[0036] **FIG. 2** shows an embodiment of a character chart for primary character forms according to the present invention. In addition, multiple alternate forms exist for nearly every character (not shown). These character forms are a product of hours of user testing and extensive iteration. In user testing, most subjects discovered and used several of the alternate character forms despite their absence from the chart, suggesting a high degree of guessability for the EdgeWrite character forms.

[0037] Though many of the characters look vaguely like their handwritten counterparts, the mnemonic power of these characters comes less from their appearance and more from their "feel." One person noted this when, after entering 20 phrases using the present invention, he said, "I don't remember any of the pictures in my mind, but I still feel them in my hand."

[0038] As in some other gestural alphabets, some letters resemble lowercase forms, while others resemble uppercase forms. All letters produce a lowercase form unless the capitalization suffix stroke is appended to the usual letter stroke. The suffix stroke may simply be a motion to, for example, the upper-left corner (think "up," to "make it big") after the regular letter form is made but before lifting the stylus. Note that, by design, no letters finish in the upper-left corner, allowing for this suffix stroke to be appended. In user studies, subjects had no trouble with this method of capitalization.

[0039] Another thing to notice about the character chart (**FIG. 2**) is that it is representational, not literal. We faced a design challenge in depicting the strokes on paper, as many characters have strokes that pass over the same edge more than once. If such a "double pass" is drawn literally, then the result is merely a single line. We chose to arc the paths into the intended corners. These arcs make it possible to depict a double pass over the same edge. In the present invention, all movements are, in the ideal case, straight lines. As mentioned, however, straight line motion is not necessary for recognition, only hitting the corners in the proper order.

[0040] If we define a "segment" to be a straight line stroke between two vertices (or corners), then for gestures made inside a closed shape with v vertices, the number of possible character forms using s segments is given by the formula:

$$\text{forms} = \sum_{i=0}^{s} v \cdot (v-1)^i$$

[0041] This formula treats a tap at a vertex as a legal stroke, and assumes that the same corner is never used twice in a row.

[0042] For the preferred embodiment using a square, v=4. If s=0, meaning we use no segments, we see from the formula that we have 4 possible forms available to us: a tap in each of the square's four corners. With 1 segment, there are 16 possible forms (4+4×3), with 2 segments we get 52 forms, and with 3 segments we get 160 forms. Thus, there is a wealth of forms to choose from with relatively few segments.

[0043] The character chart in **FIG. 2** represents 100 characters: 26 lowercase letters, 26 uppercase letters via the capitalization suffix stroke, 10 digits, 4 white space characters, 2 punctuation mode-setters, and 32 punctuations. We do not count period twice, as it is the same form in and out of punctuation mode. Not pictured in the chart are the four directional arrow keys, which are also implemented, making for 104 unique characters in one embodiment of the current set of characters.

[0044] The average primary character form as shown in **FIG. 2** has 2.47 segments in it, excluding capitalization. If we include capitalization and its associated suffix stroke, this average increases to 2.84. The average number of segments per character for the whole character set, including all alternates and capitals, is 3.49. Incidentally, the whole character set in the preferred embodiment contains 228 forms. Note that these values exclude the punctuation mode setting stroke required for some characters.

[0045] Because we have 102 characters excluding punctuation mode-set, the forms equation above dictates that we must use 3 segments for at least some of the characters—50 of them to be exact. If we designed the character set with the fewest number of possible segments and no modes, and with only one form for each character, then the average number of segments per character in that set would be 2.39. So even with high learnability and guessability, the average segments per primary character according to the present invention (2.84) is not much higher than this theoretical lower bound (2.39). For entry without capitals (e.g., instant messaging), the average is even closer (2.47). This is due, in part, to the use of a punctuation mode, allowing for the reuse of certain character forms. It is also due to the choice of minimal-length character forms, without sacrificing their mnemonic feel.

[0046] The corners began naively as points rather than areas, and this proved to be inadequate, as users rarely hit the exact pixel in the corners. This was because users held their styluses at various angles. An angled stylus **105** impacts the edge of the plastic template hole **104** a few millimeters above its tip, causing the tip to jut a few pixels into the square even when the stylus is flush against the edge (**FIG. 1**).

[0047] After we increased the corner size to an appreciable area, two other problems emerged. Once moving, users would accidentally hit corners, particularly when making a diagonal stroke, as in an "s." But if the corners were made too small, users would often fail to hit them on pen-down, particularly in the backspace stroke (across the top or bottom edge from right to left). It seemed we needed large corners for when the stylus went down, but then small corners thereafter.

[0048] The next step in our design process added precisely this (**FIG. 6**): We inflated the corners until the stylus was detected within one of them, and then deflated all of them while the stylus was moving. Thereafter, users were able to easily hit the corners on pen-down and also avoid hitting them accidentally while moving the stylus.

[0049] An observation during a user study prompted the next iteration on the corners. A right-handed user with a chronic wrist injury held the stylus at a fairly shallow angle relative to the PDA screen. The result was that the elevated edge of the plastic square prevented the tip of the stylus from getting close to the right side of the square. We provided