Deep Feature Embedding for Accurate Recognition and Retrieval of Handwritten Text

Praveen Krishnan*, Kartik Dutta* and C.V. Jawahar
CVIT, IIIT Hyderabad, India
{praveen.krishnan, kartik.dutta}@research.iiit.ac.in and jawahar@iiit.ac.in

***Abstract*—We propose a deep convolutional feature representation that achieves superior performance for word spotting and recognition for handwritten images. We focus on :- (i) enhancing the discriminative ability of the convolutional features using a reduced feature representation that can scale to large datasets, and (ii) enabling query-by-string by learning a common subspace for image and text using the embedded attribute framework. We present our results on popular datasets such as the IAM corpus and historical document collections from the Bentham and George Washington pages. On the challenging IAM dataset, we achieve a state of the art mAP of 91.58% on word spotting using textual queries and a mean word error rate of 6.69% for the word recognition task.**

***Keywords*-Word spotting, word recognition, embedded attributes.**

I. Introduction

Accurate recognition of handwritten text has remained as a prime problem of interest for many decades. When the domain (or lexicon) is limited or when the text is written by a limited number of individuals, there have been many successful solutions such as [1], [2]. However, performance of unconstrained handwritten word recognition and retrieval has been far from satisfactory. We attempt to bridge this gap with deep learned features. This paper reports a deep feature representation that results in superior recognition and retrieval performance on the popular datasets. In addition to advancing the state of the art, we also argue that a representation that can be learnt from a dataset is the key direction for furthering handwritten text recognition. Our method uses limited amount of manually annotated data and leverage extensively by using synthetically generated handwritten data [3] and enable large scale learning in handwritten documents.

Utility of deep learned features for a wide range of recognition tasks in computer vision is now widely appreciated [4], [5]. Networks trained on large data sets also learn generic feature representation that can be used for related tasks [6], and in many cases, have reported state of the art results as compared to the handcrafted features. When the data is limited, fine tuning a pre-trained network has also been demonstrated to be very effective [7]. Motivated by these, we learn a feature representation that suits handwritten images. Considering the relatively small quantity of annotated data publicly available, we pre-train the network on a synthetic dataset and fine tune it on a real world corpus. We further improve the representation by embedding both the CNN representation and text labels into a common subspace where both the images and their corresponding text representation lie close to each other.

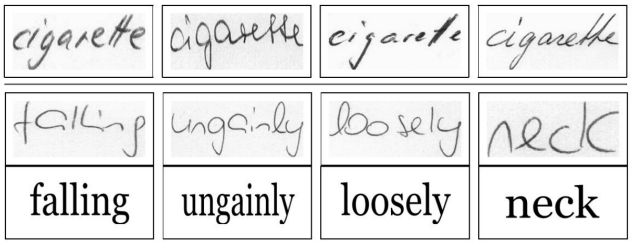

Figure 1. (a) Top row shows the top retrieved word images for the textual query "cigarette" from the IAM dataset. (b) Recognition results for a few word images from the IAM dataset are shown in bottom two rows.

Given the inherent challenges in handwritten documents, the problem of text retrieval is typically posed either as word spotting [2], [8] from a given candidate corpus or word recognition [8], [9] constrained on a given lexicon. Under both scenarios it is assumed that the word images are segmented and available at hand. In word spotting, one is interested in learning holistic word image features which is useful in predicting similarity between a query and candidate image without recognition. Here the query can be text or an exemplar image. Initial works such as [2], [10] used variable length feature representation which use dynamic time warping, or HMM for computing the distance. Later bag of visual words based method using local features such as SIFT, HOG and better encoding schemes such as Fisher vectors [11], [12], gave promising results both in terms of scalability and performance as compared to variable length representation. In [8], [13], a new representation based on character level attributes referred to as pyramidal histogram of characters (PHOC) was proposed which encodes the spatial and lexical properties of a word image and its label seamlessly. In [8], Almazan et.al. used Fisher based word image representation for attribute embedding which

*Equal Contribution

gave state of the art results until recently before the success of deep CNN features [14], [15]. In the recognition domain, most of the popular methods uses HMM [16] or recurrent neural networks such as LSTM [17]–[19]. In [9], Poznanski et. al. uses a very deep CNN architecture for recognition of PHOC attributes using multiple parallel fully connected layers, thereby resulting in a high dimensional representation. The learned representation is further embedded into a common subspace of text and images using canonical correlation analysis [20] and they report the current state of the art results for word recognition.

In this work we improve the discriminative ability of deep CNN features using HWNet [14] by learning the PHOC based attribute representation and embed both the text and image representation into a common subspace. The proposed representation gives state of the art results in word spotting and comparable results with [9] in recognition while having a shallow network and a compact representation which is preferable for large scale datasets. Figure 1 shows sample results from the proposed framework. The top row shows retrieved results for the query "cigarette" and the bottom two rows show the recognition results for a few challenging word images along with their recognized output. In Section II, we present our deep CNN architecture, its learning scheme and introduce the framework for embedding the image and text labels into a common subspace. In Section III, we present results on standard datasets and compare the results on word spotting and recognition with the state of the art methods. Finally we conclude in Section IV along with future works.

II. Deep Features and Embedding

In this work, we use holistic features learned from a CNN network (HWNet) [14] on handwritten word images. The activation features from the last fully connected (FC) layer are found to be generic enough to build robust word spotting systems in a query-by-example setting. We use the same CNN architecture as proposed in [14] which has been pre-trained on a large corpus of synthetic handwritten word images and later fine-tuned on a real world corpus. The transfer of domain from synthetic to real world data leads to better feature learning and quick adaption to different writing styles.

A. HWNet Architecture

HWNet consists of five convolutional layers with 64, 128, 256, 512 and 512 square filters of dimensions 5, 5, 3, 3 and 3 respectively. The next two layers are fully connected with 2048 neurons each. The last layer uses a fully connected (FC) layer with the dimension equal to the number of word classes and is further connected to the soft-max layer to compute the class specific probabilities. It uses a multinomial logistic regression loss function to predict the class labels, and the weights are updated using the mini batch gradient descent algorithm. The network is pre-trained using the subset of IIIT-HWS [3] synthetic handwritten word image corpus of size 1M and later fine tuned on a real world corpus to learn the natural variations in writer styles. The activation features from the second fully connected layer of dimension 2048 are taken as the holistic representation for word images. To make learning invariant to affine transformations, we apply a random amount of rotation ($+/-5$ degrees), shear ($+/-0.5$ degrees along horizontal direction) and perform translation in terms of padding on all four sides to simulate incorrect segmentation of words.

A fundamental assumption used while training HWNet using a multinomial logistic regression loss function is that each class is assumed to be independent. In reality, different classes of word images share a considerable amount of visual information. For example, the words "School" and "Schooling" differ by just an inflection of "ing" in the suffix part of the second word, and we would prefer the feature space to obey the corresponding lexical ordering. Note that in this work, we only focus on lexical similarity and not on the actual semantics, i.e., in case of the words "car" and "cat", both will be constrained to be near, although they differ a lot in the semantic space. One can argue that such sharing of information is implicitly learned in the convolutional layers of the network. However, there is a need to make such relationships more explicit. In this work, we exploit the fine-grained relationships present among the word images using the word attribute framework [8] along with embedding the label information into a common reduced subspace where both the text and image representations lie close to each other. Given such a reduced space, we have the following advantages:- (i) The reduced space is of a much lower (~ 200) dimensions as compared to the original 2048 dimensions with no loss in accuracy, (ii) seamlessly enable both query-by-string and query-by-example based retrieval, and (iii) less memory footprint which enables large scale retrieval and recognition.

B. Text and Image Embedding

Let $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_n\}$ be the set of n images from the training data and $\mathcal{Y} = \{\mathcal{Y}_1, \mathcal{Y}_2, \ldots, \mathcal{Y}_n\}$ be the corresponding text labels. In [8], [13], a new representation called as pyramidal histogram of characters (PHOC) was introduced which maps the text label into word attributes space. The concept is similar to spatial pyramid pooling in natural images but in case of the text, the divisions of pyramids is done in a horizontal direction where at level n, the entire text is divided into n equal parts. From each part, the histogram of uni-grams (characters) and bi-grams are extracted and concatenated for final representation. We refer to this representation as word attributes [8] since it encodes the presence or absence of a particular character in a specific region of the text. As argued earlier, such a representation enables sharing between the words with lexical similarities.

Figure 2. Sample word images from datasets used in this paper. Top row IAM [21], middle row GW [2] and bottom row Bentham [22]

For e.g. the Euclidean distance in attribute space would be relatively less for words such as "car" and "cat" as compared to word "the". Let $\phi_{\mathcal{Y}} : \mathcal{Y} \rightarrow \mathbb{R}^d$ be the text label embedding function which gives the PHOC representation. Here d is the number of attributes which is equivalent to PHOC dimension.

Let $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$ where $\mathcal{X}_i$ is the CNN representation for $\mathcal{I}_i$ word image. We can learn a similar attribute space for CNN features by training d attribute classifiers which predict the probability of a particular attribute given its image representation. This would result in having both text and images in a $\mathbb{R}^d$ space and mutually comparable although not in an optimal manner. We train the attribute classifiers using linear SVMs since the CNN features act as explicit feature maps which encode the non-linearities present in feature space. Here each attribute classifier is trained discriminatively to predict a particular attribute such as "the presence of character 'x' in the first half of the word image" and so on. Given d attribute classifiers, the attribute embedding function is given as $\phi_{\mathcal{X}} : \mathcal{X} \rightarrow \mathbb{R}^d$ which encodes the classifier score in predicting each attribute.

As mentioned earlier, even though both image and text representation lie in $\mathbb{R}^d$ space, they are not optimally comparable because the scores lie in different ranges. To calibrate the scores along with maximizing the correlation between the multivariate vectors (text and images) we formulate the problem as common subspace regression (CSR) with a closed form solution as proposed in [8]. In CSR the objective function is to minimize the distance between an image and its corresponding text representation in a common subspace ($\mathbb{R}^{d'}$) found by the optimal projection matrices. In typical cases, $d' < d$, which is set empirically by tuning on a validation dataset. Note that in the common subspace, both word spotting and recognition can be posed as a simple nearest neighbor based search.

III. Experiments and Analysis

We use three most popular datasets used in the handwritten document analysis community. Table I shows different datasets and its statistics in terms of pages, words and number of writers. Here, the GW and Bentham datasets are historical collections where one can find a peculiar writing style and different set of ligatures as compared to a modern dataset such as IAM. Figure 2 shows few sample word images taken from these datasets.

Table I
WE USE THREE STANDARD DATASETS IAM, GW AND BENTHAM. HERE GW AND BENTHAM DATASETS ARE HISTORICAL DOCUMENTS WRITTEN PRIMARILY BY A SINGLE AUTHOR ALONG WITH A FEW ASSISTANTS(*).

Dataset	Historical	#Pages	#Words	#Writers
IAM [21]	No	1,539	1,15,320	657
GW [10]	Yes	20	4,894	1*
Bentham [22]	Yes	796	1,54,470	1*

IAM Handwriting Database [21]: It includes contributions from 657 writers, making a total of 1539 handwritten pages comprising of 115,320 words. The database is labeled at the sentence, line and word levels. We use the standard partition for training, testing, and validation provided along with the corpus.

George Washington (GW) [2]: It contains 20 pages of letters written by George Washington and his associates in 1755. Images are also annotated at the word level and contain approximately 5000 words. Since there is no official partition, we use a random set (similar to [8]) of 75% for training and validation, and the remaining 25% for testing.

Bentham manuscripts [22]: It is a large set of documents written by the renowned English philosopher and reformer Jeremy Bentham (1748-1832) and his secretaries that are currently being transcribed under the Transcribe Bentham project. Under the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task [23], a training set of 363 pages and a development set of 433 pages were provided. We use the development set as our test set and use the training set for both validation and training. In total, 796 pages of annotated manuscripts are used, containing 154,470 labeled and annotated words. For our query set, we use the set of single word queries without stop words which was provided by ImageCLEF for their development set. A total of 73 queries are used.

For comparing results, we use a standard information retrieval evaluation measure, mean Average Precision (mAP) which is equal to the mean area under the precision-recall curve. The selection of queries follows the protocol used in [8] where we filter the stopwords from the test corpus while all words (including stopwords as distractors) are kept in the dataset where the search is performed. In the query-by-string (QBS) scenario, we only take unique words from the test dataset vocabulary as queries. In the query-by-example (QBE) case, since the query image is taken from the corpus, the first retrieved image is not included in the mAP calculation. Also, the evaluation of the performance is done in a case-insensitive manner.

A. Word Spotting

Table II shows quantitative results of word spotting on all three datasets for both QBE and QBS setting using the

Figure 3. Qualitative results of word spotting on datasets. The first three rows correspond to the IAM dataset, the next two rows to the GW dataset and the last three rows to the Bentham dataset respectively. Retrieved results which are relevant to the query are outlined in green while the false positives are marked in red.

proposed method and compares it with recent state of the art methods. We use the Fisher Vector (FV) representation [26] as our baseline method, which is computed from SIFT features, reduced to 64 dimensions using PCA, and then aggregated into the Fisher Vector. Note that the Fisher representations are not learned in a supervised setting and thus cannot be directly compared to other methods. It emphasizes the importance of supervised techniques to capture the multi-writer styles and its variations. The HWNet [14] architecture, originally trained for the QBE scenario, shows a significant performance gain as compared to the embedded attribute framework (KCSR) trained on FV mainly because of robust CNN features. We tested the QBS case in HWNet by synthesizing textual queries from synthetic fonts and reported a mAP of 0.7037 on IAM, which is slightly inferior to KCSR. The generalization of HWNet for synthetic queries was quite natural since the original network was pre-trained on the synthetic dataset and thereby the performance did not deteriorate much.

The proposed method, which uses CNN features using embedded attribute framework, gives superior results in both QBE and QBS scenarios on all the three datasets. Note that we use 10 uni-gram levels and 6 levels of bi-grams for PHOC representation. In case of bi-grams we only take the top-50 commonly used ones. We also perform test side augmentation similar to [9] where given a test image we make its additional variants using a combination of rotations, shear and translation with same parameters as done in training. The final representation is taken to be the norm of the sum of all the individual representations. We achieve better results from the original HWNet features along with a significant reduction in dimensionality from 2048 to 200 (for IAM). This is quite important from the perspective of building scalable retrieval systems. In terms of QBS, the improvements are quite high, where we now report an mAP of $\sim 91\%$ on IAM, 92% on GW and 86% on the Bentham corpus. We also performed a hybrid query expansion (HQExp) similar to [8] where both the text and image based representations are combined in a weighted manner to form the query which is used for searching. Here again, we consistently perform better in all of the cases. In general, we see the performance on Bentham corpus is lesser than other datasets mainly due to incorrect segmentation of words and, presence of historical ligatures which can be seen in the last row of Figure 2. In Figure 3, we present few qualitative results. Our system is able to retrieve word images successfully irrespective of style variation, upper and lower case keywords and degradation. There are a few false positives that are marked in red boxes which are visually quite similar and hence confused with the true positives.

Since the original HWNet was trained on a large synthetic dataset, we would like to measure the performance of proposed features on out of vocabulary (OOV) words in the test set. In Table III, the performance on both OOV words

Table II
QUANTITATIVE EVALUATION OF WORD SPOTTING ON STANDARD DATASETS IN BOTH QUERY-BY-EXAMPLE (QBE) AND QUERY-BY-STRING (QBS) PARADIGMS.

Dataset	Method	QBE-mAP	QBS-mAP
IAM	DTW	0.1230	-
	FV	0.1566	-
	Frinken et. al. [24]	-	0.7800
	Sebastian et. al. [15]	0.7251	0.8297
	KCSR [8]	0.5573	0.7372
	KCSR+HQExp [8]	-	0.7570
	HWNet [14]	0.8061	0.7037
	Proposed	**0.8424**	**0.9158**
	Proposed+HQExp	-	0.9152
GW	DTW	0.6063	-
	SC-HMM [25]	0.5300	-
	FV	0.6272	-
	Frinken et. al. [24]	-	0.8400
	Sebastian et. al. [15]	**0.9671**	0.9264
	KCSR [8]	0.9290	0.9111
	KCSR+HQExp [8]	-	0.9674
	HWNet [14]	0.9484	0.6129
	Proposed	0.9441	**0.9284**
	Proposed+HQExp	-	0.9726
Bentham	FV	0.3738	-
	KCSR [8]	0.7451	0.7689
	KCSR+HQExp [8]	-	0.8313
	HWNet [14]	0.8121	0.6089
	Proposed	**0.8641**	**0.8634**
	Proposed+HQExp	-	0.8986

Table III
OUT OF VOCABULARY ANALYSIS ON IAM DATASET. THE RESULTS REPORTED SHOWS THE GENERALIZATION OF THE PROPOSED FEATURES FOR OOV QUERIES.

	Query	Proposed Method	KSCR [8]
In Vocab.	QBE	83.42	49.66
	QBS	91.25	71.64
Out of Vocab.	QBE	87.13	57.45
	QBS	91.91	73.89

and in vocabulary words are shown for the IAM dataset and compared with the KCSR [8] method. Note that for the proposed method, the vocabulary comprises of union of words present in the training data of synthetic dataset and the training corpus of IAM. We observe that the performance is slightly better for OOV words which is not quite usual. On further analysis we found that the OOV words are typically rarer words and larger in size (in terms of no. of characters) which gives enough context information for extracting better features. Moreover it also shows the advantages of the attribute based framework and its sharing property which makes it applicable to zero-shot learning.

Table IV
WORD RECOGNITION PERFORMANCE OF THE PROPOSED METHOD IN COMPARISON WITH STATE OF THE ART METHOD FROM BOTH RECOGNITION DOMAINS. HERE (T) REFERS TO TEST CORPUS LEXICON AND (L) REFERS TO A LARGE LEXICON HAVING NEARLY 90K WORDS.

Method	WER	CER
Almazán et al. [8]	20.01	11.27
Bluche et al. [19]	11.90	4.90
Doetsch et al. [17]	12.2	4.70
Arik et al. [9]	**6.45**	**3.44**
Proposed Method (T)	6.69	3.72
Proposed Method (L)	14.07	6.33

B. Word Recognition

In word image recognition, the task is to find out the transcription of a given query word. We use a lexicon based approach for the IAM dataset, where we limit the transcription to words appearing in lexicons derived from the test set of IAM. We use two evaluation measures, namely the mean word error rate (WER) and the mean character error rate (CER). The CER between two words is defined as the Levenshtein distance between the two words by computing the minimum number of character insertions, deletions and substitutions required to transform one string to another, normalized with respect to the number of characters in a word. The mean WER is defined as the average percentage of words that are wrongly transcribed. In Table IV, we compare our proposed method with techniques in the word spotting domain and also with the state of the art handwritten text recognition systems [9], [17]. Here, we obtain better results than state of the art RNN based framework, which uses costly pre-processing techniques and language models. We report a mean WER of 6.69 and mean CER of 3.72 which is comparable with recent n-gram based CNN framework [9] with a very deep architecture. Figure 4 shows few challenging word images from the IAM dataset and their recognized outputs.

Usually, on increasing the lexicon size, lexicon based recognition methods perform badly. In order to test such a scenario, we took a large vocabulary of size 90K words from a popular open source English dictionary Hunspell and added it into the existing test corpus lexicon. This indirectly acts as a distraction in the recognition process and enables us to measure the robustness of the proposed features. As shown in the table, the results using the large lexicon (L) hasn't deteriorated much.

IV. CONCLUSION

In this work, we present a framework for robust word spotting and recognition in handwritten word images using deep feature embedding learned from a CNN architecture. The generality of the attribute based framework in both the images and the text helped us to exploit and represent

Figure 4. Word recognition results from the proposed method. Many of the words are quite challenging and most of the incorrect words have some order of ambiguity in the original image space.

both modalities into a common subspace which seamlessly enabled query-by-string and query-by-example paradigms. The reduced dimensionality of the final subspace will be a boost for developing scalable and robust systems in the future. In addition to this, we believe that this work has opened up new directions for developing handwritten word recognition systems using CNN architectures which were until recently attempted using recurrent neural networks such as BLSTMs. In the future, we would like to relax the assumption of having segmented words and prefer to work in a near segmentation-free approach.

ACKNOWLEDGMENT

Praveen Krishnan is supported by TCS Research Scholar Fellowship 2013.

REFERENCES

[1] R. J. Milewski, V. Govindaraju, and A. Bhardwaj, "Automatic recognition of handwritten medical forms for search engines," *IJDAR*, 2009.

[2] T. M. Rath and R. Manmatha, "Word spotting for historical documents," *IJDAR*, 2007.

[3] P. Krishnan and C. Jawahar, "Generating Synthetic Data for Text Recognition," in *CoRR*, 2016.

[4] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *IJCV*, 2016.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] A. S. Razavian, J. Sullivan, A. Maki, and S. Carlsson, "A baseline for visual instance retrieval with deep convolutional networks," *arXiv preprint arXiv:1412.6574*, 2014.

[7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," *arXiv preprint arXiv:1310.1531*, 2013.

[8] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," *PAMI*, 2014.

[9] A. Poznanski and L. Wolf, "CNN-N-Gram for handwriting word recognition," in *CVPR*, 2016.

[10] A. Fischer, A. Keller, V. Frinken, and H. Bunke, "Lexicon-free handwritten word spotting using character HMMs," *PRL*, 2012.

[11] R. Shekhar and C. V. Jawahar, "Word image retrieval using bag of visual words," in *DAS*, 2012.

[12] M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós, "Efficient segmentation-free keyword spotting in historical document collections," *PR*, 2015.

[13] J. A. Rodriguez-Serrano, A. Gordo, and F. Perronnin, "Label embedding: A frugal baseline for text recognition," *IJCV*, 2015.

[14] P. Krishnan and C. Jawahar, " Matching Handwritten Document Images," in *ECCV*, 2016.

[15] S. Sudholt and G. A. Fink, "PHOCNet: A deep convolutional neural network for word spotting in handwritten documents," *CoRR*, 2016.

[16] S. E. Boquera, M. J. C. Bleda, J. Gorbe-Moya, and F. Zamora-Martínez, "Improving offline handwritten text recognition with hybrid HMM/ANN models," *PAMI*, 2011.

[17] P. Doetsch, M. Kozielski, and H. Ney, "Fast and robust training of recurrent neural networks for offline handwriting recognition," in *ICFHR*, 2014.

[18] V. Pham, T. Bluche, C. Kermorvant, and J. Louradour, "Dropout improves recurrent neural networks for handwriting recognition," in *ICFHR*, 2014.

[19] T. Bluche, H. Ney, and C. Kermorvant, "A comparison of sequence-trained deep neural networks and recurrent neural networks optical modeling for handwriting recognition," in *SLSP*, 2014.

[20] H. Hotelling, "Relations between two sets of variates," *Biometrika*, 1936.

[21] U.-V. Marti and H. Bunke, "The IAM-database: an english sentence database for offline handwriting recognition," *IJDAR*, 2002.

[22] T. Causer and V. Wallace, "Building a volunteer community: results and findings from transcribe bentham," *Digital Humanities Quarterly*, 2012.

[23] M. Villegas, J. Puigcerver, A. H. Toselli, J. A. Sánchez, and E. V. from UPV. ImageCLEF handwritten scanned document retrieval task 2016. [Online]. Available: http://www.imageclef.org/2016/handwritten

[24] V. Frinken, A. Fischer, R. Manmatha, and H. Bunke, "A novel word spotting method based on recurrent neural networks," *PAMI*, 2012.

[25] J. A. Rodríguez-Serrano and F. Perronnin, "A model-based sequence similarity with application to handwritten word spotting," *PAMI*, 2012.

[26] F. Perronnin and J. A. Rodríguez-Serrano, "Fisher kernels for handwritten word-spotting," in *ICDAR*, 2009.