[0075] "typical" -(W_{freq}=6)

[0076] "typically" -(W_{freq}=5)

[0077] "typing" -(W_{freq}=5)

[0078] For "g"

[0079] "tg" -(W_{freq}=0)

[0080] For "h"

[0081] "the" -(W_{freq}=10)

[0082] "they" -(W_{freq}=9)

[0083] "this" -(W_{freq}=9)

[0084] "that" -(W_{freq}=8)

[0085] "there" -(W_{freq}=8)

[0086] "these" -(W_{freq}=8)

[0087] The W_{freq} indicated is the relevant W_{freq} from the dictionary. The default value is 0, where a string does not appear there. Thus whilst "tt" and "tg" do not appear in the dictionary, they are still deemed possible and appear in this list with W_{freq} of 0. For "ty" and "th", there are many more examples than just the six illustrated. However, there is no point in obtaining those for scoring, since no more than six possibilities will appear in the final list. The top six scoring W_{freq} words for any possibility are chosen. Where two words have the same W_{freq}, they are chosen and listed in alphabetical order.

[0088] Using formula (1) [$W_{final}=a*W_{freq}+b*W_{distance}$], with the constants "a" and "b" given the values 1 and 15, respectively, the total scores given to the candidate words/strings indicated above are calculated in step S212 as:

[0089] "tt" -(W_{final}=4.9)

[0090] "type" -(W_{final}=16.8)

[0091] "types" -(W_{final}=16.8)

[0092] "typed" -(W_{final}=15.8)

[0093] "typical" -(W_{final}=14.8)

[0094] "typically" -(W_{final}=13.8)

[0095] "typing" -(W_{final}=13.8)

[0096] "tg" -(W_{final}=6.7)

[0097] "the" -(W_{final}=18.3)

[0098] "they" -(W_{final}=17.3)

[0099] "this" -(W_{final}=17.3)

[0100] "that" -(W_{final}=16.3)

[0101] "there" -(W_{final}=1 6.3)

[0102] "these" -(W_{final}=16.3)

[0103] The scores are compared in step S214 and the list generated in step S216, containing the top six candidate strings in score order, with alphabetical order being secondary, is:

[0104] "the", "they", "this", "type", "types", "that".

[0105] This list of words is then displayed in the list display area 26 in step S112. Step S114 determines if any symbol has yet been confirmed. In this case, the initial "t" not yet been confirmed, as there is no space or some such following it. The second letter is also not confirmed as nothing has been selected from the list yet, so the negative answer takes the process back to step S100.

[0106] In order to continue inputting the word "that", the user does not need to type in the letters "a" and "t", he just needs to touch the word "that" in the list display area 26. The relevant position signals are provided in step S100 and step S102 determines that the new selected position 52 is not within the virtual keyboard. So it is succeeded by step S104, which determines that the new selected position 52 falls within the list display area 26. In the following step S18, the word "that" appears in the message line 24. Step S118 is followed by step S116 for the re-calibration operation.

[0107] Where a selection is made from a word list generated by step S216, the existing current symbol string (in this case "th") is deleted and replaced in step S118 with the chosen word, in this example "that". The deletion of the existing string, or at least the latest symbol placed there in the previous working of step S108, is useful to make sure that the correct word is displayed, since the current displayed symbol string (resulting from previous step S108) may not be consistent with the selected word from the word like (for example if "type" had been chosen, rather than "that").

[0108] In this example, the word "that" is selected by the user. The re-calibration step S116 has two keys to re-calibrate, as only two letters "t" and "h" were selected (although the "a" and the second "t" are part of "that", they were not selected keys or symbols as such). For the "h", using the figures given above, the selected position is offset 1.2 mm left of the centre (which co-exists with the representative position in this example) and 1.35 mm above it. As this is the first time "h" has been reset, "ΣXoff-cent-old" and "ΣYoff-cent-old" are precept at 0, and "n" is precept at 100. Then using formulae (2) and (3) above:

$$Xnew=(-1.2+0)/100=-0.012$$

$$Ynew=(1.35+0)/100=0.014$$

[0109] Thus, the new representative position for "h" is 0.012 mm left of the centre of the "h" key and 0.014 mm above the centre of the "h"0 key. The representative position of the "t" key would be re-calculated in a similar manner based on the relevant selected position which led to its input.

[0110] On the other hand, had the user wanted to input a different word, such as "these", which was not one of the displayed list, he would go straight to inputting another letter, without touching the list, and the process would go from step S102 to step S106 instead of to S104 and proceed in a similar manner as that which led to the display of the letter "h", described above.

[0111] The above embodiment has each representative position calculated and stored separately. However, in another alternative, representative positions can all be moved together. This is based on the fact that if there is a parallax problem, it is likely to be the same for every key and therefore the offset in the selected position is likely to be the same or similar for every selected key. Thus all the offsets