*International Journal of Engineering Science Invention*
*ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726*
*www.ijesi.org ||Volume 5 Issue 2|| February 2016 || PP.15-23*

# Study on a Hybrid Segmentation Approach for Handwritten Numeral Strings in Form Document

Tianxiang Zheng
*(Shenzhen Tourism College/Jinan University, China)*

**ABSTRACT** *: This paper presents a hybrid approach to segment single- or multiple-touching handwritten numeral strings in form document, the core of which is the combined use of foreground, background and recognition analysis. The algorithm first located some feature points on both the foreground and background skeleton images containing connected numeral strings in form document. Possible segmentation paths were then constructed by matching these feature points, with an unexpected benefit of removing useless strokes. Subsequently, all these segmentation paths were validated and ranked by a recognition-based analysis, where a well-trained two-stage classifier was applied to each separated digit image to obtain its reliability. Finally, by introducing a locally optimal strategy to accelerate the recognition process, the top ranked segmentation path survived to help make a decision on whether to accept or not. Experimental results show that the proposed method can achieve a correct segmentation rate of 96.2 percent on a large dataset collected by our own.*

**KEYWORDS -** *character recognition, character segmentation, form document, handwritten numeral strings, single- or multiple-touching*

## I. INTRODUCTION

Optical Character Recognition (OCR) has received considerable attention in recent decades and form document recognition has become an important field of OCR research and application. There are two crucial problems in the development of form document: integrated character extraction and segmentation of touching characters. The main focus of this work is dedicated to the segmentation of grades on handwritten numeral strings(two-digits) in form document.

Segmentation of handwritten numerals is a key step for automatic recognition system since the segmentation results have great effect on the recognition rate. So far, a large number of methods for segmentation of handwritten numerals have been proposed in the literature. Casey[1], Lu[2], Ribas[3] and Saba[4] presented respectively a survey or an overview on various techniques for segmenting handwritten characters. It is well recognized that the difficulties of segmentation stem not only from variations existing in shape, but also from various ways of touching. There are two major categories of types of touching numeral strings: single-touching and multiple-touching, which can be further grouped into five subtypes[5].

To segment the touching numeral strings, the literature shows that there are commonly three families: the foreground-based method(FBM), the background-based method(BBM) and the recognition-based method(RBM). The FBM focuses on foreground pixels, and features like contour tracing[6, 7], highest curvature[6], corner points[8], stroke analysis[9-11], abstractive primitives[12], etc., are examples of this

category. On the contrary, the BBM works on background pixels and utilizes the feature points on the background regions (such as face-up valley, face-down valley, loop region)[13, 14] or those on the background skeletons(such as upper segment, lower segment, hole segment)[15]. The RBM, however, involves a recognizer to help separate the connected strings[16-19].

All the three methods are very useful in different situations under the property of the actual data. But there are some formidable limitations when they are put into practical applications. For example, most of the connected numeral strings of Type 1 and Type 2 in Chen's work can be successfully segmented by the FBM or BBM, but they tend to become less powerful to get precise identification in separating those of Type 2-5[5]. The RBMs, with the correct rate of segmentation depending too much on the robustness of recognizer and at the cost of more computation time, usually fail to separate those of Type 2, 4 or 5. Some other RBM suggests the whole entity of numeral string to be recognized without segmentation[20]. Although it offers an opportunity to minimize the risk of making errors caused by incorrect segmentation, the approach needs a rather complex classifier to solve all kinds of touching styles, which appears to be impractical.

Apparently, the combined use of existing approaches, if possible, will enhance the segmentation results. This idea has been developed and investigated since last decade, in which the foreground and background analysis were utilized[6, 21, 22]. But either a lack of rigor by the absence of ligatures or the unstable rule on the selected features is a major limitation when put into practical use. In form documents, due to the large variability of handwriting, different writing styles of characters and all kinds of ligatures, a more robust rule-based technique is needed. A recognizer might be helpful in such a complicated situation, because in many cases, the segmentation and the recognition could not be considered independently when touching between characters/digits is present[23]. Previous studies also attempted to mingle segmentation with recognition, but some of them required large number of segmentation cuts[24] or aimed at multi-oriented touching[25], others may fail to achieve a high accuracy[26]. In this paper, we propose a hybrid segmentation algorithm to segment touching numeral strings in form document, which combines the foreground, background and recognition analysis to get the best segmentation path. In the rest of the paper, a brief introduction of the extraction of characters in form document is presented, followed by the methodology underlying the new segmentation approach. Experimental results and conclusion are included in the end.

## II. METHODOLOGY

The flowchart of the algorithm is shown in Fig. 1 and more details can be found in Fig. 2.

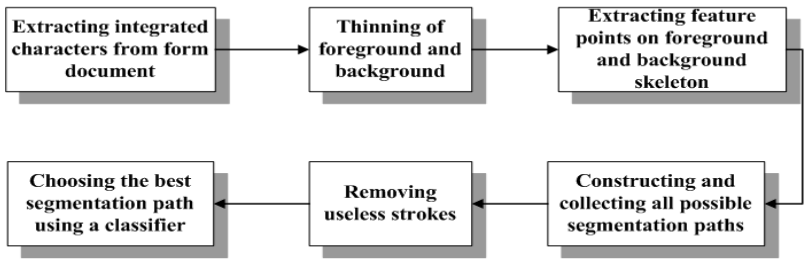


Figure 1 Flowchart of the proposed approach

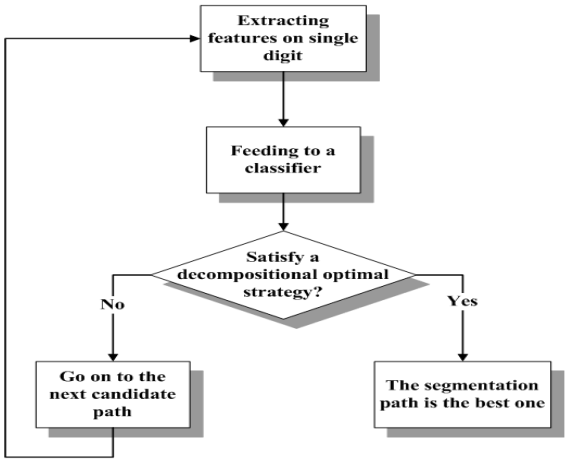


Figure 2 Outspread of the last module "choosing the best segmentation path using a classifier" in Fig. 1

### 2.1 Integrated extraction of characters

The prerequisite for character segmentation is the extraction of characters from form document where, in reality, the input characters(numeral strings) are intermixed with structured line patterns. In order to extract the characters, elimination of the structured line patterns is crucial. With the method documented in previous study[27], we manage to remove these line patterns and mend successfully the broken strokes. Fig. 3 gives some resultant examples, from which one can see that the characters are accurately extracted and the broken strokes are perfectly repaired.

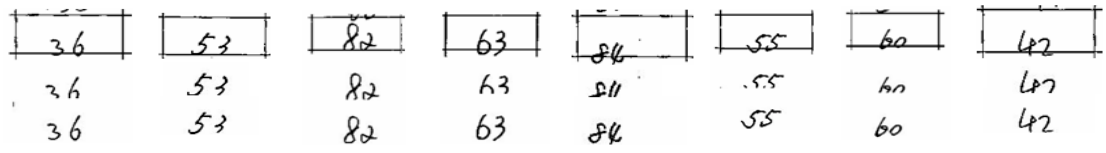

Figure 3 Results of character extraction from form document. For each column, the top image is the original cell unit filled with the character, the middle image is the character extracted by simply removing the structured line, and the bottom image is the character extracted by the method documented in previous study[27]

### 2.2 Strings segmentation

Based on our observation, large majority of touching in the simulation belong to single-touching, most of which are ligatures. Maybe this is because the cell unit in form document is wide enough to input a numeral. As a result, multi-touching seldom occurs. In other words, Type 1-4 in Chen's study[5] cover most cases of touching. For the time being, we omit multi-touching numeral strings from our simplified study to alleviate the computation burden.

2.2.1 Extraction of feature points from foreground and background skeleton

An input form image is thinned to obtain its thin-line representation. The purpose of thinning is to reduce the width of the form image to just one single pixel to facilitate the extraction of feature points from the digitized pattern. To find all of the feature points on the foreground and background of the image, a thinning algorithm[28] is first applied to foreground and background regions. Then the fork points, end points and corner points[5] are calculated and marked from the skeleton of the foreground and background regions. To illustrate the process, we give here some examples of foreground and background thinning results, as shown in Fig. 4.

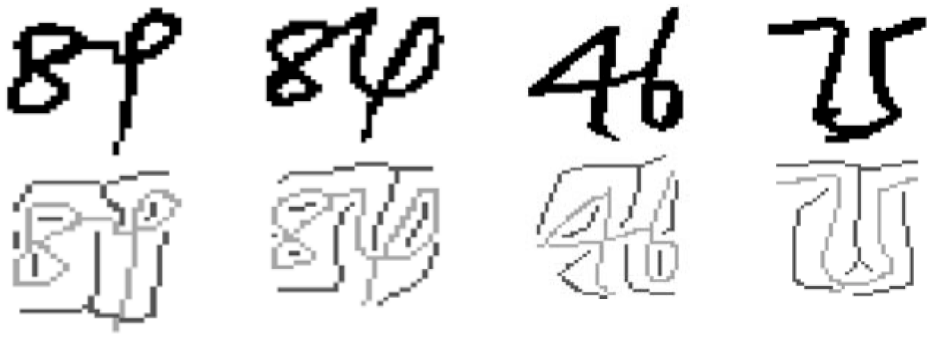

Figure 4 Some examples of thinning results on foreground and background regions. For each column, the first one is the original image, and the second one is the result after thinning

2.2.2 Construction of possible segmentation paths with the removal of useless strokes

According to the assumption, the touching style of the numeral strings is limited to single-touching during the construction. The details of constructing single-touching segmentation paths can be found in Chen's work[5], and the troublesome ligatures are removed in this stage, an unexpected bonus. All of the possible segmentations are recorded for further analysis.

2.2.3 Decision of the best segmentation path using a recognizer

(1) System architecture

We utilize a classifier to recognize the single digit images separated by each segmentation path to decide whether it is the best. The neural network classifier we opt for here is a Multi-Layer Perceptron (MLP) trained by the traditional Back-Propagation(BP) algorithm with the scaled conjugate gradient applied to a sum-of-squares performance function. The transfer function employed is the familiar sigmoid function.

Note that a single Neural Network(NN) often exhibits the overfitting behavior which results in a poor generalization performance when trained on a limited set of training data[29]. Considering this, we use two NNs for the classification instead. The first NN is designed to perform fast classification and provide a low misclassification rate using a strong rejection criterion. Rejected patterns are forwarded to the second NN that uses additional, more complex features, with a well-balanced rejection criterion.

The first set of features contains 103 easy-to-extract features that are inputs for the first NN classifier(103-40-10). From the patterns that are rejected from this network, additional 64 more robust features are extracted. All 167 features are used in the second classification stage where difficult-to-classify patterns are forwarded to the second NN (167-40-10).

(2) Feature extraction on single digit

Previous study pointed out that features that offer better discriminative power are usually more complicated and harder to extract regarding processing time that can not be neglected when building a fast recognition system[29]. For this reason, we use a two-stage classification strategy instead to make a compromise between accuracy and complexity.

The first 103 features used in the first classification stage are simply horizontal, vertical and diagonal projections. Since the character images are of different size, the projection vectors are linearly rescaled to get 20 features from the horizontal projections, 15 from the vertical, and 34 from each of the two diagonal projections.

The second 64 features we choose are invariant contour curvature[30] due to its capability of discriminating similar handwritten numerals(such as '4' and '9').

(3) Classification based on a locally optimal strategy

Classification is performed in two stages where the first stage classifier sends rejected patterns to the second one.

In the first stage, our goal is to perform fast classification of easy-to-classify patterns and keep low misclassification rate. The input feature set is the global characteristic containing features extracted from pattern projections described above. The rejection criterion is based on the "top 2" NN outputs. Each sample for which the highest NN output $O_1$ was smaller than a certain threshold $T_1$ ( $O_1 < T_1$ ) or for which the difference between the "top 2" classifier outputs was smaller than a certain threshold $T_2$ ( $O_1 - O_2 < T_2$ ) are rejected. The rejection criterion is set to $T_1$ = 0.95 and $T_2$ = 0.6 to obtain low misclassification rate. In the second stage, the NN uses 64 additional features(contour curvature) extracted from the digit image. Varying the thresholds we have found suitable values $T_1$ = 0.95 and $T_2$ = 0.6, a relatively high level since misclassification carries a higher risk than rejection.

To accelerate the classification, we introduce a locally optimal strategy. The classification terminates until one of the candidates(segmentation path) satisfies the locally dominate criterion: the reliability of the image with connected digits, which is defined as the minimal outputs of the two single digit images, is no less than 0.98 within a certain scope of testing. By introducing a container B(of size N) similar to that in Support Vector Machine(SVM) as working set[31], the strategy works by the following rule, as depicted in Table 1.

Table 1 Algorithm of Locally Optimal Strategy

| **Algorithm 1 Locally Optimal Strategy** |
|---|
| While the locally dominate criterion is violated do<br>Select N segmentation paths for the container B<br>Find the maximal reliability of the image with connected digits from B<br>end while |

## III. EXPERIMENTAL RESULTS

After training the classifier, 14875 numeral strings (two digits) are used to test the performance of the reported algorithm. In an inside test of 14875 images, 207 are rejected (rejected rate = 1.4 percent) and 14668 are accepted. Among the 14668 accepted images, 14109 test images are successfully segmented (correct rate = 96.2 percent) and 559 fail (error rate = 3.8 percent).

Fig. 5 shows some numeral strings that are correctly segmented, from which one can see that our algorithm can successfully segment the strings in many difficult cases, especially those with ligatures. In Fig. 6, we illustrate two examples that are erroneously segmented, together with two rejected patterns. The ligatures caused by scratchy handwriting are the major reason for the error segmentation, while the rejection is induced by the large overlapping part of the two connected digits and the insufficient amount of training samples.

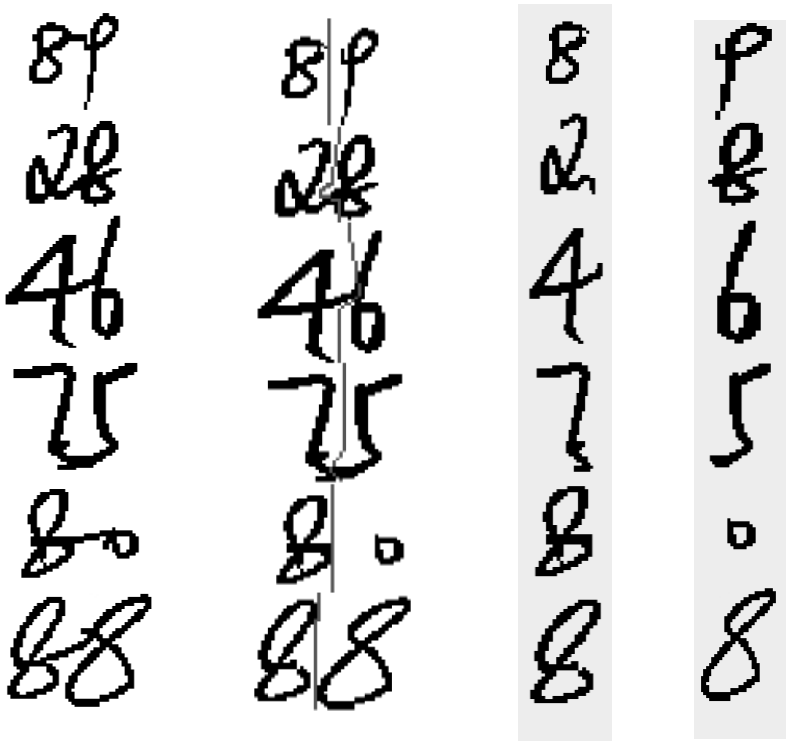

Figure 5 Examples of separation for connected numeral strings. For each column: (a) Test images (b) The best segmentation paths (gray lines) (c) Left part of the numeral strings after segmentation (d) Right part of the numeral strings after segmentation

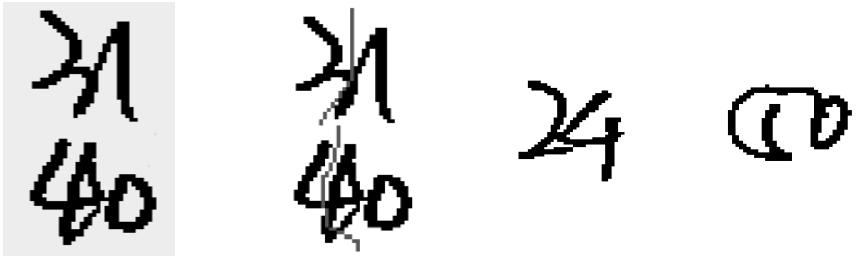

Figure 6 Examples of erroneously segmented or rejected images. For each column: (a) Test images (b) The segmentation path (gray lines) found by the proposed algorithm (c) Rejected image (d) Rejected image

## IV. CONCLUSION

In recognizing connected characters, a high quality segmentation technique is essential. In this paper, we put forward a novel segmentation algorithm that combines the foreground, background and recognition analysis for segmentation of touching handwritten numeral strings in form document. The algorithm first applies respectively the thinning procedure to the foreground and background regions. The feature points on the foreground and background skeletons are then extracted and linked for finding the possible segmentation paths in single-touching numeral strings. An unexpected benefit of removing the useless strokes is achieved, which can eliminates the troublesome step of constructing path on them and facilitate the subsequent recognition

process. Finally, a recognizer with a two-stage classification is employed to rank all possible segmentation paths and decide the best one, with a fast strategy developed by our own.

The proposed algorithm is validated on 14875 images of connected numeral strings. Experimental results demonstrate that the hybrid system reaches a correct segmentation rate of 96.2 percent. This efficiency is beneficial from a locally optimal strategy as well as carefully optimized classifiers and elaborate choice of features, a clear win between accuracy and computational overhead.

The major limitation that arises in connection with this novel approach is that we haven't taken into account the multi-touching strings in the process of segmentation. Though it rarely happens, it should be noted that an accidental touching caused by scratchy handwriting or preprocessing (such as binaryzation and broken-stroke mending) might lead to multi-touching. Segmentation on multi-touching strings is necessary when a higher correct rate is imposed. Another way to improve the overall performance of segmentation is to use a SVM classifier[32] instead of Neural Network or to introduce verification strategy[24] in the recognition-based procedure to avoid over-segmentation or under-segmentation. All these will be part of our future work.

## V. Acknowledgements

The author would like to thank Professor Lihua Yang of School of Mathematics and Computational Science, Sun Yat-sen University, for his technical assistance, valuable suggestions and critical comments without which the research could not have proceeded to its present form.

This work was supported in part by: Special Funds of Undergraduate Course Center of Educational Reform Research Project of Jinan University in 2015, Grant No. JG2015094.

## REFERENCES

[1] R. G. Casey and E. Lecolinet. Strategies in character segmentation: a survey, *3rd International Conference on Document Analysis and Recognition*, Montreal, Que., Canada, IEEE, 1995, 1028-1033.

[2] Y. Lu and M. Shridhar, Character segmentation in handwritten words - an overview. *Pattern Recognition*, 29(1), 1996, 77-96.

[3] F. C. Ribas, et al., Handwritten digit segmentation: a comparative study. *International Journal on Document Analysis and Recognition*, 16(2), 2013, 127-137.

[4] T. Saba, A. Rehman, and M. Elarbi-Boudihir, Methods and strategies on off-line cursive touched characters segmentation: a directional review. *Artificial Intelligence Review*, 42(4), 2014, 1047-1066.

[5] Y.-K. Chen and J.-F. Wang, Segmentation of single-or multiple-touching handwritten numeral string using background and foreground analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 2000, 1304-1317.

[6] N. W. Strathy, C. Y. Suen, and A. Krzyzak. Segmentation of handwritten digits using contour features, *2nd International Conference on Document Analysis and Recognition*, Tsukuba Science City, Japan, IEEE, 1993, 577-580.

[7] J. M. Hu, D. G. Yu, and H. Yan. Algorithms for partitioning path construction of handwritten numeral strings, *14th International Conference on Pattern Recognition*, Brisbane, Australia, Ieee Computer Soc, 1998, 372-374.

[8] Y. Lei, et al., Recognition-based system for segmentation of handwritten numeral strings. *Journal of Tsinghua University*, 45(4), 2005, 433-436.

[9] J. Ding, Z. Lou, and J.-y. Yang, Segmentation of numeral strings using stroke grouping. *Journal of Image and Graphic*, 14(8), 1993, 1609-1614.

[10] Z. Shi, et al. A system for segmentation and recognition of totally unconstrained handwritten numeral strings, *4th International*

*Conference on Document Analysis and Recognition*, ULM, GERMANY, IEEE COMP SOC, 1997, 455-458.

[11] J. Ding and J.-Y. Yang. Biomimetic segmentation of unconstrained touching numeral strings, *2nd International Conference on Information Science and Engineering*, IEEE, 2010, 1073-1076.

[12] D. K. You and G. Kim. An approach for locating segmentation points of handwritten digit strings using a neural network, *7th International Conference on Document Analysis and Recognition*, Edinburgh, Scotland, Ieee Computer, Society, 2003, 142-146.

[13] M. Cheriet, Y. S. Huang, and C. Y. Suen. Background region-based algorithm for the segmentation of connected digits, *11th International Conference on Pattern Recognition*, Hague, Netherlands, 1992, 619-622.

[14] J. Luo and L. Wang, A new segmentation method of unconstrained handwritten connected numeral string based on concave-and-convex feature. *Microcomputer Information*, 23(25), 2007, 275 276.

[15] Z. Lu, et al., A Background-thinning-based Approach for Separating and Recognizing Connected Handwritten Digit Strings. *Pattern Recognition*, 32(6), 1999, 921 933.

[16] G. Congedo, et al. Segmentation of numeric strings, *3rd International Conference on Document Analysis and Recognition*, Montreal, Que., Canada, IEEE, 1995, 1038-1041.

[17] T. M. Ha, M. Zimmermann, and H. Bunke, Off-line handwritten numeral string recognition by combining segmentation-based and segmentation-free methods. *Pattern Recognition*, 31(3), 1998, 257-272.

[18] S. W. Lee and S. Y. Kim, Integrated Segmentation and Recognition of Handwritten Numerals with Cascade Neural Network. *IEEE Transactions on Systems, Man and Cybernetics*, 29(2), 1999, 285 - 290.

[19] W. J. Song, et al., Method for handwritten numeral string segmentation based on recognition. *Computer Engineering and Design*, 28(21), 2007, 5198 5200.

[20] X. Wang, V. Govindaraju, and S. Srihari. Holistic recognition of touching digits, *6th International Workshop on Frontiers Handwriting Recognition*, Taejon, South Korea, 1998, 295 303.

[21] H. Fujisawa, Y. Nakano, and K. Kurino. Segmentation methods for character recognition: from segmentation to document structure analysis, *Proceedings of the IEEE*, Hitachi Ltd., Tokyo, Japan, IEEE, 1992, 1079-1092.

[22] J. Sadri, C. Y. Suen, and T. D. Bui. Automatic Segmentation of Unconstrained Handwritten Numeral Strings, *9th International Workshop on Frontiers in Handwriting Recognition*, Kokubunji, Tokyo, Japan, IEEE, 2004, 317-322.

[23] G. L. Martin, M. Rashid, and J. A. Pittman, Integrated segmentation and recognition through exhaustive scans or learned saccadic jumps. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(4), 1993, 831-847.

[24] L. S. Oliveira, et al., Automatic recognition of handwritten numerical strings: a recognition and verification strategy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(11), 2002, 1438-1454.

[25] P. P. Roy, et al., Multi-oriented touching text character segmentation in graphical documents using dynamic programming. *Pattern Recognition*, 45(5), 2012, 1972-1983.

[26] X. Zhao, Z. Chi, and D. Feng. An improved algorithm for segmenting and recognizing connected handwritten characters, *11th International Conference on Control, Automation, Robotics and Vision*, Singapore, 2010, 1611 - 1615.

[27] T. X. Zheng, L. Xie, and L. H. Yang, Integrated extraction of handwritten numeral strings in form document based on hybrid binarization. *Pattern Recognition and Artificial Intelligence*, 21(3), 2008, 369 375.

[28] T. Y. Zhang and C. Y. Suen, A fast parallel algorithm for thinning digital patterns. *Communications of the Acm*, 27(3), 1984, 236-239.

[29] D. Gorgevik and D. Cakmakov. An Efficient Three-Stage Classifier for Handwritten Digit Recognition, *17th International*

*Conference on Pattern Recognition*, IEEE Computer Society, 2004, 507-510.

[30] L. Yang, et al., Discrimination of similar handwritten numerals based on invariant curvature features. *Pattern Recognition*, 38(7), 2005, 947-963.

[31] T. Joachims, Making large-scale svm learning practical, in B. Scho lkopf, C.J.C. Burges, and A.J. Smola (Ed.) *Advances in kernel methods : support vector learning*, 3(Cambridge, USA: MIT Press, 1999) 169-184.

[32] Z. Jian and S. Wan-Juan. Handwritten numerical string recognition based on SVM verifier, *3rd International Conference on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, China, IEEE Computer Society, 2011, 186-188.