

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
22 March 2007 (22.03.2007)

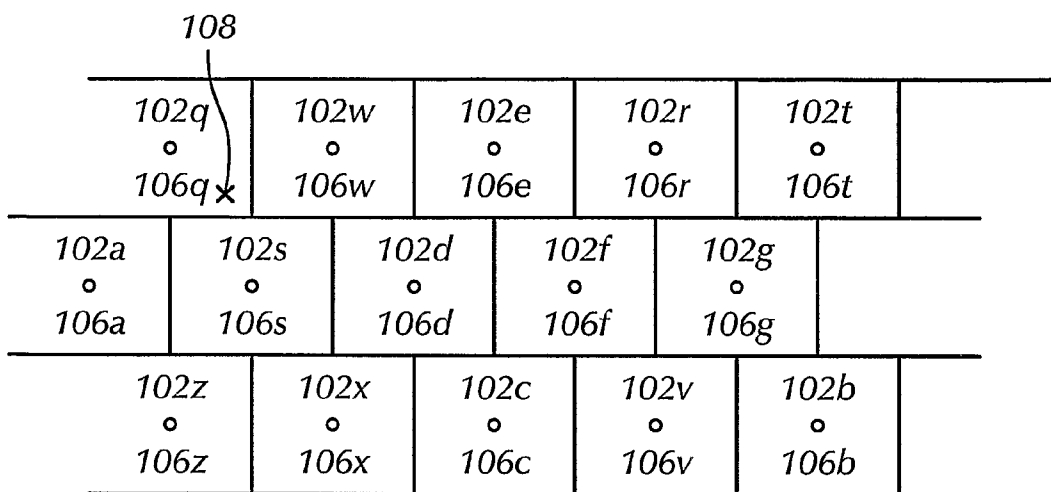
PCT

(10) International Publication Number
WO 2007/032843 A2

- (51) International Patent Classification: *G06F 3/048* (2006.01)
 - (21) International Application Number: PCT/US2006/031523
 - (22) International Filing Date: 11 August 2006 (11.08.2006)
 - (25) Filing Language: English
 - (26) Publication Language: English
 - (30) Priority Data: 11/228,737 16 September 2005 (16.09.2005) US
 - (71) Applicant (for all designated States except US): **APPLE COMPUTER, INC.** [US/US]; 1 Infinite Loop, Cupertino, CA 95014 (US).
 - (72) Inventors; and
 - (75) Inventors/Applicants (for US only): **ANDRE, Bartley, K.** [US/US]; 655 Fourteenth Avenue, Menlo Park, CA 94025 (US). **ORDING, Bas** [NL/US]; 1119 Dolores Street, #4, San Francisco, CA 94110 (US). **CHRISTIE, Greg** [US/US]; 1112 Kelly Drive, San Jose, CA 95129 (US).
 - (74) Agent: **ALLEN, Billy, C., III; WONG, CABELLO, LUTSCH, RUTHERFORD & BRUCCULERI LLP**, 20333 SH 249, Suite 600, Houston, TX 77070 (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, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
 - (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, 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, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, 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

[Continued on next page]

(54) Title: ACTIVATING VIRTUAL KEYS OF A TOUCH-SCREEN VIRTUAL KEYBOARD



(57) Abstract: A method of operating a touch screen to activate one of a plurality of virtual keys is provided. A touch location is determined based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of the plurality of virtual keys. Each of the plurality of virtual keys has a set of at least one key location corresponding to it. For each of the virtual keys, a parameter (such as physical distance) is determined for that virtual key that relates the touch location and the set of at least one key location corresponding to that virtual key. The determined parameters are processed to determine one of the virtual keys. For example, the determined one virtual key may be the virtual key with a key location (or more than one key location, on average) being closest to the touch location. A signal is generated indicating activation of the determined one of the virtual keys.

WO 2007/032843 A2



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.

**ACTIVATING VIRTUAL KEYS OF A TOUCH-
SCREEN VIRTUAL KEYBOARD**
CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority from 11/228,737 filed September 16, 2005 which is a continuation-in-part of prior Application No. 10/903,964, from which priority under 35 U.S.C. § 120 is claimed, which is hereby incorporated by reference in its entirety. This application is also related to the following co-pending applications: U.S. Serial No.: 10/840,862, Filed May 6, 2004; U.S. Serial No: 11/048,264, Filed July 30, 2004; U.S. Serial No.: 11/038,590, filed July 30, 2004; Atty. Docket No.: APL1P307X3 (U.S. Serial No. 11/228,758 filed September 16, 2005; and U.S. Serial No. 11/228,700, filed September 16, 2005; all of which are hereby incorporated herein by reference in their entirety for all purposes.

TECHNICAL FIELD

[0002] The present invention is in the field of touch screens and, in particular, relates to operating a touch screen to activate one of a plurality of virtual keys.

BACKGROUND

[0003] A touch screen is a type of display screen that has a touch-sensitive transparent panel covering the screen. When using a touch screen, a user makes a selection on the display screen by pointing directly to GUI objects on the screen (usually with a stylus or finger). The GUI objects may be considered as virtual keys (for example, of a keyboard). It is desirable to increase the usability of display screens for activating virtual keys.

SUMMARY

[0004] In accordance with one aspect, a method of operating a touch screen to activate one of a plurality of virtual graphical user interface (GUI) items is provided. A touch location is determined based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate

one of the plurality of virtual GUI items. Each of the plurality of virtual GUI items has a set of at least one item location corresponding to it. For each of the virtual GUI items, a parameter (such as physical distance) is determined for that virtual GUI item that relates the touch location and the set of at least one item location corresponding to that virtual GUI item. The determined parameters are processed to determine one of the virtual GUI items. For example, the determined one virtual GUI item may be the virtual GUI item with an item location (or more than one item location, on average) being closest to the touch location. A signal is generated indicating activation of the determined one of the virtual GUI items.

[0005] In accordance with another aspect, a method of operating a touch screen to activate one of a plurality of virtual keys is provided. Thus, for example, the virtual keys may be considered specific examples of virtual GUI items. A touch location is determined based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of the plurality of virtual keys. Each of the plurality of virtual keys has a set of at least one key location corresponding to it. For each of the virtual keys, a parameter (such as physical distance) is determined for that virtual key that relates the touch location and the set of at least one key location corresponding to that virtual key. The determined parameters are processed to determine one of the virtual keys. For example, the determined one virtual key may be the virtual key with a key location (or more than one key location, on average) being closest to the touch location. A signal is generated indicating activation of the determined one of the virtual keys.

[0006] Weights may be associated with each key location, and the weights may be adjusted dynamically based on a sequence of virtual keys previously activated and, in some examples, also based on external considerations such as statistical probabilities of particular virtual keys being activated (based, for example, on dictionaries or letter occurrence frequency data).

[0007] In one example, a dictionary lookup is used, and all of the letters of a word are considered in doing the dictionary lookup. In other examples, an “x” (e.g., 4) letter rolling window is considered. Thus, for a word that is (or is, thus far) less than or equal to “x” letters, the entire word is used in the dictionary

lookup. Once the “x” letter window is full, only the “x” letters are considered. This can reduce the amount of processing relative to considering greater than “x” letters. In addition, a hybrid approach may be used. For example, a dictionary lookup may be used (perhaps with a relatively small dictionary) initially and, if the word is not in the dictionary, then the letters of the rolling window is considered. In some examples, the positioning of the letters in the words (particularly, but not limited to, at the beginning of a word) is considered. In one example, this is accomplished by considering “space” as the first letter of a word. Furthermore, in some examples, the frequency of occurrence of a word (e.g., within a particular document) may be used as input for adjusting the weights.

[0008] Thus, the usability of touch screen virtual keyboards is enhanced.

BRIEF DESCRIPTION OF FIGURES

Figs. 1-1A through 1-1C illustrate a typical mismatch between the touch area of a user’s finger, to activate a GUI item on a touch screen, and a visual target associated with that GUI item.

Fig. 1 illustrates a keyboard GUI (having multiple keys) displayed on a touch screen, where each dot (not displayed on the touch screen) indicates a key location corresponding to a separate virtual key (which typically are not, but may be, coincident with the each displayed key of the keyboard GUI).

Fig. 2 is a flowchart illustrating one example method to determine which virtual key a user has activated.

Fig. 3 illustrates the shapes (typically not displayed on the touch screen) of the virtual keys corresponding to the displayed keys of the Fig. 1 keyboard GUI, as well as illustrating the dots that indicate the key locations.

Fig. 4 is a schematic diagram useful to explain the Fig. 2 flowchart.

Fig. 5 illustrates an example layout of virtual keys on a touch screen where each of at least one of the virtual keys has a set of greater than one key location corresponding to that virtual key.

Fig. 6 is a flowchart illustrating a portion of the Fig. 2 flowchart for a virtual key such as is shown in Fig. 5, having more than one key location corresponding to it.

Fig. 7 illustrates an example layout of key locations corresponding to virtual keys on a touch screen based on an initial “touch” by a user of fingers simultaneously in a “home key” position on the touch screen.

Fig. 8 illustrates an example arc-shaped keyboard, particularly suited for activation of the virtual keys using a thumb.

Fig. 9 illustrates an example of virtual GUI items on a touch screen display where, in this example, the virtual GUI items are not virtual keys of a keyboard.

DETAILED DESCRIPTION

[0009] We now describe a method of operating a touch screen to activate one of a plurality of virtual keys. The touch screen is an input device of a computing system which, for example, operates under program control (where the program is stored as instructions in a memory, for example) based at least in part on user input. The computing system may be, for example, a personal computer, a computing system embedded within an electronic device (such as, for example, a cellular telephone, media player or other consumer electronic device.) In operation, a user interacts with the touch screen and, in response, the touch screen (and or hardware/software associated with the touch screen) provides location data pertaining to touch input on the touch screen.

[0010] The touch screen provides a graphical user interface that includes GUI items that can be activated by the user as input to the program operating on the computer system. As can be seen from Fig. 1-1A, 1-1B and 1-1C, the touch area of a user's finger, to activate a GUI item on a touch screen, typically does not match a visual target associated with that GUI item. Fig. 1-1A shows a finger 12 approaching the touch-sensitive layer 14 of the touch screen. Fig. 1-1A also shows a visual target 10 displayed on the screen 16 of the touch screen. Fig. 1-1B shows the finger 12 touching the touch sensitive layer 14. It can be seen from Fig. 1-1B that the finger 12 contacts the touch sensitive layer 14 at a touch area 18. The mid-point 20 of the touch area 18 does not match the visual target 10 displayed on the screen 16 of the touch screen. Fig. 1-1C shows a plan view of the finger 12, visual target 10 and mid-point 20 of the touch area 18. Fig. 9 illustrates, in plan view, a situation where a finger 902 is being used to

potentially activate two somewhat small GUI items, a keep GUI item 904 and a delete GUI item 906.

[0011] It is desirable to process touches on a touch screen in a way that does not necessarily depend on a match between the visual target 10 and a touch area of a touch to activate a GUI to which the visual target 10 corresponds. It is desirable to process touches on a touch screen in a way that “makes sense,” which may include considering factors beyond (or instead of) a correspondence of the visual target and the touch area of a touch to activate the GUI to which the visual target corresponds.

[0012] In one example, the GUI items are virtual keys of a virtual keyboard. The GUI items are not limited to being virtual keys of a virtual keyboard, however. Fig. 1 illustrates a portion of a touch screen, where the gridlines 104 are displayed to the user, visually defining “keys” that are present on a conventional qwerty keyboard. The shapes and locations of the visually-defined keys need not be operationally correlated to the shapes of the virtual keys. This will be seen later, particularly with reference to Fig. 7. The visually-defined keys are labeled in Fig. 1 as 102q, 102w, 102e, etc., to indicate the letter to which each key corresponds on a conventional qwerty keyboard.

[0013] Each of the plurality of virtual keys has a set of at least one key location corresponding to it. The dots 106q, 106w, 106e, etc indicate key locations. In the Fig. 1 example, each set of key locations (corresponding to a separate virtual key) is a set of one key location. Later, sets of key locations are illustrated that include more than one key location.

[0014] In Fig. 1, a touch location is denoted by the “X” 108. In accordance with a broad aspect, a parameter is determined for each virtual key that relates the touch location to the set of at least one key location corresponding to that virtual key. The determined parameters are processed to determine one of the virtual keys, and a signal is generated indicating activation of the determined one of the virtual keys.

[0015] We now reference Fig. 2, which is a flowchart illustrating a particular example of a virtual key activation method. For simplicity of illustration, we initially describe a situation like in Fig. 1 where each set of key locations is a set of one key location. At step 202, the touch location is

determined, based on location data pertaining to touch input on the touch screen. The location data may be provided, for example, by low level program code being executed by a processor associated with the touch screen. Referring to the Fig. 1 example, the touch location is denoted by the "X" 108.

[0016] At step 204, for each virtual key, a distance is determined between the touch location "X" 108 and the key location 106 (generically) corresponding to that virtual key. Step 206 is a step included in some examples, and step 206 is discussed later. At step 208, it is determined to which of the virtual keys the shortest distance determined in step 204 corresponds. At step 210, a signal is generated to indicate activation of the virtual key determined in step 208.

[0017] Having described a particular example of a virtual key activation method, we now refer to the illustration in Fig. 3. The hexagonal shapes 302q, 302w, 302e, etc. illustrate an example of the shapes of the virtual keys, superimposed on the visually-defined keys 102. The example touch location "X" 108 shown in Fig. 1 is also shown in Fig. 3. Typically, the hexagonal (or other) shapes of the virtual keys would not be displayed to the user. Generically, the particular configuration of each virtual key 302 is based on the number and configuration of the key locations in the set of key locations corresponding to the virtual key. Other example configurations will be described later.

[0018] Fig. 3-1 illustrates an example in which, in contrast to Fig. 3, there is not such a regular correspondence between the visual targets and the key locations.

[0019] Fig. 4 illustrates an example of the distance determination step 204 of Fig. 2. Specifically, the length of each line segment 402q, 402w, 402e, etc. represents the distance between the touch location "X" 108 and each respective key locations 106q, 106w, 106e, etc. It can be seen that the line segment 402 having the shortest such length is 402q, which would be the result of step 208 as applied to the example of Fig. 1 and Fig. 3.

[0020] With Fig. 3 in particular as background, we discuss step 206 of the Fig. 2 flowchart. At step 206 (shown as a dashed rectangle), the distances determined at step 204 may be weighted by a weighting factor. For example, each distance may be multiplied by a weighting factor. The weighting factor for each virtual key affects the size of that virtual key. The smaller the weighting

factor associated with a first virtual key relative to the weighting factor associated with a second virtual key, the smaller the first virtual key is relative to the second virtual key (that is, the closer a touch location must be to a key location, relatively, to cause activation of the virtual key to which the key location corresponds.) In some examples, the weighting factors are normalized such that a weighting factor of one has no effect on the size of a virtual key, while a weighting factor greater than one has the effect of enlarging a virtual key, while a weighting factor less than one has the effect of diminishing the virtual key. The application of the weighting factors need not be linear. For example, the square root of the distance may be multiplied by the weighting factor.

[0021] We now turn to Fig. 5, which illustrates an example layout of virtual keys on a touch screen where each of at least one of the virtual keys has a set of greater than one key location corresponding to that virtual key. For example, referring to Fig. 5, a plurality of virtual keys (for example, designated by reference numerals 502a through 502h) are illustrated. The boundary of each virtual key (502, generally) is a function of the position (and, if applicable, weighting) of the plural key locations corresponding to that virtual key. It is noted that the keyboard graphic actually viewable by the user may not directly coincide with the irregularly-shaped boundaries of the virtual keys 502.

[0022] Fig. 6 illustrates steps 602 and 604, which may be included in step 204 of Fig. 2. In particular, step 602 determines, for each key location in the set of at least one key location corresponding to a virtual key, the distance between the touch location and the key location. With regard to the specific example of the virtual "m" key in Fig. 5, step 602 includes determining the distances d_a and d_b . The distance provided to step 206 or step 208 (Fig. 2) for the virtual "m" key is then determined, at step 604, as the average of d_a and d_b . In some examples, a step 606 (which, like step 206 in Fig. 2, is shown as a dashed rectangle) is included in which the distances determined in step 602 (e.g., d_a , d_b and d_c) may be weighted before an average is determined at step 604. Some reasons that it may be desired to weight particular key locations corresponding to a virtual key are discussed later.

[0023] Referring still to Fig. 6, steps 602 and 604 (and, sometimes, step 606) are also performed for virtual "j" and "n" keys (and the other virtual keys).

For the virtual “j” key, distances dx , dy and dz are determined. For the virtual “n” key, distances dj , dk and dk are determined. In some examples, some optimizations are realized such that the distances are not determined for every virtual key and/or for every key location associated with a particular virtual key.

[0024] Referring back to Fig. 2, optional step 206 and step 208 are carried out using the distances determined in the step 604 processing for the each virtual keys. As a result, the activated virtual key is the one virtual key for which there is the shortest physical distance between the key locations, associated with that virtual key, and the touch location.

[0025] We now discuss how the key locations are determined, in some examples. While in some examples, the key locations may be predetermined, in other examples, the key locations are dynamically determined. For example, a keyboard interface may be initially activated on a touch screen by processing that detects simultaneous touching of the touch screen by a plurality of fingers, where the particular number may vary by example. In one example, the keyboard interface is only activated if the relative positions of the touching fingers on the touch screen are consistent (within some threshold) with the positions of fingers on an actual keyboard. Once the initial positions of some of the keys (more properly, the initial key locations associated with virtual keys) are determined based on the respective positions of the touching fingers, initial key locations for remaining virtual keys of the keyboard, even though not directly corresponding to touch locations, may be determined. For example, the key locations for the remaining virtual keys of the keyboard may be set to have a predetermined relationship to the key locations for the virtual keys whose key locations are determined directly from the touch locations.

[0026] Fig. 7 illustrates a portion of a virtual keyboard. The lines 702 indicate the boundaries of the virtual keys, whereas the lines 704 indicate the key boundaries as displayed to the user on the touch screen. The “dots” (for example, 706a, 706b and 706c; 708a, 708b and 708c; and 710a and 710b) are not typically displayed but, rather, indicate key locations for the virtual keys.

[0027] In some examples, rather than strictly considering distance between a touch location and key locations, statistical parameters of the distribution of the key locations is taken into account. Thus, for example, a touch location that is

“closer” in standard deviation terms to the distribution of key locations for a first virtual key than to the distribution of key locations for a second virtual key is considered to correspond to an activation of the first virtual key. Thus, in one example, a first virtual key has corresponding to it a relatively closely-spaced distribution of key locations, relative to the distribution of key locations corresponding to a second virtual key. With these example distributions of key locations, in order for a touch location to correspond to activation of the a touch location would have to be relatively closer (in physical distance) to the center of the relatively closely-spaced distribution of key locations corresponding to the first virtual key than to the center of the distribution of relatively further-spaced key locations corresponding to the second virtual key.

[0028] We now discuss how, in one example, how particular virtual keys come to have multiple key locations associated with them. In particular, when it is determined which virtual key a touch location has activated, that touch location may be considered as an additional key location for the activated virtual key. In some examples, the key locations for the virtual keys are accumulated, although the number of key locations for each virtual key may be limited such that, for example, the key locations corresponding to the touch locations earliest in time may be discarded.

[0029] Furthermore, some key locations may be “backed out” based on conditions that occur later in time than when the key locations was first considered to be a key location. One such later-occurring condition is that a later touch condition is determined to correspond to a “backspace” virtual key that, when activated, would nullify the activation of the virtual key to which the to-be backed out key locations corresponds.

[0030] In addition, the weights associated with particular virtual keys (see step 202 in Fig. 2 and step 606 in Fig. 6) may be adjusted dynamically. In one example, the weights are adjusted based on the sequence of virtual keys already activated. For example, the weight associated with the most likely next virtual key to be activated, based on the sequence of virtual keys already activated, may be set to be higher (or lower) than the weights associated with other virtual keys. What virtual key is most like to be activated may be determined, for example, using a dictionary (word-based), statistical probability for individual letters (for

example, based on statistical frequency of occurrence), or a combination of both. In some examples, virtual keys deemed to be “dangerous” (e.g., whose the result of whose activation may be difficult to reverse) are given a smaller weight. Potential examples of such “dangerous” keys may include, for example, a “delete” key or “cancel” key (for example, the “delete” GUI item 906 in Fig. 9.) Furthermore, in some examples, the slower virtual keys are activated (corresponding, nominally, to more deliberate activation), the less any particular virtual key would be weighted.

[0031] The described examples may be particularly useful where the virtual keyboards are small relative to the touch area with which a user may activate a virtual key. With a conventional keyboard (with which, for example, it is determined that a particular virtual key is activated by determining if the touch area is within a boundary corresponding to that virtual key), the touch area may correspond to more than one virtual key.

[0032] For example, Fig. 8 illustrates an example of an arc-shaped keyboard that may be particularly suited for thumb activation of the relatively small and close-together virtual keys. The visual display 802 associated with the Fig. 8 keyboard, as well as (nominally) the layout of the key locations used to determine which virtual key is activated) is arc-shaped. An output area 804 is also shown, which provides an indication of the activated virtual keys.

[0033] The thumb, for example, is relatively large and difficult to control. The arc-shaped keyboard may be initiated by a particular gesture such as “swiping” the thumb across a corner of the touch screen. The arc-shaped keyboard may be, for example, located in a corner of the touch screen easily reachable for activation of the virtual keys by a thumb of a hand holding the touch screen computer.

[0034] While this invention has been described in terms of several preferred embodiments, there are alterations, permutations, and equivalents, which fall within the scope of this invention. It should also be noted that there are many alternative ways of implementing the methods and apparatuses of the present invention. It is therefore intended that the following appended claims be interpreted as including all such alterations, permutations, and equivalents as fall within the true spirit and scope of the present invention.

WHAT IS CLAIMED IS:

1. A method of operating a touch screen to activate one of a plurality of virtual GUI items, comprising:
 - determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of the plurality of virtual GUI items;
 - for each of the plurality of virtual GUI items, each virtual GUI items having a set of at least one GUI item location corresponding to it, determining a parameter for that virtual GUI item that relates the touch location and the set of at least one item location corresponding to that virtual GUI item;
 - processing the determined parameters to determine one of the virtual GUI items; and
 - generating a signal indicating activation of the determined one of the virtual GUI items.

2. The method of claim 1, wherein:
 - the plurality of virtual GUI items are virtual keys of a virtual keyboard, and the set of at least one items locations is a set of at least one key location;
 - the determined parameter for each virtual key indicates, when considered relative to the determined parameters for the other virtual keys, a likelihood that virtual key is the one the touch input is intended to activate; and
 - processing the determined parameters for the virtual keys includes determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood.

3. The method of claim 2, wherein:
 - determining the parameter for each virtual key includes
 - for each key location of the set of at least one key location corresponding to that virtual key, determining a parameter relating the touch location and that key location; and

processing the determined parameters for each of the set of at least one key location corresponding to that virtual key to determine the parameter for that virtual key.

4. The method of claim 3, wherein:
the determined parameter for each virtual key includes an indication of a relationship between a touch location and a distribution of key locations corresponding to that virtual key.
5. The method of claim 4, wherein:
the distribution of key locations corresponding to that virtual key is represented by at least one statistical parameter characterizing the distribution.
6. The method of claim 5, wherein:
the at least one statistical parameter includes standard deviation.
7. The method of claim 3, wherein:
the determined parameter for each virtual key is an average of the determined parameters for the set of at least one key location corresponding to that virtual key.
8. The method of claim 3, wherein:
for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.
9. The method of claim 4, wherein:
for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.
10. The method of claim 4, wherein:

the parameter for each virtual key is an average of the determined parameters for the set at least one key location corresponding to that virtual key, weighted to account for a relative size of that virtual key.

11. The method of claim 3, wherein:

for each of at least one of the plurality of virtual keys, at least one of the set of key locations corresponding to that virtual key is a touch location determined previously to be intended to activate that virtual key.

12. The method of claim 2, wherein:

determining a parameter for each virtual key includes accounting for a relative size of that virtual key.

13. The method of claim 2, wherein:

determining a parameter for each virtual key includes heuristic considerations for that virtual key.

14. The method of claim 12, wherein:

the heuristic considerations include considering the meaning of collective input to the touch screen.

15. The method of claim 13, wherein:

considering the collective input includes matching the collective input to a dictionary.

16. The method of claim 13, wherein:

considering the collective input includes considering a likelihood of occurrence of sequences of input.

17. The method of claim 15, wherein:

considering a likelihood of occurrence of sequences of input includes considering a likelihood of occurrence of sequences of a rolling window of input.

18. The method of claim 16, wherein:
considering the collective input includes a combination of considering a likelihood of occurrence of sequence of input and matching the collective input to a dictionary.
19. The method of claim 17, wherein:
considering a likelihood of occurrence of sequences of input includes considering a likelihood of occurrence of sequences of a rolling window of input
20. The method of claim 5, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
21. The method of claim 6, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
22. The method of claim 9 wherein:
for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.
23. The method of claim 18, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
24. The method of claim 1, further comprising:

initially determining at least some of the key locations.

25. The method of claim 20, wherein:
the step of initially determining at least some of the key locations includes, for each of the at least some of the key locations, setting that key location to be a touch location.
26. The method of claim 21, wherein:
the step of setting a key location to be a touch location occurs based on detection of a touch locations determined based on a plurality of simultaneous touch input.
27. A method of operating a touch screen to selectively activate a virtual GUI item, comprising:
determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input may be intended to activate the GUI item;
for each item location of a set of at least one item locations corresponding to the virtual GUI item, determining a distance between the touch location and that item location; and
processing the determined distances to determine whether the GUI items is activated.
28. The method of claim 27, wherein:
processing the determined distances to determine whether the GUI items is activated includes
determining whether a representative distance is within a particular threshold; and
based thereon, selectively activating the virtual GUI item.
29. The method of claim 28, wherein:
the representative distance is determined by averaging the determined distances.

30. The method of claim 28, wherein:
the representative distance is determined based on the determined distances and also based on a weighting factor, to account for a relative size of the virtual GUI item.
31. A computer-readable medium having a computer program tangibly embodied thereon, the computer program including steps for operating a touch screen to activate one of a plurality of virtual GUI items, the steps of the computer program comprising:
determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of the plurality of virtual GUI items;
for each of the plurality of virtual GUI items, each virtual GUI items having a set of at least one GUI item location corresponding to it, determining a parameter for that virtual GUI item that relates the touch location and the set of at least one item location corresponding to that virtual GUI item;
processing the determined parameters to determine one of the virtual GUI items; and
generating a signal indicating activation of the determined one of the virtual GUI items.
32. The computer-readable medium of claim 31, wherein:
the plurality of virtual GUI items are virtual keys of a virtual keyboard, and the set of at least one items locations is a set of at least one key location;
the determined parameter for each virtual key indicates, when considered relative to the determined parameters for the other virtual keys, a likelihood that virtual key is the one the touch input is intended to activate; and
processing the determined parameters for the virtual keys includes determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood.
33. The computer-readable medium of claim 32, wherein:

determining the parameter for each virtual key includes

for each key location of the set of at least one key location corresponding to that virtual key, determining a parameter relating the touch location and that key location; and

processing the determined parameters for each of the set of at least one key location corresponding to that virtual key to determine the parameter for that virtual key.

34. The computer-readable medium of claim 33, wherein:
the determined parameter for each virtual key is an average of the determined parameters for the set of at least one key location corresponding to that virtual key.
35. The computer-readable medium of claim 33, wherein:
for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.
36. The computer-readable medium of claim 34, wherein:
for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.
37. The computer-readable medium of claim 34, wherein:
the parameter for each virtual key is an average of the determined parameters for the set at least one key location corresponding to that virtual key, weighted to account for a relative size of that virtual key.
38. The computer-readable medium of claim 33, wherein:
for each of at least one of the plurality of virtual keys, at least one of the set of key locations corresponding to that virtual key is a touch location determined previously to be intended to activate that virtual key.

39. The computer-readable medium of claim 32, wherein:
determining a parameter for each virtual key includes accounting for a relative size of that virtual key.
40. The computer-readable medium of claim 32, wherein:
determining a parameter for each virtual key includes heuristic considerations for that virtual key.
41. The computer-readable medium of claim 40, wherein:
the heuristic considerations include considering the meaning of collective input to the touch screen.
42. The computer-readable medium of claim 41, wherein:
considering the collective input includes matching the collective input to a dictionary.
43. The computer-readable medium of claim 41, wherein:
considering the collective input includes considering a likelihood of occurrence of sequences of input.
44. The computer-readable medium of claim 35, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
45. The computer-readable medium of claim 36, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
46. The computer-readable medium of claim 37, wherein:

for each key location, the determined parameter relating the touch location and that key location is an indication of the physical distance between the touch location and that key location.

47. The computer-readable medium of claim 36, wherein:
determining from the determined parameters the virtual key for which the determined parameter indicates the highest likelihood includes determining for which virtual key the determined parameter indicates the smallest physical distance.
48. The computer-readable medium of claim 31, further comprising:
initially determining at least some of the key locations.
49. The computer-readable medium of claim 39, wherein:
the step of initially determining at least some of the key locations includes, for each of the at least some of the key locations, setting that key location to be a touch location.
50. The computer-readable medium of claim 35, wherein:
the step of setting a key location to be a touch location occurs based on detection of a touch locations determined based on a plurality of simultaneous touch input.
51. A computer-readable medium having a computer program tangibly embodied thereon, the computer program including steps for operating a touch screen to selectively activate a virtual GUI item, comprising:
determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input may be intended to activate the GUI item;
for each item location of a set of at least one item locations corresponding to the virtual GUI item, determining a distance between the touch location and that item location; and

processing the determined distances to determine whether the GUI items is activated.

52. The computer-readable medium of claim 51, wherein:
processing the determined distances to determine whether the GUI items is activated includes

determining whether a representative distance is within a particular threshold; and

based thereon, selectively activating the virtual GUI item.

53. The computer-readable medium of claim 52, wherein:
the representative distance is determined by averaging the determined distances.

54. The computer-readable medium of claim 52, wherein:
the representative distance is determined based on the determined distances and also based on a weighting factor, to account for a relative size of the virtual GUI item.

55. A method of operating a touch screen, comprising:
determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of a plurality of virtual keys of a virtual keyboard provided on the touch screen;
for each of the plurality of virtual keys, each virtual key having a set of at least one key location corresponding to it, determining a parameter for that virtual key that relates the touch location and the set of at least one key location corresponding to that virtual key; and
generating a signal indicating activation of a determined one of the virtual keys based on processing the determined parameters.

56. A computer readable medium having embodied thereon a program to operate a touch screen of a touch screen computer, the program comprising steps of:

determining a touch location based on location data pertaining to touch input on the touch screen, wherein the touch input is intended to activate one of a plurality of virtual keys of a virtual keyboard provided on the touch screen;

for each of the plurality of virtual keys, each virtual key having a set of at least one key location corresponding to it, determining a parameter for that virtual key that relates the touch location and the set of at least one key location corresponding to that virtual key; and

generating a signal indicating activation of a determined one of the virtual keys based on processing the determined parameters.

1/6

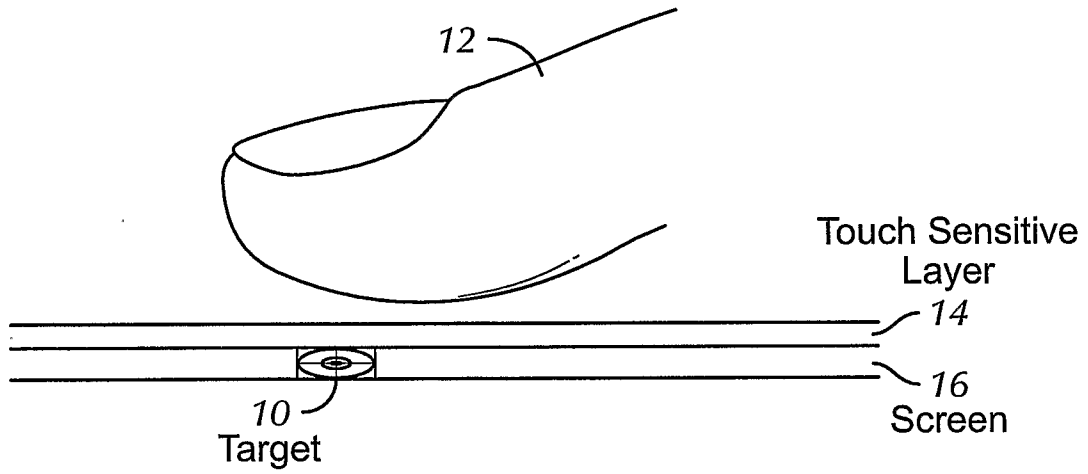


FIG. 1-1A

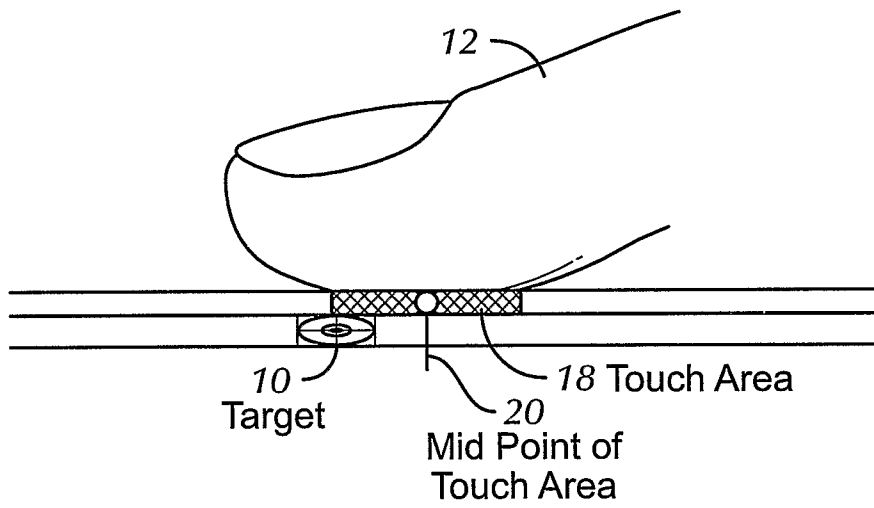


FIG. 1-1B

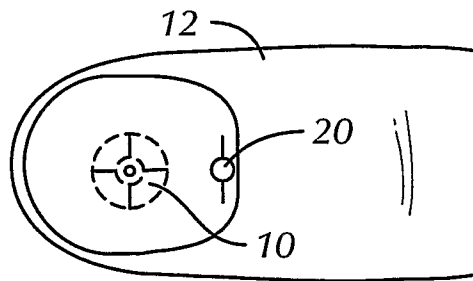


FIG. 1-1C

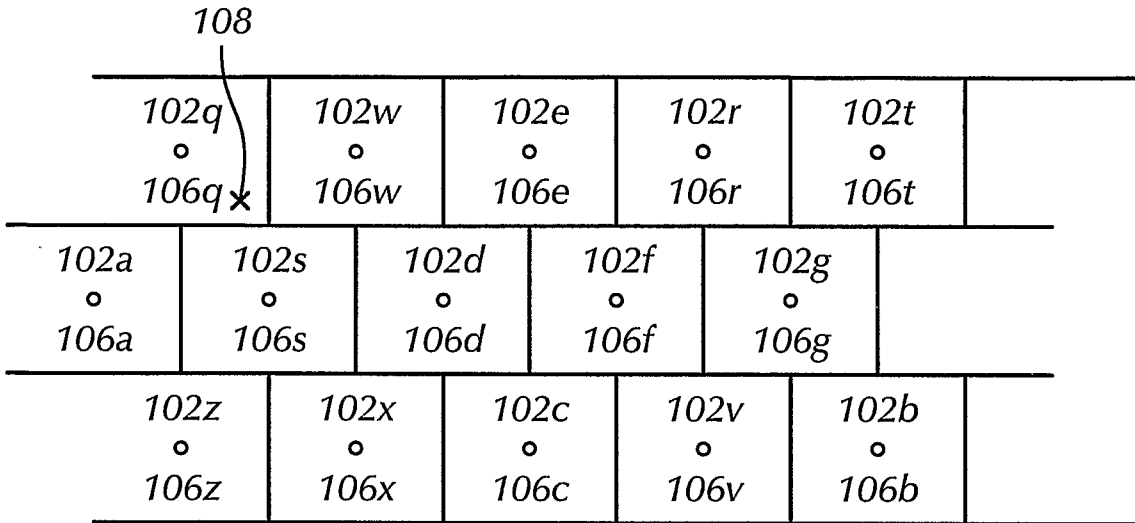


FIG. 1

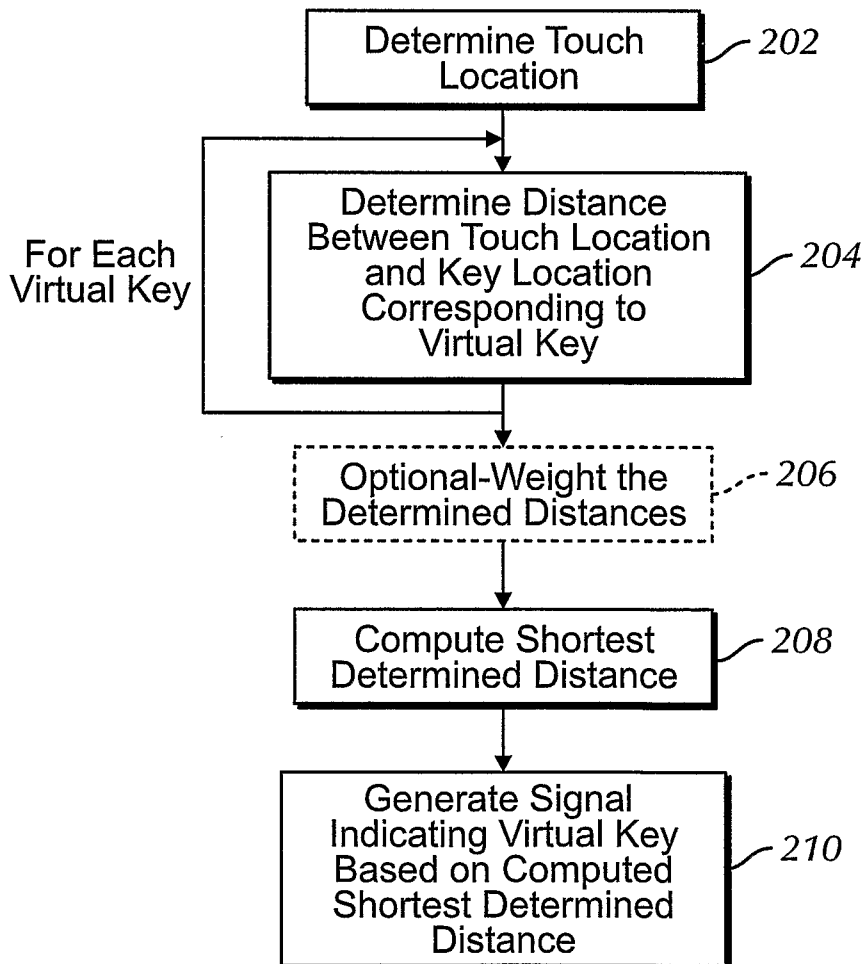


FIG. 2

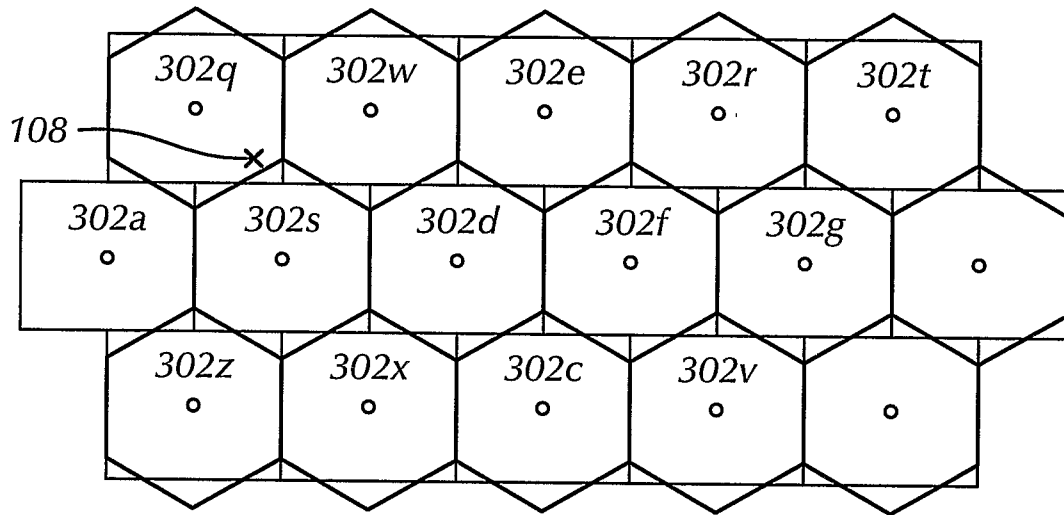


FIG. 3

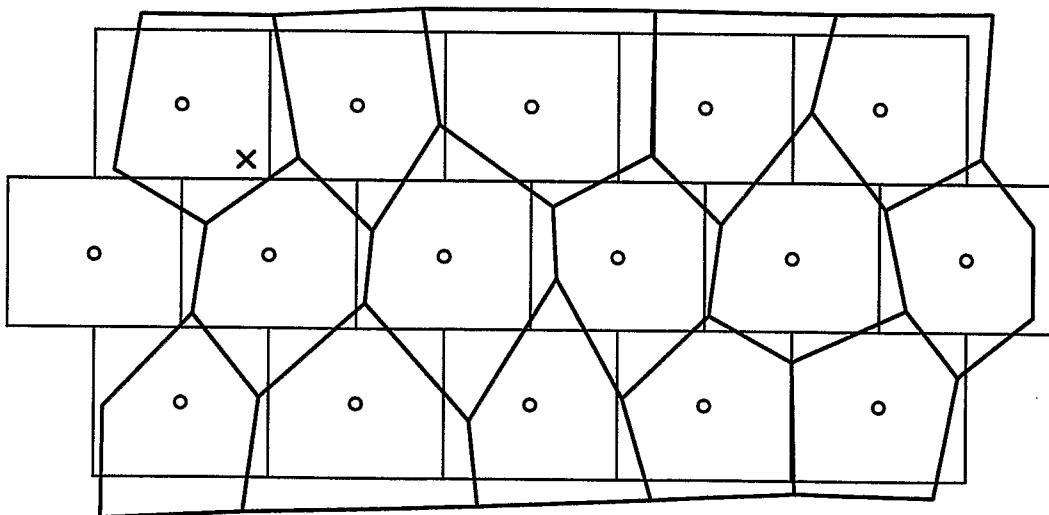
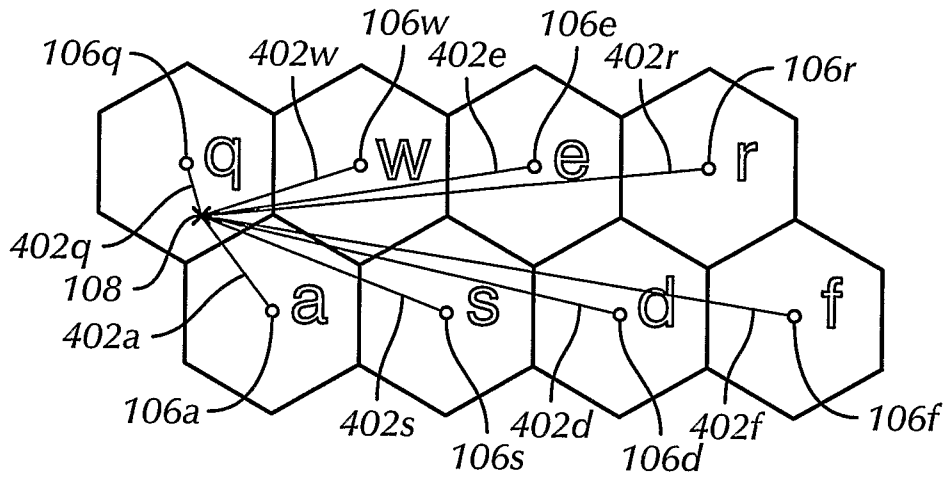


FIG. 3-1



d_q is Shortest

FIG. 4

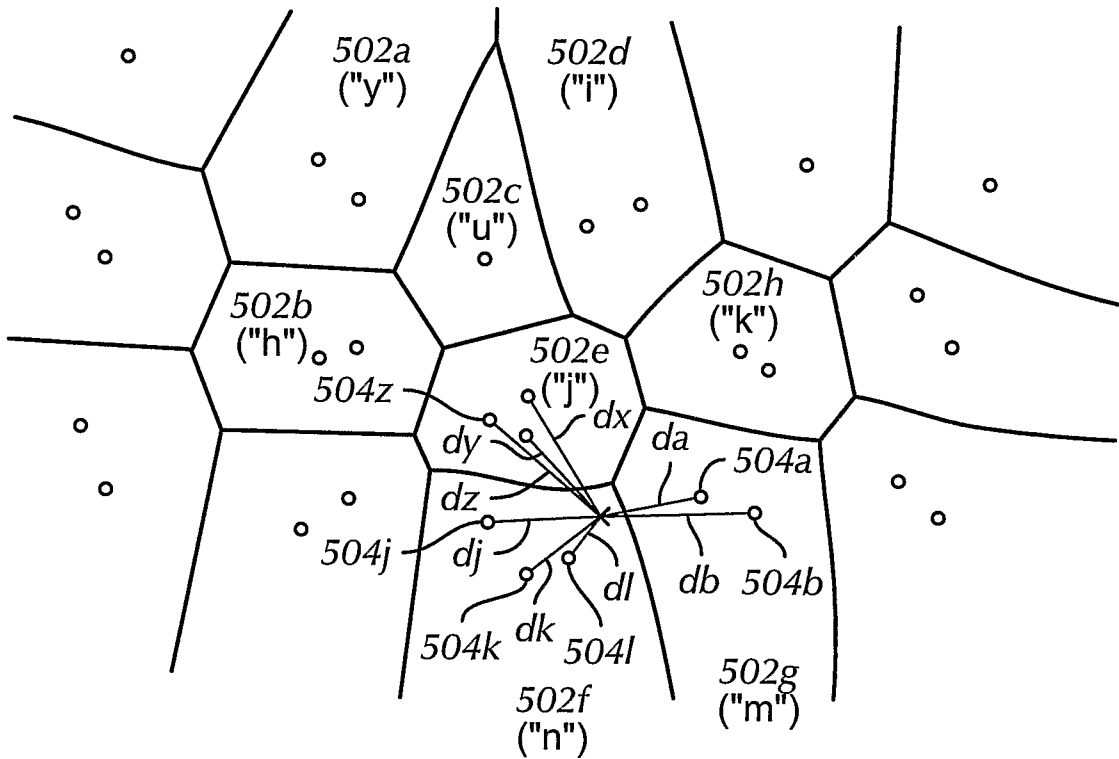


FIG. 5

5/6

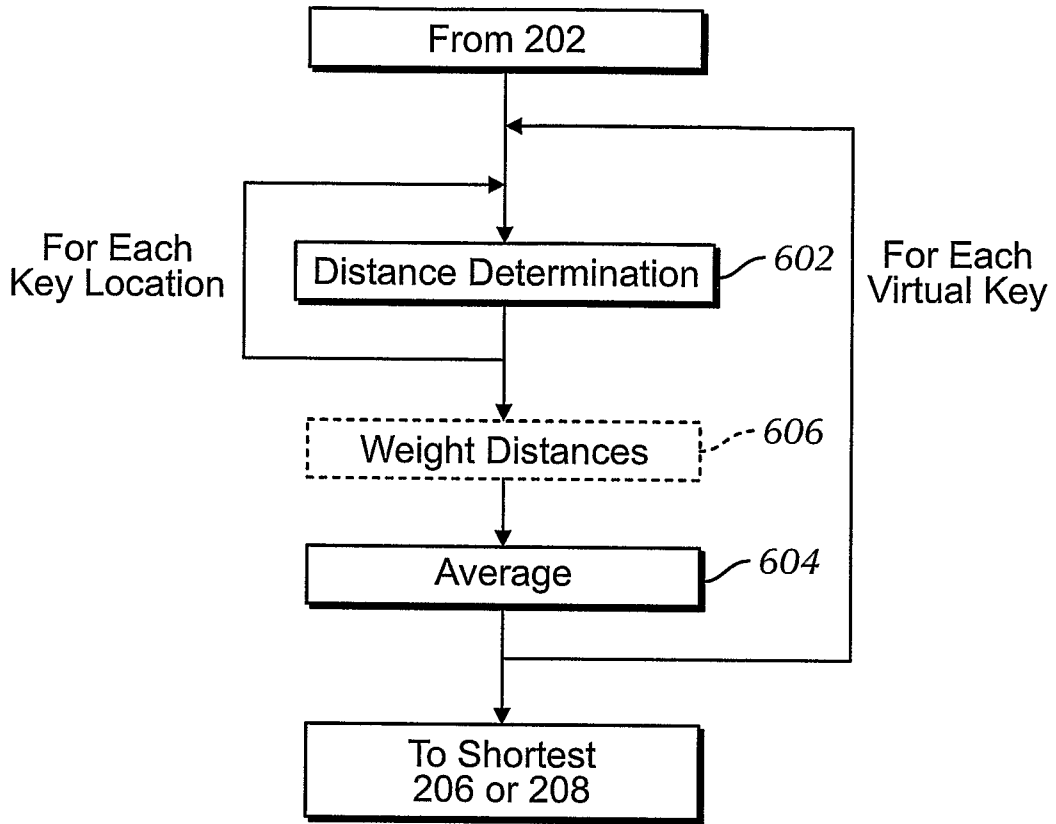


FIG. 6

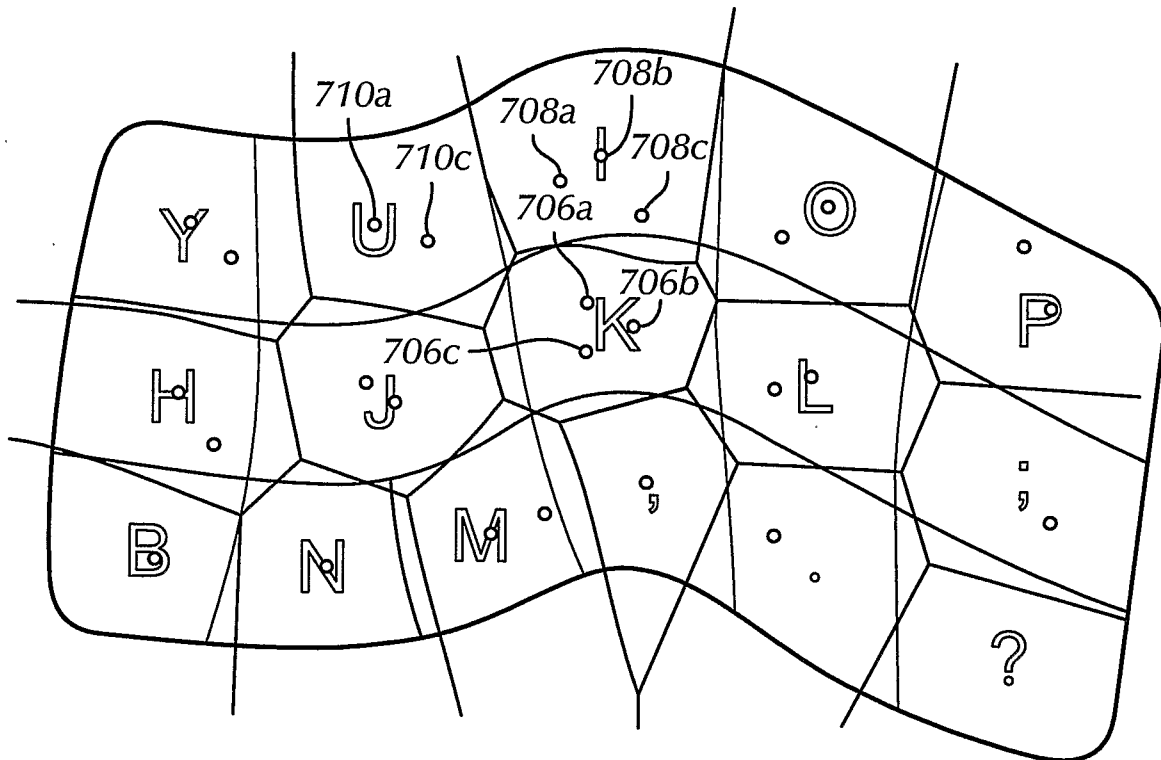


FIG. 7

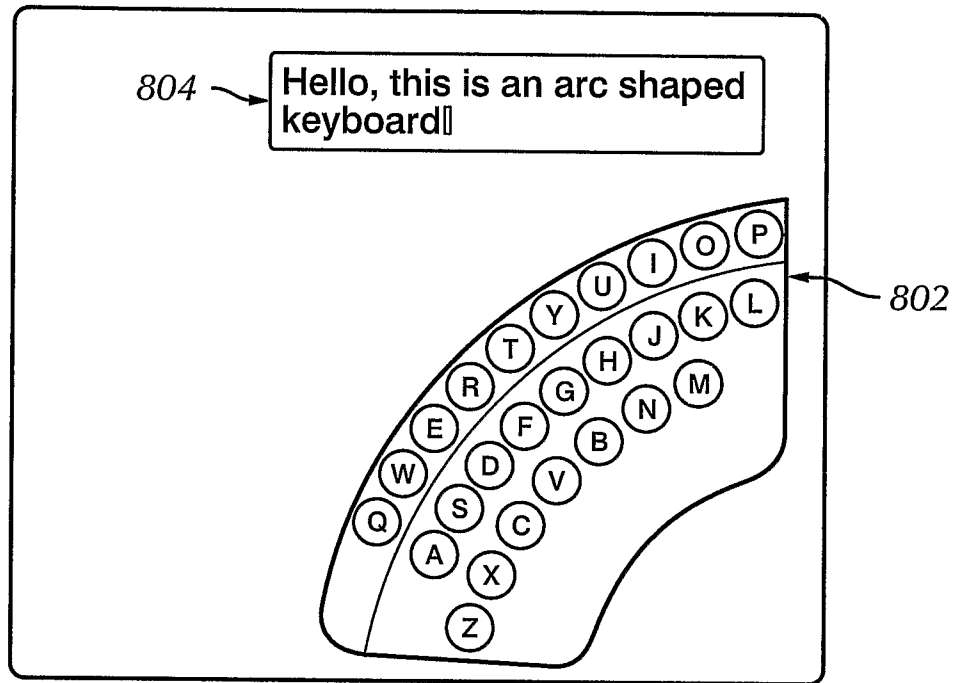


FIG. 8

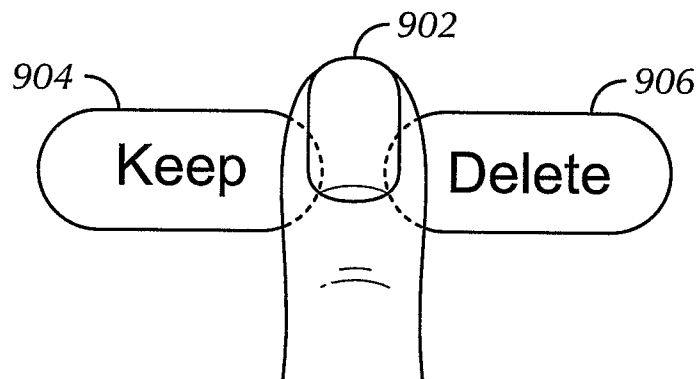


FIG. 9