Chapter 1

Introduction

In our everyday life we benefit from intelligent machines that accomplish many operations for us. Sometimes we are not even aware of how many intelligent devices aid us in performing common tasks, such as driving a car, buying goods with a credit card, booking a trip at a travel agency, or writing a document in an advanced text editor. It is a challenge to artificial intelligence to improve such systems to make life more comfortable.

One specific research area in artificial intelligence is the field of natural language processing (NLP). NLP aims to automate generation and understanding of natural human languages, i.e., to convert information from computer databases into human language and vice versa. The automated transcription of written or spoken data into a machine readable format is closely related to this field. This is an essential step to understand human language. The research community focuses on speech recognition, machine printed text recognition, and handwritten text recognition. Assuming Roman script, the latter sub-field can be further divided into isolated character recognition, cursive word recognition, and cursive word sequence recognition. While there already exist mature solutions for isolated character recognition, cursive word sequence recognition is the most complex task in the field of handwriting recognition of Roman script.

This book considers a relatively new task, the recognition of text written on a whiteboard. As people stand, rather than sit, during writing and the arm does not rest on a table, handwriting rendered on a whiteboard is different from handwriting produced with a pen on a writing tablet. It has been observed that the baseline of handwritten text usually cannot be interpolated with a simple polynomial up to degree 2. Furthermore, the size and width of the characters often tend to become smaller as the writer moves to the right.

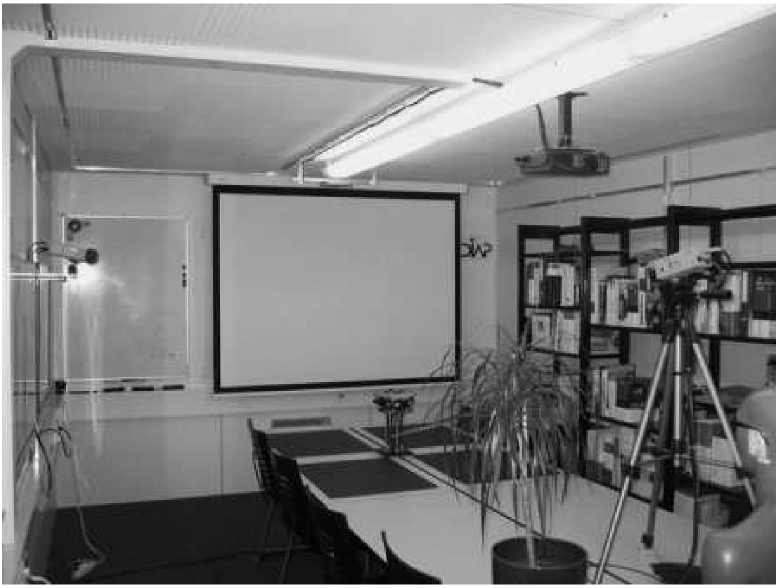

Fig. 1.1 Picture of the IDIAP Smart Meeting Room with the whiteboard to the left of the presentation screen

This chapter first describes the motivation of the work described in this book and embeds the topic into the context of smart meeting room research in Section 1.1. Second, in Section 1.2 a detailed overview of related research in handwriting recognition is presented including recent work and studies. Next, Section 1.3 states issues that need to be considered when comparing the results of different recognition systems with each other. Other topics in the area of handwriting are outlined in Section 1.4. Following, Section 1.5 describes the contributions of this book, and finally, Section 1.6 outlines the remaining chapters.

1.1 Motivation

The main motivation of this book is to develop smart meeting room applications [McCowan *et al.* (2005); Moore (2002); Reiter and Rigoll (2004); Waibel *et al.* (2003)] which are described in Section 1.1.1. Handwriting recognition of whiteboard notes can also be applied to recorded lectures. For example, the E-Chalk system [Friedland *et al.* (2004); Rojas *et al.*

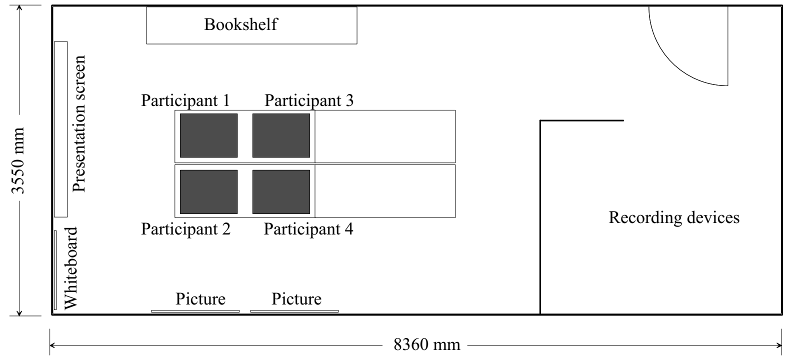

Fig. 1.2 Schematic overview of the IDIAP Smart Meeting Room (top view)

(2001)] is a platform independent recording system for lectures involving an electronic whiteboard. While recognition systems for specific data, such as mathematical formulas [Tapia and Rojas (2003)] or symbols [Liwicki and Knipping (2005)], already exist, a recognition system for unconstrained handwriting is still an open issue.[1]

Throughout this book linguistic knowledge is supplied to the recognition system in forms of a lexicon and a statistical language model. This is mainly motivated by the observation that humans also have a higher reading performance if they know the language. A detailed description and an example are given in Section 1.1.2.

1.1.1 *Smart Meeting Rooms*

In smart meeting room applications not only speech and video data of a meeting are recorded, but also notes written on a whiteboard are captured. The aim of a smart meeting room is to automate standard tasks usually performed by humans in a meeting. These tasks include, for instance, note taking and extracting important issues of a meeting. To accomplish these tasks, a smart meeting room is equipped with synchronized recording interfaces for audio, video, and handwritten notes. Figure 1.1 shows an example picture of a smart meeting room.

The challenges in smart meeting room research are manifold. In order

[1]The authors are aware that there exist commercial software for whiteboards, such as the whiteboard system from SMART© Technologies. However, these systems are often based on the Microsoft© HWR, which is also used in this book (see Chapter 6.

to allow indexing and browsing of the recorded data [Wellner *et al.* (2004)], speech [Morgan *et al.* (2001)], handwriting [Liwicki and Bunke (2005a)], and video recognition systems [Fasel and Luettin (2003)] need to be developed. Another task is the segmentation of the meeting into meeting events. This task can be addressed by using single specialized recognizers for the individual input modalities [Reiter and Rigoll (2004)] or by using the primitive features extracted from the data streams [McCowan *et al.* (2005)]. Further tasks are the extraction of non-lexical information such as prosody, voice quality variation, and laughter. To authenticate the meeting participants and to assign utterances and handwritten notes to their authors, identification and verification systems have been developed. They are based on speech [Mariéthoz and Bengio (2002)] and video interfaces [Grudin (2000); Sanderson and Paliwal (2003)] or on a combination of both [Czyz *et al.* (2003)].

The handwriting recognition system presented in this book has been developed for the IDIAP Smart Meeting Room [Moore (2002)]. This meeting room is able to record meetings with up to four participants. It is equipped with multiple cameras, microphones, electronic pens for note-taking, a projector, and an electronic whiteboard. A schematic overview of this meeting room is presented in Fig. 1.2. Note that the picture in Fig. 1.1 shows the same meeting room. The camera is located in the lower right corner of the schema, facing the whiteboard and the presentation screen on the left hand side.

The whiteboard shown in Figs. 1.1 and 1.2 is equipped with the eBeam[2] system, which acquires text written on the whiteboard in electronic format. A normal pen in a special casing is used to write on the board. The casing sends infrared signals to a triangular receiver attached in one of the corners of the whiteboard. The acquisition system outputs a sequence of (x, y)-coordinates representing the location of the pen-tip together with a time stamp for each location. An illustration of the data acquisition process is shown in Fig. 1.3.

1.1.2 *Human Performance*

If a human person is asked to transcribe a handwritten text, a high recognition performance is expected. Often the human recognition performance is seen as an upper bound on any automatic transcription system. The goal of the automated recognition is to process the data with the same or

[2]eBeam© system by Luidia©, Inc. – www.e-Beam.com

Fig. 1.3 Recording session with the data acquisition device positioned in the upper left corner of the whiteboard

nearly the same accuracy as humans [Bunke (2003); Plamondon and Srihari (2000); Vinciarelli (2002)].

However, a high performance is only achieved by the human person if he or she is familiar with the underlying language of the handwriting. To demonstrate this effect, an example text written by the same writer is presented in Figs. 1.4 and 1.5. The example is taken from "The Feynman Lectures" [Feynman *et al.* (1977)] available in English and in Hungarian. If the reader is not familiar with Hungarian the word recognition rate will drop to about 30 % in the second text, while it will be nearly perfect for the first text. Note that the # symbol in the transcription denotes that the writer has crossed out a word.

This example shows that the quality of the transcription is highly dependent on the linguistic background and the capability of understanding the text. It also follows that the recognition accuracy would be poor if only a character by character recognition is performed. Therefore it is wise to utilize linguistic knowledge in automatic transcription systems to improve the recognition performance. Generally speaking, the task specific knowledge has a significant impact on the resulting recognition performance.

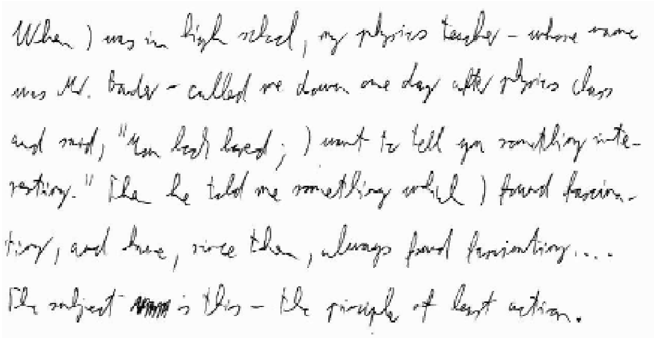

Fig. 1.4 Handwritten English text (transcription below)

When I was in high school, my physics teacher - whose name
was Mr. Bader - called me down one day after physics class
and said, “You look bored; I want to tell you something interesting.”
Then he told me something which I found fascinating,
and have, since then, always found fascinating....
The subject # is this - the principle of least action.

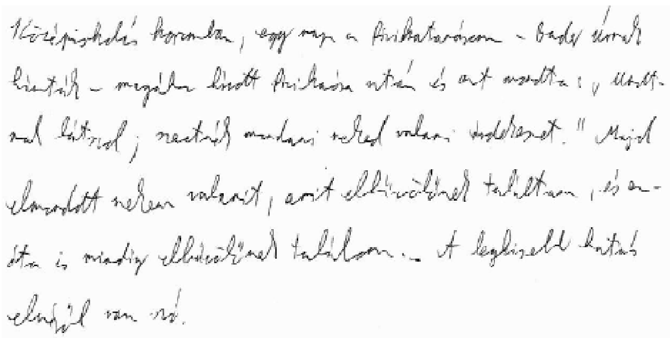

Fig. 1.5 Handwritten Hungarian text written by the same writer as in Figure 1.4 (transcription below)

Középiskolás koromban, egy nap a fizikatanárom - Bader úrnak hivták - magához hívott fizikaóra után és azt mondta: “Unottnak látszol; szeretnék mondani neked valami érdekeset.” Majd elmondott valamit, amit elbûvlõnek találtam, és azóta is mindig elbûvölõnek találom ... A legkisebb hatás elvérõl van szó.

1.2 Handwriting Recognition

Handwriting recognition is the transcription of handwritten data into a digital format. It is a classical pattern recognition problem. The task lies in assigning a pattern to one class out of a set of classes [Duda *et al.* (2001)], for example, assigning a handwritten sample to a specific character. As stated above, the goal of the automated recognition is to process the data with the same or nearly the same accuracy as humans. Although automated handwriting recognition has been a research topic for more than forty years [Bunke (2003); Plamondon and Srihari (2000); Vinciarelli (2002)], there are still many open challenges in this field, especially in the domain of unconstrained handwritten text line recognition.

The handwritten data may be present in online or offline format. In the case of online recognition, a time ordered sequence of coordinates, representing the pen movement, is available. This may be produced by any electronic sensing device, such as a mouse, an electronic pen on a tablet, or a camera recording gestures. In the case of offline recognition, only the image of the text is present, which usually is scanned or photographed from paper.

In this section an overview of offline and online handwriting recognition and its applications is presented. First, Section 1.2.1 describes the main units for recognition systems. Next, in Section 1.2.2 related work for the offline case is given and discussed. Finally, the online case is treated in Section 1.2.3.

1.2.1 *Recognition System Overview*

A recognition system for unconstrained Roman script is usually divided into consecutive units which iteratively process the handwritten input data to finally obtain the transcription. The main units are illustrated in Fig. 1.6 and summarized in this section. Certainly, there are differences between offline and online processing, but the principles are the same. Only the methodology for performing the individual steps differs.

The first unit is preprocessing, where the input is raw handwritten data and the output usually consists of extracted text lines. The amount of effort that need to be invested into the preprocessing depends on the given data. If the data have been acquired from a system that does not produce any noise and only single words have been recorded, there is nothing to do in this step. But usually the data contains noise which need to be removed

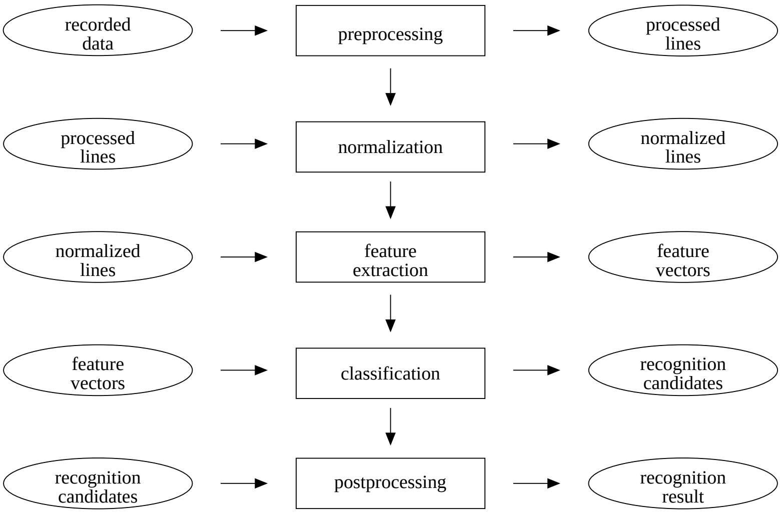

Fig. 1.6 Recognition system overview

to improve the quality of the handwriting. Another preprocessing step is the extraction of the handwritten text lines [Kavallieratou *et al.* (2002); Li *et al.* (2006); Liwicki *et al.* (2007b); Manmatha and Rothfeder (2005); Yu and Jain (1996)], which may also include the separation of text and graphics [Rossignol *et al.* (2004); Shilman *et al.* (2003); Wenyin (2003)]. Depending on the recognition system, the preprocessing may also include word extraction [Kavallieratou *et al.* (2002); Kim *et al.* (2002); Liwicki *et al.* (2006a); Mahadevan and Srihari (1996); Varga and Bunke (2005)] and even character segmentation [Kavallieratou *et al.* (2002); Tappert *et al.* (1990)].

However, the task of character segmentation is especially difficult if considered separately. This is because a word often can not be correctly segmented before it has been recognized, and can not be recognized without previously segmenting it into characters. This phenomenon is known as Sayre's paradox [Sayre (1973)].

After preprocessing, the recorded data need to be normalized. This is a very important part of handwriting recognition systems, because the writing styles of the writers differ with respect to skew, slant, height, and width of the characters. In the literature there is no standard way of normalizing the data, but many systems use similar techniques. First, the text line is corrected in regard to its skew, i.e., it is rotated, so that the baseline is

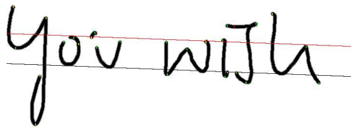

Fig. 1.7 Baseline and corpus line of a part of an example text line

parallel to the x-axis. Then, slant correction is performed so that the slant becomes upright. The next important step is the computation of the baseline and the corpus line (see Fig. 1.7). These two lines divide the text into three areas: the upper area, which mainly contains the ascenders of the letters; the middle area, where the corpus of the letters is present; and the lower area with the descenders of some letters. These three areas are normalized to predefined heights. Often, some additional normalization steps are performed, depending on the domain. In offline recognition, thinning and binarization may be applied. In online recognition the delayed strokes, e.g., the crossing of a "t" or the dot of an "i", are usually removed, and equidistant resampling is applied.

In general, normalization can be seen as a special preprocessing step, i.e., the data are processed before the recognition takes place. Because of this reason the normalization is sometimes not treated separately from the preprocessing in the literature. Depending on the point of view even the following step, feature extraction, may be seen as a preprocessing operation for the classifier.

Feature extraction is essential for any handwriting recognition system, for any classifier needs numeric data as input. However, no standard method for computing the features exists in the literature. One common method in offline recognition of handwritten text lines is the use of a sliding window moving in the writing direction over the text [Marti and Bunke (2001c)]. Features are extracted at every window position, resulting in a sequence of feature vectors. In the case of online recognition the points are already available in a time-ordered sequence, which makes it easier to get a sequence of feature vectors in writing order. If there is a fixed size of the input pattern, such as in character or word recognition, one feature vector of a fixed size can be extracted [Bunke (2003); Suen *et al.* (2000)] for each pattern.

After feature extraction, the same methods can be used for offline and online recognition. The next unit is the classification of the input features.

If there is a fixed length feature vector, neural networks [Lee (1996); Oh and Suen (2002)], support vector machines [Bahlmann *et al.* (2002)], nearest neighbor classifiers [Liu and Nakagawa (2001)], and other techniques can be applied. For sequences of feature vectors, *hidden Markov models* (HMMs) [Bunke *et al.* (1995)], *time delay neural networks* (TDNNs) [Jäger *et al.* (2001)], or hybrid approaches [Brakensiek *et al.* (1999); Caillault *et al.* (2005b); Marukatat *et al.* (2001); Rigoll *et al.* (1998); Schenkel *et al.* (1995)] can be used for classification. The output is a single recognition result or a list of alternative candidates. This list usually includes a probability for each output candidate.

The last recognition step illustrated in Fig. 1.6 is the postprocessing. This step is only possible if additional knowledge about the domain is available. For example, in the case of word recognition, the output sequence of characters can be assigned to the most similar word in the dictionary. In the case of word sequence recognition, statistical language models can be applied.

1.2.2 *Offline Handwriting Recognition*

In offline handwriting recognition the data which are to be transcribed have usually been scanned and stored as an image. A large variety of problems have been considered in research, and also some commercial systems have been developed. For example, there exist systems for postal address reading [Srihari (2000)] as well as for bank check [Impedovo *et al.* (1997)] and forms processing [Gopisetty *et al.* (1996); Ye *et al.* (2001)]. These systems take advantage of prior knowledge, working in narrow domains with limited vocabularies, where task specific knowledge and constraints are available.

In this section recent studies in offline handwriting recognition are described. Many systems have been developed in the last few decades, and one could think that the problem of offline handwriting recognition should already be solved. However, there is an increasing interest in the research community, as, in fact, many issues are unsolved in this field. This section focuses on some aspects only; for a complete overview of early work in this domain see [Bunke (2003); Plamondon and Srihari (2000); Steinherz *et al.* (1999); Suen *et al.* (2000); Vinciarelli (2002)].

One research direction in recent years is the combination of multiple classifiers. A study combining twelve different classifiers and evaluating several combination schemes has been published in [K. Sirlantzis and Hoque (2001)]. An in-depth investigation of different ensemble methods based on modifications of the training set has been performed in [Günter and Bunke (2004b)], where an improved recognition rate has been achieved for sufficiently large training sets. In [Bertolami and Bunke (2005)] multiple classifier system techniques have been used for the first time to unconstrained handwritten text line recognition. A new language model based approach has been used to automatically generate classifiers, and the ROVER [Fiscus (1997)] scheme for combination has been applied to handwritten word sequence recognition.

Many studies focus on improving existing classifiers. New algorithms have been published for each stage of the general recognition system depicted in Fig. 1.6 . In the preprocessing stage some recent works concentrate on image enhancement [Shi and Govindaraju (2004)]. For normalization, a nonuniform slant correction method has been proposed in [Taira *et al.* (2004)]. A novel 2D approach for feature extraction and for classification has been introduced in [Chevalier *et al.* (2005)]. There are many other approaches for recognition which have been published recently [Brakensiek *et al.* (1999); Choisy and Belaïd (2000); Rigoll *et al.* (1998)]. Consequently there is a wide spectrum of classifiers available.

A specific topic is automated word spotting, where a given word need to be found in handwritten documents. This is useful to search in a large number of handwritten manuscripts for which the transcription is not available. Several new approaches have been proposed in recent years [Ball *et al.* (2006); Rath and Manmatha (2003)]. Related to this topic is the problem of mapping a given transcription to handwritten text [Huang and Srihari (2006); Zimmermann and Bunke (2002a)], which is also useful for historical document indexing if the transcription, e.g., a published book based on the manuscript, is available but not linked to the handwriting.

An emerging research topic is document image analysis for digital libraries. Recently, the first workshop on this topic was held [Baird and Govindaraju (2004)]. The workshop was devoted to new technologies that help integrating scanned and encoded documents into digital libraries, so that ideally everything that can be done with encoded documents is also possible for scanned documents. An essential part of preprocessing is document layout analysis [Kennard and Barrett (2006); Srihari and Kim (1997); Srihari and Shi (2004)], where the structure of a document is investigated and the text is located. Another contribution to this workshop is a systematic approach of recognizing words in historical documents [Lavrenko *et al.* (2004)], where a fixed-length feature vector is extracted for each word and the recognition is performed with hidden Markov models (HMMs). The topic of text alignment also plays an important role for digital libraries [Kornfield *et al.* (2004)].

1.2.3 *Online Handwriting Recognition*

In online handwriting recognition the data to be transcribed are acquired with an electronic interface, such as a mouse or an electronic pen on a tablet. Thus a time ordered sequence of points representing the location of the tip of the pen is available. In some cases, even the pressure, the tilt, and the angle of the pen are known for each point. During the last two decades a significant growth of activities in online handwriting recognition research has occurred. For the task of isolated character and digit recognition, high recognition rates have been reported. In this field highly accurate commercial systems have also become available, such as recognizers running on PDAs. However, in the case of general word or sentence recognition, where no constraints are given and the lexicon is large, the state of the art is still limited and recognition rates are rather low. While this section gives an overview of important milestones and recent work, complete surveys are available in [Tappert *et al.* (1990); Plamondon and Srihari (2000)].

In the field of online isolated character recognition much research has been done [Tappert *et al.* (1990)]. A writer-dependent method for symbol recognition has been proposed in [Wilfong *et al.* (1996)], where the user defines arbitrary symbols that are to be recognized. This method assumes that the user always writes a given symbol in the same stroke order and direction. Motivated by the domain of speech recognition, HMMs have also been applied to isolated character recognition. In [Hu *et al.* (2000)], for example, a writer-independent approach has been proposed. It was evaluated on the UNIPEN database [Guyon *et al.* (1994)], which also served as a test bed in the following references of this paragraph. A hybrid technique called *cluster generative statistical dynamic time warping* (CSDTW) has been introduced in [Bahlmann and Burkhardt (2004)]. CSDTW combines dynamic time warping with HMMs. It embeds clustering and statistical sequence modeling in a single feature space. Recently, in [Bahlmann *et al.* (2002)] an approach based on *support vector machines* with a novel Gaussian dynamic time warping kernel has been proposed, which gives similar accuracies as the HMM-based approach. Depending on the number of characters in the test set, the error rates range from 3 % (digits) to about 10 % (lower case characters).

Compared to cursive handwritten word and sentence recognition, the task of isolated character recognition is considered to be rather simple. In spite of Sayre's paradox, many segmentation-based approaches have been developed. Common approaches for character segmentation are based on unsupervised learning and data-driven knowledge-based methods [Plamondon and Srihari (2000)]. Other strategies first segment the text into basic strokes rather than characters. These methods are usually based on special techniques, such as splitting a word at the minima of the velocity, at the minima of the y-coordinates, or at the location of maximum curvature. For example, an approach which first segments the data at the points of the minima of the y-coordinates and then applies self-organizing maps has been presented in [Schomaker (1993)]. Another approach [Kavallieratou *et al.* (2002)] uses the minima of the vertical histogram for an initial estimation of the character boundaries and then applies various heuristics to improve the segmentation.

To avoid the problem of segmenting words into characters, some systems have been developed which use the words directly as target recognition symbols. One of these systems is described in [Wilfong *et al.* (1996)], where only a small set of 32 words was used for testing. Despite the small vocabulary, a rather high error rate of 4.5 % for writer-dependent recognition was reported. One of the main reasons is that the order of writing strokes differs even when the same writer is writing a word. For larger vocabularies the expected error is even higher. Another drawback of using the words as targets is that each word must be present in the training set, which makes the acquisition of a database unfeasible for large vocabularies. The conclusion is that it is better to model characters individually.

A widely accepted technology to overcome the problem of segmenting a word into its constituent characters is the use of HMMs [Hu *et al.* (1996)]. The idea of using HMMs is motivated by previous experiences in speech recognition [Rabiner (1989)]. Each character is represented by one single HMM. The models are then concatenated resulting in models for complete words from a given dictionary. Early work addressing the use of HMMs for online word recognition has been presented in [Bercu and Lorette (1993); Starner *et al.* (1994); Hu *et al.* (1996)]. The approach of [Hu *et al.* (2000)] proposes an advanced writer-independent HMM-based system. It combines point oriented and stroke oriented features to obtain an improved accuracy.

Recurrent neural networks (RNNs) are powerful sequence learners that are, in principle, capable of approximating any sequence-to-sequence mapping to any arbitrary precision [Haykin (1994)]. One of their main advantages is the ability to access contextual information, an important asset for handwriting recognition. This is particularly true for state-of-the-art RNN architectures, such as *bidirectional long short-term memory* (BLSTM) [Graves and Schmidhuber (2005)]. Nevertheless, there have been only few previous applications of RNNs to handwriting recognition [Bourbakis (1995)], and these have used neural networks to classify individual characters only.

To combine the advantages of neural networks and HMMs, hybrid approaches have been developed. Hybrid systems use HMMs to model long-range dependencies in the data and neural networks to provide localized classifications. Hybrid approaches which use RNNs combined with HMMs have been proposed in [Senior and Robinson (1998); Rigoll *et al.* (1998); Schenk and Rigoll (2006)]. There exist also a number of hybrid approaches based on other forms of neural networks. Classical neural networks have been used for state modeling in [Marukatat *et al.* (2001)]. Time delay neural networks combined with HMMs have been applied to online word recognition in [Schenkel *et al.* (1995)]. The application of a similar hybrid approach as in [Schenkel *et al.* (1995)] has been proposed in [Jäger *et al.* (2001)] on an advanced feature set. This application has further been improved with pruning techniques, enabling it to be used in real-time with large dictionaries. A more detailed analysis of TDNN/HMM systems is given in [Caillault *et al.* (2005a)].

Another important issue in the field of online handwriting recognition is the extraction of handwritten text. In [Shilman *et al.* (2003)] a system for extracting text from documents written on a tablet PC has been presented. This system uses the time stamps to derive an initial guess about the text line locations, and groups the lines and blocks afterwards. Then a set of local and global features is extracted and classified using a decision tree. A similar task is the detection of modes, e.g., drawings, text, and gestures, in an interactive pen-based system. A system that distinguishes between three modes (*drawing*, *text*, and *gesture*) has been presented in [Rossignol *et al.* (2004)].

Recently, approaches for combining online and offline handwritten data have been proposed. The combination of online and offline Japanese characters has been studied in [Velek *et al.* (2003)]. For isolated digits a combination has been investigated in [Vinciarelli and Perrone (2003)]. Both approaches show a significant increase in recognition performance. Hence, combining online and offline recognizers for handwritten text lines is a promising idea, which is also investigated in this book.

1.3 Comparability of Recognition Results

In the field of handwriting recognition a large number of studies have been published. The reported recognition results range from values lower than 50 % up to nearly 100 %. Sometimes even a perfect recognition rate has been achieved in specific environments. However, it is nearly impossible to identify a universally best recognition system. In general, a single recognition system which performs best for all tasks cannot exist, for each level of generalization causes a loss of knowledge in specific domains. For example, if a postal address in a specific country needs to be recognized, the system can take advantage of the knowledge of possible city and zip code relations to improve the accuracy. Of course, it may be possible to apply an initial preprocessing step recognizing the specific domain first, but all preprocessing steps produce errors, which limits the possible overall recognition accuracy.

In short, three problem areas in comparing results from different systems with one another exist: the considered recognition task, which defines the overall difficulty of the problem; the data set, which usually differs from other data sets in quality and quantity of the data; and the amount of data used for training and testing. In the remainder of this section a detailed description of the three problems is provided.

The performance of a system crucially depends on the considered task. For example, in the case of isolated digit recognition the performance is usually higher than in unconstrained handwritten text line recognition. When comparing recognition results, one must be aware of the following aspects:

- *number of classes*: With an increasing number of classes the task becomes more difficult. In character recognition, for example, it is assumed that recognizing ten different digits produces higher classification results than recognizing all the 25×2 letters of the Latin alphabet.
- *input (isolated vs. sequence)*: The task becomes easier if the boundaries of the characters are known, in which case a recognition system for isolated characters can be applied instead of a word recognizer.
- *vocabulary size*: As a rule of thumb, a higher performance can be expected for smaller vocabulary sizes under the constraint that the vocabulary covers all data in the test set.

- *number of writers*: The most difficult task is writer-independent recognition, i.e., when there is no training data available of the writers represented in the test set. For writer-dependent recognition it must be considered whether there is one recognition system for only one writer or for a number of writers.
- *language model*: Recognition systems can gain additional information from statistical language models. One can expect higher recognition rates from systems utilizing a language model.

These elementary facts are well known in the research community. However, there are many publications which do not clearly state all of the relevant details and therefore make it difficult to use them as a reference.

The second important issue is the data set used for evaluating the system. If the database is not publicly available, it is impossible to make a direct comparison of the results. Therefore it is important that existing databases for training and testing are shared in the research community. There are some databases which are publicly available. The UNIPEN database [Guyon *et al.* (1994)] is a large online handwriting database. It contains mostly isolated characters, single words, and a few sentences on several topics. Another online word database is IRONOFF [Viard-Gaudin *et al.* (1999)]. It additionally contains the scanned images of the handwritten words. For the task of offline handwriting recognition there are also databases available, including CEDAR [Hull (1994)], created for postal address recognition, NIST [Wilkinson *et al.* (1992)], containing image samples of hand printed characters, CENPARMI [Lee (1996)], consisting of handwritten numerals, and the IAM-DB [Marti and Bunke (2002)], a large collection of unconstrained handwritten sentences. All these databases vary in the amount of data, the quality of the raw data, the quality of the transcription, and also in the range of possible applications. Therefore it should be considered which database, or which part of a database, is to be used to train and test a system when comparing performance results.

Even if the same data have been used and the same recognition task has been conducted, there may be differences in the difficulty of the task. Usually the training and test data are randomly selected from the database causing different recognition results. Another possible difference is the amount of data used for training and testing. Common experience is that the larger the training set is, the better the recognizer performs. An example of problematic comparability between several systems is the UNIPEN database. Many results have been reported on this database, but in most cases the data have been manually preprocessed and manually selected, which makes a comparison of the results difficult [Vuurpijl *et al.* (2004)].

Because of all these problems, it is useful to define benchmark tasks where all recognition conditions are described as precisely as possible. The data should be divided at least into a training and a test set, and possibly a validation set for validating training parameters. Both the recognition classes and the dictionary need to be listed.

1.4 Related Topics

In the field of document analysis there are several other topics related to handwriting recognition. This section addresses some of them.

The first important topic is writer identification and verification based on online and offline data, including signature verification. The task of writer identification is to assign a handwritten text sample to one specific writer out of a given set of writers, while the task of writer verification is to determine whether a given handwritten sample or a signature stems from a claimed writer or is a forgery. Surveys in this domain are given in [Gupta and McCabe (1997); Leclerc and Plamondon (1994); Plamondon and Lorette (1989); Plamondon and Srihari (2000)]. In recent works, various approaches based on dynamic time warping, neural networks, hidden Markov models, and Gaussian mixture models have been investigated [Feng and Wah (2003); Liwicki *et al.* (2006b); Richiardi and Drygajlo (2003)]. It is evident that writer identification can help the process of handwriting recognition, for if the writer is known, a writer-specific recognizer can be applied for recognition. A higher recognition accuracy can be expected from such a specific recognizer.

Another related topic is the recognition of hand-sketched figures. It has been a research topic for more than 20 years [Murase and Wakahara (1986)], and a number of systems using sketched inputs for various types of applications have been developed. The DENIM application [Lin *et al.* (2000)], for example, allows users to build web pages by drawing, SketchySPICE [Hong and Landay (2000)] is a simple-circuit CAD tool, Tahuti [Hammond and Davis (2002)] is used for creating UML diagrams by sketches, and ASSIST [Alvarado and Davis (2006)] is a sketch-based CAD tool. A recognition system for hand-drawn diagrams that uses conditional random fields has been proposed in [Szummer and Qi (2004)]. For the E-Chalk system mentioned above, applications for the animation of algorithms [Esponda

Argüero (2004)], the simulation of biological and pulse-coded neural networks [Knipping (2005); Krupina (2005)], and the simulation of logic circuits [Liwicki and Knipping (2005)] have been realized.

A population of individuals can often be partitioned into sub-categories based on various criteria. Dividing a population into sub-categories is an interesting research topic for numerous reasons, for example, if one is only interested in one specific sub-category, or if specifically processing each sub-category leads to improved results. Especially the classification of gender from handwriting has been a research topic for many decades [Broom *et al.* (1929); Newhall (1926); Tenwolde (1934)]. Some automatic systems have been proposed in [Cha and Srihari (2001)], [Bandi and Srihari (2005)], and [Liwicki *et al.* (2007c)].

1.5 Contribution

The main contribution of this book is the development of recognition systems for online handwritten text written on a whiteboard. This is a novel task in the research community. A similar task has been considered in [Fink *et al.* (2001); Munich and Perona (1996)]. However, in [Fink *et al.* (2001); Munich and Perona (1996)] a video camera was employed to capture the handwriting, whereas the eBeam© interface based on infrared sensing is used in this book. This system is easier to use than a video camera and is less vulnerable to artifacts arising from poor lighting conditions, self-occlusion and low image resolution.

To be more specific, in the context of the research described in this book, a novel online handwritten database has been compiled, and four individual handwriting recognition systems have been developed. The four systems consist of an offline and an online recognition system, a system combining offline and online data, and a writer-dependent recognition system. A short summary of the contributions is provided in this section.

During the work described in this book the IAM Online Handwriting Database (IAM-OnDB) has been compiled [Liwicki and Bunke (2005b)]. This database is a large publicly available collection of online handwritten English text acquired from a whiteboard. In addition to the recorded data and its transcription, some information about the writers, which could be useful for future work, is stored in the IAM-OnDB. Two benchmark tasks have been defined for this database for comparison issues.

Moreover, an offline recognition system for handwritten text has been developed [Liwicki and Bunke (2005a)] which includes the optimization of several training parameters as well as the integration of a statistical language model. This system has been further investigated with respect to the influence of different training sets, and the size of the training data [Liwicki and Bunke (2007c)].

Next, a new online recognition system has been introduced [Liwicki and Bunke (2006)]. This system includes novel preprocessing and normalization strategies which have been developed especially for whiteboard notes. A recently introduced classification strategy based on bidirectional long short-term memory networks has been applied for the first time in the field of handwriting recognition [Liwicki *et al.* (2007a)].

A combination of online and offline approaches for handwriting recognition is investigated in this book as well [Liwicki and Bunke (2007a)]. In some initial experiments three HMM-based recognition systems have been combined. In a broader experimental study, more recognition systems have been included in the combination, involving external recognizers from Microsoft© and Vision Objects©.

Another main contribution of this book is the generation of writer-dependent recognition systems [Liwicki and Bunke (2008)]. For that purpose a writer identification system, which has been developed for offline writer identification, has been adapted to online features [Liwicki *et al.* (2006b); Schlapbach *et al.* (2008)]. Furthermore, the writer identification system has been used to investigate the problem of automatic handwriting classification [Liwicki *et al.* (2007c)].

1.6 Outline

The structure of this book is as follows. Chapter 2 reviews general methods used throughout this book. These methods are not a novel contribution, but their main ideas are important to understand the work described in the other chapters.

In Chapter 3 the main resources that have been used for training and testing the recognizers are introduced. One database, namely the IAM Online Handwriting Database, has been specifically compiled for the research described in this book. Thus this database is explained in more detail. Further resources for post-processing are also described in Chapter 3.

The offline recognition approach is introduced in Chapter 4. Various experiments with this recognition system and extended versions are reported.

The first results may be seen as baseline results for the experiments described in the remainder of this book.

Chapter 5 is devoted to the online recognition system. The description includes preprocessing, enhanced normalization for whiteboard notes, and feature extraction techniques. The results of various experiments on these features are discussed. The chapter also reports on feature selection experiments using the proposed feature set.

The combination of online and offline classifiers for handwritten text lines is investigated in Chapter 6. First, the focus is on combination and voting strategies. Next, experiments with several classifiers and combinations schemes are reported.

An approach based on writer-dependent recognition is introduced in Chapter 7. For this purpose, a writer identification system for handwritten whiteboard notes is proposed. Experiments based on an adaptation of the online recognizer are then reported. Furthermore, Chapter 7 also describes the handwriting classification experiments.

Finally, Chapter 8 provides the main conclusions of this book, shortly summarizes the results, and gives an outlook on future research.