CHAPTER-3

3 Review of Literature

3.1 Introduction

The advancements in pattern recognition has accelerated recently due to the many emerging applications which are not only challenging, but also computationally more demanding, such as Optical Character Recognition (OCR), Document Classification, Computer Vision, Data Mining, Shape Recognition, and Biometric Authentication, for instance. The area of OCR is becoming an integral part of document scanners, and is used in many applications such as postal processing, script recognition, banking, security (i.e. passport authentication) and language identification. The research in this area has been ongoing for over half a century and the outcomes have been astounding with successful recognition rates for printed characters exceeding 99%, with significant improvements in performance for handwritten cursive character recognition where recognition rates have exceeded the 90% mark (Alginahi, 2010).

Cursive handwriting recognition is a challenging task for many real world applications such as document authentication, form processing, postal address recognition, reading machines for the blind, bank cheque recognition and interpretation of historical documents. Therefore, in the last few decades the researchers have put enormous effort to develop various techniques for handwriting recognition. This chapter presents the current state of the art in cursive handwriting recognition. This chapter also presents segmentation strategies for automated recognition of off-line unconstrained cursive handwriting from static surfaces. This chapter provides a comprehensive literature with basic and advanced techniques and comparison of research results of various researchers in the domain of handwritten words recognition (Verma and Blumenstein, 2008).
Chapter 3

Review of Literature

The research on cursive handwriting recognition has grown significantly in recent years. In the literature, many papers have been published with research detailing new techniques for the classification of handwritten numerals, characters and words (Plamondon and Srihari, 2000; Suen et al., 1993; Cho, 1997; Casey and Lecolinet, 1996; Dunn and Wang, 1992; Lu, 1995; Lu and Shridhar, 1996; Elliman and Lancaster, 1990; Fujisawa et al., 1992; Wang et al., 2005; Britto Jr et al., 2004; Singh and Amin, 1999; Gader et al., 1997; Blumenstein et al., 2004; Suen and Tan, 2005; Marinai et al., 2005; Liu and Fujisawa, 2005; Yanikoglu and Sandon, 1998; Dimarco et al., 1998; Xiao, X. and Leedham, G. 2000; Chiang, 1998; Martin et al., 1993; Eastwood et al., 1997; Srihari, 1993; Gilloux, 1993).

In the literature (Verma and Blumenstein, 2008), some researchers have obtained very promising results for isolated/segmented numerals and characters using conventional and intelligent techniques. However, the results obtained for the segmentation and recognition of cursive handwritten words have not been satisfactory in comparison (Kapp et al., 2007; Blumenstein and Verma, 2001; Gang et al., 2002; Verma et al., 1998; Blumenstein et al., 2003; Verma, 2003; Blumenstein and Verma, 1999; Fan and Verma, 2002; Verma et al., 2001; Gunter and Bunke, 2004; Vinciarelli et al., 2003; Verma et al., 2004; Arica and Yarman-Vural, 2002; Camerra and Vinciarelli, 2003; Hanmandlu et al., 2003; Gader et al., 1997; Günter and Bunke, 2005; Viard-Gaudin et al., 2005; Schambach, 2005; Chevalier et al., 2005; Lee and Coelho, 2005; Srihari, 2006; Gatos et al., 2006; Koerich et al., 2006; Xu et al., 2003; Wen et al., 2007).

The reason for not achieving satisfactory recognition rates is the difficult nature of cursive handwriting and difficulties in the accurate segmentation and recognition of cursive and touching characters (Verma and Blumenstein, 2008).

This chapter reports on the state-of-the-art in handwriting recognition research and methods for preprocessing, segmentation, feature extraction and recognition of cursive handwritten words.
3.2 Typical Handwriting Recognition System

A typical handwriting recognition system is characterized by a number of steps, which include (a) Digitization/Image acquisition, (b) Preprocessing, (c) Segmentation (d) Feature Extraction and (e) Recognition/Classification. Fig.3.1 (Verma and Blumenstein, 2008) illustrates one such system for handwritten word recognition.

![Diagram of a Typical Segmentation-Based Handwriting Recognition System]

3.3 Preprocessing

Preprocessing is the preliminary step which transforms the data into a format that will be more easily and effectively processed. Therefore, the main task in preprocessing of the captured data is to decrease the variation that causes a reduction in the recognition rate and increases the complexities, as for example, preprocessing of the input raw stroke of characters is crucial for the success of efficient character recognition systems. Thus, preprocessing is an essential stage prior to feature extraction since it controls the suitability of the results for the successive stages. The stages in a pattern recognition system are in a pipeline fashion meaning that each stage depends on the success of the previous stage in order to produce optimal/valid results. However, it is evident that the most appropriate feature vectors for the classification stage will only be produced with
the facilitation from the preprocessing stage. The main objective of the preprocessing stage is to normalize and remove variations that would otherwise complicate the classification and reduce the recognition rate (Alginahi, 2010).

Thus, the use of preprocessing techniques may enhance a document image preparing it for the next stage in a character recognition system. Below is a list of preprocessing techniques that have been employed by various researchers in an attempt to increase the performance of the segmentation / recognition process:

- Thresholding
- Noise Removal
- Size Normalization
- De-skewing and Slant Correction
- Thinning and Skeletonization

### 3.3.1 Thresholding

Image thresholding is the process of separating the foreground information (objects) of an image from its background. Hence, thresholding is usually applied to grey-level or colored document scanned images. Thresholding can be divided into two main categories: Global and Local.

Global Thresholding methods choose one threshold value for the entire document image, which is often based on the estimation of the background level from the intensity histogram of the image.

Local Adaptive Thresholding uses different values for each pixel according to the local area information. There are hundreds of thresholding algorithms which have been published in the literature, for example, Sahoo et al. compared the performance of more than 20 global thresholding algorithms using uniformly or shape measures. The comparison showed that Otsu class separability method gave best performance (Sahoo et al., 1988; Otsu, 1979). On the other hand, in an evaluation for change detection by Rosin
Chapter 3 Review of Literature

and Ioannidis concluded that the Otsu algorithm performed very poorly compared to other global methods (Rosin and Ioannidis, 2003; Otsu, 1979). The OCR goal directed evaluation study by Trier and Jain examined four global techniques showing that the Otsu method outperformed the other methods investigated in the study (Trier and Jain, 1995). In addition, Fischer compared 15 global methods and confirmed that the Otsu method is preferred in document image processing (Fischer, 2000). The Otsu method is one of the widely used techniques used to convert a grey-level image into a binary image then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal (Alginahi, 2010).

3.3.2 Noise Removal

Document analysis systems benefit from the reduction of noise in the preprocessing stage. This can provide a substantial improvement in the reliability and robustness of the feature extraction and recognition stages of the OCR system. A common appearance of noise in binary images takes the form of isolated pixels, salt-and-pepper noise or speckle noise, thus; the processing of removing this type of noise is called filling, where each isolated pixel salt-and-pepper “island” is filled in by the surrounding “sea” (O’Gorman et al., 2008; Alginahi, 2010).

Noise (small dots or blobs) may easily be introduced into an image during image acquisition (Verma and Blumenstein, 2008). Noise elimination in word images is important for further processing; therefore these small foreground components are usually removed. Chen et al. (1992) used morphological opening operations to remove noise in handwritten words. Kim et al. (1999) identified noise in a word image by comparing the sizes and shapes of connected components in an image to the average stroke width. Madhvanath et al. (1999) also analyze the size and shape of connected components in a word image and compare them to a threshold to remove salt and pepper noise. In postal address words and other real world applications, larger noise is sometimes present such as underlines. Therefore some researchers have also applied some form of underline removal to their word images (Dimauro et al., 1997).
3.3.3 Size Normalization

Scaling may sometimes be necessary to produce words of relative size. In the case of (Burges et al., 1992), the authors used a neural network for the segmentation stage of their system. The neural network accepted areas between the upper and lower baselines of each word as input. This area, called the core, must be of fixed height to be used in conjunction with the neural net. Therefore it was necessary to scale the words so that all cores are of an identical height (Verma and Blumenstein, 2008).

3.3.4 De-skewing and Slant Correction

De-skewing is the process of first detecting whether the handwritten word has been written on a slope, and then rotating the word if the slope's angle is too high so that the baseline of the word is horizontal (Verma and Blumenstein, 2008). Some examples of techniques for correcting slope are described in (Senior, 1994; Brown and Ganapathy, 1983). Some degree of skew is unavoidable either a paper is scanned manually or mechanically (Sarfraz and Rasheed, 2008; Sadri and Cheriet, 2009; Saba et al., 2011).

Slant estimation and correction is an integral part of any word image preprocessing (Verma and Blumenstein, 2008). Bozinovic and Srihari (1989) employed an algorithm that estimated the slant of a word by first isolating those parts of the image that represented near vertical lines (this is accomplished by removing horizontal strokes through run-length analysis). Secondly, an average estimation of the slant given by the near-vertical lines is obtained. The word image is then slant corrected by applying a transformation. In their system, the presence of a slant correction procedure was essential for segmenting their words using vertical dissection. Other estimation and correction techniques have been employed in the literature. Some have accomplished this using the chain code histogram of entire border pixels (Kimura et al., 1993; Ding et al., 1999), while others have estimated the slope through analysis of the slanted vertical projections at various angles (Guillevic and Suen, 1994).
3.3.5 Thinning and Skeletonization

The process of slant correction introduces noise in the contour of the image in the form of bumps and holes. Therefore some sort of smoothing technique is usually applied to remove contour noise. As also previously described, some researchers have used the skeleton of the word image to normalize the stroke width.

Thinning is a data reduction process that erodes an object until it is one-pixel wide, producing a skeleton of the object making it easier to recognize objects such as characters. Thinning erodes an object over and over again (without breaking it) until it is one-pixel wide. On the other hand, the medial axis transform finds the points in an object that form lines down its center (Davies, 2005). The medial axis transform is similar to measuring the Euclidean distance of any pixel in an object to the edge of the object, hence, it consists of all points in an object that are minimally distant to more than one edge of the object (Russ, 2007; Alginahi Y, 2010).

This operation is still a topic of debate as there are some advantages as well as some disadvantages of using the skeleton of the word image for word recognition.

3.4 Segmentation

Segmentation is a difficult and error prone process because of the Sayre's paradox (1973), a character cannot be segmented before having been recognized and cannot be recognized before having been segmented. It seems that the character segmentation process requires that the properties of a character be known; this information may be obtained through recognition. Unfortunately, to obtain knowledge of a character's appearance, segmentation is required. Therefore it is obvious that one stage is dependent on the other and knowledge of character symbol structure in a word is helpful in segmentation (Rehman and Saba, 2012).

Character segmentation is an operation that seeks to decompose an image of a sequence of characters into sub-images of individual symbols (Rehman and Saba, 2012).
Several review papers highlighted different issues in cursive script segmentation and acknowledged the segmentation stage as the most difficult step in the process of cursive handwriting recognition (Casey and Lecolinet, 1996; Dunn and Wang, 1992; Lu, 1995; Lu and Shridhar, 1996; Elliman and Lancaster, 1990; Fujisawa et al., 1992; Steinherz et al., 1999; Plamondon and Srihari, 2000; Blumenstein and Verma, 2001; Vinciarelli, 2002; Gang et al., 2002; Koerich et al., 2003; Bortolozzi et al., 2005; Rehman and Dzulkifli, 2008; Saba et al., 2011).

In the literature, for achieving high recognition accuracy, several segmentation techniques are proposed that can be broadly classified into three categories, namely Explicit Segmentation (Pure Segmentation), Implicit Segmentation (Recognition Based Segmentation) and Holistic (Segmentation Free) Approaches as shown in Fig.3.2.

3.4.1 Explicit Segmentation

When explicit segmentation (pure segmentation) is adopted for recognition; segmentation becomes the most crucial step of the handwritten word recognition problem. In this classical approach, input word image of sequence of characters is portioned into sub images of individual characters, which are then classified. The process of cutting up the word images into classifiable character sub images is termed as dissection. Many researchers in the literature adopted this dissection based segmentation techniques (Saba et al., 2011; Al Hamed and Zitar, 2010; Cheriet, 1993). These
techniques are used to find all the interconnections between character images (also called ligatures) and cut the word image through all the detected ligatures.

According to (Rehman and Saba, 2012), most of the researchers perform dissection via pre-segmentation. It is used to locate areas in the word containing explicit features that are likely to occur within or between characters in the form of valley such as ligatures. However, it also cuts the characters ‘w’, ‘v’ etc, whose contours contain a valley and therefore, deduce as a ligature.

The algorithms propose by (Maier, 1986; Lecolinet and Crettez, 1991) are mainly based on the detection of the valleys of the upper profile of the word and do not use further information about the actual shape of the ligatures. These techniques, because of their extreme simplicity, are prone to erroneous ligature detection, such as, in case of not actually closed loops or when a valley occurs inside a character.

Some systems investigate ligatures close to the baseline, but such efforts cannot brought fruitful results due to inherited nature of certain characters such as ‘u’, ‘w’, ‘g’ etc that do not contain ligatures close to the baseline. Holt et al. (1992) detect ligatures by locating minima in the upper contour of words, location of holes, contour direction and core region position. Segmentation points are marked if a minima in the upper contour is located, except if the contour component in question formed part of a hole. Similarly, Kimura et al. (1993) propose segmentation–recognition system for handwritten postal words; for segmentation part, they analyze upper contour. According to their investigation, prospective segmentation points are laid in those local minima that are deep enough and are adjacent to local maxima. Finally, segmentation points shift horizontally to the right or left to obtain valid segmented characters.

Veloso et al. (2000) hypothesized segmentation of handwritten cursive words based on natural segmentation points and ligatures. Accordingly, natural segmentation points are analyzed using histogram projection taken from five different angles and ligature candidates obtained from morphological operations of opening and closing.
Verma (2002) over-segment cursive handwritten word and extracted left, centre characters and segmentation point to obtain a character confidence via neural validation. Finally, all confidence values are fused to turn out correct segmentation points and true-segmented characters. Average segmentation accuracy is reported up to 73.62%.

Verma (2003) propose rule-based segmentation of handwritten words. Following heuristic segmentation, a sequence of rules proposed to check the validity of the existing segmentation points and to cover miss-segmentation. Five reference lines are detected that made the entire process computationally expensive. Finally, rules for removing and inserting segment lines based on weak assumption; even though, neural network is trained for those assumptions but require a lot of training. Despite all efforts, over-segmentation is 10.02% and bad segmentation is up to 8.7%, however miss-segmentation is minimum up to 0.2%. Overall 81.08% segmentation accuracy on CEDAR database is claimed.

Ghosh et al. (2004) propose direct segmentation approach in their fully automated off-line handwriting recognition system. The segmentation phase employs many heuristic based set of rules in an iterative procedure and finally a neural network validation system is implemented. Accurate segmentation rate is 83.6%. However, over-segmentation and bad segmentation is considerably high up to 10.8 and 5.4% respectively, whereas, missed segmentation rate is 0.2%.

Cheng and Blumenstein (2005b) propose feature-based heuristic segmentation algorithm consisted of two steps. In first step, prospective segmentation points are found by analyzing ligatures and global characteristics of handwriting. In the second step, fused left and centre character confidence values.

Additionally, trained ANN are used for segmentation points validation based on modified direction features propose by Blumenstein et al. (2003). The improved segmentation algorithm is examined on test set of CEDAR database.
Later, Cheng and Blumenstein (2005a) improve their own previous work (Cheng et al., 2004; Cheng and Blumenstein, 2005b) and propose enhanced heuristic segmenter (EHS) to improve segmentation of cursive handwriting. In the first step, enhanced heuristic segmenter makes use of two enhanced features: ligature detection and neural assistance to locate prospective segmentation points. In the second step, left, right character confidence outputs are fused with neuro-segmentation point’s validation. CEDAR benchmark database is employed for training and testing steps.

Samrajya et al. (2006) investigate hypergraph model to segment a cursive handwritten word image into isolated characters. Hypergraph model treats an image as packets of pixels. Authors claim that by recombining these packets of different sizes a given word image can be segmented into characters if at least one of the combinations provided a correct segmentation. However, neither segmentation results are presented for comparison nor the technique seems to yield successful results for horizontal overlapped and touching characters.

Dawoud (2007) introduce iterative cross section sequence graph (ICSSG) for the character segmentation. ICSSG tracks the characters growth at equally spaced thresholds. The iterative thresholding reduces the effect of information loss associated with image binarization. However, the experiments are performed on handwritten digits only.

Recently, Lee and Verma (2008a) propose a new segmentation algorithm for off-line cursive handwriting recognition. Initially, word images are dissected heuristically based on pixel density between upper and lower baselines. Each segment passed through multiple expert based validation processes to determine valid character boundaries. An average segmentation error up to 5.25% for miss-segmentation, over-segmentation and bad segmentation is reported on 218 test words of CEDAR.

Additionally, Lee and Verma (2008b) propose over-segmentation and validation strategy based segmentation algorithm for off-line cursive handwriting recognition. In the
first step, word image is over-segmented such that all valid segmentation points are marked. In the second step, invalid segmentation points are detected and extracted through a validation module. An average segmentation error up to 5.50% for miss-segmentation, over-segmentation and bad segmentation is reported on 311 test words of CEDAR.

3.4.2 Implicit Segmentation

Implicit segmentation (recognition based segmentation) based recognition, in which the system searches the image for components that match classes in its alphabet. However, implicit segmentation-based methods are employed as an alternative to integrate segmentation and recognition processes. Accordingly, Hidden Markov Models (HMM) based approaches are emerged. Actually, this approach is developed for speech recognition where it brought fruitful results (Rabiner, 1989). Therefore, its success diverts researcher's attention to apply HMM in word recognition. Bose and Kuo (1994), Elms et al. (1989) prove benefits of applying HMM based techniques to recognize printed words. The main interest of this category of methods is that they bypass the segmentation problem: No complex "dissection" algorithm has to be built and recognition errors are basically due to failures in classification. The approach has also been called "segmentation-free" recognition.

Cavalin et al. (2006) propose two-stage HMM based method for recognition of strings of characters (words or numerals). In first stage, an implicit segmentation scheme is applied to segment either words or numeral strings and verification performs in the second stage. Accordingly, foreground and background features are combined to compensate the loss in terms of recognition rate during implicit segmentation in previous stage. Word recognition accuracy up to 88.2% is reported on lexicon of size 3,771.

Hamamura et al. (2007) propose an analytic word recognition algorithm based on improved posteriori probability ratio. Accordingly, a new evaluation function is proposed and they claimed 9.1% improvement in recognition accuracy. The development of automatic procedures that is able to learn segmentation rules from training data. Finally,
automatically inferred parameters guided searching process for fitting the optimal character hypotheses. However, no benchmark database is employed for experimentations.

The challenge is to find some way to compensate the loss in recognition performance resulting from the necessary trade-off between segmentation and recognition carried out in an implicit segmentation-based method (Bortolozzi et al., 2005). Implicit methods use argument that in case of cursive script, segmentation cannot be attained without recognition, because without understanding the character included in the word there is no good criteria to avoid segmentation errors.

Nevertheless, there are evidences that implicit segmentation approaches for word recognition has some drawbacks. The words with broken, touched, illegible or missed characters cannot be recognized. Classical approaches for segmentation also face serious problems, such as collapse problem. Therefore, success seems in the hybrid strategies.

3.4.3 Hybrid Approaches

The literature is replete with hybrid approaches proposed by a number of researchers to optimize algorithms with linear searching techniques, contextual and lexicon knowledge (Casey, 1992; Kimura et al., 1992; Favata and Srihari, 1992; Bruel, 1994; Sinha et al., 1993; Kim and Govindaraju, 1997; Kim et al., 2000; Hanhong, 2002; Liu et al., 2002; Grandidier, 2003; Koch et al., 2004; Farah et al., 2005).

Recently, Rehman and Dzulkifli (2008) proposed a new fast segmentation approach for off-line cursive handwritten words with accuracy up to 91.21% on a subset of IAM database. Authors proposed certain rules to analyze ligatures along with knowledge of character shape. The detailed analysis (Blumenstein and Verma, 2001; Verma et al., 2004; Chen and Leedham, 2005; Rehman and Dzulkifli, 2008) has shown that most existing segmentation algorithms have three major problems: (1) inaccurately cutting characters into parts; (2) missing many segmentation points; and (3)
over-segmenting a character many times, which contributes to errors in the word recognition process. Most researchers have evaluated their segmentation accuracy as an overall word recognition performance. Additionally, database and experimental setup is different among the researchers. Hence it is difficult, if not impossible, to compare their results. However, some of the top results for segmenting cursive words are outlined in Table 3.1 (Rehman and Saba, 2012) for fair comparison.

Table 3.1 Comparison of Segmentation Results

<table>
<thead>
<tr>
<th>Author</th>
<th>Segmentation Approach</th>
<th>Segmentation Rate (%)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blumenstein and Verma (1997)</td>
<td>ANN+conventional Method</td>
<td>81.21</td>
<td>800 Words</td>
</tr>
<tr>
<td>Verma and Gader (2000)</td>
<td>Feature based+ANN</td>
<td>76.52</td>
<td>Words number not mentioned</td>
</tr>
<tr>
<td>Verma et al. (2001)</td>
<td>Fusion of multiple Word recognition Techniques</td>
<td>86</td>
<td>317 words used for Testing</td>
</tr>
<tr>
<td>Blumenstein and Verma (2001)</td>
<td>Feature based+ANN</td>
<td>78.85</td>
<td>Words number not Mentioned</td>
</tr>
<tr>
<td>Verma (2002)</td>
<td>Feature based+ANN</td>
<td>84.87</td>
<td>300 test words only</td>
</tr>
<tr>
<td>Cheng and Blumenstein (2005a)</td>
<td>Ligature detection+ANN</td>
<td>84.19</td>
<td>317 test words</td>
</tr>
<tr>
<td>Samrajya et al. (2006)</td>
<td>Hypergraph+ligature analysis</td>
<td>Not mentioned</td>
<td>Not mentioned</td>
</tr>
<tr>
<td>Rehman and Dzulkifli (2008)</td>
<td>Ligature and shape analysis</td>
<td>88.21</td>
<td>2,936 words</td>
</tr>
</tbody>
</table>
3.4.4 Holistic Approaches

A holistic (Segmentation Free) process recognizes an entire word as a unit. A major drawback of this class of methods is that their use is usually restricted to a predefined lexicon. Since they do not deal directly with letters but only with words, recognition is necessarily constrained to a specific lexicon of words. This point is especially critical when training on word samples is required. A training stage is thus mandatory to expand or modify the lexicon of possible words. This property makes this kind of method more suitable for applications where the lexicon is statically defined (and not likely to change), like bank cheque recognition. They can also be used for on-line recognition on a personal computer (or notepad), the recognition algorithm being then tuned to the writing of a specific user as well as to the particular vocabulary concerned (Casey and Lecolinet, 1996).

Dynamic Programming was employed in (Moreau et al., 1991; Plessis et al., 1993) for cheque amount and city name recognition. Words are represented by a list of features indicating the presence of ascenders, descenders, directional strokes and closed loops. Hidden Markov Models are used (Nag et al., 1986) for the recognition of literal digits and for off-line cheque recognition (Gilloux et al., 1993). Angular representation is used in the first system to represent the feature, while structural off-line primitives are used in the second case.

3.5 Feature Extraction

The purpose of feature extraction is to achieve most relevant and discriminative features to identify a symbol uniquely (Blumenstein et al. 2007). Many feature extraction technique are proposed and investigated in the literature that may be used for numeral and character recognition. Consequently, recent techniques show very promising results for separated handwritten numerals recognition (Wang et al., 2005), however the same accuracy has not been attained for cursive character classification (Blumenstein et al., 2007). It is mainly due to ambiguity of the character without context of the entire word (Cavalin et al., 2006). Second problem is the illegibility of some characters due to nature
of cursive handwriting, distorted and broken characters (Blumenstein et al., 2003). Finally, the segmentation process may cause some irregularities depending on the approach adopted (Blumenstein and Verma, 2001). According to Suen (1986), there are two main categories of features; Statistical Features and Structure Features.

3.5.1 Statistical Features

These features are derived from statistical distribution of every point in a character matrix such as moments, histograms, profile projection and zoning (Kimura et al., 1992; Blumenstein et al., 2007; Kim et al., 2000; Vamvakas et al., 2007). Statistical features are also known as global features as they are usually extracted and averaged in sub-images such as meshes (Kang and Kim, 2004). Initially, statistical features are developed to recognize machine printed characters (Suen et al., 1980).

3.5.2 Structural Features

On the other hand, structural features are based on geometric and topological features of characters such as contours, loops, end points (Koerich et al., 2003). In this regard, Trier et al. (1996) present a detailed review of feature extraction methods for off-line isolated character recognition such as template matching, deformable templates, zoning, contour profile, profile projection, geometric moments invariants, zernike moments, fourier descriptors, spline curve estimation. The methods are applicable to gray level character images, binary character images, thinned character images, character contours and character graphs.

A number of techniques extract features from character's contours. Kimura and Shridhar (1991) divide contour profile into two halves and discrete function of each half is approximated to extract features. Yamada and Nakano (1996) explore direction histogram in character image to extract features. A multi-template based strategy with clustering feature is adopted to recognize segmented characters. Likewise, Kimura et al. (1997) evaluate features by calculating local histograms based on chain code information for segmented character classification. Krzyzyzak et al. (1990) extract features from inner and outer contours of characters: simple topological features extracted from the inner
contours and fifteen fourier descriptors are extracted from the outer contours. Oh and Suen (1998) extract two feature set based on distance transformation and Directional Distance Distribution (DDD). In the first feature set, distance of each white pixel to the nearest black pixel in the character image is calculated without character skeletonization. The second feature set composes of information encoding both black/white and directional distance distributions. Additionally, a new method of map tiling is introduced and is applied to the DDD feature to improve its discriminative ability. All experiments are carried out on three different sets of characters consisting of numerals, English letters, and Hangul letters. Promising results reported to confirm the best combination of DDD feature and the map tiling. Blumenstein et al. (2007), Verma (2003) and Verma et al. (2004) use directional features extracted from character contours. The technique replaces foreground pixels of character contours with suitable direction values. Finally, image is divided into windows to extract features. Likewise, Mitrpanont and Limkonglap (2007) also analyze contours of Thai characters to capture movement of features for Thai character recognition.

3.6 Recognition

A number of classification techniques has been developed and investigated for the classification of numerals, characters and words. The recognition techniques are divided into two main categories; statistical techniques and intelligent techniques. The statistical classifiers make decision based on statistical decision function. Many successful recognition techniques are based on this strategy such as template matching, Bayesian classifier, polynomial discriminate classifier, fuzzy logic/rules, k-nearest-neighbor (K-NN). However, some statistical methods require all training samples to be stored and compared for the classification process (Liu and Fujisawa, 2005). Recently, neural network classifiers are proved to be powerful and successful for character/word recognition (Verma et al., 2004; Blumenstein et al., 2007). However, to improve the intelligence of these ANNs, huge iterations, complex computations, and learning algorithms are needed, which also lead to consume the processor time. Therefore, if the recognition accuracy is improved, the consumed learning time will increase and vice versa. Which is the main drawback of ANN based approaches (Aburas and Rehiel, 2008).
HMM-based classifiers remained highly successful for numeric recognition and recognition rates above 98% for off-line handwritten isolated numerals are reported in the literature (Cavalin et al., 2006; Britto et al., 2004; Arica and Yarman-Vural, 2002; Cai and Liu, 1999). Likewise, for the global word recognition problem, HMMs based techniques are growing successfully (Gunter and Bunke, 2005; Schambach, 2005; Viard-Gaudin et al., 2005; Grandidier, 2003; Kundu and Chen, 2002; Senior and Robinson, 2002). On the other hand, for analytical approaches, neural network classification has been commonly used in conjunction with dynamic programming (Gader et al., 1997). Recently, few researchers have employed support vector machines for numeral/character classification successfully and promising results above 99% are reported (Liu and Fujisawa, 2005). Moreover, support vector machines also have been used successfully for classification of words in recent studies (Gatos et al., 2006b). Summary of recognition performances of recent off-line script recognition systems in chronological order year wise are shown in Table 3.2.

<table>
<thead>
<tr>
<th>Author</th>
<th>Classifier</th>
<th>Lexicon Size (in words)</th>
<th>Problem Domain</th>
<th>Recognition Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gader et al. (1994)</td>
<td>ANN</td>
<td>100</td>
<td>Address mail</td>
<td>85.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>USPS</td>
<td></td>
</tr>
<tr>
<td>Gilloux et al. (1995b)</td>
<td>RBF/HMM</td>
<td>30</td>
<td>Legal amount words of postal/cheque</td>
<td>83.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knerr et al. (1997)</td>
<td>NN/HMM</td>
<td>30</td>
<td>Legal &amp; courtesy amount of bank</td>
<td>76.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Guillevic and Suen</td>
<td>HMM/KNN</td>
<td>30</td>
<td>LA words(ENG)</td>
<td>86.7</td>
</tr>
<tr>
<td>Reference</td>
<td>Method</td>
<td>Dataset</td>
<td>Accuracy</td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------</td>
<td>---------------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>Chiang (1998)</td>
<td>NN</td>
<td>USPS database mail</td>
<td>87.4</td>
<td></td>
</tr>
<tr>
<td>Kim et al. (2000)</td>
<td>HMM/MLP</td>
<td>LA words</td>
<td>92.2</td>
<td></td>
</tr>
<tr>
<td>Oliveira et al. (2002)</td>
<td>MLP</td>
<td>Numerical strings</td>
<td>87.2</td>
<td></td>
</tr>
<tr>
<td>Kundu and Chen (2002)</td>
<td>HMM</td>
<td>Postal words</td>
<td>88.2</td>
<td></td>
</tr>
<tr>
<td>Koch et al. (2004)</td>
<td>MLP</td>
<td>Letters (FR)</td>
<td>67.8</td>
<td></td>
</tr>
<tr>
<td>Günter and Bunke (2004)</td>
<td>HMM+Ensembled methods</td>
<td>IAM</td>
<td>71.58</td>
<td></td>
</tr>
<tr>
<td>Günter and Bunke (2005)</td>
<td>HMM+Ensembled methods</td>
<td>IAM</td>
<td>75.61–82.28</td>
<td></td>
</tr>
<tr>
<td>Koerich et al. (2005)</td>
<td>HMM</td>
<td>SRTP</td>
<td>77.62–99.29</td>
<td></td>
</tr>
<tr>
<td>Schambach (2005)</td>
<td>HMM</td>
<td>Siemens</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>Koerich et al. (2006)</td>
<td>HMM</td>
<td>SRTP</td>
<td>78</td>
<td></td>
</tr>
<tr>
<td>Gatos et al. (2006a)</td>
<td>K-NN</td>
<td>IAM</td>
<td>81.05</td>
<td></td>
</tr>
<tr>
<td>Gatos et al. (2006b)</td>
<td>SVM</td>
<td>IAM</td>
<td>87.68</td>
<td></td>
</tr>
<tr>
<td>Tomoyuki et al. (2007)</td>
<td>Posterior probability/DP</td>
<td>IAM</td>
<td>80.2</td>
<td></td>
</tr>
</tbody>
</table>
3.7 Conclusion

In this chapter, a state of the art in off-line cursive script recognition and its associated components are presented with the great emphasis on segmentation-based off-line cursive script recognition technique. A critical literature review of existing techniques and comparative study of recent achievements in the area has also been presented. Novel strategies by the authors to tackle existing problems in preprocessing, segmentation-based script recognition have also been presented.

By the detailed analysis of the literature, it is observed that the research is almost matured in area of numeral recognition however the same accuracy level is not met with alphabets. The problem of cursive character recognition remains very much an open problem. It is mainly due to presence of noisy, broken, multi-stroke, incomplete and ambiguous characters in words. To handle this type of problem new feature extraction/selection techniques and multistage classifiers are desired.

As far as word recognition is concerned, the problem is seemed to be solved in small and static lexicons using holistic strategy. However, recognition accuracy dropped significantly for larger lexicon. Therefore, segmentation based word recognition is an alternative solution. On the other hand, segmentation algorithms have three major problems: first, inaccurately cutting characters into parts; second, missing many segmentation points; third, over-segmenting a character many times, which contributes to errors in the word recognition process; finally, negative effects on speed are also observed. Still, algorithms to tackle the variety of writing styles as well as appropriate features to describe the suitable segmentation points of interest and for subsequently determining correct/incorrect segmentations are lacking.