A DEEP LEARNING APPROACH TO DOCUMENT IMAGE QUALITY ASSESSMENT

Le Kang[1]*, Peng Ye*[1]*, Yi Li*[2]*, and David Doermann* [1]

[1]University of Maryland, College Park, MD, USA
[2]NICTA and ANU, Canberra, Australia

[1]{lekang,pengye,doermann}@umiacs.umd.edu [2]yi.li@cecs.anu.edu.au

ABSTRACT

This paper proposes a deep learning approach for document image quality assessment. Given a noise corrupted document image, we estimate its quality score as a prediction of OCR accuracy. First the document image is divided into patches and non-informative patches are sifted out using Otsu's binarization technique. Second, quality scores are obtained for all selected patches using a Convolutional Neural Network (CNN), and the patch scores are averaged over the image to obtain the document score. The proposed CNN contains two layers of convolution, location blind max-min pooling, and Rectified Linear Units in the fully connected layers. Experiments on two document quality datasets show our method achieved the state of the art performance.

Index Terms— Convolutional neural networks, document, image quality

1. INTRODUCTION

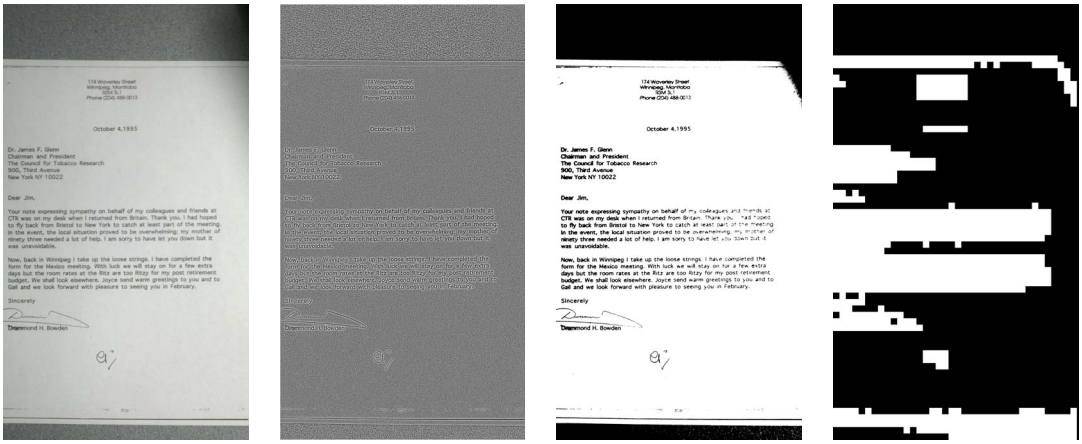
This paper presents a deep learning approach to no-reference quality assessment of document images. Document quality has a direct impact on the OCR performance. Thus it is desirable to estimate document quality before applying OCR. In this paper, we assume the quality of a degraded document image is directly correlated with the performance of optical character recognition (OCR) software run on it. A document quality prediction system can be used in many practical applications [1]. For example, it can be used to filter out highly degraded document image for which the OCR system will fail, or it can also be used to select high quality document frames in a video capture system [2]. Moreover, when applying a document enhancement method we may be able to avoid further degradation under the guidance of a quality measure.

Early work on document image quality assessment (DIQA) focused on deriving solutions for specific types of document degradations and relied on hand-crafted features. In [3, 4, 5], several quality factors for typewritten document images are proposed including: Font Size (FS), Small Speckle Factor (SSF), Stroke Thickness Factor (STF), White Speckle Factor (WSF) and Broken character factor (BCF). These metrics were computed empirically based on connected-components and were chosen because they may have a high correlation with the OCR error rate. They have been used to predict the OCR accuracy and to choose the best restoration method for preprocessing. However, these metrics cannot be directly applied to general document distortions for the following reasons: 1) the computation of these metrics depends on font size. Thus they are only effective under the assumption that the sizes of individual characters are similar. However, a complex document image may contain characters of different font sizes or stroke sizes, and some scripts such as Arabic typically show varying stroke sizes. 2) The touching of handwritten characters could be due to the writing style of writer, and not typically related to quality.

Recently, Kumar et al. [6] proposed a sharpness measure for camera-captured document images, which is specifically designed to measure the blur distortion and may only be applied to camera-captured document images. The first general-purpose method seen in the literature is the feature learning method introduced by Ye et al. [7], which is an extension of the CORNIA system [8]. This method is based on unsupervised feature learning which can automatically learn discriminant features for different types of document degradations. However, this is a rather empirical feature learning solution. We propose a deep learning method for DIQA, which provides a more unified and principled way for feature learning and regression.

Recently the research community has seen great success using deep learning for computer vision tasks. The Convolutional Neural Network (CNN) is one of the most widely used methods for object detection/recognition [9, 10, 11]. Kang et al. [12] used CNN in no-reference image quality assessment for natural images and achieved the state of the art performance. With minimal preprocessing on raw images, CNNs efficiently learn features and a classifier/regressor in one process, typically with stochastic gradient decent and backpropagation. In this way, learned features tend to be more effective than handcrafted features, and the process is clean and clear.

The partial support of this research by DARPA through BBN/DARPA Award HR0011-08-C-0004 under subcontract 9500009235, the US Government through NSF Awards IIS-0812111 and IIS-1262122 is gratefully acknowledged.



(a) (b) (c) (d)

Fig. 1. (a) A document image of size 1860×3264 (b) local normalization result (intensity rescaled for better visualization) (c) binary map obtained from the original image using Otsu's method (d) mask of non-constant 48×48 patches (white).

The proposed CNN contains a special max-min pooling which reduces the unnecessary computation while keeping important features for quality estimation. Exploiting the homogeneous nature of typical distortions, we divide document images into patches to significantly increase the number of training samples. This allows our method to work for large images. An efficient patch selection process is employed and only informative patches are fed to the CNN. The proposed method shows state of the art performance on two document quality datasets.

2. APPROACH

We first introduce the overall process of estimating document image quality. We preprocess a grayscale document image with local normalization, crop the image into patches, use the CNN to estimate quality scores for selected patches, and average the scores to obtain a score for the image. The goal is to predict quality scores that correlate with OCR accuracies as much as possible.

2.1. Preprocessing

Preprocessing is typically required for general image quality assessment, in order to be robust to intensity and contrast change. As in [13], we perform a local normalization over the entire image. Each pixel is subtracted by the mean and divided by standard deviation of the pixels in a surrounding window. Fig.1(a) and (b) show a document image and its local normalization result.

2.2. Patch sifting

We perform Otsu's binarization [14] on the raw image, and obtain a binary map corresponding to foreground and background. We crop patches from the preprocessed (i.e. locally normalized) images and check their corresponding patches on the binary map. If the patch on the binary map is constant, i.e. all ones or all zeros, then this patch is discarded. Since the patch size is chosen to be larger than the typical stroke width, text patches are preserved. Most patches sifted out in this way are background patches or non-text foreground patches. Fig. 1 (c) shows the result of Otsu's binarization, and Fig. 1 (d) shows the locations of patches that are selected after sifting.

Dividing an image into patches has two major benefits. First, it is easier for the CNN to handle patches instead of the entire image. Document images in one of the datasets have the size 1860×3264, which makes it nearly impossible to process with a CNN. But an image patch (e.g. 48×48) can be comfortably processed by a CNN of reasonable size. Second, by dividing images into patches, the number of samples is significantly increased, which is typically desired for training a CNN.

Since we predict quality with respect to OCR performance, we would like to focus on the patches that contain characters. In our document dataset, most image content is either text or background, thus we just need to sift out the background patches and use the rest for quality estimation.

2.3. Network Architecture

Once the patches are obtained, we feed them into a network. Fig. 2 shows the architecture of the proposed network. The network contains two convolution and pooling layers, two fully connected layers and one output layer. The input is sifted patches of size 48×48. The first convolution layer contains 40 kernels each of size 5×5, followed by a 4×4 max pooling, then the second convolution layer with 80 kernels each of size 5×5. Following the second convolution layer is a special max-min pooling that we will explain later. Each of the two

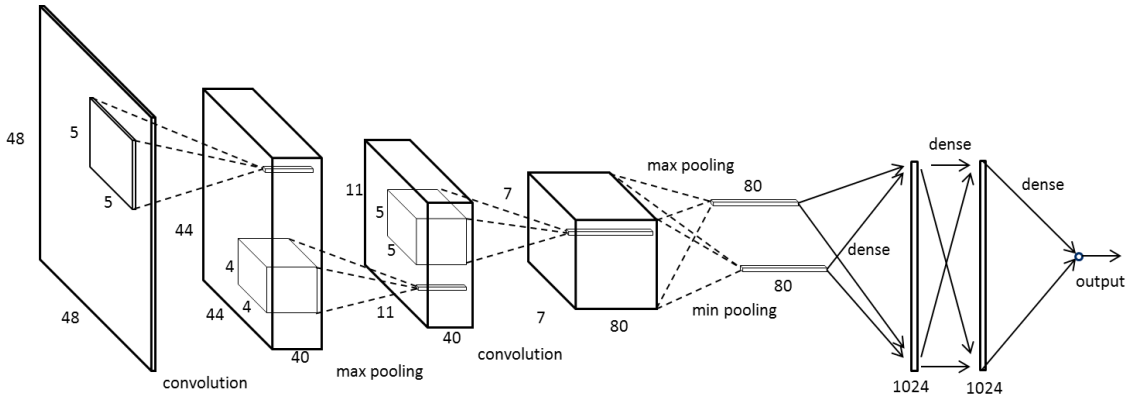


Fig. 2. The architecture of the proposed CNN.

fully connected layers contains 1024 nodes. The last layer is a linear regression that outputs the quality score.

We use Rectified Linear Units (ReLUs) [15] as the neurons in the two fully connected layers. Formally ReLUs can be expressed as $f(x) = max(0, x)$, where x denotes the input. ReLUs let positive signals pass and suppresses the negative signals. Compared to traditional sigmoid or tanh neurons, ReLUs are robust to input range and leads to faster training as demonstrated in [10]. It is worth noting that in convolution layers no nonlinear transform is applied, or equivalently a linear neuron ($f(x) = x$) is applied, since in experiments we didn't observe any benefit from using ReLUs in the convolution layers.

As we mentioned previously, a special max-min pooling after the second convolution layer is used. Specifically, each feature map obtained by the second convolution layer is pooled into *one* max value and *one* min value. Suppose there are 80 kernels in the second convolution layer, after max-min pooling we get 80 max values and 80 min values for a total of 160 outputs. Through the max-min pooling, the location information of features is discarded, but the filter responses characterized by maxs and mins are enough to capture the statistics of the quality.

2.4. Learning Procedure

By cropping images into patches we have plenty of training samples. In fact the important thing is that the training patches are all labeled. We simply make the labels of patches the same as the ground truth OCR scores of their original images.

During training, for each patch we try to train the network to predict a score close to the ground truth. The error between the last layer's output (predicted score) and the patch's ground truth is measured by the l_1 norm. We use Stochastic Gradient Decent (SGD) and backpropagation to solve the minimization and update the parameters. Training is performed on mini-batches of samples for a given number of epochs, and we select the model parameters that achieve the best performance on the validation set.

3. EXPERIMENTS

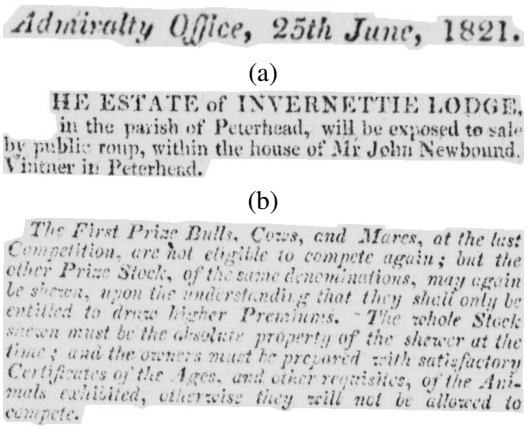
3.1. Dataset and protocol

Datasets: We conduct experiments on the following two datasets.

(1) Sharpness-OCR-Correlation (SOC) dataset [16]: a total of 175 color images with resolution 1840×3264. These images are captures of 25 documents using a cell phone camera. Each document contains machine printed English, and 6-8 photos with varying focal lengths were taken to generate different levels of blur. Fig. 1(a) shows a sample image of this dataset. A commercial OCR software (ABBYY Fine Reader) was run on each of the 175 images, and the OCR results were evaluated by the ISRI-OCR evaluation tool [17] to obtain OCR accuracies in the range $[0, 1]$. The OCR accuracy is the ground truth for each image in our quality assessment task.

(2) Newspaper dataset [7]: 521 grayscale images with various resolution. These images are a subset of a historical collection, and contain machine printed English and Greek. Each image in this dataset is a text region instead of an entire page. The OCR accuracies were obtained for each image using ABBYY Fine Reader and ISRI-OCR in the same way as SOC dataset. On this dataset, the OCR performance is mainly affected by broken strokes. Fig. 3 shows sample images from this dataset.

Evaluation protocol: Following the tradition in natural image quality assessment, we compute the correlation between the predicted quality scores and ground truth OCR accuracies. Specifically, we use the Linear Correlation Coefficient (LCC) and the Spearman Rank Order Correlation Coefficient (SROCC) to evaluate the performance of the proposed algorithm and compare it to previous methods. LCC is a measure of the degree of linear relationship between two variables. SROCC measures how well the relationship between



(a)

(b)

(c)

Fig. 3. Sample images from Newspaper dataset

	BRISQUE-L[13]	CORNIA[8]	CORNIA-SF[18]	CNN
LCC	0.904	0.937	0.927	**0.950**
SROCC	0.836	0.862	0.854	**0.898**

Table 1. median LCC and SROCC over 100 random sampling experiments on SOC dataset

two variables can be described using a monotonic function.

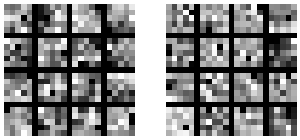
In our experiments, we randomly sample 60% of the data as the training set, 20% as the validation set, and leave the remaining 20% as a test set. This random split of dataset is reasonable for the Newspaper dataset, but not on the SOC dataset. The images of SOC dataset are organized in groups where each group only contains images taken from the same document, thus the random split is conducted at the group level. For both datasets, this random split of data is repeated 100 times, each time the LCC and the SROCC are computed, and we report the median LCC and SROCC over the 100 iterations.

3.2. Evaluation

Implementation: The proposed network is implemented using the python library Theano [19]. With Theano we are able to easily run our algorithm on a GPU to speed up the process without much optimization.

Evaluation on SOC: On the SOC dataset, the image resolution is high thus we use a relatively large patch size of 48. With this patch size we get roughly 9×10^4 training patches, 3×10^4 validation patches and 3×10^4 test patches. We show the experimental results and compare with previous approaches in Table 1. The proposed method achieved a higher LCC and SROCC than other competing methods.

We visualize the learned filters in the first convolution layers, in Fig. 4(a). The learned filters do not show patterns that are immediately intuitive to human. Some patterns tend to gradually change intensity along a direction which may correspond to blurred character boundary, while most patterns seem to be of irregular structure. We believe this is due to the fact that the filters are learned from locally normalized images instead of the original images.



(a) (b)

Fig. 4. Learned convolution kernels on (a) SOC dataset (b) Newspaper dataset

	BRISQUE-L	CORNIA	CORNIA-SF	CNN
LCC	0.722	**0.751**	0.735	0.731
SROCC	0.709	0.725	0.708	**0.726**

Table 2. median LCC and SROCC over 100 random sampling experiments on Newspaper dataset

Evaluation on Newspaper: The image size has large variations on the Newspaper dataset. Images can be as small as 569×38. Also the characters in the images are much smaller than those in SOC. Thus a smaller patch size of 32 is used for this dataset. There are approximately 6×10^4 patches for training, and 2×10^4 each for validation and test. Table 2 shows the experimental results on Newspaper dataset. Our method achieved similar performance compared to the state of the art. All competing methods show a decrease of performance on this dataset. The major noise present in Newspaper dataset is eroded/broken strokes, which are likely part of the document itself rather than introduced by the imaging process. This inherent distortion mainly relates to the semantics, thus it may not be measured well by those methods that typically focus on statistics of low level features.

We also visualize the first convolution layer filters learned on Newspaper dataset in Fig. 4(b). It is also difficult to find obvious structures from the learned filters.

4. CONCLUSION

We proposed a CNN based method for document image quality assessment. Our method first predicts quality scores on document image patches, then the patch scores are averaged to obtain a image quality estimation. Dividing the image into patches brings abundant samples for training CNN. The proposed CNN effectively learns quality related features and achieved the state of the art performance.

5. REFERENCES

[1] Peng Ye and David Doermann, "Document Image Quality Assessment: A Brief Survey," *2013 12th International Conference on Document Analysis and Recognition*, pp. 723–727, Aug. 2013.

[2] Jayant Kumar, Raja Bala, Hengzhou Ding, and Phillip Emmett, "Mobile Video Capture of Multi-page Documents," *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 35–40, June 2013.

[3] L.R. Blando, J. Kanai, and T.a. Nartker, "Prediction of OCR accuracy using simple image features," *Proceedings of 3rd International Conference on Document Analysis and Recognition*, vol. 1, pp. 319–322, 1995.

[4] Andrea Souza, Mohamed Cheriet, Satoshi Naoi, and Ching Y Suen, "Automatic filter selection using image quality assessment," in *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*. IEEE, 2003, pp. 508–512.

[5] Michael Cannon, Patrick Kelly, and Judith Hochberg, "Quality assessment and restoration of typewritten document images," *International Journal on Document Analysis and Recognition*, vol. 2, no. 2-3, pp. 80–89, Dec. 1999.

[6] Jayant Kumar, Francine Chen, and David Doermann, "Sharpness estimation for document and scene images," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 3292–3295.

[7] Peng Ye and David Doermann, "Learning features for predicting ocr accuracy," in *Pattern Recognition (ICPR), 2012 21st International Conference on*. IEEE, 2012, pp. 3204–3207.

[8] Peng Ye, Jayant Kumar, Le Kang, and David Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1098–1105.

[9] Koray Kavukcuoglu, Pierre Sermanet, Y-Lan Boureau, Karol Gregor, Michaël Mathieu, and Yann LeCun, "Learning convolutional feature hierachies for visual recognition," in *Advances in Neural Information Processing Systems (NIPS)*, 2010.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[11] Dan Claudiu Ciresan, Ueli Meier, and Jürgen Schmidhuber, "Multi-column deep neural networks for image classification," in *Computer Vision and Pattern Recognition*, 2012, pp. 3642–3649.

[12] Le Kang, Peng Ye, Yi Li, and David Doermann, "Convolutional neural networks for no-reference image quality assessment," in *IEEE Conf. On Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[13] A. Mittal, A. Moorthy, and A. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[14] Nobuyuki Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[15] Vinod Nair and Geoffrey E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010, pp. 807–814.

[16] Jayant Kumar, Peng Ye, and David Doermann, "A Dataset for Quality Assessment of Camera Captured Document Images," *International Workshop on Camera-Based Document Analysis and Recognition (CBDAR)*, pp. 39–44, 2013.

[17] Ray Smith, "Isri-ocr evaluation tool," `http://code.google.com/p/isri-ocr-evaluation-tools/`.

[18] Peng Ye, Jayant Kumar, Le Kang, and David Doermann, "Real-time no-reference image quality assessment based on filter learning," in *Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 987–994.

[19] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio, "Theano: a CPU and GPU math expression compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, Jun. 2010.