

106 at the cursor **107** in the text input field **102**, as is seen in **FIG. 12**. Alternatively, the suffix **133b** may be appended to the partial word input **106** in the text input field to form the full word **112**; the end effect will be the same: automatic completion of a partial word input into a desired and selected full word **112**. Once this has taken place, the prefix **132** and the full word candidates **133a-d** will be removed from presentation on top of the space bar **130**, thereby again revealing all of the activation area **134** of the latter. When the text input field **102** is updated in this manner with a selected full word candidate, a trailing space character **114** may be appended automatically at the end of the full word **112**.

[**0088**] In one embodiment, as has already been explained, the user may himself control whether or not such a trailing space character is to be added by choosing between selecting the candidate with a tapping action (stylus-down followed by stylus-up within the activation area of the candidate) or with a dragging action (stylus-down within the activation area, followed by stylus-dragging outside of the activation area and then stylus-up).

[**0089**] Note that the user is not obliged to select any of the presented full word candidates; if he is not happy with any of them he may keep on tapping on the various keys of the virtual keyboard **110** to keep on inputting text character by character. Each time an additional character has been added to a partial word, the word completion functionality will use this extended partial word and derive new full word candidates for presentation on top of the space bar **130**. When the user has completed a word manually by tapping on different keys, he may select the space bar **130**, wherein a space character will be added after the last word input and any presented full word candidates will be removed from the space bar **130**.

[**0090**] In one embodiment the word completion functionality may be adapted to derive all possible full word candidates using said dictionary. In such a case, it is likely that the predetermined area on the space bar **130** will not be able to present the suffices for all such possible candidates at the same time. Optionally, the suffices of the most likely candidates, as judged from aforesaid statistics, or of the first candidates in some structured order such as alphabetic, may be shown initially in the predetermined area, and the suffices of the other candidates may be made accessible by tapping a "more" element in the user interface. Upon selection of this element, a pop-up menu with more candidates may be launched somewhere on the display screen.

[**0091**] In one embodiment, as already explained, single-character candidates are not presented, since there will be no real saving in the amount of key taps for such candidates.

[**0092**] In one embodiment, the minimum length L_{dyn} of the part of the space bar's activation area **134** that is left uncovered by full word candidates **133a-d** corresponds to at least the length of two characters, and a maximum of four candidate suffices are shown at the same time in the predetermined area. Four suffices are believed to be efficient for the user to quickly browse and select among. In cases where the predetermined area would not fit four candidates, e.g. because of long suffices, the number of presented candidate suffices may either be reduced to three (or less), or candidates may be switched so that shorter candidate suffices will be presented instead of longer ones. Still one alternative would be to truncate candidate suffices (e.g. present only "mplet . . ." instead of "mpletion").

[**0093**] The invention has mainly been described above with reference to a number of embodiments. However, as is readily appreciated by a person skilled in the art, other embodiments than the ones disclosed above are equally possible within the scope of the invention, as defined by the appended patent claims. One alternative aspect of the invention involves presenting a set of full word candidates in their entirety in the predetermined area on top of the space bar. Thus, even if such an alternative aspect will not make the benefits of using candidate suffices only, it will still have all the benefits associated with presenting word completion candidates on top of the space bar, as compared to the locations where such candidates have been presented in the prior art.

[**0094**] With reference to **FIGS. 13 and 14**, an alternative embodiment will now be described which employs handwriting recognition (HWR) functionality instead of a virtual keyboard as text input means. **FIG. 14** illustrates the typical main components of such an embodiment. Except for the following, the elements of this embodiment are the same as, or equivalent to, the elements of the embodiment previously described with reference to **FIG. 5**. Therefore, identical reference numerals have been used in **FIG. 14** and in **FIG. 5** to represent such same or equivalent elements. Correspondingly, identical reference numerals have been used in **FIG. 13** and in **FIG. 11** to represent same or equivalent elements in the display screen snapshots shown therein.

[**0095**] The user interface **51** of the alternative embodiment includes a handwriting recognition (HWR) module **59** with word completion functionality, instead of (or in addition to) the virtual keyboard module **56** of **FIG. 5**. As is known per se in the art, handwriting recognition typically involves detecting a hand-written input, made by a human user by writing with a stylus or other writing tool on the surface of a touch-sensitive display screen, and interpreting the hand-written input as a symbol among a set of possible symbols in a predefined symbol set.

[**0096**] In the situation shown in **FIG. 13**, the user has thus made a text input **104** which includes four complete words ("He", "was", "asked", "to") and one partial word input **106** ("re"). More specifically, the partial word input **106** has been hand-written, character by character, by the user in a handwriting input area **180**, which is provided as part of a handwriting palette **170** in the lower region of the touch-sensitive display screen, and has been interpreted by the handwriting recognition functionality in the HWR module **59** as representing the individual characters "r" and "e", respectively. Upon successful interpretation, the characters are shown as typographic letters **182**, instead of the original free-hand version as drawn by the user.

[**0097**] Now, the word completion functionality of the HWR module **59** is called upon, like in the corresponding situation previously described for **FIG. 11**, to derive a set of full word candidates for the partial word input. As seen in **FIG. 13**, the derived set of full word candidates includes the candidates "reply", "record", "receive" and "result".

[**0098**] As with **FIG. 11**, these candidates are not shown in their entirety on the display screen; instead, the prefix **132** that corresponds to the partial word input **106** is presented together with the suffices **133a-133d** of the full word candidates in the predetermined area on top of the space bar