(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
29 July 2004 (29.07.2004)

PCT

(10) International Publication Number
WO 2004/063862 A2

(51) International Patent Classification[7]: G06F

(21) International Application Number:
PCT/US2004/000162

(22) International Filing Date: 7 January 2004 (07.01.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/438,499	8 January 2003 (08.01.2003)	US
10/748,685	31 December 2003 (31.12.2003)	US
10/748,686	31 December 2003 (31.12.2003)	US
10/748,684	31 December 2003 (31.12.2003)	US

(71) Applicant *(for all designated States except US)*: **SILICON GRAPHICS, INC.** [US/US]; 1500 Crittenden Lane, M/S 710, Mountain View, CA 94043 (US).

(72) Inventors: **FITZMAURICE, George**; 36 Metcalfe St., Apt. 1, Toronto, Ontario M4X 1R8 (CA). **KURTENBACH, Gordon**; 126 Spruce Street, Toronto, Ontario M5A 2J5 (CA). **MILLER, Lynn**; 567 Broadway Avenue, Toronto, Ontario M6G 2S2 (CA). **DI VITTORIO, Joe**; 17 Fairfield Road, Toronto, Ontario M4P 1S9 (CA).

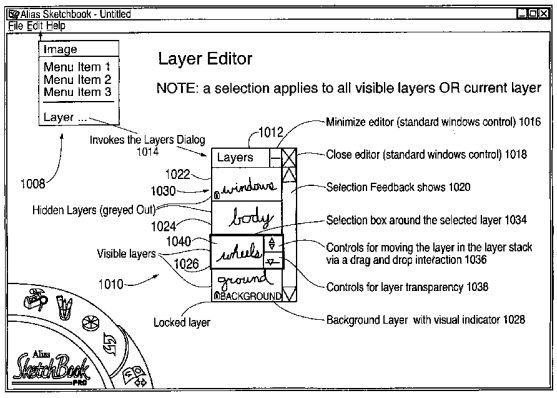
(74) Agent: **BECKERS, J., Randall**; Staas & Halsey LLP, 1201 New York Avenue, NW, Suite 700, Washington, DC 20005 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR,

[Continued on next page]

(54) Title: USER INTERFACE TECHNIQUES FOR PEN-BASED COMPUTERS



(57) Abstract: The present invention is a system that provides a layer editor representing layers using box like controls. The layer controls are arranged in vertical stack representing the ordering of the layers in the paint application and provided with a highlight frame that indicates an active layer. The name or graphic of each layer depicted in the corresponding box for that layer can be created using drawing strokes of a stylus or pen of a pen-based computer. The pen is also used to select the controls, pop-up menus and perform selections or operations with underlying menus and/or functions. Making a mark with the pen can be used to select layer editing functions. A drawing dialog is used to write or draw the names applied to the layers. The present invention is a system that positions an interface for a pen-based computer at an end of a natural user motion arc, such as an arc a hand travels when an elbow is pivoted. Positioning of the interface in a lower corner of a display or window (left corner for a righthanded person and right corner for a left-handed person) allows easy approach when the user is working in the center of the display, such as when drawing on the pen-based computer display. An arc or curved interface improves approach zones and reduces unintended selections of controls or buttons within the interface. The arc allows the natural motion to be extended by wrist or finger motions to access all the controls in the corner command interface. Controls on the arc are placed to allow any underlying pop-up menus to be completely accessible when an activated. Overflow menu items of activated controls of the interface are located in conformity to the natural arc. The present invention is a system that uses natural user position and natural user motion to position and layout interface elements for a pen-based computer display. Graphical user interfaces, such as a slider or menu, are popped-up at a position convenient to the user, such as at the current position of the cursor. A rectilinear interface is oriented along a natural motion arc of the user, such as an elbow arc. An arc shaped interface can also be positioned along a natural motion arc, such as the elbow arc, and be shaped according the elbow are or be shaped by another natural motion arc such as a wrist arc of the user. The interface arc, whether shaping or orienting the interface, can be a single motion arc, such as an elbow arc, a composite arc of an elbow arc and a wrist arc, a sequence of an elbow arc and a wrist arc, a compound arc where an elbow arc blends into a wrist arc or an arc followed by a linear interface section.

WO 2004/063862 A2

GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

TITLE OF THE INVENTION

USER INTERFACE TECHNIQUES FOR PEN-BASED COMPUTERS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is related to and claims priority to U.S. provisional application entitled "User Interface Techniques For Pen-Based Computers" having serial number 60/438,499 (S&H Docket 1252.1083P), by Fitzmaurice et al, filed January 8, 2003, this application is also related to and claims priority to U.S. application entitled "A Layer Editor System For A Pen-Based Computer" having serial number **(1252.1083), by Fitzmaurice et al, filed December 31, 2003, this application is also related to and claims priority to U.S. application entitled "A User Interface Having A Placement And Layout Suitable For Pen-Based Computers" having serial number **(S&H Docket 1252.1087), by Fitzmaurice et al, filed December 31, 2003, and this application is also related to and claims priority to U.S. application entitled "Biomechanical User Interface Elements For Pen-Based Computers" having serial number **(S&H Docket 1252.1088), by Fitzmaurice et al, filed December 31, 2003, and all of which are incorporated by reference herein.

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0002] The present invention is directed to a layer editor for a pen-based computer and, more particularly, to system where user identification of layers does not require text input, commands can be applied to the layers using a marking menu approach and where the invocation of the marking menus is facilitated by large "hit areas" for the menus in the layer editor.

[0003] The present invention is directed to an interface positioned, typically in a bottom corner of a display or window of a pen-based computer, to allow a natural motion approach to controls of the interface and, more particularly, to an interface that allows a natural arc path approach with expanded approach zones, reduced selection interference and menu pop-ups where all commands can be accessed.

[0004] The present invention is directed to interface elements oriented and/or shaped in accordance with a natural motion of a user, such as conforming to an arc made by the hand when the elbow is pivoted and, more particularly, to a system that positions interface elements

at a position determined by the hand position of the user, allowing the interface to pop-up where it is comfortable to use and lays out the interface elements according to a natural arc that is also easy to use.

2. Description of the Related Art

[0005] Pen or stylus based computers, such as tablet personal computers (PCs) and personal digital assistants (PDAs) are becoming popular. These computers typically allow the user to interact with the computer through a graphical user interface using a stylus or pen (or possibly even a finger). A keyboard and a mouse are typically not available. Pen-based type interaction can be limiting as compared to other types of interaction, such as using the mouse or the keyboard, alone or in combination. What is needed are improvements to the interaction capability of pen-based computers.

[0006] User interfaces - UI (layer editors) providing access to the layer functionality of paint systems are designed for mouse based user interfaces. Layer editors provide a representation of each paint layer that can be manipulated or edited. Typical examples are the layer editor in PhotoShop or the layer editor in "Fractal paint". These types of UI present difficulties when operated on pen-based computers. Specifically, the naming of layers in a traditional UI requires text input and this is hard to do with a pen. Also, buttons and widgets on a traditional UI can be too small to accurately hit with a pen. What is needed is a system that will allow writing based naming of layers and controls that can be more easily targeted with a pen or stylus.

[0007] In addition, most keyboard and mouse based user layer editor interfaces use hot-keys to allow a user to quickly access layer commands that are normally in menu-items on the menu bar or on pop-up menus. With a pen based user interface no keyboard may be available or the user may prefer not to use the keyboard. What is needed is to provide pen-based computers fast access capability somewhat equivalent to hot key capability.

[0008] Operating pen-based user interfaces in the tablet-PC form factor can be extremely awkward and uncomfortable. Repetitive actions will be common and operating the user interface along the borders of the screen is problematic. First, the digitizer behaves poorly around the borders of the screen and thus it is hard to control the cursor using the stylus. Secondly, positioning the hand along the digitizer edge can be uncomfortable and unstable to operate with precision. What is needed are solutions that employ a design that moves away from the digitizer edge and adhere to comfortable human motions will be valuable.

[0009] Typical interface elements, such as a slider, are rectilinearly positioned, such as located horizontally or vertically relative to the display, often times along the sides of the display, such as a top of the display menu bar. Using such interface elements with a pen typically requires a compound movement by the user that may include rotating at the shoulder to essential slide the elbow, pivoting the elbow during the sliding motions and rotating the wrist. Such compound movements are more difficult that simple movements, such as pivoting of the elbow, at can cause fatigue in the user when receptively done. What is needed is an interface element that conforms to more natural simple motions by the user.

[0010] Typical interface elements, such as a slider or a menu, are rectilinearly oriented, such as oriented horizontally and/or vertically relative to the display. Using such interface elements with a pen typically requires a compound movement by the user that may include rotating at the shoulder to essentially slide the elbow, pivoting the elbow during the sliding motion and rotating the wrist at the same time. Such compound movements are more difficult than simple movements, such as pivoting at the elbow, and can cause fatigue in the user when receptively done. What is needed is an interface element that conforms to more natural simple motions by the user.

SUMMARY OF THE INVENTION

[0011] It is an aspect of the present invention to provide a layer editor where user identification of layers does not require text input.

[0012] It is another aspect of the present invention to allow commands to be applied to the layers using marking menu technology.

[0013] It is an additional aspect of the present invention to provide large hit areas or zones for the controls of the layers.

[0014] It is an aspect of the present invention to provide an interface at a location that allows a natural motion to approach the interface.

[0015] It is another aspect of the present invention to provide an interface approach with expanded approach zones and reduced selection interference.

[0016] It is an additional aspect of the present invention to provide interface elements at a comfortable and convenient position for a user of a pen-based computer.

[0017] It is another aspect of the present invention to provide interface elements that are oriented and shaped by natural motion arcs of the users hand, elbow and/or shoulder.

[0018] It is a further aspect of the present invention to layout or design interface elements that take advantage of natural use motions.

[0019] The above aspects can be attained by a system that provides a layer editor representing drawing layers using box like controls where the name of each layer depicted in the box for that layer can be created using writing strokes of a stylus or pen of a pen-based computer. The pen is also used to select the controls, pop-up menus and perform marking selections or operations with underlying menus or functions. A drawing dialog is used to write the names applied to the layers.

[0020] The above aspects can be attained by a system that positions the interface at an end of a natural user motion arc, such as an arc a hand travels when an elbow is pivoted. A position of the interface in a lower corner of a display or window allows easy natural arc motion, approach when the user is working in a center of the display, such as when drawing on a pen-based computer display. An arc or curved interface in the corner location improves approach zones and reduces unintended selections. The arc also positions controls away from a display edge.

[0021] The above aspects can be attained by a system that uses natural user position and natural user motion to position and layout interface elements for a pen-based computer display. Graphical user interfaces, such as a slider or menu, are popped-up at a position convenient to the user, such as at the current position of the cursor. The interface is oriented along or shaped like a natural motion arc of the user. The interface arc can be a single motion arc, such as an elbow arc, a composite arc of an elbow arc and a wrist arc, a sequence of an elbow arc and a wrist arc, a compound where an elbow arc blends into a wrist arc or an arc followed by a linear interface. The arc is set by allowing a user to draw strokes on a display and the arc is determined based on the strokes.

[0022] These together with other aspects and advantages which will be subsequently apparent, reside in the details of construction and operation as more fully hereinafter described and claimed, reference being had to the accompanying drawings forming a part hereof, wherein like numerals refer to like parts throughout.

BRIEF DESCRIPTION OF THE DRAWINGS

[0023] Figure 1001 illustrates a layer editor user interface (UI) of the present invention.

[0024] Figure 1002 depicts marking menus of the layer editor of the present invention.

[0025] Figure 1003 depicts a drawing dialog box of the present invention used for naming the layers of the layer editor of the present invention.

[0026] Figure 1004 depicts the hardware of the present invention.

[0027] Figure 1005 shows the operations of the present invention.

[0028] Figure 1006 illustrates the preferred data structure of the present invention.

[0029] Figures 1007 - 1013 depict the components of the UI in more detail.

[0030] Figures 1014 - 1021 illustrate additional embodiments of the layer editor of the present invention.

[0031] Figure 1022 illustrates using the present invention for a slide presentation editor.

[0032] Figures 1023 - 1025 depict using gestures for layer editing.

[0033] Figures 2001 and 2002 illustrate natural arcs and an interface layout that takes advantage of the natural arcs.

[0034] Figure 2003 depicts a preferred interface for a drawing application.

[0035] Figure 2004 depicts hardware of the present invention.

[0036] Figure 2005 depicts additional locational and arrangement details of the drawing interface.

[0037] Figures 2006 – 2015 depict details of underlying marking menus

[0038] Figures 2016 – 2031 depict additional embodiments of the interface of the present invention.

[0039] Figures 2032 and 2033 illustrate operations of the present invention.

[0040] Figures 2034A – 2036B illustrate design principles of an interface according to the present invention.

[0041] Figure 3001 illustrates right-hand elbow and wrist arcs made by moving a person hand.

[0042] Figure 3002 depicts the elbow and wrist arc for a left-handed user.

[0043] Figure 3003 depicts arcs for a standing user.

[0044] Figure 3004 depicts hard ware of the present invention.

[0045] Figure 3005 shows elbow arc drawing strokes.

[0046] Figure 3006 depicts a linear interface element oriented to a stoke arc.

[0047] Figure 3007 shows an arc shaped interface oriented to the stroke arc.

[0048] Figure 3008 shows a multiple control interface oriented to a stroke arc.

[0049] Figure 3009 shows multiple control elements staggered along an arc stroke.

[0050] Figure 3010 shows arc shaped multiple controls.

[0051] Figures 3011A and 3011B show pop-up menus with rectilinear and arc shaped elements.

[0052] Figure 3012 shows arc influenced control zones.

[0053] Figure 3013 shows arranging interface elements along an arc with the most often used element under the cursor.

[0054] Figure 3014 shows an arc shaped interface that can be reoriented.

[0055] Figure 3015 shows arc shaped corner targets.

[0056] Figure 3016 depicts an arc influenced overflow palette.

[0057] Figure 3017 shows elements along an elbow arc and interface element wrist arc shaped.

[0058] Figures 3018 and 3019 show elements along an arc with different depths and widths.

[0059] Figure 3020 shows multiple arc selection.

[0060] Figures 3021 and 3022 show combinations of rectilinear menus with arc shaped menus.

[0061] Figure 3023 shows a menus with an arc shaped portion and a linear portion.

[0062] Figure 3024 depicts the operations of the present invention.

[0063] Figure 3025 shows arc that can be combined.

[0064] Figure 3026 shows an alternative positioning of a linear arc oriented interface.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0065] The present invention is directed to a layer editor system that enables users to avoid text input by drawing the names of layers by hand with the pen on the layers represented in the layer editor. The invention also provides larger hit zones or areas so button/control operation with a pen is easier. Command engagement is further facilitated by the use of marking menus on each layer representation to make command engagement very easy with a pen.

[0066] The layer editor 1010 of the present invention, as depicted in figure 1001, can be displayed on the display screen of a pen-based or other type of computer using any common method of command invocation. Typically this is done from a "Layers Editor" menu-item in the menu bar. Figure 1001 shows an example where the layers editor 1010 is displayed by selecting, for example, Edit-> Layers in the menu-bar or pop-up menu 1008. Invoking the layer editor 1010 can be done in many different ways, such as with a special button on the screen, or by pressing a key on the keyboard.

[0067] The layer editor 1010 is a graphical user interface that preferably includes a box like graphic 1012 that encloses the different controls, etc. of the editor 1010. To make the layer editor 1010 easier to use with a pen or stylus, the controls in the layer editor 1010 should be relatively large or present relatively large targets for the pen user to hit. Large controls, in the context of a pen or stylus, relate to the accuracy of the pointing device. Typically, pointing with a pen moves the cursor to that location +/- e, where e is the distance "error accuracy of the pen". Preferably, a target has to be at least 2e in size to assure that if the user does point to the center of the target the cursor will be within the target. With current tablet pc pens or styli is about 1-2mm, so targets are preferably at least 4 mm high and wide. This of course is an extreme minimum: 10mm is much more acceptable and preferable. In addition, "Fitts Law" of pointing indicates that the ease of pointing is proportional to the width of the target, so bigger is generally better. As a result, we prefer that a layer editor button or control have a target size of about 20 mm (10e) for minimum height and width.

[0068] When the layer editor 1010 is displayed it preferably has many of the common window features typically found in modern graphical user interfaces. These features include a title bar 1014 that allows a user to invoke a layers dialog. A minimize button or control 1016 that allows the editor 1010 to be minimized. A close button 1018 closes the editor 1010. A scrollbar 1020 allows the different layers depicted by the editor 1010 to be displayed where the typical editor

displays five layers and can be scrolled to display all of the layers. These controls may be enlarged as needed to allow easier operation with a pen.

[0069] The present invention provides for the use of boxes displayed to represent the ordering of the drawing layers in a paint application. Figure 1001 shows a bottom to top ordering of four layers 1022-1028 where the background layer 1028, hand letter named "ground", is at the bottom of the layer editor and subsequent layers are "layered on top over" each other. This vertical ordering represents the underlying ordering of the data structure, such as a pointer directed list implementing a layer stack, that maintains the layers for the particular drawing. The stack entries also contain the parameter data for each of the layers, such as locked/unlocked status, hidden/visible status, layer name bitmap, active, etc. Each layer can be a control that allows a function to be performed, such as display the layer in a one-shot type function or display a menu of operations that can be performed on the layer in a menu access type multiple operation type function. Each layer box displays its name and some state information about the layer (a layer may be hidden or visible, selected or unselected, locked, etc.). For example, layer 1024 is the named the "body" layer and is a hidden layer indicated by using a gray overlay to indicate the hidden status. Layer 1022 is named the "windows" layer and is also a hidden layer, but it is locked as depicted by the lock icon 1030. This icon 1030 is typically not a control because it is a very small target; however, the icon 1030 could be a control providing a toggle between locked and unlocked. A locked layer cannot be moved. The background layer 1028 is a visible layer, as indicated by the transparent overlay indicator and is also locked. The background layer 1028 also preferably has a visual indicator 1032, in this case the text "BACKGROUND", indicating that the layer is the background layer. Layer 1026 is named the "wheels" layer. Layer 1026 is a visible layer. Layer 1026 is shown as having been selected by the user by the selection indication frame graphic 1032 that has been placed around the layer box 1026, thereby highlighting box 1026. When a layer is selected, controls specifically for that layer are also presentable. Figure 1001 shows a layer move control 1036 that allows the layer to be moved up or down in the editor layer stack. A transparency control 1038 for the layer allows the user to set the transparency degree for the layer, in this case layer 1026. These controls 1036 and 1038 appear when the layer is selected.

[0070] As noted above, the status icon 1030 is preferably not a control. A reason that it is preferred that the lock icon 1030 not be a control is the desire to make the target area for the layer control large for easy targeting and activation with a stylus. As a result, the present

invention makes the menu target areas large by overlapping status indicators (like the lock icon 1030) with the menu target area. In the present invention rather than having the status change or a status related function be activated when the icon is clicked, independent of the function of menu target area (for example, clicking on the lock unlocks the layer, preferably, clicking on the status icon pop-ups of the marking menu of the menu target area. In this way the menu target area includes the status areas and hence is larger and easier to hit.

[0071] A particular layer can be selected by tapping on it with the pen, which can result in a visual indication on the layer box, such as the highlight frame, that it is selected. This selection is conventionally performed by comparing the position of the cursor/pen to a map of the control hit zones for the editor 1010 and detecting a selection and a corresponding control when the cursor is within a particular layer or control hit zone of the map. The layer box is divided into three zones, the hit zones for and coincident with controls 1036 and 1038 and zone 1040 corresponding to and activating a pop-up marking type menu 1050 (see figure 1002 and the discussion below).

[0072] To facilitate fast access to multiple functions for each layer, each layer box (or control) preferably has a pop-up menu 1050 associated with it as shown in figure 1002. To pop-up this menu 1050, preferably the user has to press and hold the pen on a layer box and this pops-up a menu of commands that can be applied to the pressed on a layer. Note that press and hold first selects the layer box so a user does not have to separately click to select the layer, then with the press and hold selection of a particular layer activated, the application or execution/initiation of a command can be done with a single drag action (a mark) through the desired item of the menu, which is a conventional marking menu type selection with the menu visible (or invisible as desired). As discussed above, an aspect of this invention is that the menus can be implemented as conventional "marking menus" where activation of one of the controls via a mark simultaneously selects a corresponding layer and selects an operation on the layer. Other types of pop-up menus could be used, such a traditional linear menus or "pie menus", however, marking menus are particularly suited to pen-based applications because a selection can be made very quickly by a simple straight stroke with the pen in the direction of the desired menu item.

[0073] Figure 1002 also depicts a preferred pop-up, drag operation selection menu 1050 open over two layer editors. Each of the menus 1050 is popped up for the active layer selected by the user. This menu has radial menu selections including a new layer selection 1052 that

inserts a new layer above or preferably below the selected or active layer. A clear layer control 1054 that conventionally causes the system to clear the contents of the active layer. A rename layer selection 1056 that allows the user to rename the layer using gestures or strokes of the pen/stylus to write the new name into the layer box, as will be discussed in more detail later. A delete layer menu button 1058 conventionally deletes the current active layer by removing it from the stack when activated and the layer immediately below the deleted layer becomes the active layer. The merge control 1060 causes the system to conventionally combine or compose the active layer with the layer below the active layer so that the content of the layers are combined. A lock layer selection 1062 allows the user specify that the system lock/unlock the active layer. A hide layer selection 1064 causes the active layer to be converted by the system to/from being a hidden layer when activated. The position layer control 1066 puts the system into a mode in which the dragging on the layer itself (i.e., the cursor turns into a "dragging hand icon" and the user can drag the cursor on the image) moves the currently selected layer in 2 dimensions relative to the rest of the image. Note this is different from changing the ordering of the layers using the layer move control 1036. This position function is exited by selecting another tool, such as a pencil drawing tool.

[0074] Another feature of the invention is that the name contents of each layer box can be hand-drawn by the user. As shown in figure 1003, when a user selects the "Rename Layer" menu item 1056 of the marking menu 1050, a conventional drawing dialog box 1080 is displayed or pops-up. The dialog box 1080 includes a drawing zone or area 1082 where the user can simply draw any type of graphics to identify the layer. The dialog box 1080 also includes controls or buttons for clear 1084 that causes the system to clear the contents of the area 1082, cancel 1084 that causes the dialog box to be canceled by the system and OK 1088. Pressing the OK control 1088 causes the system to un-display the dialog box 1080, store the hand drawn graphic of the drawing area contents in the layer editor data structure and place the hand drawn graphic into the corresponding layer box.

[0075] The drawing area 1082 of the dialog box 1080 can be conventionally configured to support all drawing operations such as different colors, pen brushes, and visual effects found in paint and draw programs. The user input to the drawing area does need to be hand-drawn only. A visual reduction of the corresponding actual drawing layer can conventionally be provided to identify the layer (supplied by the system or the user). Alternatively, any small image could be input or a large image input and conventionally reduced by the system to identify the layer. One

variation of this embodiment is to use a special menu item on the marking menu, entitled "make thumbnail," to allow a user to create a layer box image by conventionally copying and reducing the image associated with a layer into the layer box. The image for the layer box can also be created by a "snapshot" style of interaction. For example, the "Rename Layer" dialog box 1080 can have an addition button labeled "snapshot" that, when clicked, allows the user to select some portion of the screen to be used as the image for the layer box. The source of an image for a layer can even be a video or any dynamic visual element. The input to label a layer could also be in the form of "ink strokes" as specified by the Microsoft Tablet PC specification. This allows other types of commands, such as "typeface" or "bold", to be conventionally applied to the hand-drawn strokes. Speech recognition or character recognition for the drawn strokes could also be implemented to provide a textual string used to name the layer. Different types of input could be combined into the "Rename Label" dialog image. For example, the user may paste an image into the dialog box and then write on top of that image. Another variation of the "Rename Layer" functionality is to have the layer box be directly drawn into (meaning that the rename commands and dialog box would not needed). In a further variation embodiment, the hand drawn area of the layer box could also pop-up and be separated from the area that has a pop-up menu associated with it.

[0076] The present invention preferably uses a hardware platform 1100, as depicted in figure 1004. This platform 1100 includes a conventional storage 1102 (RAM, ROM, hard disc, floppy disc, CD, DVD, etc.) for storing, paint programs, data structures, data, etc. used by the present invention as well as the processes, bitmap graphics and bitmap hit zones of the invention discussed herein. The platform 1100 also includes a conventional processor 1104, typically a tablet personal computer (Tablet PC) class machine. The platform 1100 further includes a conventional pen based I/O system 1106, such as a display with a pen based digitizer, and optionally includes conventional I/O components, such as a key board, floppy disc, mouse, etc.

[0077] The operations of the process for the pen based layer editor of the present invention are depicted in figure 1005. When the layer editor is invoked 1120, the system creates 1122 a layer display from a list of the layers in the paint system. This involves determining the number of layers and producing a layer editor graphic with the number of boxes needed and applying it to the display at a desired or predetermined location, obtaining the graphics for the layer names from storage and applying them to the corresponding boxes, obtaining the graphic for the other controls such as the minimize button and applying them to the display at corresponding

locations, determining whether a hidden layer overlay or locked icon needs to be applied, etc. Then each layer is activated 1124 as the corresponding controls so that a user can "click-on" each layer or on the other buttons to perform an associated function or display an associated pop-up display. The system then waits 1126 for one of the layer controls to be activated. If a layer box control has been activated as discussed above, the system selects 1128 or activates the layer under the pen cursor and provides the box on the display for the activated layer with the frame visually indicating that the layer has been activated. The system them determines the type of control that has been activated. This includes determining 1130 whether the control is the transparency control 1038 or the reorder control 1036. If one of these controls have been activated, the system pops up the appropriate control and performs the called for interaction 1132. If theses controls are not selected, the system determines 1134 whether the user is making a marking or radial menu type selection of items in the layer operation menu 1050. If so, the menu is popped up and/or a selection is made and the system performs the selected function.

[0078] The preferred data structure of the present invention is depicted in figure 1006. As depicted linked data structure includes:

a Layer Editor 1140 includes:
- a List of LayerEditorBoxes 1142
- a Scroll Control 1144

a LayEditorBox 1146 includes:
- Layer Image 1148
- Layer Box Image 1150
- Popup-menu 1152
- Popup menu target zone 1154
- Move layer control 1156
- Move layer control target zone 1158
- Transparency control 1160
- Transparency control target zone 1162

[0079] Figures 1007-1013 illustrate the controls of the layer editor 1010 of the present invention in more detail. Figure 1007 depicts the layer editor 1500 with two layers 1164 and 1166 "New" and "Background". In this example, New 1166 is the active layer as indicated by the border. This layer has three control zones: marking menu commands 1168; re-order layers 1170; and layer opacity 1172 controls. Figure 1008 shows the re-order layers transparent control visual 1170 with both up 1174 and down 1176 controls for moving the active layer respectively within the layer stack. Figure 1009 depicts the layer opacity control 1172 in more detail. Selecting

control 1172 brings up a dialog box 1178 depicted in figure 1010. The layer opacity dialog box 1178 includes a slider 1180 for setting the opacity of the layer from completely opaque to completely transparent. Figure 1011 illustrates the commands of the marking menu control 1182 for each layer with the preferred icons. These layer marking menu commands (starting North and going clockwise) include Add new layer icon/control 1184 depicting a layer with a "+", Delete layer icon/control 1186 depicting a layer with a "-", Rename layer icon/control 1188 depicting a layer with lettering, Flatten image icon/control 1190 depicting layers being pushed together by arrows, Merge with Below icon/control 1192 depicting two layers and a merge arrow, Lock layer icon/control 1194 depicting a layer and a lock, Hide/Show layer icon/control 1196 depicting a layer with an eye, and Move layer icon/control 1198 depicting a grab capable, open hand with a layer, the functions of the controls are discussed above with respect to figure 1002. Figure 1012 depicts the reordering of the layer stack by a user selecting the transparent "re-order" graphic 1170 on the active layer and dragging to the new position with the drag icon (closed grab hand) being depicted. Figure 1013 depicts the New Layer dialog box 1200 in which the user can draw in a textual name or sketch a graphic using a pen or stylus and this will be the new image associated to the active layer.

[0080] The layer editor interface as discussed above represents the layers as boxes arranged vertically. Other arrangements or embodiments are possible. Figure 1014 illustrates an arrangement of horizontal connected boxes 1212. Figure 1015 depicts tiled overlapping boxes 1214. Figure 1016 shows non-connected trapezoid type boxes 1216. Figure 1017 illustrates an alternate embodiment of circular layer controls 1218. Figure 1018 illustrates a mixed numerically labeled and overlapping display order of variously shaped layer representation controls 1220. Figure 1019 depicts controls 1222 with no spatial ordering where each box is displayed one at a time, possibly in order of time of creation. The layer editor controls can also be graphically "attached" to the display of the paint layers themselves as depicted in figure 1020. Figure 1020 shows a list of tabs 1224 along the display edge of the painted layered image 1226 where the layer box (the tab in this case) is visually connected to the actual image.

[0081] Another embodiment, as depicted in figure 1021 provides layer names 1252 and pop-up menus 1254 on the layers themselves. Figure 1022 is an image where three semi-transparent layers are placed over another image and on each layer a handwritten name or drawing is used. Each layer also has a spot where the pop up menu can be activated (the pop-up menu is shown

with numbered menu items but these items could be typical layer commands such as: new layer, clear layer, rename layer etc.).

[0082] Ultimately any ordered list or array could be used. Furthermore this invention is applicable to other areas beyond layer editors. For example, the list or array could be the thumbnails for a set or presentation slides 1266 which can be hand labeled by the user and controlled by marking menus (1267) on each slide as depicted in figure 1022. A 2d array of thumbnail images for a slide presentation could also be used. The slides are labeled by hand drawn graphics and a marking menu is used to apply commands to the slide boxes.

[0083] The user can also interact with the layer editor by using gestures and conventional gesture recognition and command invocation rather than menus. For example, as shown in figure 1023, rather than popping up a menu on a layer and selecting the rename command to invoke the rename dialog, a user can invoke the "Rename" dialog by writing an "R" symbol over the intended layer 1278. Gestures that span several layer boxes can also be use to specify several layer boxes simultaneously. For example, as shown in figure 1024, the letter "M" drawn between two layer boxes 1280 and 1282 can be used to cause the system to merge the two layers into one layer. The gestures do not necessarily have to have "ink trails" which are the sequence of pixels activated when a pen stroke is made. Also, a gesture can span outside of a layer box, perhaps with some point (typically the starting point) in/on the layer control 1284 as depicted in figure 1025. A gesture could also be a scratch-out-mark 1286 to indicate clearing or deleting of layers as shown in figure 1026.

[0084] As can be seen from the above discussion, the present invention has several advantages. First, user identification of layers does not require text input. This is extremely important for a pen-based computer especially when no keyboard is available or keyboard or text entry is cumbersome. Second, commands can be applied to the various layers very quickly and easily using marking menu technology. Third, the invocation of the marking menus is facilitated by large "hit areas" for the menus.

[0085] The present invention can also be used with mouse or other I/O device systems.

[0086] The system also includes permanent or removable storage, such as magnetic and optical discs, RAM, ROM, etc. on which the process and data structures of the present invention can be stored and distributed. The processes can also be distributed via, for example, downloading over a network such as the Internet.

[0087] The present invention is directed to a user interface, widget or control optimally placed on a display of a tablet-based personal computer (tablet PC) to take advantage of natural user motion. The interface, called the command corner or lagoon for convenience, specifically factors in the biomechanical properties of human user. The main context of this invention is on portable, pen-based computers but expands to other input scenarios such as electronic white boards, tablet digitizers, desktop PCs or virtual reality applications.

[0088] The present invention provides a core region (see figure 2003) where the main user interface components are housed that maximizes the ability to uniquely hit the targets and issue command gestures on the targets (e.g., marking menus). The command corner, or lagoon, is preferably located in the bottom left side of the screen for right-handed users (and vice versa for left handed users). This is an optimal placement due to the natural arc motion of the arm pivoting at the elbow and a user working in the center of the display screen.

[0089] A significant feature of the present invention is the strategic placement of the target components within the command corner. The target components can be one-shot commands or trigger sub-components/sub-commands (such as pop-up menu technology). The invention specifies that the optimal placement of the target components is an arc shape. This allows the broadest unique angle of approach from the center of the working canvas to the command targets. Having a unique angle of approach allows users to more reliably hit the target without a concern for selecting nearby targets. As a consequence of this design, the inner area of the command center is maximized given the minimum outer edge (which is called the leading edge).

[0090] The targets denoted by icons can be spaced to best accommodate marking menu strokes, gestures and pop-up menus/widgets. Thus, having the targets away from the edges of the screen allows for more gesture space. In addition, extra space is left at both ends of the lagoon arc.

[0091] The entire command corner (or some portion of it) can be displayed in a semi-transparent fashion to see the underlying application data and minimize visual distraction.

[0092] Additional user interface controls, indicators and feedback can be provided in the interior region of the command corner.

[0093] Multiple arc shaped rows can be defined to accommodate more functionality.

[0094] A special sub-component or control of the command corner can be designated as a toggle to collapse or minimize the command corner, which can later be selected to restore the command corner.

[0095] As discussed above, the interface of the present invention is placed at a location to take advantage of the natural arc motion of the arm of pivoting at the elbow, moving a hand holding a stylus of the tablet PC across the display where the user working in the center of the screen. Users of PC menus typically think of menus as having a top to bottom or left to right order and a learned movement is to access a menu item by moving the cursor in a typically downward direction. The present invention operates primarily with the dominant natural movement. This dominant natural elbow arc 2010 (for a right handed person) is depicted in figure 2001 by typical drawing strokes 2011 of a user along with the arcs formed by strokes made via motion of a wrist 2012 and fingers 2013 of the user. This arc 2010, when combined with the downward access direction learned behavior noted above, naturally carries the hand of a right-handed person holding the stylus to a lower left corner 2014 of a display 2016 (depicted by dashed lines) from a central region of the display. A command menu placed at the end of this arc 2010, such as in the corner 2014, can be more naturally accessed by the user with a neutral posture (no motion of the wrist or fingers) than other types of menus.

[0096] By making the command menu itself arced, such as depicted by the dot-dashed line motion extension arc (or anti-dominant arc) 2018, a slight motion by the wrist along the wrist arc 2012 or a slight motion by the finger arc along the finger arc 2013, the will allow travel of the stylus from the central region of the display 2016 toward the corner to be altered to travel a path that takes the stylus to any point along the extension arc 2018. Thus, a dominant motion around the elbow and a slight motion by the wrist or fingers allows easy natural access to commands in the command corner. The extension arc 2018 preferably intersects the dominant arc at a ninety (90) degree angle. A curved or arced interface as compared to a rectilinear interface has expanded approach zones are reduced interaction interference as will be discussed in more detail later.

[0097] Figure 2001 also depicts a pop-up menu 2020 having four (4) commands popping up at a location near the left edge of the display 2016. As can be seen, one of the commands is actually positioned off of the left edge of the display 2016 and is not accessible to the user. To prevent this, the arc 2018 preferably has a radius that positions the pop-up controls so that all of any underlying commands are accessible. That is, the arc 2018 radius is at least the equal to a

radius of any pop-up menu that pops up by activating a control located on the arc 2018. In addition, the pop-up menus for the controls of the menu arc 2018 are preferably located along the arc 2018 at a distance from the display edge greater than or equal to the radius of the of any pop-up menu located on the arc 2018. The menu 2022 depicts an example of a location for a pop-up menu for a nearby control on arc 2018. A pop-up warping alternative to such an inward location of a pop-up menu is to not place commands of the pop-up on a side of the hot spot closest to the display edge.

[0098] The arcs 2010, 2012 and 2013 (as well as for arcs for other types of displays, such a shoulder arc for a white board) can be standardized or custom as discussed in the related biomechanical interface application noted above.

[0099] The arcs and preferred interface locations relative to the display for a left-handed person are depicted in figure 2002 as in the lower right hand corner of the display. Of course the command corner can be located in the upper corners (upper right corner right for a right-handed person and upper left corner for a left-handed person), since the arc 2010 also extends upward.

[00100] Figure 2003 depicts a command corner 2040 preferably located and arranged in a display window 2042 according to the natural motion and command accessibility characteristics for a right-handed person as discussed herein. This command corner 2040 will be discussed in more detail later herein. This command corner interface 2040 is conventionally produced by a display graphic bit map with a curved design as depicted in figure 2003 and having the corresponding bit mapped control or hit zones for the commands of the interface.

[00101] The present invention preferably uses a hardware platform 2060 as depicted in figure 2004. This platform 2060 includes a conventional storage 2062 (RAM, ROM, hard disc, floppy disc, CD, DVD, etc.) for storing layer editors, paint programs, data structures, data, etc. used by the present invention as well as the processes, bitmap graphics and bitmap hit zones of the invention discussed herein. The platform 2060 also includes a conventional processor 2064, typically a tablet personal computer (tablet PC) class machine. The platform 2060 further includes a conventional pen based I/O system 2066, such as a display with a pen based digitizer, and optionally includes conventional I/O components, such as a key board, floppy disc, mouse, etc.

[00102] The command corner 2080 (40) of figure 2005 (see also figure 3) is designed for a drawing application (Alias SketchBook ™) and includes six controls 2082-2092 in display or

display window 2093. These irregular shaped controls icons include a page or file flipping control 2082 (next/previous page/file), a redo/undo or edit control 2084, a color selection control 2086 that allows the color applied by a tool to be selected, a tool (brush or pen) type control 2088, a tool selection control 2090 and a interface change control 2092 that allows the user to hide, suppress or minimize the interface 2080 or to move the interface to a different location, such as a different corner or a position along a side of the display or window 2093. This control 2092 can be explicitly shown (see figure 2005) or invisible or be shown in a conventional shadow mode as indicated by the dashed line in figure 2004. In shadow mode the interface is shown by some mechanism that indicates the presence of the interface but in an inconspicuous way. For example, the interface can be shown with dashed lines or with a slight change in the color or contrast of the portion of the interface in shadow. An inner radius 2094 of the interface 2080 is preferably offset from the opposite edge by at least the radius of a pop-up menu. For example, the control 2090 is offset from the bottom edge 2096 by the downward radius of any menu or control that appears when the control 2090 is activated. The controls 2090 and 2082 are also preferably located a distance in from their corresponding edges by at least the radius of any underlying menu.

[00103] For example, control 2082 is located inward along arc 2094 from edge 2096 by the downward radius of any menu that pops up when the control 2082 is activated. Rather than position pop-up menu type controls next to the display edge, it is preferable to place one-shot controls next to the edges. For example, control 2082 can be a one-shot control that always causes the display 2093 to display the next page (or reverse and display the previous page when the last page is being displayed). Note, the hit zone underlying each control need not conform to the shape or size of the graphic or icon for the control. The controls along the arc, alternatively, could correspond to the traditional linear menu bar at the top of the screen. For example, the controls could be: File, Edit, etc.

[00104] The command corner of the present invention, when designed for a drawing application, preferably has at least two controls, one control 2088 (i.e., marking menu) for selecting brushes and one control 2086 for selecting colors. Other controls can also be used in a drawing application and so the preferred command corner has five main control groupings: tools, brushes, colors, edit and file. In addition, we the preferred command corner has a sixth control element that manages the lagoon placement and window configuration.

[00105] Figure 2006 depicts the command corner in more detail showing the preferred icons as a page flip icon for control 2082, a reversing arrows icon for the edit control 2084, a color palette icon for the colors control 2086, a group of brushes icon for the tool control 2088, a set of edit tools icon, such as a magnifying glass for the control 2090 and page display icon for control 2092.

[00106] Figure 2007 illustrates the marking menu for the control 2092 showing in iconographic form the controls available with this Modes marking menu showing controls for: Last Brush (West), Actual size (NorthWest), Move and Zoom (North), Fit to View (NorthEast), Select tool (East), and Layer Dialog (South).

[00107] Figure 2008 depicts the default Choose Brush marking menu for control 2088 showing controls for: Highlighter (West), PaintBrush (NorthWest), 2H Pencil (North), 2B Pencil (NorthEast), Airbrush (East), Eraser (SouthEast), Brush Selector (South), and Ballpoint Pen (SouthWest).

[00108] Figure 2009 shows the default Choose Color Set marking menu for control 2086 showing controls for (starting from North and going clockwise): Black, Green, Red, White, Color Selector Dialog, Purple, Blue, and Yellow. When the user selects the color dialog selector by marking south as depicted in figure 2010, a color dialog box as depicted in figure 2011 is displayed.

[00109] Figure 2012 shows the default edit or Undo and Clear marking menu for control 2084 showing controls for: Clear (North), Redo (East), Grow Page (South), and Undo (West).

[00110] Figure 2013 shows the default Page marking menu for control 2082 showing controls for: Previous Page (West), Open (NorthWest), Save (North), New (NorthEast), and Next Page (East

[00111] Figure 2014 shows the default Command corner marking menu controller for control 2092 with the user is issuing command (a stroke to the east) to move command corner to right side of screen. The menus for this Menus are: Title bars; Full Tool Interface (North), No Title bars; Full Tool Interface (North East), Move Tool Interface to Right Side (East), No Title bars No Tool Interface (South East), and Title bars; No Tool Interface (South).

[00112] Figure 2015 shows the Command corner on right edge of window. The positioning of the marking menus for command corner controller have changed slightly: Title bars; Full Tool Interface (North), No Title bars; Full Tool Interface (North West), Move Tool Interface to Right

Side (West), No Title bars No Tool Interface (South West), and Title bars; No Tool Interface (South). Note that all other marking menus remain unchanged.

[00113] The present invention has a preferred mark selection that assists workflow. In this convention a downward stroke (the south item in a marking menu) will bring up additional tool palettes or dialogs. For example, the color marking menu (see figures 2008-2011) has individual colors on all marking menu items except for the south item that brings up the color chooser dialog.

[00114] The command corner of the present invention can be semi-transparent or blend/filter with the underlying canvas window it is on top of (e.g., you can see through the command corner but the content looks slightly blue tinted). The relative size of the command corner compared to the icon size and rest of application window is important, as the targets must be large enough to be hit easily as discussed in the related application.

[00115] Figure 2016 shows the command corner 2110 (2040) with a marking menu 2112 associated with control 2088 activated. This menu 2112 has an overflow menu 114 including overflow selections that are greater in number than can be displayed in marking menu format. The overflow menu 2114 is preferably located along the dominant arc from the pop-up or hot spot of the menu 2112. The items in the menu 2112 are also preferably located in conformity to the dominant arc and the extension arcs discussed above. The menu 2112 can also be laid out responsive to the dominant arc (or a combination of the dominant arc with an extension arc), such that a menu item can have an orientation consistent with the arc or an approach zone consistent with the arc as discussed in the related biomechanical application noted above.

[00116] Figures 2017 - 2031 depict alternate embodiments of the command corner.

[00117] Figure 2017 depicts a convex arc shaped interface 2120 with circular targets.

[00118] Figure 2018 shows a linear interface 2130 with circular targets 2132.

[00119] Figure 2019 has oval targets 2140 without a graphic interface border or outline where the larger axis of each oval is aligned with the approach path associated with the dominant arc and where the arrangement of controls is as a convex curve corner.

[00120] Figure 2020 depicts a convex interface with two rows or layers 2152 and 2154 of targets with the targets offset relative to the approach path along the dominant arc.

[00121] Figure 2021 shows targets arranged in a box shape 2160 with convex corner.

[00122] Figure 2022 illustrates an array 2170 of targets.

[00123] Figure 2023 illustrates a concave interface 2180 with circular targets 2182.

[00124] Figure 2024 shows polygonal targets arranged in an offset array 2190.

[00125] Figure 2025 shows circular targets 2200 in a bent oval interface 2202.

[00126] Figure 2026 depicts a convex interface 20210 of wedge shaped targets 2212.

[00127] Figure 2027 illustrates an interface 2220 with circular targets where the interface 2220 has a straight or linear portion 2222 and convex arc shaped portion 2224.

[00128] Figure 2028 depicts an interface 2226 with controls and underlying hit zones having curved sides set according to approach paths associated with the dominant arc producing curved sided trapezoid type targets.

[00129] Figure 2029 depicts an interface 2230 is existentially a circle of controls intersecting two sides of the display area edge and that allows additional off-screen functions or controls to be rotated into view on the display. Functions or controls A-E are currently visible and functions F-K can be rotated into view. The rotation can be conventionally performed by allowing the user to drag/rotate the interface around until a desired function is in view.

[00130] Figures 2030A and 2030B show an embodiment in which the target zones are expanded. A sectioned pie shaped command corner interface 2240 (see figure 2030A) in display 2241 has pie slice shaped targets 2242 (and corresponding hit zones) that can be accessed in a conventional way by positioning the cursor/stylus over the slice target and taping down on the display. If an additional control is available on the stylus, such as a conventional button the state of which is sensed by the digitizer system, the activation of the button causes the targets 2242 (and corresponding hit zones) to expand to a predetermined size, such as covering the entire display as depicted in figure 2030B. In this embodiment the demarcation between the targets can be invisible, shown explicitly with an extension of the interface graphics or shown in shadow mode as indicated by the dashed lines in figure 2030B. Note the sides of the targets are depicted as linear, however, the sides can be curved responsive to the dominant arc thereby producing curve sided triangle type targets.

[00131] Figure 2031 depicts a command interface 2250 that has been moved from a corner 2252 of the display 2254 to a position 2256 on a bottom side 2258 of the display 2254. This interface preserves the natural stoke path approach to the interface 2250 for most of the

controls with a near neutral posture and the remaining controls can be accessed via the natural stoke path with an extension as discussed above. This interface also preserves an expanded approach zone for the controls. The command interface 2250 can be moved to such a side position by dragging the interface to a desired and the system conventionally maps the quarter circle interface of a corner embodiment into a half circle.

[00132] Figure 2032 illustrates the operations of the present invention in displaying and allowing a user to use the command control of the present invention. The process starts with determining 2280 whether the command interface has been activated. If so, the list of controls for the interface is accessed 2282 and the controls are mapped 2284 to their designated locations. This mapping can be to a pre-designated corner or a display side position and is preferably positioned responsive to the arc or arcs discussed herein. The mapping also depends on the radius of the arc in the corner of the display or display window. If the window is big the radius can be correspondingly bigger than for a small window. The radius is also affected by the size of any underlying menus activatable by the controls. The location of the controls placed along the arc also depends on the function of the control (one-shot) versus menu with a radius) as previously discussed. The user can also specify a desired custom position. The system then awaits a control activation 2286, which typically occurs by taping the stylus on the surface of the tablet PC display within a hit zone of the desired control. During this wait period conventional display operations occur such as tracking the position of the stylus and positioning a cursor under the stylus, and highlighting a control when the cursor/stylus passes over (or close to) the control.

[00133] The system conventionally determines whether a control or button has been activated by comparing a cursor position to a stored hit zone map when an input selection signal is received, such as the detection of a tap of the stylus on the display. If a control has been activated, the system determines 2288 whether the control is the interface change control (see 2092 of figure 2005). If so, the function 2290 of that control, which collapses/minimizes the interface or allows the user to position/move the interface to another location, is performed. If minimize is activated, the interface can be shrunk into a portion of a dot in the corner. If a change to the interface is not activated, the function of the activated control is performed 2292. If the control is a one-shot control, the corresponding command is executed. If the control activates another menu, the menu is popped up and displayed if needed away from the edge or

warped to pull all commands on the display. If the control activates some other function, that function is performed. The system then awaits another control activation.

[00134] In the present invention a special case can exist for moving the interface to another location when the move to the other side of the screen is requested. In the present invention the user is allowed to perform a "flick" gesture on the control 2092 toward the other side of the screen to move the lagoon to the other side. Once the lagoon is on the other side of the screen, the opposite, reverse flick direction will move the lagoon back. Note that the lagoon shape and hit zones change when it is moved to the other side of the screen to a type of mirror image as discussed above. The flip is not a complete mirror flip on of the lagoon shape and contents. For example, most of the icons on the controls do not flip, nor does textual labels, such as the logo "Alias SketchBook" in figure 2005).

[00135] The performance of the action associated with the activated control is preferably performed to allow multiple approaches to selection of underlying or deeper menu level operations as depicted in figure 2033. This preferred sequence starts with a button or control input being detected 2320. The system then conventionally determines whether the input indicates a tap 2322 or a dwell 2324 or a drag (i.e., stroke). A tap is a momentary touching of or contact by stylus with the display, a dwell is a hovering of the stylus over a control location for a predetermined period of time and a stroke is a contact of the stylus with the display and a movement of the stylus while in such contact. If the input indicates a tap, the underlying menu is displayed 2326 for a predetermined period of time. A tap event needs to be distinguished because sometimes a user's behavior is to explore the interface and they will do so by making quick taps on controls. If the system does not distinguish or detect taps, a user may not discover the underlying marking menus because they were not dwelling long enough to invoke the marking menu popup.

[00136] The marking menu delay or dwell is necessary to prevent the marking menu pop up from displaying when an expert user of marking menus just wants to perform a flick gesture to issue a command. Without detecting a tap event, the system would have to always display the marking menu visuals when any pen-down event occurred. By detecting a tap, the system allows both worlds. Note that this design also allows the system to issue a default command when a tap is detected while still supporting standard marking menu interaction. If the input is a dwell, the menu is displayed 2328 and a menu interaction is performed. If the input indicates a stroke, a marking menu type selection 2330 is conventionally made responsive to the stroke.

[00137] The present invention is designed to improve the zone of approach for approaches to the interface by the movement of a stylus over a table PC type display in a natural motion, such as the arc shaped path of an elbow pivot dominated stroke. The design principle associated with the natural motion arc has been discussed above. Additional design principles will be discussed below.

[00138] Figures 2034A – 2034C illustrate an interference principle associated with designing such an interface. In figure 2034A a linear interface 2350 and arrangement of controls 2352 along a bottom 2354 of a display 2356 is shown. To approach one these controls without taking a path that crosses over another one of these controls, this interface provides approach areas 2358 and 2360 that overlap and cause approach interference. This approach area interference is about 90 degrees. That is, the hit zones, if mapped to these approach areas, will overlap or interfere in the interface graphic 2350. This can cause a user to mis-select a control. In figure 2034C the interference caused by the approach areas 2360-2368 have been reduced to zero. However, the approach path to some of the controls, such as in area 2368, is awkward. Figure 2034B depicts an interface, such as described herein, where the interference has been reduced. The invention reduces the approach area interference.

[00139] Figures 2035A and 2035B illustrate another design principle of the interface of the present invention called approach zone improvement. As depicted in figure 2035A, a linear interface 2380 has a control 2382 with an approach zone 2384 having a width distance 2386 determined by the boundaries of the interface 2380. This approach zone with width distance 2392 can be increased by an arced or curved interface 2394 that increase the angle of the zone 2390 by arcing the entrance into the interface as discussed herein and as shown in figure 2035B. The present invention expands the interface approach zone.

[00140] Figures 2036A and 2036B illustrate another design principle of the interface of the present invention when multiple layers of controls are provided in the interface. As depicted in figure 2036A, controls 2410, 2411 and 2412 in different layers can be aligned or coincident with the same approach path 2414. The principle is to position the controls, if possible with non-coincident, dominant arc approach paths. This can cause path position selection errors where the user moves in along a path and moves too far or too little and mis-selects a control, such as control 2411 when control 2410 was desired. In an interface according to the present invention, an approach for two or more controls that have a path coincidence is designed to have a layer offset where controls of adjacent layers are offset in a direction perpendicular to the path and a

path direction offset where controls on the same path are in different layers as depicted in figure 2036B. This improves the spatial separation between controls and reduces mis-selection. The present invention offsets adjacent control layers and puts coincident path controls in different layers.

[00141] In the present invention, that if a user is inking in the drawing canvas and crosses into the command corner, inking still occurs on the underlying canvas (not on the command corner). In addition the command corner can include internal holes below the main arc. With such holes, a user can start laying down ink if the initial pen-down event falls within a hole and then they can continue and cross through the command corner and a continuous ink stroke will occur as expected.

[00142] The present invention has been described with respect to pen-based systems but touch screen or computer vision techniques for tracking human input are also applicable. Moreover, the present invention will work well for two-handed input systems. In a two handed input situation the display could have two command corners active (one for each hand). A two-handed drawing program on hardware such as a large, rear-projected screen having multiple inputs (e.g., the SmartBoard from SmartTechnology), works well to have the non-dominant hand operate the command corner to switch color, brushes, modes, etc., while the dominant hand remains relatively stationary where the user is currently drawing with their dominant hand. In the context of the present invention there can be a distinction between a display screen and an application window. In the present invention, the command corner may change shape depending on the placement of the application window within the display screen as well as the command corner changing shape depending on the aspect ratio of the application window.

[00143] Tool mode and status indicators can also be housed within the command corner. For example, the current brush color and current tool mode can be visually shown within the command corner. Control elements housed within the command corner can also have "tool tips" where if a user hovers over an icon, a temporary text label pops-up to tell them what the icon is and it is dismissed when the pen moves.

The system also includes permanent or removable storage, such as magnetic and optical discs, RAM, ROM, etc. on which the process and data structures of the present invention can be stored and distributed. The processes can also be distributed via, for example, downloading over a network such as the Internet.

[00144] The present invention provides an improved interface element for pen-based type computers by altering the shape of the user interface widgets, positioning them away from the edge of the screen and conforming them to the biomechanics of the human body. Simply put, the present invention provides interface widgets that are reshaped into or reoriented along "arc" shapes or arc like curves. For example, a slider control may not be straight or rectilinear but instead have a curved shape. Or the slider can be oriented along a particular portion of an arc. A row of button controls (such as in a toolbox) can also follow this arc shape. These interface elements or controls, sometimes called widgets, can be implemented using conventional bitmap graphics (or any other graphics drawing system such as vector graphics) with irregular hit or activation zones defined (such as, curved hit zones). Detection of the selection of a control (hit detection) can be efficiently performed by defining an internal data structure that is also a bitmap that matches the shape of the widget/palette. Each button or clickable item can be colored a unique color. Continuous items such as sliders can have a color gradient.

[00145] The interface elements of the present invention can include or be oriented along a specific "idealized" or generalized element arc (or curve) shape (in degrees) for the user to target when interacting with the interface, although a custom user specific arc can be used. This element arc is based on the arc 3010 of the movement of the hand, as depicted in figure 3001, when the arm is pivoted at the elbow and/or on the arc 3012 of the hand when the hand is pivoted at the wrist. The element arc can also be based on the arc of the shoulder and the arc of the fingers, as will be discussed later herein. This element arc can generally depend on the characteristics of each user as shoulder, arm, hand, finger lengths, etc. are somewhat different from person to person. The element arc varies slightly between the large gestures for an arm pivoting at the elbow and/or shoulder and a hand pivoting at the wrist. The arc also may vary depending on what device is being used (e.g., a PocketPC, TabletPC, etc. or an electronic whiteboard). Different orientations for the arcs 3014 and 3016 are typically defined for right-handed user (figure 3001) and left-handed users (figure 3002). An idealized arc can be determined by having a cross section of users draw arcs on a display, having a processor sample the drawn arcs, conventionally fit arc curves to the drawings, and then select an idealized element arc by one of a number of approaches. The arcs or radii of the cross section of users can be averaged to obtain the idealized element arc.

[00146] Alternatively, the arc radii can be fit to a normal, standard deviation or other type distribution and the idealized arc determined based on the distribution. In addition, a user could

define or customize a baseline arc of comfort to be used instead of the idealized element arc in a similar manner by having the user draw an arc strokes on the display, a processor sample the drawn arcs, and conventionally fit an arc curve to the drawing strokes. In such an idealized or customized element arc, the radius need not be constant and can vary along the curve to match the particular combinations of pivots or motions by the user(s). It is also possible to have different "standard" arcs for different types of devices, such as a large radius shoulder arc for a white board, a medium radius elbow and/or wrist arc for a laptop computer or a digitizing tablet and a small radius finger arc for a PDA.

[00147] The present invention also recognizes that widgets can be oriented differently depending on the operating position of the widget relative to the user. For example, a user 3018 is working at an electronic whiteboard 3020 (see figure 3003) and the user is controlling a slider; the slider can be popped-up directly in front of the user working at the electronic whiteboard, e.g., in front of their chin. In this situation, an arc shape slider 3022 can reflect roughly a downward arc, like a frown on a smile-face. However, if the user popped-up the slider a significant distance to their right, a slider 3024 can be oriented sideways, like a right handed, close-parentheses ")".

[00148] The present invention also provides for the most frequently used commands to be positioned along the main "dominant" arc or curve pathway. So, for example, the Alias SketchBook ™ application, for right-handed users, has it's main command toolbox located at the bottom left of the screen (the base of the dominant arc) where the hand moves from the command base to the main canvas region as discussed in the related application.

[00149] Pop-up widgets, such as menus, can also conform to the arc shaped layout as discussed later herein.

[00150] The present invention preferably uses a hardware platform 3040 as depicted in figure 3004. This platform 3040 includes a conventional storage 3042 (RAM, ROM, hard disc, floppy disc, CD, DVD, etc.) for storing layer editors, paint programs, data structures, data, curve fitting functions, etc. used by the present invention as well as the processes, bitmap graphics and bitmap hit zones of the invention discussed herein. The platform 3040 also includes a conventional processor 3044, typically a tablet personal computer (Tablet PC) class machine. The platform 3040 further includes a conventional pen based I/O system 3046, such as a display with a pen-based digitizer, and optionally includes conventional I/O components, such as a keyboard, floppy disc, mouse, etc. The I/O system 3048 can also be the type that allows

touch input by a finger and input by a 3D position sensing system that senses the 3D position of a finger, hand, eyes, etc. as well as via a mouse or other conventional input system.

[00151] In designing an arc shaped or oriented element, the user(s) is allowed to make pen strokes 3060 on a pen-based computer display using a conventional drawing tool as depicted in figure 3005. This results in naturally shaped, neutral posture (no motion of fingers or wrist), arc shaped strokes due to the arm pivoting at the elbow (and possibly with some added curve influence of the hand pivoting at the wrist). The strokes are sampled and a curve is fit to these strokes in a conventional manner. These stroke arcs have a radius R that generally defines the curve of the arc.

[00152] A linear interface element, such as a slider 3062, can be oriented essentially tangent to the arc shaped strokes 3062 (or the fit curve) at a point along the strokes, such as a mid point, as depicted in figure 3006. Note the text "100" at the top end of the interface can be rectilinear aligned with the display or curved. Such an arc oriented interface element 3062 is easier to use than the typically vertically or horizontally oriented interface element as the user can move along the element 3062 more easily with a more to natural simple type, non-compound motion.

[00153] An interface element, such as slider 3064, can also be bent or arced to follow the arc shaped strokes 3062 as depicted in figure 3007. This curved slider 3064 conforms to the element arc discussed above, and is much easier and more natural to use than the typical rectilinear interface elements. This curved slider 3064 can be created by conventionally mapping the pixels of the bitmap of a conventional slider to a bitmap curve or vector graphic curve for the curved element corresponding to the arc. For example, a bit, pixel or graphic element at a particular length distance from a reference end of the conventional slider along a centerline of the conventional slider and at a width distance from the center line on one side of the centerline of the conventional slider are mapped to the corresponding position relative to the arc curve in the arc shaped slider 3064. This mapping is performed for all of the graphic elements of the slider. A corresponding mapping occurs for the hit zone. The interaction behavior of the standard slider maps to the new arc shape space and orientation. Even invisible elements such as an out-of-range border for the standard slider maps to an arc shaped out-of-range border for the arc-based slider.

[00154] A multiple control interface element, such as a menu 3080, can also be oriented to the arc shaped strokes 3060 and maintain it's rectilinear shape, as depicted in figure 3008. That is,

the sides 3081 of the controls are linear. In this case the menu 3080 is also aligned with a curve 3082 that is perpendicular to the strokes 3060. This curve 3082 is called an anti-arc, anti-curve or counter-arc, and is a curve that is oriented 90° from the elbow dominant arc of the strokes 3060. This arc allows selection of items along this arc by slightly changing the shape of the dominant arc with a slight added motion added by the wrist and/or fingers, that causes the dominant arc to be extended above or below its natural path. Such a multiple control menu 3100 can also have the rectilinear controls staggered along the arc shaped strokes 3062 as depicted in figure 3009. And a multiple control menu 3110 can itself be arc shaped as depicted in figure 3010, such that the controls are non-rectilinear and conform to the arc shaped strokes 3062. That is, the sides 3112 of each of the controls is arc shaped.

[00155] Pop-up toolbars 3120 can also adopt the orientation of the arc as depicted in figure 3011A. Note how the marking menu icon remains horizontally orientated for easy reading while the individual tool targets (hot spots) 3122-3128 adhere to the anti-arc extended arc orientation. This figure also shows the concept of a marking menu 3126 as one of the tools, with visible commands in this case, and that has an "overflow" area 3130 that, in this example, is up and to the right of the marking menu center along the dominant elbow type curve. As can be seen, the positioning of the overflow area 3130 can also conform to the underlying arc paradigm. The shape of the overflow area 3130 can take on the similar arc contour shaped palette depicted previously in figures 3009 and 3010. Figure 3011B illustrates the pop-up menu with an elbow arc 3131 shaped overflow area 3132 with arc shaped controls 3134 therein. This embodiment also shows the marking menu regions 3136 oriented and aligned along a wrist shaped arc 3138.

[00156] The targets or hit zones of the controls (as well as the graphics thereof) can be curved as more particularly illustrated by the wedge shapes targets 3140-3146 of figure 3012. This figure also shows a marking menu 3148 with visible commands popping-up as a result of the activation of the control for zone 3142.

[00157] Figure 3013 depicts a pop-up menu 3160 positioned at the location of the cursor 3162 and with the selectable items 3164-3170 oriented along the biomechanical element arc 3172. In this case, the pop-up 3160 has four items 3164-3170 represented graphically as circles intersecting the arc 3172. The pop-up 3160 remains fixed until the mouse-up event occurs. On initial pop-up, the default item 3168 is conventionally presented directly under the current position of the cursor 3162. The user can move up or down the arc 3172 to select the different items 3164-3170.

[00158] In figure 3014 an arc-shaped command strip 3190 is defined. The strip 3190 is anchored at a base 3192 and contains user interface elements 3194-3198. In this case, each element contains an icon and a marking menu. Alternatively, other widgets and status information can be housed on, within or near the command strip. The command strip 3190 can be manually re-oriented (twisted or bent) and the elements follow the new orientation. The command strip can behave like real-world materials and have the same dynamic properties (e.g., a metal strip with a given flexibility and stiffness where each item is a ball with a given physical weight). The user can manipulate the command strip 3190 to enlarge or shrink it or to move it out of the way. For example, the strip 3190 can be bent downwards, upwards, pulled-out (to expose more elements on the command strip, to expand individual targets on the strip, or to increase spacing between items on the strip). Conventional move, expansion, etc. techniques are used to re-orient, the strip 3190.

[00159] Figure 3015 depicts another arc based interface embodiment 3200. In this figure the bottom left corner region of the screen 3202 houses a number of "targets" A-D roughly in the shape of triangles with arc shaped sides. Each of the regions A-D houses a different command that in this case is a different marking menu. The "blades of grass" each having a length or width W delineate the different hit zones. This interface allows for extremely wide targets when the user is starting roughly from the center of the display screen and is moving the cursor to the bottom left corner of the screen. It is known that wide targets allow for fast selection performance. Note that the bottom right corner with the blades of grass pointing towards the center could be used if the user were left-handed.

[00160] Figure 3016 is an example of a marking menu overflow region 3210 positioned up and to the right of a circular menu region 3212 (see the related application). In this example, the overflow region houses a variety of paint brushes, pencils and erasers. The shape of the palette 3210 and the layout of the individual items are arranged roughly to match the biomechanical arc. This figure also shows the interface 3212 positioned at a location accessible by a user when making a natural elbow pivot to swing a hand across a display toward the lower right corner.

[00161] Figure 3017 is an example of an interface with a mixture of elbow and wrist orientation. Here we see a collection of sliders 3220-3226 oriented along a main or elbow arc 3228 and each slider is oriented or shaped to match the wrist shape arc. This allows the user to select a slider by elbow pivoting and then control the slider by wrist rotation.

[00162] In figure 3018 control items 1-3 are arranged along an elbow type arc 3240 where each item has a narrow but deep hit zone (a-c). The hit zone for each item also includes a region within the dominant arc widget. Figure 3019 depicts a similar layout with wider hit zones (a-c) for the items 1-3.

[00163] Figure 3020 shows a sequence of arc motions 3250-3260 that can be used to set different parameter attributes. Alternatively, it can increment values (in this case, 3 increments are specified).

[00164] Figures 3021 and 3022 illustrate combinations or rectilinear and arc shaped interface elements. A rectilinear menu 3280 with an arc shaped sub-menu 3282 is shown in figure 3021 while an arc shaped menu 3284 with rectilinear sub-menus 3286 and 3288 is shown in figure 3022.

[00165] Figure 3023 shows a hybrid menu 3290 with an arc shaped portion 3292 and a rectilinear portion 3294 allowing a combination of easy to access elements and elements that are harder to access such as for functions with less used consequences, such as a delete function.

[00166] The operations of the processor in producing pop-up arc shaped or oriented interface (GUI) elements of the present invention are shown in figure 3024. When the pen-based I/O system hardware detects an activation signal, such as a tap or pen down, the location of the cursor is determined and a determination 3300 is conventionally made as to whether an arc based interface element has been activated by comparing the position of the pen/cursor to hit zone maps in storage. If the cursor/pen is in a region where a pop-up menu can pop-up, the system determines 3302 whether the user has designated that the pop-up interface be a custom interface. If so, the custom arc is retrieved 3304. If not, the standard or element arc is retrieved 3306. The system then positions (3308) or locates the arc along, at or in association with the cursor/pen position. If the location also affects the orientation of the arc, such as in the whiteboard example previously discussed, the arc is oriented responsive to the location. Because the interface element pops up where specified by the user, the user can position the interface away from a display screen edge and at a position where it is comfortable to use. The interface or widget is then mapped or applied 3310 along/on the arc by mapping the widget graphic bitmap to the arc along with a similar mapping of any control zone for the widget. The system then monitors further movement of the pen/cursor to perform operations associated with the interface.

[00167] A non-pop-up or fixed position element can also be produced in a similar manner by starting with a known position for the interface, retrieving a custom (3304) or standard (3306) arc and applying (3310) the interface at the predetermined location. To reduce the interference with interface operation by the display edge, the fixed position arc shaped or oriented interfaces elements can be located to a preferred side of the display, such as on the left side for a right-handed user or spaced in from the display edge by the distance of a typical wrist. To additionally reduce the interference by the edge of the display, the fixed position elements preferably also "grow" towards the center of the screen and are pop-up elements.

[00168] As previously discussed, the invention provides for the use of a combination of a number of arcs in setting a position of an interface and/or determining the shape of the interface as depicted in figure 3017 the combination of the different arcs into a single composite arc. The elbow and wrist arcs have been primarily discussed with respect to previously presented figures, such as figure 3001. Figure 3025 illustrates the relationships and shapes of the shoulder arc 3320, the elbow arc 3322, the wrist arc 3324, the finger arc 3326 and the anti-arc 3328 or extension arc. In a three dimensional input space, other arcs can also be added to or combined with the arcs of figure 3025 to provide a natural motion interaction in such a space. For example, an arc or curve associated with a waist twist or waist bend can be added or combined with the curves of figure 3025.

[00169] Figure 3006 was discussed with respect to positioning the linear arc oriented interface element on a tangent to the element arc. It is also possible to position the element 3340 across the arc 3342 as depicted in figure 3025. Using this approach the fingers are moved inside the arc 3342 rather than outside the arc when manipulating the linear arc oriented interface element.

[00170] It should be noted that, while the examples discussed herein show a particular curved arc; the invention is not limited to this style or shape of arc. Specifically, widget or interface elements could be a design based other shapes of arcs or curves. For example, "S" shaped arcs or segmented straight lines effectively creating a curved interface element can be provided. The curve of the interface element of the invention is also not limited to paths that reflect biomechanical advantages either. For example, non-arc shapes could be used to produce a certain visual look as opposed to a particular motion.

[00171] The present invention has been described with respect to an interface useful for a tablet-stylus type personal computer. The interface and processes discussed herein are also

useful in non-tablet type personal computers. The hit zone where a user can activate a control conforms to the arc of the natural motion curve has been described as coinciding with the interface graphic. However, the hit zone can be larger, smaller or a different shape than the graphic.

[00172] The present invention has been described with respect to pen-based systems but also applies to touch-screen systems and computer-vision techniques used for sensing human input, as well as systems that sense position in three dimensions of a finger, hand, etc.

[00173] The present invention has been described as conforming the approach path to a UI to a natural motion curve, conforming the shape of an overall UI graphic to the natural motion curve and as well as shaping and positioning the control elements in conformance with the natural motion curve. A natural motion curve is predominantly a smooth arc; however, a natural motion curve can include small bumps or natural variations caused by a number of different things, including a motion that uses coarse motor muscles and fine motor muscles.

[00174] The present invention has also been described with respect to positioning the controls along or on the natural motion arc or curve. However, such a positioning can, in some instances, obscure the view of the user of the control because the user's hand could cover part or all of the control. The controls can thus be positioned above or below the curve with above the curve being preferred because it is less likely to obscure the control.

[00175] The system also includes permanent or removable storage, such as magnetic and optical discs, RAM, ROM, etc. on which the process and data structures of the present invention can be stored and distributed. The processes can also be distributed via, for example, downloading over a network such as the Internet.

[00176] The many features and advantages of the invention are apparent from the detailed specification and, thus, it is intended by the appended claims to cover all such features and advantages of the invention that fall within the true spirit and scope of the invention. Further, since numerous modifications and changes will readily occur to those skilled in the art, it is not desired to limit the invention to the exact construction and operation illustrated and described, and accordingly all suitable modifications and equivalents may be resorted to, falling within the scope of the invention.

CLAIMS

What is claimed is:

1. A layer editor interface, comprising:

layer representation graphic having layer names; and

a pop-up menu control activatable for each layer that allows the layer to be edited.

2. A layer editor as recited in claim 1, wherein the pop-up menu control is a marking menu control.

3. An interface as recited in claim 2, wherein the marking menu control includes selections for new layer, clear layer, rename layer, delete layer, merge layer, lock layer, hide layer and position layer.

4. A layer editor interface, comprising:

layer representation graphics having layer names and selection targets with a box shape; and

controls associated with the graphic that are coincident with the targets.

5. A layer editor as recited in claim 4, wherein a status indicator overlaps the selection targets.

6. An interface, comprising:

layer representation graphic having layer names inputable by a user and displayable as hand drawn strokes; and

a control associated with the graphic that allows a corresponding layer to be edited.

7. An interface as recited in claim 6, further comprising a drawing dialog box invoked by the control and allowing the user to input the layer names.

8. An interface as recited in claim 6, wherein the graphic has target areas with target sizes of at least 2e where e is the distance error accuracy of an input device.

9. An interface as recited in claim 6, wherein the control invokes a marking menu layer editing menu.

10. An interface as recited in claim 6, wherein an active layer is highlighted with a frame surrounding the name.

11. An interface as recited in claim 6, wherein each layer control comprises:
a marking menu control for layer editing;
a move control for moving a position of a layer in a layer editor stack; and
a transparency control controlling the transparency of a corresponding drawing layer.

12. An interface as recited in claim 6, wherein each layer graphic has an indicator indicating whether a corresponding drawing layer is hidden/visible.

13. An interface as recited in claim 6, wherein each layer graphic has an indicator indicating whether a corresponding drawing layer is hidden/visible [locked or unlocked?].

14. An interface as recited in claim 6, wherein a background layer has a text label.

15. An interface as recited in claim 6, wherein activation of one of the controls via a mark simultaneously selects a corresponding layer and selects an operation on the layer.

16. An interface as recited in claim 6, wherein making a gesture in association with the layer representation graphic initiates a function with respect to one or more of the layers.

17. A layer editor interface, comprising:
layer representation graphic having layer names inputable by a user and displayable as hand drawn strokes, and having selection targets with a box shape, each selection box comprising:
a marking menu control activatable for each layer that allows the layer to be edited where the marking menu control comprises edit controls for new layer, clear layer, rename layer, delete layer, merge layer, lock layer, hide layer and position layer;
a re-order layers control; and

a layer opacity control having a dialog box with a slider for setting layer opacity,

wherein each of the controls has a target size of at least 2e where e is the distance error accuracy of an input device,

wherein an active layer is highlighted with a frame surrounding the name,

wherein each layer graphic has an indicator indicating whether a corresponding drawing layer is hidden/visible, and

wherein each layer graphic has an indicator indicating whether a corresponding drawing layer is locked.

18. A layer editor, comprising:

linear list of layers; and

a marking menu accessible through the layer list and providing layer editing functions.

19. A method, comprising:

displaying a layer of a layer editor interface having a user entered graphic layer name on a display of a pen-based computer;

activating the layer graphic as a control; and

allowing the user to invoke a layer editing operation with using a pen of a pen-based computer to activate the control.

20. A method as recited in claim 19, further comprising:

displaying a marking menu responsive to the activation; and

allowing the user to select a layer edit function by making a mark with the pen.

21. A method as recited in claim 19, further comprising:

displaying a drawing dialog box responsive to the selection of the layer edit function;

allowing the user to write the name in the dialog box; and

displaying the name in a graphic layer representation corresponding to the layer.

22. A method as recited in claim 19, further comprising allowing a user to edit layers using a gesture.

23. A method as recited in claim 18, further comprising allowing the user to invoke the layer editing operation via a mark that simultaneously selects a corresponding layer and selects an operation on the layer.

24. A computer readable storage for controlling a computer by displaying a layer of a layer editor having a user entered graphic layer name on a display of a pen-based computer and activating the layer graphic as a control.

25. An apparatus, comprising:

a display; and

a processor displaying a layer editor interface on said display, said interface comprising a layer representation graphic having a user entered graphic name and a corresponding control.

26. A computer readable data structure for controlling a computer, comprising a layer editor linked to a list of layer editor boxes and a scroll control with each layer editor box comprising a layer image, a layer box image, a popup-menu, a popup menu target zone, move layer control, a move layer control target zone, a transparency control and a transparency control target zone.

27. An interface, comprising:

an interface area located responsive to a natural motion by a user and, comprising:

a graphic defining the interface area; and

controls located in the interface area and accessible via the natural motion.

28. An interface as recited in claim 27, wherein the natural motion is a curve associated with movement of a hand of the user when an elbow of the user is pivoted.

29. An interface as recited in claim 28, wherein a location responsive to the natural motion of the user hand is defined by the natural motion passing through a substantial center area of a display area.

30. An interface as recited in claim 27, wherein the natural motion is a curve associated with movement of a hand of the user when an elbow of the user is pivoted and one of a wrist of the user is rotated and fingers of the user are moved.

31. An interface as recited in claim 27, wherein an interface location responsive to the natural motion of the user is a lower corner of a display area.

32. An interface as recited in claim 27, wherein the graphic is a shape corresponding to an arc shaped curve and the controls are positioned in accordance with the curve.

33. An interface as recited in claim 32, wherein a radius of the arc shaped curve is at least a radius of a menu of one of the controls.

34. An interface as recited in claim 32, wherein a control closest to a display area is positioned along the curve at least a radius of a menu of the control.

35. An interface as recited in claim 27, wherein a menu associated with one of the controls has a layout responsive to the curve.

36. An interface as recited in claim 27, wherein a marking menu associated with one of the controls has a layout where a downward stroke brings up additional tool palettes and/or dialogs.

37. An interface as recited in claim 27, wherein the interface is located in a lower left corner of a display area and the controls of the interface are arranged as one of a convex arc across the corner, a concave arc across the corner, a line across the corner, an array in the corner, a convex corner across the corner, a convex arc with a linear portion across the corner, a sectioned pie in the corner, a sectioned pie in the corner and extending across the display area, and a rotatable circle intersecting both sides of the corner.

38. A graphical user interface, comprising:

an interface having an interface arc shape, located in corner of a display area, having graphics for controls arranged along the interface arc and having control hit zones each with a zone shape responsive to an approach arc defined by a dominant motion arc of a motion of a user with the graphics of the controls being located responsive to one-shot function or menu pop-up function with a pop-up menu radius.

39. An interface as recited in claim 38, wherein the zone shape comprises one of a wedge, a curved sided triangle and a curved sided trapezoid.

40. An interface as recited in claim 38, wherein the zones have non-coincident, dominant arc approach paths.

41. An graphical user interface for a digitizer based drawing application, comprising:

a semicircular graphic located in a corner of a display area of the drawing based application; and

controls located essentially in an arc in the graphic, said controls comprising:

a tool control located adjacent the minimize control and providing a menu for selecting a drawing tool of the application; and

a color control located adjacent the undo control and providing a menu for selecting paint color applied by a drawing tool of the application.

42. An interface as recited in claim 41, wherein said controls further comprise:

a minimize control located on a side edge of the graphic and providing a minimize function for the interface;

a page/file control located adjacent a bottom edge of the graphic and proving a page change function for drawing pages of the application;

an edit control located adjacent to the page control and providing an undo function for the application; and

a tool type control located between the tool control and the color control and providing a menu for selection a tool type of the application

43. An interface as recited in claim 42, wherein the graphic comprises a semicircular band.

44. An interface as recited in claim 42, wherein pop-up menus pop-up in association with the selected control and at a distance from side and bottom edges of the graphic to allow all menu commands to be displayed.

45. An graphical user interface for a tablet personal computer based drawing application using a stylus, comprising:

a semicircular graphic located in a corner of a display area of the drawing based application responsive to a natural motion by a user wherein the natural motion is a curve associated with movement of a hand of the user when an elbow of the user is pivoted; and

controls located essentially in an arc in the graphic and activated by the stylus, said controls comprising:

a minimize control located on a side edge of the graphic and providing a minimize function for the interface;

a page control located adjacent a bottom edge of the graphic and proving a page change function for drawing pages of the application;

an undo control located adjacent to the page control and providing an undo function for the application;

a tool control located adjacent the minimize control and providing a menu for selecting a tool of the application;

a color control located adjacent the undo control and providing a menu for selecting paint color applied by a tool of the application; and

a tool type control located between the tool control and the color control and providing a menu for selection a tool type of the application,

wherein a radius of the arc shaped curve is at least a radius of a menu of one of the controls,

wherein a control closest to a display area is positioned along the curve at least a radius of a menu of the control, and

wherein a marking menu associated with one of the controls has a layout where a downward stroke brings up additional tool palettes and/or dialogs.

46. A method, comprising:

mapping controls of an graphical user interface in an arc shape at a location responsive to an approach arc and with a radius responsive to an underlying menu activatable via one of the controls; and

allowing a user to activate the controls.

47. A method as recited in claim 46, wherein the location comprises a display area corner.

48. A method as recited in claim 46, wherein the corner is lower right corner for a left-handed person and a lower left corner for a right-handed person

49. A method as recited in claim 46, wherein the mapping maps controls on the arc responsive to a function of the controls.

50. A method as recited in claim 46, further comprising minimizing the interface responsive to activation of a minimize control.

51. A method as recited in claim 46, wherein the allowing comprises:

displaying a menu upon a touch input and allowing a user to select an item of the menu;

displaying a menu and performing an interaction upon a dwell input; and

performing a function upon a stroke input.

52. A method as recited in claim 51, wherein if a user is inking from a drawing canvas and the inking crosses into the menu, inking still occurs on the canvas.

53. A computer readable storage for controlling a computer by mapping controls of a graphical user interface in an arc shape at a location responsive to an approach arc and with a radius responsive to an underlying menu activatable via one of the controls and allowing a user to activate the controls.

54. An apparatus, comprising:

a display; and

a processor positioning a graphical user interface of multiple controls in a lower right corner of the display, the interface having an interface arc shape and positioning the controls on the interface arc at positions responsive to a natural motion arc of a user when moving a hand from a center of the display toward the corner.

55. An apparatus as recited in claim 54, wherein the processor positions the controls responsive to a function of the controls.

56. An apparatus as recited in claim 54, further comprising a stylus-based input system coupled to the processor and the display, and activating the controls responsive to a tap of a stylus on one of the controls, a dwell of the stylus over one of the control and a stroke of the stylus on one of the controls.

57. A display, comprising:

a control zone for a function of an interface; and

an interface element graphic aligned with the control zone and indicating the function with the interface graphic and control zone aligned to a natural user motion.

58. A display as recited in claim 57, wherein the alignment orients the graphic and zone with the motion.

59. A display as recited in claim 57, wherein the alignment follows the natural user motion.

60. A display as recited in claim 57, wherein the alignment positions the graphic and zone at a location accessible via the natural user motion.

61. A display as recited in claim 57, wherein the natural user motion comprises a curve determined by a stroke of the user on the display.

62. A display as recited in claim 61, wherein the curve includes natural motion variations.

63. A display as recited in claim 61, wherein the user natural motion stroke comprises one of an elbow motion curve, a wrist motion curve, a finger motion curve, a shoulder motion curve and a combination of two or more of the curves.

64. A display as recited in claim 63, wherein the curve is a curve determined by a single user.

65. A display as recited in claim 57, further comprising an interface location at which the zone and graphic are positioned.

66. A display as recited in claim 65, wherein the interface location is specified by a cursor positioned by the user.

67. A graphical user interface, comprising:

a cursor positioned on a display by a user at a location; and

a function control positioned on the display responsive to the location of the cursor, having a interface graphic indicating a function of the control and having a shape conforming to a motion arc of a hand caused by motion of an arm about an elbow of the user.

68. An interface as recited in claim 67, wherein the control comprises plural controls and the controls are aligned along the arc.

69. An interface as recited in claim 68, wherein a default control is positioned under the cursor.

70. An interface as recited in claim 68, wherein the controls can be one of re-oriented and moved.

71. An interface as recited in claim 68, wherein the controls are one or oriented and shaped to conform to a wrist arc caused by a hand moving about a wrist of the user

72. An interface as recited in claim 67, wherein the control comprises plural controls and the controls are aligned along an arc intersecting the motion arc at 90 degrees.

73. An interface as recited in claim 67, wherein the control comprises plural controls and the shape of the sides of each of the controls is one of rectilinear, arc shaped, wedge shaped and triangular shaped.

74. An interface as recited in claim 67, further comprising an overflow interface positioned responsive to the motion arc.

75. An interface as recited in claim 67, wherein text of the control is rectilinear aligned with a display.

76. An interface as recited in claim 75, wherein the overflow interface is natural motion arc shaped.

77. An interface as recited in claim 68, wherein the control is oriented to an extended arc.

78. A graphical user interface for a tablet personal computer having a stylus input system, comprising:

a cursor positioned on a display by a user at a location on the display designed by the stylus;

a function control positioned on the display responsive to the location of the cursor, having a interface graphic indicating a function of the control and having a graphic shape and position conforming to a natural motion arc of a hand caused by motion of an arm about an elbow and of the hand moving about a wrist of the user, having plural controls with a default control positioned under the cursor, controls aligned along the arc and controls aligned along a counter arc intersecting the motion arc at 90 degrees and where the controls are shaped responsive to the natural motion arc with natural variations; and

an overflow interface and shaped positioned responsive to the motion arc.

79. A method, comprising:

determining a position of a cursor as designated by the user; and

positioning an arc shaped graphical user interface responsive to the position where the arc of the shape is defined by a natural user motion.

80. A method as recited in claim 79, further comprising determining whether the user has specified a custom arc and positioning one of a custom and standard arc shaped interface responsive to the determination.

81. A computer readable storage for controlling a computer by determining a position of a cursor as designated by the user, and positioning an arc shaped graphical user interface responsive to the position where the arc of the shape is defined by a natural user motion.

82. A method, comprising:
allowing a user to make strokes with an input device;
determining an arc from the strokes; and
laying out a graphical user interface to conform to the arc.

83. A method as recited in claim 82, further comprising:
determining a position of a cursor specified by the user; and
positioning the interface responsive to the position; and
allowing the user to activate a function of the interface.

84. A method as recited in claim 82, wherein plural users are allowed to make strokes and the arc is determined from the strokes of the plural users.

85. An apparatus, comprising:
a display; and
a computer producing an arc shaped graphical user interface on the display where the arc of the shape is defined by a natural user motion.

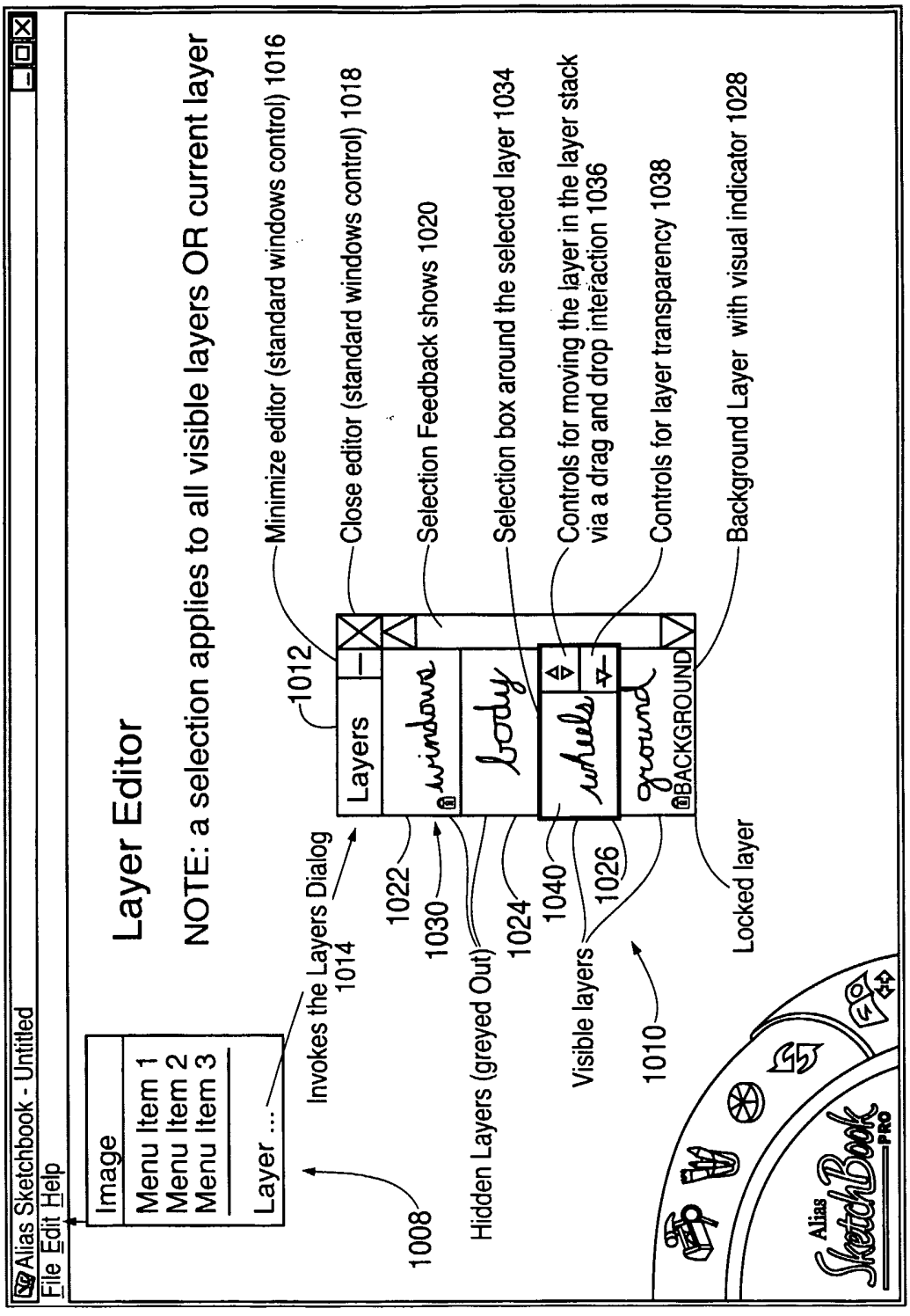

Alias Sketchbook - Untitled
File Edit Help
Image
Menu Item 1
Menu Item 2
Menu Item 3
Layer ...
Layer Editor
NOTE: a selection applies to all visible layers OR current layer
Minimize editor (standard windows control) 1016
Close editor (standard windows control) 1018
Selection Feedback shows 1020
Selection box around the selected layer 1034
Controls for moving the layer in the layer stack via a drag and drop interaction 1036
Controls for layer transparency 1038
Background Layer with visual indicator 1028
Invokes the Layers Dialog 1014
Hidden Layers (greyed Out)
Visible layers
Locked layer
1008
1010
1012
1022
1024
1026
1030
1040
Layers
BACKGROUND
Alias SketchBook PRO

FIG. 1001

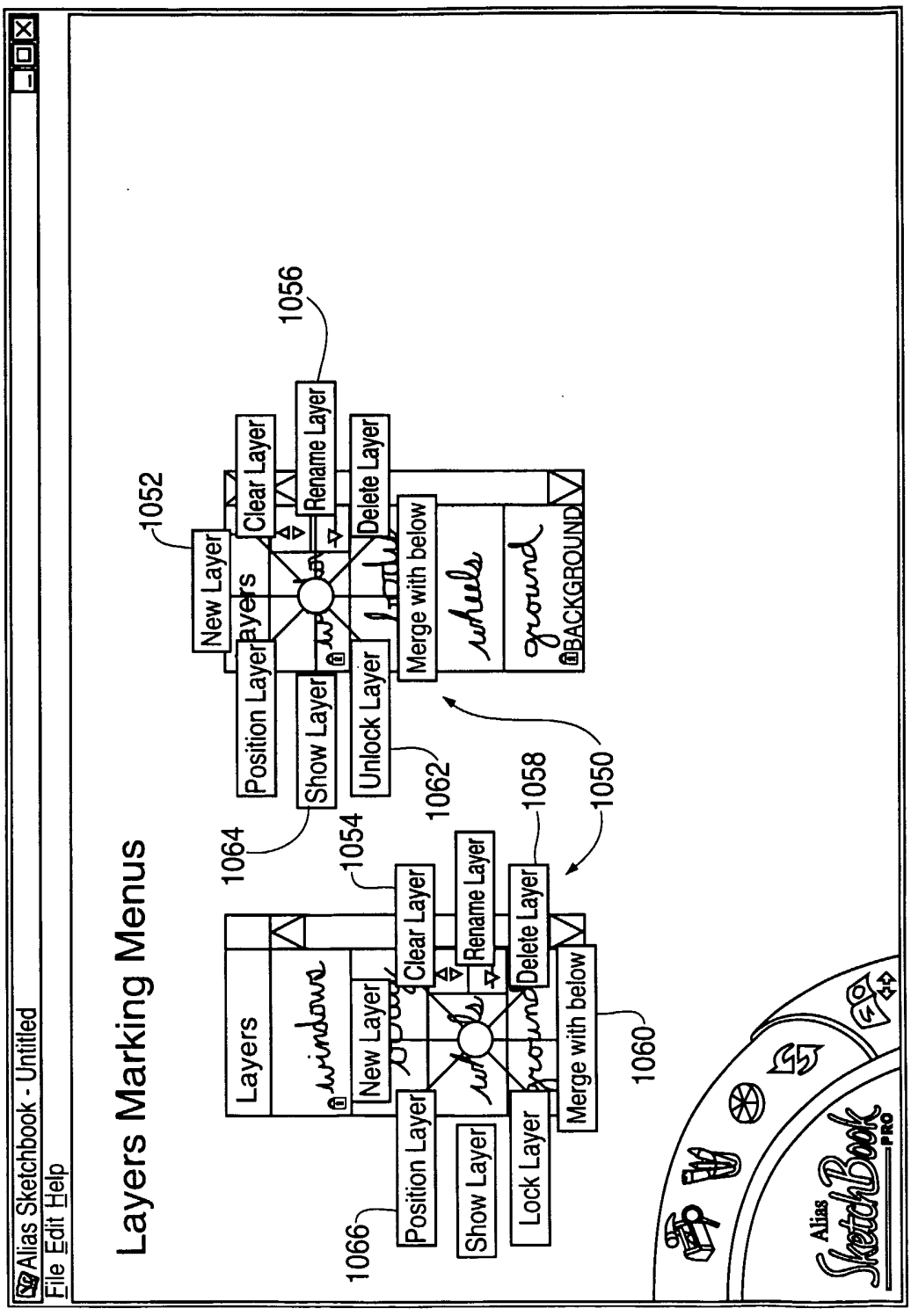

Alias Sketchbook - Untitled
File Edit Help
Layers Marking Menus
1052
1056
1064
1054
1062
1058
1050
1066
1060
New Layer
Clear Layer
Rename Layer
Delete Layer
Position Layer
Show Layer
Unlock Layer
Merge with below
BACKGROUND
Layers
Lock Layer
Alias SketchBook PRO

FIG. 1002

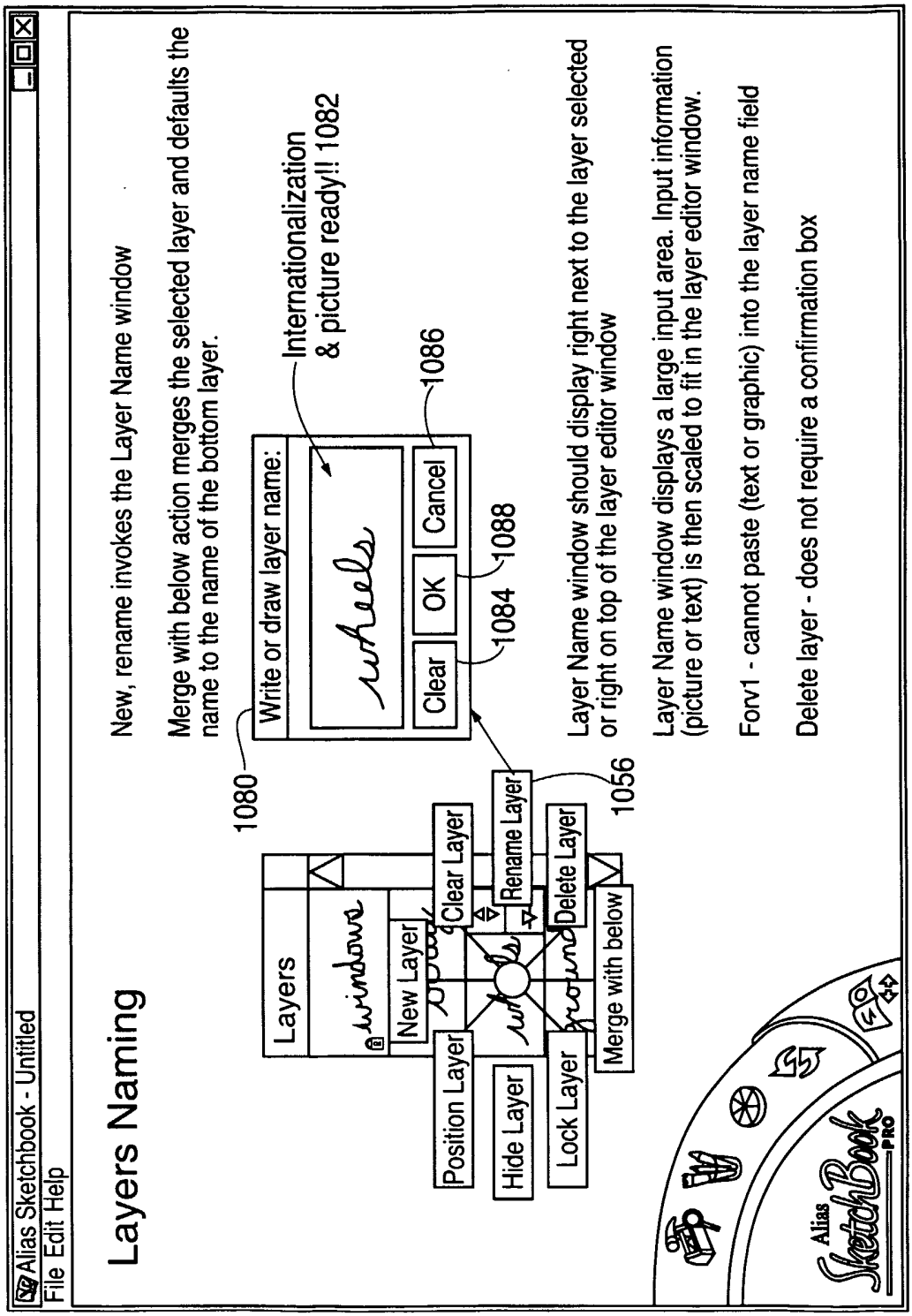
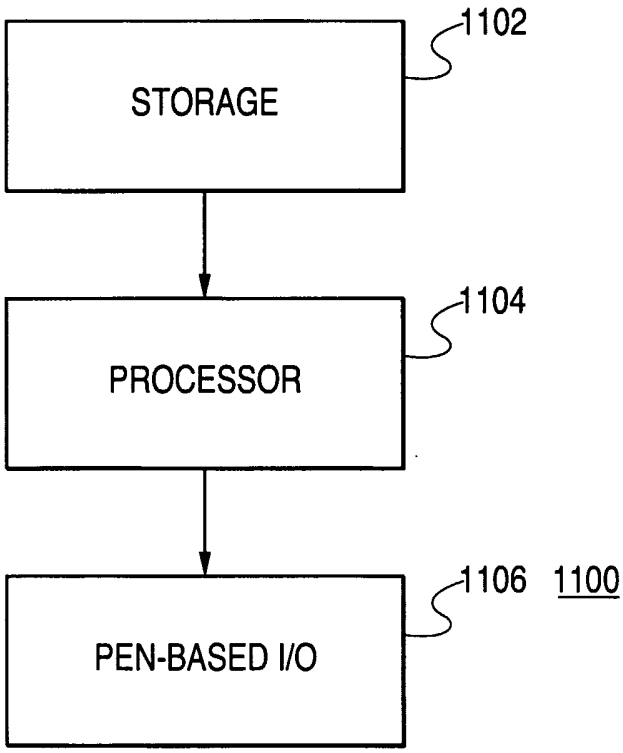

Alias Sketchbook - Untitled
File Edit Help
Layers Naming
New, rename invokes the Layer Name window
Merge with below action merges the selected layer and defaults the name to the name of the bottom layer.
Layers
New Layer
Clear Layer
Position Layer
Hide Layer
Rename Layer
Delete Layer
Lock Layer
Merge with below
1080
Write or draw layer name:
Internationalization & picture ready!! 1082
Clear
OK
Cancel
1086
1084
1088
1056
Layer Name window should display right next to the layer selected or right on top of the layer editor window
Layer Name window displays a large input area. Input information (picture or text) is then scaled to fit in the layer editor window.
Forv1 - cannot paste (text or graphic) into the layer name field
Delete layer - does not require a confirmation box
Alias SketchBook PRO

FIG. 1003

FIG. 1004



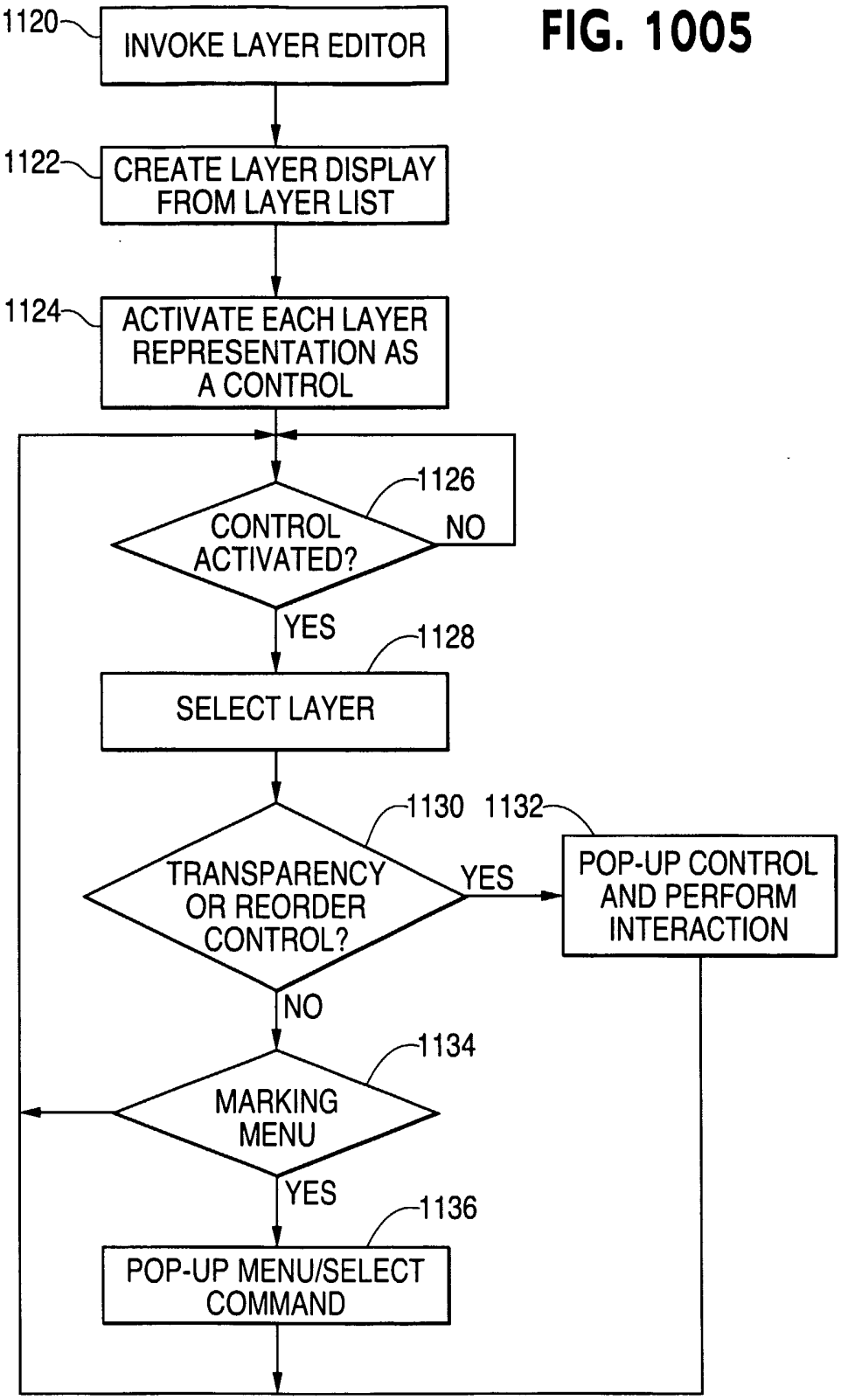

1120
INVOKE LAYER EDITOR
1122
CREATE LAYER DISPLAY FROM LAYER LIST
1124
ACTIVATE EACH LAYER REPRESENTATION AS A CONTROL
1126
CONTROL ACTIVATED?
NO
YES
1128
SELECT LAYER
1130
1132
TRANSPARENCY OR REORDER CONTROL?
YES
POP-UP CONTROL AND PERFORM INTERACTION
NO
1134
MARKING MENU
YES
1136
POP-UP MENU/SELECT COMMAND

FIG. 1005

FIG. 1006

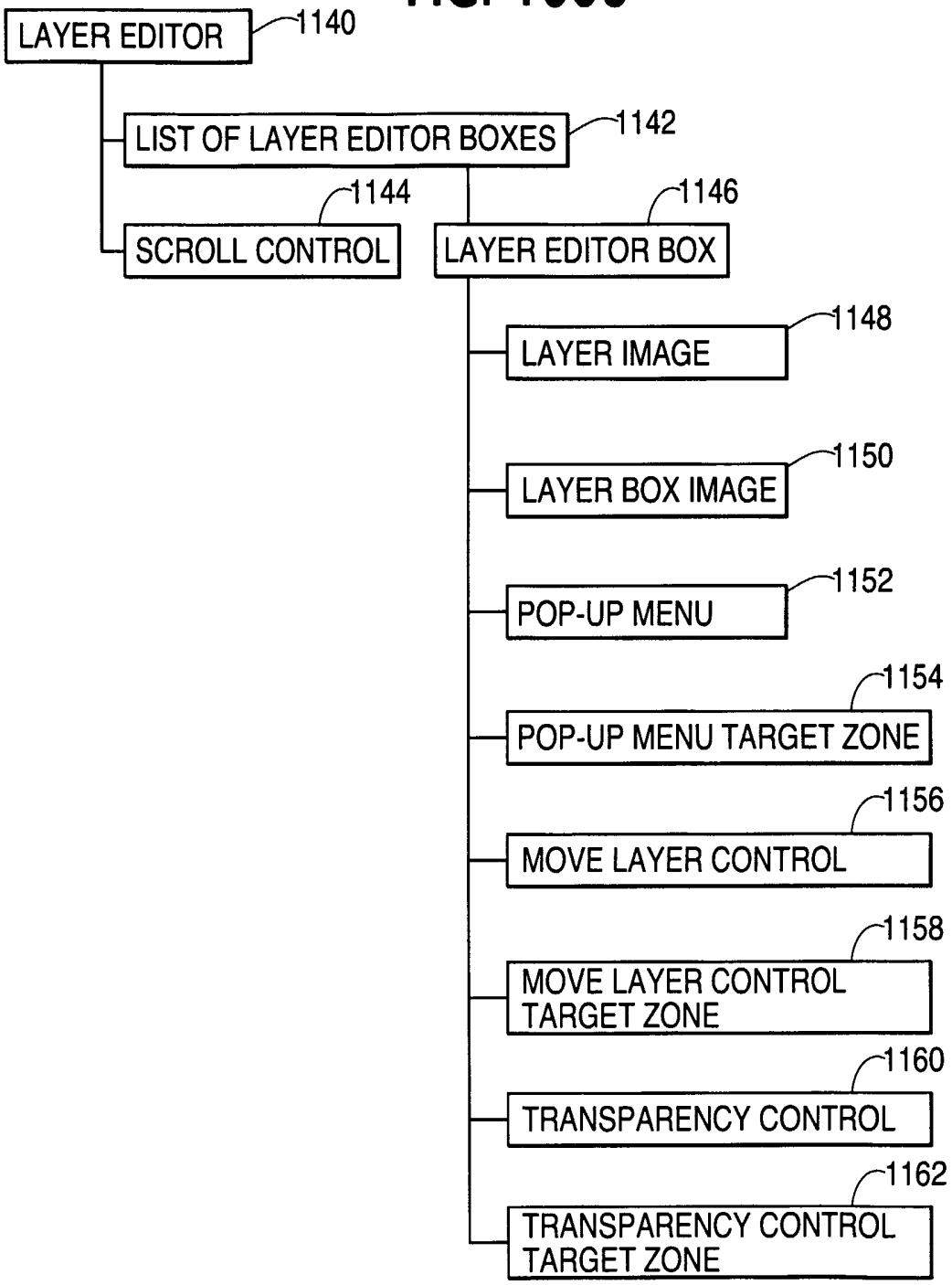

LAYER EDITOR
1140
LIST OF LAYER EDITOR BOXES
1142
SCROLL CONTROL
1144
LAYER EDITOR BOX
1146
LAYER IMAGE
1148
LAYER BOX IMAGE
1150
POP-UP MENU
1152
POP-UP MENU TARGET ZONE
1154
MOVE LAYER CONTROL
1156
MOVE LAYER CONTROL TARGET ZONE
1158
TRANSPARENCY CONTROL
1160
TRANSPARENCY CONTROL TARGET ZONE
1162

FIG. 1007

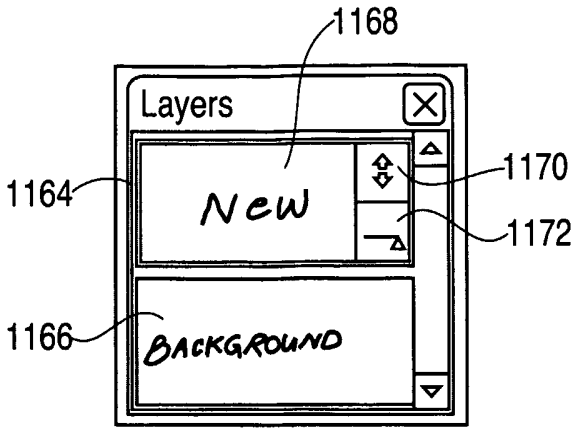


FIG. 1008

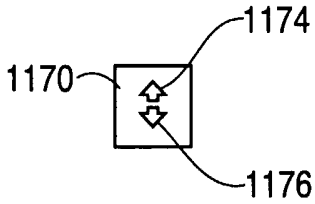


FIG. 1009

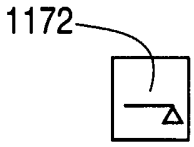


FIG. 1010

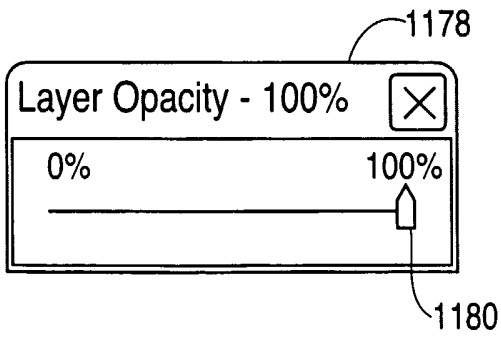


FIG. 1011

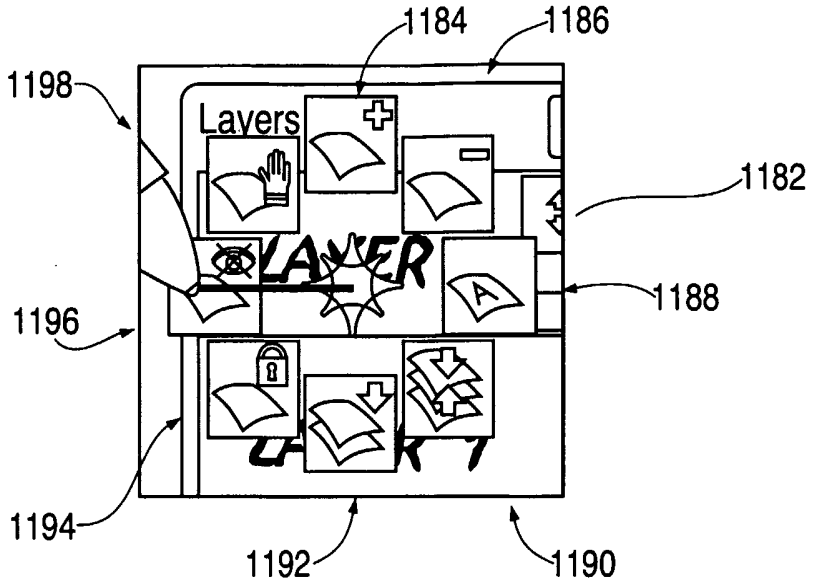


FIG. 1012

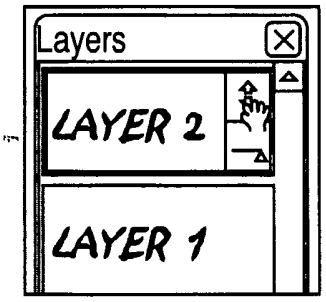


FIG. 1013

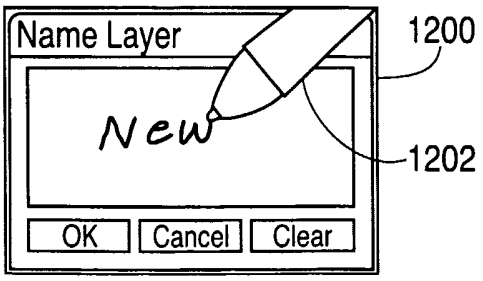


FIG. 1014

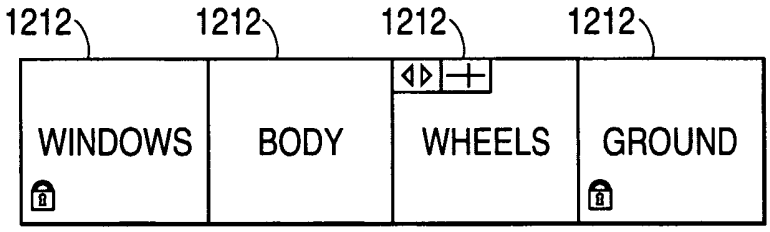


FIG. 1015

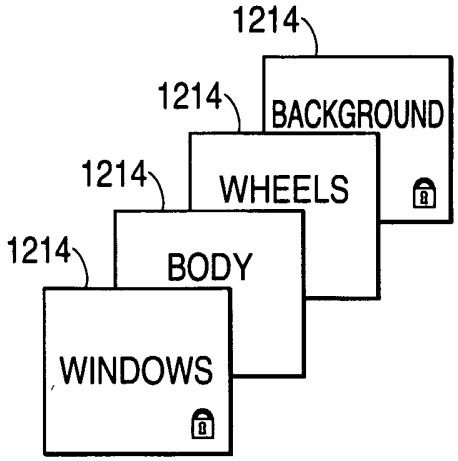


FIG. 1016

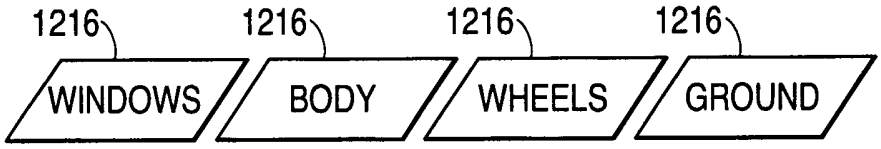


FIG. 1017

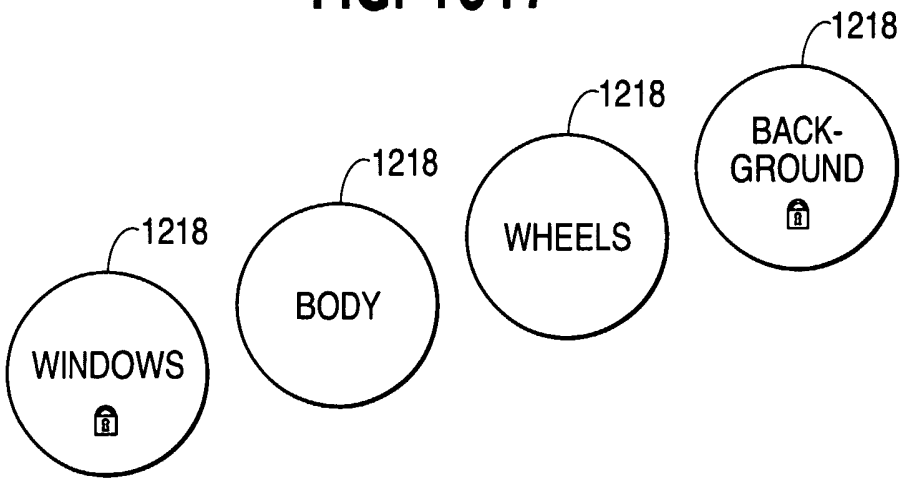


FIG. 1018

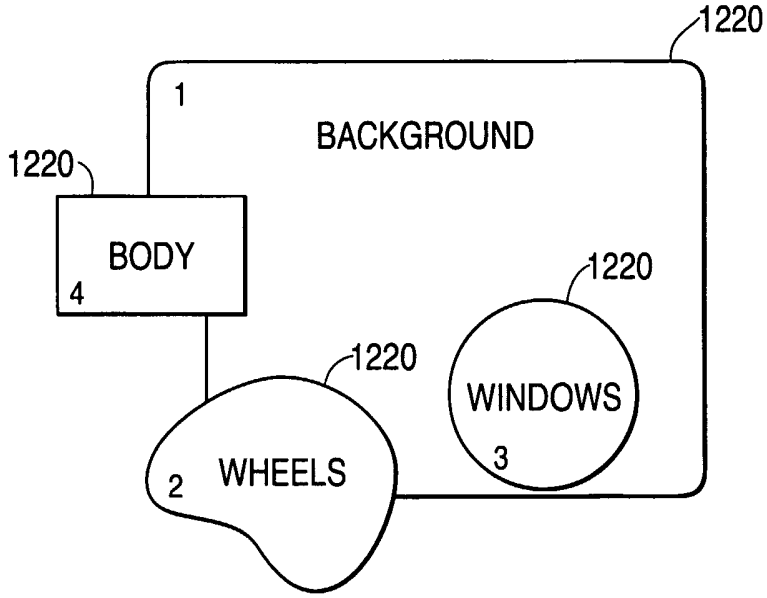


FIG. 1019

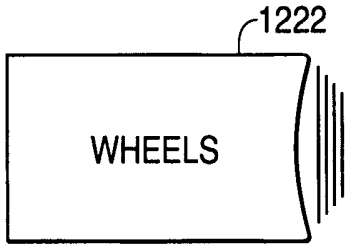


FIG. 1020

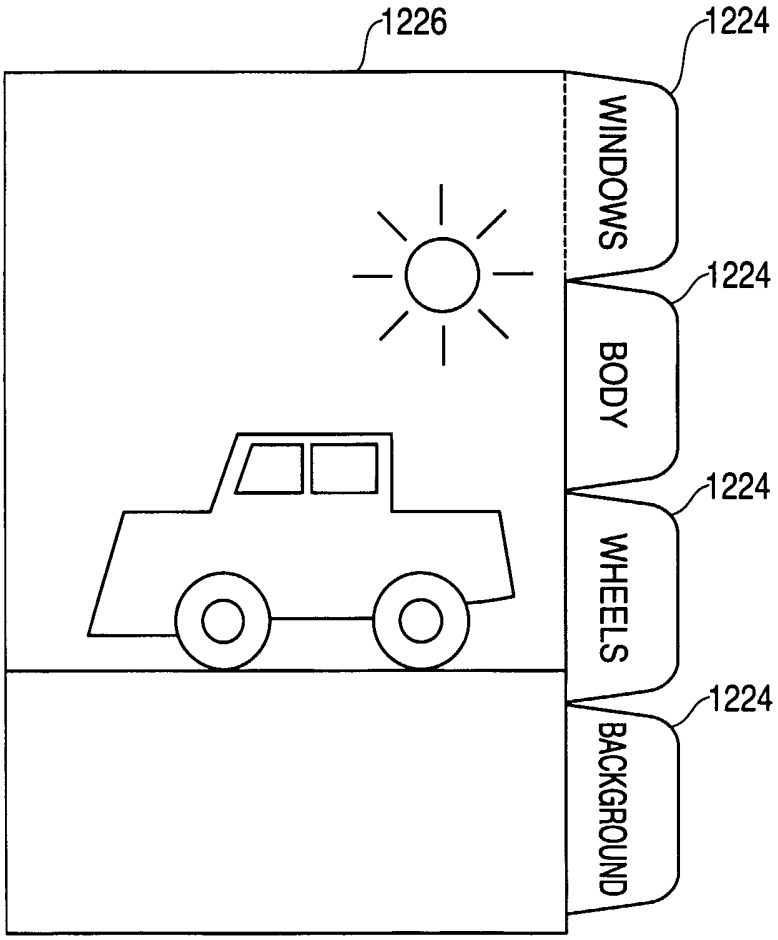
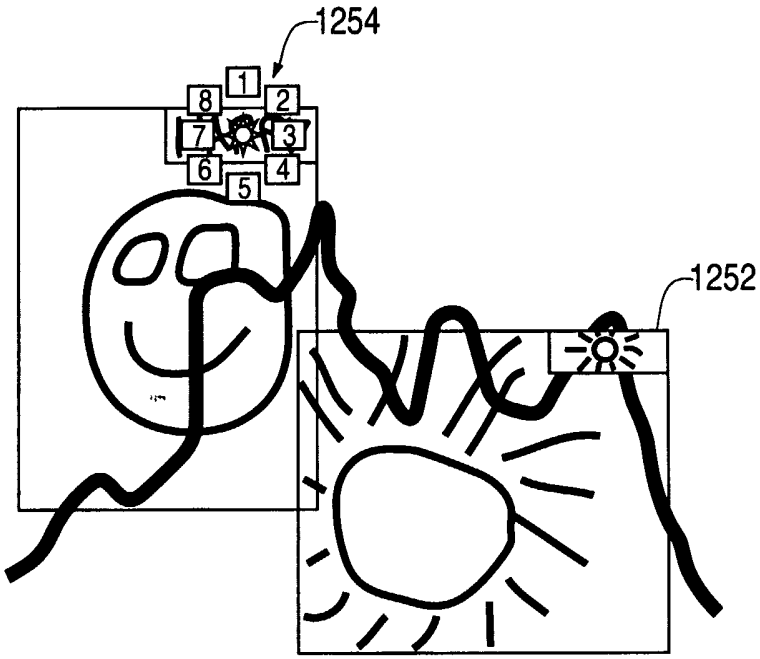


FIG. 1021



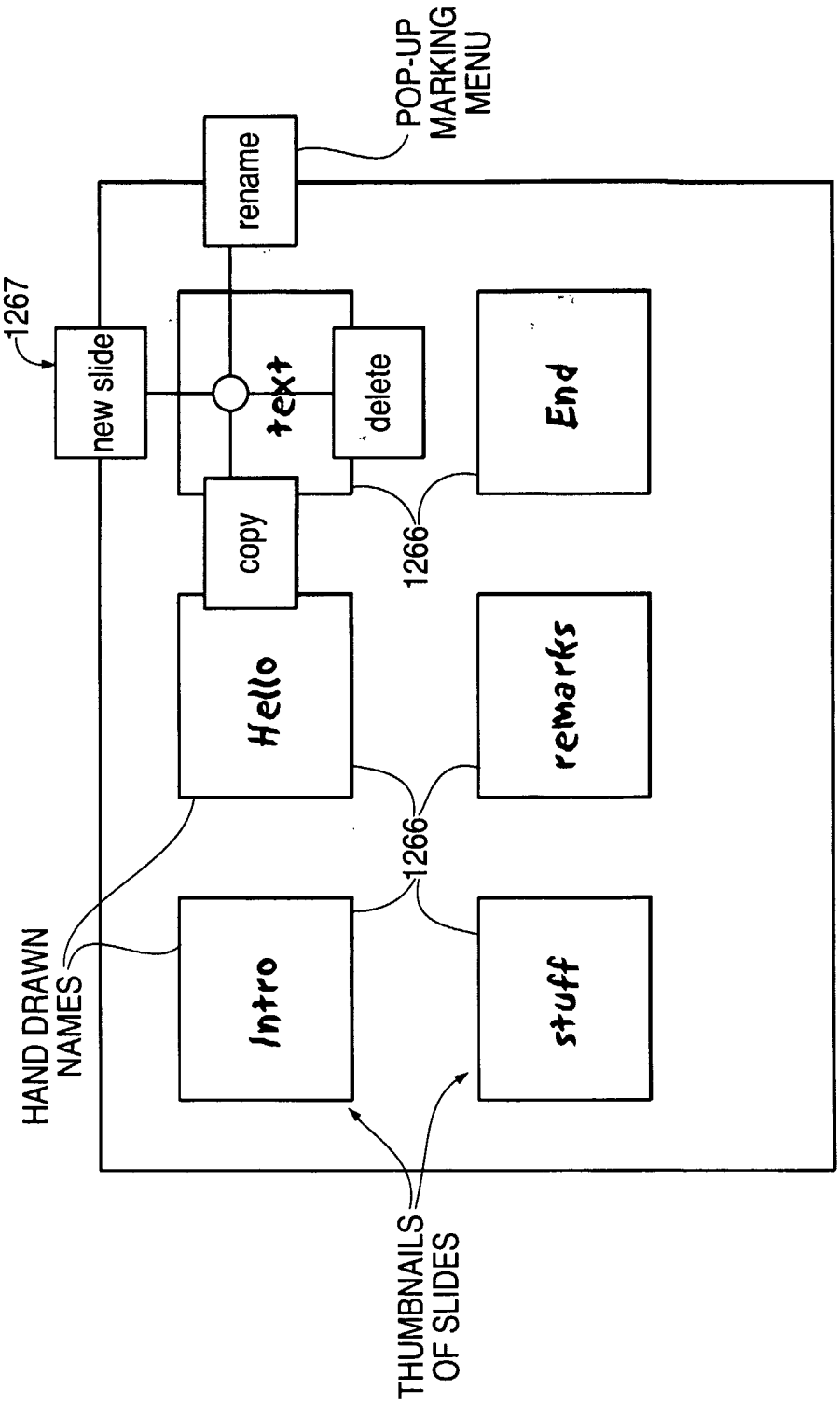

POP-UP MARKING MENU
rename
1267
new slide
text
delete
End
copy
1266
Hello
remarks
1266
HAND DRAWN NAMES
Intro
stuff
THUMBNAILS OF SLIDES

FIG. 1022

FIG. 1023

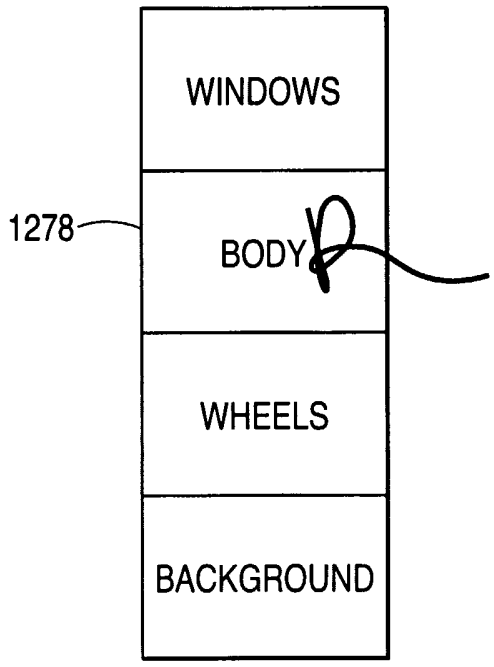


FIG. 1024

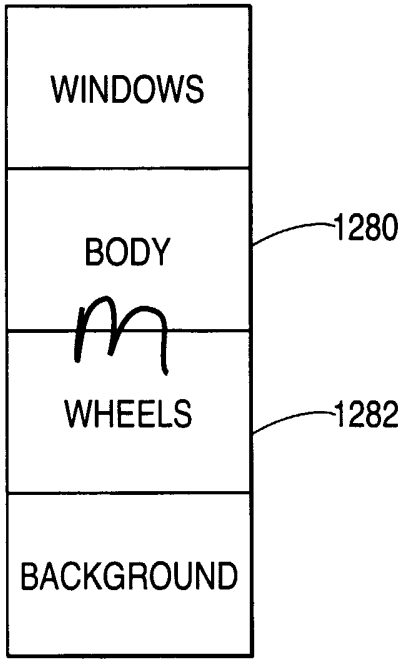


FIG. 1025

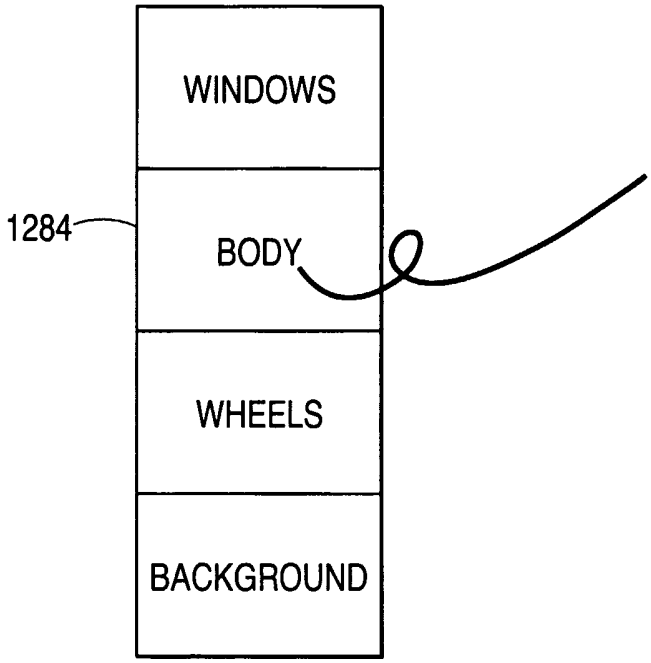
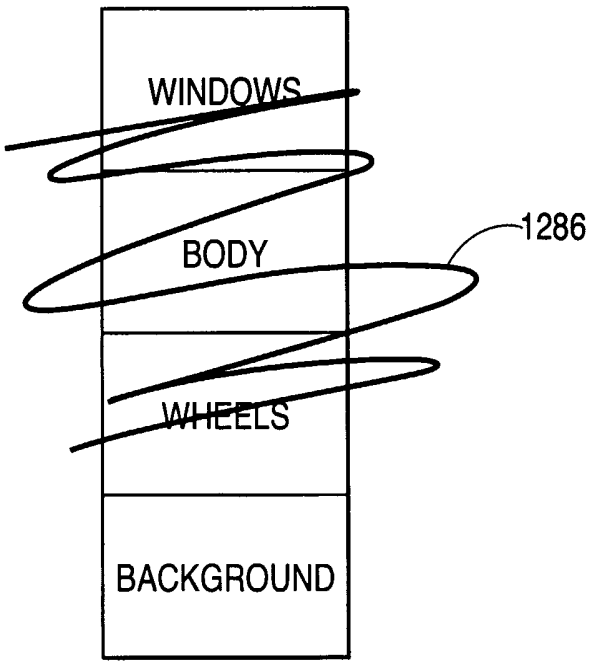


FIG. 1026



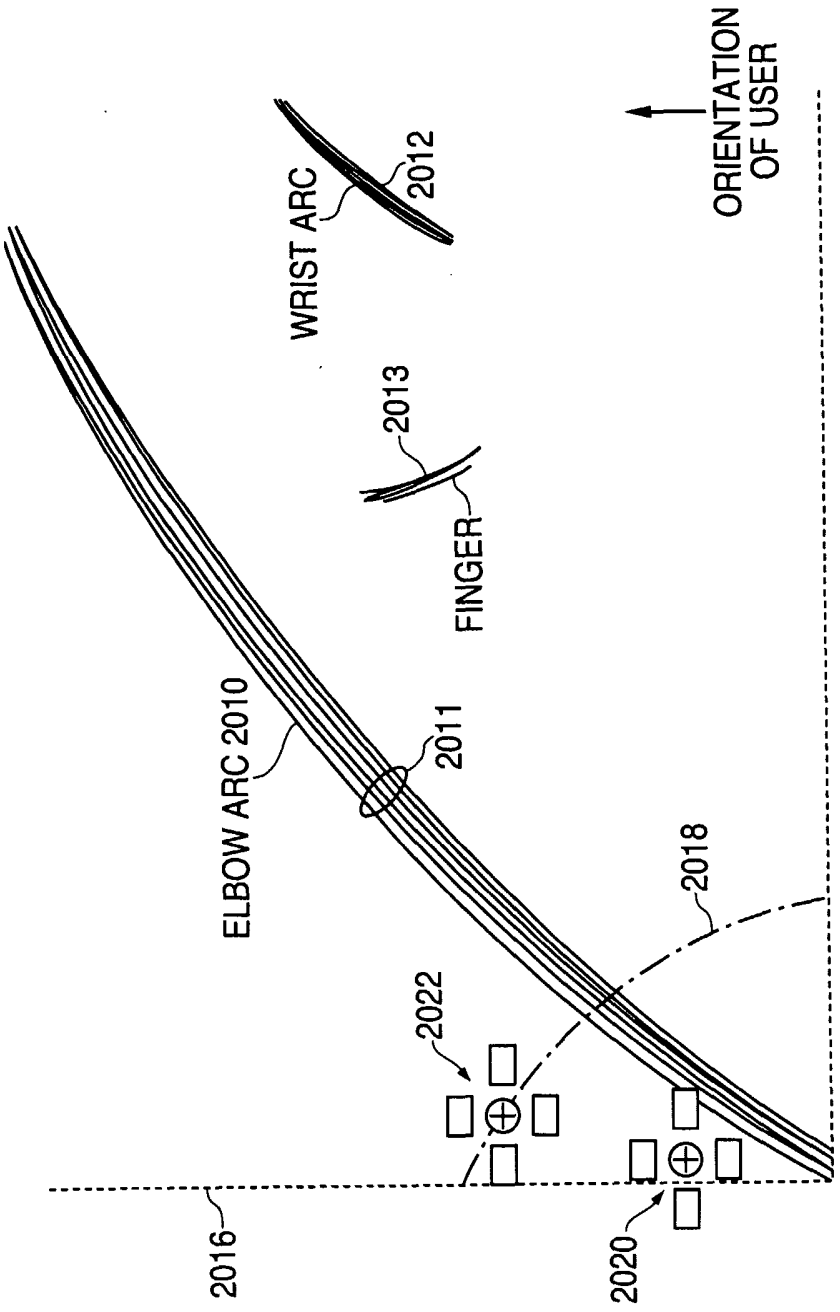

ORIENTATION OF USER
WRIST ARC
2012
2013
FINGER
ELBOW ARC 2010
2011
2018
2022
2016
2020
2014

FIG. 2001

FIG. 2002

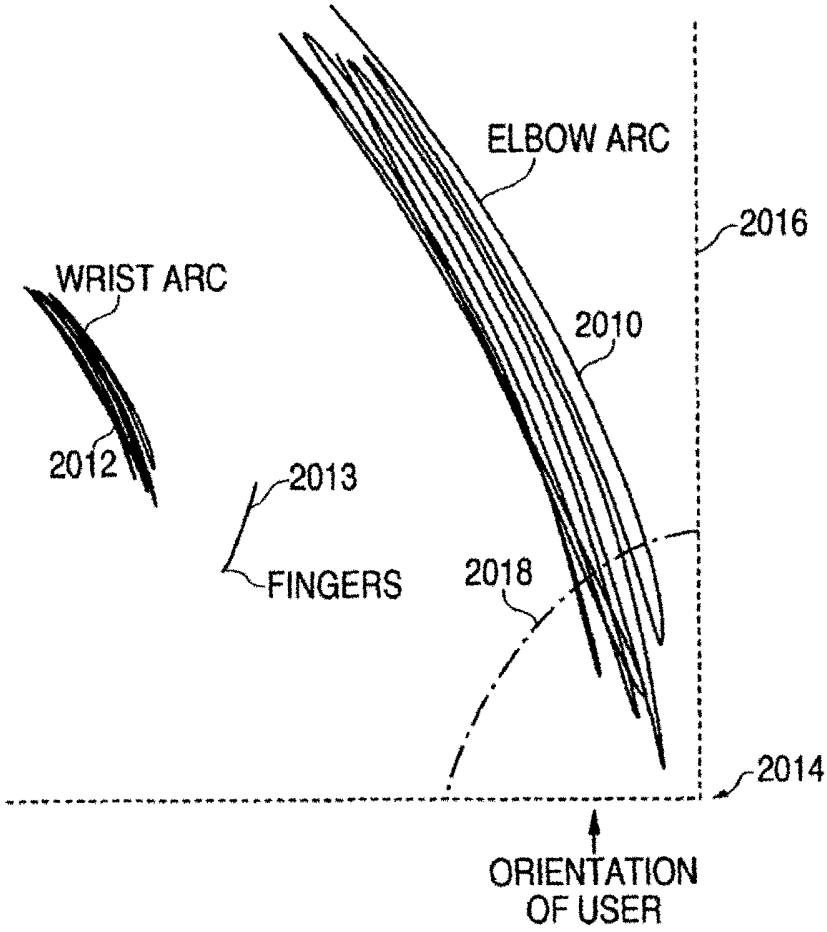

ELBOW ARC
2016
WRIST ARC
2010
2012
2013
FINGERS
2018
2014
ORIENTATION
OF USER

FIG. 2003

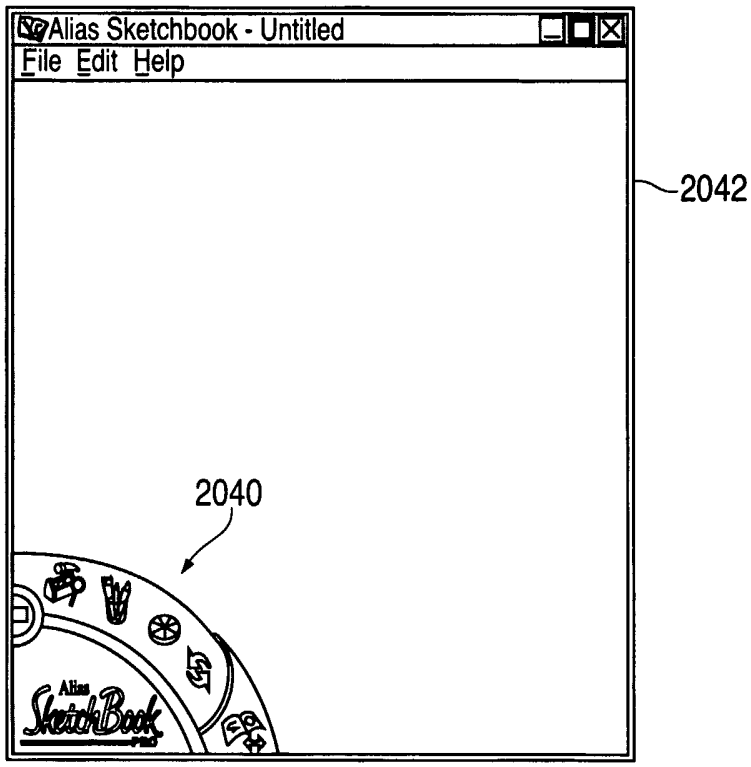

Alias Sketchbook - Untitled
File Edit Help
2042
2040
Alias
SketchBook

FIG. 2004

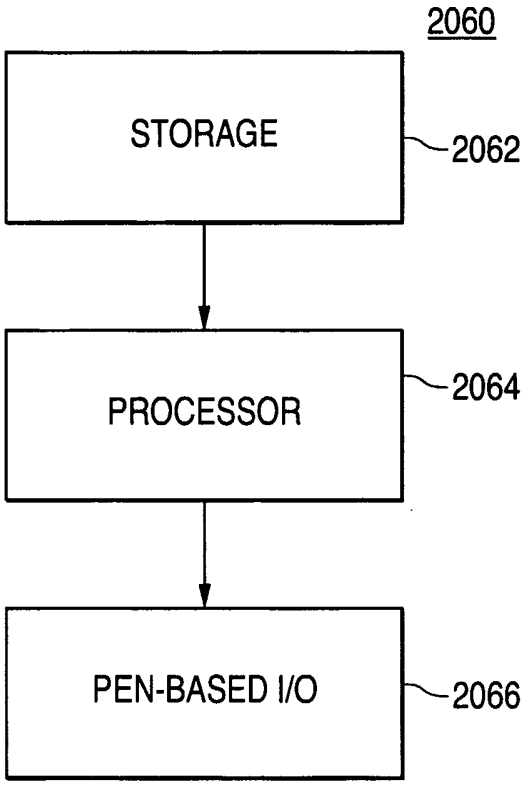


FIG. 2005

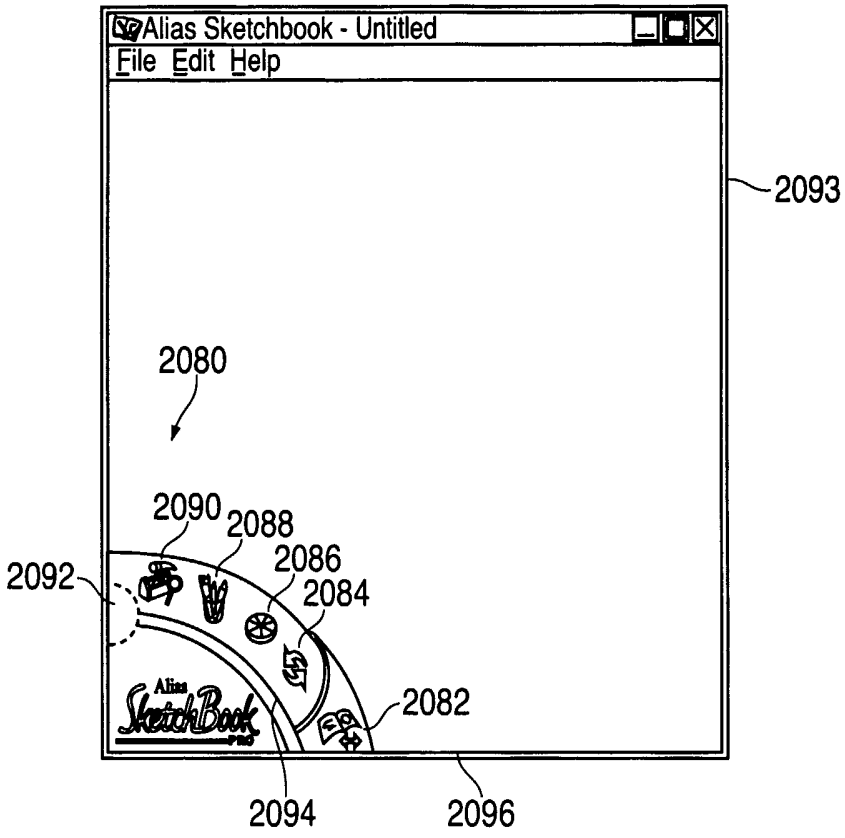

Alias Sketchbook - Untitled
File Edit Help
2093
2080
2090
2088
2086
2084
2092
2082
2094
2096

FIG. 2006

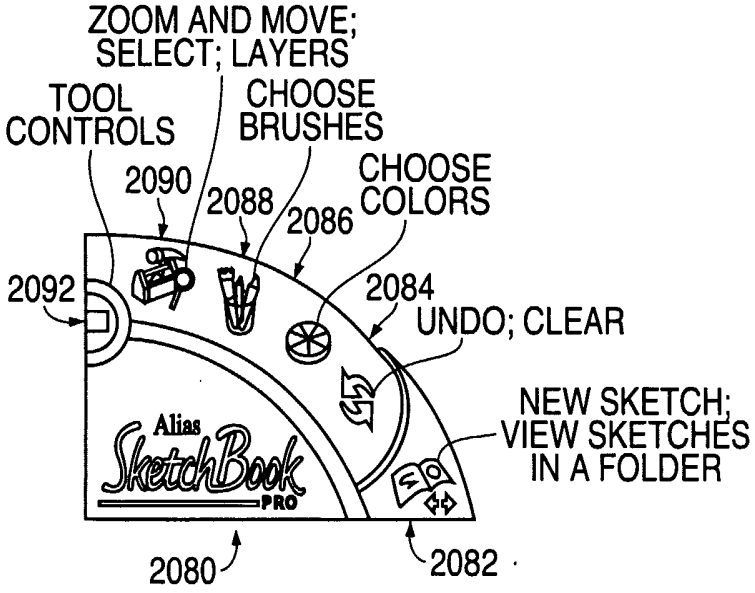

ZOOM AND MOVE; SELECT; LAYERS
TOOL CONTROLS
CHOOSE BRUSHES
CHOOSE COLORS
2090
2088
2086
2092
2084
UNDO; CLEAR
NEW SKETCH; VIEW SKETCHES IN A FOLDER
Alias
SketchBook
PRO
2080
2082

FIG. 2007

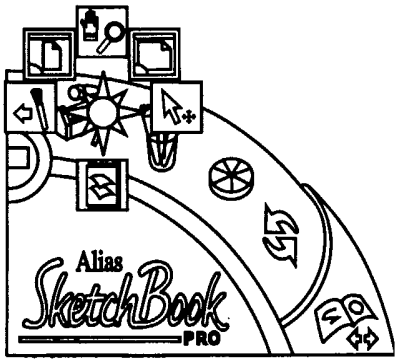


FIG. 2008

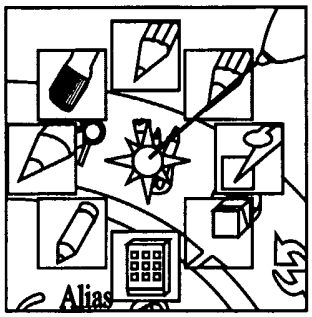


FIG. 2009

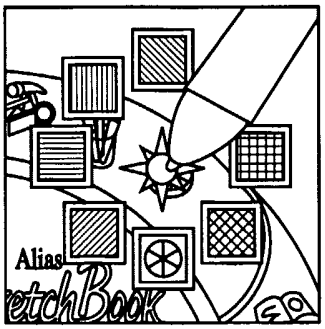


FIG. 2010

FIG. 2011

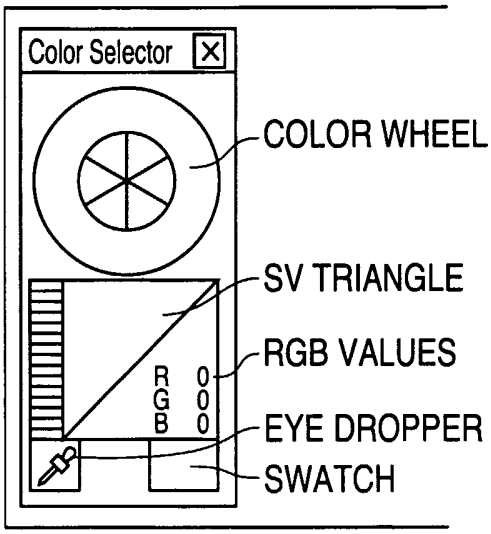


FIG. 2012

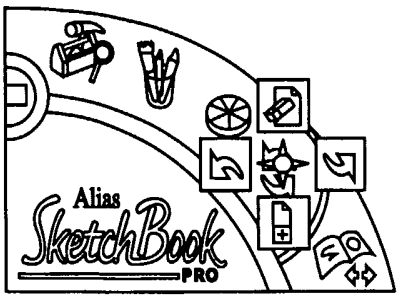


FIG. 2013

FIG. 2014

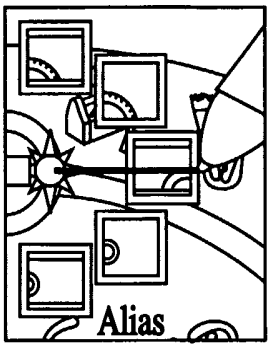


FIG. 2015

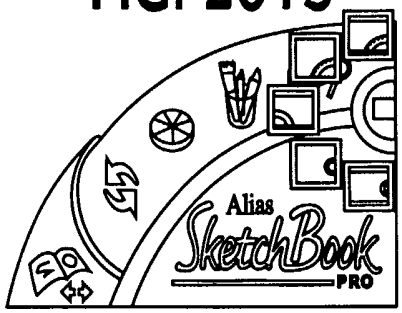


FIG. 2016

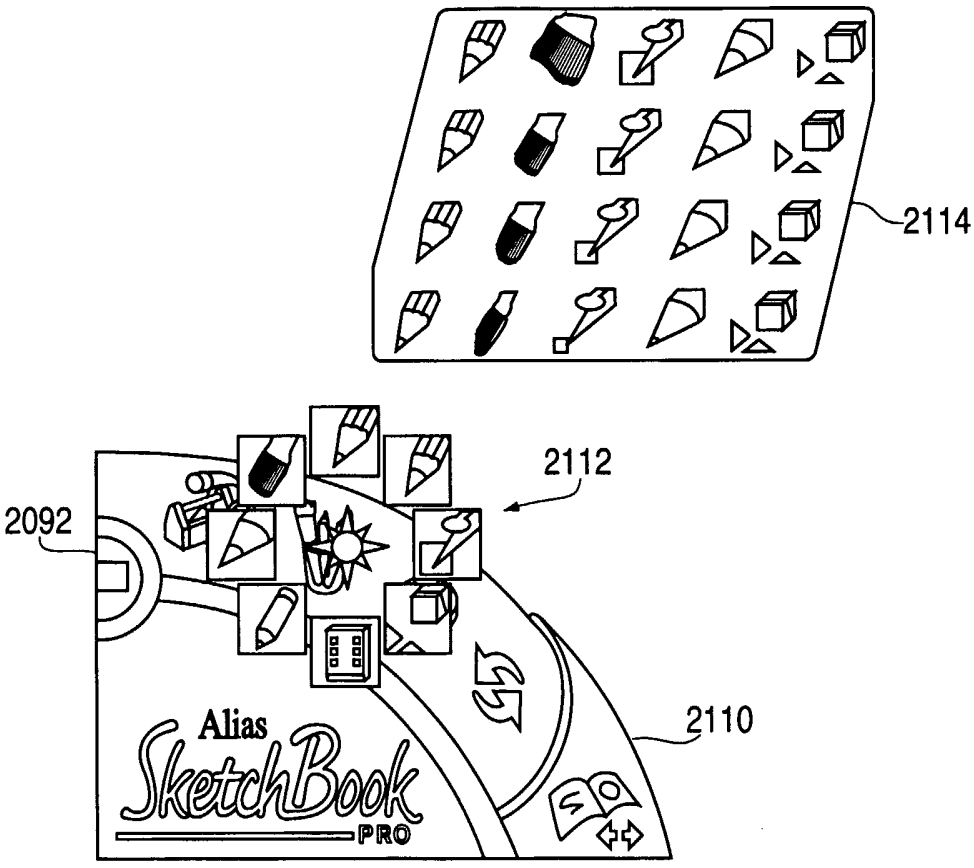


FIG. 2017

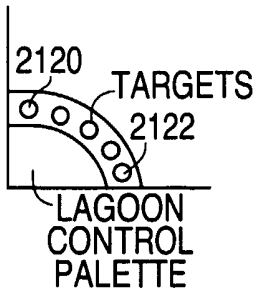


FIG. 2018

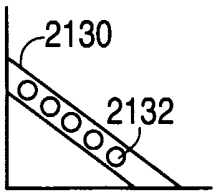


FIG. 2019

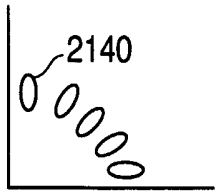


FIG. 2020

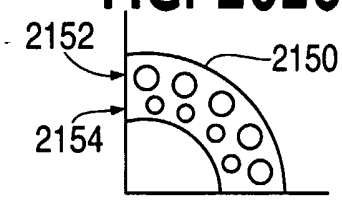


FIG. 2021

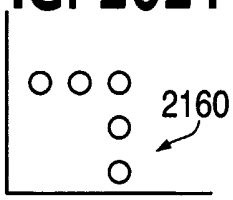


FIG. 2022

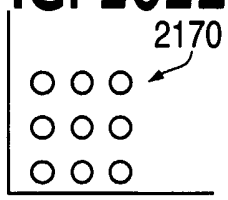


FIG. 2023

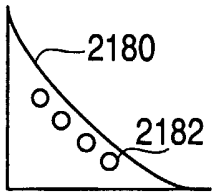


FIG. 2024

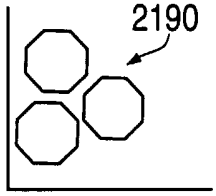


FIG. 2025

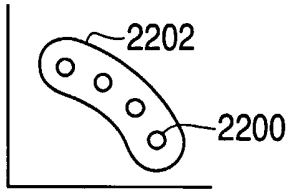


FIG. 2026

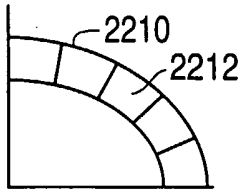


FIG. 2027

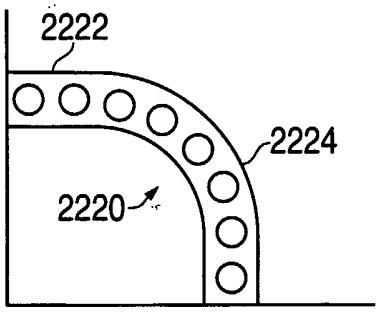


FIG. 2028

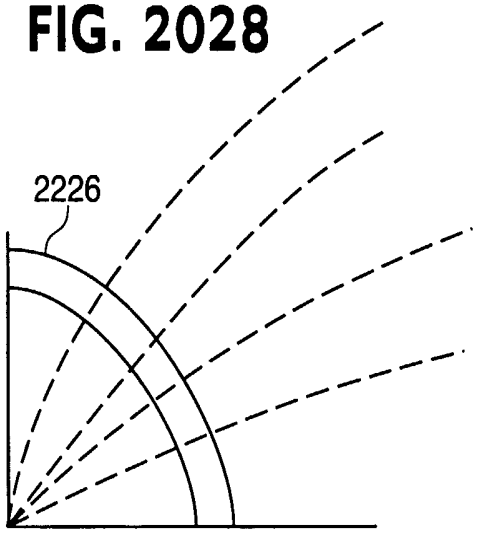
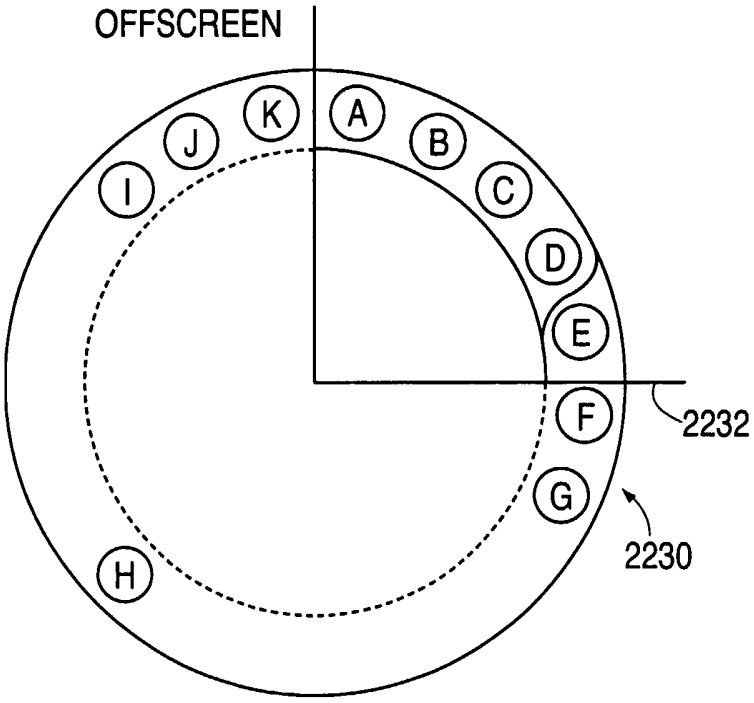


FIG. 2029



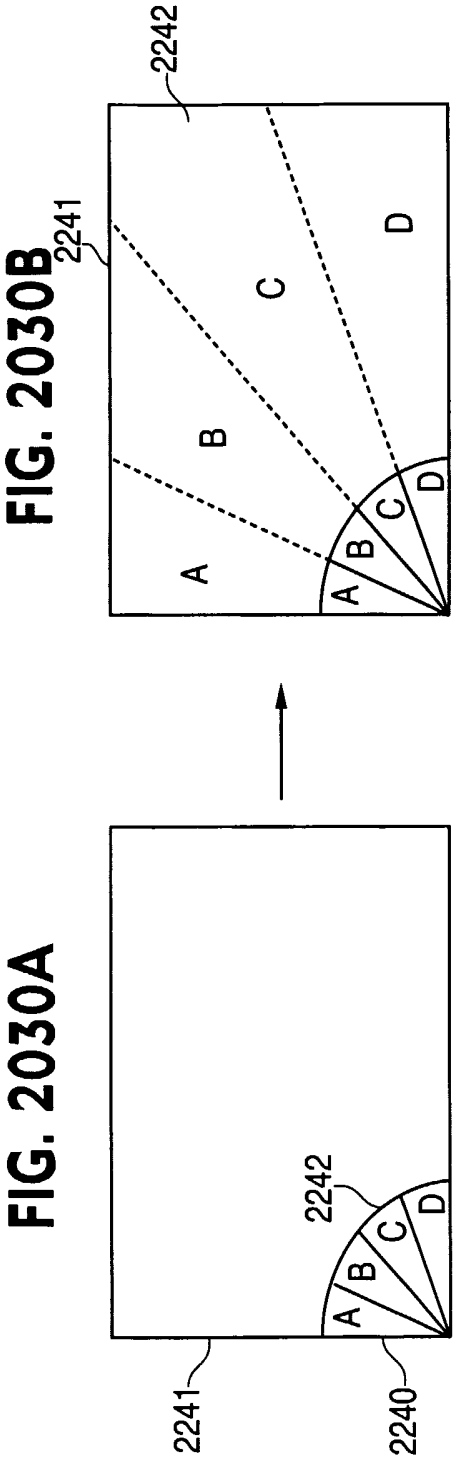


FIG. 2030A
2241
2240
2242
A
B
C
D
FIG. 2030B
2241
2242
A
B
C
D
A
B
C
D

FIG. 2031

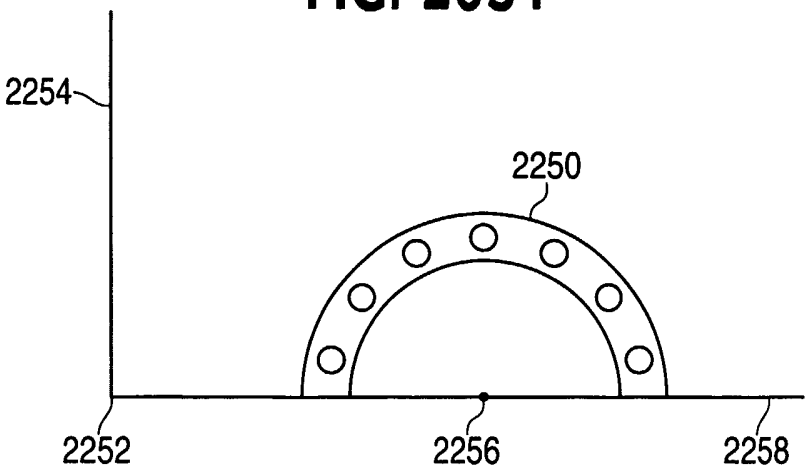


FIG. 2032

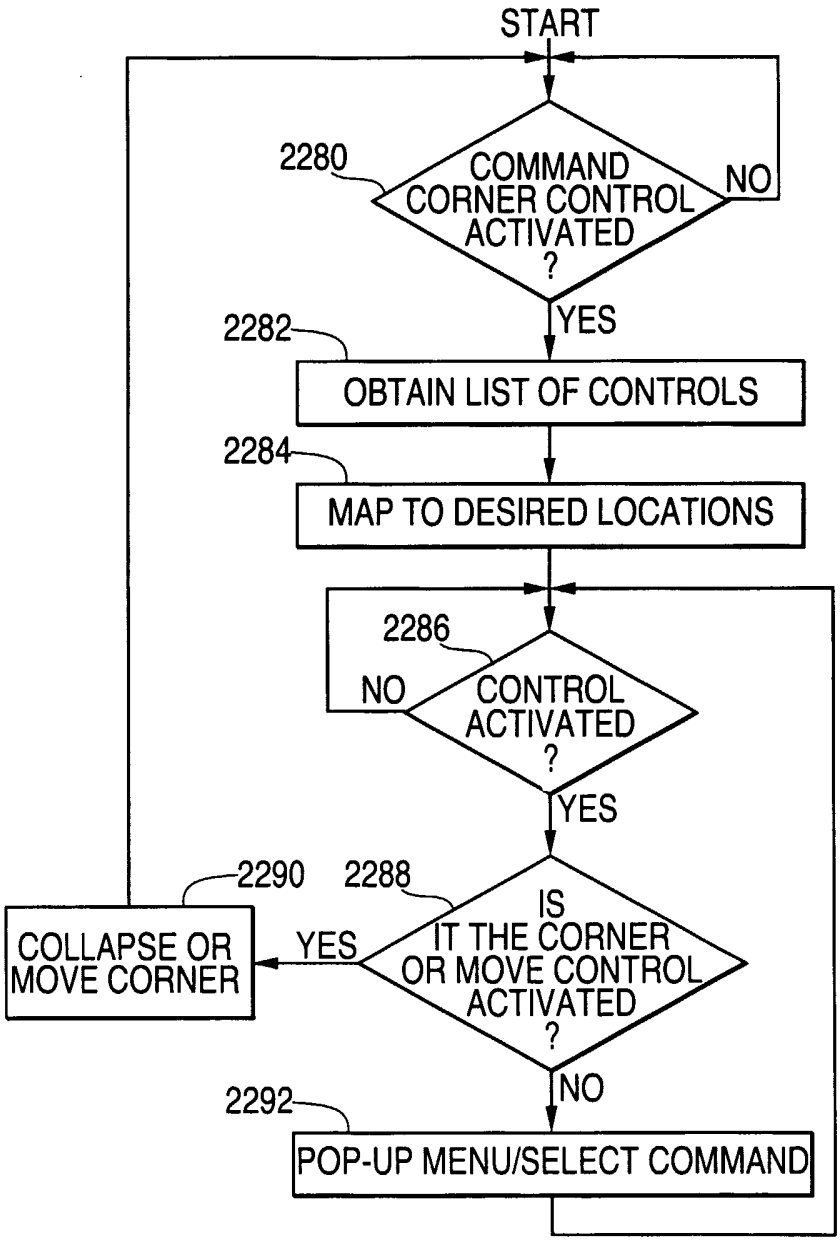

START
2280
COMMAND CORNER CONTROL ACTIVATED ?
NO
YES
2282
OBTAIN LIST OF CONTROLS
2284
MAP TO DESIRED LOCATIONS
2286
CONTROL ACTIVATED ?
NO
YES
2288
IS IT THE CORNER OR MOVE CONTROL ACTIVATED ?
YES
2290
COLLAPSE OR MOVE CORNER
NO
2292
POP-UP MENU/SELECT COMMAND

FIG. 2033

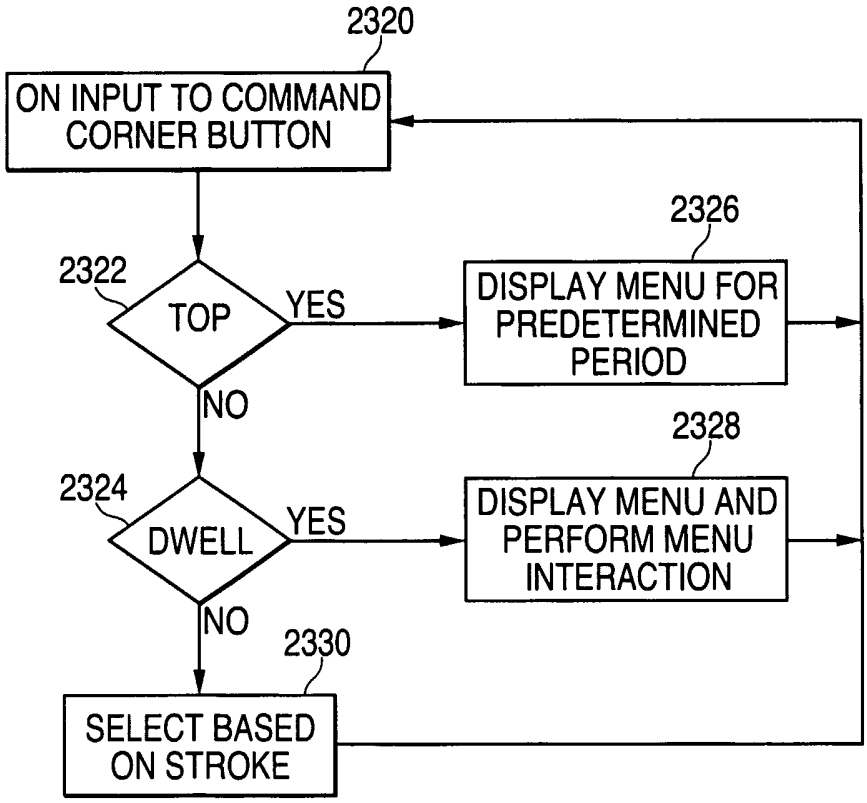

2320
ON INPUT TO COMMAND CORNER BUTTON
2322
TOP
YES
2326
DISPLAY MENU FOR PREDETERMINED PERIOD
NO
2324
DWELL
YES
2328
DISPLAY MENU AND PERFORM MENU INTERACTION
NO
2330
SELECT BASED ON STROKE

FIG. 2034A

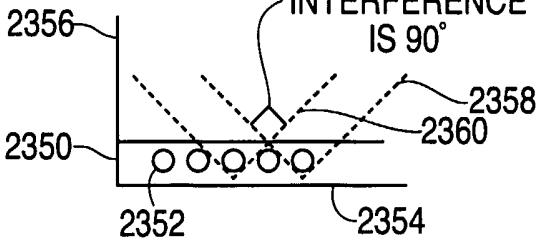


FIG. 2034B

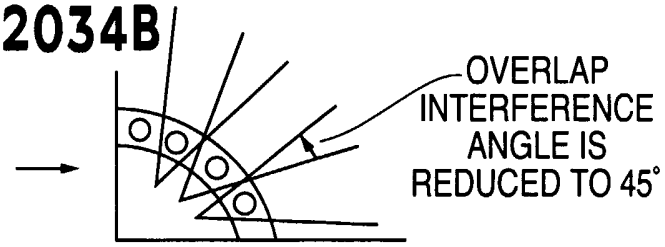


FIG. 2034C

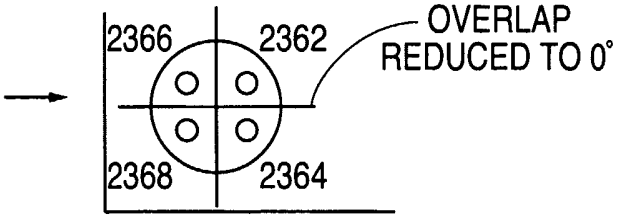


FIG. 2035A

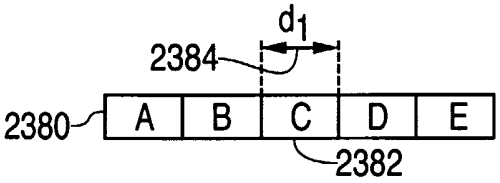


FIG. 2035B

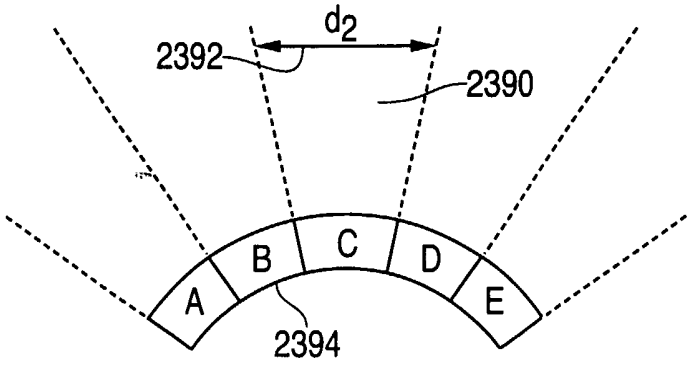


FIG. 2036A

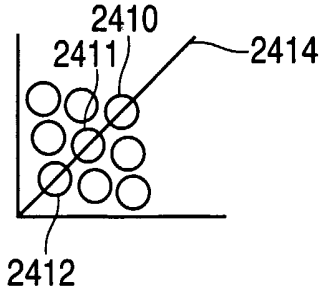


FIG. 2036B

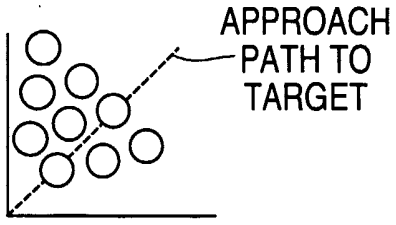


FIG. 3001

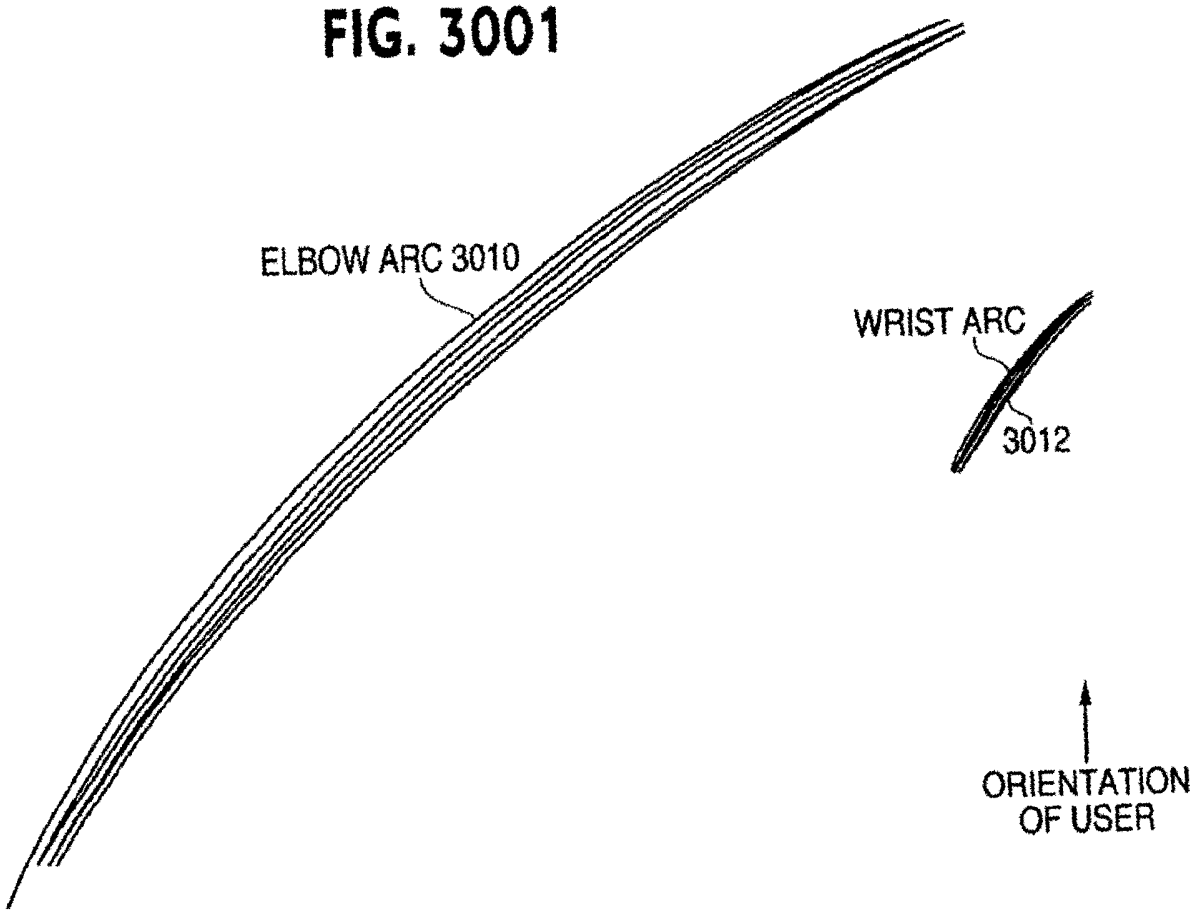


FIG. 3002

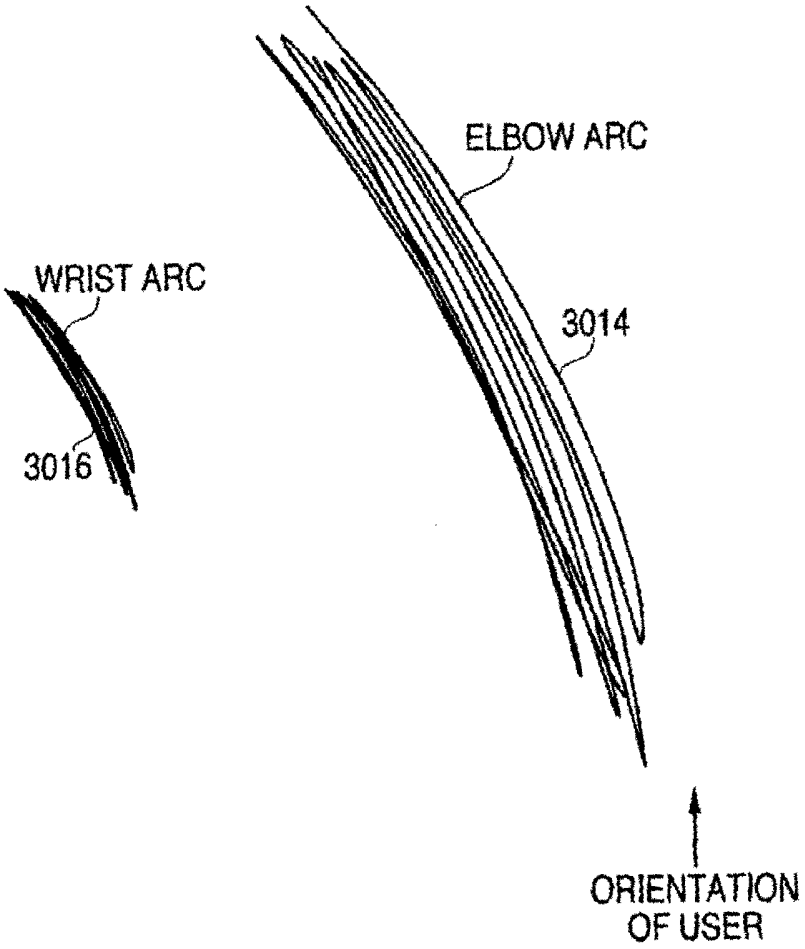


FIG. 3003

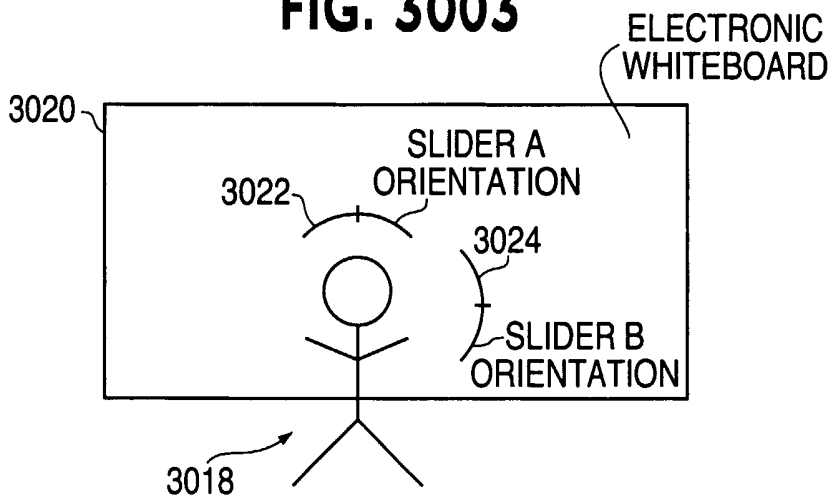

ELECTRONIC WHITEBOARD
3020
SLIDER A ORIENTATION
3022
3024
SLIDER B ORIENTATION
3018

FIG. 3004

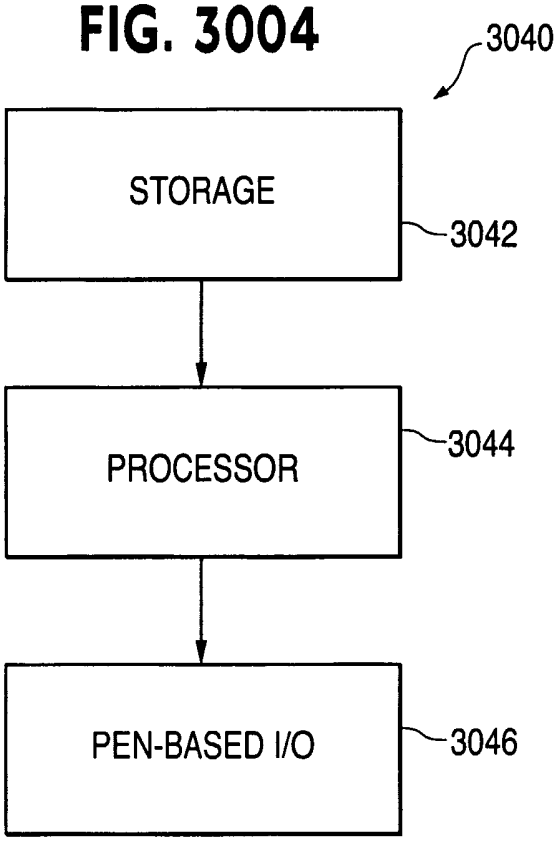


FIG. 3005

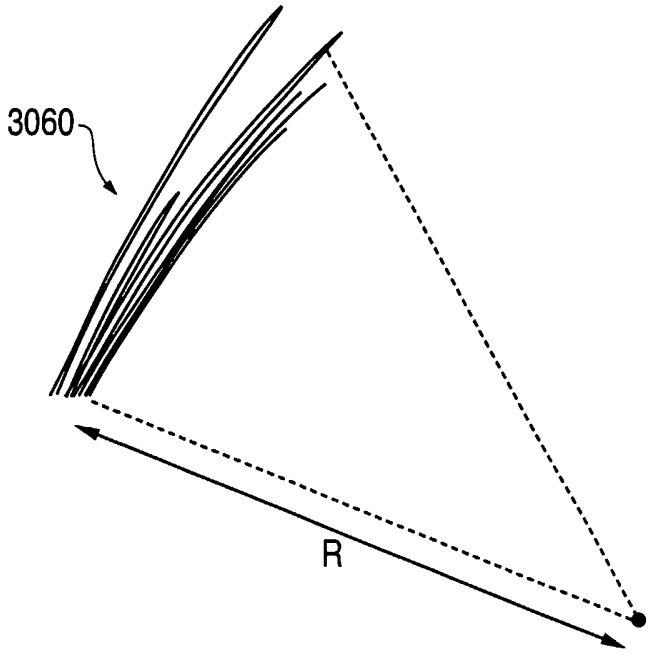


FIG. 3006

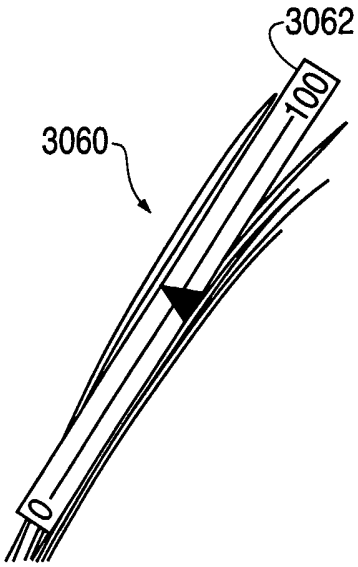


FIG. 3007

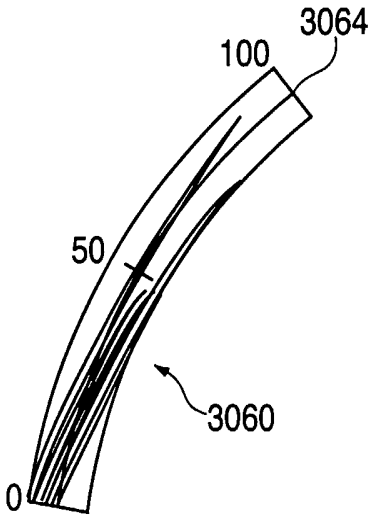


FIG. 3008

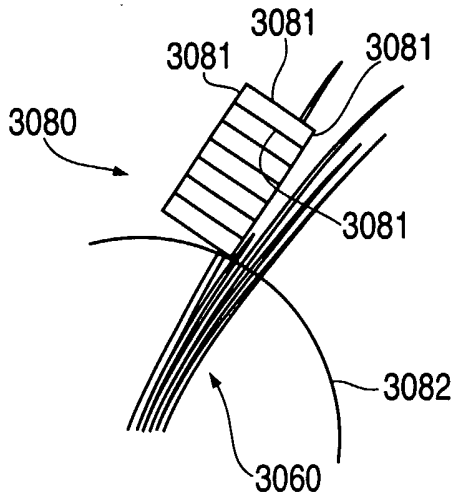


FIG. 3009

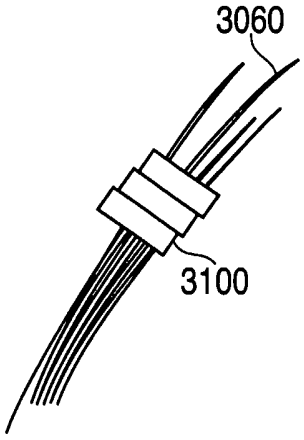


FIG. 3010

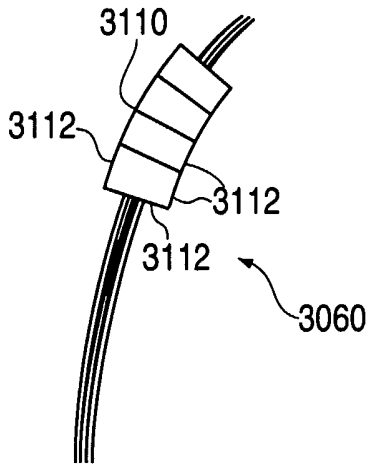


FIG. 3011A

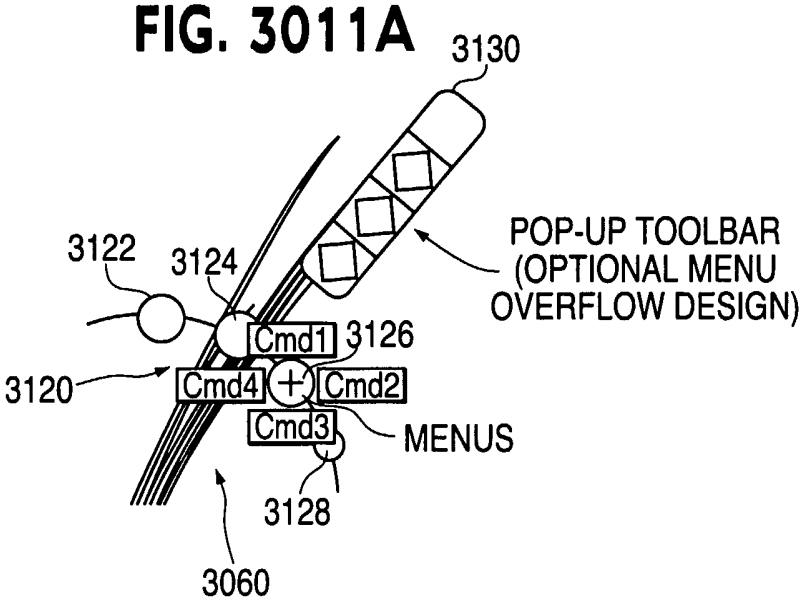

3130
POP-UP TOOLBAR
(OPTIONAL MENU
OVERFLOW DESIGN)
3122
3124
Cmd1
3126
3120
Cmd4
Cmd2
Cmd3
MENUS
3128
3060

FIG. 3011B

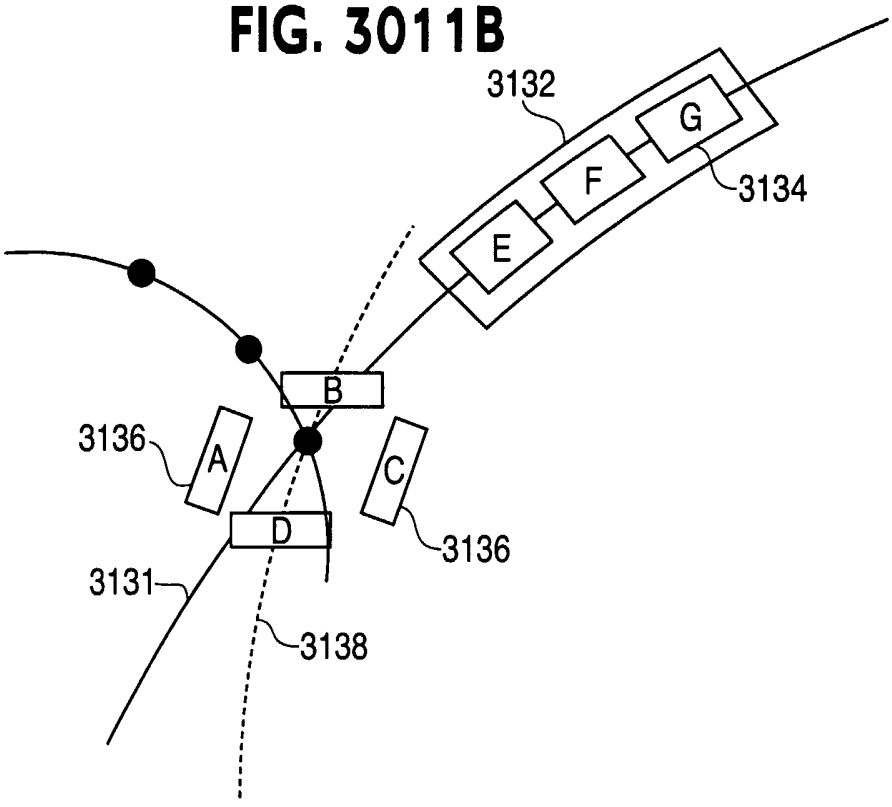


FIG. 3012

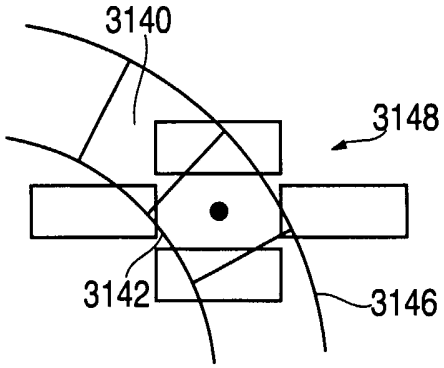


FIG. 3013

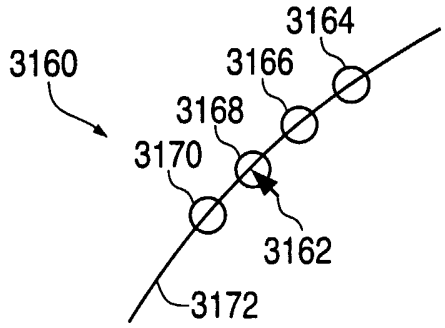


FIG. 3014

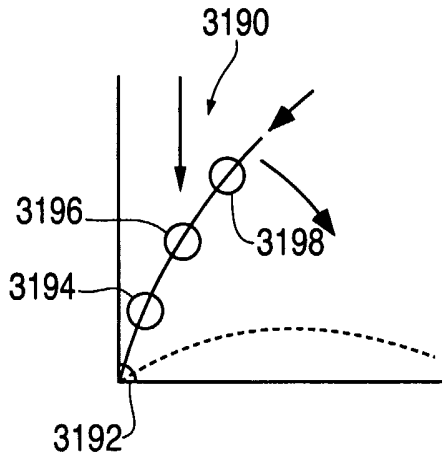


FIG. 3015

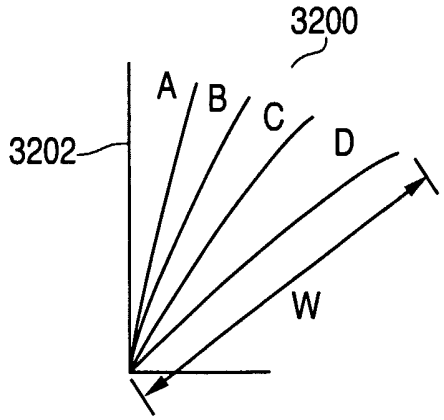


FIG. 3016

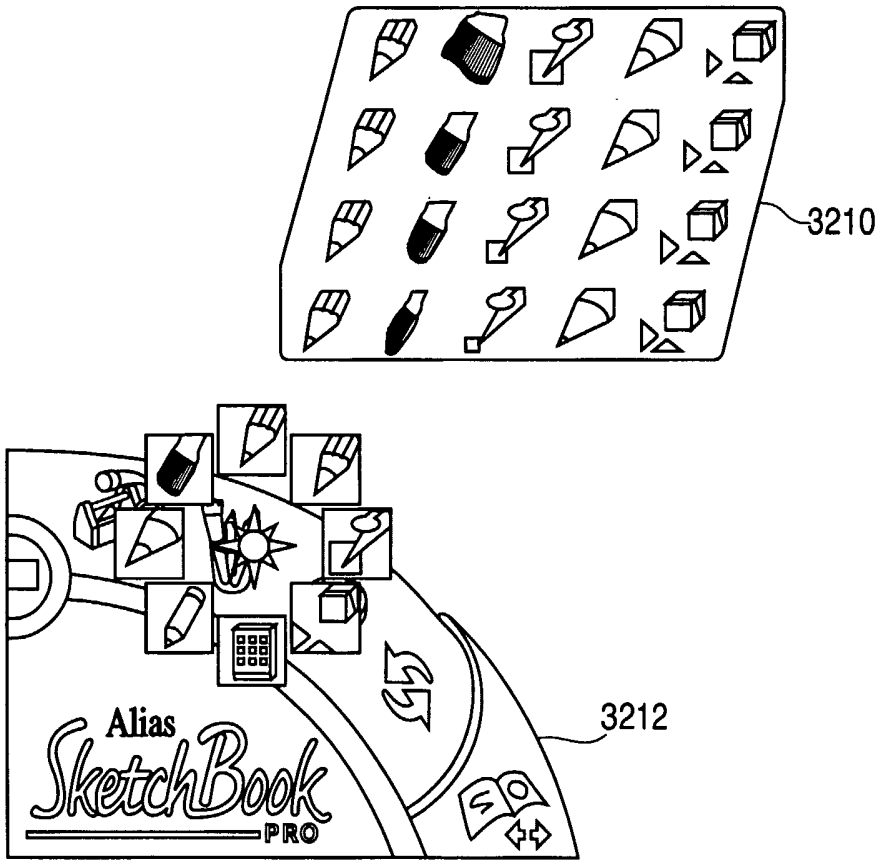


FIG. 3017

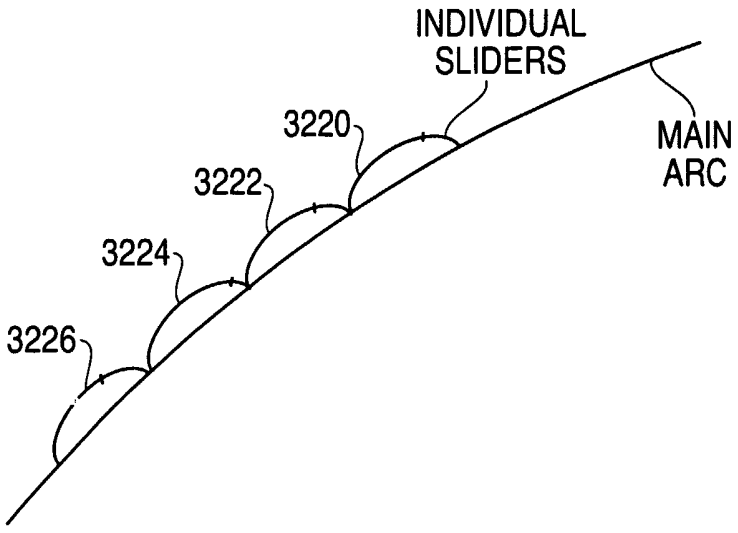


FIG. 3018

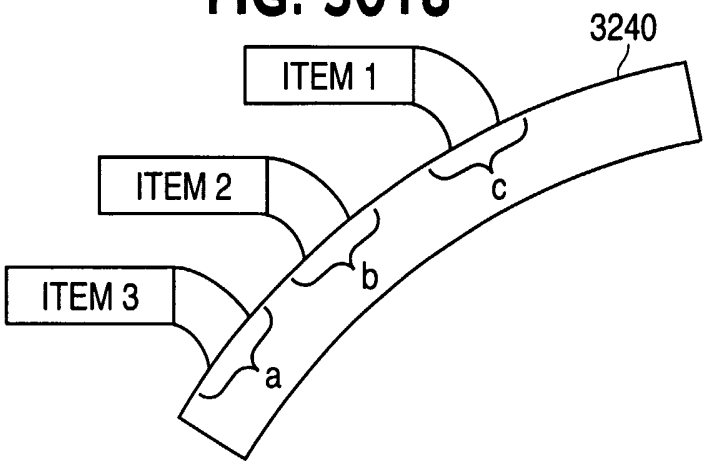


FIG. 3019

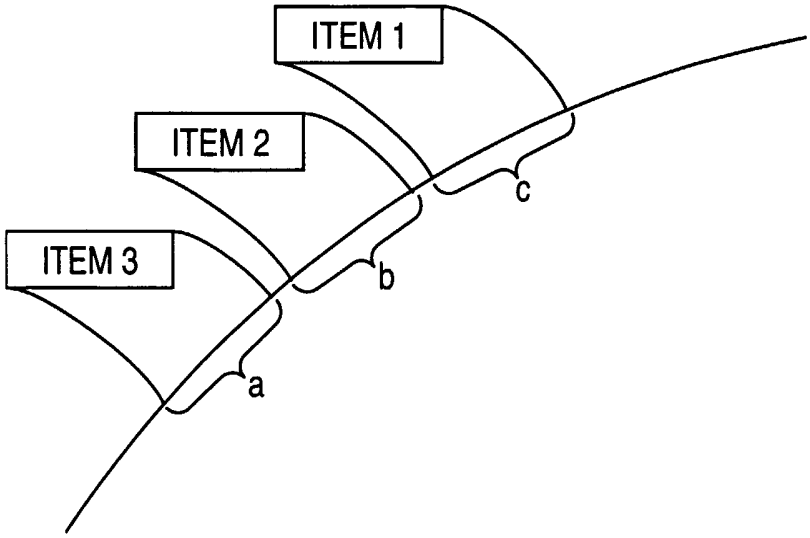


FIG. 3020

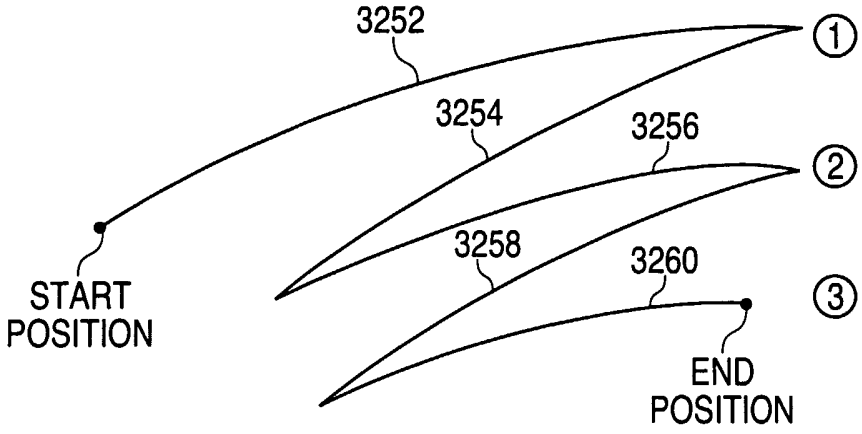


FIG. 3021

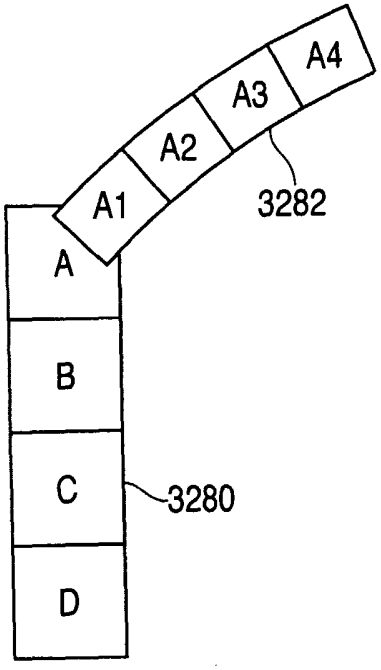


FIG. 3022

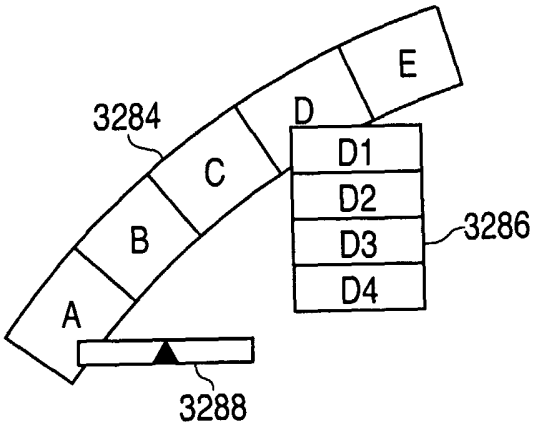


FIG. 3023

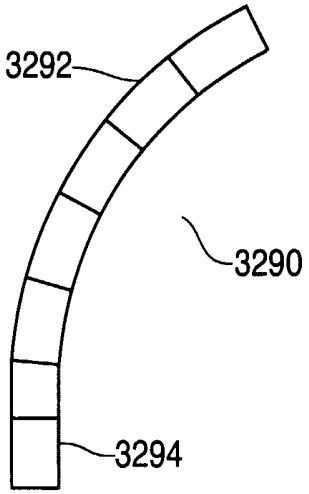
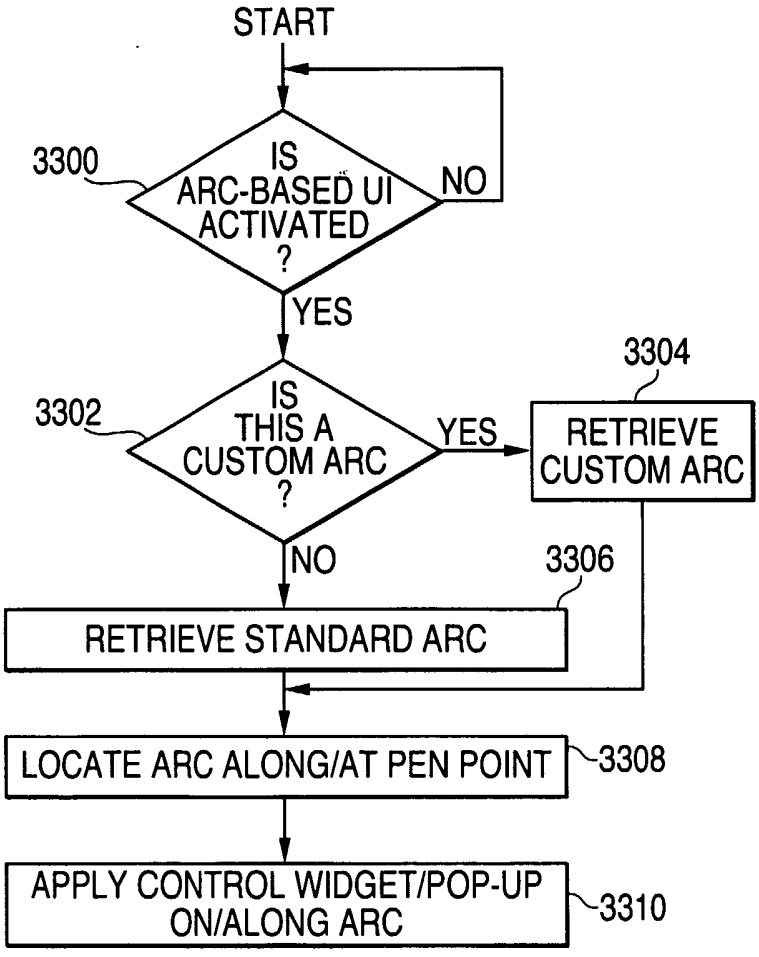


FIG. 3024


START
3300
IS ARC-BASED UI ACTIVATED ?
NO
YES
3302
IS THIS A CUSTOM ARC ?
YES
3304
RETRIEVE CUSTOM ARC
NO
3306
RETRIEVE STANDARD ARC
LOCATE ARC ALONG/AT PEN POINT
3308
APPLY CONTROL WIDGET/POP-UP ON/ALONG ARC
3310

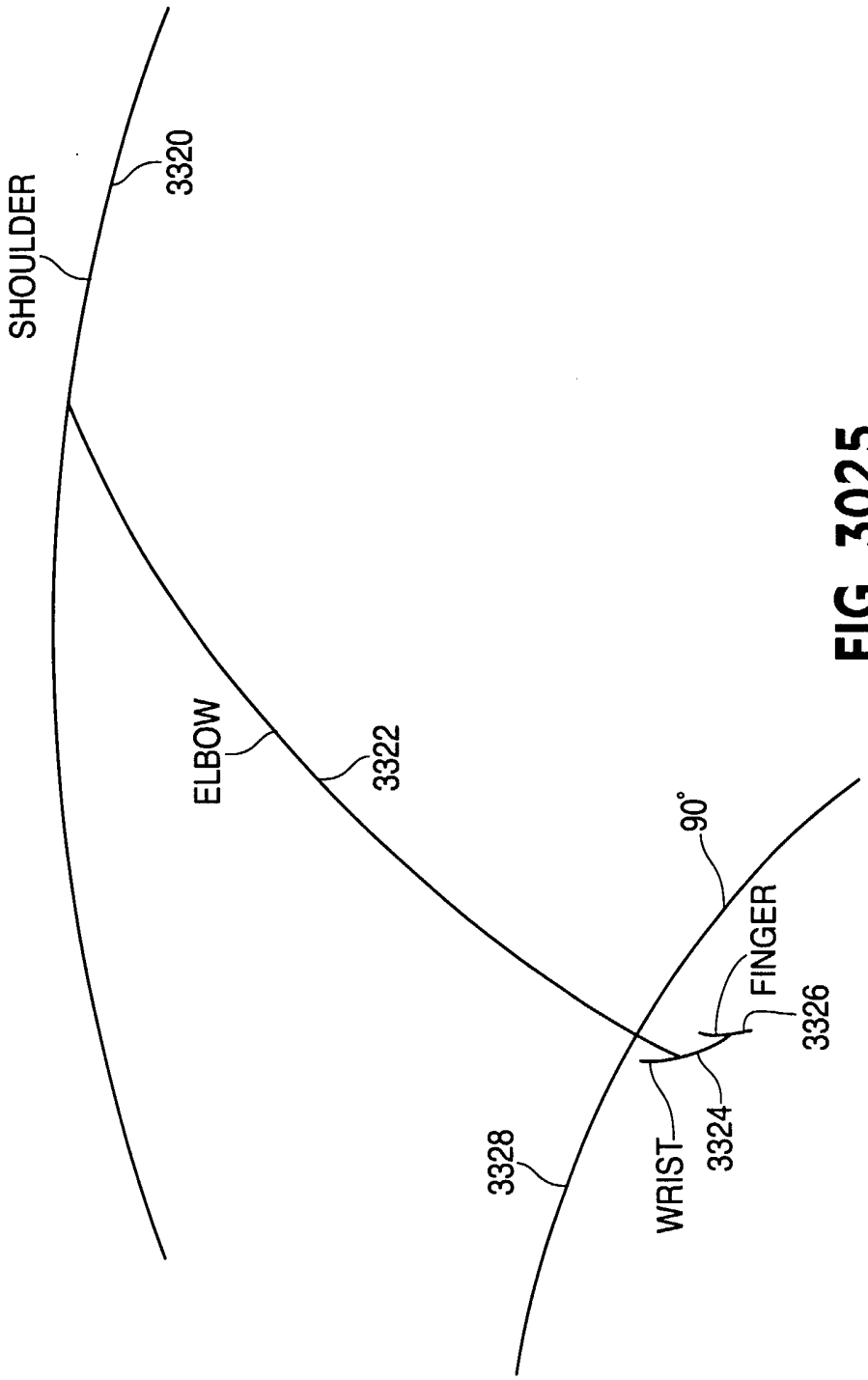
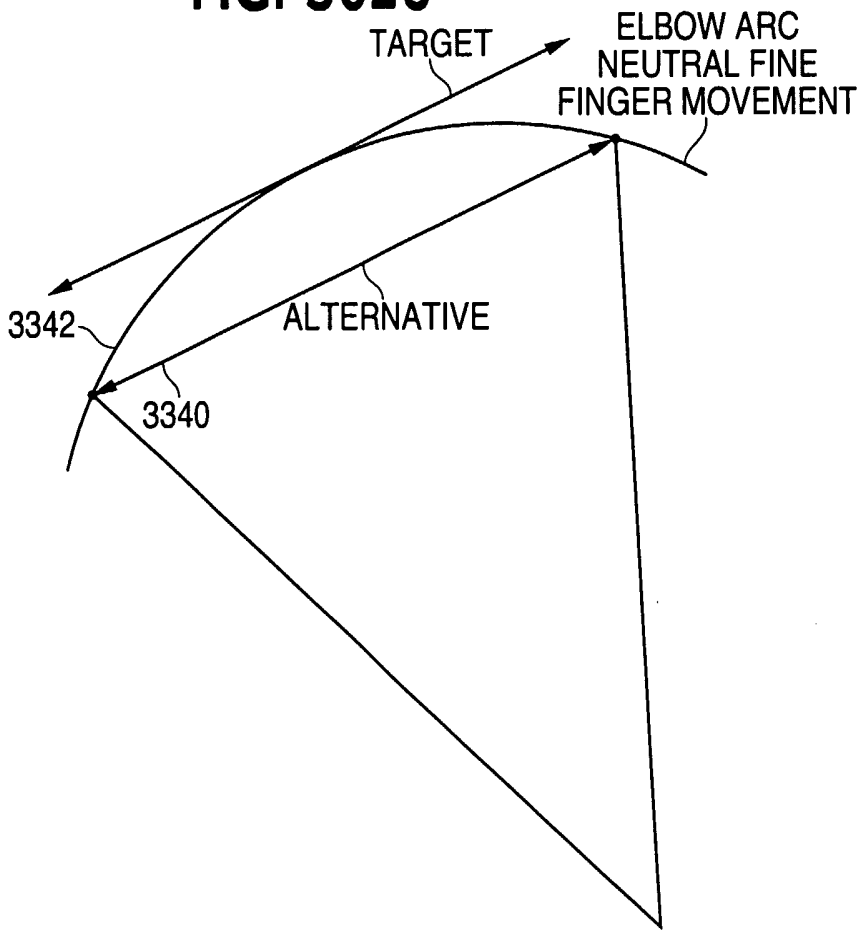


FIG. 3025

FIG. 3026


TARGET
ELBOW ARC
NEUTRAL FINE
FINGER MOVEMENT
3342
ALTERNATIVE
3340