Bag-of-Features Representations for Offline Handwriting Recognition Applied to Arabic Script

Leonard Rothacker, Szilárd Vajda, Gernot A. Fink
Faculty of Computer Science, TU Dortmund, Germany
{*leonard.rothacker, szilard.vajda, gernot.fink*}*@tu-dortmund.de*

***Abstract*—Due to the great variabilities in human writing, unconstrained handwriting recognition is still considered an open research topic. Recent trends in computer vision, however, suggest that there is still potential for better recognition by improving feature representations. In this paper we focus on feature learning by estimating and applying a statistical bag-of-features model. These models are successfully used in image categorization and retrieval. The novelty here is the integration with a Hidden Markov Model (HMM) that we use for recognition. Our method is evaluated on the IFN/ENIT database consisting of images of handwritten Arabic town and village names.**

***Keywords*-Arabic handwriting recognition; Hidden Markov Models; feature learning; Bag-of-Features; local image features;**

I. Introduction

Handwriting recognition is successfully applied when the recognition domain can be constrained (cf. [1]). When, for example, recognizing postal addresses the postcode can be used for restricting possible street names. In unconstrained handwriting recognition no such assumptions are possible. Handwritten text can appear in any context making the use of a dictionary inefficient. Also no prior information about the visual appearance of text in a document is given, thus making it necessary to handle great inter and intra writer variabilities. In order to approach these challenges, the method proposed in this paper uses learned feature representations with an HMM. HMMs model the generation of observation sequences stochastically and are, therefore, well suited for handwriting recognition (cf. [2], [3], [1]). Features that are commonly used with HMMs can be classified in two categories namely heuristic and analytic [1]. Heuristic features often describe geometric properties of the pen stroke (cf. e.g. [2]) where analytic features, in contrast, often incorporate a statistical model of the pen stroke's pixel intensity distribution (cf. e.g. [3]). A principle advantage of these analytic features is that they are designed on a structural level and their explicit parameters can be estimated from training data instead of fine-tuning them manually. Another important aspect is that a problem specific adaptation is possible. The method is not entirely fixed by the decisions of a human expert. Although these decisions have proven to work well in general, it is not apparent that no better solution exists.

The learned feature representations, considered here, are called *bag-of-features* and are a standard approach in image categorization and retrieval (cf. [4]). However, their application in document analysis and recognition is rather new. To our best knowledge only two publications exist where a bag-of-features model is applied to word spotting in historic handwritten documents [5] and text detection / character recognition in natural scene images [6] (cf. Section II). The conceptual idea is to base the representation upon approximations of typical image patches. These are in no spatial relation thus making the histogram of their occurrence an unordered *bag*-of-features. Finding these approximations is considered as learning because typical representatives for the data in the recognition domain are obtained. This is accomplished by clustering image patches from a training dataset in an unsupervised manner. Due to the application to images, the set of representatives is often referred to as visual vocabulary and its elements as visual words[1]. In order to represent an image according to this previously estimated vocabulary, each image patch is associated with its most similar visual word. The number of occurrences of each visual word in the entire image serves as bag-of-features statistic. In practice, the locations of image patches and their discriminative representations are often determined by interest point detectors and descriptors (cf. [4]).

The main contribution of the method presented in this paper is the integration of a learned statistical bag-of-features model with an HMM for unconstrained offline handwriting recognition. For this purpose two important aspects have to be taken into account:

1) For using an HMM, a *sequence* of bag-of-features representations must be created from a given text line.
2) The *generation* of bag-of-features representations must be modeled within the HMM.

In order to show that features for Arabic script can be learned with the bag-of-features model, we evaluate our method on the IFN/ENIT dataset [7]. We use a semi-continuous HMM with geometrical features as baseline system [8] (features originate from [2]). When using this system with the bag-of-features representations, a direct

[1]The terms go back to the bag-of-words principle where texts are represented by the number of word stem occurrences.

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.185

feature comparison becomes possible.

The remainder of this paper is organized as follows: Section II gives an overview of previous publications covering part-based feature representations for handwritten text or character recognition. Afterwards, our method is discussed in Section III and evaluated in Section IV. A final conclusion is given in Section V.

II. Related work

The application of local image features is very popular in various computer vision tasks, like image retrieval and categorization or object recognition. Because the features describe the local neighborhood of a particular interest point, also bag-of-features representations are based upon them [4]. Inspired by their success, local image features as well as bag-of-features have been applied in character recognition and word spotting, before.

In [9] isolated handwritten digits are recognized by a nearest-neighbor classifier that is based on local gradient-based image features. A sufficient number of features is extracted from each annotated training image and labeled accordingly. In a formerly unknown image the digit recognition then consists of two steps. First, its local image features are classified according to the 1-nearest-neighbor rule with respect to the reference features. Then, a digit category can be assigned by applying a majority voting scheme. Although the method does not incorporate the bag-of-features concept it is still relevant. It shows that local gradient-based features have sufficient descriptive capabilities for handwritten digit classification even if no spatial feature information is included.

That a bag-of-features approach can be used for text detection and character recognition in complex scenes is shown in [6]. Local features are obtained from small image patches that are statistically preprocessed by whitening. A visual vocabulary is created from training data using an unsupervised learning approach. It is important to note that the visual vocabulary is given as a set of basis vectors. In order to associate an image patch with the items from the vocabulary, a basis transformation is performed. Bag-of-features are computed by averaging the representations obtained within the cells of a sparse grid over the input image. The method shows how the feature representation is adapted to the problem domain. The visual words or basis vectors can be interpreted as images. The authors illustrate that by estimating a visual vocabulary from character images the vocabulary consists of different character parts, like differently curved strokes. Depending on how their SVM-based classifier is trained, the method can be used in order to detect text or recognize characters.

In [5] a bag-of-features word spotting method is presented. It is based on gradient-based SIFT (Scale Invariant Feature Transform) descriptors [10] and can be divided into several steps. As usual, a visual vocabulary must be created. For this purpose descriptors are calculated on a dense grid over the training corpus and clustered with the k-means algorithm. Next, the document collection must be prepared for retrieval. Each document is divided into overlapping patches and for each patch a bag-of-features statistic is created. Descriptors are quantized with respect to the visual vocabulary. In order to include some spatial information, bag-of-features representations are computed for each cell in a spatial pyramid (cf. [5]). The feature vector for the complete patch is then obtained by concatenating all localized bag-of-features statistics. The high dimension of the resulting vector is reduced by latent semantic indexing (cf. [5]). Patches similar to a query image can now be retrieved by evaluating a distance measure between the feature representations. The method is tested on a historic handwritten and two typewritten document collections.

The methods presented in this section are applied to character recognition and word spotting. They served to illustrate the descriptive power of gradient-based local image features and the bag-of-features model. We, in contrast, apply our method to unconstrained Arabic handwriting recognition. This is more challenging because continuously written script has to be transcribed. The bag-of-features integration with an HMM has, to our best knowledge, not been published before.

III. Bag-of-Features for Arabic handwriting recognition

The method proposed in this section uses learned bag-of-features representations within an HMM for Arabic handwriting recognition. The process starts with preprocessing the word images. In our experiments we use skew and slant normalizations. The skew normalization is based on baseline estimations and the slant normalization is based on the mean gradient in a word segment. For further details refer to [8]. After normalization, local image features can be extracted (Section III-A). In the learning phase features from the training dataset are used for creating the visual vocabulary. With a given vocabulary, bag-of-features representations can be obtained for each word image (Section III-B). Because HMMs process feature vector sequences, a sequence of bag-of-features representations needs to be generated accordingly. Finally, two methods for modeling these bag-of-features sequences as output of the HMM will be presented (Section III-C).

A. Feature extraction

In order to compute local image features, interest points must be detected. A feature representation describing their local neighborhoods is provided by an interest point descriptor. For a bag-of-features model it is very important to compute a sufficient number of interest points as otherwise no meaningful statistic can be obtained (cf. [5]). Here we accomplish this by applying the Harris Corners [11] detector

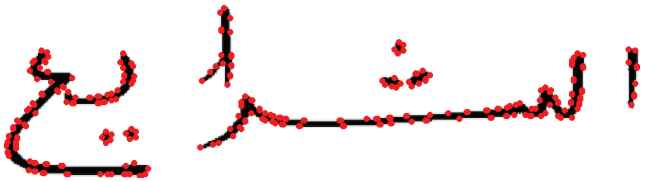

Figure 1. Harris Corners points[2]

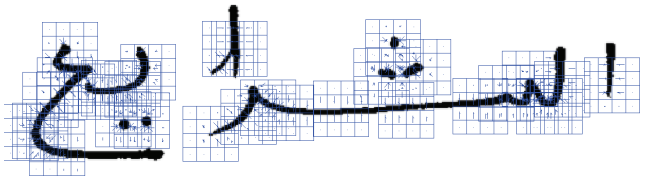

Figure 2. A few SIFT descriptors[2]

to the binarized word image. A typical result is shown in Figure 1 (best viewed in color). Because the Harris Corners detector responds rather to corners than edges, especially curved parts of the pen-stroke are densely covered with interest points. Prior analysis of the Arabic word images led to the suggestion that these parts are very discriminative for the different characters. Elongated horizontal structures, in contrast, are covered by less interest points.

For calculating a discriminative representation of the interest point neighborhood, we use the 128-dimensional SIFT descriptor [10] consisting of four eight-bin sub-histograms of oriented gradients. This is a very popular choice for bag-of-features applications (cf. e.g. [4], [5], [12]). Figure 2 is illustrating an example for SIFT descriptors at a few Harris Corners points. The SIFT descriptor is computed relative to the interest point's scale and orientation. This makes it invariant to these transformations. In our case only the Harris Corners points are given, thus leaving the other parameters to be determined for all descriptors in an image. With respect to the orientation, prior experiments suggested that orientation invariance is not helpful for discriminative representations of the pen-stroke (cf. [9], [5]). This is intuitive, because horizontal and vertical structures would be represented similarly in descriptor space. For that reason the interest point orientations are always set to zero. Regarding the scale, please note that we do not compute the descriptor with respect to a scale space representation (cf. [10]). The scale only specifies the spatial descriptor size in the word image (cf. Figure 2). The choice of a scale parameter for all descriptors in an image is very important. Too small descriptors are not discriminative enough as they cover only a small part of the pen-stroke. Too big descriptors cover several characters and, therefore, include character contexts within a word. The optimal descriptor size for the IFN/ENIT database has been

[2]Based on image from IFN/ENIT [7].

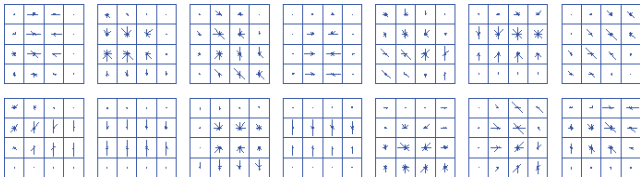

Figure 3. Visual vocabulary

determined experimentally.

Finally, we de-correlate the SIFT descriptors with Principle Component Analysis (PCA). In [13] it was shown that image retrieval can be improved this way. This aspect will also be of interest when quantizing the de-correlated descriptors with respect to the visual vocabulary (Section III-B). Note that descriptors visualized in Figures 2 and 3 have not been de-correlated.

B. Clustering and quantization

The purpose of clustering descriptors from the training dataset is to estimate a visual vocabulary. Afterwards, the bag-of-features statistic can be created by quantizing descriptors in a given image. As Figure 3 shows exemplarily, the visual words correspond to parts of the pen-stroke. Horizontal, vertical as well as curved structures can be observed in the depicted visual word SIFT descriptors[3]. Due to the huge number of descriptors extracted from training data, MacQueen's k-means algorithm [14] is used for clustering. The size of the codebook is determined experimentally (Section IV). Based on the clustering result, a Gaussian mixture model (GMM) is estimated. Because the descriptors have been de-correlated, it is sufficient to use diagonal covariances. Afterwards, the GMM is used in order to quantize descriptors in a hard or soft manner. For hard quantization the maximum a-posteriori probability is considered. For soft quantization each descriptor can be associated with multiple visual words. When creating the bag-of-features statistic, their a-posteriori probabilities are accumulated in the histogram. The positive effect of soft over hard quantization for bag-of-features image categorization has been investigated in [12]. At the top of Figure 4 the result from hard quantization with respect to the vocabulary in Figure 3 is shown. The colors indicate visual word affiliations. Note that similar structures in the word have similar color patterns.

C. Bag-of-Features HMM integration

In order to use a Hidden Markov Model for recognition, a sequence of feature vectors must be created. Usually a sliding window approach is used for that purpose (cf. [2], [1]). Therefore, the window is moved over the image in writing direction and at each window position a feature representation is extracted based on the current window content. Inspired

[3]In the 4x4 cells the eight-bin orientation histograms are outlined.

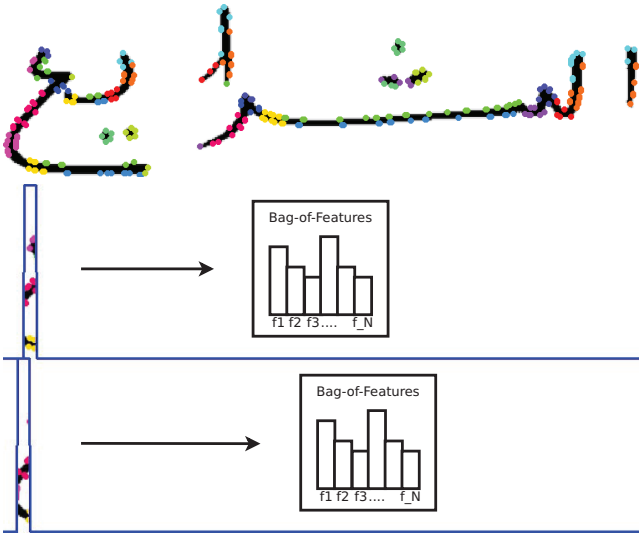

Figure 4. Sliding Bag-of-Features[2]

by this concept we create a bag-of-features statistic at each window position by only considering the interest points within the window. The bag-of-features statistics are spatially localized within the image this way. Figure 4 visualizes the method (best viewed in color).

HMMs model the generation of observation sequences. The traditional way of output modeling is to use a GMM. For robust estimations, however, the dimension of feature representations is limited. In our method these observations correspond to bag-of-features statistics. Because these are high dimensional in comparison to geometric features, they cannot be used directly. Geometric features presented in [2] are 9-dimensional where bag-of-features representations have easily several hundreds to thousands of dimensions (cf. e.g. [5], [12]). For that reason, we reduce the bag-of-features dimensionality to the same order of magnitude with PCA. As in the baseline system we use a semi-continuous HMM (SC-BoF-HMM).

A second approach for output modeling is to directly estimate the visual word probabilities within each HMM state. In order to obtain visual word probabilities from a bag-of-features statistic, the frequencies are normalized. A special case where no interest points can be observed in the sliding window is modeled by an additional "pseudo" visual word. Otherwise the normalization would fail. Let f_k be the probability for visual word $k \in \{1 \dots \mathcal{V}\}$. The overall probability of observing an entire bag-of-features representation in a specific state j is then given by:

$$b_j(\boldsymbol{f}) = \sum_{k=1}^{\mathcal{V}} c_{jk} f_k. \quad (1)$$

During HMM training only the coefficients c_{jk} have to be estimated for each state and visual word. These represent the learned visual word probabilities. The approach is related to a discrete HMM: In our method not only one symbol is observable at a point in time but $\mathcal{V}$ symbols with a certain probability. We refer to this concept as *Bag-of-Features HMM* (BoF-HMM).

Table I
SC-BoF-HMM: FEATURE DIMENSION

Feature dimension	WER (*abc – d*)
20	5.0 %
30	**4.7 %**
40	5.0 %

IV. EVALUATION

We evaluate our method on the writer-independent Arabic handwriting recognition tasks defined on the IFN/ENIT dataset [7]. In the current version it consists of 32,492 word images of Tunisian town and village names that are divided in the subsets *a – e*. Additionally, the unpublished subsets *f* and *s* exist [15]. In our experiments we consider the following *training – test* configurations: *abc – d* for validation and *abcd – e*, *abcde – f*, *abcde – s* for testing. In order to directly measure the benefit from incorporating learned feature representations, experiments are performed with a baseline system [16] as well. It uses 18-dimensional features (9 geometrical and their derivatives) in a semi-continuous HMM consisting of Arabic character models with a linear or Bakis topology. Its codebook contains 2,000 densities. The parameter configuration was chosen according to the best performance in its validation. Despite its simplistic model, the results are comparable to the current state-of-the-art (cf. Table V, [15]).

The recognition performance is always measured by the relative number of false word classifications, i.e. word error rate (WER). Results in the validation (subset *d*) that are differing more than $\pm 0.5\%$ differ significantly at a significance level of 95%. The confidence intervals for the subsets *e*, *f* and *s* are given in Table V.

Bag-of-features parameters are optimized within validation experiments (Tables I, II, III, IV). These are all oriented towards the best configuration observed (highlighted in bold). In order to simplify the evaluation, HMMs consist of uniformly initialized, linear character models. We start with the SC-BoF-HMMs: Their codebook contains 2,000 densities modeling bag-of-features representations that are reduced in their dimensionality by PCA. The codebook size is chosen according to prior experiments. Because the results are generally worse than those of the BoF-HMM, word error rates are only reported for different feature dimensions. Table I shows that no significant change can be observed in the feature dimension range examined. The following results refer to the BoF-HMM. Table II shows the effect of the SIFT descriptor parameters. The descriptor size is

Table II
BoF-HMM: DESCRIPTOR PARAMETERS

Size	Orientation	Dimension	WER (*abc* – *d*)
40	fixed	128_{PCA}	4.0 %
52	**fixed**	**128_{PCA}**	**3.8 %**
64	fixed	128_{PCA}	5.1 %
52	fixed	128	4.2 %
52	invariant	128_{PCA}	5.4 %

Table III
BoF-HMM: BoF PARAMETERS

Visual words	Quantization	WER (*abc* – *d*)
1,000	soft	4.6 %
2,000	**soft**	**3.8 %**
3,000	soft	3.7 %
2,000	hard	4.3 %

Table IV
BoF-HMM: SLIDING WINDOW SIZE

Window size	Window shift	WER (*abc* – *d*)
1	1	6.2 %
2	**2**	**3.8 %**
3	2	4.5 %
5	2	5.6 %

Table V
RESULTS (WER) ON TEST SETS *e*, *f*, *s*

System	*abcd* – *e* ±0.6%	*abcde* – *f* ±0.6%	*abcde*– *s* ±2.0%
Baseline	8.1 %	14.4 %	19.6 %
SC-BoF-HMM	8.0 %	10.3 %	21.0 %
BoF-HMM	7.1 %	**9.8 %**	19.3 %

most important. It is specified as the squared area in the image (in pixels). The sizes mainly depend on the image resolution in the IFN/ENIT dataset. Experiments also indicate that descriptor de-correlation does not improve the word error rate significantly. As assumed initially (cf. Section III-A), results are clearly worse when using rotation invariant descriptors. Table III presents results regarding the visual vocabulary creation and quantization. A vocabulary consisting of 1,000 visual words is too small. With 2,000 visual words the WER is significantly better. Beyond that no relevant improvement is possible. Furthermore, the soft quantization shows significantly better results in the experiments. The effect of different sliding window sizes is presented in Table IV. A window size of 2 pixels is significantly better than the smaller and larger window sizes of 1, 3 and 5 pixels.

Finally, we report our results on the test datasets *e, f* and *s*. For these experiments we use Bakis models that are initialized based on the results obtained with linear models. With the parameter configuration producing best results in the validation, the most impressive word error rate could be achieved on set *f*: The BoF-HMM reached 9.8%. The baseline system, in contrast, achieved only 14.4%. The relative improvement of 31.9% demonstrates clearly that feature representations can successfully be learned for Arabic script. On set *e* a relevant relative improvement of 12.3% is possible. Only the differences on set *s* are not significant. Results of the SC-BoF-HMM are relatively similar to the BoF-HMM. The differences are only significant for set *e*. However, given the overall performance it can still be considered worse. The test results for our three competing systems are summarized in Table V. In comparison to the word error rates accomplished by the participants of the Arabic Handwriting Competitions (2007 – 2011), the BoF-HMM is among the top five results [15].

V. CONCLUSION

In this work we presented the integration of a feature learning method with a Hidden Markov Model for Arabic handwriting recognition. We demonstrated that learned feature representations can outperform well-established heuristically designed features. In our experiments only three parameters were of major importance. Consequently, the representation is mostly estimated automatically without extensive parameter optimization. Besides using bag-of-features representations for estimating a semi-continuous HMM we were furthermore able to find a much more direct interpretation for visual words with respect to the HMM output modeling. This approach is easier, more effective, and also more efficient, as the estimation of the Gaussian mixture model can be omitted.

In future research especially other datasets will be investigated. The feature representations should be adaptable to different problem domains as well.

ACKNOWLEDGMENTS

The authors would like to express their sincere thanks to Haikal El Abed for providing the IFN/ENIT datasets *f* and *s*.

This work is partially 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] T. Plötz and G. A. Fink, "Markov Models for Offline Handwriting Recognition: A Survey," *Int. Journal on Document Analysis and Recognition*, vol. 12, no. 4, pp. 269–298, 2009.

[2] U.-V. Marti and H. Bunke, "Handwritten sentence recognition," in *Proc. of the Int. Conf. on Pattern Recognition*, vol. 3, Barcelona, Spain, 2000, pp. 467–470.

[3] M. Pechwitz and V. Märgner, "HMM based approach for handwritten Arabic word recognition using the IFN/ENIT-database," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, vol. 2, Edinburgh, Scotland, 2003, pp. 890–894.

[4] S. O'Hara and B. A. Draper, "Introduction to the bag of features paradigm for image classification and retrieval," *Computing Research Repository*, vol. arXiv:1101.3354v1, 2011.

[5] M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós, "Browsing heterogeneous document collections by a segmentation-free word spotting method," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, Sep. 2011, pp. 63–67.

[6] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. Wu, and A. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, Sep. 2011, pp. 440–445.

[7] M. Pechwitz, S. S. Maddouri, V. Märgner, N. Ellouze, and H. Amiri, "Ifn/enit - database of handwritten arabic words," in *Proc. of Colloque Int. Francophone sur l'Écrit et le Document*, 2002, pp. 129–136.

[8] M. Wienecke, G. A. Fink, and G. Sagerer, "Toward automatic video-based whiteboard reading," *Int. Journal on Document Analysis and Recognition*, vol. 7, no. 2–3, pp. 188–200, 2005.

[9] S. Uchida and M. Liwicki, "Part-based recognition of handwritten characters," in *Proc. of the Int. Conf. on Frontiers in Handwriting Recognition*, Nov. 2010, pp. 545–550.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[11] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conf.*, 1988, pp. 147–151.

[12] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *European Conf. on Computer Vision*, vol. 3, 2008, pp. 696–709.

[13] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 506–513, 2004.

[14] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1, 1967, pp. 281–296.

[15] V. Märgner and H. Abed, "ICDAR 2011 - Arabic handwriting recognition competition," in *Proc. of the Int. Conf. on Document Analysis and Recognition*, Sep. 2011, pp. 1444 –1448.

[16] G. A. Fink and T. Plötz, "Developing pattern recognition systems based on Markov models: The ESMERALDA framework," *Pattern Recognition and Image Analysis*, vol. 18, no. 2, pp. 207–215, 2008.