International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 4, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Myanmar Language Sketch Soft Keyboard on Android

Nandar Pwint Oo
University of Computer Studies, Yangon
Email ID : nandarpwintoo@gmail.com

Ni Lar Thein
University of Computer Studies, Yangon
Email ID : nilarthein@gmail.com

***Abstract* — This paper describes a gesture stroke recognition technique for sketch touch screen input method on Android. With the expansion of touch screen based computing in today mobile devices such as mobile phones and tablets, new user interface prototypes may incorporate gesture stroke input method on mobile devices. There are many gesture input methods for Latin, Chinese and Japanese language. According to the text entry methods in the Human Computer Interaction (HCI) literature, Myanmar language gesture input method is still under developed research area on today Android smart phone. Myanmar language is not the same as Latin language. It is a complex text language with so many multi-stroke sketches. Thus, there are many issues in the consideration of gesture stroke input method for Myanmar language. This aim of this paper is to propose a new sketch soft keyboard on Android platform with the combine used of multi-stroke handwritten recognizer and rule based disambiguation method in gesture stroke recognition. In addition, the text entry speed and error rate experiments of the proposed gesture stroke recognition technique with native users are also analyzed.**

***Keywords* — IME, HCI, Android, Gesture.**

I. Introduction

Researchers have been actively investigated many techniques and mechanism to obtain more efficient text entry on touch screen devices [1-3, 5 and 9]. There are handwriting recognition, keyboard layout optimization and word or syllable prediction, etc. The current trends of researches are to investigate more natural interactions based on gestures. Moreover, a new gesture can be added to the library in an on-line manner without expensive retraining, and it can immediately contribute to future recognition [9]. One promising example is commercial Swype keyboard. Swype keyboard has the ability to degrade to a standard QWERTY keyboard but can also offer a fast gesture based word typing by taking the leverage of dictionary and language model based algorithm.

Handwritten recognition without character recognition produces “digital ink”. This is fine for some applications such as annotation, visual art and graphic design. However, digital ink requires more memory and in general it is not well managed by computing technology. Specifically, digital ink is difficult to index and search. In Android, for handwritten text entry, it uses gesture API and some recognition mechanism to recognize the handwritten stroke of the user. Gesture is interpreted based on three sources such as a library of learned gestures from many users, online gesture stroke from the mobile phone user and built-in handwriting recognizer.

However, in Myanmar language some alphabets lead to ambiguity for recognizer because they have got the same gesture stroke. For example, Myanmar consonant (Da Oke Chike –) is the same glyph with Myanmar upper vowel (Lone Gyi Tin - -) in gesture path and stroke. In this case, there is an ambiguity for the built-in recognizer for prediction the correct alphabet. To overcome this problem, this paper proposed a solution for the problem of recognizer ambiguity by using rule based disambiguation mechanism according to the user’s previously sketched alphabet.

The rest of the paper is organized as follows. Firstly, Section 2 reports the previous work on handwritten input method for Latin alphabet. In Section 3, the motivation points and background of Myanmar language is presented. Section 4 is the place of detail system implementation. Subsequently, Section 5 and 6 showed experiments performed in order to show the practicability of the proposed approach. Finally, Section 7 draws some conclusions and future work of the proposed soft keyboard is in Section 8.

II. Related Work

Many approaches and mechanisms to gesture recognition were presented in this section, including single stroke $1 recognition [7], multi-stroke $N recognition [4], recognition with motion vector [6] and handwritten recognition with Hidden Markov Models (HMMs) [8].

Jacob O. Wobbrock and a group proposed $1 recognizer that is easy, cheap and usable almost anywhere in about 100 lines of code. Their proposed recognizer recognized the gesture by using simple four algorithm steps such as re-sampling the point path on the inductive angle, scaling, translating and finding the optimal angle for the best score. Their approach is also robust in candidate gesture position variant. But, $1 recognizer does not use time, so gestures cannot be differentiated on the basic of speed.

The approach of $1 recognizer is efficient for single stroke, but many gestures comprise multiple strokes. To enable easy incorporation of multistrokes recognition Lisa Anthony, Jacob O and Wobbrock proposed an alternative method of $N (extension of $1 recognizer) again based on simple geometry and trigonometry. Although $N recognizer shared with $1, $N is much more versatile by recognizing gestures comprising multiple strokes, automatically generalizing from one multistrokes template to all possible multistrokes with alternative stroke orderings and directions, recognizing 1D gestures such as lines and providing bounded rotation invariance.

Meanwhile, Chebanyuk O.V. proposed handwritten recognition based on Analysis of motion vectors. The process of recognition consists of analyzing the user input, removing redundant information, smoothing of the input image, analyzing of motion vectors, generation of character schema and searching of the character in knowledgebase. Their proposed method of motion vector

International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 4, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

is only suitable for Latin alphabet because they tried to distinguish 8 possible directions of motions and point motion such as up, up-right, right, down-right, down, down-left, left, up-left and point. Their proposed mechanism cannot suitable for Myanmar script because in Myanmar script there are many rounded handwritten nature.

In addition, Zhengxing Sun and followers figured out an approach for adaptive online multi-stroke sketch recognition based on Hidden Markov Model (HMM). In their proposed approach, the drawing sketch was viewed as the result of a stochastic process that is governed by a hidden stochastic model and identified according to its probability of generating the output.

Although many techniques have been proposed to improve text input on touch screens, the majority of this research ignores handwritten input for Myanmar language. In the proposed system, recognition is done by examining the closest minimum average Squared Euclidean Distance (SED), stroke count, stroke length distance and rule based disambiguation for ambiguous alphabet.

III. Background

Myanmar language is syllabic languages that derived from Brahmi script and have got common writing natures with South Indian languages. Writing system is not the same as Latin script and writing nature is composed of a adding vowel and medials to the consonant. Moreover, the number of alphabet is more than Latin alphabet and the shape of glyph is changed according to the logical structure. In Myanmar alphabet there are 33 consonants, 12 vowels, 4 medials and 10 digits. Writing order is left to right and space is not used between words or syllables except after some words and clause.

Fig.1. Myanmar Consonants

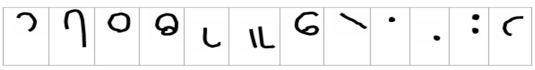

Fig.2. Myanmar Vowels

Fig.3. Myanmar Meidals

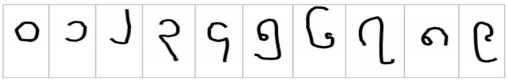

Fig.4. Myanmar Digits

Most of the handwritten nature of Myanmar alphabet is multi-stroke. Android's built-in gesture recognizer can only recognize successfully for single stroke. There is a problem for multi stroke Myanmar language handwritten recognizer. However, gesture library of an Android API can store several trained gestures for each Myanmar alphabet. In each gesture, there are many gesture strokes. For each stroke it can represent with gesture point. So, the architecture of the gesture library can be seen according to the following figure.

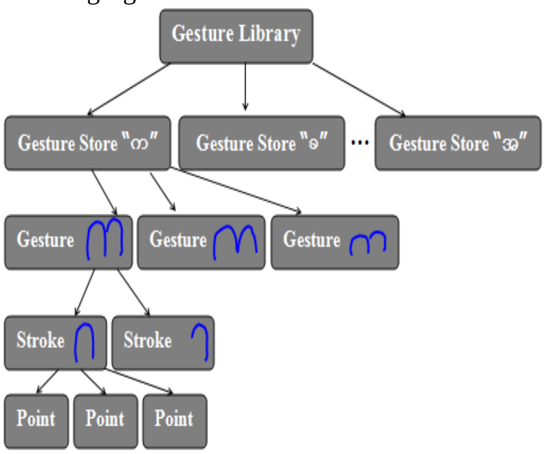

Fig.5. The structure of Android gesture library

IV. Implementation

Touch-screen handwriting gestures have many special properties such as figure-based input, limited drawing area, limited computational resources to supports natural multi-stroke letter as well as mixed cursive handwritten. Thus, the architecture of the proposed system is considered to form quick recognizer as depicted in figure 6. The gesture from the gesture overlay is sampled to 2D gesture float vector. The recognizer compares on unknown gesture stroke U it a set of stored gesture template T. The closest to U is the recognition result, determined by the average minimum SED, minimum stroke length distance and stroke count between corresponding points in U and each gesture template T_i.

International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 4, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

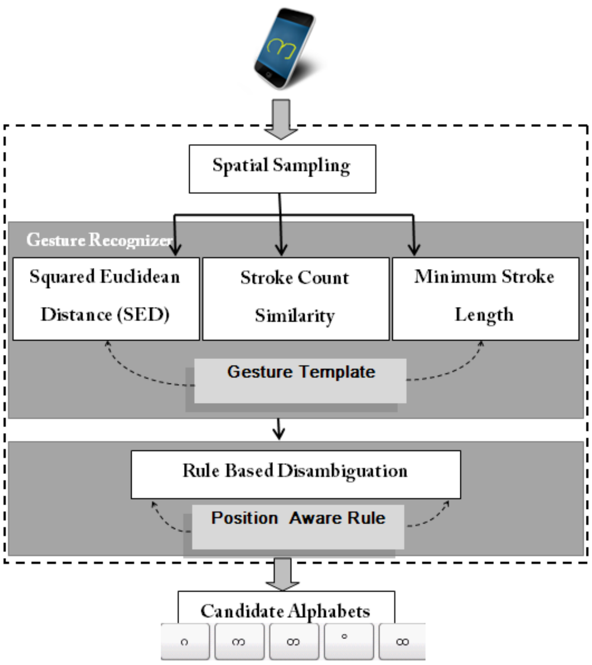

Fig.6. Architecture of the recognizer

A. Spatial Sampling

Spatial sampling is the process of sampling the gesture to form a 2D array of pixels in rows and columns. Each sample point contains information on the variable of interest at that spatial location. After spatial sampling, 2D float vector is used in calculation of minimum SED and minimum stroke length.

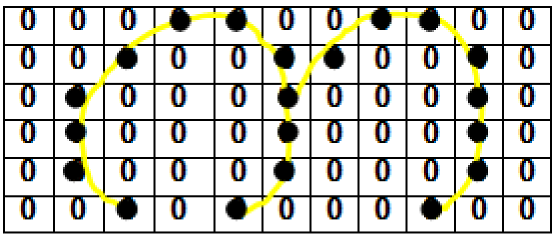

Fig.7. Spatial Sampling

B. Squared Euclidean Distance

In the proposed recognizer, to find the closest of unknown gesture *U* from the set of template *Ti*, Square Euclidean Distance between two float vectors *U* = [*u1 u2*] and *T* = [*t1 t2*] is calculated as depicted in equation 1.

$$SED = \sqrt{\sum_{i=0}^{n} (U_i - T_i)^2} \qquad (1)$$

As mention above, in each gesture store there are many gestures for predefined template gesture for particular alphabet. Thus, it is needed to find the Average Minimum Distance (AMD) again, over predefined gesture template for particular alphabet as shown in equation 2.

$$AMD = \min\left[\frac{\sum_{g=1}^{n} SED}{n}\right] \qquad (2)$$

C. Minimum Distance of Stroke Length

Moreover, the gesture recognizer also needs to consider the Minimum Distance of Stroke Length (MDSL) and gestures Stroke Count Similarity.

$$MDSL = \min[|U\ length| - |T_i\ length|] \qquad (3)$$

According to the experiments, in Myanmar alphabet there are many alphabets that have the same gesture although they are different glyph to the position of handwritten. The same gesture of Myanmar alphabet as depicted in Table1 leads to ambiguity to the recognizer. Rule based disambiguation can lessen the recognizer ambiguity. It takes the previous user typed alphabet from the input connection and makes recognition decisions. Moreover, the experiment show that Anusvara [-□], Lower Dot [-□] and Visarga [-□] are difficult to recognized with the proposed method. To ease of usability, the most difficult recognized alphabet need to use with traditional key button input via some control key.

Table I. Ambiguous Alphabet

<table>
<tr><th>Ambiguity</th><th>Myanmar Alphabet</th></tr>
<tr><td>Ambiguity 1</td><td>- | -□ | |</td></tr>
<tr><td>Ambiguity 2</td><td>-□ | □</td></tr>
<tr><td>Ambiguity 3</td><td>□ | □ | -□ | □</td></tr>
<tr><td>Ambiguity 4</td><td>-□ | □</td></tr>
<tr><td>Ambiguity 5</td><td>-□ | -□ | -□</td></tr>
<tr><td>Ambiguity 6</td><td>-□ | □</td></tr>
<tr><td>Ambiguity 7</td><td>-□ | □ | -□</td></tr>
</table>

After recognized with the recognizer, the system check the candidate alphabet contains ambiguous alphabet and prioritized the candidate alphabet according to the following rule fragment.

Table II. Example of Rule Fragment

No	Rule
Rule 1	*Consonant* → *Upper Vowel*
Rule2	*Consonant* → *Down Vowel*
Rule3	*E-vowel* → *Consonant*
Rule 4	*Anusvara* → *Lower Dot*
Rule 5	*Consonant* → *Right Medial*

V. PARTICIPATION FOR EVALUATIONS

Twenty eight university students participated in handwritten gesture recognition. Among the students, nine students had not experimented on touch screen device earlier. Gestures were collected on Android mobile phone running modified gesture builder that recorded student's gesture strokes. Students supplied gesture for 72 alphabets (most frequent used Myanmar alphabet) 3 times as part of their gesture template building. The final gesture template containing 6048 gestures, 62% were unistrokes and 48% were multistrokes.

Fig.8. Myanmar Handwritten Gesture Collector

VI. Experimental Results

The recognition results are evaluated according to the Gesture per Minutes (GPM) and Error rate (ER) on six native users is shown in the figure 10 and figure 11.

TC=Total Correct Character
T= Total Numbers of Time
CG= Time to input Correct Gesture
IG = Time to input Incorrect Gesture
F= Key Stroke to fix Gesture Errors
E= Total Number of Errors

Fig.9. Myanmar Handwritten Soft Keyboard

Before making evaluation, all of the volunteers were given 5 minutes demonstration time and 10 minutes practice time. The candidate users were asked to type all of the Myanmar alphabets and their preferred random short sentences. The average time taken and gesture inputting performance are collected separately and measured the performance of the proposed handwritten soft keyboard. However, the inputting performance of handwritten soft keyboard is under desirable as compare with traditional soft keyboard with key button.

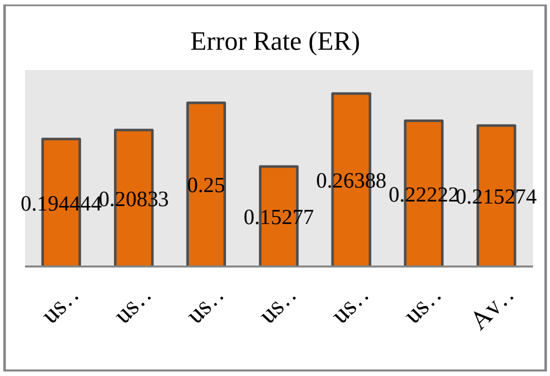

Fig.10. ER of Myanmar Handwritten Soft Keyboard

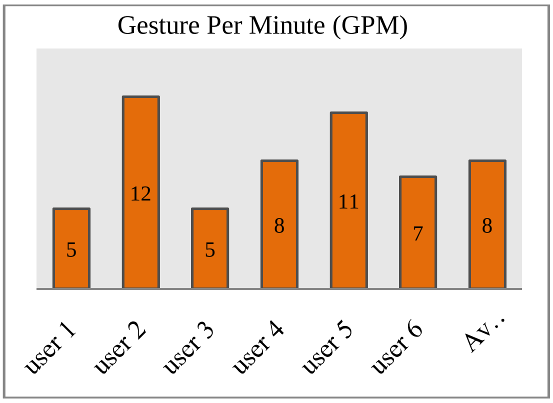

Fig.11. GPM of Myanmar Handwritten Soft Keyboard

VII. Conclusion

The recognizer that used in the proposed gesture stroke input method editor requires no complex mathematical procedures and completes with approaches that use minimum distance calculation and rule based disambiguation techniques. According to the experiments the proposed Myanmar sketch soft keyboard reaching 97.0% accuracy rate and eight gestures can input correctly in one minute.

VIII. Future Work

The proposed soft keyboard can extend to add new gesture to the gesture library on-line manner without expensive retraining so that the system can continually learn from the user's gestures over time and can adapt to the particular drawing or handwriting gesture style of the particular user.

Acknowledgment

I would like to thank all of my friends from Ph.D. 7th batch for their handwritten gestures in gesture library building and participation in performance evaluation procedure.

International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 4, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

REFERENCES

[1] C. Chris, F. S. Thomas, K. Tolga and B. K. Levent, "Recognizing Multi-Stroke Symbols", Sketch Understanding, AAAI Spring Symbosium 2002.

[2] F. Leah, Q. L. Ben and O. W. Jacob, "Beyound QWERTY: Augmenting Tocuh-Screen Keyboards with Multi-Touch Gestures for Non-Alphanumeric Input", HCI 2012.

[3] L. Yang, "Protractor: A Fast and Accurate Gesture Recognizer", Google Research.

[4] L. Anthony, J.O. Wobbrock, "A Lightweight Multistroke Recognizer for User Interface Prototypes", In Graphics Interface 2010, 245-252.

[5] L. Marcus, W. Markus, Z. Tobias and D. Andreas, "Seamless Integration of Handwriting Recognition into Pen-Enabled Displays for Fast User Interaction", 10th IAPR International Workshop on Document Analysis Systems, 2012.

[6] O. V. Chebanyuk, "Handwritten Text Recognition Based on Analysis of Motion Vectors", National Aviation University.

[7] O. W. Jacob, D. W. Andrew, L. Yang, "Gesture without Libraries Toolkits or Training A $1 Recognizer for User Interface Prototypes"

[8] S. Zhengxin, J. Wei and S. Jianyong, "Adaptive Online Multi-stroke Sketch Recognition Based on Hidden Markov Model", Lecture Notes in Computer Science, 2006, Volume 3930/2006, 948-957.

[9] Y. O. Tom and L. Yang, "Bootstrapping Personal Gesture Shortcuts with the Wisdom of the Crowd and Handwritten Recognition" CHI'12, May 5-10, Austin Texas, USA, ACM 2012.

AUTHOR'S PROFILE

Nandar Pwint Oo is a Ph.D. research student. She received B.C.Sc. degree in 2004, B.C.Sc. (Hons.) degree in 2005 and M.C.Sc degree in 2008. She is also one of the Tutors from University of Computer Studies, Yangon and have got over 6 years of experiences in teaching and software development in mobile platform. Her research interested areas are Human Computer Interaction (HCI), Mobile computing and Cloud Computing. She have been involved in developing Android application and sharing knowledge with the online community, developer conference Yangon and Barcamp Yangon.

Ni Lar Thein is a Rector of University of Computer Studies, Yangon (UCSY). She obtained M.Sc. (Computer of Science) from Yangon University and Ph.D. (Computer Engineering) from Nanyang Technological University, Singapore in 2003. Her research interests include Software Engineering, Artificial Intelligence and Natural Language Processing. She has published papers in International Conferences and International Journals.