A Robust Method for Recognizing Accents in Vietnamese Handwriting Characters

Trong-Nguyen Nguyen, Huu-Hung Huynh
DATIC, Department of Computer Science
University of Science and Technology
Danang, Vietnam
Email: ntnguyen.dn {at} gmail.com

Jean Meunier
DIRO, University of Montreal
Montreal, Canada

***Abstract*— Handwriting character recognition is one of the most common research topics. Many approaches have applied to English characters and achieve high accuracy. However, the complexities in the language of each country are not same. Vietnamese handwriting character recognition is facing many problems, most of them come from the accent. This paper focuses on accent recognition, especially when there is a connection between two accents - a common problem which affects the identification result. Our approach starts with separating accent from character using the connected-component labeling method. The obtained accent then is checked if it is single or multiple (the combination of many accents). The recognition is performed using support vector machines with the single accent, or hidden Markov model if the accent is multiple. Proposed solution has been tested and obtained high accuracy.**

Keywords*—Vietnamese handwriing character, accent, corner detector, branch separating, invariant moment, hidden Markov models (HMMs)

I. INTRODUCTION

Handwriting character recognition has been studied for more than 40 years. The most recognized language is English, in which there is no accent, so results are often significantly higher than in other languages. There are many researches on Latin alphabet recognition than accented character (Latin based) identification because of the high complexity obtained from the appearance of accent. Vietnamese is one of the most complex languages, in which each character can consist of one or two accents. Therefore, the results of researches on Vietnamese language are very limited or high accuracy with ideal writing ways. Our approach deals with the problem that the connection of two accents is exist – a common phenomenon which affects the recognition result. For identifying, we used two different methods, corresponding with cases that there is only one accent and many connected accents. Proposed solution is a combination of many image processing techniques on the spatial domain for providing high efficiency in solving the problems that the researches on Vietnamese handwriting character recognition are facing.

Vietnamese accents consist two groups are shown in the Fig. 1, in which it is acceptable to combine two accents of two groups.

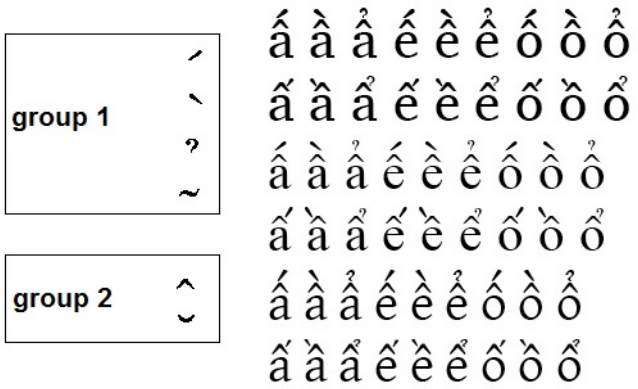

Fig. 1. Six Vietnamese accents and accented character examples

II. RELATED WORK

Recently, many approaches on recognizing handwritten character using image processing techniques have been implemented. The overall objective of these methods is to help the computer receiving and interpreting intelligible handwritten input from images or other devices. An overview of common handwritten character recognition systems is shown in Fig. 2.

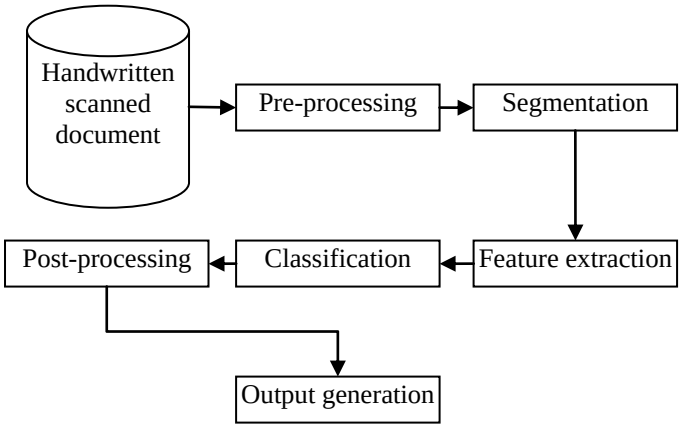

Fig. 2. Common handwritten character recognition systems

Two newest approaches on accented character recognition are proposed in [1, 2]. The authors used the combination of on-line and off-line data, in which, the off-line features are

extracted from the artificial image which is obtained by converting on-line data (captured from a tablet) into an image. First of all, seven statistical and structural feature families were used, including: 7 Hu invariant moments; horizontal and vertical projections; top, bottom, left and right profiles; intersections with horizontal and vertical straight lines; holes and concave arcs; the top, bottom, left and right extremities; the end points and junctions. Then they added some more features extracted from off-line data that include Radon invariants [3] and Zernike moments [4]. Finally, the features extracted from on-line data such as starting and ending points, number of strokes, each direction of strokes at starting and ending are added to the features set. Each character is represented by a large vector of 254 dimensions, which is a concatenation of the off-line and the on-line features mentioned above. Logically, a process for selecting relevant features is needed to eliminate redundancies. For the selection matter, first of all, the best-first algorithm [5] is used. It helps to eliminate many features that are not relevant. Then, the selected features are re-evaluated individually by Weka tools [6]. Finally, 45 features considered as the most pertinent are retained, and the recognition step was performed using SVM.

The accuracy which they obtained is very high. However, the writing ways are very ideal, in which the accent(s) and letter are separated clearly. This is not always true in handwritten documents. In addition, it takes a lot of time to recognize each character because of the large number of extracted features.

Our approach deals with these problems: recognizing connected accents and requiring lower computational cost.

III. Proposed Approach

In this section, we propose the process to recognize the accent, even if there is a connection between accents.

An overview of our approach is shown in the Fig. 3.

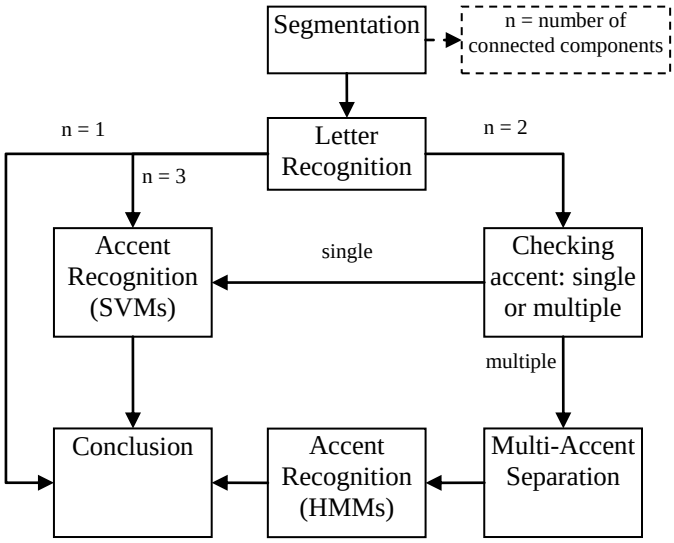

Fig. 3. Proposed approach overview

At first, the segmentation step is used to separate letter and others in a character. The largest number of connected components can be obtained is 3 (one letter and two accents). Particular examples for each case are: ‘o’, ‘ó’, and ‘ô’ corresponding 1, 2, and 3 components, respectively.

The simplest case is that the object has only one region. We suppose that this is a single letter. Many proposed methods on recognizing English handwriting character can be used to identity it. In the case that the character has three components, we have two single-accents which are need to be recognized. This process is done using SVMs with robust features which are described in the later section.

The remaining case (the character has two components) is more complex because the accent part can be a combination of accents. To solve the recognition problem, we need to check if this accent part is a single-accent or multiple-accent (connected accents). With the single-accent, we use SVMs to identify it as mentioned above. Otherwise, the accent part is separated and each two-pieces combination is recognized using HMMs combined with chain code feature. The detail processing steps in Fig. 3 are presented as following.

A. *Pre-processing*

The noise created by the optical scanning devices leads to poor recognition accuracy. These imperfections must be removed before performing main processing steps. Noise can be appeared in an image during image acquisition and transmission. For performing noise removal, we used median filter combined with morphological operations such as dilation and erosion [7].

B. *Segmentation*

Segmentation is a necessary process since Vietnamese characters usually consists of a letter and accent(s) such as ‘á’, ‘â’, ‘ă’, ‘ấ’ with the corresponding letter is ‘a’. At first, the connected-component labeling method [8] is used to separate each part of the character. Then these parts are classified base on the position, with the bottom part is the letter (see Fig. 4).

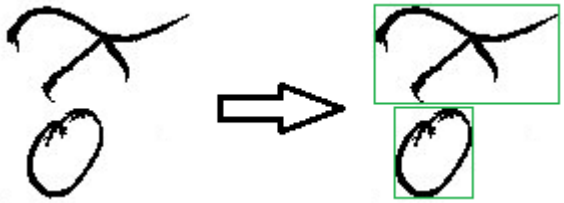

Fig. 4. Character ‘ỗ’ with two accents (letter ‘o’, accents ‘^’ and ‘~’)

C. *Letter recognition*

The individual letter in Vietnamese alphabet can be recognized easily by approaches that introduced for English language. This step is not the focus of our paper, so this content is not discussed in detail.

D. *Single-accent recognition*

1) Feature extraction: For identifying single-accent, we use 10 features, include the ratio between dimensions of the

original object, five invariant moments and four values of the rate between the number of object's pixels in each quadrant and total object's pixels of the resized image. Before performing feature extraction, we fill holes inside the object using the flood fill algorithm [9].

a) Ratio between dimensions: Six Vietnamese accents have different rates between two dimensions, so that this characteristic is useful to distinguish them. This feature is very simple to compute:

$$ratio = \frac{width}{height} \quad (1)$$

After getting this value, the accent is resized to be a square image with the size

max(*width*, *height*) × max(*width*, *height*)

Fig. 5. Resizing a 19×70 tilde to size of 70×70

This process helps that the remaining features bring higher identification accuracy.

b) Invariant moments: To determine invariant features in scaling and rotating, we used the normalized central moments defined as (Hu, 1962):

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad (2)$$

where $\gamma = \frac{p+q}{2} + 1 \quad \forall p+q \geq 2$, μ_{pq} values are centralized moments that are described in [10].

There are seven invariant moments, and we use five first values for recognizing, consist two second-order and three third-order moments:

$$M1 = \eta_{20} + \eta_{02} \quad (3)$$

$$M2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (4)$$

$$M3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (5)$$

$$M4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (6)$$

$$M5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{03} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \quad (7)$$

c) Rate of each quadrant

In experiment, we found that these characteristics have large differences between accents (see Fig. 6). These four values are obtained using below equations:

$$R_i = \frac{\sum_{4} pixel_{object} \in Area_i}{\sum_{i=1} pixel_{object} \in Area_i} \quad (8)$$

where $i \in \{1,2,3,4\}$ and $Area_i$ is the i^{th} quadrant of the image. By combining these 4 values and 6 features described earlier, we have the feature vector that contains 10 elements for recognizing single-accent.

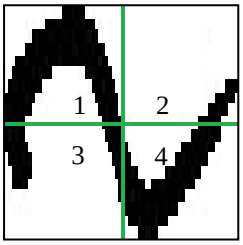

Fig. 6. Four quadrants for computing rates

2) Training and recognition

In machine learning, SVMs are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In more details, when the data cannot be separated by a hyper plane in their original domain, we can project them into a higher dimensional Hilbert space [11] and attempt to linearly separate them in the new space using kernel functions. Therefore, the decision boundary is given by

$$f(x) = sign(\sum a_i y_i K(x, x_i) + b) \quad (9)$$

where $K(x, x_i)$ is a kernel function, a_i and b are parameters and y_i represents one of the two classes (y_i = 1 or -1). Frequently used kernel functions are the linear kernel, the polynomial kernel, and the Gaussian radial basis function kernel, which is defined as

$$K(x_i, x_j) = \exp(-\gamma \left\| x_i - x_j \right\|^2) \quad (10)$$

Since sign language recognition is applied for more than two gestures, we used the multi-class SVM for classification. One common method is building binary classifiers which distinguish between one of the labels and the rest (one-against-all) or between every pair of classes (one-against-one). We used the one-against-one approach because of the large number of classes [12]. The classification is done by a max-wins voting strategy, in which every classifier assigns the instance to one of

the two classes, then the vote for the assigned class is increased by one, and finally the class with the most votes determines the instance classification.

E. Checking accent is single or multiple

Some popular cases of multiple-accent are shown in Fig. 7.

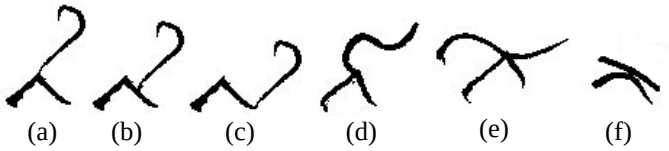

Fig. 7. Some connected accents excamples ('^', '^~' and '^`')

We can see that each of these characters has a corner with 3 (a, b, d) or 4 (e, f) corresponding branches, or two 2-branches corners (c), while single accent can has most only one corner with two branches (accent '^'). Therefore we can check if the current accent is single or multiple bases on this property. The used algorithm is shown clearly in the diagram in Fig. 8.

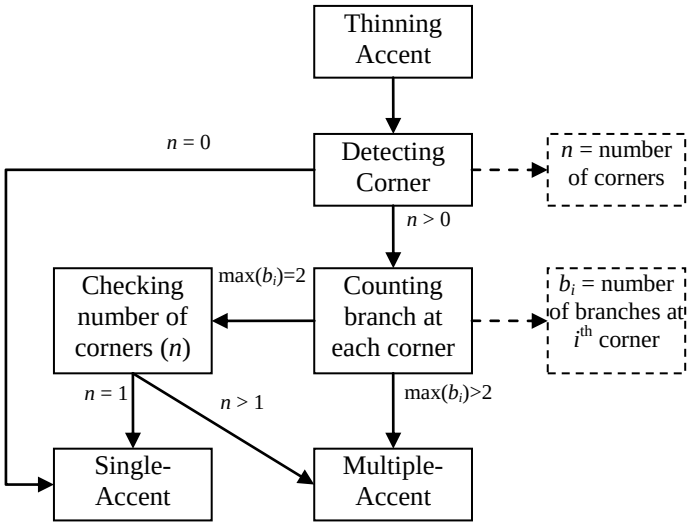

Fig. 8. Algorithm for detecting that accent is single or multiple

Our used corner detector is based on Plessey operator which is described in [13]. The number of branches at each corner is counted using a simple method. We defined a circle with the radius is 3 pixels and the center is at corner. Then we determine thinned object's pixels which are located on the circle. The number of branches at this corner is equal to the number of connected components obtained from determined pixels.

If the accent is detected as single, it is classified using the described SVMs method. Otherwise, we need to separate multiple-accent and recognize with suitable method.

F. Multiple-accent separation

This step is done using the combination of the original and thinned multiple-accent. The corner in original image is removed based on its coordinates (and its neighborhood) obtained from the thinned accent to get separated pieces (branches). The current problem needs to be solved is clustering these pieces into two groups corresponding two accents which are recognized. The detail is presented below.

1) Separating branches

At the step which checks the accent is single or multiple, we obtained the coordinates of each corner point. Therefore we can use them to get each branch of the object as follow:

- Remove current corner point
- Set $r = 1$ (radius of neighborhood)
- Set n = number of connected component
- Loop following steps until $n > 1$
  - $r = r + 2$
  - Remove neighborhood $r \times r$
  - n = number of connected component
- Get all connected components

After applying this process for each corner, we have a set of separated pieces from the multiple-accent.

2) Clustering branches into two groups

Dividing separated branches into two groups is performed by taking two of them to the first group and the others to the remaining group. Therefore, we have $\binom{n}{2}$ dividing ways.

If we just combine all pieces in each group based on the original coordinates, the obtained accent may be not intact because corners are removed. Therefore, we used subtraction to get the complete accent corresponding to each group. With each group, we subtract all pieces from the original image to obtain the accent corresponding to the remaining group. A specific example is illustrated in Fig. 9.

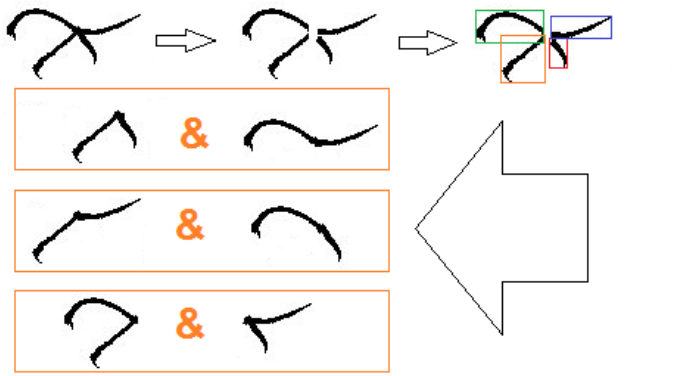

Fig. 9. Three ways for separating into two groups (accents '^' & '~')

3) Recognizing each pair of accents

We do not used SVMs method described above to recognize these pairs of accents because a comparison between these pairs is required to choose the best result. Therefore, in this step, we use the chain code description combined with hidden Markov models [14]. The training process is performed with the chain code features. With each pair, we compute six

probabilities of each accent corresponding six HMMs. Each accent is assigned to the class which gives the highest value.

After getting the largest probabilities of accents in each pair, we compute the sum of two values and use this result for the comparison. The recognition result which is corresponding to the pair with highest sum is the final result of recognizing multiple-accent.

IV. Experimental Result

Currently, there is no standard Vietnamese handwriting character dataset, so our approach is experimented with local data. The used data are collected from our laboratory and the open data of some students [15]. Each image in the dataset contains one Vietnamese character (accented and non-accented). Accent separation is done manually to serve the training process. Some characters in the dataset are shown in Fig. 10.

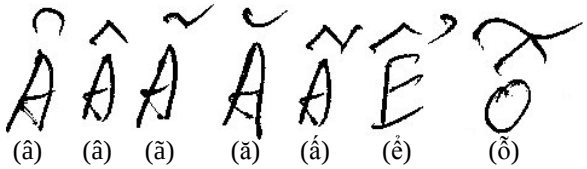

Fig. 10. Some accented characters in dataset

The training is performed with 500 samples for each accent, corresponding to 3000 used samples; and the number of testing samples is 1200. Testing results of single-accents using SVMs method combined with our features are shown in Fig. 11.

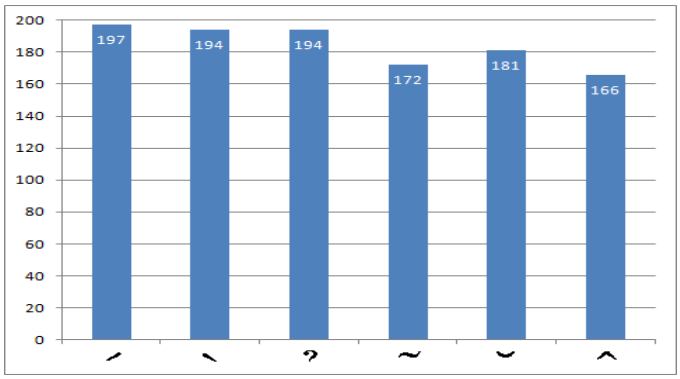

Fig. 11. Testing results of single-accents

Testing results with multiple-accents are also quite promising with the average recognition accuracy is up to 90%. The details are shown in Fig. 12.

Most of accents are recognized with high rates. However, there are still some accents that have been less successful in recognizing such as '^' and '~'. The recognition of these accents is affected because the writing way is incorrect (rotation angle is too large), therefore it has mistakes in identification.

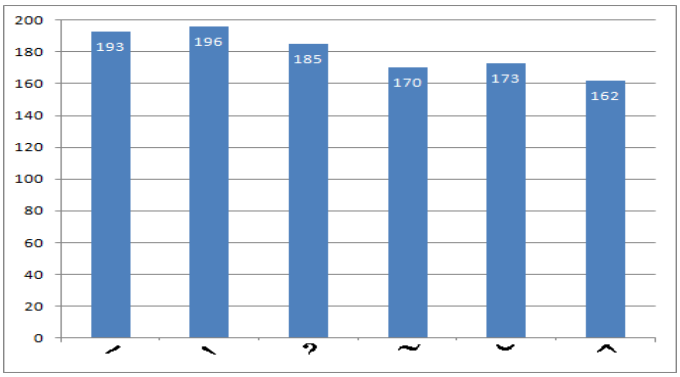

Fig. 12. Testing results of multiple-accents

V. Conclusion and Discussion

In this paper, a robust approach is proposed to recognize accent of Vietnamese handwriting characters. Our approach starts with separating accent from character using the connected-component labeling method. The obtained accent then is checked if it is single or multiple (the combination of many accents). The recognition is performed using support vector machines with the single-accent, or hidden Markov model if the accent is multiple. The most advantage of our approach is that the accents can be recognized, even if there is a connection between them. In addition, the used features for using SVMs are very powerful and easy to compute. Our system has low computational cost because of the simplification of used operations. As further work, our method will be improved to recognize the accents, even when they are written in wrong ways. In addition, some spelling rules are combined to get higher recognition accuracy.

Acknowledgment

This work was supported by the DATIC, Department of Computer Science, University of Science and Technology (DUT), The University of Danang, Vietnam and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] D. C. Tran, "An efficient method for on-line Vietnamese handwritten character recognition," Proceedings of the Third Symposium on Information and Communication Technology (SoICT), pp. 135-141, August 2012.

[2] D. C. Tran, P. Franco, and J. M. Ogier, "Accented Handwritten Character Recognition Using SVM – Application to French," International Conference on Frontiers in Handwriting Recognition (ICFHR), Kolkata, pp. 65-71, November 2010.

[3] D. V. Jadhao and R. S. Holambe, "Feature Extraction and Dimensionality Reduction Using Radon and Fourier Transforms with Application to Face Recognition," Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), vol. 2, pp. 254-260, 2007.

[4] M. Zhenjiang, "Zernike moment-based image shape analysis and its application," Pattern Recognition Letters 21, pp. 169 – 177, 2000.

[5] P. Pudil, J. Novovicova and J. Kittler, "Floating search methods in feature selection," Pattern Recognition Letters, pp. 1119-1125, 1994.

[6] Weka home page: http://www.cs.waikato.ac.nz/~ml/weka

[7] G. Bradski and A. Kaehler, Learning OpenCV. O'Reilly Media, 2008.

[8] S. Rajaraman and A. Chokkalingam, "Connected Components Labeling and Extraction Based Interphase Removal from Chromosome Images," International Journal of Bio-Science and Bio-Technology, vol. 5, no. 1, pp. 81-90, February 2013.

[9] C. Bond., 2011. An Efficient and Versatile Flood Fill Algorithm for Raster Scan Displays. Available at: http://www.crbond.com/ papers/fldfill_v2.pdf [Accessed: 26 December 2013].

[10] M. Nixon and A. Aguado, Feature Extraction & Image Processing 2nd. Academic Press, UK, 2008.

[11] B. K. Sriperumbudur, K. Fukumizu, and R. G. Lanckriet, "Learning in Hilbert vs. Banach Spaces: A Measure Embedding Viewpoint," Advances in Neural Information Processing Systems 24 (NIPS), Cambridge, MA: MIT Press, pp.1773-1781, 2011.

[12] J. Milgram, M. Cheriet and R. Sabourin, "One Against One or One Against All: Which One is Better for Handwriting Recognition with SVMs?," Proceedings of 10th International Workshop on Frontiers in Handwriting Recognition, France, October 2006.

[13] D.Parks and J.P.Gravel, "Corner Detection," International Journal of Computer Vision, 2004.

[14] S. J. Cho, "Introduction to Hidden Markov Model and Its Application," Samsung Advanced Institute of Technology (SAIT), 2005.

[15] "Sapphire-ocr data", http://sapphire-ocr.googlecode.com/files/samples-full.zip.