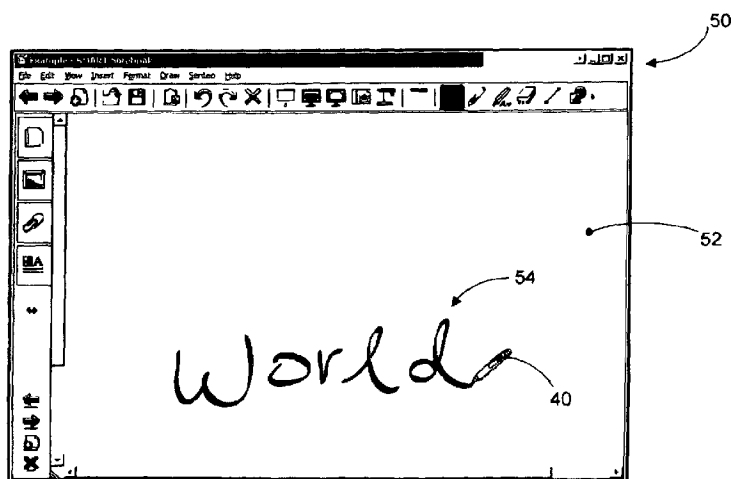
**Declarations under Rule 4.17:**

— as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

**Published:**

— with international search report (Art. 21(3))

(54) **Title:** METHOD AND APPARATUS FOR DRAWING AND ERASING CALLIGRAPHIC INK OBJECTS ON A DISPLAY SURFACE



**FIG. 3A**

(57) **Abstract:** A method of generating a calligraphic ink object, comprising sampling contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing; generating an ink envelope, the ink envelope comprising line segments joining pointer instances at the sampled contact coordinates; generating a smoothed ink envelope at least by fitting curves to points on the ink envelope; and drawing the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

WO 2011/044677 A1

**METHOD AND APPARATUS FOR DRAWING AND ERASING  
CALLIGRAPHIC INK OBJECTS ON A DISPLAY SURFACE**

**Field Of The Invention**

[0001] The present invention relates to a method and apparatus for drawing and erasing calligraphic ink objects on a display surface.

**Background Of The Invention**

[0002] Interactive input systems that allow users to inject input (e.g. digital ink, mouse events etc.) into an application program using an active pointer (e.g., a pointer that emits light, sound or other signal), a passive pointer (e.g., a finger, cylinder or other suitable object) or other suitable input device such as for example, a mouse or trackball, are known. These interactive input systems include but are not limited to: touch systems comprising touch panels employing analog resistive or machine vision technology to register pointer input such as those disclosed in U.S. Patent Nos. 5,448,263; 6,141,000; 6,337,681; 6,747,636; 6,803,906; 7,232,986; 7,236,162; and 7,274,356 assigned to SMART Technologies ULC of Calgary, Alberta, Canada, assignee of the subject application, the entire contents of which are incorporated by reference; touch systems comprising touch panels employing electromagnetic, capacitive, acoustic or other technologies to register pointer input; tablet personal computers (PCs); laptop PCs; personal digital assistants (PDAs); and other similar devices.

[0003] A number of applications that generate calligraphic ink from a user's handwritten input (i.e. ordinary ink) have been considered. For example, these applications include Microsoft™ Tablet PC applications, and the calligraphy tool implemented in the Inkscape Open Source vector graphics editor. Compared to ordinary ink, which is represented as a pixel-based image object, calligraphic ink is represented as a vectorized image. As a result, calligraphy ink has a smoother appearance than ordinary ink, and can be arbitrarily zoomed without showing jagged or zig-zag lines (i.e. stair-stepping) that are usually seen at the edges of a zoomed ordinary ink object. Unfortunately existing calligraphic ink generating applications still exhibit disadvantages. In particular, existing calligraphic ink generating applications

-2-

are slow, do not present the outlines of calligraphic ink objects with the desired degree of smoothness and do not readily permit portions of calligraphic ink objects to be erased.

[0004] Methods of erasing ink objects have been considered. For example, U.S. Patent No. 6,326,954 to Van Ieperen et al. entitled "Method For Erasing On An Electronic Writeboard", assigned to SMART Technologies ULC of Calgary, Alberta, Canada, assignee of the subject application, teaches a method of erasing at least a portion of an ink object displayed on an electronic writeboard. However, when this method is applied to calligraphic ink objects, the degree of smoothness of the unerased portion of the calligraphic ink object typically does not meet the desired standard.

[0005] Therefore, there is a need to provide a novel method of generating and erasing calligraphic ink objects on a display surface.

### **Summary Of The Invention**

[0006] Accordingly, in one aspect there is provided a method of generating a calligraphic ink object, comprising sampling contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing; generating an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates; generating a smoothed ink envelope at least by fitting curves to points on said ink envelope; and drawing the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

[0007] In one embodiment, the sampling contact coordinates further comprises processing the sampled contact coordinates to reduce the number of sampled contact coordinates used to generate the ink trajectory. In particular, selected sampled contact coordinates that are co-linear with other sampled contact coordinates are discarded.

[0008] In one embodiment, the ink envelope generating is performed upon completion of writing, such as for example when a pointer up event occurs. The ink envelope generally completely surrounds the ink trajectory.

The ink envelope generating comprises surrounding each sampled contact coordinate along the generated ink trajectory with a pointer instance and for each consecutive pair of sampled coordinates, generating line segments on opposite sides of a line segment joining the sampled contact coordinates of the pair that extend between the pointer instances surrounding the sampled contact coordinates of the pair. The generated line segments extending between the pointer instances surrounding the sampled contact coordinates of each pair may be substantially tangential to the peripheries of the pointer instances.

[0009] In one embodiment, generating the smoothed ink envelope comprises fitting Bezier curves to points on the ink envelope.

[0010] In another embodiment, the ink trajectory comprises ink trajectory segments. An ink envelope segment is generated for each ink trajectory segment and the ink envelopes segments are combined to form the ink envelope.

[0011] In one embodiment, the drawing comprises drawing the smoothed ink envelope as a vectorized image. Prior to the drawing, the smoothed ink envelope may be colored or shaded.

[0012] According to another aspect there is provided a method of generating a calligraphic ink object comprising generating an ink trajectory representing writing input on a coordinate input device using an object, the ink trajectory comprising a subset of contact coordinates generated by the coordinate input device; generating an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and drawing the ink envelope on a display thereby to generate the calligraphic ink object.

[0013] In one embodiment, the curved line segments are Bezier curves and the ink envelope resembles the boundary of the path of a geometric pointer tip following the ink trajectory.

[0014] According to yet another aspect there is provide an apparatus comprising at least one processor; and memory storing a calligraphic ink object generating routine, the calligraphic ink object generating routine, when

-4-

executed by the at least one processor, causing the apparatus to: sample contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing; generate an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates; generate a smoothed ink envelope at least by fitting curves to points on said ink envelope; and draw the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

[0015] According to yet another aspect there is provided an apparatus comprising at least one processor; and memory storing a calligraphic ink object generating routine, the calligraphic ink object generating routine, when executed by the at least one processor, causing the apparatus to: generate an ink trajectory representing writing input on a coordinate input device using an object, the ink trajectory comprising a subset of contact coordinates generated by the coordinate input device; generate an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and draw the ink envelope on a display thereby to generate the calligraphic ink object.

[0016] According to yet another aspect there is provided a computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to sample contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing; generate an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates; generate a smoothed ink envelope at least by fitting curves to points on said ink envelope; and draw the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

[0017] According to yet another aspect there is provided a computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to: generate an ink trajectory representing writing input on a coordinate input device using an object, the ink

-5-

trajectory comprising a subset of contact coordinates generated by the coordinate input device; generate an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and draw the ink envelope on a display thereby to generate the calligraphic ink object.

[0018] According to yet another aspect there is provided a method of erasing at least a portion of a calligraphic ink object delineated by an eraser trace intersecting said calligraphic ink object, said method comprising calculating intersection points between said calligraphic ink object and said eraser trace; generating additional points along line segments that extend between the intersection points on opposite sides of the eraser trace; fitting curves to the intersection points and the additional points; modifying the outline of said calligraphic ink object using said curves; and deleting the portion of said calligraphic ink object between said curves.

[0019] According to yet another aspect there is provided an apparatus comprising at least one processor; and memory storing a calligraphic ink object erasing routine, the calligraphic ink object erasing routine, when executed by the at least one processor, causing the apparatus to: calculate intersection points between said calligraphic ink object and said eraser trace; generate additional points along line segments that extend between the intersection points on opposite sides of the eraser trace; fit curves to the intersection points and the additional points; modify the outline of said calligraphic ink object using said curves; and delete the portion of said calligraphic ink object between said curves.

[0020] According to still yet another aspect there is provided a computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to calculate intersection points between said calligraphic ink object and said eraser trace; generate additional points along line segments that extend between the intersection points on opposite sides of the eraser trace; fit curves to the intersection points and the additional points; modify the outline of said

calligraphic ink object using said curves; and delete the portion of said calligraphic ink object between said curves.

### **Brief Description Of The Drawings**

[0021] Embodiments will now be described more fully with reference to the accompany drawings in which:

[0022] Figure 1 is a simplified block diagram of an interactive input system;

[0023] Figure 2 is a perspective view of a portion of the interactive input system of Figure 1;

[0024] Figures 3A to 3C illustrate a graphical user interface presented on an interactive whiteboard forming part of the interactive input system of Figure 1;

[0025] Figure 4 is a flowchart showing the steps performed during calligraphic ink object generating;

[0026] Figure 5 shows an exemplary ink trajectory after removing co-linear points;

[0027] Figure 6 shows a curve-fitted and normalized ink trajectory overlaid on the ink trajectory of Figure 5 before curve-fitting;

[0028] Figure 7 illustrates an outer envelope and an inner envelope surrounding points on the normalized ink trajectory;

[0029] Figures 8A to 8C show an example of raw envelope point generating and curve fitting;

[0030] Figure 8D is a flowchart showing the steps performed during indexing of points along the normalized ink trajectory;

[0031] Figure 8E shows an exemplary calligraphic ink object;

[0032] Figure 9A is a flowchart showing alternative steps performed during calligraphic ink object generating;

[0033] Figure 9B shows two exemplary segments of an ink object;

[0034] Figure 9C shows another exemplary calligraphic ink object;

[0035] Figure 10 a flowchart showing the steps performed during calligraphic ink object erasing;

-7-

[0036] Figures 11A to 11D show an exemplary procedure of erasing a portion of a calligraphic ink object; and

[0037] Figures 12A to 12D show another exemplary procedure of erasing a portion of a calligraphic ink object.

### **Detailed Description Of The Preferred Embodiment**

[0038] Turning now to Figure 1, a block diagram of an interactive input system is shown and is generally identified by reference numeral 10.

Interactive input system 10 comprises a coordinate input device 12 such as for example a touch panel, interactive surface or interactive whiteboard on which pointer contacts can be made. The coordinate input device 12 communicates with processing structure 14 executing one or more application programs. Image data generated by the processing structure 14 is displayed on a display surface of the coordinate input device 12 allowing a user to interact with the displayed image via pointer contacts on the coordinate input device 12. The processing structure 14 interprets pointer contacts as input to the running application program and updates the image data accordingly so that the image displayed on the display surface of the coordinate input device 12 reflects the pointer activity. In this manner, the coordinate input device 12 and processing structure 14 allow pointer interactions with the coordinate input device 12 to be recorded as handwriting or drawing or used to control execution of the running application program. The coordinate input device 12 of the interactive input system 10 may be separate from the processing structure 14, or may be combined with the processing structure 14 to form an integral compact unit as in the case of personal computers (PCs), tablet PCs, laptop PCs, personal digital assistants (PDAs), cellular telephones or other suitable devices.

[0039] Figure 2 illustrates an exemplary coordinate input device 12 in the form of an interactive whiteboard (IWB) 20. In this embodiment, the IWB 20 is a 600i series interactive whiteboard manufactured by SMART Technologies ULC, of Calgary, Alberta, Canada, assignee of the subject application. The IWB 20 comprises a large, analog resistive touch screen 22

-8-

having a touch surface 24, which also acts as a display surface on which computer generated images are presented. The touch surface 24 is surrounded by a bezel 26. A tool tray 28 is affixed to the bezel 26 adjacent the bottom edge of the touch surface 24 and accommodates one or more pen tools that are used to interact with the touch surface 24. The touch screen 22 is mounted on a wall or other suitable support surface via a mounting bracket 30. A boom assembly 32 is also mounted on the wall above the touch screen 22 via the mounting bracket 30. The boom assembly 32 comprises a speaker housing 34 accommodating a plurality of speakers (not shown), a generally horizontal boom 36 extending outwardly from the speaker housing 34 and a projector 38 adjacent the distal end of the boom 36. The projector 38 is aimed back towards the touch screen 22 so that the image projected by the projector 38 is presented on the touch surface 24.

[0040] When the user uses a pen tool 40 or a finger to contact the touch surface 24, the touch screen 22 detects the contact and transmits contact information to the processing structure 14 (see Figure 1). The processing structure 14 obtains or calculates the pointer coordinates at the contact location(s) on the touch surface 24 using the contact information transmitted from the touch screen 22. Depending on the operating mode of the IVB 20, the pen tool contact coordinates are interpreted as a digital ink drawing object, a digital ink erasing object, or a command (mouse) event.

[0041] The processing structure 14 in this embodiment is a general purpose computing device in the form of a personal computer or other suitable processing device. The computer comprises for example, a processing unit, system memory (volatile and/or non-volatile memory), other non-removable or removable memory (a hard disk drive, RAM, ROM, EEPROM, CDROM, DVD, flash memory etc.) and a system bus coupling the various computer components to the processing unit. It will be understood that the computer may also include a network connection to access shared or remote drives, one or more networked computers, or other networked devices over wired and/or wireless communication links using an appropriate communications format (e.g. Ethernet, WiFi, BlueTooth etc.)

[0042] The processing structure 14 runs a host software application such as SMART Notebook™ offered by SMART Technologies ULC of Calgary, Alberta, Canada. As is known, during execution, the SMART Notebook™ application provides a graphical user interface comprising a canvas page or palette, that is presented on the touch surface 24, and on which freeform or handwritten ink objects together with other computer generated objects can be input and manipulated via pointer interaction with the touch surface 24.

[0043] Figures 3A to 3C show the graphical user interface 50 presented on the touch surface 24 as a SMART Notebook™ application window. As shown in Figure 3A, when a user writes on the canvas page 52 inside the SMART Notebook™ application window 50 using the pen tool 40, as the pen tool 40 is moved along the touch surface 24, the touch screen 22 transmits contact information to the processing structure 14. The processing structure 14 in turn uses the contact information to determine the pen tool contact coordinates and interprets the pen tool coordinates as input digital ink. The input digital ink is used by the processing structure 14 to create a calligraphic ink object 54 representing the user's writing on the touch surface 24. The processing structure 14 also updates the image data conveyed to the projector 38 so that the calligraphic ink object 54 is presented on the touch surface 24. The image data is updated substantially immediately in "real time" by the processing structure 14 so that the projected image includes the user's writing substantially as soon as it is input.

[0044] As shown in Figure 3B, when an eraser tool 60 is dragged along a path or trace 62 across the calligraphic ink object 54 displayed on the canvas page 52, the portion 64 of the ink calligraphic ink object 54 that intersects the trace 62 is erased. The calligraphic ink object 54 after the erasing operation is shown in Figure 3C.

[0045] Figure 4 shows the general steps performed by the processing structure 14 during generation of a calligraphic ink object. At step 88, the processing structure 14 receives contact information from the touch screen 22 in response to a user writing on the touch surface 24 using a pen tool 40 or

-10-

finger. In this embodiment, the contact information is a series of sampled coordinates representing contact points on the touch surface 24.

[0046] The processing structure 14 generates the calligraphic ink object from the sampled contact points based on the brush type the user is using. In this embodiment, the brush type is a virtual pointer tip selected by the user from a set of predefined virtual pointer tip shapes, including circles, ellipses, polygons, etc., together with a user selected pointer tip size. In this case, the sampled contact points obtained at step 88 form the trajectory of the ink object, where each sampled contact point represents the centroid of the corresponding instance of the user selected brush type.

[0047] For ease of discussion, the calligraphic ink object generating method employed by the processing structure 14 will be described assuming that an elliptical brush has been selected by the user. At step 100, the processing structure 14 checks the sampled contact points of the ink trajectory, and removes co-linear contact points. For example, if  $P_1, P_2, \dots, P_n$  represent the sampled contact points of the ink trajectory, during the co-linear contact point removal process, the processing structure 14 calculates the slope of a line segment connecting the contact points  $P_i$  and  $P_{i+1}$  according to:

$$L_i = (y_{i+1} - y_i)/(x_{i+1} - x_i)$$

where  $i=1, 2, \dots, n-2$ .

Then, the processing structure 14 calculates the slope of a line segment connecting the contact points  $P_{i+1}$  and  $P_{i+2}$  according to:

$$L_{i+1} = (y_{i+2} - y_{i+1})/(x_{i+2} - x_{i+1})$$

If the calculated slopes are not equal i.e.  $L_i \neq L_{i+1}$ , the processing structure 14 continues to compare the slope of a line segment connecting contact points  $P_{i+1}$  and  $P_{i+2}$  with that of a line segment connecting contact points  $P_{i+2}$  and  $P_{i+3}$ . If the calculated slopes are equal i.e.  $L_i = L_{i+1}$ , the contact points  $P_i, P_{i+1}$  and  $P_{i+2}$  are co-linear points and the contact point  $P_{i+1}$  is removed. In this case, the processing structure 14 compares the slope of the line segment connecting contact points  $P_i$  and  $P_{i+2}$  with that of the line segment connecting contact points  $P_{i+2}$  and  $P_{i+3}$ . The process continues in this manner until all

-11-

sampled contact points are checked. Figure 5 shows an example of the ink trajectory after removing co-linear contact points.

[0048] Turning back to Figure 4, at step 102 following removal of co-linear contact points, the ink trajectory is smoothed by piecewise curve-fitting. One of a number of known piecewise curve fitting techniques may be used. Generally, the resulting ink trajectory is a set of curve-fitting functions  $f_k(\mathbf{A})$ , where  $k=1, 2, \dots$ , and  $\mathbf{A}$  is a set of parameters determined by the sampled contact points. In this embodiment, the processing structure 14 uses the piecewise curve-fitting technique described in the publication entitled "An Algorithm for Automatically Fitting Digitized Curves" authored by Philip J. Schneider, Graphics Gems, pages 612 – 626, Academic Press Professional, Inc. (1990), the entire content of which is incorporated herein by reference. The resulting ink trajectory is a set of connected Bezier curves, where each Bezier curve is a Bezier function determined by four control points calculated from the sampled contact points.

[0049] At step 104, each Bezier curve is sampled with respect to a predefined point distance. The obtained sampled contact points, which form a normalized ink trajectory, are indexed in accordance with a predefined order (e.g., the order of writing). Figure 6 shows the curve-fitted and normalized ink trajectory 120 (the curve without dots) overlaid on the ink trajectory before curve-fitting 122 (the curve with dots).

[0050] Turning back to Figure 4, at step 106 following curve-fitting, the shape of the selected virtual pointer tip is applied to each sampled contact point of the normalized ink trajectory to form an ink envelope. For any three successive sampled contact points of the normalized ink trajectory, an outer envelope and an inner envelope are defined. Referring to Figure 7, three successive sampled contact points  $T_1$ ,  $T_2$  and  $T_3$  are shown. The line segment 128 between contact points  $T_1$  and  $T_2$ , and the line segment 130 between contact points  $T_2$  and  $T_3$  form a first angle 132 greater than  $\pi$ , and a second angle 134 smaller than  $\pi$ . The envelope 136 that is within the range of the first angle 132 is defined as the outer envelope, and the envelope 138

-12-

that is within the range of the second angle 134 is defined as the inner envelope.

[0051] Figures 8A and 8B illustrate generation of the raw points of the ink envelope. As mentioned above, in this example the pointer tip has an elliptical shape. In Figure 8A, reference numerals 140, 140A, 140B and 140C represent the sampled contact points of the normalized ink trajectory. A pointer tip instance 142, 142A and 142B is applied to and centered at each sampled point contact 140. Then, the processing structure 14 calculates two line segments 146 for each two adjacent sampled contact points 140, which are tangential to the pointer tip instances at tangent points 148. In the cases where a fixed-size pointer tip is used, the line segments 146 are also parallel to the line segment 144 between the two adjacent sampled contact points 140. In other cases where the pointer tip may vary in size, shape, and/or angle, the line segments 146 may not be parallel to the line segment 144. For the two adjacent sampled contact points 140A and 140B, the processing structure 14 calculates an upper line segment 146A that is parallel to the line segment 144A between the adjacent sampled contact points 140A and 140B, and is tangential to the pointer tip instances 142A and 142B, respectively, at tangent points 148A and 148B. The processing structure 14 also calculates a lower line segment 146B that is parallel to the line segment 144A between the adjacent sampled contact points 140A and 140B, and is tangential to the pointer tip instances 142A and 142B, respectively, at tangent points 148C and 148D. The tangent points, e.g., points 148A, 148B, 148C and 148D, are recorded as raw envelope points.

[0052] For any three successive sampled contact points of the ink trajectory, the corresponding outer envelope comprises a portion of the pointer tip instance centered at the middle sampled contact point. This portion of the pointer tip instance is then sampled with respect to a predefined point distance, and the obtained sampled contact points are also recorded as raw envelope points. For example, for the three successive contact points 140A, 140B and 140C, the raw outer envelope points comprise the tangent points 148A and 148B, the points 150A and 150B of the pointer tip instance

-13-

142B, the tangent point 148E, and the intersection point 152B of the tangential line segments 146C and 146D.

[0053] To determine the raw inner envelope points for any three successive sampled contact points of the normalized ink trajectory, two cases need to be considered. The first case is illustrated in Figure 8A, where the tangential line segment 146B in the inner envelope area intersects with the tangential line segment 146C in the inner envelope area. In this case, the intersection point 152A, as well as the tangent points 148D and 148F are recorded as raw inner envelope points. Note that the tangent points 148C and 148G are dropped.

[0054] The second case is illustrated in Figure 8B, where at least a portion of the tangential line segments 160 and 162, respectively, is within the range of the first angle 164, which is smaller than  $\pi$ . In this case, although the line segments 160 and 162 form the raw inner envelope, as illustrated in Figure 8B, the line segments 160 and 162 do not intersect. In this case, the tangent point 166A, which is the end point of the tangential line segment 160 that is closest to the tangent line segment 162, and the tangential point 166B, which is the end point of the tangent line segment 162 that is closest to the tangent line segment 160, are defined as break-points.

[0055] Turning back to Figure 8A, for each end point of the normalized ink trajectory (e.g., the end point 140A), the outer part of the corresponding pointer tip (e.g., 142A) is also sampled to obtain the raw envelope points for closing the end of the ink trajectory (e.g., points 150C, 150D and 150E).

[0056] The generated raw envelope points are indexed. As shown in Figure 8C, each raw envelope point is associated with two index numbers: a natural index number, denoted as  $P_n$ , where  $n=0, 1, 2, \dots$ , and the index number of the nearest sampled contact point of the normalized ink trajectory. For example, in Figure 8C, there are five raw envelope points  $P_3, P_4, P_5, P_6$  and  $P_{37}$  near the sampled contact point  $C_1$  of the ink trajectory. Therefore, these raw envelope points are also associated with the index of the sampled contact point  $C_1$ , and are denoted as  $P_3/1, P_4/1, P_5/1, P_6/1$  and  $P_{37}/1$ , where the number before "/" represents the natural index and the number

-14-

after “/” represents the index of the nearest sampled contact point of the normalized ink trajectory. Note that the natural index of the raw envelope points may be in any convenient order, and the order shown in Figure 8C is for illustrative purposes only.

[0057] Referring again to Figure 4, following raw envelope point generation at step 106, the obtained raw envelope points are curve-fitted at step 108 to generate a smoothed ink envelope and hence a resultant smooth outline of the calligraphic ink object. Each set of successive raw outer envelope points are piecewise curve-fitted. Each set of successive raw inner envelope points between two break-points are also piecewise curve-fitted. Each pair of successive break-points are connected by a straight line. Thus, the smoothed outline of the calligraphic ink object is described by a set of curve functions, each determined by a set of calculated parameters, as well as the straight line functions connecting successive break-points. In this embodiment, the smoothed outline of the calligraphic ink object is described by a set of Bezier functions, each determined by four control points, as well as the straight line functions connecting successive break-points.

[0058] As an example, Figure 8C shows the control points of seven resulting fitting curves E1, E2, ..., E7. In this figure, the control points of the resulting curves are moved outward for the purpose of clear illustration. The actual resulting curves would be at the proximity of the raw envelope points. Also for the purpose of clear illustration, the fitting curves are not drawn in Figure 8C, but, instead, the control points are connected by dashed lines.

[0059] Each of the resulting fitting curves fits to a number of raw envelope points. The curve-fitting technique will return the range of raw envelope points to which each resulting fitting-curve fits. For example, in Figure 8C, the fitting curve E1 fits to the raw envelope points P0 to P6, and the fitting curve E3 fits to the raw envelope points P6 to P12. With this information, the sampled contact points of the normalized ink trajectory returned by the curve-fitting technique are indexed.

[0060] Figure 8D shows the steps performed during indexing of the sampled contact points of the normalized ink trajectory. First, a counter  $i$  is

-15-

initialized to one (1) (step 180) and the first fitting curve E1 is selected. At step 182, each raw envelope point to which the fitting curve E1 fits, is checked to obtain the index of the associated sampled contact point of the normalized ink trajectory. For example, in Figure 8C, raw envelope points P0 to P6 to which the fitting curve E1 fits are checked. Seen from the second part of their denotation (i.e., the number after the slash "/"), raw envelope points P0, P1 and P2 are associated with sampled contact point C0. The sampled contact point C0 is assigned an index 1 to indicate that the sampled contact point C0 is associated with fitting curve E1 and the assigned index is added to an index matrix.

[0061] Thereafter, the counter *i* is increased by 1 (step 184), and a check is made to determine whether all fitting curves have been processed (step 186). If not, the process reverts to step 182 to check the next fitting curve, in the case fitting curve E2. Otherwise, the generated index matrix is returned (step 188).

[0062] At step 182 for fitting curve E2, raw envelope points P39 and P38 fit to the fitting curve E2 and are associated with the sampled contact point C0. As a result, the sampled contact point C0 is also assigned the index 2 to indicate that the sampled contact point C0 is associated with the fitting curve E2 and the assigned index is added to the index matrix. As will be appreciated, after the raw envelope points to which each fitting curve fits have been checked, the resultant index matrix includes one or more indices for each sampled contact point identifying each fitting curve that is associated with each sampled contact point. For example with reference again to Figure 8C, sampled contact point C1 is assigned indices 1 and 2 and optionally index 3 (because raw envelope point P6 fits to both fitting curve E1 and fitting curve E3). Sampled contact point C2 is assigned indices 3 and 2 and optionally index 4. When a raw envelope point fits to multiple fitting curves, the raw envelope point can be assigned to a selected fitting curve or to both fitting curves.

[0063] Once the index matrix has been completed at step 188, the obtained fitting curve functions and the corresponding control points, as well

-16-

as the sampled contact points of the normalized ink trajectory and the index matrix are stored. This information provides a mathematic description of the calligraphic object outline. With the mathematic description of the calligraphic ink object outline obtained, the outline of the calligraphic ink object may be filled in with a user-selected color or texture by the processing structure 14 using a rendering program. Then, the calligraphic ink object may be drawn on the touch surface 24 as a vectorized image so that it can be arbitrarily zoomed while maintaining the smoothness of its outline. Figure 8E shows an exemplary calligraphic ink object generated by the processing structure 14 using the above described method.

[0064] Figure 9A shows the steps performed by the processing structure 14 during an alternative calligraphic ink object generating technique. This technique is suitable for generating calligraphic ink objects substantially in real-time and employs multi-thread/multi-core technology to calculate the smoothed ink outline segments of the calligraphic ink objects in parallel.

[0065] During this calligraphic ink object generating method, the sampled contact points of the user writing are segmented by time, i.e., starting from the first sampled contact point, the sampled contact points received by the processing structure 14 within every T milliseconds are grouped as a segment, where T is a predefined parameter. The value of the parameter T is selected from a range typically determined by the user based on writing experience and/or the processing structure performance. At step 200, the processing structure 14 receives the sampled contact points of a segment of the input ink object. At step 202, if the segment is not the first segment of the input ink object, the last sampled contact point of the previous segment is prefixed to the current segment and a storage pointer to these sampled contact points is placed into a segment heap. The ink object segment is then assigned to a background thread for generating the smoothed ink outline of the calligraphic ink object segment.

[0066] The background thread processes the ink object segment using a similar method as described with reference to Figure 4 but with a slight modification. In particular, during this method if the ink object segment is the

-17-

first segment of the input ink object, the outer part of the pointer tip instance centered at the first sampled contact point of the normalized ink trajectory is sampled to obtain raw envelope points for closing the first end of the calligraphic ink object. The envelope around the last sampled contact point of the calligraphic ink object segment is left open. If the ink object segment is not the first segment of the input ink object, then both ends of the envelope are left open.

[0067] Figure 9B shows two exemplary segments S1 and S2 of an input ink object. The raw envelope points are also marked in Figure 9B as dots. Because calligraphic ink object segment S1 is the first segment of the input calligraphic ink object, the outer part of the first pointer tip instance 220 is sampled to obtain the raw envelope points 222 for closing the calligraphic ink object segment outline. On the other hand, because calligraphic ink object segment S2 is not the first segment of the input calligraphic ink object, both ends are left open.

[0068] Referring again to Figure 9A, after sending the ink object segment to a background thread, the process checks at step 204 whether the user's writing is complete (e.g., by detecting whether a pen tool-up event occurs). If not, the process goes back to step 200 to collect another calligraphic ink object segment and send it to another background thread for processing. In this manner, calligraphic ink object outlines for different ink object segments are generated by different threads in parallel reducing the lag between the user writing the ink and written ink being displayed on the touch surface 24.

[0069] At step 204, if it is detected that the user has completed writing (e.g., when the pen tool-up event is received), the last ink object segment is then sent to a background thread for processing (step 206). In the processing of the last ink object segment, the pointer tip instance centered at the last sampled contact point of the normalized ink trajectory is also sampled at its outer part to obtain raw envelope points for closing the calligraphic ink object outline.

-18-

[0070] After all of the ink object segments have been processed, the calligraphic ink object segments are collected and smoothly connected together to form the final calligraphic ink object (step 208). Figure 9C shows such an exemplary calligraphic ink object, where each calligraphic ink object segment is contained within a respective rectangular box.

[0071] The method of erasing calligraphic ink objects employed by the processing structure 14 when a user moves an eraser across the touch surface 24 along a path or trace that intersects a calligraphic ink object will now be described. In the following, for ease of discussion, the erasing method will be described with specific reference to a circle-shaped eraser. However, those skilled in the art will appreciate that any other shape may be used as the eraser including, for example, an ellipse, triangle, rectangle, square, or other polygonal shape. The eraser may have a fixed size, or may be variable in size. When the eraser is variable size, the eraser size may be selected based on the magnitude of the pressure the user applies to the touch surface 24 using the eraser or on a user-selected size-varying scheme. Again for the ease of description, only the case where the trace of the eraser intersects the calligraphic ink object once will be discussed although those skilled in the art will appreciate that the method applies to cases where the trace of the eraser intersects the calligraphic ink object multiple times.

[0072] Figure 10 shows the steps performed by the processing structure 14 during erasing of a calligraphic ink object. At step 220, the processing structure 14 initially finds the four points on the calligraphic ink object outline where the trace of the eraser intersects the calligraphic ink object. For example, in Figure 11A, the trace of the eraser 260 intersects the outline of the calligraphic ink object 262 at points A, B, C and D.

[0073] At step 222, the calligraphic ink object outline is split at intersection points A and B and at intersection points C and D. At step 224, additional outline points are generated between points A and C, and between points B and D, respectively. Then, an additional fitting curve is generated to fit the points A, C and the additional outline points therebetween (curve 264 in Figure 11B). An additional fitting curve is also generated to fit the points B, D

and the additional outline points therebetween (curve 266 in Figure 11B). Then, the processing structure 14 closes the calligraphic ink object outline by connecting the outline to the additional fitting curves. For example, in Figure 11C, the outline of the left part 268 of the calligraphic ink object is closed by connecting it to the curve 264, and the outline of the right part 270 of the calligraphic ink object is closed by connecting it to the curve 266.

[0074] At step 230, the portion of the calligraphic ink object outline that is outside of the newly generated outline, e.g. the portion of the calligraphic ink object outline outside of the curves 264 and 266, is deleted. The newly generated calligraphic ink object outline may be filled in with a user-selected color or texture by the rendering program. The resulting calligraphic ink object outline is shown in Figure 11D. Optionally, at step 232, the newly generated calligraphic ink object outline may be checked to see if all calligraphic ink object outline components of the calligraphic ink object are connected. If not, the calligraphic ink object can be split into separate calligraphic ink objects. The separate calligraphic ink objects may be automatically grouped together or may be separate. The decision to make them separate or automatically grouped may be based on the distance between the ends of the separate calligraphic ink objects. If the separate calligraphic ink objects are spaced by a distance greater than a threshold amount, then the calligraphic ink objects will be treated as separate ink objects; otherwise, they are automatically grouped.

[0075] Depending on the nature of the processing structure employed, this erasing method may be slow because step 220 typically requires a large amount of calculation to find the intersection points. In certain instances, a fast erasing methodology may be employed as will now be described.

[0076] The fast erasing methodology generally follows the same steps as shown in Figure 10. However, at step 220, the fast erasing methodology does not directly determine the intersection points of the calligraphy ink object outline and the eraser trace. Instead, the fast erasing methodology first finds the intersection points of the normalized ink trajectory and the eraser trace. Referring to Figure 12A, each sampled contact point of the normalized ink

-20-

trajectory is associated with an index matrix, which is a pair of indices (x,y) in this figure, where x refers to the index of the fitting curve above the normalized ink trajectory, and y refers the index of the fitting curve below the normalized ink trajectory. These indices are obtained when the calligraphic ink object is generated as previously described.

**[0077]** The processing structure 14 first searches the sampled contact points of the normalized ink trajectory to find a sampled contact point within the eraser trace and nearest to the left edge of the eraser trace, which is, in the case shown in Figure 12A, contact point C4. The processing structure 14 also searches the sampled contact points of the normalized ink trajectory to find a sampled contact point within the eraser trace and nearest to the right edge of the eraser trace, which is, in the case shown in Figure 12A, contact point C5. In an alternative implementation, the actual intersection points of the normalized ink trajectory and the calligraphic ink object outline may be found (which are interpolated points between contact points C3 and C4, and contact points C5 and C6, respectively). The index of each actual intersection point may be copied from its nearest sampled contact point of the normalized ink trajectory.

**[0078]** After the intersection contact points C4 and C5 are found, their indices are checked to find the corresponding fitting curve on the calligraphic ink object outline. The indices of intersection contact point C4 are (3, 4), which correspond to fitting curve E3 above and fitting curve E4 below. Similarly, the indices of intersection contact point C5 are (5, 6), which correspond to fitting curve E5 above and fitting curve E6 below.

**[0079]** Then, for each identified fitting curve, a split point is calculated. Figure 12B illustrates how the split point is calculated for the fitting curve E3 (numeral 304). In Figure 12B, the four control points of the Bezier fitting curve 304 are marked as P1, P2, P3 and P4, where P1 and P4 are the two end points. Line 300 represents the left edge of the eraser trace.

**[0080]** The processing structure 14 first calculates a line segment 302 between the two end control points P1 and P4. Then, the intersection point

-21-

P5 of the line segments 300 and 302 is calculated. A split ratio  $t$  is calculated as:

$$t = |P1P5| / |P1P4|$$

where:

$|P1P5|$  is the length of the line segment between control points P1 and P5; and

$|P1P4|$  is the length of the line segment between control points P1 and P4.

[0081] The split point P6 on the fitting curve 304 is then calculated such that the ratio of the length of the curve P1P6 over that of the curve P1P6P4 is equal to  $t$ . In this embodiment, the BezierSplit function described in above-incorporated publication entitled "An Algorithm for Automatically Fitting Digitized Curves" authored by Philip J. Schneider, Graphics Gems, Academic Press Professional, Inc. (1990) is used as expressed by:

BezierSplit(OriginalControlPoints, ResultingLeftCurve, ResultingRightCurve,  $t$ )

where:

OriginalControlPoints are the coordinates of the original control points (input parameter);

ResultingLeftCurve is the resulting fitting curve on the left side of the split point (output parameter);

ResultingRightCurve is the resulting fitting curve on the right side of the split point (output parameter); and

$t$  is the split ratio.

Thus, the BezierSplit function not only splits the fitting curve 304, but also generates a new fitting curve (ResultingLeftCurve) between control points P1 and P6, and another fitting curve (ResultingRightCurve) between control points P6 and P4.

[0082] For each fitting curve to be split, the BezierSplit function is called. For the fitting curve on the left side of the eraser trace, the ResultingLeftCurve is kept, and the ResultingRightCurve and the original

-22-

fitting curve to be split are dropped. For the fitting curve on the right side of the eraser trace, the ResultingRightCurve is kept, and the ResultingLeftCurve and the original fitting curve to be split are dropped. Figure 12C shows the new calligraphic ink object outline after splitting, where E3', E4', E5' and E6' are the new fitting curves generated by the BezierSplit function, which are kept for using as the modified outline fitting curves.

[0083] The next step of the fast erasing methodology is to generate additional outline points (corresponding to step 224 in Figure 10). Referring to Figure 12C, for the left part of the calligraphic ink object outline, a line segment between the two split points L1 and L2 is calculated. Along this line segment, a point L3 is calculated such that  $|L1L3|=|L1L2|/4$ . Similarly, a point L4 on the line segment L1L2 is also calculated such that  $|L4L2|=|L1L2|/4$ . Then, a point L5 is calculated such that the line segment L3L5 generally perpendicularly extends from point L3 towards the center of the eraser trace and has a length of  $|L1L3|$ . Similarly, a point L6 is calculated such that the line segment L4L6 generally perpendicularly extends from point L4 towards the center of the eraser trace and has a length of  $|L4L2|$ . An additional fitting curve  $E_L$  is obtained by using the points L1, L5, L6 and L2 as the control points to close the left part of the calligraphic ink object outline.

[0084] The above procedure generates two additional points so that these two points, together with the two end points L1 and L2 of the split outline, can be directly used as the four control points to obtain the additional fitting curve  $E_L$ . However, those skilled in the art will appreciate that a different number of additional points may also be generated at different positions for generating the additional fitting curve. The actual number of additional points and the positions they are located may depend on the fitting curve used or may be arbitrary.

[0085] Two additional points R5 and R6 are also calculated for the right side of the calligraphic ink object outline in a similar manner. An additional fitting curve  $E_R$  is then obtained by using the points R1, R5, R6 and R2 as the control points to close the right part of the calligraphic ink object outline.

[0086] The rest of the fast erasing methodology is similar to steps 230 and 232 in Figure 10. In this embodiment, each unconnected portion of the calligraphic ink object after erasing is copied to a new calligraphic ink object. In the example shown in Figures 12A to 12D, the calligraphic ink object becomes two unconnected parts after erasing. Therefore, each part is copied to a separate calligraphic ink object. The original calligraphic ink object is deleted after copying each unconnected portion of the calligraphic ink object to a new calligraphic ink object. Figure 12D shows the calligraphic ink object outlines (two ink objects) after erasing.

[0087] Those skilled in the art will appreciate that the calligraphic ink generation and erasing methodologies described above that are performed by the processing structure are exemplary. Other variations and modifications are available. For example, although an elliptical shape is used as the pointer tip in the above embodiments, those skilled in the art will appreciate that other pointer tip shapes may be used, including, rectangular, square and other polygon shapes. The pointer tip may have a fixed size, or may be size-variable based on for example the pressure the user applies to the touch surface using the pointer or according to another user-selected size-varying scheme.

[0088] If a machine vision-based touch panel is employed rather than an analog resistive touch panel, the brush type may be determined from an image of the pointer tip contacting the touch surface 24 captured by the coordinate input device 12. In this case, each sampled contact point obtained at step 80 is derived from a captured pointer tip image. The centroids of the captured pointer tip images form the trajectory of the ink object.

[0089] Although embodiments have been described with reference to the drawings, those of skill in the art will appreciate that other variations and modifications from those described may be made without departing from the spirit and scope of the invention, as defined by the appended claims.

**What is claimed is:**

1. A method of generating a calligraphic ink object, comprising:  
sampling contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing;  
generating an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates;  
generating a smoothed ink envelope at least by fitting curves to points on said ink envelope; and  
drawing the smoothed ink envelope on a display thereby to generate the calligraphic ink object.
2. The method of claim 1 wherein said sampling contact coordinates further comprises processing the sampled contact coordinates to reduce the number of sampled contact coordinates used to generate the ink trajectory.
3. The method of claim 2 wherein said processing comprises discarding selected sampled contact coordinates.
4. The method of claim 3 wherein the discarded selected sampled contact coordinates are co-linear with other sampled contact coordinates.
5. The method of claim 4 wherein said processing comprises determining line segments that join consecutive pairs of sampled contact coordinates having the same slope and discarding sampled contact coordinates at intermediate locations along the length of the line segments.
6. The method of any one of claims 1 to 5 wherein the ink envelope generating is performed upon completion of writing.

-25-

7. The method of claim 6 wherein completion of writing is signaled by a pointer up event.
8. The method of claim 6 or 7 wherein said ink envelope generally completely surrounds said ink trajectory.
9. The method of any one of claims 6 to 8 wherein said smoothed ink envelope comprises curved and straight lined segments.
10. The method of any one of claims 1 to 9 wherein said ink envelope generating comprises surrounding each sampled contact coordinate along the generated ink trajectory with a pointer instance and for each consecutive pair of sampled contact coordinates, generating line segments on opposite sides of a line segment joining the sampled contact coordinates of the pair, that extend between the pointer instances surrounding the sampled contact coordinates of the pair.
11. The method of claim 10 wherein the generated line segments extending between the pointer instances surrounding the sampled contact coordinates of each pair are substantially tangential to the peripheries of the pointer instances.
12. The method of claim 10 or 11 wherein the points on said ink envelope to which the curves are fitted are positioned along the peripheries of each pointer instance.
13. The method of any one of claims 10 to 12 wherein each pointer instance has a geometrical shape.
14. The method of claim 13 wherein the geometrical shape is one of circular, elliptical, triangular, rectangular, ovate and polygonal.

-26-

15. The method of any one of claims 10 to 14 wherein generating the smoothed ink envelope comprises fitting Bezier curves to points on said ink envelope.

16. The method of any one of claims 1 to 5 wherein said ink trajectory comprises ink trajectory segments, wherein an ink envelope segment is generated for each ink trajectory segment and wherein the ink envelope segments are combined to form said ink envelope.

17. The method of claim 16 wherein said ink envelope generally completely surround the ink trajectory.

18. The method of claim 16 or 17 wherein each ink trajectory segment is represented by a subset of sampled contact coordinates.

19. The method of claim 18 wherein each ink trajectory segment is represented by sampled contact coordinates generated by said coordinate input device over a threshold period of time.

20. The method of any one of claims 16 to 19 wherein said ink envelope generating comprises for each ink trajectory segment surrounding each sampled contact coordinate along the ink trajectory segment with a pointer instance and for each consecutive pair of sampled contact coordinates, generating line segments on opposite sides of a line segment joining the sampled contact coordinates of the pair, that extend between the pointer instances surrounding the sampled contact coordinates of the pair.

21. The method of claim 20 wherein the line segments extending between the pointer instances surrounding the sampled contact coordinates of each pair are substantially tangential to the peripheries of the pointer instances.

-27-

22. The method of claim 20 or 21 wherein the points on said ink envelope to which the curves are fitted are positioned along the peripheries of each pointer instance.

23. The method of any one of claims 20 to 22 wherein each pointer instance has a geometrical shape.

24. The method of claim 23 wherein the geometrical shape is one of circular, elliptical, triangular, rectangular, ovate and polygonal.

25. The method of any one of claims 20 to 24 wherein generating a smoothed ink envelope comprises fitting Bezier curves to points on said ink envelope.

26. The method of any one of claims 1 to 25 wherein said drawing comprises drawing the smoothed ink envelope as a vectorized image.

27. The method of claim 26 further comprising coloring and/or shading the smoothed ink envelope prior to said drawing.

28. A method of generating a calligraphic ink object comprising:  
generating an ink trajectory representing writing input on a coordinate input device using an object, the ink trajectory comprising a subset of contact coordinates generated by the coordinate input device;  
generating an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and  
drawing the ink envelope on a display thereby to generate the calligraphic ink object.

29. The method of claim 28 wherein the curved line segments are Bezier curves.

-28-

30. The method of claim 28 or 29 wherein said ink envelope comprises a plurality of joined ink envelope segments, each ink envelope segment being associated with a respective ink trajectory segment.
31. The method of any one of claims 28 to 30 wherein said ink envelope resembles the boundary of the path of a geometric pointer tip following the ink trajectory.
32. The method of claim 31 wherein the shape of the pointer tip is one of circular, elliptical, triangular, rectangular, ovate and polygonal.
33. The method of any one of claims 28 to 32 wherein said drawing comprises drawing the smoothed ink envelope as a vectorized image.
34. The method of claim 33 further comprising coloring and/or shading the smoothed ink envelope prior to said drawing.
35. An apparatus comprising:  
at least one processor; and  
memory storing a calligraphic ink object generating routine, the calligraphic ink object generating routine, when executed by the at least one processor, causing the apparatus to:  
sample contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing;  
generate an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates;  
generate a smoothed ink envelope at least by fitting curves to points on said ink envelope; and  
draw the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

-29-

36. An apparatus comprising:  
at least one processor; and  
memory storing a calligraphic ink object generating routine, the calligraphic ink object generating routine, when executed by the at least one processor, causing the apparatus to:  
generate an ink trajectory representing writing input on a coordinate input device using an object, the ink trajectory comprising a subset of contact coordinates generated by the coordinate input device;  
generate an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and  
draw the ink envelope on a display thereby to generate the calligraphic ink object.
37. An apparatus comprising at least one processor and memory storing calligraphic ink object generating instructions, which when executed by the at least one processor cause the apparatus to carry out the method according to any one of claims 1 to 34.
38. A computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to:  
sample contact coordinates generated by a coordinate input device during writing thereon using a pointer to generate an ink trajectory generally representing the writing;  
generate an ink envelope, said ink envelope comprising line segments joining pointer instances at said sampled contact coordinates;  
generate a smoothed ink envelope at least by fitting curves to points on said ink envelope; and  
draw the smoothed ink envelope on a display thereby to generate the calligraphic ink object.

-30-

39. A computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to:

generate an ink trajectory representing writing input on a coordinate input device using an object, the ink trajectory comprising a subset of contact coordinates generated by the coordinate input device;

generate an ink envelope surrounding the ink trajectory, the ink envelope being represented by curved and straight line segments; and

draw the ink envelope on a display thereby to generate the calligraphic ink object.

40. A method of erasing at least a portion of a calligraphic ink object delineated by an eraser trace intersecting said calligraphic ink object, said method comprising:

calculating intersection points between said calligraphic ink object and said eraser trace;

generating additional points along line segments that extend between the intersection points on opposite sides of the eraser trace;

fitting curves to the intersection points and the additional points;

modifying the outline of said calligraphic ink object using said curves; and

deleting the portion of said calligraphic ink object between said curves.

41. The method of claim 40 further comprising:

creating at least one replacement calligraphic ink object following said deleting.

42. The method of claim 40 or 41, wherein calculating the intersection points comprises:

calculating intersection points between an ink trajectory of said calligraphic ink object and the eraser trace;

-31-

identifying piecewise curves on said outline related to each of said intersection points on said ink trajectory; and  
calculating intersection points on said identified piecewise curves.

43. An apparatus comprising:  
at least one processor; and  
memory storing a calligraphic ink object erasing routine, the calligraphic ink object erasing routine, when executed by the at least one processor, causing the apparatus to:  
calculate intersection points between said calligraphic ink object and said eraser trace;  
generate additional points along line segments that extend between the intersection points on opposite sides of the eraser trace;  
fit curves to the intersection points and the additional points;  
modify the outline of said calligraphic ink object using said curves; and  
delete the portion of said calligraphic ink object between said curves.

44. A computer readable medium embodying computer program code, which when executed by at least one processor, causes an apparatus to:  
calculate intersection points between said calligraphic ink object and said eraser trace;  
generate additional points along line segments that extend between the intersection points on opposite sides of the eraser trace;  
fit curves to the intersection points and the additional points;  
modify the outline of said calligraphic ink object using said curves; and

-32-

delete the portion of said calligraphic ink object between said curves.

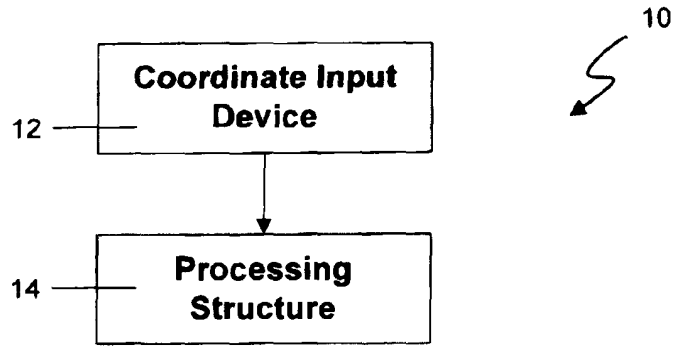


Fig. 1

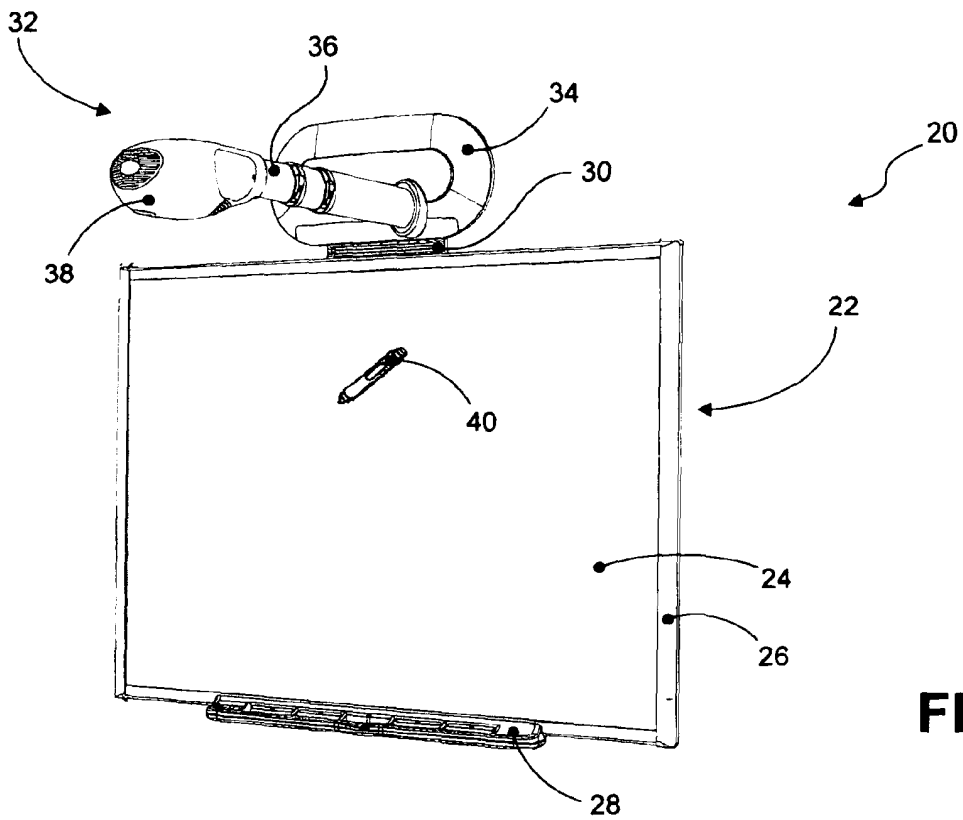


FIG. 2

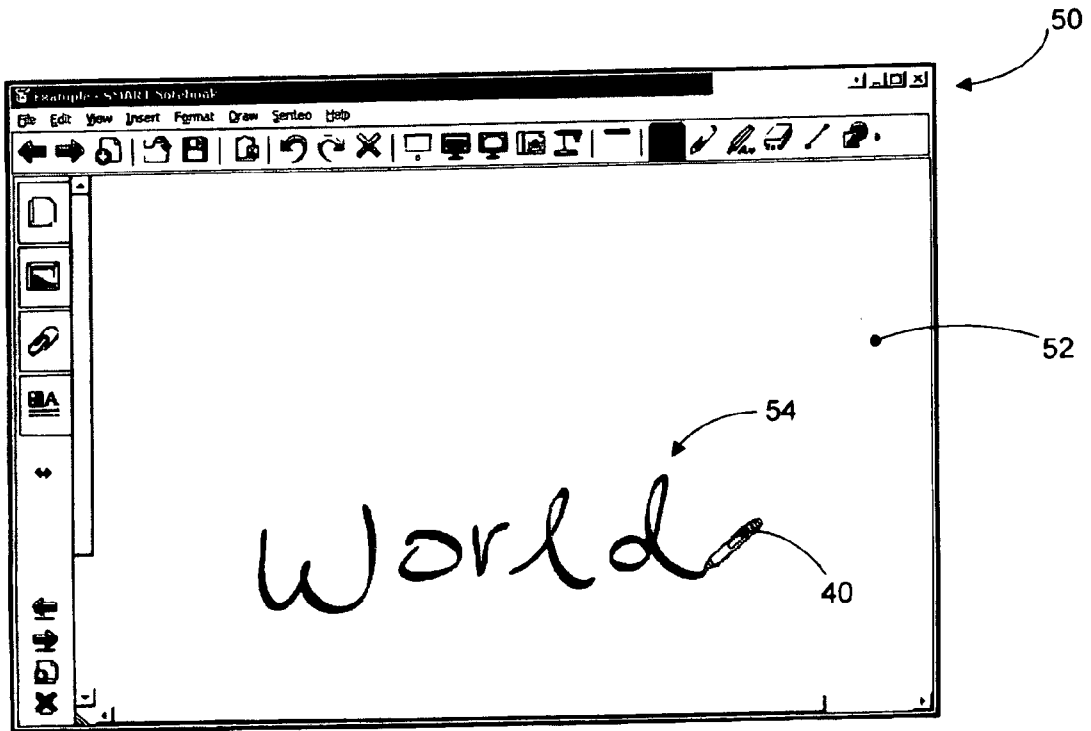


FIG. 3A

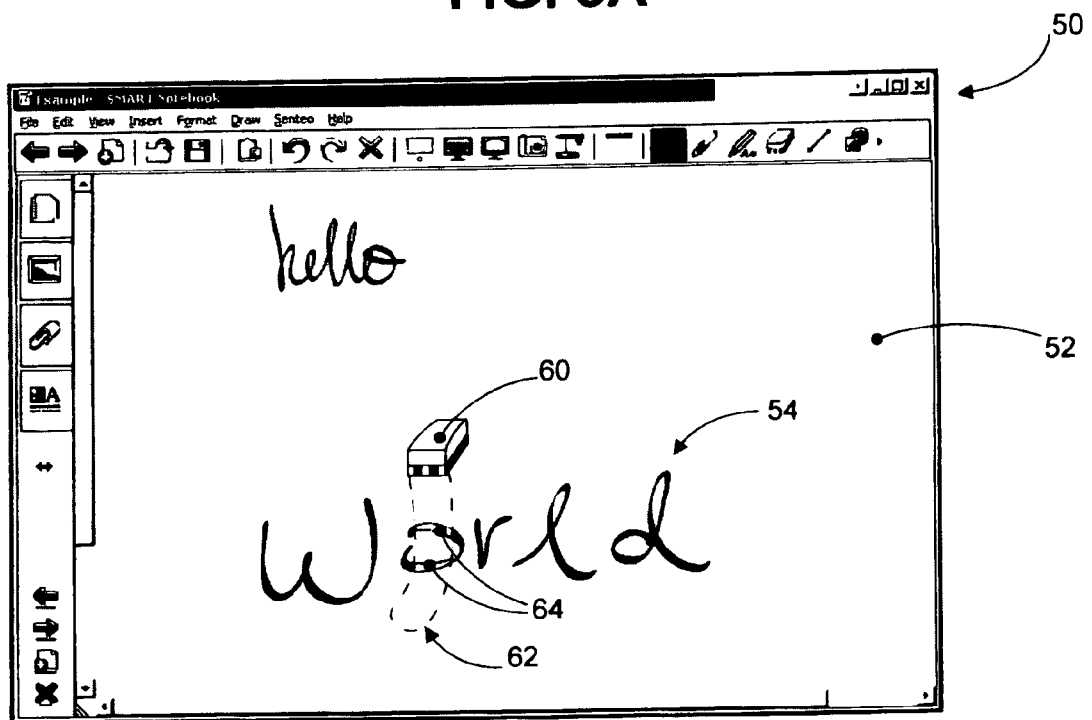
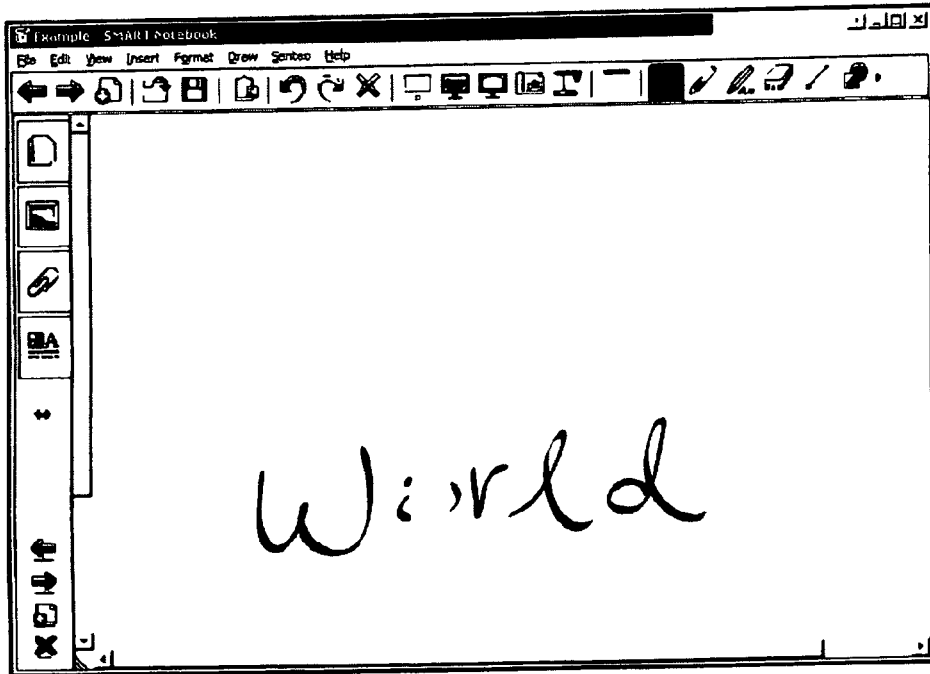
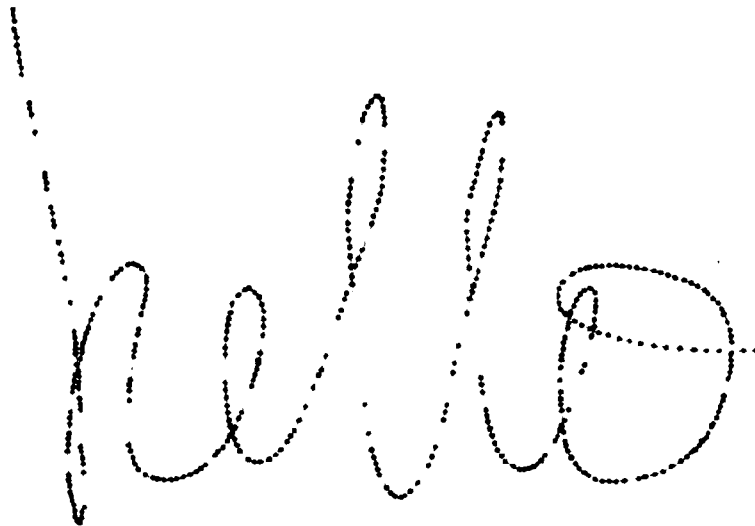


FIG. 3B

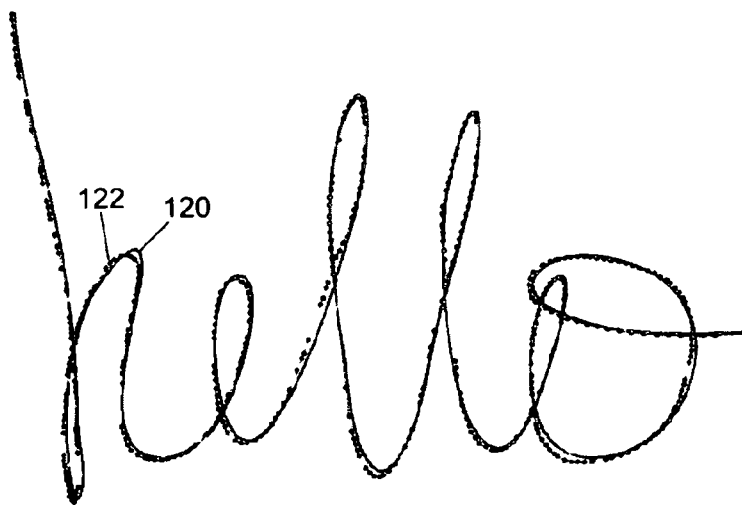
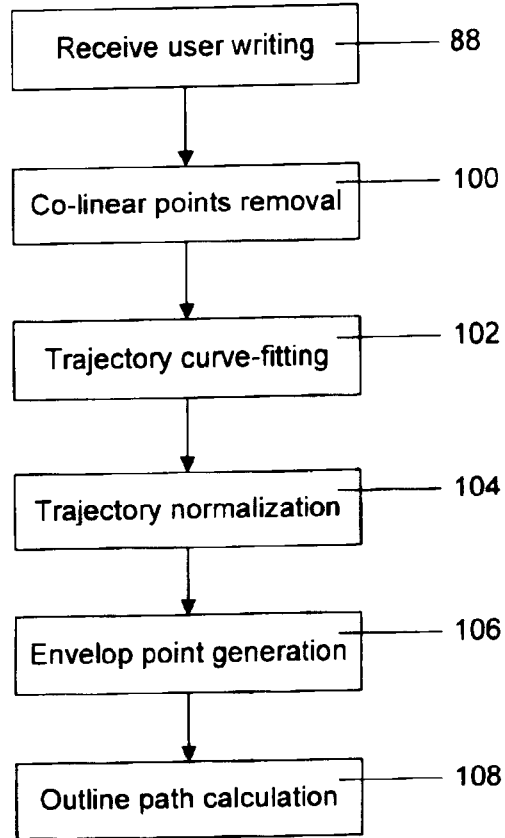


**FIG. 3C**

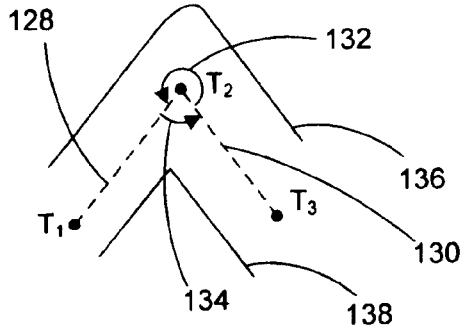


**Fig. 5**

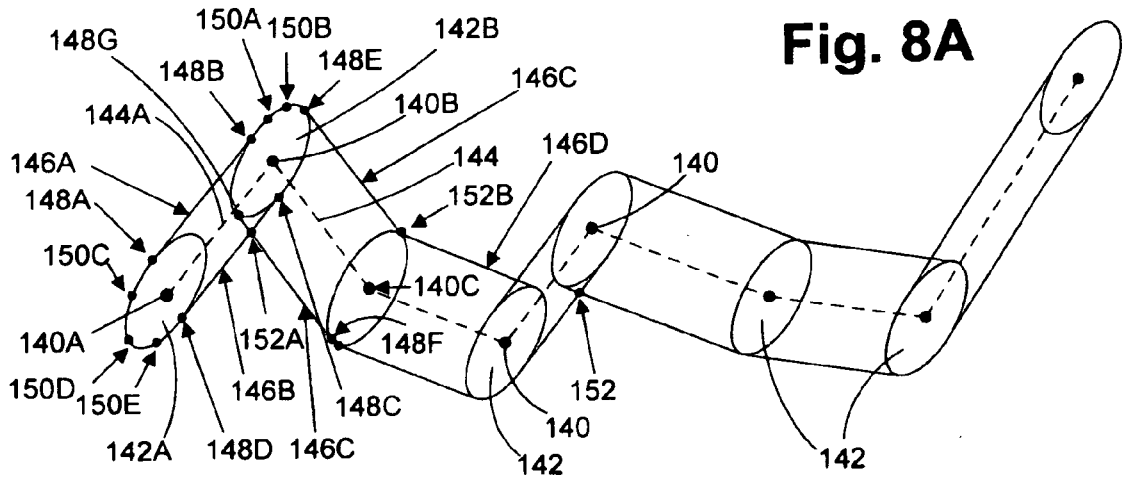
**Fig. 4**



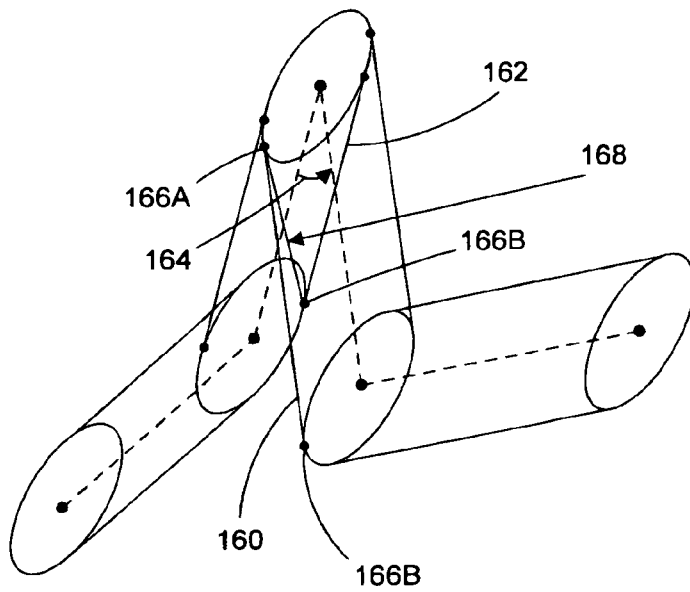
**Fig. 6**



**Fig. 7**



**Fig. 8A**



**Fig. 8B**

Fig. 8C

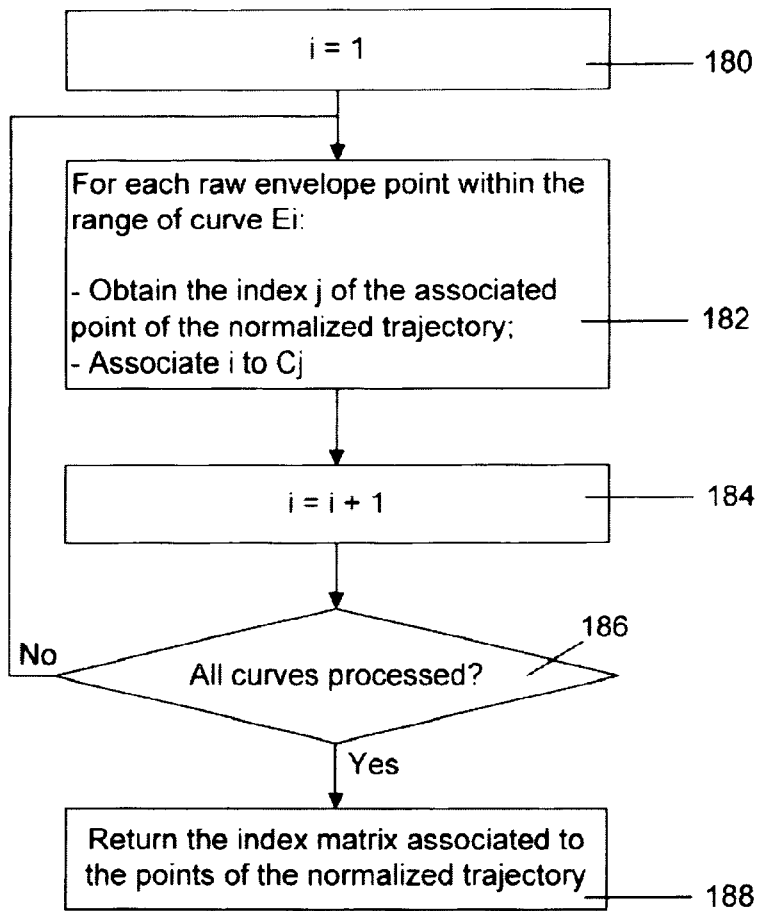
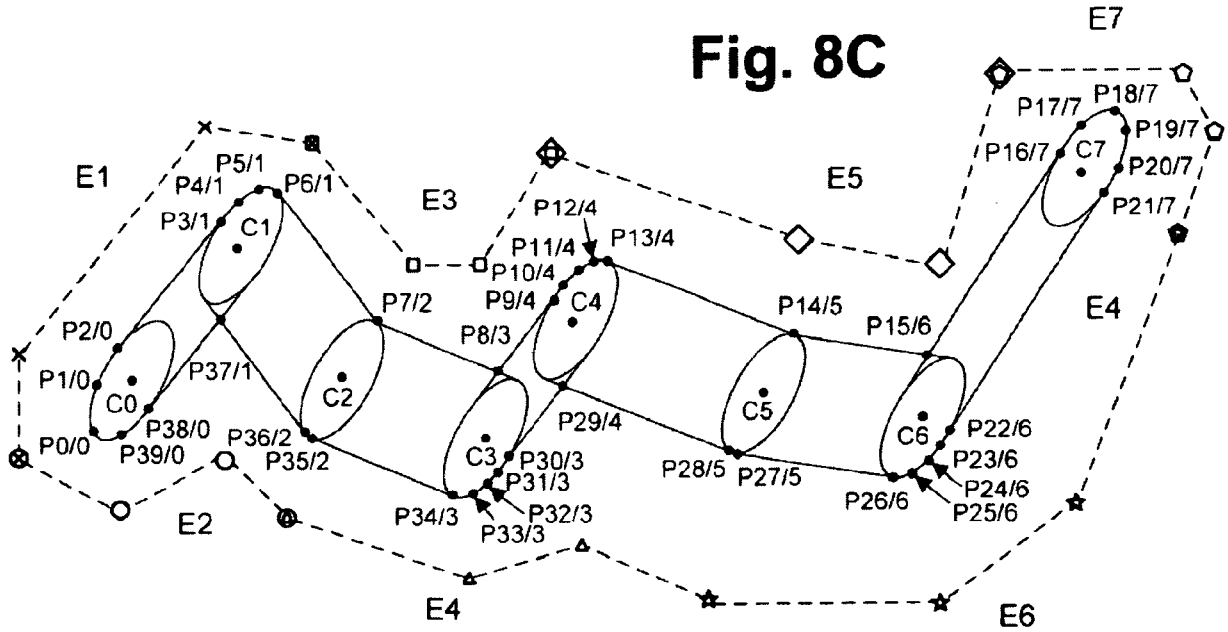


FIG. 8D

Fig. 8E

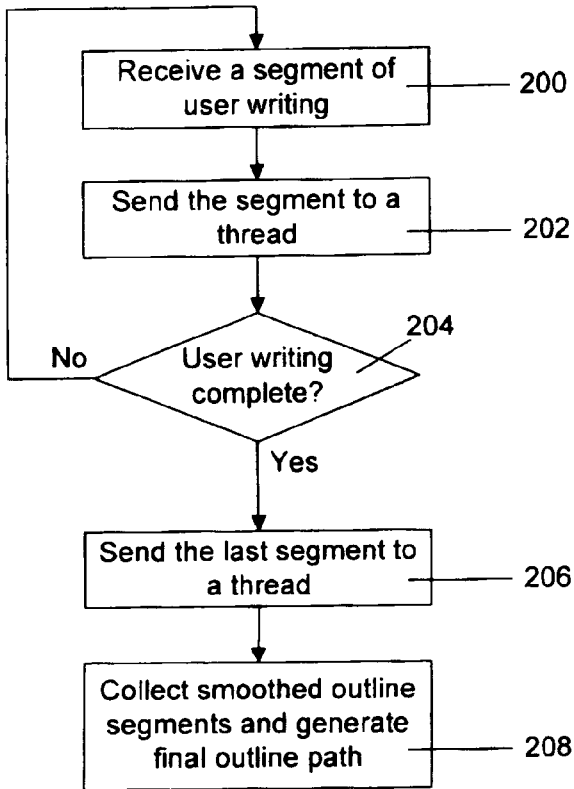
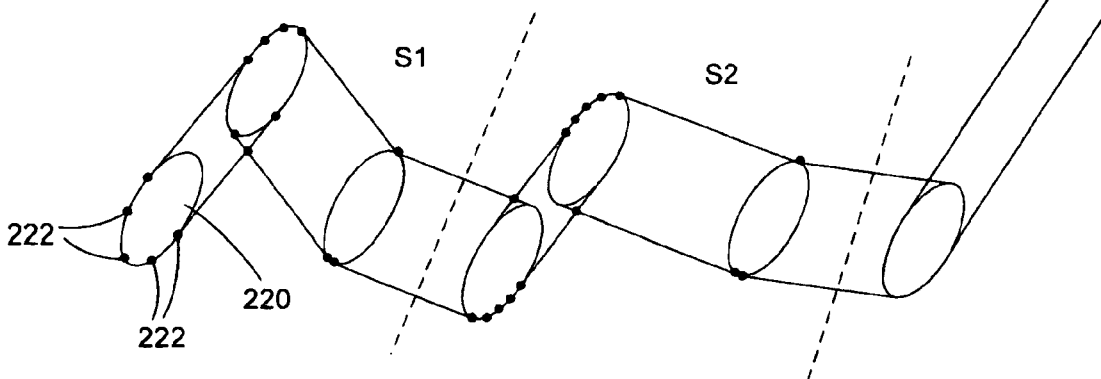
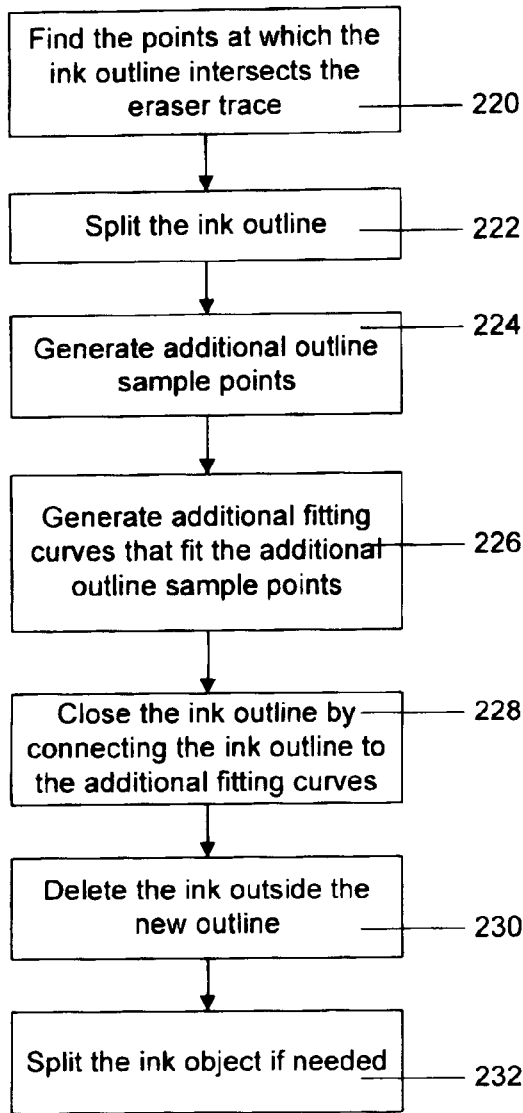
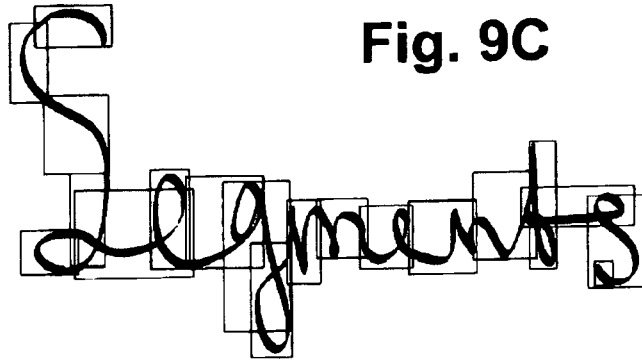


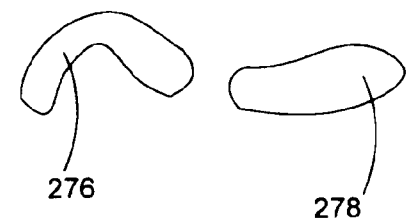
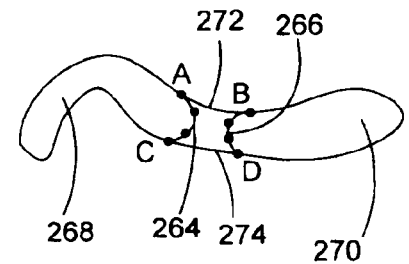
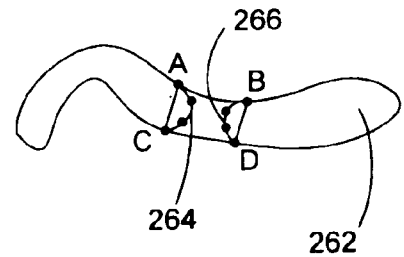
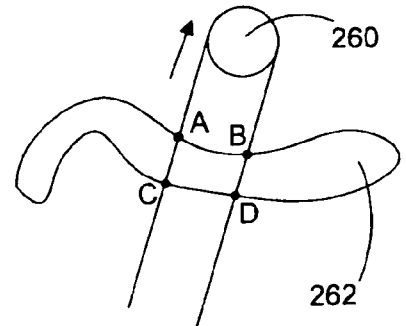
FIG. 9A

Fig. 9B



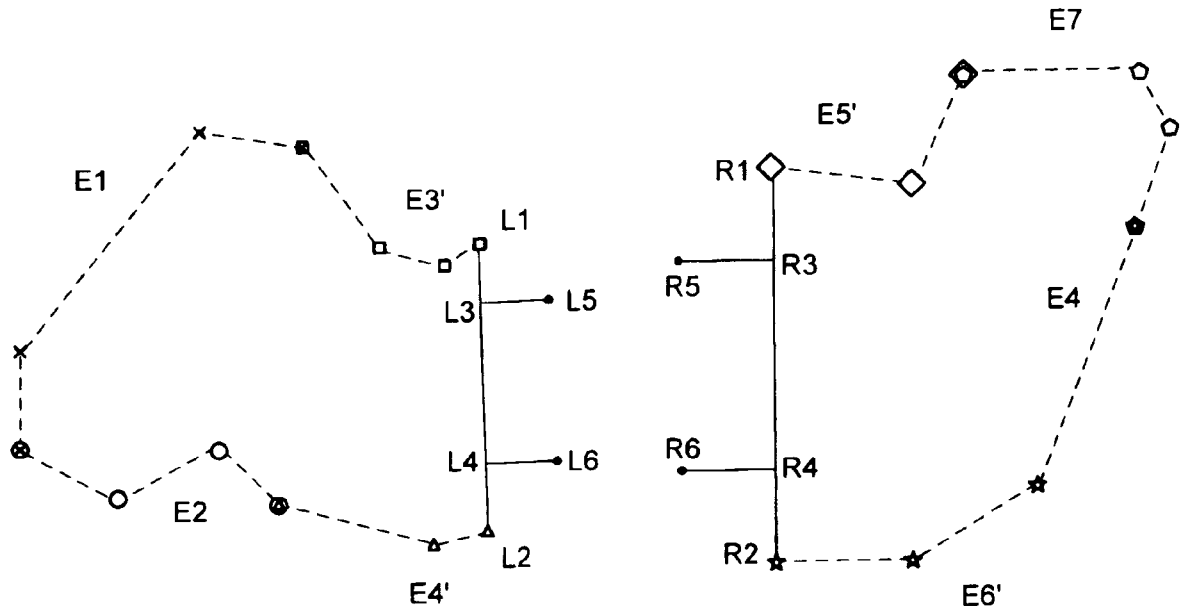


**Fig. 10**

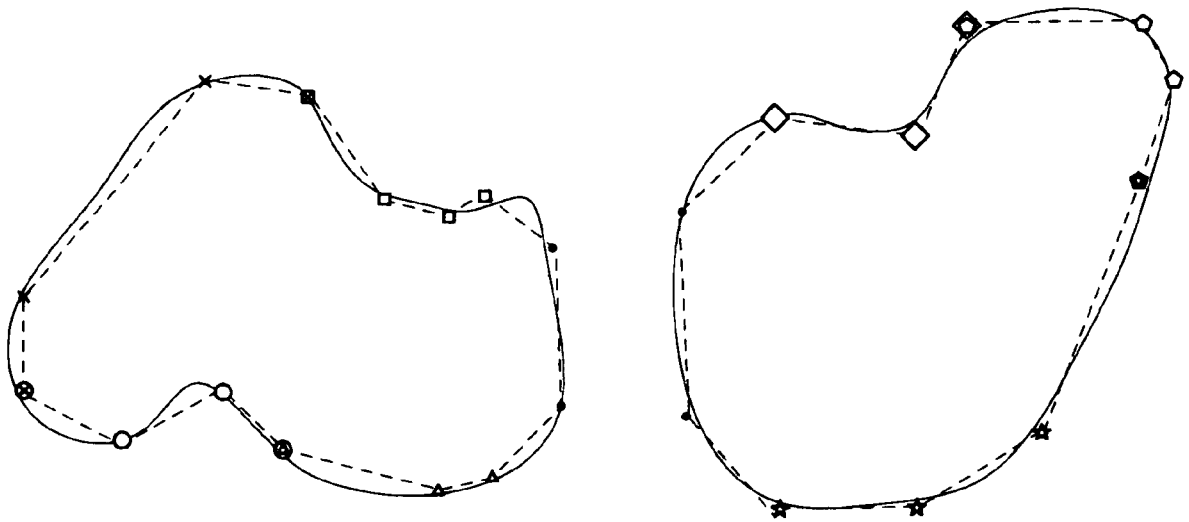


**Fig. 11D**





**Fig. 12C**



**Fig. 12D**

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/CA2010/001589

A. CLASSIFICATION OF SUBJECT MATTER IPC: <b>G06F 3/03</b> (2006.01) , <b>G06F 3/14</b> (2006.01) According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) <b>G06F 3/14</b>		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used) -Canadian Patents Database, EPOQUE, TotalPatent, Internet Database search engines -electronic display surface, interactive white board [IWB], cursive/ calligraphic writing style, ink/ print trajectory segment computation/ calculation, ink/ script segments, sampled contact coordinates, multitouch interactive whiteboard/ surface, hand-drawn/ hand-written, s/w recognition systems for sketching/ writing		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	“Creation, Presentation, Capture, and Replay of Freehand Writing in e-Lecture Scenarios” ( <i>L. Belenkaia et al.</i> ) 2004 Paper presented at the ED-MEDIA World Conference on Educational Multimedia, Hypermedia & Telecommunications, accessed from the internet on 10 November 2010 at: <a href="http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.93.5398">http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.93.5398</a> *page 4, Section 3; Table 1 on page 6-7; page 7, Section 4 paragraph 3; page 8, paragraph 1; page 8, Section 4, paragraph 2; page 9, paragraphs 4-6; page 9, subsection 5.1, pages 10- 12, section 5.2 ; page 13, subsection 6.1;page 13, subsection 6.1; page 17, Appendix 1, page 24; Appendix 2, pages 26 and 29; Appendix 3	1-44
Y	US2009245654A1, ( <i>Xing et al.</i> ) 01 October 2009 (01-10-2009) *page 2, paragraph [0046]; Figures 3b, 5a-5d, 8, 11a, 11b, 12a and 12b	1-44
A	US6952803B1, ( <i>Bloomberg et al.</i> ) 04 October 2005 (04-10-2005) *see whole document*	1-44
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents :	“T”	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
“A” document defining the general state of the art which is not considered to be of particular relevance	“X”	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
“E” earlier application or patent but published on or after the international filing date	“Y”	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	“&”	document member of the same patent family
“O” document referring to an oral disclosure, use, exhibition or other means		
“P” document published prior to the international filing date but later than the priority date claimed		
Date of the actual completion of the international search 29 November 2010 (29.11.2010)	Date of mailing of the international search report 11 January 2011 (11-01-2011)	
Name and mailing address of the ISA/CA Canadian Intellectual Property Office Place du Portage I, C114 - 1st Floor, Box PCT 50 Victoria Street Gatineau, Quebec K1A 0C9 Facsimile No.: 001-819-953-2476	Authorized officer  Tanya Novo-Verde (819) 934-4891	

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/CA2010/001589

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US7224847B2, ( <i>Zhang et al.</i> ) 29 May 2007 (29-05-2007) *see whole document*	1-44
P, A	“METRICS AND NEATENING OF HANDWRITTEN CHARACTERS” ( <i>Maria Teresa Infante Velazquez</i> ) 29 June 2010 Master of Computer Science Thesis at The University of Western Ontario, accessed from the internet on 10 November 2010 at: <a href="http://www.csd.uwo.ca/~watt/home/students/theses/MTInfanteVelazquez2010-msc.pdf">http://www.csd.uwo.ca/~watt/home/students/theses/MTInfanteVelazquez2010-msc.pdf</a> *see whole document*	1-44
A	“Add Support for Digital Ink to Your Windows Applications” ( <i>Paul Yao</i> ) MSDN Magazine 2004 accessed from the internet on 10 November 2010 at: <a href="http://msdn.microsoft.com/en-us/magazine/cc163869.aspx">http://msdn.microsoft.com/en-us/magazine/cc163869.aspx</a> *see whole document*	1-44
A	“Digital Ink, Breakthrough Technology in Tablet PC, Brings the Power of the Pen to the Desktop” ( <i>Microsoft News Center</i> ) 29 October 2002 accessed from the internet on 15 November 2010 at: <a href="http://www.microsoft.com/presspass/features/2002/oct02/10-29tabletinking.msp">http://www.microsoft.com/presspass/features/2002/oct02/10-29tabletinking.msp</a> *see whole document*	1-44

**INTERNATIONAL SEARCH REPORT**  
Information on patent family members

International application No.  
**PCT/CA2010/001589**

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
US2009245654A1	01 October 2009 (01-10-2009)	AU2009227934A1 CA2718636A1 US2009245645A1 WO2009117810A1	01 October 2009 (01-10-2009) 01 October 2009 (01-10-2009) 01 October 2009 (01-10-2009) 01 October 2009 (01-10-2009)
US6952803B1	04 October 2005 (04-10-2005)	None	
US7224847B2	29 May 2007 (29-05-2007)	CN1525300A CN1324444C EP1460851A2 JP2004260823A JP4482348B2 KR20040076216A US2004165768A1	01 September 2004 (01-09-2004) 04 July 2007 (04-07-2007) 22 September 2004 (22-09-2004) 16 September 2004 (16-09-2004) 16 June 2010 (16-06-2010) 31 August 2004 (31-08-2004) 26 August 2004 (26-08-2004)