Towards semi-supervised transcription of handwritten historical weather reports

Jan Richarz
Department of Computer Science
TU Dortmund, Germany
Email: jan.richarz@udo.edu

Szilárd Vajda
Department of Computer Science
TU Dortmund, Germany
Email: szilard.vajda@udo.edu

Gernot A. Fink
Department of Computer Science
TU Dortmund, Germany
Email: gernot.fink@udo.edu

***Abstract*—This paper addresses the automatic transcription of handwritten documents with a regular tabular structure. A method for extracting machine printed tables from images is proposed, using very little prior knowledge about the document layout. The detected table serves as query for retrieving and fitting a structural template, which is then used to extract handwritten text fields. A semi-supervised learning approach is applied to this fields, aiming at minimizing the human labeling effort for recognizer training. The effectiveness of the proposed approach is demonstrated experimentally on a set of historical weather reports. Compared to using all labels, competitive recognition performance is achieved by labeling only a small fraction of the data, keeping the required human effort very low.**

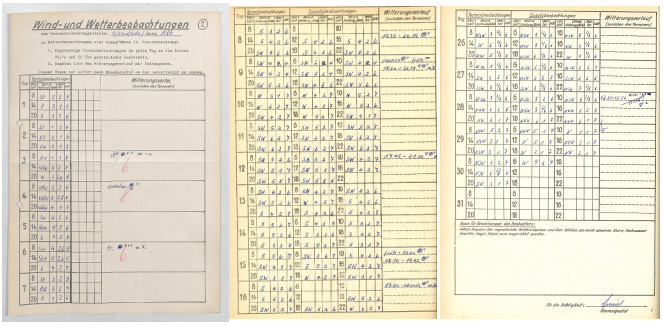

Fig. 1. Examples for the documents considered in this work.

I. Introduction

In archives and museums, huge collections of handwritten documents exist that are of great value for historians and scientists. However, accessing the information contained therein involves browsing through printed catalogues, or searching through piles of documents by hand. In many cases, direct access is not possible at all because the documents may be irreparably damaged in the process. Consequently, there is much effort in digitizing such collections. But scanning documents to images is not sufficient when searching for some specific information, in which case the documents should be indexed or transcribed. This is still mostly done manually by expert annotators, a very time-consuming and tiresome work, and only comparatively small amounts of documents can be recorded this way. Thus, considerable effort has been dedicated to automate the process, which is a challenging problem.

Old document scans often are of bad quality and exhibit artifacts and disturbances that are rarely encountered in modern documents (cf. e.g. [13]). Additionally, opposed to printed characters, handwriting recognition depends heavily on the script style, and thus on the writer. Consequently, recognizers typically have to be trained specifically for a given writer or writing style, and cannot be easily reused or adapted for a different one. This requires frequent re-training and huge amounts of labeling effort. In order to make this process more efficient, the involved manual effort has to be reduced. This is where this work seeks to contribute.

In the following, a specific document collection is considered (cf. Figure 1), namely weather reports recorded between the years 1877 and 1999, provided by the German Weather Service (”Deutscher Wetterdienst”, DWD). They contain daily reports on the weather conditions at a given time and location, written into the fields of a printed table by an observer using a well-defined syntax and vocabulary, and are of great interest for creating statistics and analyzing long-term weather fluctuations.

In order to reduce the labeling effort, firstly a method for automatically extracting tabular structures from document images is developed. The structure may be arbitrary, but it is assumed that all table fields are delineated by graphical lines. Errors are corrected by fitting document templates, using the extracted structure as query. From the retrieved table, text fields are extracted. It is demonstrated how a recognizer can be trained on these by labeling only a small fraction of the data. First experiments on a realistic data set are presented, showing promising results for the proposed approach.

II. Related work

In order to achieve automatic transcription of documents, the contents have to be analyzed in detail. Opposed to retrieval (cf. e.g. [6], [23]) or indexing approaches (cf. e.g. [17]), where only a subset of relevant keywords needs to be recognized, full text recognition is required. Since, ideally, a complete electronic representation is generated, transcribing documents enables, e.g., full text or contextual search, efficient storage and transmission, and automatic content analysis.

Typically, the transcription process consists of several processing steps. After preprocessing, the logical structure of the document has to be analyzed. This includes separating text from figures, identifying titles, sections, etc., and is referred to as layout analysis (cf. e.g. [3]). Afterwards, text areas have to be split into individual lines. For free-form text, this is

typically achieved by explicitly detecting text lines [13]. For official documents and forms that often have a known structure, text extraction can be achieved by searching for the expected structure with some template, and then using the information from the template to extract the text fields (cf. e.g. [14]). However, this generally requires additional effort in creating the templates and fitting them to the documents.

After the extraction of text lines, the text has to be transcribed. Since single characters are hard to segment in handwritten script, recognition is either performed at word level (cf. e.g. [8]) or based on time-series analysis of oversegmented connected components (cf. e.g. [2]). Successful methods that have been used for this purpose are, e.g., Hidden Markov Models [16], [22] or connectionist approaches [11], [20].

The goal of the presented work is similar to [2], in the sense that it also aims at analyzing official documents with a known structure. It is also closely related to recent work on table structure detection and graphic lines extraction (e.g. [14]). For recognition, this paper focuses on isolated digits and characters, leaving recognition of handwritten words for future work. The main goal is to demonstrate that, firstly, complex tabular structures can be extracted with high accuracy, and secondly, high recognition rates can be achieved by manually annotating only a small fraction of the data. For this purpose, methods from the fields of semi-supervised learning [1], [21] and classifier ensembles [10] are adopted. The ultimate goal is the development of an interactive content analysis system, where the automatic recognition modules support a human expert, and incrementally learn to improve their predictions based on the given feedback, similar to [19].

III. Semi-supervised recognition approach

The proposed approach to document analysis consists of three steps: Table extraction, template fitting, and recognition of extracted text fields using semi-supervised learning. In the following, each step will be explained in detail.

A. Table structure detection

The table extraction process starts with a binarization of the image by applying Otsu's method [15]. For the data set considered here, this efficient global method was sufficient.

Next, the Hough parameter space $\mathcal{H}(r, \phi)$ [7] is calculated from the binary document image. In this representation, local maxima correspond to line-like image structures. Since the expected tabular structure can be quite complex (cf. Fig. 1), with considerable variations in line length and spacing, applying a global threshold on the Hough accumulator values will not yield satisfactory results. Therefore, a locally adaptive peak search is applied in a sliding window scheme.

Given a sliding window of size (w, h), strong local maxima are extracted as peak-over-average locations $(\hat{r}_i, \hat{\phi}_i)$: $\mathcal{H}(\hat{r}, \hat{\phi}) > \mu_{w,h} + \gamma\sigma_{w,h}$, where $\hat{r}, \hat{\phi}$ are the coordinates of the window's center bin, and $\mu_{w,h}, \sigma_{w,h}$ are the mean and standard deviation of bin values inside the window, respectively. The parameter γ controls the sensitivity of the peak detection and was set to $\gamma = 2.5$ in all reported experiments. Additionally,

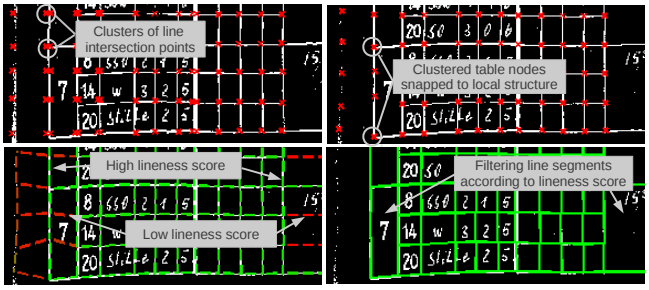

Fig. 2. Example of the grid extraction (best viewed in color). From top left to bottom right: Pair-wise line intersection points. Grid points after clustering and snapping to local image distortions. Lineness scores (green: High lineness, red: Low lineness). Grid structure after thresholding the lineness scores.

non-maximum suppression is applied, keeping only peaks that are the global maximum within their local neighborhood. This results in a list of line hypotheses $y = (\hat{r}_i - x\cos\hat{\phi}_i)/\sin\hat{\phi}_i$.

Adjusting γ such that most table lines are extracted will typically yield some false positives. Most of them can be reliably discarded using a simple criterion: Assuming a rectangular tabular grid, the pairwise inclination angles between valid line hypotheses should be approximately 0 or $\frac{\pi}{2}$. Thus, for each line, a histogram of these angles is calculated. A hypothesis is discarded if its maximum bin does not correspond to the expected values. The histogram bin size specifies the tolerance of the procedure.

One reason for choosing the Hough Transform for table line extraction, as opposed to, e.g., profiles (cf. e.g. [14]), is that this procedure does not require an upright, rotation- or skew-corrected image. In-plane rotations of the document can be corrected at this stage by analyzing the distribution of angle parameters of the extracted lines.

Let α be the in-plane rotation angle of the document. Assuming that most line hypotheses actually correspond to grid lines, the values of $\hat{\phi}$ will cluster around either $-\alpha$, $\frac{\pi}{2} - \alpha$ or $\pi - \alpha$, depending on the direction of the rotation and whether the respective line is horizontal or vertical. By calculating

$$\alpha_i^{min} = min(|\hat{\phi}_i|, |\hat{\phi}_i - \frac{\pi}{2}|, |\hat{\phi}_i - \pi|) \cdot s_i^{min}$$

for each line, where s_i^{min} is the sign of the respective minimum element before taking the absolute value, a histogram in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$ is formed. This yields a prominent peak around the histogram bin corresponding to the true rotation angle. Thus, α is calculated as weighted average over the bin center values that contribute to this peak.

Next, the line segments that correspond to actual lines in the document have to be determined. For this purpose, all pairwise intersection points between the line hypotheses are calculated. These are then merged and arranged in a rectangular axis-parallel grid by applying Mean Shift clustering [4] separately to the x and y coordinates of extracted points. Note that Mean Shift clustering does not require the number of clusters to be known, so no assumption about the document is made here.

The cluster centers are then snapped to the local image structure as follows. A subimage is extracted around each

center's position, the size of which determines the maximum snapping distance. The horizontal and vertical projection profiles of this subimage are then calculated. Elongated foreground structures parallel to the summation direction will generate peaks in the profiles. A Gaussian weighting function $\mathcal{N}(\mu, \sigma)$ is applied, with 4σ corresponding to the size of the profile in the respective direction, penalizing large displacements. The locations of the maximum peaks in the weighted profiles give the snapping position. This procedure thus fits the grid nodes to local distortions of the table structure (Fig. 2 top right).

For each line segment connecting neighboring nodes, a lineness score s_L is then computed as follows: The subwindow defined by the extremal coordinates of adjacent nodes is extracted. A minimum size of the subwindow is enforced, in order to be robust against small localization errors. Then, the horizontal or vertical profile is calculated, depending on whether the grid points are neighbored horizontally or vertically. For an ideal line with no noise and distortions, the profile should be perfectly flat. If the image structure in the subwindow is not a line (e.g. text or broken lines), it will deviate from this ideal shape. Consequently, the normalized profile is treated as a discrete probability distribution $\mathbf{p} = (p_i, i = 1...l)$, and a robust measure for comparing discrete distributions is adopted, the Bhattacharyya distance, originally proposed in [5]:

$$s_L(\mathbf{p}, \mathbf{q}) = 1 - \sqrt{1 - \sum_i \sqrt{p_i \cdot q_i}}. \quad (1)$$

Thus, $s_L(\mathbf{p}, \mathbf{q})$ measures the similarity between $\mathbf{p}$ and a uniform distribution $\mathbf{q}$ (i.e. $q_i = \frac{1}{l}$). Since $s_L(\mathbf{p}, \mathbf{q})$ is normalized to the range $[0, 1]$, it yields a comparable measure for how closely the underlying image structure resembles a line. Using Otsu's method again, an adaptive threshold on the lineness scores is defined to reject false positive line segments. Finally, isolated lines are found and removed, yielding the final table hypothesis (Fig. 2 bottom right).

B. Template retrieval and field extraction

The extracted table structure will typically exhibit errors, such as false positive and missing line segments. Therefore, it is used as a query to find the best match in a template database. The template fitting is based on profiles, similar to [14].

Given an extracted grid and a template, the task is to find the translation τ and scale ρ optimizing their alignment. Let $\mathcal{P}_E$ be a profile (horizontal or vertical) calculated from the extracted grid by summing up distances between adjacent nodes and normalizing with the image size, and $\mathcal{P}_T(\tau, \rho)$ the corresponding transformed profile of the template. The objective function is given by $f(\tau, \rho) = d(\mathcal{P}_E, \mathcal{P}_T(\tau, \rho))$, where $d(...)$ is the Euclidean distance, and is minimized using a Simplex algorithm. Because of the symmetric grid structure, $f(\tau, \rho)$ has many local minima. Consequently, the profiles are smoothed with a Gaussian, yielding a smoother objective function. A good value range for ρ can be determined based on statistics of table cell sizes, and τ is initially selected by aligning dominant profile peaks. This way, the optimization is seeded with a small set of reasonable starting points.

The above procedure is carried out for horizontal and vertical profiles independently, and the best template is selected by minimizing the combined score. After applying the optimal transformation, the template is finally fitted to local image structures using the snapping algorithm described in Section III-A. Text fields can then easily be extracted from the fitted grid. In order to discard enclosing grid lines, the local snapping procedure is applied again to the field's corner nodes, but this time fitting them to the background instead of the foreground, and restricting the snapping direction to the field's interior.

C. Semi-supervised sample labeling

After extracting the text fields, their contents have to be analyzed. The proposed recognition approach keeps the required human effort low while maintaining high recognition rates. Since the manual labeling of training data is a laborious and costly procedure, a labeling method involving minimal human effort was proposed in our previous work [21]. The idea behind the method is simple: Label as few samples as possible and infer the labels for other samples automatically.

The labeling process consists of three major steps. First, the data samples should be represented differently in order to exploit their separability in different feature spaces. Consequently, an ensemble of r feature representations $\mathcal{R}_j, j = 1...r$ is created. Then, each representation is clustered into k_j clusters, where k_j may differ for each representation. Instead of labeling all the samples, only the centroids of the clusters are labeled, and the rest of the samples in the cluster inherit the label. This implies $\sum_j k_j$ manual labeling operations. However, inheriting the labels from the cluster centroids will yield some incorrectly labeled samples, since, generally, the clusters will not be pure.

Thus, the final step of the labeling procedure is to robustly infer reliable labels for the remaining data. For each data point, r labels are assigned based on the clustering. Applying a voting procedure results in an ensemble decision for a specific class label. Opposed to conventional multiview learning, the labels are not assigned blindly, but the ensemble decision is used to select only those samples for subsequent classifier training where the class membership is determined with high confidence. In the unanimity voting scenario [10] used here, samples will only be selected for training if all their labels agree.

IV. Experiments

In the following, experimental results on realistic data are presented. The target document collection contains thousands of pages, but unfortunately, only a limited amount of scanned data was available for the experiments reported here.

A. Data description

The data set consists of 58 scanned pages written by three different writers. They contain wind directions and strengths, written in the fields of a printed table. There are five different types of documents, with different table structures and arrangement. The documents suffer from scanning artifacts and moderate fainting, as well as considerable yellowing.

	Total	Correct	Missing	False Pos.
Nodes	19,805	19,417 (98.0%)	388 (2.0%)	492 (2.5%)
Lines	34,418	33,748 (98.1%)	670 (1.9%)	868 (2.5%)

TABLE I
RESULTS OF THE TABLE EXTRACTION APPROACH.

Ground truth for the text recognition was obtained by manually correcting and labeling connected components. Then, samples were created with the proposed approach by extracting connected components from the text fields. Labels were assigned by finding the best matching annotation in terms of bounding box overlap. In total, the data consists of $7,860$ samples in 17 classes (digits 0–9, characters N, S, W, O, E, T, I, L). Note that '0' and 'O' are treated as one class, since they are indistinguishable even for a human. The data set is unbalanced, since the classes 9, E, T, I, and L occur very rarely.

B. Features

For the semi-supervised labeling, $r = 4$ different data representations were considered: The raw image, normalized and centered similarly to the MNIST handwritten digit dataset [11], a Principal Component Analysis (PCA) representation, dimensionality reduction based on an autoencoder (AE) network [9], and a representation obtained by non-negative matrix factorization (NNMF, [12]). The raw image was selected for its high capabilities of separation [11], while the PCA discards low variance components and decorrelates the feature values. NNMF is reported to decompose objects into meaningful parts. Finally, the AE is a feature learning method based on neural network training. For the PCA, the first 80 principal components were used. Similarly, the AE bottleneck size and NNMF data dimensionality were also set to 80. This is a heuristic choice, motivated by the approaches in [11], [20].

C. Automatic table structure extraction

In order to evaluate the table detection, the extraction algorithm was applied to all documents using identical parameters. The extracted grid structures were then visualized on top of the respective images, and errors were counted manually. There are four types of errors: Missing grid nodes, missing line segments, false positive nodes and false positive segments. A fifth type would refer to correct nodes not located reasonably on the grid. However, this error never occurred.

The evaluation results are given in Table I. For nodes and lines, a correct detection rate of around 98% is achieved, with a moderate false positive rate of 2.5%. Most of the latter come from five documents where an enclosing rectangle was detected around the circumference of the image due to artifacts from scanning and rotation correction. Overall, only 13 grid lines were missed completely. These results clearly show the effectiveness of the table extraction approach. In the subsequent template retrieval, the correct template was selected in all cases. In 4 out of 58 cases, the localization had to be corrected manually, mainly due to missing lines on the table boundary.

Class.	# Labels	Raw	PCA	AE	NNMF
3-NN	6,550	84.1±0.8	86.9±0.8	81.0±0.9	82.0±0.9
3-NN	200	85.6±0.8	86.9±0.8	81.9±0.9	83.2±0.8
RVM	200	n/a	82.7±0.9	74.4±0.9	80.7±0.9
MLP	200	n/a	82.2±0.9	78.4±0.9	82.4±0.9
3-NN	6,550	87.1±0.8	88.9±0.7	84.0±0.8	85.4±0.8
3-NN	200	87.9±0.7	89.0±0.7	83.7±0.8	86.9±0.8
RVM	200	n/a	85.3±0.8	80.0±0.9	84.9±0.8
MLP	200	n/a	85.9±0.8	83.0±0.9	86.3±0.8

TABLE II
OVERVIEW OF HANDWRITING RECOGNITION RESULTS. TOP HALF: RESULTS WITHOUT FIELD TYPE CONTEXT. BOTTOM HALF: RESULTS UTILIZING FIELD TYPE KNOWLEDGE FROM THE TEMPLATE. ALL VALUES ARE IN %. CONFIDENCE INTERVALS ARE GIVEN FOR A CONFIDENCE LEVEL OF 0.95.

D. Semi-automatic labeling and recognition

In the following, the different data representations were clustered to $k_j = 50$ clusters, respectively, using the k-means algorithm. All experiments were performed in a 6-fold cross validation scheme using ten (eight for set six) documents for testing and the rest for training. Thus, all documents were considered once in the overall test set. A K nearest neighbors (K-NN, $K = 3$) classifier, a Relevance Vector Machine (RVM, [18]) with a Gaussian kernel and a multi-layer perceptron (MLP, cf. e.g. [20]) were investigated as classifiers. For multi-class classification with the RVM, a 1-vs-1 majority voting setup was used. The concensus sample set and labels obtained by unanimity voting were used for training.

Table II gives an overview of the results. For comparison, the scores obtained by using all ground truth labels for training are also presented. All four feature representations were considered for classifier training. The best overall result was obtained with the PCA data, yielding 86.9% correct classifications. With the proposed semi-supervised labeling approach, also 86.9% were achieved. It should be noted that, due to the cluster-based labeling, the effective number of classes tends to decrease. Samples belonging to very rare classes do not form individual clusters, but are assigned randomly and eliminated during the voting process. This occurs more often the smaller the number of clusters gets. The effective number of classes, averaged over cross validation sets, was 12.17 for the semi-supervised experiment, and 16.83 when using all ground truth labels. Hence, the scores are not directly comparable.

Nevertheless, the experiment shows that competitive performance can be achieved by labeling only a small fraction of the data using the proposed approach. Instead of labeling $6,550$ samples on average per cross validation set, only 200 manual labeling operations (3.1%) were necessary. The best results were consistently achieved using the 3-NN classifier with the PCA data, while results for the RVM classifier were always significantly worse. Note that only a very rough parameter optimization was carried out for the RVM.

In a second experiment, it was assumed that context information about the field type (numeric or character) was available (e.g. from the document template). Two separate sets of classifiers were trained using only samples of the corresponding type, and test samples were assigned based on their context.

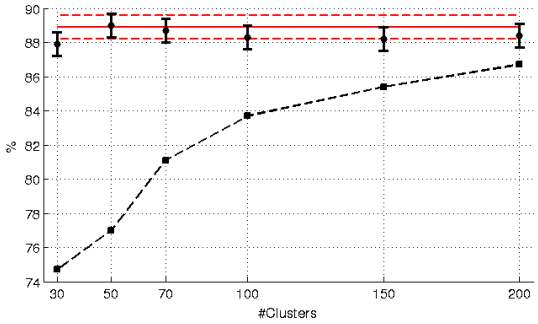

Fig. 3. Recognition rates (PCA, 3-NN) when varying the number of clusters. Red horizontal lines: Confidence interval using all ground truth labels ($88.9 \pm 0.7\%$). Confidence intervals for the measurements are indicated by vertical bars. Dashed curve: Amount of samples selected by the unanimity vote.

Not surprisingly, this led to a significant improvement in performance (Table II, lower half). The best result is again obtained with PCA data and the 3-NN classifier, yielding 89.0% correct classifications. For comparison, a different clustering approach using a 2D self-organizing map (SOM) with 7×7 nodes was investigated. The results were generally significantly worse than those presented above, and therefore are not shown in detail. Here, the best performance was achieved with PCA data and the 3-NN classifier at $84.1 \pm 0.8\%$ without and $86.4 \pm 0.8\%$ with context, respectively.

Finally, Figure 3 shows the recognition rates when varying the number of clusters from 30 to 200 (1.8% to 12.2% of manual labels, respectively) using PCA data, 3-NN and field type context. The variations are insignificant when the number of clusters is increased beyond 50. Most results are also within the confidence interval of the optimal result obtained by using all ground truth labels. This clearly shows the effectiveness of the proposed semi-supervised sample selection procedure. The relative amount of training samples selected by the unanimity vote is given by the dashed curve. As expected, it increases with the number of clusters, since the granularity gets finer. Consequently, the average effective number of classes increased from 11.67 ($k_j = 30$) to 13.33 ($k_j = 200$). However, this does not have a positive effect on the performance. Apparently, a sufficient subset of "good" training samples can already be obtained with relatively few clusters.

V. Conclusion

This paper considered the problem of recognizing handwritten fields in tabular historical weather reports with minimum human effort. To this purpose, an approach for automatically detecting delineated tabular structures was presented. These were then used to retrieve matching document templates from a database. Relevant fields were extracted automatically from the fitted tables. Furthermore, a semi-supervised labeling approach based on an ensemble decision was presented that requires only very few manual labeling operations. The proposed approach was evaluated on a set of real documents, and very promising results were achieved for both the table extraction and semi-supervised recognition. Specifically, it was shown that competitive recognition rates can be achieved by manually labeling only small fractions of the available data.

VI. Acknowledgments

This work is supported by the German Federal Ministry of Economics and Technology on a basis of a decision by the German Bundestag within project **KF2442004LF0**.

References

[1] G. R. Ball and S. N. Srihari. Semi-supervised learning for handwriting recognition. In *Proc. Int. Conf. on Document Analysis and Recognition*, pages 26–30, 2009.

[2] M. Bulacu, A. Brink, T. van der Zant, and L. Schomaker. Recognition of handwritten numerical fields in a large single-writer historical collection. In *Proc. Int. Conf. on Document Analysis and Recognition*, 2009.

[3] R. Cattoni, T. Coianiz, S. Messelodi, and C. M. Modena. Geometric layout analysis techniques for document image understanding: a review. Technical report, 1998.

[4] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 24(5):603–619, 2002.

[5] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 25(2):564–575, 2003.

[6] D. Doermann. The indexing and retrieval of document images: A survey. *Computer Vision and Image Understanding*, 70(3):287–298, 1998.

[7] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Comm. of the ACM*, 15:11–15, 1972.

[8] S. L. Feng and R. Manmatha. Classification models for historical manuscript recognition. In *International Conference on Document Analysis and Recognition*, pages 528–532, 2005.

[9] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, July 2006.

[10] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Intelligent Signal Processing*, pages 306–351. IEEE Press, 2001.

[12] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[13] 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.

[14] H. Nielson and W. Barrett. Consensus-based table form recognition of low-quality historical documents. *Int. Journal on Document Analysis and Recognition*, 8(2):183–200, 2006.

[15] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. on Systems, Man and Cybernetics*, 9(1):62–66, January 1979.

[16] T. Plötz and G. A. Fink. Markov Models for Offline Handwriting Recognition: A Survey. *Int. Journal on Document Analysis and Recog.*, 12(4):269–298, 2009.

[17] T. M. Rath and R. Manmatha. Word spotting for historical documents. *Int. Journal on Document Analysis and Recognition*, 9(2):139–152, 2007.

[18] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[19] A. H. Toselli, E. Vidal, and F. Casacuberta. *Multimodal Interactive Pattern Recognition and Applications*, chapter Active interaction and learning in handwritten text transcription, pages 119–133. Springer, 2011.

[20] S. Vajda and G. A. Fink. Strategies for training robust neural network based digit recognizers on unbalanced data sets. In *Proc. Int. Conf. on Frontiers in Handwriting Recognition*, pages 148–153, 2010.

[21] S. Vajda, A. Junaidi, and G. A. Fink. A semi-supervised ensemble learning approach for character labeling with minimal human effort. In *Proc. Int. Conf. on Document Analysis and Recognition*, pages 259–263, 2011.

[22] M. Wüthrich, M. Liwicki, A. Fischer, E. Indermühle, H. Bunke, G. Viehhauser, and M. Stolz. Language model integration for the recognition of handwritten medieval documents. In *Proc. Int. Conf. on Document Analysis and Recognition*, 2009.

[23] G. Zhu and D. Doermann. Logo matching for document image retrieval. In *Proc. Int. Conf. on Document Analysis and Recognition*, 2009.