On-Line Cursive Handwriting Recognition: A Survey of Methods and Performances

[1]Dzulkifli Mohamad*, [2]Muhammad Faisal Zafar*, and [3]Razib M. Othman*

*Faculty of Computer Science & Information Systems, Universiti Teknologi Malaysia (UTM), 81310 Skudai, Johor, Malaysia.

[1]dzul@fsksm.utm.my, [2]faisal@gmm.fsksm.utm.my, [3]razib@fsksm.utm.my

ABSTRACT

This paper presents a detailed review of prior techniques and applications for on-line cursive handwriting recognition. This survey is divided into two parts, the first one dealing with the review of main approaches used in character recognition, since most have been used in cursive script recognition as well. The second one is focusing on the prior techniques for on-line cursive handwriting recognition and their applications.

Key words: Character recognition Cursive script recognition Recognition strategies

1. Introduction

Handwriting has long been studied by numerous disciplines for various different aspects and purposes, and they include experimental psychology, neuroscience, engineering, computer science, anthropology, education, forensic science, etc (Plamondon, 1993; Plamondon and Leedham, 1990; Simner *et al.*, 1994; Simner *et al.*, 1996; Van Galen and Morasso, 1998; Van Galen and Stelmach,1993; Wan *et al.*, 1991). It is a natural means of communication which nearly every one learns at an early age. Thus it provides a means of data entry for computers in which user needs virtually no training. In this regards, continuous efforts are being made to develop algorithms and techniques to enable computers to recognize almost every type of handwriting.

There is extensive work in the field of handwriting recognition, and a number of reviews exist. General methodologies in pattern recognition and image analysis are presented in Mantas (1986). Character recognition is reviewed in Suen *et al.*, (1980), Govindan and Shivaprasad (1990), Steinherz et al. (1999), Alessandro (2002), Koerich *et al.* (2003) and Bortolozzi *et al.* (2005) for off-line recognition, and in Nouboud and Plamondon (1990), Plamondon and Srihari (2000) and Leedham et al. (2002) for on-line recognition. Segmentation and contextual analysis techniques are outlined in Elliman and Land caster (1990) and Casey and Lecolinet (1996). Cursive writing recognition is summarized in Tappert, Suen and Wakahara (1990), Lecolinet and Baret (1994), Gaofeng (1998), and Bunke (2003). Work in signature (Justino *et al.*, 2002) and user identification is reviewed in Plamondon and Lorette (1989).

Below, we review in detail prior techniques and applications for on-line English cursive handwriting recognition. We first review the main approaches used in character recognition, since most have been used in cursive script recognition as well. We then present prior techniques for on-line cursive handwriting recognition and their applications.

2. Character Recognition

A number of methods can be applied to the on-line recognition of characters. Some methods rely on prior analysis of the characters of the alphabet-for example, features such as ascenders, descenders and closures are alphabet specific. Other methods, such as most signal processing based approaches in curve matching are essentially independent of the alphabet. *Feature analysis* views each character to be a set of features that are based on the static or dynamic properties of the characters. The features can be binary or non-binary.

Frishkopf and Harmon (1961) proposed a method based on a decision tree using binary features such as descender or not descender, and dot or not dot. For example, for lower case English letters, the presence of a descender reduces the choices to the letter f, g, j, p, q, y and z. Then if a dot is present, the only choice is the letter j. The disadvantage of this method is that it does not produce alternative character choices, which could be used to increase the robustness of the system.

Fujisaka, Nagai and Hida.ka (1971) report a non-binary feature method. They used linear-discriminant functions to divide a feature space into decision regions, and then used pattern recognition techniques to identify characters.

Methods using *sequence of zones* and *extrema* rely primarily on dynamic information. A sequence of coded zones represents a character. The zones are specified by dividing up a rectangle that is superimposed on the written character. The sequence of zones traversed by the pen tip is determined. This sequence is compared with the zone sequences of a database, and is assigned as the character of the closest sequence of the dictionary. For details, see Brown (1964), and Pobgee and Parks (1972). A similar method describes a character as a sequence of local extrema. (usually left, right, up, down). Such sequences are called *chain codes* (Ward and Kuklinski,1988).

Curve matching is a signal processing method. Curves of unknown characters are matched against those of prototype characters. The curves matched are usually functions of time, such as coordinates, angular variations of the pen path., and curvatures. For details, see Ichikawa and Yoshida (1974) and Impedovo (1984).

Curve matching becomes equivalent to pattern matching in feature space when a one-to-one correspondence can be found between the unknown and the reference patterns. A nonlinear matching method (Tappert, 1982;1984) called *elastic matching* has been successful for many sequence comparison problems. Because elastic matching is computationally intensive, the prototypes are often pruned prior to matching in order to reduce the number of matches (Kurtz berg and Tappert, 1982). Elastic matching is often employed in online character recognition for establishing sample point correspondence between input and reference patterns (Hiroto *et al*, 2005). Dynamic programming (DP) matching is a classic elastic matching technique (Ikeda et al.,1981; Yoshida and Sakoe, 1982; Burr,1983) and still very popular (Hiroto *et al*, 2005).

Methods based on *stroke codes* classify subparts, called strokes, of a character and identify the character from the sequence of classified strokes. Examples of stroke codes can be found in Guberman and Rozentsveig (1976) and Schomaker (1993). Guberman and Rozentsveig (1976) identify ten basic strokes that form all letters of the alphabet.

Much research have been conducted to mitigate the stroke order variation, stroke number variation, and stroke deformation (Yoshida and Sakoe, 1982; Odaka *et al*., 1986; Ishigaki and Morishita , 1988; Wakahara *et al*., 1996; Tappert *et al*., 1990). On-line recognition, in contrast with off-line recognition, has the important advantage of being able to use stroke order and connection information, because the character pattern is expressed by ordered time sequences. Two approaches based on the stroke order and stroke connection information are considered to realize online character recognition. The first approach actively uses the stroke order and connection information; the second takes into account the inevitable changes in stroke order and connection from person to person, and hence realizes the "*free stroke order*" and "*free stroke number*". To build up the system for stroke order and number free recognition, the algorithm requires correct performance of stroke correspondence between input pattern and reference pattern for higher recognition performance (Wakahara *et al*., 1996). In these algorithms, the enormous percentage of stroke correspondences that do not actually occur are also carried out. In his research, Jungpil Shin (2002) focuses on the style analysis of stroke order variation and connection between strokes. The large improvement on both computational time and recognition accuracy are demonstrated by experiments.

Analysis by synthesis uses strokes and rules for connecting them to build characters. Characters generated from the inventory of strokes are the ideal representation of the characters that are to be recognized. An approximation to handwritten characters is achieved by specifying these strokes by mathematical models that describe the motion of the pen tip as a function of time. Handwritten words are divided into strokes, the identified strokes are classified using parameters of the model, and letter sequences and words are recognized (Mermelstein and Eden,1964). This approach suffers from the variability of the handwriting. In particular a model used for the recognition may not reflect the particular handwriting style.

Fisher discriminant analysis (FDA) (Duda and Peter, 1993; Ripley, 1996) is an improvement of more conventional linear projection methods like Principal Component Analysis (PCA). A notable weakness of PCA is that the projection it performs scatters data in the projection space without considering of the class specific distribution structures. In contrast, the projection matrix of FDA is constructed by taking the class specific regularities into account. More specifically the FDA tries to maximize the between-class scatter while minimizing the within-class scatter in the projection space. The result is a clearer class boundaries and thus easier separation between the classes. While the principle has been known for decades, practical application dealing with high dimensional representation space had not been tried until recently when the face recognition community used it successfully (Peter et al., 1997).

A substantial advantage of using linear techniques like FDA is that the training is much faster and requires relatively smaller amount of training data compared with the more popular methods like neural networks and hidden Markov models. Therefore it has a potential to make user tailored training feasible (Jong, 2001).

3. Cursive Script Recognition

Cursive script is a common form of handwriting. Cursive script recognition is difficult because several characters are typically connected together. The major recognition approaches are word-based, segmentation-based, and subsequence-based segmentation. These three approaches are described below.

3.1. Word-based recognition

Word-based recognition identifies the entire word without any attempt to segment or locate individual characters. The approaches used in word-based recognition are usually similar to those of character recognition. These approaches avoid the letter segmentation problem. However, the possible variability of the way whole words are written, is much higher than in the case of single characters. This may make the recognition task very difficult. The problem is less evident when the number of words is small (Gaofeng, 1998).

Farag (1979) uses chain codes and Markov chains for word-based recognition. A recognition rate of 100% is reported for ten cursive words and one writer.

Brown and Ganapathy (1983) use feature vectors and an estimate of the length of the word to represent the word characteristics. The recognition is based on the extracted feature vectors using the K-nearest neighbor method. In a test, the recognition domain consisted of 43 words. Three users wrote ten samples, each containing 22 words. The recognizer was trained on data of one of the users and tested on data of the other two. Recognition rates ranged from 63.2% to 80.3%.

Powalka., Sherkat and Whit row (1994) report a word-based recognizer which uses a very limited set of features consisting of a sequence of ascenders and descenders and an estimate of word length. A fuzzy logic based matching algorithm is used. Average recognition rates obtained for a 200 word lexicon were 40% and 60.6% for the first choice and the top five alternatives, respectively. Handwriting of 18 users was evaluated, each writing 200 words.

Bramall and Higgins (1995) report cursive recognition system based on the human reading process. The system uses the blackboard paradigm of artificial intelligence. It initially uses easily extracted features to reduce a large lexicon to a small list of candidate words. Later stages use increasingly sophisticated knowledge sources, based on a diverse set of AI paradigms and other pattern recognition techniques, to determine and subsequently refine a confidence value for each candidate. The word with the highest confidence value is the output of the system. The system *was* trained on 3,600 words from 34 users. The test was performed on 23 randomly selected of the 34 users using a 20,000 lexicon. The recognition rate is from 29% to 72%.

Jong (2001) developed an on-line handwriting recognition system which integrates local bottom-up constructs with a global top-down measure into a modular recognition engine. The bottom-up process uses local point features for hypothesizing character segmentations and the top-down part performs shape matching for evaluating the segmentations. The shape comparison, called Fisher segmental matching, is based on Fisher's linear discriminant analysis. The component character recognizer of the system uses two kinds of Fisher matching based on different representations and combines the information to form the multiple experts paradigm. Along with an efficient ligature modeling, the segmentations and their character recognition scores are integrated into a recognition engine termed Hypotheses Propagation Network (HPN), which runs a variant of topological sort algorithm of graph search. The HPN improves (73% to 88%) on the conventional Hidden Markov Model and the Viterbi search by using the more robust mean-based scores for word level hypotheses and keeping multiple predecessors during the search. He also implemented a geometric context modeling termed Visual Bigram Modeling that improves the accuracy (from 88% to 93%)of the system's performance by taking the geometric constraints into account, in which the component characters in a word can be formed in relation with the neighboring characters. Four different settings of 100 words were test with a lexicon of 450 words.

Anja *et al*. (2002) describe a writer-independent on-line handwriting recognition system which is comparing the effectiveness of several confidence measures. Their recognition system for single German words is based on Hidden Markov Models (HMMs) using a dictionary. They compare the ratio of rejected words to misrecognized words using four different confidence measures: One depends on the frame-normalized likelihood, the second on a garbage model, the third on a two-best list and the fourth on an unconstrained character recognition.

They use a large on-line handwriting database of several writers consists of cursive script samples of 166 different writers, all writing several words or sentences on a digitizing surface. The training of the writer independent system is performed using about 24400 words of 145 writers. Testing is carried out with 2071 words of 21 different writers (about 100 words per writer). The recognition results are determined using an increasing threshold τ and using the baseline system without rejection a word recognition rate of 87.0% (1801 words are recognized correctly) is achieved testing the entire test-set of 2071 words. The presented results refer to a single word recognition rate using a dictionary of about 2200 German words. Table 1 below shows the summery of performances of word-based recognition. For each work cited in the first column, the second column shows the number of writes involved. The table reports the sizes of lexicon and database (DB) used in the experiments and performance (in terms of words correctly recognized) obtained.

Table 1: Summery of statistics cited for word-based recognition

Author	# of writers	Lexicon size	DB size	Performance
Farag (1979)	1	10		100%
Brown and Ganapathy(1983)	3	43	660	63.2% – 80.3%
Powalka *et al.*	18	200	3600	40% – 60.6%

(1994)				
Bramall and Higgins (1995)	34	20,000	3600	29% – 72%
Jong Oh(2001)	Not mentioned	450	100	88% - 93%
Anja *et al*. (2002)	166	2200	24400	87%

3.2 Segmentation-based recognition

Segmentation-based recognition segments each word into its component characters and employs a recognition technique to identify each letter. Unfortunately, the letter segmentation points in cursive script can only be identified when the correct character sequence is known. On the other hand, recognition of characters can only be done successfully when the segmentation is correct. Relaxed segmentation criteria are commonly used whereby a large number of potential segmentation points are generated. A recognition system examines all possible combinations of the segmentation points. The recognition is accomplished on a best-match basis (Gaofeng, 1998).

Higgins and Ford (1992) segment on-line data at intersections, cusps, points of inflection, and endpoints. A number of extra rules are applied to eliminate accidental segmentation points. A database of prototypes is used which contains allographs-topological variations of the same letter-for every variety of letter and letter join recognized by the system. A. further personal database is also used. A tree-based lexicon is employed to hypothesize possible words. Recognition rates between 75% and 91% are reported for user-independent and user-adaptive systems, respectively. The paper says that a large lexicon was used, but does not provide details about the lexicon or the test data.

Systems described in Morassa *et al.* (1993) and Schomaker (1993) share some features. Both systems segment the on-line data at the points of minima in the tangential pen tip velocity. Both recognizers use Kohonen self-organizing networks for the classification of the obtained segments and both employ the same tree-based lexical lookup technique proposed by Wells *et al.* (1990) using a 4000 word lexicon. The recognizer presented by Morasso *et al.* (l993) uses a number of neural networks, each of them trained to recognize letters composed of different numbers of segments. Schomaker (1993) employs a. single neural network so that the clustering capability of the Kohonen network is used over all possible combinations of strokes.

Schomaker (1993) presents two systems, segmentation-based and letter-based. The segmentation-based system attempts to recognize individual segments, which are then combined into allographs to form letters. The letter-based system attempts to recognize letters directly. Both systems are user-dependent. Recognition rates are reported for three users, each of them writing around 100 words. The letter-based system achieved recognition rates between 32% and 73% for the top five alternatives. The recognition rates of the segmentation-based system are reported again for the top five alternatives are reported to be 36-47%, 6884%, and 80-92% when one, two, and three result alternatives, respectively, were obtained.

The letter-based system of Morasso *et al.* (1993) achieved between 78% and 93% recognition for the top five alternatives.

Flann (1994) uses a lexicon of 10,748 words. The original data are encoded as a sequence of uniform segment descriptions. Segment boundaries are determined at points of zero vertical velocity and at the beginning and end of each stroke. Identified segments are processed by six forward neural networks. Each network is designed to recognize letters of different size. Words are recognized by searching all possible segmentations. Flann (1994) reports recognition rates between 85.5% and 98.3% for user-dependent recognition and from 68.9% to 94.8% for user-independent recognition. Tests were performed with ten users, each writing 100 words. User-independent tests were performed using the data of nine users for training and using the data of the tenth user for testing.

Powalka *et al.(*1994) use several lexicons containing up to 15,012 words. Recognition is based on a multiple interactive segmentation algorithm and multi-scale letter recognizer. An average user-independent recognition rate of 67% is reported for six users writing 200 words each and a 15,012 word lexicon.

Manke, Finke and Waibel (1995) report a segmentation-based system using time delay neural network (TDNN) and hidden Markov model (HMM) approaches. The system was trained on 5,700 words from 80 users. Using a. 20,000 word lexicon, the average recognition rate was 91.4% from an independent set of 40 users.

Connell and Jain (2002) present an approach to writer-adaptation. They demonstrate the feasibility of the approach using HMMs trained on a combination of discretely written lowercase characters and digits, and cursively written lowercase characters that have been handsegmented from word-level data. The word recognition experiments were conducted using a training data of 122,410 examples from 93 character classes. The data was collected from 100 writers. Models were tested using a test set of 571-614 words written by each of the eight writers, from a lexicon of 483 different words. The overall accuracies reported are 83.7% to 92.8%. Table 2 summarizes the statistics of the above cite works.

Table 2: Summery of statistics cited for segmentation-based recognition

Author	# of writers	Lexicon size	DB size	Performance
Higgins and Ford (1992)	1	10	Not mentioned	100%
Schomaker (1993)	3	4000	300	80.92%
Morassa *et al.* (1993)	3	4000	300	78% – 93%
Powalka *et al.* (1994)	6	15,012	1200	67%
Flann (1994) (user-depdt) (user-indepdt)	9 10	10,748	1000	85.5%– 98.3% 68.9%– 94.8%
Manke *et al.* (1995) (user-indepdt)	80	20,000	5700	91.4%
Connell and Jain (2002)	100	483	122,410	83.7% – 90.8%

3.3 Subsequence-based segmentation

Subsequence-based segmentation approaches adopt an intermediate position between the above approaches. Entire words are processed. An attempt is made to locate some smaller entities such as letters/substrings within the word. The combination of letters/substrings that best matches the processed word constitutes the solution. Segmentation in this approach is secondary and becomes available only after the recognition is performed. The number of entities to locate and recognize is limited and does not depend On the lexicon size. Thus, subsequence-based segmentation can use larger lexicons than the word-based recognition, while still avoiding any direct segmentation.

Berthod and Ahyan (1980) use features including local extrema, intersection and inflection points, cusps, and curvatures within a word. The sequence of extracted features is parsed to generate all possible sequences of features which match the ones stored in the database. Some linking features are allowed to account for ligatures (connecting parts of characters). The analysis employs pruning of a tree-based lexicon. The database stores sequences of features representing single letters. Recognition results reach 87% for two users writing 100 words each. The letter databases are user-dependent. The cited paper does not provide the size of the lexicon.

Tappert (1982) employs dynamic programming and compares the input data with letter prototypes point by point. Each data point is characterized by two parameters: the vertical position and the tangent of the pen path at the current point. Experiments involved careful writing and user-dependent databases of letter prototypes. An average recognition rate of 97% is reported for three users writing 30 words each. It should be noted that Tappert stresses the need for careful writing and user-defined prototypes. On the other hand, no lexicon is used in this system.

Oulhadj *et al*. (1990) introduce a prediction verification strategy for the recognition of cursive handwriting. Words are encoded into a chain code using sixteen directions and then converted into four directions in order to reduce the variability of the data. The resulting chain code is parsed from left to right and then searched for all possible patterns representing known letters. The process is recursively repeated after any letter alternative is located. A lexicon is used to prune letter candidates disallowed in the current context. The best path through the chain code which produces a valid word is taken as the recognition result. Tests were performed using a 110 word lexicon and five users, each writing 50 words. Handwriting of one user was used to create letter prototypes. Reported recognition rates are 40% and 94% before and after training, respectively.

Rao (1995) introduces a synthesis approach for cursive script recognition. The individual characters are characterized as a feature matrix using local extrema, curvatures, and slopes. A script word can be generated using individual characters and transaction strokes. A database stores all possible generated scripts for each word of a. lexicon. A written word is analyzed and its extracted feature matrix is compared with the prototypes of the database. Recognition is accomplished on a. best match basis. Tests were performed using a 63 word lexicon and ten users. An average recognition rate of 91% is reported.

Schenkel *et al*. (1995) report a recognition based segmentation system using TDNN (time delay neural network) and HMM (hidden Markov model) approaches. The observation probabilities of the system are estimated by the TDNN, trained with a back propagation algorithm. A four state HMM is used to represent characters. The system was trained on 20,000 words from 59 users, using a 25,000 word lexicon. The average recognition rate was 80% for users that did not participate during the training.

Seni *et al*. (1996) introduce a system of cursive script recognition. The system first uses a filter model based on simple character features to reduce a large reference dictionary to a more manageable size. Explicit segmentation of handwritten words into characters is avoided by sequentially presenting the word to a TDNN based recognizer. The outputs of the recognizer are converted into a string of characters that is matched against the reduced lexicon using an extended Damerau-Levenstein function. The system was trained on 2,443 words from 55 users. Reported average recognition rates for a 21,000 word lexicon are 97.9% and 82.4% for the top five alternatives on user-dependent and user-independent tests, respectively. Table 3 presents the summery of the above cite works.

Table 3: Summery of statistics cited for subsequence-based segmentation recognition

Author	# of writers	Lexicon size	DB size	Performance
Berthod and Ahyan (1980)	2	Not mentioned	200	87%
Tappert (1982)	3	Not mentioned	330	97%
Oulhadj *et al*. (1990)	5	110	250	40% – 94%
Rao(1995)	10	63		91%
Schenkel *et al*. (1995)	59	25,000	20,000	80%
Seni *et al*. (1996) (user-dependent) (user-independent)	55	21,000	2443	97.9% 82.4%

4. Recognition Strategies

Numerous techniques for handwriting recognition have been investigated based on four general approaches of pattern recognition, as suggested by Jain *et al*. (2000): template matching, statistical techniques, structural techniques, and neural networks. Such approaches are neither necessary independent nor disjointed from each other. Occasionally, a technique in one approach can also be considered to be a member of other approaches.

4.1 Template matching

Template matching operations determine the degree of similarity between two vectors (groups of pixels, shapes, curvatures, etc) in the feature space. Matching techniques can be grouped into three classes: direct matching (Gader *et al*., 1991), deformable templates and elastic matching (Dimauro *et al*., 1997), and relaxation matching (Xie and Suk, 1988). In conventional elastic matching-based recognizers, a matching cost obtained as a by-product of their matching optimization procedure is directly used as a discriminant function. Although dynamic programming (DP) matching and other elastic matching based recognizers generally perform well, they often suffer from misrecognitions due to *overfitting*, which is the phenomenon that the distance between the reference pattern of an incorrect category and an input pattern is underestimated by unnatural matching. One possible remedy against the overfitting problem is the incorporation of probabilistic/statistical techniques. Statistical DP (Bahlmann and Burkhardt, 2004)and hidden Markov model (HMM) (Nag *et al* , 1986; Hu *et al*, 1996; Nakai *et al*., 2001) are probabilistic extensions of DP and can avoid the over-fitting by regulating the probability of feature values. Although they often outperform naive DP techniques, they cannot exclude all overfittings because their Markovian property allows to regulate only a "local" and "individual" probability of the feature value at each sample point. Thus, those techniques cannot regulate a "global" and "mutual" probability of whole sample points (Hiroto *et al*, 2005).

4.2 Statistical techniques

Statistical techniques are concerned with statistical decision functions and a set of optimal criteria, which determine the probability of the observed pattern belonging to a certain class. In statistical representation, the input pattern is described by a feature vector, while the model database (also called parameter database in this case) contains the classification parameters. The statistical scheme is receiving increasing attention in recent years (Liu et al., 2004). Statistical techniques use concepts from statistical decision theory to establish decision boundaries between pattern classes (Jain *et al*., 2000; Vuurpijl and Schomaker, 2000).
Several popular handwriting recognition approaches belong to this domain:

- The k-Nearest-Neighbor (Favata, 2001) rule is a popular non-parametric recognition method, where a posteriori probability is estimated from the frequency of nearest neighbors of the unknown pattern. Compelling recognition results for handwriting recognition have been reported using this approach (Guillevic and Suen, 1995). The drawback of this method is the high computational cost when the classification is conducted. To surpass such a problem some researchers have proposed faster k-NN methods. A comparison of fast nearest neighbor classifiers for handwriting recognition is given in (Mico, 1999).
- The Bayesian classifier assigns a pattern to a class with the maximum a posteriori probability. Class prototypes are used in the training stage to estimate the classconditional probability density function for a feature vector (Duda *et al*., 2001).
- The polynomial discriminant classifier assigns a pattern to a class with the maximum discriminant value which is computed by a polynomial in the components of a feature vector. The class models are implicitly represented by the coeficients in the polynomial (Schurmann, 1996).
- Hidden Markov Model (HMM) is a doubly stochastic process, with an underlying stochastic process that is not observable (hence the word hidden), but can be observed through another stochastic process that produces the sequence of observations (Rabiner, 1989). An HMM is called discrete if the observations are naturally discrete or quantized vectors from a codebook or continuous if these observations are continuous. HMMs have been proven to be one of the most powerful tools for modeling speech and later on a wide variety of other real-world signals. These probabilistic models offer many desirable properties for modeling characters or words. One of the most important properties is the existence of efficient algorithms to automatically train the models without any need of labeling presegmented data. HMMs have been extensively applied to handwritten word recognition (Yacoubi *et al*., 1999; Kundu and Chun, 2002; Senior, 2002) and their applications to handwritten digit recognition (Cai and Liu, 1999; Britto *et al*., 2001) have been growing. The literature presents two basic approaches for handwriting recognition using HMM: Model-Discriminant HMM and Path-Discriminant HMM. In the former, a model is constructed for each class (word, character, or segmentation unit) in

the training phase. In the latter, a single HMM is constructed for the whole language or context. The performances of these two approaches are compared in various experiments by utilizing different lexicon sizes (Kundu and Chun, 2002).

- Fuzzy set reasoning is a technique that employs fuzzy set elements to describe the similarities between the features of the characters. Fuzzy set elements give more realistic results when there is not a priori knowledge about the data, and therefore, the probabilities cannot be calculated. The literature reports different approaches based on this technique such as fuzzy graphs (Abuhaiba and Ahmed, 1993), fuzzy rules (Gader *et al*., 1995), and linguistic fuzzy (Lazzerini and Marcelloni, 2000).
- Support Vector Machine (SVM) is based on the statistical learning theory (Vapnik, 1995; Wang *et al*, 2000) and quadratic programming optimization. An SVM is basically a binary classifier and multiple SVMs can be combined to form a system for multi-class classification. In the past few years, SVM has received increasing attention in the community of machine learning due to its excellent generalization performance. More recently, some SVM classification systems have been developed for handwriting digit recognition, and some promising results have been reported in Ayat *et al*. (2002), Byun *et al*. (2002) and Oliveira *et al*. (2004).

4.3 Structural Techniques

In structural techniques the characters are represented as unions of structural primitives. It is assumed that the character primitives extracted from handwriting are quantifiable, and one can find the relationship among them. Basically, structural methods can be categorized into two classes: grammatical methods (Shridhar *et al*., 1986) and graphical methods (Kim H and Kim J, 1998).

4.4 Neural Network Techniques

A Neural Network (NN) is defined as a computing structure consisting of a massively parallel interconnection of adaptative "neural" processors. The main advantages of neural networks lies in the ability to be trained automatically from examples, good performance with noisy data, possible parallel implementation, and efficient tools for learning large databases. NNs have been widely used in this field and promising results have been achieved, especially in handwriting digit recognition. The most widely studied and used neural network is the Multi-Layer Perceptron (MLP) (Bishop, 1995). Such an architecture trained with back-propagation (LeCun *et al*., 1998a) is among the most popular and versatile forms of neural network classifiers and is also among the most frequently used traditional classifiers for handwriting recognition. See (Zhang, 2000) for a review. Other architectures include Convolutional Network (CN) (LeCun *et al*., 1998b), Self-Organized Maps (SOM) (Zhang *et al*., 1999), Radial Basis Function (RBF) (Bishop, 1995), Space Displacement Neural Network (SDNN) (Matan *et al*., 1992), Time Delay Neural Network (TDNN) (Lethelier and Gill, 1995), Quantum Neural Network (QNN) (Zhou, 1999), and Hopfield Neural Network (HNN) (Ling *et al*., 1997).

Another strategy that can increase the recognition rate in a relatively easy way with a small additional cost is through the use of verification. Such a scheme consists of refining the top few candidates in order to enhance the recognition rate economically. Such a kind of scheme has been successfully applied to handwriting recognition in (Zhou, 1999; Britto, 2001; Koerich, 2002; Oliveira, 2002).

5. Conclusions

The above review indicates that there are many recognition techniques available for handwriting recognition systems. A description of major approaches is given as well as an overview of the applications presented in the literature. All of them have their own advantages and drawbacks. In the recent years, many researchers have combined such techniques in order to improve the recognition results. The idea is not rely on a single decision making scheme.

The performances of the systems presented in the literature were finally reported showing, in some cases, recognition rates sufficient for real world applications. On-line Cursive Handwriting Recognition is a still evolving field. The important results related to specific application domains can not be considered conclusive. The open issues to achieve a general Cursive Handwriting Recognition system are still many and important.

It is concluded, regarding major recognition approaches, that segmentation-based systems achieve higher recognition rates than the word-based recognition approaches even though significantly larger lexicons are used. Similarly the subsequence-based segmentation systems provide better recognition rates than the word-based recognition systems. This is due to two facts User-dependent databases are used in the described systems, and the symbols to be matched have potentially decreased variability.

References

[1] Abuhaiba S. I., and Ahmed P. (1993). A fuzzy graph theoretic approach to recognize the totally unconstrained handwritten numerals. *Pattern Recognition*, 26(9):1335-1350.

[2] Alessandro Vinciarelli (2002). A survey on off-line Cursive Word Recognition. *Pattern Recognition*, 35 (2002) 1433–1446.

[3] Anja Brakensiek, Andreas Kosmala, and Gerhard Rigoll, (2001). Writer Adaptation for Online handwriting Recognition. *B. Radig and S. Florczyk (Eds.): DAGM 2001, LNCS 2191, 32–37, 2001*. @ Springer-Verlag Berlin Heidelberg.

[4] Anja Brakensiek, Andreas Kosmala, and Gerhard Rigoll (2002). Evaluation of Confidence Measures for On-Line Handwriting Recognition. *L. Van Gool (Ed.): DAGM 2002, LNCS 2449*, pp. 507–514, Springer- Berlin Heidelberg 2002.

[5] Ayat N. E., Cheriet M., and Suen C. Y. (2002). Optimization of the SVM kernels using an empirical error minimization scheme. *In Proc. of the International Workshop on Pattern Recognition with Support Vector Machine*, Niagara Falls-Canada, August, pp. 354-369.

[6] Bahlmann C. and Burkhardt H.(2004). The writer independent online handwriting recognition system flog on hand and cluster generative statistical dynamic time warping. *IEEE Trans. PAMI*, 26(3): 299-310, 2004.

[7] Berthod, M., a.nd Ahyan, S. On-Line Cursive Script Recognition: A Structural Approach With Learning. *Fifth International Conference on Pattern Recognition,* 1980, 723-725.

[8] Bishop M. (1995). *Neural Networks for Pattern Recognition*. Oxford Univ. Press, Oxford-U.K.

[9] Bortolozzi, F., Britto Jr,. A., Oliveira, L. S. and Morita, M., (2005). *Recent Advances in Handwriting Recognition*. In Umapada Pal et al editors, Document Analysis, 1-31.

[10] Bramall, P. E., and Higgins, C. A. (1995). A Cursive Script Recognition System Based on Human Reading Models, *Machine Vision and Applications,* 8 (1995): 224-231.

[11] Britto Jr., Sabourin R., F. Bortolozzi, and C. Y. Suen (2001). A two-stage hmm based system for recognizing handwritten numeral strings. In *Proc. 6th International Conference on Document Analysis and Recognition*, USA, 396-400.

[12] Brown M. K. (1964). On-Line Computer Recognition of Handprinted Characters. *IEEE Trans on Electronic Computers,* EC13: 750-752.

[13] Brown M. K. and Ganapathy S., Preprocessing techniques for cursive script word recognition. Pattern Recognition, 16(1983): 447-458.

[14] Bunke, H., (2003). Recognition of cursive Roman handwriting past, present and future. *ICDAR03* :448-459.

[15] Burr D. J.,(1983). Designing a handwriting reader. *IEEE Trans. PAMI,* 5(5): 554-559.

[16] Byun H., and Lee S. W. (2002).

Applications of support vector machines for pattern recognition. In *Proc. of the International Workshop on Pattern Recognition with Support Vector Machine*, Niagara Falls-Canada, pp. 213-236.

[17]Cai J. and Z. Q. Liu (1999). Integration of structural and statistical information for unconstrained handwritten numeral recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(3):263-270.

[18]Casey, R.G., and Lecolinet, E., (1996). A Survey of Methods and Strategies in Character Segmentation. *PAMI* , 18(7): 690-706.

[19]Connell Scott D., Jain Anil K. (2002). Writer Adaptation for Online Handwriting Recognition. *IEEE Trans. PAMI. 24*(3): 329-346.

[20]Dimauro G., Impedovo S., Pirlo G., Salzo A. (1997). Automatic bankcheck processing: A new engineered system. In S.Impedovo et al, editor, *International Journal of Pattern Recognition and Artificial Intelligence*, World Scientific, 467-503.

[21]Duda Richard O., and Peter E. Hart (1993). *Pattern Classification and Scene Analysis*. New York, Wiley.

[22]Duda R. O., Hart P. E., and Stork D. G. (2001). *Pattern Classification*. John Wiley and Sons, second edition edition.

[23]Elliman David G., and Lancaster I. T. (1990): A review of segmentation and contextual analysis techniques for text recognition. *Pattern Recognition 23*(3-4): 337-346.

[24]Farag R. F. (1979). Word-level recognition of cursive script. *IEEE Trans. On Computer*, Feb., C-28:172-175.

[25]Favata J T. (2001). Offline general handwritten word recognition using an approximate beam matching algorithm. *IEEE Trans PAMI,* 23(9):1009–1021.

[26]Flann, N. S. (1994). Recognition-Based Segmentation of On-Line Cursive Hand-writing. *Advances* in *Neural Information Processing Systems,* 6 (1994) 777-784.

[27]Frishkopf L.S. and Harmon L.D., (1961). Machine Reading of Cursive Script. Proc. Fourth Information Theory London Symp.

[28]Fujisaki H., Nagai S., and Hidaka N., (1971). On-line recognition of handwritten numerals. *Annual Report, Faculty Eng., Univ. Tokyo, Japan*, vol. 30: 103-110, 1971.

[29]Gader P. D., Forester B., Ganzberger M., A. Billies, B. Mitchell, M. Whalen, T. Youcum (1991). Recognition of handwritten digits using template and model matching. *Pattern Recognition,* 5(24):421-431.

[30]Gader P. D., . Keller J. M, and Cai J. (1995). A fuzzy logic system for detection and recognition of street number fields on handwritten postal addresses. *IEEE Transactions on FuzzySystems*, 3(1):83-95.

[31]Gaofeng Qian, (1998). An Analytic Approach for On-line Cursive Handwriting Interpretation. *PhD thesis,* The University of Texas at Dallas, USA.

[32]Govindan, V. K.. and Shivaprasad, A. P. (1990), Character Recognition-a Review, *Pattern Recognition,* 23 (1990) 671-683.

[33]Guberman S. A. and Rozentsveig V. V.(1976). Algorithms for the recognition of handwritten text. Automatika Telemekhanika, 5(1976): 122-129.

[34]Guillevic D., and Suen C. Y. (1995). Cursive script recognition applied to the processing of bank cheques. In *Proc. of 3th International Conference on Document Analysis and Recognition*, Montreal-Canada, pp. 11-14.

[35]Higgins, C. A., and Ford, D. M. (1992). On-Line Recognition of Connected Handwriting by Segmenta.tion and Template Ma.tching. *Proceedings of 11th IAPR International Conference on Pattern Recognition,* 2 (1992) 200-203.

[36]Hiroto Mitoma, Seiichi Uchida, and Hiroaki Sakoe. (2005). Online character recognition based on elastic matching and quadratic discrimination. *Proceedings of 8th International Conference on Document Analysis and Recognition* (ICDAR 2005, Seoul, Korea).1(2):36—40.

[37]Hu J., Brown M. K. and Turin W. (1996). HMM Based On-Line Handwriting Recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*,18(10): 1039-1044.

[38]Ichikawa T. and Yoshida J.(1974). On-line recognition of handprinted characters with associative read-out of patterns in a memory. *In Proc. 2nd Int. Joint Conf. On Pattern Recognition*: 206- 207.

[39]Ikeda K., Yamamura T., Mitamura Y., Fujiwara S., Y. Tominaga,and T. Kiyono.(1981). On-line recognition of handwritten characters utilizing positional and stroke vector sequence. *Pattern Recognition*, 13(3): 191-206, 1981.

[40]Impedovo S. (1984). Plane curve classification through Fourier descriptors: An application to Arabic hand-written numeral recognition. *In Proc. of 7th Int. Conf. On Pattern Recognition,* 1069-1072.

[41]Ishigaki K. and Morishita T., (1988). A Top-Down Online Handwritten Character Recognition Method via the Denotation of Variation. *Proc. 1988 Int'l Conf. on Computer Processing on Chinese and Oriental Languages,* 141-145.

[42]Jain K, Duin R. P. W., Mao J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4-37.

[43]Jain, A.K Mao, J. and Mohiuddin K. M. (1996). Artificial Neural Networks: A Tutorial. *IEEE computer Magzine*, 31-44.

[44]Jong Oh. (2001). An On-Line Handwriting Recognizer with Fisher Matching, Hypotheses Propagation Network and Context Constraint Models. *PhD thesis, Department of Computer Science New York University* USA.

[45]Jungpil Shin (2002). On-Line Handwriting Character Recognition Using Stroke Information" Hendtlass and M. Ali (Eds.): IEA/ AIE 2002, LNAI 2358, 703–714, @ Springer-Verlag Berlin Heidelberg 2002.

[46]Justino, E.J.R., Bortolozzi, F., Sabourin, R., (2002). The interpersonal and intrapersonal variability influences on off-line signature verification using HMM, SIBGRAPI 2002. XV Brazilian Symposium on Computer Graphics and Image Processing. vol. 1. Fortaleza, Ceará, Brazil. pp. 197–202.

[47]Kim H. Y. and Kim J. h. (1998). Handwritten korean character recognition based on hierarchical randon graph modeling. In *Proc. 6th International Workshop on Frontiers of Handwriting Recognition*, Taegon-Korea, August, pp. 557-586.

[48]Kittler J., Hatef M., R. Duin, and J. Matas (1998). On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226-239.

[49]Koerich L. (2002). *Large Vocabulary Off-Line Handwritten Word Recognition*. PhD thesis, École de Technologie Supérieure, Montreal-Canada, August.

[50]Koerich A. L., Sabourin R., Suen C. Y. (2003). Large vocabulary off-line handwriting recognition:A survey. *Pattern Anal Applic,* 6: 97–121.

[51]KOHONEN T. (1987). Adaptive, Associative, And Self-Organizing Functions In Neural Computing. APPLIED OPTICS 26(23):4910-4918

[52]Kundu, He Y., and Chen M. (2002). Alternatives to variable duration hmm in handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1275-1280.

[53]Kurtzberg J. M. and Tappert C. C. (1982). Segmentation procedure for handwritten symbols and words. *IBM Tech. Disclosure Bull*, vol. 25: 3848-3852, Dec. 1982.

[54]Lazzerini, and Marcelloni F. (2000). A linguistic fuzzy recognizer of off-line handwrriten characters. *Pattern Recognition Letters*, 21(4):319-327.

[55]Lecolinet E. E. and Baret O. (1994)., Cursive word recognition: Methods and strategies. *Fundamentals in Handwriting Recognition*, ed. S. Impedovo, 235-263.

[56]LeCun Y., Bottou L., Orr G. B., Muller K. R. (1998a). Eficient backprop. In Orr G. and K. Miller, editors, *Neural Networks: Tricks of the Trade*. Springer.

[57]LeCun Y., Bottou L., Bengio Y., and Haffner P. (1998b). Gradient-Based

Learning Applied to Document Recognition. *Proc. IEEE*, 86(11): 2278-2324.

[58]Leedham Graham, Guerfali Wacef, and Plamondon, R. (2002). Special Issue: Handwriting Processing and Applications. *Pattern Recognition, (*35) 5: 981-982.

[59]Lethelier, Leroux M., and Gilloux M. (1995). An automatic reading system for handwritten numeral amounts on french checks. *In Proc. 3th International Conference on Document Analysis and Recognition*, Montreal-Canada, August, pp. pages 92-97.

[60]Ling, M. Lizaraga, N. Gomes, and A. Koerich (1997). A prototype for brazilian bankcheck recognition. In S.Impedovo et al, editor, *International Journal of Pattern Recognition and Artificial Intelligence*,World Scientific, pp. 549-569.

[61]Manke,S., Finke, M., and Waibel, A. (1995). The *Use* of Dynamic Writing Information in a Connectionist On-Line Cursive Handwriting Recognition System. *Advances in Neural lnformation Processing Systems,* 7, MIT Press, 1995.

[62]Mantas, J. (1986), An Overview of Character Recognition Methodologies, *Pattern Recognition,* 19 (1986) 425-430.

[63]Matan O., Burges J. C., LeCun Y., Denker J. S. (1992). Multi-digit recognition using a space displacement neural network. In J. E. Moody, S. J. Hanson, and 165 R. L. Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 4, Morgan Kaufmann, pp. 488-495.

[64]Mermelstein P. and Eden M.,(1964). Experiments on computer recognition of connected handwritten words. Information and Control, June , 7(1964): 255-270.

[65]Mico L., and Oncina J. (1999). Comparison of fast nearest neighbour classifier for handwritten character recogniton. *Pattern Recognition Letters*, 19(3-4):351-356.

[66]Morassa, P., Barberis, L., Pagliano, S., and Vergano, D. (1993). Recognition Experiments of Cursive Dynamic Handwriting with Self-Organizing Xetworks. *Pattern Recognition,* 26 (1993) 451-460.

[67]Morita M, Sabourin R., F. Bortolozzi, and C. Suen. (2002). Segmentation and recognition of handwritten dates. *In Proc. 8*th *IWFHR*: 105–110.

[68]Nag R., K. Wong H., and Fallside F., (1986). Script recognition using hidden Markov models. *Proc. ICASSP*, 3(4): 2071–2074.

[69]Nakai M., Akira N., Shimodaira H., and Sagayama S. (2001). Substroke approach to HMM-based on-line Kanji handwriting recognition," *Proc. ICDAR*, 491–495, 2001.

[70]Nouboud, F., and Plamondon, . (1990), On-Line Recognition of Handprinted Chara.cters: Survey and Beta Tests, *Pattern Recognition,* 23 (1990) 1031-1044.

[71]Odaka K., Wakahara T. and Masuda I., (1986). Stroke-Order-Independent On-Line Character Recognition Algorithm and Its Application. *Rev. Electrical Comm. Laboratories*, 34(1): 79-85.

[72]Oliveira L. S., and Sabourin R. (2004). Support Vector Machines for Handwritten Numerical String Recognition, 9th *International Workshop on Frontiers in Handwriting Recognition*, October 26-29, Kokubunji, Tokyo, Japan, pp 39-44.

[73]Oliveira L. S., Sabourin R., Bortolozzi F., and C. Y. Suen. (2002). Automatic recognition of handwritten numerical strings: A recognition and verification strategy. *IEEE PAMI*, 24(11):1438–1454.

[74]Oulhadj, H., Petit, E., Lemoine, J., Gaudaire, M., and Lorett.e G. (1990). A Prediction- Verification Strategy for Automatic Recognition of Cursive Hand-writing, *Computer Processing of Handwriting,* Plamondon, R., and Leedham, C. G.(eds.), 1990, 181-206.

[75]Peter N. Belhumeur, J. P. Hespanha, and David J. Kriegman, (1997). Eigenfaces vs. Fisher Faces: Recognition Using Class Specific Linear Projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7).

[76]Plamondon R., (1993). Special issue on handwriting processing and recognition. *Pattern Recognition* ed.

[77]Plamondon R. and Leedham G. (1990). Computer Processing of Handwriting. eds.: *World Scientific*, Singapore.

[78]Plamondon R. and Lorette G. (1989). Automatic signature verification and writer identification the state of the art. *Pattern Recognition*, 22(2):107-131.

[79]Plamondon Rejean, and Sargur N. Srihari, (2000). On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey. 1EEE Transactions on Pattern Analysis and Machine Intelligence. 22(1): 63-84.

[80]Plamondon R., Suen C. Y., Bourdeau M and Barriere C. (1993). Methodologies for Evaluating Thinning Algorithms for Character Recognition. *Int'l J. Pattern Recognition and Artificial intelligence*, special issue on thinning algorithms, 7(5), 1247-1270.

[81]Powalka, R. K., Sherkat, N., and Whitrow, R. J. (1994). The Use of Word Shape Information for Cursive Script Recognition. *Fourth International Workshop on Frontiers of Handwriting Recognition,* Taiwan, 67-76.

[82]Pobgee P. J. and . Parks J. R,(1972). Application of a low cost graphical input tablet. *Inform. Processing*, vol. 71: 738-741.

[83]Rabiner L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2): 257-286.

[84]Rabiner L. and Juang B.. (1986). An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, pages 4–16.

[85]Rao, V. S. (1995). A Knowledge-Based Approach for Script Recognition without Training, *IEEE Transactions* on *PAMI,*17: 1233-1238.

[86]Ripley B. D., (1996). *Pattern Recognition and Neural Networks*. Cambridge, Cambridge University Press.

[87]Schenkel, M., Guyon, I., and Henderson, D. (1995). On-Line Cursive Script Recognition Using Time Delay Neural Networks and Hidden Markov Models! *Machine Vision and Applications,* 8 (1995) 215-223.

[88]Seni, G., Srihari, R. K., and Nasrabadi, N. (1996). Large Vocabulary Recogni*tion* of On-Line Handwritten Cursive Words. *IEEE Transactions* on *Pattern Analysis and Machine Intelligence,* 18 (1996) 757-762.

[89]Schomaker, L.R.B. (1993). Using Stroke- or Character-based Self-organizing Maps in the Recognition of On-line, Connected Cursive Script. *Pattern Recognition* , 26(3): 443-450.

[90]Schurmann J. (1996). *Pattern Classification* - A unified view of statistical and neural approaches. Wiley interscience.

[91]Seni G., Srihari R. K., and Nasrabadi N. (1996). Large Vocabulary Recognition of On-Line Handwritten Cursive Words. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(7).

[92]Senior W., and Robinson A. J. (2002). An off-line cursive handwriting recognition system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):309-321.

[93]Shridhar M. and Badreldin A. (1986). Recognition of isolated and simply connected handwritten numerals. *Pattern Recognition*, 19(1):1-12.

[94]Simner M., Hulstijn W., and Girouard P. (eds.1994). Forensic: Developmental and Neuropsychological Aspects of Handwriting. *Special issue, J. Forensic Document Examination*.

[95]Simner M. L., Leedham C. G. and Thomassen A. J. W. M. (eds.1996). Handwriting and Drawing Research: Basic and Applied Issues. *Amsterdam: IOS Press*.

[96]Steinherz T., Rivlin E., and Intrator N. (1999). Offline Cursive Script Word Recognition—A Survey. *Int'l J. Document Analysis and Recognition*, vol. 2, 90-110.

[97]Suen, C. *Y.,* Berthod, M., and Mori, S. (1980), Automatic Recognition of Ha.nd printed Character-the State of the Art,

Proceedings of IEEE. 68 (1980) 469-487.

[98]Tappert C. C.,(1982). Cursive script recognition by elastic matching. *IBM J. Res. Develop*. 26:765-771, Nov. 1982.

[99]Tappert C.C. (1984). Adaptive on-line handwriting recognition. *IEEE Seventh Int. Conf. Proc. on Pattern Recognition*, 1004-1007.

[100]Tappert C.C., Suen C.Y., Wakahara T.(1990). The state of the art in on-line handwriting recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(8), 787-808.

[101]Van Galen G. P. and Stelmach G. E. (1993). Handwriting: Issues of Psychomotor Control and Cognitive Models. *Acta Psychologica*, special volume, Amsterdam: North Holland.

[102]Vapnik V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York-USA.

[103]Vuurpijl L, and Schomaker L. (2000). Two-stage character classification: A combined approach of clustering and support vector classifiers. *Proc 7th International Workshop on Frontiers in Handwriting Recognition*, Amsterdam, Netherlands, 423–432.

[104]Wang F, Vuurpij L, and Schomaker L. (2000). Support vector machines for the classification of western handwritten capitals. *Proc 7th International Workshop on Frontiers in Handwriting Recognition*, Amsterdam, Netherlands; 167–176.

[105]Wakahara T., Suzuki A., Nakajima N., Miyahara S., and Odaka K., (1996). Stroke-Number and Stroke-Order Free On-Line Kanji Character Recognition as One-to-One Stroke Correspondence Problem. *IEICE Trans. Inf. & Syst.*, E79-D(5): 529–534.

[106]Wan J., Wing A. M. and Sovik N. (1991). Development of Graphic Skills: Research, Perspectives and Educational Implications. *London: Academic Press*.

[107]Ward J. R. and Kuklinski T.,(1988). A model for variability effects in handprinting with implications for the design of handwriting character recognition systems. *IEEE Trans. Syst., Man, Cybern*.18:438-451.

[108]Wells, C. J., Evett, L. J., Whitby, P. E., and Whitrow, R. J. (1990), Fast Dictionary Look-up for Contextual Word Recognition, *Pattern Recognition,* 23 (1990): 501-508.

[109]Xie S. L., and Suk M. (1988). On machine recognition of hand-printed chinese character by feature relaxation. *Pattern Recognition*, 21(1):1-7.

[110]Yacoubi E., Gilloux M., and J. Bertille. (2002). A statistical approach for phrase location an recognition within a text line: An application to street name recognition. *IEEE PAMI*, 24(2):172–188.

[111]Yacoubi E, Gilloux M., Sabourin R., and Suen C. Y. (1999). An hmm-based approach for offline unconstrained handwritten word modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):752-760.

[112]Yoshida K. and Sakoe H. (1982). Online handwritten character recognition for a personal computer system. *IEEE Trans. Consumer Electronics*, CE-28(3): 202-209, 1982.

[113] Zhang G. P. (2000). Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics* - Part C: Applications and Reviews, 30(4):451-462.

[114]Zhang, Fu M., Yan H., and Fabri M. A. (1999). Handwritten digit recognition by adaptativesubspace self organizing map. *IEEE Trans. on Neural Networks*, 10:939-945.

[115]Zhou J (1999). Recognition and Verification of Unconstrained Handwritten Numeral. *PhD thesis*, Concordia University, Montreal-Canada.

[116]Liu Cheng-Lin, Stefan Jaeger, and Masaki Nakagawa (2004). Online Recognition of Chinese Characters:The State-of-the-Art. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(2), 198-203.