Where for a given **i**, s_i represent a particular inter-word space, **i** is a reference number to indicate which space is referenced, and **k** represents the total number of inter-word space in a text row concerned. In FIG. 3A, **L** = 8 + 6 + 5 + 7 + 6 + 9 + 6 + 6 = 53.

In one particular embodiment, the inter-word space **S** = [s_1, s_2, s_3 ... s_7, s_8] is changed into **S'** = [s_1', s_2', s_3' ... s_7', s_8'] by modifying the inter-character space [c_1, c_2 ... c_n] of each word in the text row. For each word, the inter-character space, c_i, is reduced by 1 pixel if $c_i > 2$ pixels. Hence, the overall inter-word space is increased such that for each s_i, $s_i' \geq s_i$. By increasing the values of s_i', the total length of **L'** of the new inter-word space satisfies the condition: **L'** $\geq$ **L**.

FIG. 4 shows modification 70 of inter-word spacing by changing inter-character spacing 72, 74 in accordance with an embodiment of the invention. In this example, the inter-word spacing is provided by changing the inter-word spacing in FIG. 3A. In FIG. 4, **L'** = 8 + 9 + 8 + 7 + 6 + 12 + 8 + 9 = 67.

For convenience, the function Sign([s_1, s_2 ... s_n]) is defined by:

Let s_{min} = floor integer (average of the ε smallest value in [s_1, s_2 ... s_n]).

Sign([s_1, s_2 ... s_n]) = $g_1|g_2|$ ... $|g_n$

where

g_i = + if $s_i > s_{min}$

g_i = - if $s_i \leq s_{min}$

The value ε is greater than or equal to the number of "-" g_i selected.

The data to be hidden is represented in binary form as a sequence of "1"s and "0"s.

In one particular embodiment, the inter-word space **S"** = [s_1", s_2", s_3" ... s_7", s_8"] such that: