Camera-Based Whiteboard Reading: New Approaches to a Challenging Task

Thomas Plötz
Robotics Research Institute
Dortmund University of
Technology, Germany
Thomas.Ploetz@udo.edu

Christian Thurau
Department of Computer Science
Dortmund University of
Technology, Germany
Christian.Thurau@udo.edu

Gernot A. Fink
Department of Computer Science
Dortmund University of
Technology, Germany
Gernot.Fink@udo.edu

Abstract

We suggest a system for recognizing handwriting text from a whiteboard. In contrast to the mainstream approaches, we are able to recognize text solely from still images and do not require additional hardware or tracking of the writer's hand. The proposed system contains extraction of text areas, suitable image preprocessing steps, line extraction, and finally text recognition. Since we are dealing with realistic, and thus very difficult data, a special emphasize of this contribution lies on the preprocessing steps. Experimental results on a large, challenging benchmark set clearly justify further investigations of the proposed approach. Despite the difficulty of the benchmark data used, we come close to a reference approach operating on rendered, near perfect synthetic data.

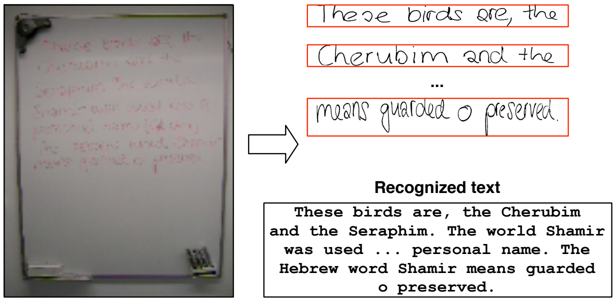
Keywords: camera-based document recognition, offline handwriting recognition, whiteboard reading, evaluation on IAM-OnDB

1. Introduction

This paper deals with the task of automatically recognizing handwritten notes taken on a whiteboard. In contrast to related approaches we tackle this challenging problem by a purely camera-based approach which is illustrated in figure 1.

Despite its popularity in computer aided collaborative working environments, the domain of automatic note recognition still provides various challenges. Interestingly, existing systems are limited to the usage of specialized pen trackers or similar hardware (e.g. the eBeam system [6]). While these systems work quite well, they are rather restrictive and suffer from certain obvious flaws which prevent natural interaction, e.g. one has to use a special eraser in order not to confuse the system. In our research, we approach the topic by means of offline, solely camera-based recognition.

The contribution of this paper is twofold. First, we will introduce a system that is able to **(a)** detect text regions, **(b)** segment and extract text lines, and finally **(c)** recognize handwritten notes, based on camera input. Second, in contrast to previous work [5, 16], we considerably improve robustness towards low quality, highly distorted images as they are often met under real-life conditions.



These birds are, the Cherubim and the Seraphim. The world Shamir was used ... personal name. The Hebrew word Shamir means guarded o preserved.

Figure 1. Exemplary input image and recognition results of our camera-based whiteboard reading system

Previously, we assumed an idealized quality of image data which, unfortunately, cannot be expected in real-life settings. Among others, we have to deal with unclear separation of text lines, worn-out pens (low contrast), cluttered images not only containing text, and various other challenging problems (not to mention the general inability of certain subjects to write on a whiteboard at all).

For benchmarking our approach we use a challenging dataset. It contains a large number of sample images and showed to reflect the before mentioned task related problems. To our knowledge, this is the first time this dataset is used for offline image based recognition. Consequently, for sake of comparison, we will provide additional results on idealized data rendered from online trajectories captured with a pen-tracking device, which will serve as reference experiments.

The remainder of this paper is organized as follows. Related work will be discussed in the next section. In section 3, we will introduce our whiteboard reading system. Section 4 finally presents experimental results.

2. Related Work

In recent years there has been considerable progress in the field of automatic handwriting recognition. Today, large vocabulary recognition is possible with good accuracy for "well behaved" input data, i.e. scanned documents. Although such tasks are primarily associated with the reading of postal addresses, the study of recognizing handwritten texts in an OCR-like style has gained considerable interest over the last decade (cf. [7]).

Current handwriting recognizers mainly use Hidden Markov Models (HMMs, cf. [3]) for automatically learning statistical models of character or word appearances. In contrast to postal applications in systems for reading handwritten notes also the restrictions on plausible word sequences imposed by a – mostly statistical – language model (cf. [3]) play an important role (cf. e.g. [12, 14]).

Camera-based recognition of notes written on a whiteboard using offline recognition techniques was first proposed in [15] and refined in [16]. The challenges in whiteboard reading arise mainly from the reduced quality of the document images captured. Furthermore, larger variations in writing style can be expected as writing on a whiteboard is an unfamiliar writing situation for most people.

Recently, a large database of notes taken on a whiteboard was collected at the University of Bern, Switzerland, using a pen-tracking device [9]. Besides online recognition results published for this task also offline recognition experiments in an idealized setting are reported in [10]. For those, the pen trajectories captured online were used to render ideal text line images of the data assuming a fixed diameter black ink pen.

3. Camera-Based Whiteboard Reading in a Real-Life Setting

In order to successfully perform camera-based whiteboard reading in the aforementioned real-life setting we substantially enhanced our baseline recognition system towards increased robustness. Figure 2 gives an overview of the system with its particular processing stages[1]. In the following, the key issues of the enhanced recognition system will be discussed:

3.1: text detection (extraction of region of interest containing text fragments)

3.2 - 3.3: preprocessing and image normalization (contrast enhancement, and global skew compensation)

3.4 - 3.5: line extraction (plus text line normalization)

3.6 - 3.8: feature extraction and handwriting recognition (writing model, language model, integrated search).

3.1. Text Detection

For whiteboard reading still images of the whiteboard containing handwriting are captured using a standard digital camera. As can be seen in figure 1 in addition to the actual handwriting these images also contain clutter, for example the whiteboard's frame, pens, or the eBeam device used for on-line recognition (not addressed by this paper). For text recognition, in an initial step the handwriting data needs to be extracted and clutter removed.

In these premises the first stage of our processing pipeline aims at the automatic extraction of the particular regions of interest (ROI) containing text lines to be recognized. We, therefore, apply a slightly modified version of the winning contribution of the ICDAR 2005 text locating competition [11]. The greyscale version of the original image is median filtered and binarized using modified local Niblack thresholding which is the pre-requisite for the extraction of connected components. Following this, the edge density of the image is calculated using the results of Sobel filtering.

For the actual text detection the probability for the region of a connected component being text or background is then calculated as the product of the following scores: contrast, ink density, overlap between foreground (the connected component's region) and background (bounding box of the connected component minus foreground), homogeneity of fore- and background, respectively, edge density, and size of the particular text region. By means of a globally optimized threshold comparison, regions of connected components are labeled as text or background. Foreground regions are grouped in a single text region representing the ROI.

3.2. Contrast Enhancement

The whiteboard document images considered in this study frequently exhibit a poor contrast between ink and background, which is mainly due to the use of bright green or worn out pens. Therefore, normalization of the ROIs for enhancing ink contrast is a primary concern in preprocessing. In order to achieve this goal with a minimum of task dependent parametrization, we apply a modified version of the color normalization scheme proposed in [2] to the grey scale ROI images. For the estimation of the "white patch" – here the local estimate of the board color – first a maximum over the image is computed. Although the "grey world assumption" is not perfectly satisfied for document images, a local averaging filter determines regions of contrast where ink might be present on the board. From this local contrast an estimate of the grey scale dynamic range for the whole image is derived by applying

[1] The necessary methods for image processing and statistical pattern recognition are realized on the basis of the tools provided by the open source development environment ESMERALDA [4].

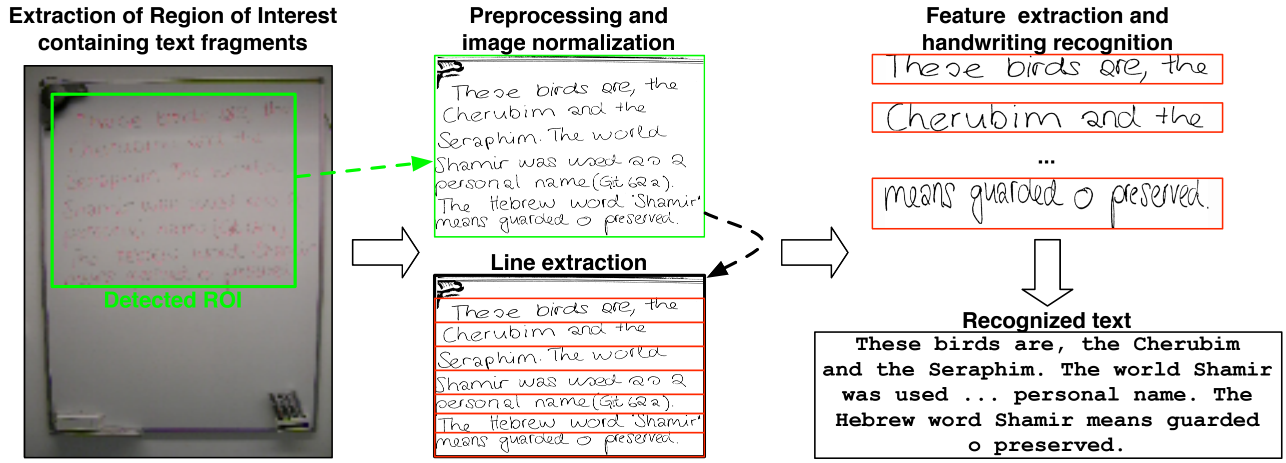

Extraction of Region of Interest containing text fragments
Detected ROI
Preprocessing and image normalization
Line extraction
Feature extraction and handwriting recognition
These birds are, the
Cherubim and the
...
means guarded o preserved.
Recognized text
These birds are, the Cherubim and the Seraphim. The world Shamir was used ... personal name. The Hebrew word Shamir means guarded o preserved.

Figure 2. Camera-based whiteboard reading – system overview

another level of smoothing. This operation is crucial as it ensures that the estimate of larger dynamic ranges in the neighborhood is used for normalizing regions with vanishing contrast. Based on the estimated average grey value and dynamic range, pixel intensities are normalized by applying a sigmoidal transfer function.

3.3. Global Skew Compensation

Once the ROIs are found and normalized w.r.t. contrast, global skew correction is performed on the respective parts of the image. In order to find the optimal transformation the binarized ROI is rotated in steps of 0.5 degrees. For every step the histogram of horizontal pixel density is calculated together with the histogram's variance. The maximum over all variances calculated at the particular rotation angles determines the optimal transformation for skew normalization.

3.4. Line Extraction

For automatic line separation within (images of) text documents often certain (global) threshold analysis of projection histograms is performed (cf. [8]).

Unfortunately, line separation techniques exploiting some sort of global thresholding are likely to fail if the lengths of the particular text lines differ substantially. Those lines shorter than the majority of lines contained by the ROI will not be extracted correctly since the threshold globally optimized on the particular projection histogram will be too high. Trying to circumvent this problem by some sort of local thresholding seems promising but, according to our practical experience, results in various special cases preventing from a general solution. Furthermore, especially if larger portions of text are written on the whiteboard, ROIs tend to be densely populated with lines whose ascenders and descenders interfere mutually. When separating lines using some kind of the aforementioned thresholding techniques these interfering parts will be damaged in both lines involved, which results in corrupt feature data and recognition will be doomed.

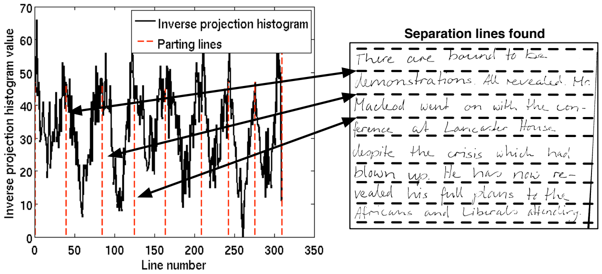


Figure 3. Line separators found within an inverse projection histogram using meanshift clustering.

In order to avoid the unnecessary damage of input data as described above we extract separation lines between text lines by means of unsupervised *meanshift clustering* [1]. As input data we use the projection histograms. Normalized projection histograms represent the probability of observing text fragments within a specific line given a document. Consequently, since we are looking for line separation, we compute the inverse projection histogram. Given a sufficient sampling of the distribution, the meanshift procedure is now used for mode finding. Every mode found can be interpreted as a line separator. The kernel bandwidth is dependent on the text size to be expected, and was verified by means of experimental validation. However, for future research we plan to exploit automatic bandwidth selection methods resulting in unsupervised, parameter-free line separation. For an example of unsupervised meanshift clustering see figure 3.

In most cases the proposed method converges to the actual number of separation lines and their positions, respectively. Moreover, due to the local clustering inherent to meanshift the problem of separation line detection for text lines with different lengths is effectively alleviated.

Given the extracted text line areas the problem of interfering ascenders and descenders is tackled by means of a connected components analysis. Connected components, which usually are (parts of) characters and extracted by analyzing the binarized image, are added to the text line, which includes their center of gravity. By means of certain (trivial) post-processing operations irrelevant text lines are excluded from further processing (e.g. a text line needs to include at least five connected components).

3.5. Text Line Normalization

After line extraction we apply the usual normalization operations to compensate for variations in local skew and slant. Additionally, the size of the binarized text line images is normalized such that the distance between local contour minima matches a preset value (cf. [16]). This size normalization is extremely important in our application. Only then models trained on scanned document images with high resolution can be applied to the recognition of low resolution text found in camera images of whiteboard data.

3.6. Writing Model

As in our previous work (cf. [16]) the appearance of handwritten characters is described by semi-continuous HMMs. We apply a sliding window approach to the normalized text lines extracted from handwritten documents. They are subdivided into a sequence of overlapping stripes of 4 pixels width and the height of the line. For each of these so-called frames a set of nine geometric features and the approximation of their first derivative over a window of 5 frames are computed (cf. [16]). Using this feature representation, models for upper and lower case letters, numerals, and punctuation symbols – 75 in total – were trained on the 485 documents of categories A to D (4222 text lines) taken from the IAM database of scanned handwritten documents [13]. All models have Bakis topology and share a codebook of 1.5k Gaussians with diagonal covariance matrices.

3.7. Language Model

In order to make knowledge about plausible word sequences available during the process of HMM decoding we combine the writing model with a word-based statistical n-gram model. The data for estimating the language model was given by the text prompts used to generate the training data of the IAM online database (IAM-OnDB) [9] consisting of 62k word tokens. On this data we estimated a bi-gram model for the 11k recognition lexicon defined for task 1 on the IAM-OnDB by applying absolute discounting and backing-off. The model achieves a perplexity of 310 on the "final" test set of IAM-OnDB task 1 (`t1f`), which can be considered a quite good result given the severely limited amount of training data.

3.8. Integrated Search

Both the HMM-based writing model and the bi-gram language model are decoded in an integrated manner using strictly time-synchronous Viterbi beam search. The recognition lexicon is compactly represented in a lexical tree. Therefore, the search process uses time-based search tree copies in order to correctly combine the HMM and n-gram scores [3, Chap. 12].

4. Results

In order to evaluate our whiteboard reading system on a larger scale in real-life settings related to the domain of collaborative working environments we conducted experiments using a large database of images of whiteboard documents. In the following the results achieved are discussed.

4.1. Data Sets

The experimental evaluations of our previous work on automatic whiteboard reading were based on the analysis of a test set of handwriting recorded at our previous affiliation (cf. [16] for details). Since the recording and labeling of such data requires a substantial manual effort the data set collected was of quite moderate size only.

Originally, the IAM-OnDB [9] was intended for collecting online handwriting data on a massive scale comparable to the previous efforts in building the offline version [13]. For the majority of documents only the online trajectories captured by the pen-tracking system were recorded. Fortunately, for the major part of the "final" test set of task 1 (`t1f`) also images of the final whiteboard documents were taken with a digital camera and made available to us by the authors of [9].

This large collection of whiteboard documents (referred to as `t1f-wb` in the following) served as a realistic benchmark in our recognition experiments. It consists of 491 documents written by 62 subjects without any constraints w.r.t. writing style. Similar to the IAM-DB the text written on the whiteboard is based on prompts taken from the LOB-corpus. For the recording of the final document images the whiteboard was captured "as is" not focusing on standardized constraints and text-only shots. An exam-

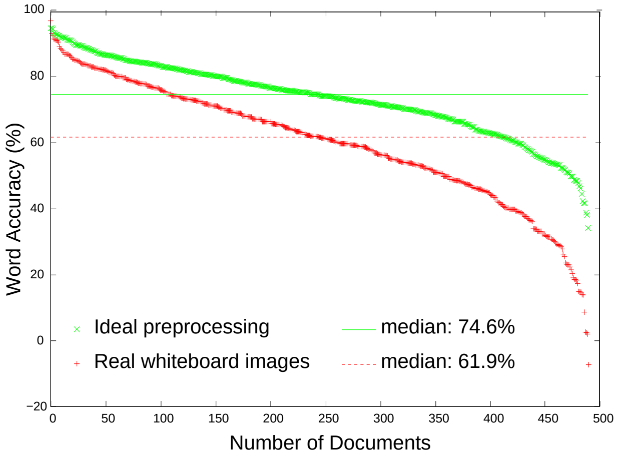


Figure 4. Comparison of word accuracies achieved on document level (both sorted in descending order)

ple image is shown in figure 1.

When working on such a challenging data set only, it would not become clear what results could have been expected with optimum processing and modeling. For the part of preprocessing document images such ideal data is part of IAM-OnDB. Besides the online trajectories there also offline data is available that was generated by rendering ideal text line images from the trajectory data assuming some fixed diameter black ink pen. In the following results on this data will serve as a reference for the best possible results that could have been achieved with optimum preprocessing.

4.2. Experiments

In a first experiment, we evaluated the capabilities of our recognition system based on the analysis of the aforementioned rendered data from IAM-OnDB. The word accuracy achieved for the complete test set is 73.5 percent, and 73.7 for the reduced set for which whiteboard images are available (`t1f-wb`). Serving as reference for all further experiments these figures compare favorably with the results published in the literature [9, 10].

The second experiment pursued was directed to actual camera-based whiteboard reading. Therefore, we applied the system as presented in this paper to the captured image data from the `t1f-wb` task of IAM-OnDB. The overall recognition accuracy achieved is 60.2 percent. Reconsidering the truly challenging quality of the images this figure indeed represents a very promising result. When considering the reference experiment as representing the optimum result – i.e. 100 percent accuracy – a relative accuracy of 81.7 percent could be achieved.

The test set consists of 3435 text lines. Our line extraction procedure generated 3587 for this data. The evaluation of recognition results was performed on the document level, as otherwise line correspondences would have to be annotated manually. Therefore, line segmentation errors are directly reflected on the level of word hypotheses mostly leading to either multiple insertions or deletions.

Table 1. Recognition results for IAM-OnDB

image data	word accuracy [%] (absolute)	relative accuracy [%] (w.r.t. reference)
rendered	73.7	100.0
captured	60.2	81.7

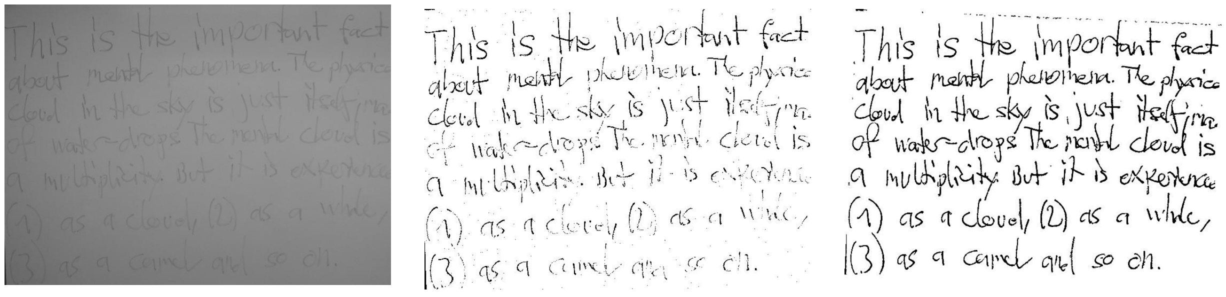
In table 1 the achieved word accuracies of both experiments are summarized. The level of significance for both experiments is ± 0.6 percent. In the right column the recognition results achieved are given w.r.t. those achieved in the idealized setting with rendered data.

For a more detailed impression of the recognition results in figure 4 the accuracies are given per document. The results are ordered according to word accuracy. It can be seen that a large portion of the whiteboard images can be treated by our system in a reasonable way. The accuracy for approximately half of the dataset is above 60 percent. For about 100 documents the accuracies drop significantly which is mainly due to poor image quality. For easier comparability the median figures for both experiments are also drawn as solid (reference: 74.6%) and dashed lines (captured images: 61.9%).

One of the biggest challenges in camera-based whiteboard reading is the low contrast inherent to certain images. Especially when worn out pens are used for writing to the whiteboard the ink is, even for humans, hardly visible. In fact the majority of images where the recognition system performs significantly worse than average are documents written with some, apparently dying, green pen. In figure 5 an example of a poor contrast document is shown together with a binarized version applying Niblack's method locally and the binarization result achieved after the proposed contrast enhancement. On the enhanced ROI image the final recognition system achieves a word accuracy of 58.3 percent which still does not match the 73.3 percent obtained with optimum preprocessing, but which is a substantial improvement with respect to the miserable failure of the original recognizer delivering only 5 percent accuracy.

5. Summary

In this paper we presented substantial advancements towards automatic camera-based whiteboard reading in real-life settings, as they are found in computer supported collaborative working environments. The main contributions are (a) a document preprocessing and normalization


This is the important fact
about mental phenomena. The physical
cloud in the sky is just itself in
of water-drops The mental cloud is
a multiplicity. But it is experience
(1) as a cloud, (2) as a whale,
(3) as a camel and so on.

Original grey scale image Binarized (local Niblack) Contrast enhanced + binarized

Figure 5. Effect of contrast enhancement on results of binarization

pipeline which is robust with respect to poor document image quality and highly distorted script and (b) the evaluation of a complete working whiteboard reading system on a challenging task of whiteboard documents collected in the setting used for building the IAM-OnDB [9]. Keeping in mind the severe differences in document quality between the images of the board's contents and idealized text lines rendered from online data, the accuracy achieved in the large vocabulary task addressed can be considered an important milestone in the area of camera-based document analysis and recognition.

Acknowledgments

This work was in part supported by the German Research Foundation (DFG) within project **Fi799/3** or **Pl554/1**, respectively.

For providing images of whiteboard documents collected for the IAM-OnDB [9] we would like to thank the Institute of Informatics and Applied Mathematics, University of Bern, namely Horst Bunke and Marcus Liwicki.

Additionally, we greatfully acknowledge the valuable support of Tobias Ramforth in developing software modules for our camera-based whiteboard reading system.

References

[1] D. Comaniciu and P. Meer, "Mean Shift: A robust approach toward feature space analysis", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2003.

[2] M. Ebner, "Combining White-Patch Retinex and the Gray World Assumption to Achieve Color Constancy for Multiple Illuminants", B. Michaelis and G. Kress, editors, *Pattern Recognition, Proc. 25th DAGM Symposium*, 2003, volume 2781 of *LNCS*, pp 60–67, Berlin. Springer.

[3] G. A. Fink, *Markov Models for Pattern Recognition*, Springer, Berlin Heidelberg, 2008.

[4] G. A. Fink and T. Plötz, "ESMERALDA: A Development Environment for HMM-Based Pattern Recognition Systems", , 2007, `http://sourceforge.net/projects/esmeralda`.

[5] G. A. Fink, M. Wienecke and G. Sagerer, "Experiments in Video-Based Whiteboard Reading", *First Int. Workshop on Camera-Based Document Analysis and Recognition*, 2005, pp 95–100, Seoul, Korea.

[6] L. Inc., "eBeam – Interactive Whiteboard Technology", Web Document, 2008, `http://www.e-beam.com/`.

[7] A. L. Koerich, R. Sabourin and C. Y. Suen, "Large Vocabulary Off-Line Handwriting Recognition: A Survey", *Pattern Analysis and Applications*, 6(2):97–121, 2003.

[8] L. Likforman-Sulem, A. Zahour and B. Taconet, "Text line segmentation of historical documents: a survey", *Int. Journal on Document Analysis and Recognition*, 9(2):123–138, 2007.

[9] M. Liwicki and H. Bunke, "IAM-OnDB – An On-Line English Sentence Database Acquired from Handwritten Text on a Whiteboard", *Proc. Int. Conf. on Document Analysis and Recognition*, 2005, volume 2, pp 956–961, Seoul, Korea.

[10] M. Liwicki and H. Bunke, "Combining On-Line and Off-Line Systems for Handwriting Recognition", *Proc. Int. Conf. on Document Analysis and Recognition*, 2007, pp 372–376, Curitiba, Brazil.

[11] S. M. Lucas, "ICDAR 2005 text locating competition results", *Proc. Int. Conf. on Document Analysis and Recognition*, 2005, volume 1, pp 80–84.

[12] U.-V. Marti and H. Bunke, "On the Influence of Vocabulary Size and Language Models in Unconstrained Handwritten Text Recognition", *Proc. Int. Conf. on Document Analysis and Recognition*, September 2001, pp 260–265, Seattle.

[13] U.-V. Marti and H. Bunke, "The IAM-Database: An English Sentence Database for Offline Handwriting Recognition", *Int. Journal on Document Analysis and Recognition*, 5(1):39–46, 2002.

[14] A. Vinciarelli, S. Bengio and H. Bunke, "Offline Recognition of Unconstrained Handwritten Texts Using HMMs and Statistical Language Models", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(6):709–720, 2004.

[15] M. Wienecke, G. A. Fink and G. Sagerer, "Towards Automatic Video-based Whiteboard Reading", *Proc. Int. Conf. on Document Analysis and Recognition*, 2003, pp 87–91, Edinburgh, Scotland. IEEE.

[16] M. Wienecke, G. A. Fink and G. Sagerer, "Toward Automatic Video-based Whiteboard Reading", *Int. Journal on Document Analysis and Recognition*, 7(2–3):188–200, 2005.