

Adaptive Correcting Strokes Extracted From Chinese Characters in Digital Ink of Non-Native Writers Based on Comprehensive Visualization

Hao Bai^{1,2}, Xiwen Zhang¹

¹College of Information Science, ²Advanced Chinese Training
Beijing Language and Culture University
Beijing, China
E-mail: {baihao, zxw}@blcu.edu.cn

Abstract—The correcting process for strokes extracted from Chinese characters is the necessary step to extract the errors of writing errors automatically. Visualization of extracted strokes is the prerequisite for manual correction. Therefore, visualization and adaptive correction methods are proposed. To reduce the cognitive burden of correcting, color, brightness, saturation and order number is comprehensively used to visualize extracted strokes. And tag list is applied for correcting different types of extracted strokes, which provides the training set for error extraction classifier in future work. After experimental verification, the method is effective in operational complexity and efficiency.

Keywords—Correction; Visualization; Digital ink; Stroke extraction; Chinese character

I. INTRODUCTION

In the field of computer-assisted Chinese teaching, many scholars have been exploring in recent years [1-3], and developed some can be applied to the actual teaching system of Chinese characters teaching [4-8], but whether from the implementation method or teaching performance, the effect is slightly less. In general, the computer system for the basic process of teaching Chinese characters for the user (especially studying Chinese as the second language) using handwriting input device to enter the Chinese characters into the computer system, the system firstly recognizes the handwriting character to the text character and then find the corresponding template character in the template library. After extracting and matching strokes of the writing and the template character, according to the matching results, writing errors are classified and recognized. Finally, writing feedback is given. In the whole process above, the computer needs to analyze the writing error from extracted strokes. As a result, the corrected training data is necessary for the training of the error classifier. It is impossible for the computer system to completely match all the strokes of all the handwriting characters; so in the production of training data, adaptive correction method is applied to correct different types of extracted strokes. Visualization is the prerequisite to manual correction. Improved visualization could reduce cognitive burden effectively and efficiently. The correction procedure requires simple operation, accurate tag and rich expression including correction information such as many types of writing errors.

There are few researches on the correction method of extracted stroke results at present. In terms of visualization,

the color approach is used to visualize the extracted strokes [9]. However, when the number of strokes increasing, the color of strokes is repeatedly rendered that produces more cognitive burdens. In addition, only colors cannot fully express the information of extracted results, such as order information of strokes. The correction method mainly divided into gesture and tagging correction. Commonly digital ink data gesture correction method [10] utilizes mouse and other input devices to correct object for simple and direct operation; but if the amount of data is large, gesture correction requires higher operation skill and precision that results in the reduction of the correction efficiency. On the other hand, the method of tagging correction can be used to correct data along with tagging information. With the better interactive visualization method, it could achieve higher operational efficiency.

In this paper, the approach of correcting the results of extracted strokes is to use the comprehensive visualization method to reduce the cognitive burden. And then we correct the result of extracted strokes and make the necessary data tag information to provide the prerequisite for writing quality feedback in further work. The rest of this paper is organized as follows: Section 2 introduces the visualization for the result of extracted strokes. Section 3 discusses different types of tagging expression for various extracted strokes results. Sect. 4 presents the operation approach using the tagging list to correct different results of extracted strokes. Sect. 5 describes the experiment results and analysis. Sect. 6 draws conclusion from the experiment.

II. VISUALIZATION OF EXTRACTED STROKES

Visualization of extracted strokes is a prerequisite for manual correction. Intuitive and accurate visualization can effectively reduce the cognitive burden of operators and improve their efficiency of corrective operation. This paper adopts colors, graphics and digital texts to visual express the matching result of strokes in digital ink Chinese character. The use of non-repetitive colors render different Chinese characters strokes, especially for the adjacent ones, of which the colors were distinguishing; the circle graphics is used to mark the direction information of strokes, which is that the black circles represent the start point of the stroke and the white ones represent the end point of the stroke; digital labels show the writing order index of strokes in the Chinese character. According to the Chinese character strokes "from left to right, from top to bottom" writing rules, the digital

label will be placed in the stroke of the upper left corner near the bounding box of the stroke, in order to avoid overlapping with the writing strokes. Meanwhile the color of digital label is same to the stroke's, which reduces cognitive burden caused by adjacent strokes. The specific steps of the visualization are shown as follows. Fig. 1 illustrates the result of this visualization.

Step 1: The number of strokes is used to divided the half value of hue spectrum to get the weight of stroke color W .

Step 2: Parity of strokes' order is calculated.

Step 3: If the index of stroke is odd, its color is rendered by $HSB(\text{hue}, \text{saturation}, \text{brightness})$ function, of which the value of *hue* is $\text{Index} \times W$, *saturation* is 100% and *brightness* is 100%. *Index* is the index of strokes.

Step 4: If the index of stroke is even, its color is rendered by $HSB(\text{Index} \times W + 180, 50\%, 50\%)$.

Step 5: According to colors of strokes and location of up-left corner of bounding box of strokes, digital indexes of strokes are rendered.

Step 6: Circling the start point of the stroke with black color and the end with white.

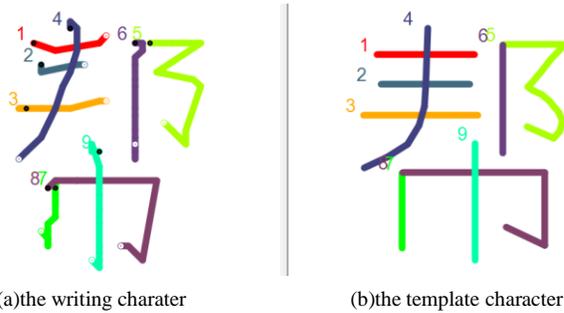


Figure 1. Example of visaulization of extracted strokes

III. TAGGING EXPRESSION OF EXTRACTED STROKES

In the teaching assistant system of Chinese characters, computer system needs to classify the error of writing according to the result of extracted strokes, so as to give the feedback of the Chinese characters handwriting. The training of the writing error classifier requires the correct training data of extracted strokes; but because of the differences in the quality of handwriting and its randomness, it is hardly for computer system to match all the strokes of all character data completely correct. As a result, manual correction is needed to adjust the results of matching strokes as training data. The correction is required as simple and accurate operation but rich of information expression such as types of writing error. Therefore, we use a tag list to reduce the two-dimensional writing data to the one-dimensional array. So that only stroke digital indexes and mark circles remain in the writing panel that solves the problem of information clutter.

A. Definition of Tag List

The approach first defines the data structure of the tag list *Matchlist* includes writing stroke index, template stroke

index and mark points coordinate information. The following TABLE 1 illustrates tagging different types of extracted strokes.

TABLE I. ILLUSTRATION OF TAGGING EXPRESSION

Type	Illustration	Examples
one to one	One handwriting stroke correspond to one template stroke	1-1; 2-3;
concatenated stroke	One handwriting stroke correspond to multiple template strokes	1.1-1; 1.2-2; 1.3-3;
broken stroke	multiple handwriting strokes correspond to one template stroke	1.1-1; 1.2-1; 1.3-1;
extra stroke	One handwriting stroke correspond to none template stroke	1-0;
redundant stroke	Sub stroke of handwriting correspond to none template stroke	1.1-1; 1.2-0;
missing stroke	None handwriting stroke correspond to any template stroke	0-1; 0-4;

B. Recording of Tag List

The method uses the xml file format to store *Matchlist*, which can be used for writing the error classifier after saving the correction information, without directly modifying the original handwriting data and maintaining the integrity of the original data. TABLE 2 illustrates each item in *Matchlist* saved in xml file. Fig. 2 is an example of a saved file fragment.

TABLE II. ILLUSTRATION OF MATCHLIST

Item Name	Illustration	Object Type
MatchItem	Each object is pair of a template stroke and its writing extracted stroke.	MatchItem
ID	Index number of each matchItem	Integer
TestIndex	Index number of strokes in the writing character	Float
TempIndex	Index number of strokes in the template character	Integer
OperatorPoints	Start point and end point in each MatchItem.	Point

```

- <MatchItem ID="0" TestIndex="1" TempIndex="1">
  <operatorPoints X="91" Y="57" />
</MatchItem>
- <MatchItem ID="1" TestIndex="2" TempIndex="2">
  <operatorPoints X="260" Y="63" />
</MatchItem>
- <MatchItem ID="2" TestIndex="3" TempIndex="3">
  <operatorPoints X="277" Y="84" />
</MatchItem>
- <MatchItem ID="3" TestIndex="4" TempIndex="4">
  <operatorPoints X="87" Y="100" />
</MatchItem>
- <MatchItem ID="4" TestIndex="5" TempIndex="8">
  <operatorPoints X="138" Y="326" />
  <operatorPoints X="258" Y="313" />
</MatchItem>

```

Figure 2. Example of a saved xml fragment

IV. CORRECTION OPERATION FOR EXTRACTED STROKES

For the different types of digital ink extracted strokes results, the correction approach proposed in this paper adopts *Matchlist* tagging operation to correct and prepare for the next step in extraction of types of writing errors. The specific operations are show as follows.

A. One to One

- Select the corresponding stroke in the *Matchlist*.
- Click "Edit" to set the stroke start and end point: click "Start" and in the writing panel click for the start point; click "End" and click for the end position in the writing panel on the left side.
- Edit the text box beneath *Matchlist*. The left side is the writing stroke index. The right side is the template stroke index.
- Click "OK" to complete correction, as shown in Fig. 3.



Figure 3. Example of "One to One" correction

B. Concatenated Stroke

- Select the corresponding stroke in the *Matchlist*.
- Click "Add" to set the stroke start and end point: click "Start" and in the writing panel click for the start point; click "End" and click for the end position in the writing panel on the left side.
- Edit the text box beneath *Matchlist*. The left side is the writing stroke index. The right side is the template stroke index. The input format is "**.*-**" and the integer part of its left side is writing stroke index and the fractional part is written in sequence; the right "***" is its corresponding template stroke index.
- Click "OK" to complete correction, as shown in Fig. 4.



Figure 4. Example of "Concatenated Stroke" correction

C. Broken Stroke

- Select the corresponding stroke in the *Matchlist*.
- Click "Edit" to set the stroke start and end point: click "Start" and in the writing panel click for the start point; click "End" and click for the end position in the writing panel on the left side.
- Edit the text box beneath *Matchlist*. The left side is the writing stroke index. The right side is the template stroke index.
- Click "OK" to complete correction, as shown in Fig. 5.



Figure 5. Example of "Broken Stroke" correction

D. Extra Stroke

- Select the corresponding stroke in the *Matchlist*.
- Click "Edit".
- Edit the text box beneath *Matchlist* as "**.*-0*". The left side is the writing stroke index. The right side "*0*" means none of the template stroke corresponding.
- Click "OK" to complete correction, as shown in Fig. 6.

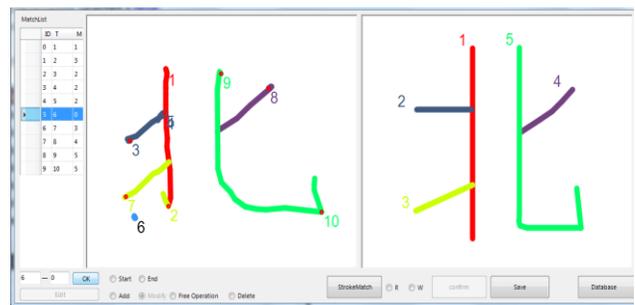


Figure 6. Example of "Extra Stroke" correction

E. Redundant Stroke

- Select the corresponding stroke in the *Matchlist*.
- Click "Edit".
- Edit the text box beneath *Matchlist* as "**.*-0*". The integer part of its left side is writing stroke index and the fractional part is written in sequence; the right side "*0*" means none of the template stroke corresponding.
- Click "OK" to complete correction, as shown in Fig. 7.

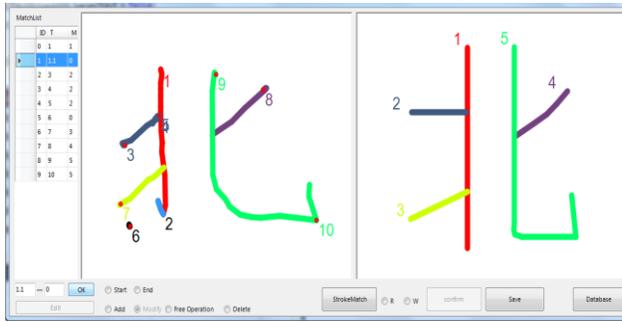


Figure 7. Example of "Redundant Stroke" correction

F. Missing Stroke

- Select the corresponding stroke in the *Matchlist*.
- Click "Add".
- Edit the text box beneath *Matchlist* as "0-*". The left side "0" means none of the writing stroke index corresponding to the template stroke. The right side is the template stroke index.
- Click "OK" to complete correction, as shown in Fig. 8.

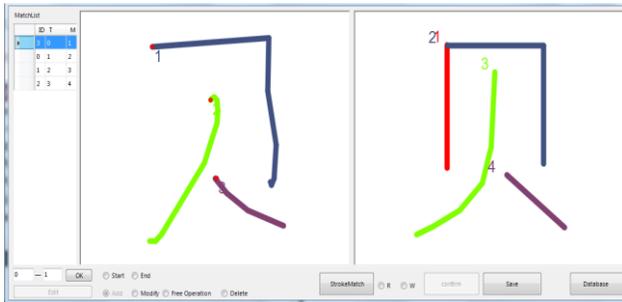


Figure 8. Example of "Missing Stroke" correction

V. EXPERIMENTAL RESULTS

The proposed approach is tested in 19815 Chinese characters of 535 kinds including 1094 characters in 6 kinds of errors after strokes extraction, which are realistically handwriting by 127 different foreign Chinese learners [9]. All experiments are run on a PC with Intel Core i7 and 16G RAM. We designed a validation experiment for the proposed correction. The experiment sets the number of keyboard operations and mouse operations for different types of matching results, to verify the complexity of correction. The experiment also records average time of the correction to verify the efficiency. The experimental results are shown in TABLE III.

TABLE III. RESULTS OF EXPERIMENT

Type	Number of characters	Number of keyboard operations (per character)	Number of mouse operations (per character)	Time (sec)
one to one	761	1.1	9.2	28.01
concatenated stroke	133	6.3	17.8	48.83
broken stroke	101	1.2	9.1	22.34
extra stroke	15	1.4	5.1	18.27
redundant stroke	65	4.5	7.3	25.77
missing stroke	19	3.3	6.1	18.38

It can be seen from the experimental results that the proportion of the one-to-one error type is the largest and its correction is within the range of the operation complexity and time consumption. Besides, the number of operation of concatenated stroke is more because strokes are need to splitting extraction, resulting in new strokes, which increases the operation complexity and time consumption.

VI. CONCLUSION

In the teaching assistant system of Chinese characters, the correction process for matching results of strokes in one character is the necessary step to further extract the types of writing errors intelligently. The visualization of extracted strokes is the prerequisite for manual correction. Therefore, this paper proposes the comprehensive visualization of extracted strokes and adaptive correcting operations. The visualization uses color, brightness, saturation and digital index number to reduce the cognitive burden when correcting. The tagging list is used to express the different types of extracted strokes and prepare for the next type of error type extraction. After experimental verification, the method is effective.

ACKNOWLEDGMENT

This work is supported in part by Science Foundation of Beijing Language and Culture University (supported by "the Fundamental Research Funds for the Central Universities") (No.17YJ080203), Young and Middle-aged Academic Backbone Support Program of Beijing Language and Culture University and WuTong innovation platform of Beijing Language and Culture University (supported by "the Fundamental Research Funds for the Central Universities") (No.16PT04). The authors would like to thank Institute of Big Data and Language Education of Beijing Language and Culture University for sharing Skeletal Chinese Character Database.

REFERENCES

- [1] C. B. Zhuang and L. W. Jin, "An Intelligently Verified Algorithm for Correctness and Calligraphy of On-line handwritten Chinese Characters," *Signal Processing*, vol. 21, Aug. 2005, pp. 276-279.
- [2] W. P. Xia and L. W. Jin, "A Method for Layout Evaluation of Online Handwritten Chinese Character Quality Based On Template," *Proc. Chinese Conference on Pattern Recognition*, 2008, pp. 354-359.

- [3] Z. H. Hu, Y. Xu, L. S. Huang, and H. Leung, "A Chinese Handwriting Education System with Automatic Error Detection," *Journal of Software*, vol. 4, Apr. 2009, pp. 101-107.
- [4] J. Li and X. Zhang, "The design and implementation of multimedia intelligent tutoring system for Chinese characters," *Proc. IEEE First International Conference on Multi-Media Engineering Education*, 1994, pp. 459-463.
- [5] H. C. Lam, W. W. Ki, N. Law, A. L. S. Chung, P. Y. Ko, A. H. S. Ho, and S. W. Pun, "Designing CALL for learning Chinese characters," *Journal of Computer Assisted Learning*, vol. 17, 2001, pp. 115-128.
- [6] C. C. Han, C. H. Chou and C. S. Wu, "An interactive grading and learning system for chinese calligraphy," *Machine Vision & Applications*, vol. 19, 2008, pp. 43-55.
- [7] V. Tam and K. W. Yeung, "Learning to write Chinese characters with correct stroke sequences on mobile devices," *Proc. International Conference on Education Technology and Computer*, 2010, pp. 395-399.
- [8] E. Xun, L. Xiaochen, A. N. Weihua, Y. Sun, and I. Ramp, "Stroke Retrieval of Handwritten Chinese Character Images for Handwriting Teaching," *Scientiarum Naturalium Universitatis Pekinensis*, vol. 51, Mar. 2015, pp. 241-248.
- [9] W. An and C. Li, "Automatic matching of character strokes for computer-aided Chinese handwriting education," *Proc. International Conference on E-Education, Entertainment and E-Management*, 2011, pp. 283-288.
- [10] X. W. Zhang, W. H. An and Y. G. Fu, "Adaptive Correction of Errors from Segmented Digital Ink Texts in Chinese Based on Context," *Proc. International Conference on Information Technology & Computer Science*, 2010, pp. 25-35.