2011 International Conference on Document Analysis and Recognition

Localization of Digit Strings in Farsi/ Arabic Document Images Using Structural Features and Syntactical Analysis

Ali Abedi, and Karim Faez
Electrical Engineering Department,
Amirkabir University of Technology,
Tehran, Iran
ali_abedi@aut.ac.ir, kfaez@aut.ac.ir

***Abstract*—This paper presents a new method for localization of digit strings with a specific syntax in Farsi/ Arabic document images. First, some features are extracted from all connected components in each text line. These features, are provided for Farsi/ Arabic scripts, and have the ability to differentiate between digits and non-digit connected components. Then, these features are classified, and the probabilities of being in each of four classes digit, slash, double-digit, and non-digit, is assigned to each connected component. Next, discrete hidden Marcov model as syntactic analyzer, localize digit strings with desired syntaxes. The results which are presented for handwritten and machine-printed text lines, separately, are very promising.**

Keywords; digit strings localization; Farsi/Arabic document image analysis; feature extraction; handwritten dates; syntax verification.

I. Introduction

Localization of digit strings can be useful for indexing, retrieval and searching specific information in document images. Usually, there is no preliminary information about the location of digit strings in a document image. They can occur in any part of the document, such as text body, heading or in margins. Instead of applying a recognizer to all connected components of a document image and then choosing the desired digit strings, which is a very complex and time consuming task, first the location of digit strings of interest are identified, and then a proper digit recognizer can be applied only to these locations of document image.

Most prior research has focused on localizing digit strings in handwritten Latin documents, [1, 2, 3, 4, and 5]. However, some special characteristics of Farsi/ Arabic scripts make them absolutely different from other scripts. Thus the presented methods for Latin scripts cannot be used directly for Farsi/ Arabic scripts. Some specifications of Farsi/ Arabic scripts that are important to localization of digit strings are as follows:

- In Farsi/ Arabic scripts the words are cursive both in machine-printed and handwritten forms. While in Latin scripts the words are cursive only in handwritten forms.
- Most of the Farsi/ Arabic characters have dots or additional small markings called diacritics. These complementary parts usually have small sizes and are written detached from the main body of the characters at the top or bottom of them [6]. While none of the digits have dots or diacritics.
- Alphabetic parts of Farsi/ Arabic scripts are written from right to left, while numeric parts are written from left to right.
- Many characters have ascenders/ descenders, and most of them are not vertical.

Except our previous work [7], there is no research about localization of digit strings in Farsi/ Arabic scripts. In this paper, we present a new method on the localization of digit strings in Farsi/ Arabic document images. The rest of the paper is organized as follows: The proposed algorithm for localization of digit strings is described in Section 2. Experimental results are presented in Section 3. Finally, concluding remarks are given in Section 4.

II. Proposed Algorithm

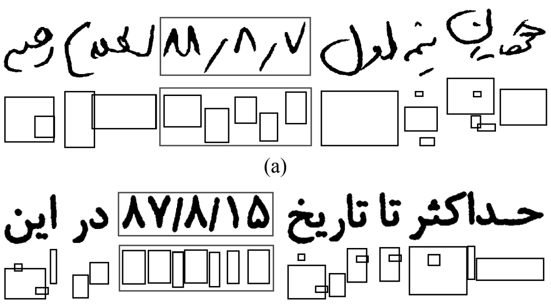
After text line detection [8], some features are extracted from all the connected components in each text line. Then these features are classified, and some probabilities are assigned to each connected component. Next, the syntactic analyzer localizes digit strings with desired syntaxes.

All the syntactic rules that should be verified for digit strings are based on the number of separators, found between digits in a digit string which is located in a text line between non-digits. So, the system should differentiate between separators (slashes in Farsi digit strings) and digits, in addition to differentiation between non-digits and digits. In handwritten digit strings, some digits are connected to each other and form double-digits. Recognition of these connected components is different from single-digits. So, the system should also differentiate between double-digits and single-digits. Therefore, we consider four classes for connected components in a text line: the class of Digits (D), Slashes (S), Double-Digits (DD), and Non-Digits (ND). Two handwritten and machine-printed Farsi text lines which contain digit strings with slashes and a double-digit are shown in Fig. 1.

A. Feature Extraction

The features for characterizing connected components should discriminate between four classes D, S, DD, and ND

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.152



(a)

(b)

Figure 1. Examples of handwritten (a), and machine-printed (b) Farsi text lines which contain digit strings and their corresponding minimum bounding rectangles.

in Farsi/ Arabic handwritten, machine-printed, and mixed text lines as much as possible. Features should be simple and easily extractable in comparison with features needed for recognition of connected components. These features must also be scale invariant, because the document images we are dealing with include text lines written or typed in different sizes. Some of these features are extracted from only one connected component. While some of them which are related to the position or size of connected components relative to each other, are extracted from more than one connected component.

The first feature is perpendicularity that specifies the angle between x-axis and major axis of the ellipse that has second order moments similar with connected component. This angle is achieved from the following relation [9]:

$$\phi = \left| \frac{1}{2} \arctan \frac{2\mu_{1,1}}{\mu_{2,0} - \mu_{0,2}} \right|,$$

In this relation, $\mu_{1,1}$, $\mu_{2,0}$, and $\mu_{0,2}$ are second order moments. This feature expresses more alignment of digits with vertical direction in comparison with non-digits. In addition, it can differentiate between digits and slashes.

The second feature is the maximum number of transitions. The number of transitions (from zero to one and from one to zero) in each row of connected component image is recorded and the maximum number is returned as feature value. This feature models more complexity of non-digit, and double-digit connected components in comparison with digits and slashes.

The next feature that rejects a major portion of non-digits is the number of connected components above or below the considered connected component. In the current text line, the columns which correspond to the width of the considered connected component are considered, and the feature value (*ncc*) is calculated as *ncc=ncc1+ncc2*, where

ncc1= the number of connected components that, the *x* coordinates of their centroid are between the columns which correspond to the width of the considered connected component.

ncc2= the number of connected components which, the *x* coordinate of centroid of considered connected component is located inside the constructive columns of those connected components widths.

ncc1 is effective to reject large non-digit connected components, and *ncc2* is effective to reject small non-digit connected components like dots and diacritics.

The next feature specifies how much the bottommost point of the considered connected component is aligned with the bottommost points of its neighbors. This feature is calculated as

$$\frac{bm_i - (bm_{i-1} + bm_{i+1})/2}{(H_{MBR_{i-1}} + H_{MBR_i} + H_{MBR_{i+1}})/3},$$

where bm_i , and H_{MBR_i} is the bottommost point, and height of the considered connected component, respectively. *i-1*, and *i+1* indexes represent the left side, and right side neighbors. Dividing by the average of heights makes the feature value scale invariant.

The next feature is solidity that equals the number of pixels in the connected component divided by the number of pixels in its convex hull. This feature is used to differentiate between slashes and digits in a digit string. The value of solidity for slashes is closer to 1 in comparison with digits.

In [1, and 4], nine features are presented for differentiation between digits and non-digits in Latin text lines. These features are extracted from heights, widths, and centroids of connected components minimum bounding rectangles. In Fig .1, the minimum bounding rectangles belonging to connected components of two handwritten and machine-printed Farsi text lines containing digit strings are shown. It can be seen that the height, width, and spacing of the minimum bounding rectangles belonging to consecutive connected components in digit strings, have a special regularity compared to the rest of the text line. These regularities in neighborhood of each connected component, is measured by considering its right and left neighbors.

Suppose MBR_i represents the minimum bounding rectangle of the considered connected component, and MBR_{i-1} and MBR_{i+1} are related to its left and right neighbors, respectively. Also, H_{MBR_i} , W_{MBR_i} , Cx_{MBR_i} , and Cy_{MBR_i} symbolize height, width, *x* coordinate, and *y* coordinate of centroid of connected components minimum bounding rectangle, respectively. Regularity/ irregularity in height, width, and spacing in the neighborhood of MBR_i can be measured by the following features [1]:

$$f_1 = \frac{H_{MBR_i}}{H_{MBR_{i-1}}},\ f_2 = \frac{H_{MBR_i}}{H_{MBR_{i+1}}},\ f_3 = \frac{W_{MBR_i}}{W_{MBR_{i-1}}},\ f_4 = \frac{W_{MBR_i}}{W_{MBR_{i+1}}},$$

$$f_5 = \frac{H_{MBR_i}}{W_{MBR_i}},\ f_6 = \frac{\left|Cx_{MBR_i} - Cx_{MBR_{i-1}}\right|}{W_{MBR_i}},\ f_7 = \frac{\left|Cx_{MBR_i} - Cx_{MBR_{i+1}}\right|}{W_{MBR_i}},$$

$$f_8 = \frac{\left|Cy_{MBR_i} - Cy_{MBR_{i-1}}\right|}{W_{MBR_i}},\ f_9 = \frac{\left|Cy_{MBR_i} - Cy_{MBR_{i+1}}\right|}{W_{MBR_i}},$$

f_1 , f_2 , f_3 and f_4 are used to measure regularity/ irregularity in height, and width of consecutive connected components. f_5 is the aspect ratio of considered connected component. f_6 and f_7 express regularity/ irregularity in spacing. Finally, f_8 and f_9 show that how much the centroids of two consecutive connected components are aligned with each other.

In addition to the last 9 features which are related to the minimum bounding rectangles of connected components, Chatelian et al. presented some other features in [2, and 3], for extracting digit strings from Latin documents which are: number of water reservoirs (metaphor to illustrate a valley in a component) [2], number of intersections with two horizontal and one vertical straight lines, number of end points [10], and number of holes [10].

Other than these structural features, we also consider one of the directional features that has a wide application in "recognition" of characters; a 100 component chaincode feature [11]. In [1, and 5], this feature is used to discriminate between alphabetic and numeric data in Latin scripts.

We presented a number of structural features to differentiate between members of digit strings and other connected components in Farsi/ Arabic text lines (the first 5 features). Moreover, structural features which are used in extracting digit strings from Latin text lines were introduced (the last 13 features). Now, by using a feature selection method applied to 18 structural features, we will obtain a feature vector that gives best results in classification of connected components to four classes D, S, DD, and ND, and also have lowest possible dimension. Forward selection algorithm [12] is used for feature selection. We have used training set for feature selection. This set includes 21800 labeled connected components in four classes D, S, DD, and, ND. Fig. 2, shows error rate for each of four classes and the overall error rate. As can be seen, at first, by increasing the number of features, error rate decreases, and then it increases again. Minimum overall error rate (7.9%) is achieved by 10 features. These 10 features are: bottommost point, number of connected components, perpendicularity, f_1, f_2 , f_3 , f_4 (height ratios and width ratios), solidity, maximum number of transitions, and f_5 (aspect ratio).

B. *Classification*

We use two different classifiers to classify 10 selected structural features and direction feature. First classifier is Support Vector Machine (SVM). We use an extended version of SVM that can assign a probability to each of four classes (D, S, DD, and ND) [13]. These probabilities will be useful in syntax verification stage by Hidden Markov Model (HMM). We empirically choose linear kernel [13] for SVM which gives best results in combination with both structural features and direction feature. Next classifier is MultiLayer Perceptron (MLP). Like SVM, MLP output can also be considered as probability estimate of each of four classes [14]. MLPs which are used in this experiment have the following characteristics. One input layer containing 10 neurons, or 100 neurons, when structural features or direction feature is used, respectively. One hidden layer containing 16 neurons, or 50 neurons, for structural features and direction feature, respectively. And an output layer that has four neurons as the number of classes. The activation function of each neuron is a hyperbolic tangent sigmoid transfer function. This multi-layer feed-forward neural network is trained with the iterative back-propagation algorithm.

C. *Syntax verification*

As it is mentioned before, we assume that there is no prior information about location of digit strings in text lines. In this part of digit string localization system, the outputs of connected component classifiers are analyzed, and digit strings with our desired syntax will be localized. In this analysis, we will pay special attention to the separators (slashes). Because characters which discriminate between different kinds of digit strings (date, phone number, account number, letter number, etc) are slashes. The system analyzes the sequences of digits and slashes, paying attention to the number of digits found between slashes.

Each of pattern recognition tools, SVM and HMM have some constraints in comparison with each other. In contrast to SVM, HMM has low discrimination ability in pattern classification problems. On the other hand, applying SVM in applications with variable length label sequence is difficult, because SVM has no temporal contextual memory [15]. In this study, we combine robust discriminating ability of SVM with dynamical modeling ability of HMM and try to localize digit strings in text lines. In addition to the SVM, performance of MLP is also studied in combination with HMM. In this work, we apply these tools with an architecture that is depicted in Fig. 3.

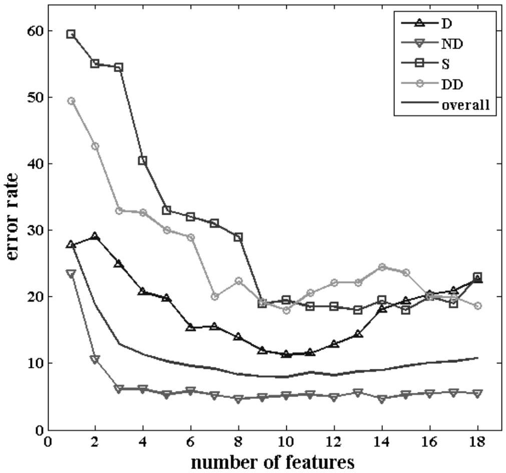


Figure 2. Feature selection through forward selection algorithm: the best classification result is achieved when 10 features are selected.

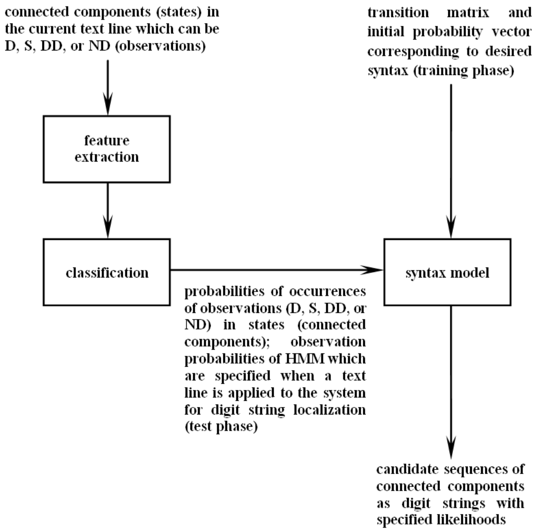

connected components (states) in the current text line which can be D, S, DD, or ND (observations)
transition matrix and initial probability vector corresponding to desired syntax (training phase)
feature extraction
classification
syntax model
probabilities of occurrences of observations (D, S, DD, or ND) in states (connected components); observation probabilities of HMM which are specified when a text line is applied to the system for digit string localization (test phase)
candidate sequences of connected components as digit strings with specified likelihoods

Figure 3. Architecture of probabilistic method for syntax verification.

In Farsi/ Arabic scripts, Alphabetic parts are written from right to left, while numeric parts are written from left to right. So, syntax models for Farsi/ Arabic digit strings should have left-right topology. For instance, syntax model for date digit string is shown in Fig. 4. The first state can be a non-digit (ND1), because in most cases, digit strings may occur in the middle of text line, after some non-digit connected components. HMM models can be constructed for any other digit string with any syntax we want to extract.

According to the presented architecture in Fig. 3, we consider each of connected components in a text line as a state. In each of these states, one of the observations D, S, DD, or ND may occur.

From labeled training connected components and through maximum likelihood estimation, a transition matrix and also an initial probability vector are constructed for each model, [16]. Another parameter that should be determined for HMM is observation probability which show the probability of occurrence of each of observations (D, S, DD, or ND) in each of states (connected components). These probabilities have been determined in the previous stages of digit string localization system (Fig. 3). As it is mentioned in the classification section, SVM and also MLP can provide four probabilities (probability of being in each of four classes) for each connected component. We apply these four-member sets of probabilities as probability of occurrence of each of observations in each of states. Thus the parameters of HMM (syntax model) are determined completely for each text line, transition matrix and initial probability vector in the training phase, and observation probabilities during test phase (applying the system). Having an HMM for each sequence of connected components in each text line of document image, we can localize our desired digit string.

For instance, if we want to localize date strings, the output of SVM for connected components in the current text line are applied to HMM that contains syntax of date. HMM returns a ranked list that corresponds to different locations (different sequences of connected components) in the current text line. The members of this list are ranked based on degree of similarity to date string (likelihood). According to these likelihoods in the ranked list, presence or absence of date string in the current text line, and its location can be determined. We only consider locations of a text line as candidates of digit strings which their corresponding likelihoods are higher than a threshold (that is empirically set to -10 in logarithm).

III. Evaluation and Experimental Results

We use 21800 connected components containing 5200 digits, 800 slashes, 800 double digits, and 15000 non-digits, for training classifiers; SVM, and MLP, and for feature selection. About 75% of these connected components are handwritten, and the rest of them are machine-printed. The values of some structural features like height ratios or bottommost point feature are extracted from two or more consecutive connected components. So training connected components as well, should be connected components in digit strings, non-digit strings, or mixed strings (not only single connected component images). To construct this training set, databases presented in [17, and 18] are used. For determining parameters of HMM (transition matrix and initial probability vector) that is used in syntax verification, and for evaluation of digit string localization system, 300 collected document images are used. There are handwritten, machine-printed, and mixed text lines containing digit strings in these document images. 100 document images are used for training HMM parameters, and 200 document images for evaluating system. Digit strings in these document images can be handwritten, or machine-printed with different syntaxes (date, phone number, letter number, zip code, etc). These digit strings may contain double digits or triple digits. There are 745 digit strings in these 300 document images. 268 of them are date strings, 179 of them are phone numbers, and the rests are digit string with other syntaxes. 58.2% of digit strings are handwritten and 41.8% of them are machineprinted. 13% of handwritten digit strings contain two or more touched connected components. Two criteria for evaluation of digit string localization system, recall and precision are defined as follows:

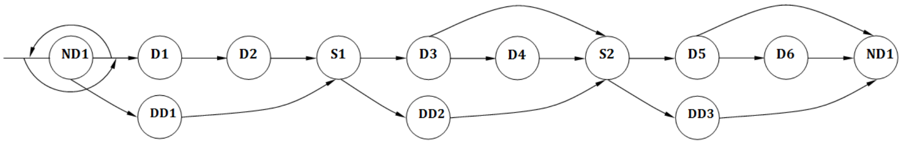


Figure 4. Syntax model fo date string.

$$recall = \frac{\#\,of\ digit\ strings\ completely\ localized}{\#\,of\ digit\ strings\ wich\ should\ be\ localized}$$

$$precision = \frac{\#\,of\ digit\ strings\ completely\ localized}{\#\,of\ localized\ strings\ as\ digit\ strings}$$

a digit string with a specific syntax is considered as completely localized, if all of its digits, slashes, and double digits are labeled as members of desired digit string.

To illustrate the performance of the system, top-n recall/ precision curves (recall and precision values when n first positions of ranked list are used) are shown in Fig. 5 and 6 for date and phone number localization, respectively. By proceeding from top-1 to top-5, in addition to finding our desired digit strings, the number of non-relevant detected strings (non-digit strings, or digit strings with non-relevant syntaxes) will increase. As a result, recall increases and precision decreases.

Generally, in all curves, precision values are acceptable, and this is firstly because of high ability of features in discrimination between members of digit strings (D, S, and DD) and non-digits. And the second reason is putting a threshold on logarithmic likelihoods in the ranked list. This threshold on likelihood values causes fewer non-digit strings to be wrongly localized as desired digit strings.

In most cases of Fig. 5, and 6, the SVM curves are higher than MLP curves, i.e. SVM provides higher recall and precision values. Also, by proceeding from top-1 to top-5, the decrement in precision of SVM is less than MLP. This is for the reason that SVM works better in differentiating connected components. In date string localization, especially in handwritten text lines, direction feature work better than structural features (because of slashes in date strings). While, in phone number localization, structural features and direction feature provide very close results. In most cases, the curves of structural features have higher precision values compared to direction features, because structural features reject non-digits better than direction feature.

IV. Concluding Remarks

In this paper, we have presented a system for localization of digit strings with a specific syntax in Farsi/ Arabic document images. Special characteristics of Farsi/ Arabic scripts were taken into consideration in the proposed methodology. A number of new structural features were presented for differentiation between connected components which are members of digit strings and other connected components in Farsi/ Arabic text lines. Effectiveness of these features is examined through different criteria. The number of final structural features is very few (10 features) in comparison with utilized features for recognition of connected components. Recall and precision values obtained by 10-component structural features are very close to the values obtained by 100-component direction features. In this paper, we only considered two syntaxes, i.e. date and phone number, but our method of syntax verification can be extended to any type of digit strings with any syntax, by modifying HMM models.

References

[1] C. Chatelain, L. Heutte, and T. Paquet, “A syntax-directed method for numerical field extraction using classifier combination,” in IWFHR04, pp.93-98, 2004.

[2] C. Chatelain, L. Heutte, and T. Paquet, “Discrimination between digits and outliers in handwritten documents applied to the extraction of numerical fields,” in IWFHR06, pp.475-480, 2006.

[3] C. Chatelain, L. Heutte, and T. Paquet, “A two-stage outlier rejection strategy for numerical field extraction in handwritten documents,” in ICPR06, pp.224-227, 2006.

[4] G. Koch, L. Heutte, and T. Paquet, “Automatic extraction of numerical sequences in handwritten incoming mail documents,” Pattern Recognition Letters 26, pp.1118-1127, 2005.

[5] L. Lam, Q. Xu, and C. Y. Suen, “Differentiation between alphabetic and numeric data using NN ensembles,” ICPR'02, pp.40-43 vol. 4, 2002.

[6] M. Ziaratban, and K. Faez, “Non-uniform slant estimation and correction for Farsi/Arabic handwritten words,” IJDAR, Vol.12 Issue 4, 2009.

[7] A. Abedi, K. Faez, and S. Mozaffari, “Detecting and Recognizing Numerical Strings in Farsi Document Images,” IVCNZ’09, pp.403-408, 2009.

[8] G. Louloudis, K. Halatsis, B. Gatos, and I. Pratikakis, “Text line detection in handwritten documents,” Pattern Recognition 41, pp.3758–3772, 2008.

[9] B. Jahne, Digital Image Processing, 5th revised and extended edition, Springer, 2002.

[10] S. Marchand-Maillet, and Y. M. Sharaiha, Binary Digital Image Processing, a Discrete Approach, Academic Press, 2000.

[11] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, “Handwritten digit recognition: benchmarking of state-of-the-art techniques,” Pattern Recognition 36, pp.2271–2285, 2003.

[12] Y. Zheng, H. Li, and D. Doermann, “Machine Printed Text and Handwriting Identification in Noisy Document Images,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 3, pp.337-352, 2004.

[13] T. F. Wu, C. J. Lin, and R. C. Weng, “Probability estimates for multi-class classification by pair wise coupling,” Journal of Machine Learning Research 5, pp.975–1005, 2004.

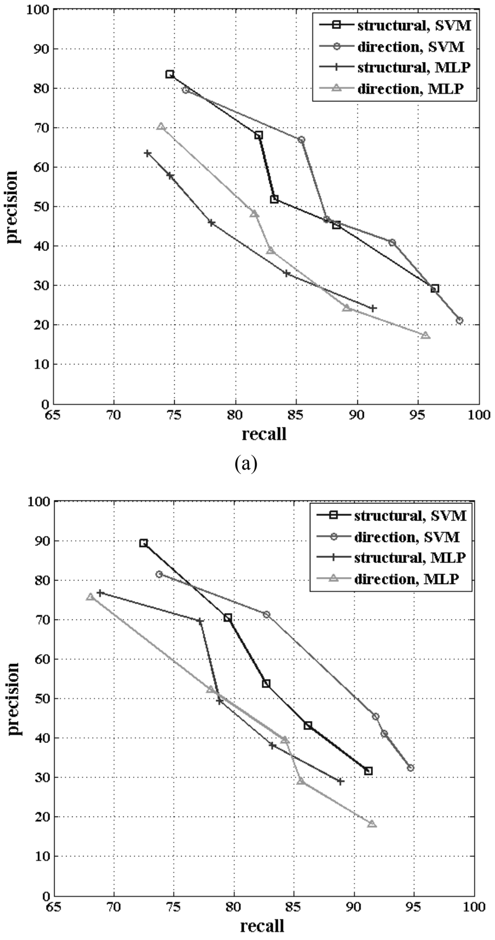
[14] Bridle, J. S “Probabilistic interpretation of feed forward classification network outputs, with relationships to statistical pattern recognition,” in Neurocomputing: Algorithms, Architectures and Applications, F. F. Soulie and J. Herault, Eds., NATO ASI, pp.227-236, 1990.

[15] B. Q. Huang, C. J. Du, Y. B. Zhang and M. T. Kechadi, “A hybrid HMM-SVM method for online handwriting symbol recognition,” in isda'06, vol. 1, pp.887-891, 2006.

[16] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” in readings in speech recognition, pp.267–296, 1990.

[17] M. Ziaratban, K. Faez, and F. Bagheri, “FHT: an unconstraint Farsi handwritten text database,” in ICDAR09, pp.281–285, 2009.

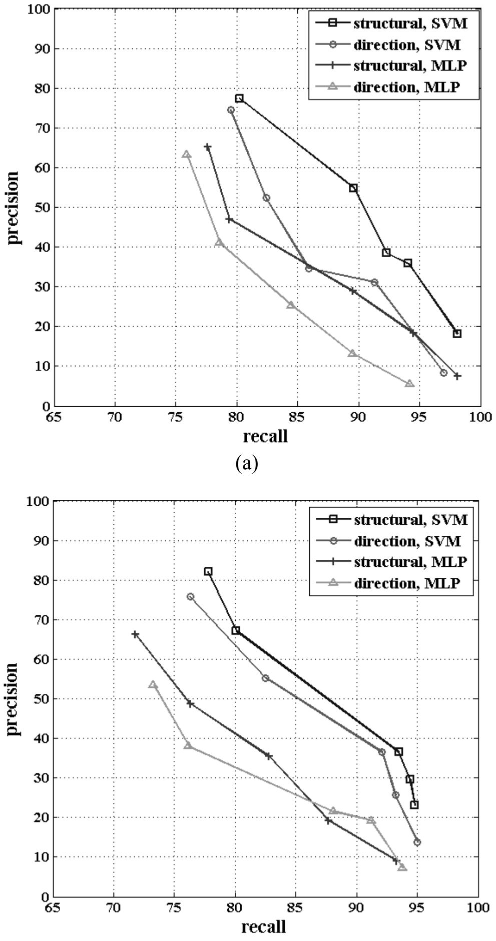
[18] F. Solimanpour, J. Sadri, and C. Y. Suen, “Standard databases for recognition of handwritten digits, numerical strings, legal amounts, letters and dates in Farsi language,” IWFHR06, pp.743–751, 2006.



(a)

(b)

Figure 5. Recall/ Precision curves for date string localization in (a) machine-printed and (b) handwritten text lines



(a)

(b)

Figure 6. Recall/ Precision curves for phone number string localization in (a) machine-printed and (b) handwritten text lines