International Journal of Computer Applications Technology and Research
Volume 7–Issue 07, 281-285, 2018, ISSN:-2319–8656

Hangul Recognition Using Support Vector Machine

Rahmatina Hidayati
Department of Electrical Engineering
University of Brawijaya
Malang, East Java, Indonesia

Moechammad Sarosa
Department of Electrical Engineering
State Polytechnic of Malang
Malang, East Java, Indonesia

Panca Mudjirahardjo
Department of Electrical Engineering
University of Brawijaya
Malang, East Java, Indonesia

Abstract: The recognition of Hangul Image is more difficult compared with that of Latin. It could be recognized from the structural arrangement. Hangul is arranged from two dimensions while Latin is only from the left to the right. The current research creates a system to convert Hangul image into Latin text in order to use it as a learning material on reading Hangul. In general, image recognition system is divided into three steps. The first step is preprocessing, which includes binarization, segmentation through connected component-labeling method, and thinning with Zhang Suen to decrease some pattern information. The second is receiving the feature from every single image, whose identification process is done through chain code method. The third is recognizing the process using Support Vector Machine (SVM) with some kernels. It works through letter image and Hangul word recognition. It consists of 34 letters, each of which has 15 different patterns. The whole patterns are 510, divided into 3 data scenarios. The highest result achieved is 94,7% using SVM kernel polynomial and radial basis function. The level of recognition result is influenced by many trained data. Whilst the recognition process of Hangul word applies to the type 2 Hangul word with 6 different patterns. The difference of these patterns appears from the change of the font type. The chosen fonts for data training are such as Batang, Dotum, Gaeul, Gulim, Malgun Gothic. Arial Unicode MS is used to test the data. The lowest accuracy is achieved through the use of SVM kernel radial basis function, which is 69%. The same result, 72 %, is given by the SVM kernel linear and polynomial.

Keywords: Support Vector Machine; SVM; Kernel Polynomial; Kernel Linear; Kernel Radial Basis Function; Hangul

1. INTRODUCTION

Optical Character Recognition (OCR) is a character introduction system with images input. It contains texts that would be converted to the edited versions[1]. The work of OCR system depends on the kind of processed text. Generally, the text is divided into three categories. They are written, printed, and typed text[2].

Some researches on OCR System have been conducted. One of methods ever used is Support Vector Machine (SVM). A kind of character which had ever been searched by using SVM is Hindi number, which is known as Numeral Kanada, upper case and lower case alphabet A-Z [2,3,4]. The SVM method is used with different data, Korean characters known as Hangul.

Hangul recognition is more difficult compared with Latin due to its complicated arrangement. Hangul is arranged from 2 dimensions (both left and right side), while Latin is arranged from left to the right [5].

A Research on Hangul recognition has ever been conducted, where the writer applies the Stochastic Relationship Modeling to the recognition process of Hangul syllable writing. The output of the research is Hangul syllable texts[6].

So far, the research on Hangul recognition is conducted with Hangul texts output. The current research will improve the image conversion of Hangul with Latin text output. The image of Hangul converted into Latin text can be used as a learning material on how to read Hangul.

OCR system, in general, is divided into three steps. They are preprocessing, feature extraction, and recognition. Pre-process includes three stages: binarization to change the grayscale image into black white; segmentation, which is processed through connected component labeling, to separate the input into individual word; and thinning to decrease some information pattern (thin Line) in order to be easily analyzed [7]. The research will employ algorithm Zhang Suen, which works faster than the other thinning algorithms[8].

The next step, after pre-process, is feature extraction. It has an important role on recognition process. It works through generating the basic component of the image called features[9]. The feature extraction used in the current research is chain code. The last process is recognition, using the SVM method with some kernels *(linear, polynomial, and radial basis function).*

2. METHODOLOGY

Generally, OCR system is divided into three main steps. They are preprocessing (binarization, segmentation, and thinning), feature extraction in which in this research uses chain code, and recognition by applying Support Vector Machine (SVM) method. Figure 1 shows the general process of Hangul recognition.

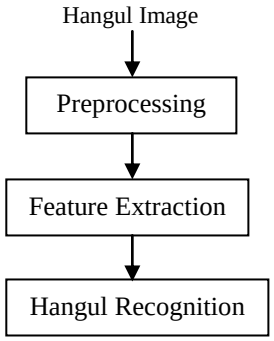

Figure 1. Block diagram of Hangul recognition

The input image used in the current research is the letter and Hangul word. The letter consists of 21 vowels and 13 consonants, shown in Figure 2 with Latin letter. Each letter has 15 different forms. The whole data are amounted to 510.

Figure 2. The Letters of Hangul and Latin[10]

Meanwhile for word, there are 6 ways on how to arrange the letter of Hangul into word. The first type is shown in Figure 3[10]. The discussion focuses on data type 2.

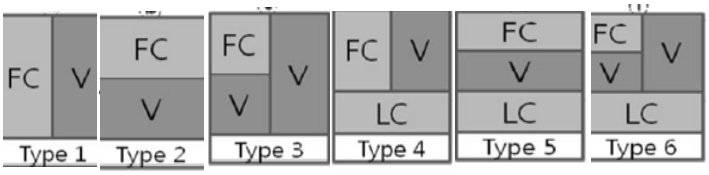

Figure 3. 6 the ways to arrange the letter of Hangul[10]

Meanwhile, the example of type 2 Hangul word is shown in Figure 4. Each word consists of 6 different forms. They are achieved by changing the word font. The used fonts are Arial, Batang, Dotum, Gaeul, Gulim, and Malgun Gothic.

	Training Data				Testing Data	Latin
보	보	보	보	보	보	Bo
뵤	뵤	뵤	뵤	뵤	뵤	BYO
부	부	부	부	부	부	BU
뷰	뷰	뷰	뷰	뷰	뷰	BYU
브	브	브	브	브	브	BEU
소	소	소	소	소	소	SO
쇼	쇼	쇼	쇼	쇼	쇼	SYO
수	수	수	수	수	수	SU
슈	슈	슈	슈	슈	슈	SYU
스	스	스	스	스	스	SEU

Figure 4. The example of Hangul word

2.1 Preprocessing

The preprocessing includes 3 steps. First, the binarization or thresholding, is implemented to change the grayscale image become black white. The process of thresholding will produce the binary image, the image which has two steps of grayish (black and white). Generally, the process of floating the grayscale image to produce the biner image are as follows[11]:

$$g(x,y) = \begin{cases} 1 \; if \; f(x,y) \geq T \\ 0 \; if \; f(x,y) < T \end{cases} \quad (1)$$

With g (x,y) is the binary image from grayscale image f (x,y) and T assert the percentage of threshold.

Second, thinning is used to decrease some information to a pattern becomes thin line in order to be easy to analyzed[7]. In this case, it will be applied the Zhang Suen algorithm which has faster performance compare with the other thinning algorithm[8]. To process thinning algorithm is shown in the Figure 6. This algorithm uses the pixel 3x3 and 8 degrees as in the Figure 5. P1 is a pixel that will be checked, if it fulfills the fixed condition, so the pixel will be deleted. The conditions are as follows[12]:

(a) $2 \leq B(P_1) \leq 6$ (2)

(b) $A(P_1) = 1$ (3)

(c) $P_2 \times P_4 \times P_6 = 0$, and (4)

(d) $P_4 \times P_6 \times P_8 = 0$ (5)

(e) $P_2 \times P_4 \times P_8 = 0$, and (6)

(f) $P_2 \times P_6 \times P_8 = 0$ (7)

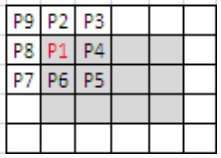

Figure 5. The pixel 3x3 with 8 degrees

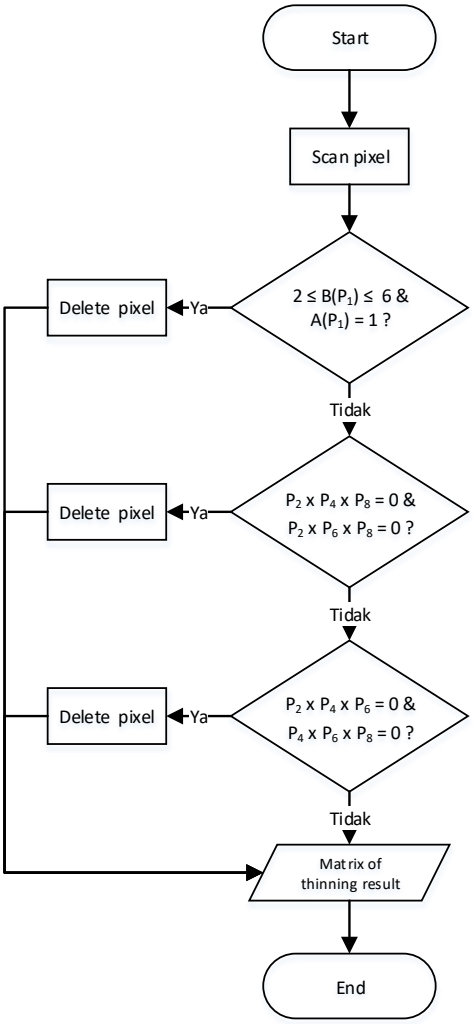

Figure 6. The Diagram algorithm of Zhang Suen

2.2 Feature Extraction

The next step after preprocessing is feature extraction, which has an important role on recognition. This process will generate the necessary component of an image called features[9]. The used feature extraction is chain code which functions as the direction search. The direction usually uses the following regulation Figure 7.

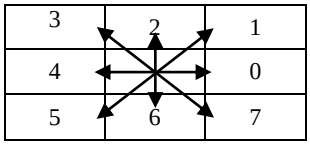

Figure 7. The direction of sequence code with 8 degrees[13]

The sequence code is made through checking the direction of a pixel connected to the other pixels with 8 degrees. Each direction has different number. The pointed sign shows the first pixel. It is a place where the next steps will be fixed[13]. Figure 8 shows letter B with its sequence code.

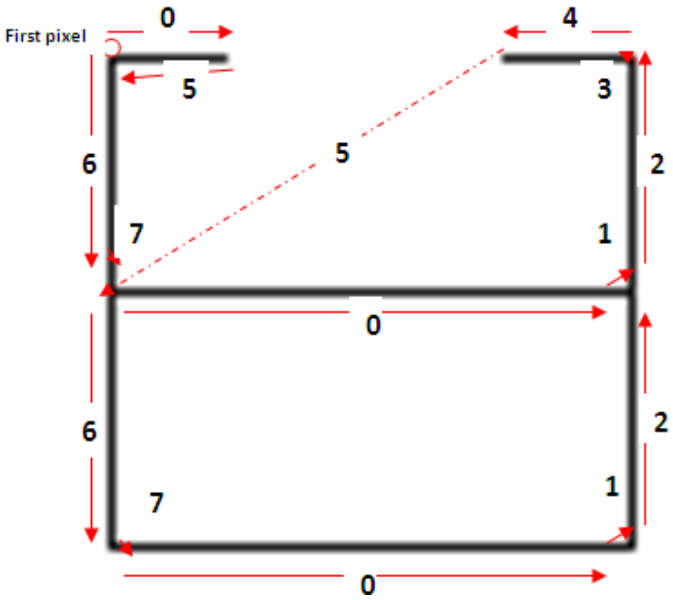

Figure 8. The direction of the sequence code in letter B

In order to be processed into SVM, the feature extraction must have the same amount of features. Therefore, normalization needs to be done. It also aims to decrease the amount of feature which reoccurs. The normalization process for chain code can be done with the following pattern[12]:

$$Fitur_n = \frac{F_n}{N_n} x\ N_{norm} \quad (8)$$

Note:
N = the amount of feature wanted
F_i = the amount of feature normalized
$\sum F_i$ = the amount of all letters normalized

2.3 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is found in 1992 by Boser, Guyon, and Vapnik. The basic concept is the combination of computation theories introduced in previous years, such as the margin hyperplane and kernel. The concept of SVM can be seen as an effort to look for the best hyperplane to separate two classes in the input grade.

Figure 9 shows some data of the two class member (+1 and -1), where (a) shows some alternative separated line (Discrimination Boundaries), and (b) shows the best hyperplane. The best hyperplane can be found by measuring the distance between the hyperplane and the nearer data known as margin [13]. The data in the margin is called support vector[14].

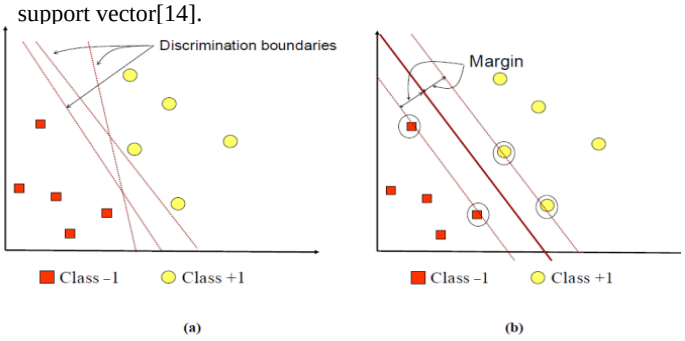

Figure 9. (a) The alternative separated field, (b) The best separated field with the biggest margin[14]

Practically, the real data does not only consist of two classes and they cannot be separated linearly. To solve this, an approach needs to be conducted so that SVM is able to classify the data input with N-Dimension or multi class, one of which is one against one[14]. Besides, to arrange the data input into higher dimension feature space, a function called kernel is needed. Table 1 shows some kernels used in SVM [15], and Figure 10 shows the process of recognition with SVM.

Table 1. Kernel function in SVM[16]

Kernel	Fungsi
Linear	$K(x,y) = x,y$
Polynomial	$K(x,y) = (\gamma x.y+r)^d$
Radial Basis Function	$K(x,y) = \exp(-\gamma\|x-y\|^2, \gamma>0$

Υ, r, and *d* are kernel parameter

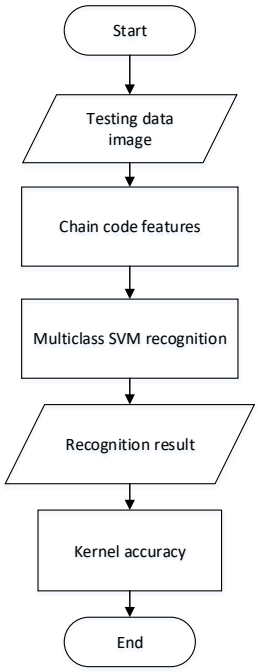

Figure 10. The diagram of Hangul recognition line with Multiclass Support Vector Machine

3. EXPERIMENT RESULT

Algorithm line Zhang Suen in Figure 6 shows that slight result is produced such as in Figure 11 (b).

(a) (b)

Figure 11. (a) Binary image, (b) Letter CH after thinning

After achieving the slight feature, the system takes it in accordance with the direction search in Figure 7. It generates the features of CH letter as follow:
0;0;0;0;0;0;0;0;0;0;5;6;6;6;6;6;6;6;6;6;6;6;6;6;6;6;6;5;4;4;
4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;4;0;0;0;0;0;0;
0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;5;6;6;6;6;6;
6;6;6;6;6;6;6;6;6;6;6;6;5;. . . ;CH;

If the feature is normalized according to the pattern (8), it has the same amount as the minimal feature from input image. Then, it will generate the following normalization feature:
0;0;0;0;0;0;0;5;6;6;6;6;6;6;6;6;6;6;6;6;5;4;4;4;4;4;4;4;4;
4;4;4;4;4;4;4;4;4;4;4;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;
0;0;0;0;5;6;6;6;6;6;6;6;6;6;6;6;6;6;5;. . . ;CH;

After getting the feature, system will do the recognition process. The letter recognition consists of 34 letters; each of them formulates 15 forms. So, they are 540 in total. The data scenario is shown in Table 2.

Table 2. Data scenario of Hangul letter recognition

Data Scenario	Training Data	Testing Data
DS1	170 (data A)	170 (data C)
DS2	340 (data A+B)	170 (data C)
DS3	510 (data A+B+C)	170 (data C)

Data testing process for DS1 and DS2 uses new data which have not been tested before while that for DS3 uses the tested ones.

Accuracy result of Hangul letter recognition for every kernel is shown in Table 3.

Table 3. Accuracy result of Hangul letter recognition

Kernel SVM	DS1	DS2	DS3
Linear	88,82%	94,11%	100%
Polynomial	89,41%	94,7%	100%
RBF	89,41%	94,7%	100%

Recognition mistake often occurs to the letter which has the same form, such as 우 (U) with 유(YU) or the other way around. Another letter which looks similar is 의(UI) with 위 (WI). The system output of Hangul letter is shown in Figure 12.

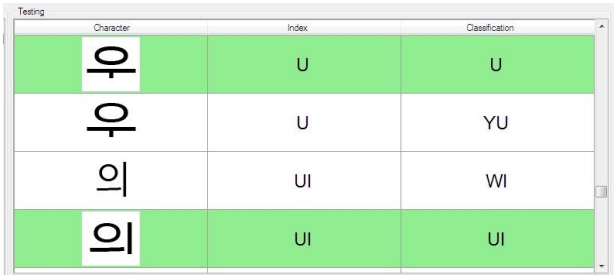

Figure 12. The result of Hangul letter recognition

In Hangul word recognition, researcher uses the form 2, where the training data are 195 and the testing data are 39. The accuracy for every kernel is shown in Table 4.

Table 4. Accuracy of Hangul word recognition

Kernel SVM	Accuracy
Linear	72%
Polynomial	72%
RBF	69%

Output from the recognition of Hangul word type 2 is shown in Figure 13. Recognition mistake occurs in the word which has almost similar form, such as 쵸 (CHYO) with 초 (CHO).

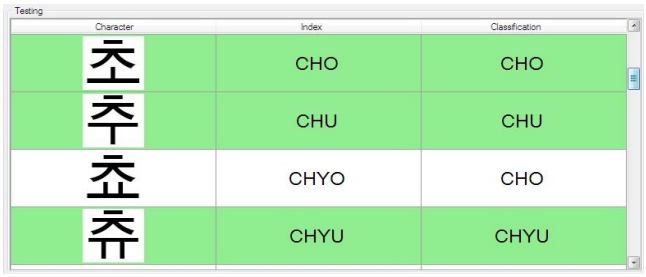

Figure 13. The recognition result of Hangul word type 2

4. CONCLUSIONS

The research concludes that the more the trained data, the higher the degree of accuracy. However, it needs to be reexamined until the fixed number of data which give the highest accuracy with SVM method is found.

In letter recognition process, kernel polynomial and RBF achieve the highest accuracy of 94,7% in data scenario 2 (DS2). On the other hand, linear process gives the lowest accuracy, 88,82%, in letter recognition, and RBF in Hangul word with 69%.

5. FUTURE WORK

The future research might employ the feature from the image with another method, while in the recognition process, the researcher can use SVM method with different kernel. Hangul recognition into Latin form may also be improved by adding the meaning of the trained word.

6. REFERENCES

[1] Seethalakshmi R., Sreeranjani T.R., & Balachandar T. 2005. Optical Character Recognition for Printed Tamil Text Using Unicode. Journal of Zhejiang University SCIENCE (2005), 1297-1305.

[2] Singh, D., Aamir Khan, M. & Bansal, A. 2015. An Application of SVM in Character Recognition with Chain Code. International Conference on Communication, Control and Intelligent Systems (CCIS). IEEE (2015), 167-171.

[3] Rajashekararadhya, S. V. & Ranjan, P. V. 2009. Support Vector Machine based Handwritten Numeral Recognition of Kannada Script. IEEE International Advance Computing Conference, 381-386.

[4] Tran, D. C., Franco, P. & Orgier, J.M. 2010. Accented Handwritten Character Recognition Using SVM – Application to French. IEEE International Conference on Frontiers in Handwriting Recognition, 65-71.

[5] Kyung-Won, K. & Jin H., Kim. 2003. Handwritten Hangul Character Recognition with Hierarchical Stochastic Character Representation. Proceedings of the Seventh International Conference on Document Analysis and Recognition.

[6] Kyung-Won, K. & Jin H., Kim. 2003. Handwritten Hangul Character Recognition with Hierarchical Stochastic Character Representation. IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 25, no. 9, 1185-1196.

[7] Lam, L., Seong-whan, L., & Suen, C. Y. 1992. Thinning Methodologies A Comprehensive Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 9, 869-885.

[8] Haseena, M. H. F & Clara, A. R. 2017. A Review on an Efficient Iterative Thinning Algoritm. International Journal Research in Sciencee, Engineering and Technology, vol. 6, no. 11, 541-548.

[9] Nasien, D., Haron, H. & Yuhaniz, S. S. 2010. Support Vector Machine (SVM) For English Handwritten Character Recognation. IEEE Second International Conference on Computer Engineering and Applications, 249-252.

[10] Ju, S. & Shin, J. 2013. Cursive Style Korean Handwriting Synthesis based on Shape Analysis with Minimized Input Data. IEEE International Conference on High Performance Computing and Communications & International Conference on Embedded and Ubiquitous Computing, 2231-2236.

[11] Putra, D. 2010. Pengolahan Citra Digital. Yogyakarta: Penerbit Andi.

[12] Zhang, T. Y. & Suen, C. Y. 1984. A Fast Parallel Algorithm for Thinning Digital Patterns. Communication of the ACM, vol. 27, no. 3, 236-239.

[13] Sutoyo. Teori Pengolahan Citra Digital. Penerbit Andi. 2009.

[14] Vijaykumar, S. & Wu, S. 1999. Sequential Support Vector Classifier and Regression. SOCO'99. http://homepages.inf.ed.ac.uk.

[15] Chih-Wei H, Chih-Chung C, Chih-Jen L. A Practical Guide to Support Vector Machine. Department of Computer Science National Taiwan University, Taipei. May 2016.

[16] Fadel, S., Ghoniemy, S., Abdallah, M., Sorra, H. A., Shour, A., Ansary, A. 2016. Investigating the Effect of Different Kernel Functions on the Performance of SVM for Recocnizing Arabic Characters. International Journal Computer Science and Aplications, vol. 7, no. 1, 446-450.