

IDENTIFYING CHARACTERS FROM SCRIBED INDICIUM

FIELD OF INVENTION

5

The present invention relates broadly to a method of assigning an adaptation template to a scribed indicium representing a character, to a method of recognising a scribed indicium representing a character, and to an electronic device.

BACKGROUND OF THE INVENTION

10

The recognition of characters from user dependent scribed indicium is one of the prominent challenges in modern human-device interface technologies. For example, with handheld devices being more and more popular, the need for an easy way to enter text such as short text messages becomes ever more important. When a touch screen is available, handwriting recognition of scribed characters (indiciu)

15 provides a most natural text input interface, especially for but not limited to, oriental ideographic character sets such as Chinese.

Online recognition of handwritten scribed characters is known as a difficult pattern recognition problem. One difficulty comes from the large shape variability within a character. Different writers may write different ink shapes in their daily lives.

20 Figure 1 illustrates two kinds of shape variations for Chinese characters. In Figure 1 (a) and (b), the characters 100, 102 are the standard shapes respectively, while characters 104, 106 illustrate user dependent variations respectively. By applying a statistical pattern recognition method, a recognizer can probably correctly recognize those character shapes that have appeared in training samples. But in practice, it is

25 almost impossible to collect enough training samples to cover all those variations.

There is therefore a need for a flexible recognizer technique that can be adapted to recognize user dependent scribed characters such as shorthand scribed indicium representing relatively complex characters.

30

SUMMARY OF THE INVENTION

In accordance with a first aspect of the present invention there is provided a method of assigning an adaptation template to a scribed indicium representing a character, the method comprising extracting a plurality of stroke segment vectors from the scribed indicium; determining for each extracted stroke segment vector the nearest of a plurality of stored probability distribution functions (PDFs) for respective reference stroke segment vectors, and assigning the adaptation template to the scribed indicium, wherein the adaptation template comprises identifiers identifying the nearest stored PDF for each extracted stroke segment vector and links the scribed indicium to the character it represents, for storing in an adaptation dictionary.

The plurality of stored PDFs may be stored in the form of a PDF dictionary and a PDF codebook; wherein the number of PDFs stored in the PDF codebook is smaller than the number of PDFs in the PDF dictionary, and wherein each PDF in the PDF dictionary is associated with a pointer identifying one most similar PDF in the PDF codebook.

The PDFs in the PDF codebook may be selected based on a quantization of the PDFs in the PDF dictionary utilising the Kullback-Leibler divergence to calculate a distortion measure.

The stored PDFs for the respective reference stroke segment vectors may be associated with different reference characters.

The method may further comprise assigning a time-of-life value, ToL, to the adaptation template, and storing the adaptation template and assigned ToL in an adaptation dictionary.

The step of storing the adaptation template may comprise replacing an existing entry in the adaptation dictionary having the highest ToL, and increasing the ToLs of other existing entries by a predetermined amount.

In accordance with a second aspect of the present invention there is provided a method of recognising a scribed indicium representing a character, the method comprising extracting a plurality of stroke segment vectors from the scribed indicium; determining, for each extracted stroke segment vector, the nearest of a plurality of stored (PDFs) for respective reference stroke segment vectors, and searching through an adaptation template dictionary to determine a match for recognition of the scribed indicium based on the determined nearest

PDFs, wherein each adaptation template comprises identifiers identifying one of the stored PDFs for each stroke segment vector of a reference character.

The method may further comprise assigning a new adaptation template according to the first aspect of the present invention.

5 In accordance with a thirs aspect of the present invention there is provided a electronic device comprising a user interface for input of a scribed indicium representing a character; a memory unit; and a processor; wherein the processor extracts a plurality of stroke segment vectors from the scribed indicium; determines for each extracted stroke segment vector, the nearest of a
10 plurality of stored PDFs for respective reference stroke segment vectors, in the memory unit, and assigns an adaptation template to the scribed indicium, and wherein the adaptation template comprises identifiers identifying the nearest stored PDF for each extracted stroke segment vector and links the scribed indicium to the character it represents, for storing in an adaptation dictionary in
15 the memory unit.

The plurality of stored PDFs may be stored in the form of a PDF dictionary and a PDF codebook in the memory unit; wherein the number of PDFs stored in the PDF codebook is smaller than the number of PDFs in the PDF dictionary, and wherein each PDF in the PDF dictionary is associated with a pointer
20 identifying one most similar PDF in the PDF codebook.

The PDF dictionary and the PDF codebook may be stored in a read only memory element of the memory unit.

The PDFs in the PDF codebook may be selected based on a quantization of the PDFs in the PDF dictionary utilising the Kullback-Leibler divergence to
25 calculate a distortion measure.

The stored PDFs for the respective reference stroke segment vectors may be associated with different reference characters.

The processor may further assign a time-of-life value, ToL, to the adaptation template, and the adaptation template and assigned ToL are stored in
30 the adaptation dictionary in the memory unit.

The processor may replace an existing entry in the adaptation dictionary having the highest ToL, and increases the ToLs of other existing entries by a predetermined amount.

The adaptation dictionary may be stored in a random access memory
35 element or flash element of the memory unit.

In accordance with a fourth aspect of the present invention there is provided a electronic device comprising a user interface for input of a scribed indicium representing a character; a memory unit; and a processor; wherein the processor extracts a plurality of stroke segment vectors from the scribed indicium; determines for each extracted stroke segment vector, the nearest of a plurality of stored PDFs in the memory unit, and searches through an adaptation dictionary in the memory unit to determine a match for recognition of the scribed indicium based on the determined nearest PDFs, wherein adaptation templates in the adaptation dictionary each comprise identifiers identifying one of the stored PDFs for each stroke segment vector of a reference character.

The processor may assign a new adaptation template to the input character, the adaptation template comprising an array of identifiers identifying the nearest stored PDF for each stroke segment vector, for storing in the adaptation dictionary.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the invention will be better understood and readily apparent to one of ordinary skill in the art from the following written description, by way of example only, and in conjunction with the drawings, in which:

Figure 1 shows examples of character shape variability.

Figure 2 is a schematic drawing illustrating an electronic device for implementation of the method and system in accordance with an example embodiment.

Figure 3 shows a flowchart illustrating a classifier for character recognition with user adaptation functionality, according to an embodiment of the present invention.

Figure 4 is a schematic drawing illustrating stroke matching based classification.

Figure 5 illustrates schematically Gaussian Probability Density Function codebook selection for segments of a scribed indicium.

Figure 6 shows a flowchart of generating a new adaptation template, according to an example embodiment.

Figure 7 shows a flowchart of adding a new adaptation template, according to an example embodiment.

5

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT OF THE INVENTION

10 With reference to Figure 2, there is illustrated an electronic device 201. The device 201 illustrated is a handheld PDA phone comprising a radio frequency communications unit 202 coupled to be in communication with a processor 203. A user interface in the form of a touch screen 205 (typically a Liquid Crystal Display) and an optional keypad 206 are also coupled to be in communication with
15 the processor 203.

 The processor 203 includes an encoder/decoder 211 with an associated code Read Only Memory (ROM) 212 storing data for encoding and decoding voice or other signals that may be transmitted or received by electronic device 201. The processor
20 203 also includes a micro-processor 213 coupled to both the encoder/decoder 211 and an associated character Read Only Memory (ROM) 214 by a common bus 218. Micro-processor 213 is also coupled to a Random Access Memory (RAM) 204 and a static programmable memory 216 by the common bus 218. The static programmable memory 216 stores, amongst other features and functions, adaptation templates for
25 character recognition, a phone book database of numbers and associated identifiers (names of people associated with each of the numbers).

 Auxiliary outputs of micro-processor 213 are coupled to an alert module 215 that typically contains a speaker, vibrator motor and associated drivers. The
30 character Read only memory 214 stores code for decoding or encoding text messages that may be received by the communication unit 202, input at the touch screen 205 or optional keypad 206. In this embodiment the character Read Only Memory 214 also stores operating code (OC) for micro-processor 213, and a fixed recognition dictionary for character recognition. The operating code (OC) is used to
35 run applications on the electronic device 201.

The radio frequency communications unit 202 is a combined receiver and transmitter having a common antenna 207. The communications unit 202 has a transceiver 208 coupled to antenna 207 via a radio frequency amplifier 209. The transceiver 208 is also coupled to a combined modulator/demodulator 210 that couples the communications unit 202 to the processor 203.

Figure 3 shows a flowchart of a classifier with user adaptation functionality, according to an example embodiment of the present invention. At box 301, a scribed indicium representing a character is entered, for example utilising a touch screen of a handheld device such as the PDA phone 201 (Figure 2).

For the scribed indicium, the block 302 performs smoothing, noise removal, size normalization and extracts stroke segment vectors. Smoothing is done by averaging the coordinates (x,y) of a point with its neighbors, and replacing the (x,y) coordinate values of the point with the computed average. After smoothing, the redundant points caused by hi-frequency noise are removed. The redundant points are those points whose (x,y) coordinate values are the same as their predecessors. Then the indicium are scaled to fit a standard rectangle, e.g. 180x180 in the embodiment of this invention.

A stroke segment in the example embodiment is the line segment between the local maximum directional angle changes of the scribed indicium. The stroke segment vector is a two-dimensional vector $o_i = (d_x, d_y)$ derived from each segment, where d_x and d_y are the coordinate differences between the starting and the ending points of the segment. Depending on the complexity of the scribed indicium, the number of stroke segment vectors may vary.

Boxes 303 and 304 compose a searching stage 308 of the flowchart 300 for the example embodiment. In the search stage 308, each stroke segment extracted at block 302 is modelled by a probability distribution function (PDF), for example a Gaussian mixture PDF in the form of $b_j(.) = \sum_{k=1}^K \omega_{jk} N(.; \mu_{jk}, \sigma_{jk})$. As shown in Figure 4, each character model in a fixed recognition dictionary is a sequence of

Gaussian mixture PDFs $(b_1, b_2, \dots, b_N)$, e.g. 402, with N the average stroke number for the character. A dynamic programming (DP) procedure is used to match each of the stored character models e.g. 402 (PDF sequence in the fixed dictionary) with stroke segment sequences modelled based on the extracted stroke segments, e.g. 406.

5 The character model that gives the maximum matching scores is the recognition result for the scribed indicium 404.

It will be appreciated by a person skilled in the art, that the character models, e.g. 402, stored in the fixed recognition dictionary originate from a pool of training samples previously collected. For each training sample, the stroke segments of a scribed indicium representing a given character are extracted. A character model, that is a sequence of Gaussian mixture PDFs of the respective extracted stroke segments, is modelled and linked to the character which the sample scribed indicium represents.

Returning to Figure 3, block 304 performs searching in an adaptation dictionary in the example embodiment. Details of the adaptation dictionary will be described below. In block 305, if an input character is not correctly recognized after the search stage 308, a new character model is generated (in block 306) and an update of the adaptation dictionary is performed (in block 307). Details of blocks 306 and 307 of the example embodiment will be described below.

An ideographic character set, such as e.g. the Chinese character set, has a very large vocabulary, and each character may have numerous stroke segments (see character 404 and segments, e.g. 406, in Figure 4). Because each stroke segment is modelled by e.g. a Gaussian mixture PDF, the fixed recognition dictionary used in block 403 may occupy very large storage memory. In the example embodiment, all the Gaussian PDFs in block 303 are quantized into a fixed number (e.g. 256) of Gaussian PDFs in a codebook. In the Gaussian PDF clustering of the example embodiment, the distortion measure between two Gaussian PDFs $N(\cdot; \mu_i, \Sigma_i)$ and $N(\cdot; \mu_j, \Sigma_j)$ is calculated as the *Kullback-Leibler (KL) divergence*. The *Kullback-Leibler divergence* of two Gaussian PDFs $N(\cdot; \mu_i, \Sigma_i)$ and $N(\cdot; \mu_j, \Sigma_j)$ is defined as:

$$J_{KL}(i, j) = \frac{1}{2} \text{tr}[(\Sigma_i - \Sigma_j)(\Sigma_i^{-1} - \Sigma_j^{-1})] + \frac{1}{2} \text{tr}[(\Sigma_i^{-1} + \Sigma_j^{-1})(\mu_i - \mu_j)(\mu_i - \mu_j)^T]$$

where “tr[]” denotes the *trace* of a matrix.

For further reference, the reader is directed to “S. Kullback, *Information Theory and Statistics*. New York: Wiley, 1959” that’s is incorporated into this specification by reference.

5

As illustrated in Figure 5, a user created (i.e. shorthand character or personalized character) scribed indicium 501 is segmented into segment model sequences. Each of the segment model sequences has a Gaussian PDF and for each Gaussian PDF e.g. 500 in the fixed recognition dictionary 502, a most similar PDF referred to as codeword 504 in the codebook 506 is found. Thus, each Gaussian PDF, e.g. 500, used in block 203 (Figure 2), is actually an index pointing to one of the 256 codewords, e.g. 504, in the codebook in the example embodiment.

10

Figure 6 describes the method to generate a new adaptation template in the example embodiment. For a scribed indicium representing a character input at block 601, and after preprocessing the indicium, a stroke segment vector sequence with T stroke segment vectors $\{o_1, o_2, o_3, \dots, o_T\}$ is extracted at block 602. A counter $i = 0$ is set (block 603), and for each stroke segment vector o_i , the nearest Gaussian codeword $N(\cdot; \mu_j, \Sigma_j)$, where $j = \arg \max_{1 \leq k \leq 256} N(o_i; \mu_k, \Sigma_k)$, is determined (block 604). $A[i]=j$ is set (block 605), identifying the nearest Gaussian codeword determined for the i th stroke vector. Blocks 604 and 605 are repeated until $i = T$ (block 606). The identifiers $A[0..N-1]$ are assigned as the adaptation template linking the scribed indicium to the character which the scribed indicium represents in the example embodiment.

20

25

The character which the scribed indicium represents may be specified as part of the initial input of the scribed indicium, for example by selection out of a plurality of characters displayed on the touch screen 205 (Figure 2) of the PDA phone 201 (Figure 2), in a “learning mode”. However, it will be appreciated that the character which the scribed indicium represents may be specified at different points of the generation of the new adaptation template in different embodiments, including after the determination of the index array $A[0..T-1]$ during the adding of the adaptation template to the adaptation dictionary (block 607).

30

The fixed recognition dictionary mentioned in block 303 of Figure 3 can be stored in the ROM 214, while the adaptation dictionary (block 304 in Figure 3) is stored in memory 216 and downloaded into the RAM 204 during power up. The available RAM 204 for a handwriting engine may be very limited, for example, only 100 adaptation templates may be stored. In the example embodiment, a method of updating the adaptation dictionary (block 307 Figure 3) is provided as described below with reference to Figure 7.

An incremental integer number is used to record a time-of-life (ToL) value for each existing adaptation template, this number essentially identifies the order in which each adaption template is stored in an adaption dictionary. For each scribed indicium for a character input at block 701, the stroke segment vector sequences are extracted (block 702), and matched with the fixed recognition dictionary and the adaptation templates in the adaptation dictionary (block 703). The matching with the adaptation templates comprises determining, for each extracted stroke segment vector, the nearest of a plurality of stored PDFs for respective reference stroke segment vectors, and searching for an adaptation template which identifies the same PDFs as those determined for the extracted stroke segment vectors.

From block 703 there are two outputs feeding into blocks 704 and 708. Considering the feed into block 708 from block 703, if the top candidate (C1) is from the adaptation dictionary (block 708), then the ToL value of this template is set to 0 (block 709). Alternatively, if at block 708 it is determined that the top candidate (C1) is not from the adaptation dictionary then block 710 is bypassed and the ToL value is unchanged.

Considering the feed into block 704 from block 703. In block 704, if the candidate character needs to be added to the adaptation dictionary, i.e. the candidate C1 (block 703) is not a correct recognition of the input character, a new adaptation template is generated as described above with reference to Figure 6. Otherwise the method terminates.

If at step 704, it is determined that candidate character needs to be added to the adaptation dictionary then at block 705 it is checked whether the adaptation dictionary is full. If the adaptation dictionary is not full, then a new

template is added to the dictionary (block 710) and the ToL value for the template set to zero. If the adaptation dictionary is full, then the template that has the maximum ToL value is found, and the new template is used to replace it (block 706). The ToL value for the new template is also set to zero. Finally, the ToL values are increased (e.g. $ToL = ToL + 1$) for all the existing adaptation templates (block 707). It should be noted that although all ToL values are incremented at block 707, the value ToL value count may become unduly large for some characters and thus normalization of these count values may be possible, for instance, by integer division of all ToL values.

The example embodiment provides a method to recognize the user dependent characters. Based on stochastic segment matching classification and a Gaussian PDF clustering method, an adaptation template generating method and a dictionary updating method are provided in the example embodiment to improve an hand writing recognition (HWR) product's usability for a specific user. Thus, a user can scribe personalized indicium or indicia that are shorthand for relatively complex characters. The scribed indicium or indicia is associated with the complex character via the codebook and can be used to input the character into the device.

It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present invention as shown in the specific embodiments without departing from the spirit or scope of the invention as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive.

CLAIMS

We claim:

- 5 1. A method of assigning an adaptation template to a scribed indicium representing a character, the method comprising

extracting a plurality of stroke segment vectors from the scribed indicium;

determining for each extracted stroke segment vector the nearest of a plurality of stored probability distribution functions (PDFs) for respective

10 reference stroke segment vectors, and

assigning the adaptation template to the scribed indicium, wherein the adaptation template comprises identifiers identifying the nearest stored PDF for each extracted stroke segment vector and links the scribed indicium to the character it represents, for storing in an adaptation dictionary.
- 15 2. The method as claimed in claim 1, wherein the plurality of stored PDFs are stored in the form of a PDF dictionary and a PDF codebook; wherein the number of PDFs stored in the PDF codebook is smaller than the number of PDFs in the PDF dictionary, and wherein each PDF in the PDF dictionary is associated with a pointer identifying one most similar PDF in the PDF codebook.
- 20 3. The method as claimed in claim 2, wherein the PDFs in the PDF codebook are selected based on a quantization of the PDFs in the PDF dictionary utilising the Kullback-Leibler divergence to calculate a distortion measure.
4. The method as claimed in any one of the preceding claims,

25 wherein the stored PDFs for the respective reference stroke segment vectors are associated with different reference characters.
5. The method as claimed in any one of the preceding claims, further comprising assigning a time-of-life value, ToL, to the adaptation template, and storing the adaptation template and assigned ToL in an adaptation dictionary.
- 30 6. The method as claimed in claim 5, wherein the step of storing the adaptation template comprises replacing an existing entry in the adaptation dictionary having the highest ToL, and increasing the ToLs of other existing entries by a predetermined amount.
7. A method of recognising a scribed indicium representing a

35 character, the method comprising

extracting a plurality of stroke segment vectors from the scribed indicium;

determining, for each extracted stroke segment vector, the nearest of a plurality of stored (PDFs) for respective reference stroke segment vectors, and

searching through an adaptation template dictionary to determine a match for recognition of the scribed indicium based on the determined nearest PDFs, wherein each adaptation template comprises identifiers identifying one of the stored PDFs for each stroke segment vector of a reference character.

8. The method as claimed in claim 7, further comprising assigning a new adaptation template according to the method as claimed in any one of claims 1 to 6, when no match is found.

9. An electronic device comprising

a user interface for input of a scribed indicium representing a character;

a memory unit; and

a processor;

wherein the processor extracts a plurality of stroke segment vectors from the scribed indicium; determines for each extracted stroke segment vector, the nearest of a plurality of stored PDFs for respective reference stroke segment vectors, in the memory unit, and assigns an adaptation template to the scribed indicium, and

wherein the adaptation template comprises identifiers identifying the nearest stored PDF for each extracted stroke segment vector and links the scribed indicium to the character it represents, for storing in an adaptation dictionary in the memory unit.

10. The device as claimed in claim 9, wherein the plurality of stored PDFs are stored in the form of a PDF dictionary and a PDF codebook in the memory unit; wherein the number of PDFs stored in the PDF codebook is smaller than the number of PDFs in the PDF dictionary, and wherein each PDF in the PDF dictionary is associated with a pointer identifying one most similar PDF in the PDF codebook.

11. The device as claimed in claim 10, wherein the PDF dictionary and the PDF codebook are stored in a read only memory element of the memory unit.

12. The device as claimed in claims 10 or 11, wherein the PDFs in the PDF codebook are selected based on a quantization of the PDFs in the PDF dictionary utilising the Kullback-Leibler divergence to calculate a distortion measure.

13. The device as claimed in any one of claims 9 to 12, wherein the stored PDFs for the respective reference stroke segment vectors are associated with different reference characters.

5 14. The device as claimed in any one of claims 9 to 13, wherein the processor further assigns a time-of-life value, ToL, to the adaptation template, and the adaptation template and assigned ToL are stored in the adaptation dictionary in the memory unit.

10 15. The device as claimed in claim 14, wherein the processor replaces an existing entry in the adaptation dictionary having the highest ToL, and increases the ToLs of other existing entries by a predetermined amount.

16. The device as claimed in any one of claims 9 to 15, wherein the adaptation dictionary is stored in a random access memory element or flash element of the memory unit.

15 17. An electronic device comprising
a user interface for input of a scribed indicium representing a character;
a memory unit; and
a processor;
wherein the processor extracts a plurality of stroke segment vectors from the scribed indicium; determines for each extracted stroke segment vector, the
20 nearest of a plurality of stored PDFs in the memory unit, and searches through an adaptation dictionary in the memory unit to determine a match for recognition of the scribed indicium based on the determined nearest PDFs, wherein adaptation templates in the adaptation dictionary each comprise identifiers identifying one of the stored PDFs for each stroke segment vector of a reference
25 character.

18. The device as claimed in claim 17, wherein, if no match is found, the processor assigns a new adaptation template to the input character, the adaptation template comprising an array of identifiers identifying the nearest stored PDF for each stroke segment vector, for storing in the adaptation
30 dictionary.