





Multi-touch Gesture Recognition of Braille Input Based on RBF Net

Zhang Juxiao^{1,2} , Zeng Xiaoqin², and Zhu Zhaosong¹  

¹ College of Information and Mathematics Science, Nanjing Normal University of Special Education, Nanjing 210038, Jiangsu, China
3301611@qq.com, zzs2019@foxmail.com

² College of Computer and Information, Hohai University, Nanjing 210038, Jiangsu, China

Abstract. One challenging task for the blind is to input Braille while by no way could they sense the location information on touch screens. The existing Braille input methods are suffering from problems including inaccurate positioning and lack of interactive prompts. In this paper, touch gestures are recognized by trained RBF network while combined gestures are modelled. By doing so, the Braille input concerning multi-touch gesture recognition is then implemented. The experimental results show that the method is effective and blind people can friendly input Braille with almost real-time interaction.

Keywords: Input method of braille · RBF net · Multiple touch gesture recognition

1 Introduction

In recent years, although many improvements have been made in the availability of smartphones and other electronic devices for the visually impaired, there is still a huge gap between blind people and touch-screen devices in human-computer interaction.

1.1 Research Status

At present, the output stage mostly uses Text to Speech (TTS) technology or external point input devices [1], the main input methods of touch-screen mobile phones include Pinyin [2], stroke [3], speech recognition [4] and on-line handwriting recognition input methods [5]. E.g: Multi-touch input method for the blind [1], Single-Finger touch gesture input method [6], Touch action recognition input method [7].

Other common Braille input methods on touch-screen include Perkinput [8], TypeInBraille [9] and BrailleTouch [10], etc.

1.2 Deficiencies

Blind touch screen input method has two major shortcomings: one is precise localization; the other is an interactive prompt barrier. The biggest difficulty for blind people to use the touch screen is that they cannot get the location information on the screen,

and they cannot obtain a series of operations such as visual focus [11]. The keys of the virtual keyboard used in the input method are too close to each other, which also lead to an increase in the rate of input error [12]. Interactive prompt accessibility barrier means that the time and focus of fingers sliding on the touch screen cannot be fed back to blind persons.

1.3 Innovations

A Braille Input concerning Multi-Touch Gesture Recognition based on RBF Net is designed. This input method integrates Braille phonetic notation, which makes it possible to complete Braille input and output through voice input and Text to Speech (TTS) conversion. Besides, when the Braille gesture is inputting on the touch screen, it reduces the interference of orientation. It relates the high inherent logicity of Braille, which makes it easy for blind people to learn and remember. It is suitable for blind people to input Braille information on touch-screen devices, providing a new method for the human-computer interaction between blind people and touch screen.

2 Multi-touch Gesture Recognition Braille Input Module

The main algorithm modules include: multi-touch area loaded on electronic devices and the multi-touch area graphic information library module generated by this multi-touch area, multi-touch area graphic information library recognition module and Braille input module and so on.

2.1 Multi-touch Area Graphic Information Library Module

The basic graphics feature of multi-touch area is established based on the four-point position features at the top of Braille and the graphic information of multi-touch area. The graphic information library of multi-touch area is constructed by adding one or two-finger touches to the basic graphics feature, and the correspondence table between graphic information of the multi-touch area and the Braille points is generated.

2.2 Establish the Correspondence Table Between Graphic Information of the Multi-touch Area and Braille

Based on the inherent logic of Braille points, the Braille point positions composed of points 1, 2, 4 and 5 are established, with points 3 and 6 points kept empty. The graphics shapes of these four points are taken into account when establishing the Braille points, and the multi-touch area graphic information library is established based on the regional information composed by the above four points.

In order to reduce the interference of noisy graphical information caused by touch and improve the degree of distinction between graphical information in the multi-touch area, finger movement in touch area should follow the regulated order principle.

Rule 1: The principle of sequence uniqueness.

$$f(\chi_i, \chi_j) \neq f(\chi_j, \chi_i), i \neq j \wedge \chi_i, \chi_j \in X \quad (1)$$

Where X is a finite state set, $X = \{\chi_i, \chi_i \in C\}$, $C = \{1, 2, 4, 5\}$;

f is Transfer Function, $f(\chi_i, \chi_j)$ is sliding from point area χ_i to point area χ_j , $\{\chi_i, \chi_j \in X\}$.

Rule 2: The sequence of touch sliding is from top to bottom and from left to right.

Where X is a finite state set,

$$X = \{\chi_i, \chi_i \in C\}, C = \{1, 2, 4, 5\} \quad (2)$$

f is Transfer Function, $f(\chi_i, \chi_j)$ is sliding from point area χ_i to point area χ_j , $\{\chi_i, \chi_j \in X\}$;

$f(\chi_i, 0)$ denotes hitting point area χ_i , Then

$$f(\chi_i, \chi_i) = f(\chi_i, 0) = f(\chi_i) \quad (3)$$

Continuous sliding:

$$f(\chi_i, \chi_k) = f(\chi_i, \chi_j) + f(\chi_j, \chi_k) \quad (4)$$

$$\text{iff } i < j < k \wedge \chi_i, \chi_j, \chi_k \in X \quad (5)$$

$$f(\chi_i) < f(\chi_j) < f(\chi_k) \Rightarrow f(\chi_i, \chi_k) = f((\chi_i, \chi_j), \chi_k) \quad (6)$$

$$f(\chi_i) < f(\chi_j), f(\chi_i) \langle f(\chi_k) \wedge f(\chi_j) \rangle f(\chi_k) \Rightarrow f(\chi_i, \chi_k) = f(\chi_i, \chi_j) \quad (7)$$

$$f(\chi_i) > f(\chi_j) \Rightarrow f(\chi_i, \chi_k) = f(\chi_i, 0) = f(\chi_i) \quad (8)$$

2.3 Multi-touch Area Graphic Information Recognition Module

When touching the screen with both left and right hands, pressing and holding or sliding over the multi-touch area to form graphical information, the module collects the input points and area to form a graphical structure, thus draws the features of multi-touch area graphical information. Moreover, it compares the graphical information features with those in the multi-touch graphics area information library and looks up the Braille comparison table to extract the Braille.

3 Multi-touch Gesture Recognition Algorithm

3.1 Multi-touch Gesture Action Data Structure

A touch gesture action is a description of a single point touch, which can be described by a quadruple data structure [13]:

$$\text{Metagesture} = \{ID, \text{Touchstate}, \text{Coordinate}, \text{Time}\} \quad (9)$$

Where: $ID = \{1, 2, 3, 4, 5\}$ respectively represent the touchpoint identification numbers of different fingers in one hand. Each touchpoint at a certain moment has a unique ID number, so as to record the trajectory and direction of the same touchpoint.

$\text{Touchstate} = \{0, 1\}$ represents the direct touch state between the finger and the touch screen, $\text{Touchstate} = 1$ represents touch state, and $\text{Touchstate} = 0$ represents the not-touch state.

$\text{Coordinate} = P(x, y)$ represents the coordinates of the touchpoint. When $\text{Touchstate} = 1$, the touch screen records a series of coordinate data by IDs according to touchpoint detection and touchpoint tracking. When $\text{Touchstate} = 0$, it will be considered as noise data and the coordinate information will be meaningless.

Time is the time data when gestures are recorded, which is acquired by reading the system time. It represents the time parameter when the state or position of the touchpoint changes.

Multi-touch gestures consist of touch gesture element actions, which can be described by a quintuple data structure [14]:

$$\text{Multigesture} = \{\text{Number}, ID, \text{Touchstate}, \text{Coordinate}, \text{Time}\} \quad (10)$$

Where: $\text{Number} = \{1, 2, \dots\}$ represents the number of touchpoints tracked by touchpoint detection, usually, $(0 \leq \text{Number} \leq 5)$. The other parameters have the same meanings as Metagesture.

3.2 Data Preprocessing

Touch gestures on the touch-screen are recorded in the form of data, but it is impossible that the time and area scope of all actions are consistent, besides, there will be noise data, which need to be preprocessed. Preprocessing includes noise elimination, absorption of gesture graphics deformation, compression of redundant information and so on.

3.3 Normalization Processing

Network training cannot be carried out if the data volume of different touch gestures is inconsistent. Therefore, the data of touch gesture actions need to be normalized. Normalization is to take out the same number of points for all touch gestures according to different time differences. After repeated experiments, 15 sampling points are

selected from A to B and are divided equally according to time. The obtained information of the 15 sampled points are stored as 14 two-dimensional vectors and are normalized to obtain the normalized feature vectors of multi-touch gestures, which are used as the input into the RBF network.

3.4 Input Vector of Network

The multi-touch gesture action is composed of several gesture element actions arranged and combined in chronological order. The feature matrix of the touch gesture of m touchpoints is as follows:

$$\begin{bmatrix} ID_1 & Touchstate_1 & (x_{10}, y_{10}) & \cdots & (x_{1n}, y_{1n}) & t_1 \\ ID_2 & Touchstate_2 & (x_{20}, y_{20}) & \cdots & (x_{2n}, y_{2n}) & t_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ ID_m & Touchstate_m & (x_{m0}, y_{m0}) & \cdots & (x_{mn}, y_{mn}) & t_m \end{bmatrix} \quad (11)$$

3.5 Feature Extraction

The collected multi-touch gesture information is classified and merged based on their features so as to extract the graphics with corresponding feature information. We first determined the number of touchpoints m ; then we found the matching gestures in the database through the gesture features. Then we distinguished the actions of double-click and clicking twice as well as long-press and sliding actions by combining with the given threshold function. Sliding Displacement Threshold Expression:

$$f(x, y) = \begin{cases} 0, \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \geq a \\ 1, \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} < a \end{cases} \quad (12)$$

Time Threshold Expression:

$$f(t) = \begin{cases} 0, t \geq a \\ 1, t < a \end{cases} \quad (13)$$

3.6 RBF Network Design and Training

If the Gaussian function is used as the radial basis function (RBF), the expression of the radial basis function of the i^{th} hidden layer is as follows:

$$\psi_i(x) = \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right), i = 1, 2, \dots, n \quad (14)$$

The Gaussian function is adopted as the radial basis function:

$$\psi_i(t) = \exp\left(-\frac{t^2}{2\sigma_i^2}\right) \text{ where } t = \|x - c_i\|, \text{ then} \quad (15)$$

$$\psi_i\left(\|x - c_i\|^2\right) = \exp\left(-\frac{(\|x - c_i\|)^2}{2\sigma_i^2}\right) \quad (16)$$

The width of the RBF network is determined by σ . Let d be the maximum distance of the samples and let N be the number of samples, then the width is determined by fixation method.

$$\sigma = \frac{d_{max}}{\sqrt{2N}} \quad (17)$$

3.7 Recognition of Touch Gesture Element Actions

Following the previous data preprocessing steps, the module sends the input vectors of touch gestures into the trained RBF Net for classification. If the output vector of RBF is closest to the standard output of a gesture in the gesture library, it is identified as the desired gesture [15].

4 Results and Discussion

Eight students were organized to form a user test group. The average time of multi-touch gesture recognition entering Braille is respectively s1–s8: 2.49 s/square, 3.21 s/square, 2.68 s/square, 2.88 s/square, 3.75 s/square, 3.38 s/square, 3.57 s/square, 3.99 s/square.

The Braille input method of multi-touch gesture recognition features the following advantages:

- (1) Availability: the function of inputting Braille on the electronic device of touch-screen is realized. The issue of text information interaction between blind persons and touch-screen is addressed.
- (2) Simplicity: points 1, 2, 4, 5 in Braille points constitute a graph which is highly similar to Braille points. There are fewer basic graphics, and the inherent logic in Braille points distribution is retained in the expanded graphics, therefore it facilitates blind persons to learn and remember.
- (3) Efficiency: multiple touches are operated and one block of Braille is input in one operation. Braille Points 3 and 6 are totally dependent on increasing finger touches, which is highly similar to the internal logic structure of Braille. This reduces the time for transformation and thinking when inputting, thus improving the efficiency of inputting Braille.

Acknowledgement. This work was supported by The Major Programs of Natural Science Foundation of the Jiangsu Higher Education Institutions of China (No. 19KJA310002.) and The Natural Science Foundation of the Jiangsu Higher Education Institutions of China (No. 17KJD520006).

References

1. Zhang, J.X., Zeng, X.Q., Meng, C.H.H.: Design and implementation of multi-touch input method for blind usage. *Comput. Appl. Software* **10**, 231–235 (2015)
2. Liu, F., Wang, Y.H., Tang, B.Z.H., et al.: Intelligent Chinese input method based on android. *Comput. Eng.* **37**(07), 225–227 (2011)
3. Yu, T.Z.H.: *The Design and Implementation of Cross-Platform Stroke Input Method Engine*. Harbin Institute of Technology (2013)
4. Zheng, Y.D., Chen, Z.H.S., Xiao, L.S.H.: Research and implementation of a voice control audio system based on Android speech recognition. *Mod. Electron. Tech.* **42**, 93–96 (2019)
5. Yan, X.L., Wang, L.M.: Handwritten Chinese character recognition system based on neural network convolution depth. *Comput. Eng. Appl.* **53**(10), 246–250 (2017)
6. Chen, H.F., Xu, S.H., Wang, J.L.: *A Braille Input Method Based on Gesture Recognition: JiangSu*, CN102929394A, 13 February 2013
7. Hu, Y.P.: *A Method and Device for Output and Input of Braille Characters on Touch Screen*. BeiJing: CN103870008A, 18 June 2014
8. Fukatsu, Y., Shizuki, B., Tanaka, J.: No-look flick: single-handed and eyes-free japanese text input system on touch screens of mobile devices. *human computer interaction with mobile devices and services*, pp. 161–170 (2013)
9. Mascetti, S., Bernareggi, C., Belotti, M., et al.: TypeInBraille: a braille-based typing application for touchscreen devices. In: *Conference on Computers and Accessibility (ASSETS 2011)*, pp. 295–296 (2011)
10. Frey, B., Southern, C., Romero, M., et al.: Brailletouch: mobile texting for the visually impaired. In: *International Conference on Universal Access in Human Computer Interaction*, pp. 19–25 (2011)
11. Fukatsu, Y., Shizuki, B., Tanaka, J., et al.: No-look flick: single-handed and eyes-free japanese text input system on touch screens of mobile devices. In: *Human Computer Interaction with Mobile Devices and Services*, pp. 161–170 (2013)
12. Nicolau, H., Guerreiro, T.J., Jorge, J.A., et al.: Proficient blind users and mobile text-entry. In: *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics*. New York USA, pp. 19–22 (2010)
13. Ji, H.Y.: *Human-Computer Interaction Research Based on Multi-touch Technology*. Diss. East China Normal University (2011)
14. Wang, X.Q., Chen, G., Wang, D., Wang, C.H.: Research on multi-touch gesture analysis and recognition algorithm. *Comput. Sci.* **39**(S1), 522–525 (2012)
15. Li, W.S.H., Deng, C.H.J., Lv, Y.: Interaction gesture analysis based on touch screen. *Chin. J. Liquid Crystals Displ.* **26**(2), 194–199 (2011)