Geometric Representation Learning for Document Image Rectification

Hao Feng[1], Wengang Zhou[1,2⋆], Jiajun Deng[1],
Yuechen Wang[1], and Houqiang Li[1,2⋆]

[1] CAS Key Laboratory of GIPAS, EEIS Department,
University of Science and Technology of China
{fh1995,wyc9725}@mail.ustc.edu.cn, {zhwg,dengjj,lihq}@ustc.edu.cn
[2] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

Abstract. In document image rectification, there exist rich geometric constraints between the distorted image and the ground truth one. However, such geometric constraints are largely ignored in existing advanced solutions, which limits the rectification performance. To this end, we present DocGeoNet for document image rectification by introducing explicit geometric representation. Technically, two typical attributes of the document image are involved in the proposed geometric representation learning, *i.e.*, 3D shape and textlines. Our motivation raises from the insight that 3D shape provides global unwarping cues for rectifying a distorted document image, while overlooking the local structure. On the other hand, textlines complementarily provide explicit geometric constraints for local patterns. The learned geometric representation effectively bridges the distorted image and the ground truth one. Extensive experiments show the effectiveness of our framework and demonstrate the superiority of our DocGeoNet over state-of-the-art methods on both the DocUNet Benchmark dataset and our proposed DIR300 test set.

Keywords: Document Image Rectification, Geometric Constraints

1 Introduction

With the popularity of smartphones, more and more people are using them to digitize document files. Compared to typical flatbed scanners, smartphones provides a flexible, portable, and contactless way for document image capturing. However, due to uncontrolled physical deformations, uneven illuminations, and various camera angles, those document images are always distorted. Such distortions make those images invalid in many formal review occasions, and are likely to cause the failure of the downstream applications, such as automatic text recognition, analysis, retrieval, and editing. To this end, over the past few years, document image rectification has become an emerging research topic. In this work, we focus on the geometric distortion rectification for document images, aiming to rectify arbitrarily warped documents to their original planar shape.

[⋆] Corresponding Authors: Wengang Zhou and Houqiang Li.

Traditionally, document image rectification is addressed by 3D reconstruction. Generally, the 3D mesh of the warped document is estimated to flatten the document image. However, such techniques are either based on auxiliary hardware [1,3,28,46] or developed with multiview images [2,16,44,45], which are unfriendly in personal application. Some other methods assume a parametric model on the document surface and optimize the model by extracting specific representations such as shading [4], boundaries [12], textlines [17,40], or texture flow [20]. However, the oversimplified parametric models usually lead to limited performance, and the optimization process introduces non-negligible computational cost. Recently, deep learning based solutions [6,7,9,10,19,22,25] have been become a promising alternative to traditional methods. By training a network to directly predict the warping flow, a deformed document image can be rectified by resampling the pixels in the distorted image. Although these methods are reported with the state-of-the-art performance, the rich geometric constraints between distorted images and ground truth ones are largely ignored.

Generally, in a document image, the texture mainly exists in textlines. Note that there are strong geometric constraints among textlines between the distorted and ground truth image, that is, the curved textlines should be straight after rectification if they are horizontal textlines in a document. In other words, textlines provide a strong cue for the rectification. However, existing methods all just learn this prior implicitly with deep networks via the supervision on predicted warping flow, which leads to sub-optimal performance. Besides, compared to a distorted document image, the attribute of 3D shape is a more explicit representation that directly determines the unwarping process. The above two attributes bridge the distorted and ground truth image and complement each other: the distribution of textlines reflects the local deformation of a document, which serves as a complement to 3D shape on local structure detail. Based on the above motivation, we explicitly learn the geometric constraints from such attributes in a deep network to promote the rectification performance.

In this work, we present DocGeoNet, a new deep network for document image rectification. DocGeoNet bridges the distorted image and its ground truth by introducing geometric constraint representation derived from document attributes. It consists of a structure encoder, a textline extractor, and a rectification decoder. Specifically, given a distorted document image, DocGeoNet takes the structure encoder and textline extractor to model the 3D shape of the deformed document and extract its textlines, respectively. Then, to take advantage of the complementarity of such two attributes and leverage their direct constraints that link the distorted and ground truth image, we further fuse their representation and predict the rectification in the rectification decoder. During the training of DocGeoNet, the learning of 3D shape, textlines, and rectification is optimized in an end-to-end way. Besides, considering that the 3D shape is a global attribute while the textline is a local attribute, the proposed DocGeoNet adopts a hybrid network structure, which takes advantage of self-attention mechanism [36] and convolutional operation for enhanced representation learning. To evaluate our approach, extensive experiments are conducted on the Doc3D dataset [6], DocUNet Benchmark dataset [25], and our proposed challenging DIR300 Benchmark dataset. The results demonstrate the effectiveness of our method as well as its superiority over existing state-of-the-art methods.

In summary, we make three-fold contributions as follows:

- We present DocGeoNet, a new deep network that performs explicit representation learning of the geometric constraints between the distorted and target rectified image to promote the performance of document image rectification.
- We design a new pipeline to automatically annotate the textlines of the distorted document images in training set. Besides, to reflect the effectiveness of existing works, we propose a new large-scale challenge benchmark dataset.
- We conduct extensive experiments to validate the merits of DocGeoNet, and show state-of-the-art results on the prevalent and proposed benchmarks.

2 Related Work

Rectification Based on 3D Reconstruction. Early methods first estimate the 3D mesh of the deformed document and then flatten it to a planar shape. Brown and Seales [1] deploy a structured light 3D acquisition system to acquire the 3D model of a deformed document. Zhang et al. [46] use a laser range scanner and perform restoration using a physical modeling technique. Meng et al. [28] use two structured beams illuminating upon the document to recover two spatial curves of document surface. Such methods generally rely on auxiliary hardware to scan the deformed documents, which is unfriendly in daily personal use.

On the other hand, some methods make use of multiview images to reconstruct the 3D document model. Tsoi et al. [35] transform the multiple views of a document to a canonical coordinate frame based on the boundaries of the document. Koo et al. [16] build the deformed surface by registering the corresponding points in two images by SIFT [24]. Recently, You et al. [45] propose a ridge-aware surface reconstruction method based on multiview images. However, in the above works, the involvement of multiview shooting limits the further applications.

Some other methods aim to reconstruct the 3D shape from a single view. Typically, they assume a parametric model on the document surface and optimize the model by extracting specific representations, such as shading [4,37], boundaries [12], textlines [13,17,40], or texture flow [20]. Tan et al. [4] build the 3D shape of a book surface from the shading information. He et al. [12] extract a book boundary model to reconstruct the book surface. Cao et al. [13] and Meng et al. [27] represent the surface as a general cylindrical surface and extract textlines to estimate the parameter of the model.

Rectification Based on Deep Learning. For document image rectification, the first learning-based method is DocUNet [25]. By training a stacked UNet [32], it directly regresses a pixel-wise displacement field to correct the geometric distortion. Later, Li et al. [19] propose to rectify the distorted image patches first and then stitch them for rectification. Xie et al. [42] add a smooth constraint to the learning of the pixel-wise displacement field. Recently, Amir et al. [26] propose to learn the orientation of words in a document and Das et al. [6] propose

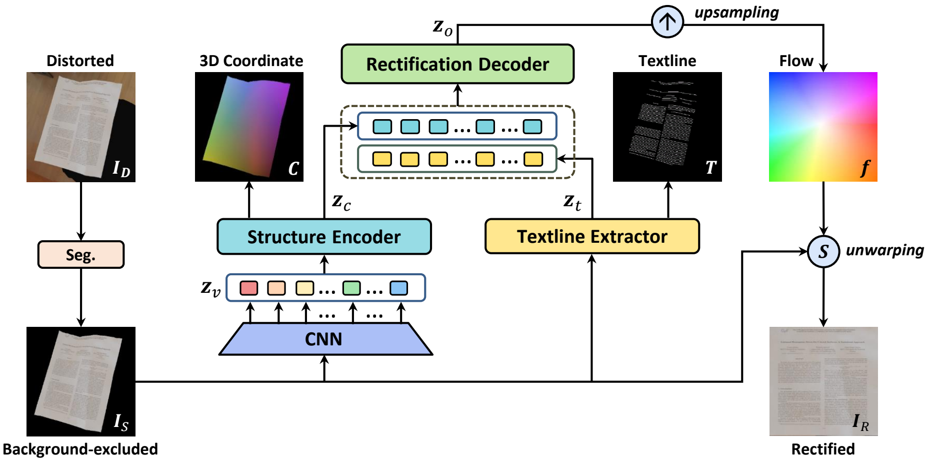

Fig. 1. An overview of our proposed DocGeoNet. It consists of three main components: (1) A preprocessing module that segments the foreground document from the clustered background. (2) A structure encoder and a textline extractor which model the 3D shape of the deformed document and extract the curved textlines, respectively. (3) A rectification decoder that estimates the warping flow for distortion rectification.

to model the 3D shape of a document with a UNet [32]. Feng et al. [9] introduce transformer [36] from natural language processing tasks to improve the feature representation. Das et al. [7] predict local deformation fields and stitch them together with global information to obtain an improved unwarping.

Different from the above methods, in this work, we approach the document image rectification by introducing the representation learning of the geometric constraints that bridge the distorted and the rectified image, which is largely overlooked by the recent state-of-the-art methods.

3 Approach

In this section, we present our Document Image Rectification Network (DocGeoNet) for geometric correction of distorted document images. Given a distorted document image $\boldsymbol{I}_D$, our DocGeoNet estimates a dense displacement field $\boldsymbol{f} = (\boldsymbol{f}^x, \boldsymbol{f}^y)$ as warping flow. Based on $\boldsymbol{f}$, the pixel (i, j) in rectified image $\boldsymbol{I}_R$ can be obtained by sampling the pixel $(i', j') = (i + \boldsymbol{f}^x(i), j + \boldsymbol{f}^y(j))$ in input image $\boldsymbol{I}_D$. As shown in Fig. 1, our framework consists of three key components: (1) preprocessing for background removal, (2) geometric constraint representation learning from two document attributes, including 3D shape and textlines, and (3) representation fusion and geometric rectification. Here, the first preprocessing stage is trained independently, and the latter two stages are differentiable and composed into an end-to-end trainable architecture. In the following, we elaborate the three components separately.

3.1 Preprocessing

For the geometric rectification of document images, taking the whole distorted image as input to the rectification network is a general operation. However, it involves extra implicit learning to localize the foreground document besides predicting the rectification, which limits the performance. Hence, following [9], we adopt a preprocessing operation to remove the clustered background first, thus the following network can focus on the rectification of the distortion.

Specifically, given a distorted RGB document image $\boldsymbol{I}_D \in \mathbb{R}^{H \times W \times 3}$, a lightweight semantic segmentation network [31] is utilized to predict the confidence map of the foreground document. Then, the confidence map is further binarized with a threshold τ to obtain the document region mask $\boldsymbol{M}_{\boldsymbol{I}_D} \in \mathbb{R}^{H \times W}$. After that, the background of $\boldsymbol{I}_D$ can be removed by element-wise matrix multiplication with broadcasting along the channels dimension of $\boldsymbol{I}_D$, and we obtain the background-excluded document image $\boldsymbol{I}_S$. The preprocessing network is trained independently with a binary cross-entropy loss [8] as follows,

$$\mathcal{L}_{seg} = -\sum_{i=1}^{N_p} \left[y_i \log(\hat{p_i}) + (1 - y_i) \log(1 - \hat{p_i}) \right], \tag{1}$$

where $N_p = H \times W$ is the number of pixels in $\boldsymbol{I}_D$, y_i and $\hat{p_i}$ denote the ground-truth and predicted confidence, respectively. The obtained background-excluded document image $\boldsymbol{I}_S$ is fed into the subsequent rectification network.

3.2 Structure Encoder and Textline Extractor

In a document image, textlines are the main texture, which contain direct geometric constraints for rectification. In other words, a distorted curved textline corresponding to a horizontal or vertical one in the ground truth should be straight after rectification. Besides, the distribution of textlines also reflect the deformation of a document. Therefore, textlines provide a strong cue for the rectification. In addition, for geometric rectification, compared to a distorted document image, the 3D shape is a more direct representation that determines the unwarping process. Hence, we propose to model 3D shape and extract the textlines of the deformed document in the network to leverage their geometric constraints that bridge the distorted image and rectified image.

Specifically, as shown in Fig. 1, given a background-excluded document image $\boldsymbol{I}_S$, we adopt two parallel sub-networks to model 3D shape and extract the textlines, respectively. We use a transformer-based [36] sub-network for the learning of 3D shape and a CNN-based sub-network for the learning of textlines. Such a design is adopted based on two considerations. First, each part in a physical distorted paper is interrelated, so we introduce the self-attention mechanism [36] to capture long-distance feature dependencies. Second, whether a pixel belongs to a textline depends more on local features, so we take advantage of convolutional operations here. In the following, we elaborate the two sub-networks, *i.e.*, the structure encoder and the textline extractor.

Structure Encoder. Given a document image $\boldsymbol{I}_S \in \mathbb{R}^{H \times W \times 3}$, a convolutional module consisting of 6 residual blocks [11] generates feature map $\boldsymbol{z} \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times C}$, where the channel dimension C is 128. Here the resolution of feature map decreases by $\frac{1}{2}$ every two blocks. Then, to adapt to the sequence input form of the subsequent transformer encoder [36], we flatten $\boldsymbol{z}$ into a sequence of tokens $\boldsymbol{z}_v \in \mathbb{R}^{N_v \times C}$, where $N_v = \frac{H}{8} \times \frac{W}{8}$ is the number of tokens.

Since that transformer layer is permutation-invariant, to make it sensitive to the original 2D positions of input tokens, we utilize sinusoidal spatial position encodings as the supplementary of visual feature. Concretely, the position encodings are added with the query and key embedding at each transformer encoder layer. We stack 6 transformer encoder layers and each of the encoder layers contains a multi-head self-attention module and a feed forward network. For the i^{th} encoder layer, the output representation is calculated as follows,

$$\begin{aligned}
&\boldsymbol{F}_0 = [\boldsymbol{z}_v], \\
&\boldsymbol{Q}_i = \boldsymbol{W}^Q \boldsymbol{F}_{i-1}, \boldsymbol{K}_i = \boldsymbol{W}^K \boldsymbol{F}_{i-1}, \boldsymbol{V}_i = \boldsymbol{W}^V \boldsymbol{F}_{i-1}, \\
&\boldsymbol{F}_i^{'} = LN(MA(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) + \boldsymbol{F}_{i-1}), \\
&\boldsymbol{F}_i = LN(FFN(\boldsymbol{F}_i^{'}) + \boldsymbol{F}_i^{'}),
\end{aligned} \tag{2}$$

where $W^Q, W^K, W^V \in \mathbb{R}^{M \times C \times C_w}$, $M = 8$ is the number of attention heads, $C_w = 256$ denotes the feature dimension in transformer, $MA(\cdot)$, $FFN(\cdot)$, $LN(\cdot)$ denote the multi-head attention, feed forward network, and layer normalization, respectively. $\boldsymbol{F}_i$ denotes the output feature of the i^{th} encoder layer. The transformer layers conducts global vision context reasoning in parallel, and outputs the advanced visual embedding $\boldsymbol{z}_v'$, which shares the same shape as $\boldsymbol{z}_v$.

We reshape the output feature $\boldsymbol{z}_v' \in \mathbb{R}^{N_v \times C}$ to $\frac{H}{8} \times \frac{W}{8} \times C$. Finally, we upsample the reshaped feature map to match the resolution of the ground truth 3D coordinate map $\boldsymbol{C} \in \mathbb{R}^{H \times W \times 3}$ based on bilinear sampling, followed by a 3 × 3 convolutional layer that reduces the channel dimension to 3. After that, we get the predicted 3D coordinate map $\hat{\boldsymbol{C}} \in \mathbb{R}^{H \times W \times 3}$, in which each pixel value corresponds to 3D coordinates of the document shape.

Textline Extractor. We segment the textlines by the per-pixel binary classification on foreground document region. Given a background-excluded document image $\boldsymbol{I}_S \in \mathbb{R}^{H \times W \times 3}$, a confidence map $\hat{\boldsymbol{T}} \in \mathbb{R}^{H \times W}$ with values in the range of $(0, 1)$ is predicted. It contains the confidence of each pixel (text/non-text).

The textline extractor adopts a compact multi-scale CNN network. It consists of a contracting part, an expansive part and a classification part. For the contracting part, we repeat the application of two 3×3 convolutional layers to encode the texture features from $\boldsymbol{I}_S$, each followed by a rectified linear unit (ReLU) and a 2×2 max pooling operation with stride 2 for downsampling. For the expansive part, after upsampling the feature map based on bilinear interpolation at each scale, we concatenate it with the corresponding feature map from the contracting path, followed by two 3×3 convolutional layers and a ReLU. In the classification part, a 1x1 convolutional layer followed by a Sigmoid function is used to generate the confidence map $\hat{\boldsymbol{T}} \in \mathbb{R}^{H \times W}$.

3.3 Rectification Decoder

Hybrid Representation Learning. To take advantage of the complementarity of the two attributes and leverage their geometric constraints that bridge the distorted and target rectified image, we further fuse their representation and predict the rectification in the rectification decoder. Specifically, we first flatten the $\frac{1}{8}$ resolution representation map in expansive part of the textline extractor into a sequence of 2D features $\boldsymbol{z}_t \in \mathbb{R}^{N_v \times C_t}$. Then, we concatenate it with $\boldsymbol{z}_c \in \mathbb{R}^{N_v \times C}$, the output representation of the 4^{th} transformer encoder layer of the structure encoder. The concatenated representation is feed into another 6 transformer encoder layers to obtain the fused representation $\boldsymbol{z}_o \in \mathbb{R}^{N_v \times (C+C_t)}$.

Rectification Estimation. The obtained $\boldsymbol{z}_o$ is feed into a learnable module to perform upsampling and predict high-resolution rectification estimation. Specifically, we first predict a coarse resolution displacement map $\hat{\boldsymbol{f}}_o \in \mathbb{R}^{(\frac{H}{8} \times \frac{W}{8}) \times 2}$ through a two-layer convolutional network. Then, in analogy to [34], we upsample $\hat{\boldsymbol{f}}_o$ to full-resolution map $\hat{\boldsymbol{f}} \in \mathbb{R}^{H \times W \times 2}$ by taking learnable weighted combination of the 3 × 3 grid of the coarse resolution neighbors of each pixel.

3.4 Training Objectives

During training, except the preprocessing module, the architecture of the proposed DocGeoNet is end-to-end optimized with the following objective as follows,

$$\mathcal{L} = \alpha \mathcal{L}_{\text{3D}} + \beta \mathcal{L}_{\text{text}} + \mathcal{L}_{\text{flow}}, \tag{3}$$

where $\mathcal{L}_{\text{3D}}$ denotes the regression loss on 3D coordinate map, $\mathcal{L}_{\text{text}}$ represents the segmentation loss of textlines, and $\mathcal{L}_{\text{flow}}$ denotes the regression loss on warping flow. α and β are the weights associated to $\mathcal{L}_{\text{3D}}$ and $\mathcal{L}_{\text{text}}$, respectively. In the following, we present the formulation of the three loss terms.

Specifically, for the learning of the 3D coordinate map, the loss $\mathcal{L}_{\text{3D}}$ is calculated as L_1 distance between the predicted 3D coordinate map $\hat{\boldsymbol{C}}$ and its corresponding ground truth $\boldsymbol{C}_{gt}$ as follows,

$$\mathcal{L}_{3D} = \left\| \boldsymbol{C}_{gt} - \hat{\boldsymbol{C}} \right\|_1 . \tag{4}$$

The segmentation loss $\mathcal{L}_{\text{text}}$ for textlines is defined as a binary cross-entropy loss [8] as follows,

$$\mathcal{L}_{text} = -\sum_{i=1}^{N_d} \left[y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i) \right], \tag{5}$$

where N_d is the pixel number of the foreground document, y_i and $\hat{p}_i$ denote the ground-truth and predicted confidence, respectively. Note that here we only compute the loss on the foreground document region. One reason is that the textlines only exist in the foreground document region. The other one is that in the input background-excluded image $\boldsymbol{I}_S$, the textline pixels have similar RGB values to the background, which would confuse the network.

The loss $\mathcal{L}_{\text{flow}}$ for warping flow is defined as the L_1 distance between the predicted warping flow $\hat{\boldsymbol{f}}$ and its ground truth $\boldsymbol{f}_{gt}$ as follows,

$$\mathcal{L}_{flow} = \left\| \boldsymbol{f}_{gt} - \hat{\boldsymbol{f}} \right\|_1 . \tag{6}$$

4 DIR300 Dataset

In this section, we present the DIR300 dataset, a new dataset for document image rectification. In the following, we first revisit the previous datasets, and then elaborate the details of the introduced one.

4.1 Revisiting Existing Datasets

The most widely adopted datasets in the field are Doc3D dataset [6] and DocUNet Benchmark dataset [25]. Doc3D dataset [6] consists of 100k synthetic distorted document images generated with the real document data and rendering software [3]. It is used for training the rectification model. For each distorted document image, there are corresponding ground truth 3D coordinate map, depth map, and warping flow. However, it does not contain the textline annotations, which we empirically demonstrate to be beneficial for rectification.

DocUNet Benchmark dataset [25] is introduced for only evaluation purpose. It contains 130 document photos captured on 65 documents, which is too small to make the evaluation results convincing. Besides, to the best of our knowledge, this is the only publicly available benchmark dataset with real image data. Thus, the introduction of a larger scale benchmark becomes an urgent demand.

Additionally, the 127^{th} and 128^{th} distorted images in DocUNet Benchmark dataset [25] are rotated by 180 degrees, which do not match the ground truth documents [10]. It is ignored by existing works [6,7,9,14,22,25,41,42,43]. In our experiments, we report the results on the corrected dataset. For clarity, we also report the performance with two mistaken samples.

4.2 Dataset Details

We make two-fold efforts to build the DIR300 dataset. On the one hand, we extend the synthesized Doc3D dataset [6] with textline annotations to build the training set. On the other hand, we capture 300 real document samples to build a larger test set against the DocUNet Benchmark dataset [25].

Training Set. Here, we describe how to generate the textline annotations on the Doc3D dataset [6] with fewer labour requirements. Typically, it is difficult to localize the textlines in a distorted document image, where the textlines take various shapes. But it is easy to achieve it in a flattened document image. Hence, we rectify all distorted images in Doc3D dataset [6] using the ground truth warping flow and then detect the horizontal textlines by the following steps.

[3] https://www.blender.org/

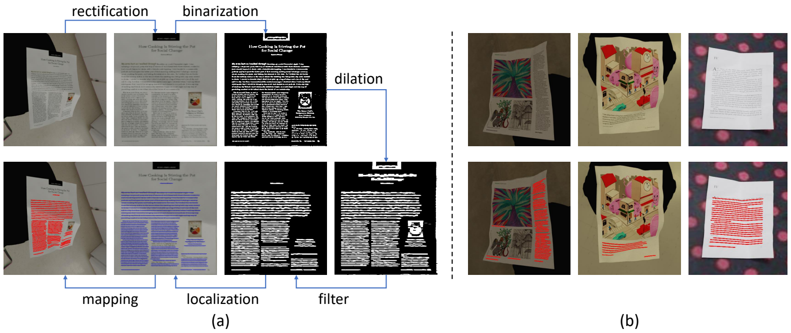

Fig. 2. An illustration of (a) the textlines annotation process, and (b) the visualization of textline annotations in corresponding distorted document images.

Specifically, as shown in Fig. 2 (a), we first convert the rectified images to gray-scale and perform adaptive *binarization* based on the local Gaussian weighted sum. Next, we conduct the horizontal *dilation* in the binary image to get the connected regions and their corresponding bounding boxes. Then, we set the thresholds on the shape of bounding boxes to *filter* out non-textline connected regions. Finally, we *localize* the center and the horizontal length of the bounding boxes to generate the horizontal textlines. After *mapping* such horizontal textlines to the original distorted image using the warping flow, we obtain the curved textlines annotations. As shown in Fig. 2 (b), we visualize some textline annotation samples, where the most textlines are annotated accurately. Notably, we note that a few annotated textlines are missed when being filtered due to their size, but they are within the fault tolerance of the network.

Test Set. We build the test set of DIR300 dataset with photos captured by mobile cameras. It contains 300 real document photos from 300 documents. Compared to the DocUNet Benchmark dataset [25], the distorted document images in DIR300 involve more complex background and various illumination conditions. Besides, we also increase the deformation degree of the warped documents. The creation details are provided in the supplemental material. To the best of our knowledge, the DIR300 test set is currently the largest real data benchmark for evaluating document image rectification.

5 Experiments

5.1 Evaluation Metrics

MS-SSIM. The Structural SIMilarity (SSIM) [38] measures how similar within each patch between two images. To balance the detail perceivability diversity that depends on the sampling density, Multi-Scale Structural Similarity (MS-SSIM) [39] calculates the weighted summation of SSIM [38] across multiple scales. Following [6,7,9,14,22,25,41,42], all rectified and ground truth images are resized to a 598,400-pixel area. Then, we build a 5-level-pyramid for MS-SSIM and the weight for each level is set as 0.0448, 0.2856, 0.3001, 0.2363, and 0.1333.

Local Distortion. By computing a dense SIFT-flow [21], Local Distortion (LD) [45] matches all the pixels from the ground truth scanned image to the rectified image. Then, LD is calculated as the mean value of the L_2 distance between the matched pixels, which measures the average local deformation of the rectified image. For a fair comparison, we resize all the rectified images and the ground truth images to a 598,400-pixel area, as suggested in [6,7,9,14,22,25,41,42].

ED and CER. Edit Distance (ED) [18] and Character Error Rate (CER) [29] quantify the similarity of two strings. ED is the minimum number of operations required to transform one string into the reference string. The involved operations include deletions (d), insertions (i) and substitutions (s). Then, Character Error Rate (CER) can be computed as follows,

$$CER = (d + i + s)/N_c, \tag{7}$$

where N_c is the character number of the reference string. CER represents the percentage of characters in the reference text that was incorrectly recognized in the distorted image. The lower the CER value (with 0 being a perfect score), the better the performance of the rectification quality.

We use Tesseract (v5.0.1) [33] as the OCR engine to recognize the text in the images. Following DewarpNet [6] and DocTr [9], we select 50 and 60 images from the DocUNet Benchmark dataset [25], respectively. Besides, on the DIR300 test set, we select 90 images. In such images, the text makes up the majority of content. Since if the text is sparse in a document, the character number N_c (numerator) in Eq. (7) is a small number, leading to a large variance for CER.

5.2 Implementation Details

We implement the whole framework of DocGeoNet in Pytorch [30]. The preprocessing module and the following rectification module are trained independently on the extended Doc3D dataset [6]. We detail their training in the following.

Preprocessing Module. During training, to generalize well to real data with complex background environments, we randomly replace the background of the distorted document with the texture images from Describable Texture Dataset [5]. We use Adam optimizer [15] with a batch size of 32. The initial learning rate is set as 1×10^{-4}, and reduced by a factor of 0.1 after 30 epochs. The network is trained for 45 epochs on two NVIDIA RTX 2080 Ti GPUs. In addition, the threshold τ for binarizing the confidence map is set as 0.5.

Rectification Module. During training, we remove the background of distorted document images using the ground truth masks of the foreground document. To generalize well to real data with various illumination conditions, we add jitter in HSV color space to magnify illumination and document color variations. We use AdamW optimizer [23] with a batch size of 12 and an initial learning rate of 1×10^{-4}. Our model is trained for 40 epochs on 4 NVIDIA GTX 1080 Ti GPUs. We set the hyperparameters $\alpha = 0.2$ and $\beta = 0.2$ (in Eq. (3)).

Table 1. Quantitative comparisons of the existing learning-based methods in terms of image similarity, distortion metrics, OCR accuracy, and running efficiency on the **corrected** DocUNet Benchmark dataset [25]. "*" denotes that the OCR metrics could not be calculated as the rectified images or models are not publicly available. "↑" indicates the higher the better, while "↓" means the opposite.

Methods	Venue	MS-SSIM ↑	LD ↓	ED ↓	CER ↓	FPS ↑	Para.
Distorted	-	0.2459	20.51	2111.56/1552.22	0.5352/0.5089	-	-
DocUNet [25]	*CVPR'18*	0.4103	14.19	1933.66/1259.83	0.4632/0.3966	0.21	58.6M
AGUN [22]*	*PR'18*	-	-	-	-	-	-
DocProj [19]	*TOG'19*	0.2946	18.01	1712.48/1165.93	0.4267/0.3818	0.11	47.8M
FCN-based [42]	*DAS'20*	0.4477	7.84	1792.60/1031.40	0.4213/0.3156	1.49	23.6M
DewarpNet [6]	*ICCV'19*	0.4735	8.39	885.90/525.45	0.2373/0.2102	7.14	86.9M
PWUNet [7]	*ICCV'21*	0.4915	8.64	1069.28/743.32	0.2677/0.2623	-	-
DocTr [9]	*MM'21*	0.5105	7.76	724.84/464.83	0.1832/0.1746	7.40	26.9M
DDCP [41]	*ICDAR'21*	0.4729	8.99	1442.84/745.35	0.3633/0.2626	**12.38**	**13.3M**
FDRNet [43]	*CVPR'22*	**0.5420**	8.21	794.54/514.90	0.2010/0.1846	-	-
RDGR [14]	*CVPR'22*	0.4968	8.51	693.38/420.25	**0.1654**/0.1559	-	-
Ours	-	0.5040	**7.71**	**692.86/379.00**	0.1797/**0.1509**	-	24.8M

Table 2. Quantitative comparisons of the existing learning-based methods in terms of image similarity, distortion metrics, OCR accuracy on the proposed DIR300 test set. "↑" indicates the higher the better, while "↓" means the opposite.

Methods	Venue	MS-SSIM ↑	LD ↓	ED ↓	CER ↓
Distorted	-	0.3148	39.98	1512.16	0.5234
DocProj [19]	*TOG'19*	0.3213	31.16	1049.36	0.3984
DewarpNet [6]	*ICCV'19*	0.4882	14.48	1096.31	0.3626
DocTr [9]	*MM'21*	0.6104	7.84	741.93	0.2320
DDCP [41]	*ICDAR'21*	0.5484	11.44	2122.44	0.5476
Ours	-	**0.6323**	**7.07**	**706.99**	**0.2271**

5.3 Experiment Results

We evaluate the proposed DocGeoNet by quantitative and qualitative comparisons with recent state-of-the-art rectification methods. For quantitative evaluation, we show the comparison on distortion metrics, OCR accuracy, image similarity, and inference efficiency. The evaluations are conducted on the corrected DocUNet Benchmark dataset [25] and our proposed DIR300 test set. For clarity, in the supplementary material, we also report the performance on the DocUNet Benchmark dataset [25] with two mistaken samples described in Sec. 4.1.

Quantitative Comparisons. On the DocUNet Benchmark [25], we compare DocGeoNet with existing learning-based methods. As shown in Table 1, our DocGeoNet achieves a Local Distortion (LD) of 7.71 and a Character Error Rate (CER) of 15%, surpassing previous state-of-the-art methods DocTr [9] and RDGR [14]. In addition, we compare the parameter counts and inference time of processing a 1080P resolution image. The test is conducted on an RTX 2080Ti GPU. As shown in Table 1, the proposed DocGeoNet shows promising efficiency.

On the proposed DIR300 test set, we compare DocGeoNet with typical rectification methods with model publicly available. As shown in Table 2, DocGeoNet outperforms the previous state-of-the-art method DocTr [9] on Local Distortion (LD) and OCR metrics, verifying its superior rectification ability.

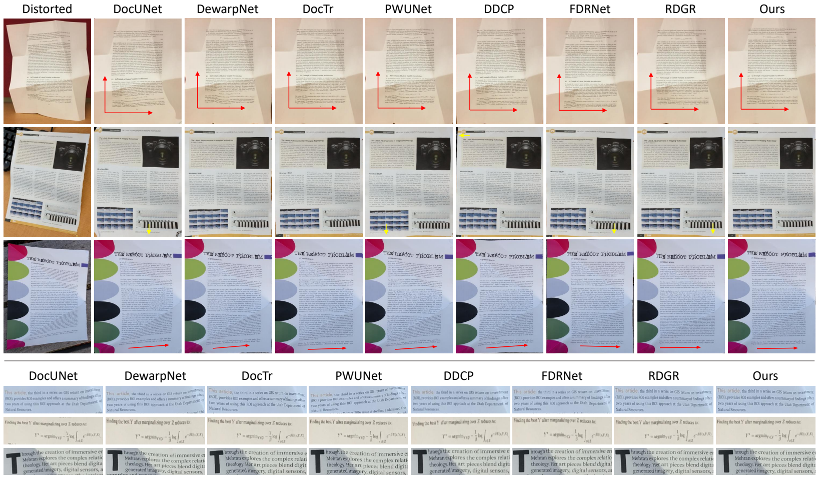

Fig. 3. Qualitative comparisons with previous methods on the DocUNet Benchmark dataset [25] in terms of the rectified images and local textline detail. For the comparisons of the rectified images, we highlight the comparisons of boundary and textlines by the yellow and red arrow, respectively.

Table 3. Ablations of the architecture settings. SE denotes the structure encoder. TE denotes the textline extracter. Here we only supervise the output warping flow.

SE	TE	MS-SSIM ↑	LD ↓	ED ↓	CER ↓
		0.4972	8.11	869.00/545.83	0.2207/0.1997
✓	✓	**0.4994**	**7.99**	**764.38/524.90**	**0.2010/0.1803**

Qualitative Comparisons. The qualitative comparisons are conducted on the DocUNet Benchmark [25] and DIR300 test set. To compare the local rectified detail, we also show the comparisons of cropped local rectified text. As shown in Fig. 3 and Fig. 4, the proposed DocGeoNet shows superior rectification quality. Specifically, for our method, the incomplete boundaries phenomenons existing in the previous methods are to a certain extent relieved. Besides, the rectified textlines of our method are much straighter than previous methods. More results on the both datasets are provided in the supplementary material.

5.4 Ablation Study

We conduct ablation study to verify the effectiveness of the proposed DocGeoNet, including the architecture and the representations to learn. The ablations are conducted on the DocUNet Benchmark dataset [25].

Architecture Setting. We first train a simple network without the structure encoder and textline extractor: the background-excluded image $\boldsymbol{I}_S$ is forward to a convolutional module and its flattened feature $\boldsymbol{z}_v$ is fed into the rectification decoder. Then, we add the structure encoder and textline extractor while their

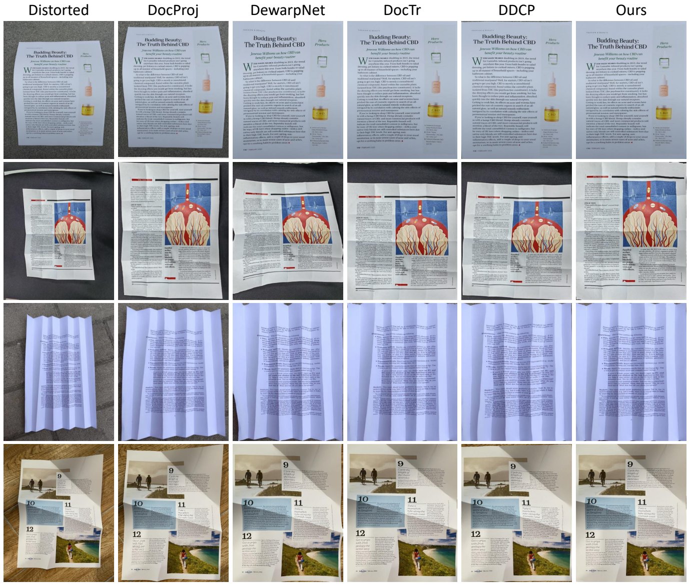

Fig. 4. Qualitative comparisons with previous methods on the proposed DIR300 test set in terms of the rectified images.

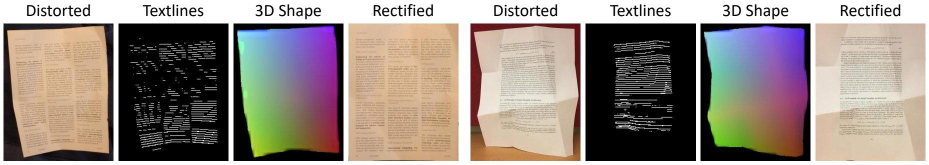

Fig. 5. Visualization of two instances on the predicted textlines and 3D coordinate map by our DocGeoNet.

supervisions are not deployed. As shown in Table 3, the latter model obtains a slight improvement. It is used as the baseline model for following study.

Geometric Representation. Based on the baseline network, we add the supervision on the structure encoder and textline extractor, respectively. As shown in Table 4, both the representations promote the learning of rectification. Furthermore, the performance is better after both the supervisions are deployed. To provide a more specific view of the predicted 3D coordinate and textlines, we showcase two examples in Fig. 5. As shown in Figure 6, we visualize the shape feature $\boldsymbol{Z}_c$ and textline feature $\boldsymbol{Z}_t$ to help understand our primary motivation. As we can see, shape feature focuses more on the page boundaries and depresses the inside text content, while textline feature does the opposite. The above results reveal the effectiveness of representation learning of the document attributes that bridge the distorted image and the rectified image.

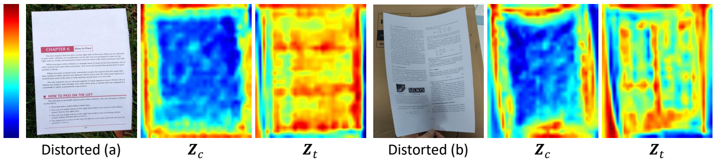

Fig. 6. Comparison of shape feature $\boldsymbol{Z}_c$ and textline feature $\boldsymbol{Z}_t$.

Table 4. Ablations of the different representation learning settings of DocGeoNet.

3D Shape	Textlines	MS-SSIM ↑	LD ↓	ED ↓	CER ↓
		0.4994	7.99	764.38/524.90	0.2010/0.1803
✓		**0.5067**	7.83	737.56/418.53	0.1826/0.1646
	✓	0.5053	7.79	748.35/466.58	0.1873/0.1693
✓	✓	0.5040	**7.71**	**692.86/379.00**	**0.1797/0.1509**

Table 5. Ablations of the different component settings of DocGeoNet.

	MS-SSIM ↑	LD ↓	ED ↓	CER ↓
Full model	0.5040	**7.71**	**692.86/379.00**	0.1797/**0.1509**
Preprocessing → None	0.4843	8.61	786.54/514.60	0.2100/0.2003
Upsampling: Learnable → Bilinear	**0.5062**	7.77	702.83/405.21	**0.1790**/0.1523

Structure Modifications. Finally, we discuss some components of our DocGeoNet. The results are shown in Table 5. (1) We first verify the preprocessing module that is adopted in DocGeoNet and the recent state-of-the-art method, and train a network without it. The results show that the performance slightly drops, which suggests that taking the whole distorted image as input burdens the network with localizing the foreground document besides the rectification prediction. (2) We compare the bilinear upsampling with our learnable upsampling module. The performance is slightly better using the learnable upsampling module. We attribute this improvement to that the coarse bilinear upsampling operation is difficult to recover the small deformations.

6 Conclusion

In this work, we present a novel deep network DocGeoNet for document image rectification. It bridges the distorted and rectified image by explicitly introducing the representation learning of the geometric constraints from two document attributes, *i.e.*, 3D shape and textlines. Extensive experiments are conducted, and the results reveal that our DocGeoNet achieves state-of-the-art performance on the prevalent benchmark and proposed large-scale challenge benchmark.

Acknowledgments: This work was supported by the National Natural Science Foundation of China under Contract 61836011 and 62021001. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

References

1. Brown, M.S., Seales, W.B.: Document restoration using 3D shape: a general deskewing algorithm for arbitrarily warped documents. In: Proceedings of the IEEE International Conference on Computer Vision. vol. 2, pp. 367–374 (2001)
2. Brown, M.S., Seales, W.B.: Image restoration of arbitrarily warped documents. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(10), 1295–1306 (2004)
3. Brown, M.S., Sun, M., Yang, R., Yun, L., Seales, W.B.: Restoring 2d content from distorted documents. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(11), 1904–1916 (2007)
4. Chew Lim Tan, Li Zhang, Zheng Zhang, Tao Xia: Restoring warped document images through 3D shape modeling. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(2), 195–208 (2006)
5. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3606–3613 (2014)
6. Das, S., Ma, K., Shu, Z., Samaras, D., Shilkrot, R.: DewarpNet: Single-image document unwarping with stacked 3D and 2D regression networks. In: Proceedings of the International Conference on Computer Vision. pp. 131–140 (2019)
7. Das, S., Singh, K.Y., Wu, J., Bas, E., Mahadevan, V., Bhotika, R., Samaras, D.: End-to-end piece-wise unwarping of document images. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4268–4277 (2021)
8. De Boer, P.T., Kroese, D.P., Mannor, S., Rubinstein, R.Y.: A tutorial on the cross-entropy method. Annals of Operations Research **134**(1), 19–67 (2005)
9. Feng, H., Wang, Y., Zhou, W., Deng, J., Li, H.: DocTr: Document image transformer for geometric unwarping and illumination correction. In: Proceedings of the ACM International Conference on Multimedia. pp. 273–281 (2021)
10. Feng, H., Zhou, W., Deng, J., Tian, Q., Li, H.: DocScanner: Robust document image rectification with progressive learning. arXiv preprint arXiv:2110.14968 (2021)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
12. He, Y., Pan, P., Xie, S., Sun, J., Naoi, S.: A book dewarping system by boundary-based 3D surface reconstruction. In: Proceedings of the International Conference on Document Analysis and Recognition. pp. 403–407 (2013)
13. Huaigu Cao, Xiaoqing Ding, Changsong Liu: Rectifying the bound document image captured by the camera: a model based approach. In: Proceedings of the International Conference on Document Analysis and Recognition. vol. 1, pp. 71–75 (2003)
14. Jiang, X., Long, R., Xue, N., Yang, Z., Yao, C., Xia, G.S.: Revisiting document image dewarping by grid regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4543–4552 (2022)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2015)
16. Koo, H.I., Kim, J., Cho, N.I.: Composition of a dewarped and enhanced document image from two view images. IEEE Transactions on Image Processing **18**(7), 1551–1562 (2009)
17. Lavialle, O., Molines, X., Angella, F., Baylou, P.: Active contours network to straighten distorted text lines. In: Proceedings of the International Conference on Image Processing. vol. 3, pp. 748–751 (2001)

18. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals **10**, 707–710 (1966)
19. Li, X., Zhang, B., Liao, J., Sander, P.V.: Document rectification and illumination correction using a patch-based cnn. ACM Transactions on Graphics **38**(6), 1–11 (2019)
20. Liang, J., DeMenthon, D., Doermann, D.: Geometric rectification of camera-captured document images. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(4), 591–605 (2008)
21. Liu, C., Yuen, J., Torralba, A.: SIFT flow: Dense correspondence across scenes and its applications. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(5), 978–994 (2011)
22. Liu, X., Meng, G., Fan, B., Xiang, S., Pan, C.: Geometric rectification of document images using adversarial gated unwarping network. Pattern Recognition **108**, 107576 (2020)
23. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: Proceedings of the International Conference on Learning Representations (2019)
24. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
25. Ma, K., Shu, Z., Bai, X., Wang, J., Samaras, D.: DocUNet: Document image unwarping via a stacked u-net. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4700–4709 (2018)
26. Markovitz, A., Lavi, I., Perel, O., Mazor, S., Litman, R.: Can you read me now? content aware rectification using angle supervision. In: Proceedings of the European Conference on Computer Vision. pp. 208–223. Springer (2020)
27. Meng, G., Pan, C., Xiang, S., Duan, J., Zheng, N.: Metric rectification of curved document images. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(4), 707–722 (2011)
28. Meng, G., Wang, Y., Qu, S., Xiang, S., Pan, C.: Active flattening of curved document images via two structured beams. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3890–3897 (2014)
29. Morris, A.C., Maier, V., Green, P.: From wer and ril to mer and wil: improved evaluation measures for connected speech recognition. In: Proceedings of the International Conference on Spoken Language Processing (2004)
30. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
31. Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O.R., Jagersand, M.: U2-Net: Going deeper with nested u-structure for salient object detection. Pattern Recognition (2020)
32. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-assisted Intervention. pp. 234–241. Springer (2015)
33. Smith, R.: An overview of the tesseract OCR engine. In: Proceedings of the International Conference on Document Analysis and Recognition. vol. 2, pp. 629–633 (2007)
34. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: European Conference on Computer Vision. pp. 402–419. Springer (2020)
35. Tsoi, Y.C., Brown, M.S.: Multi-view document rectification using boundary. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8 (2007)

36. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Proceedings of the Neural Information Processing Systems. pp. 6000–6010 (2017)
37. Wada, T., Ukida, H., Matsuyama, T.: Shape from shading with interreflections under a proximal light source: Distortion-free copying of an unfolded book. International Journal of Computer Vision **24**(2), 125–135 (1997)
38. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
39. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: Proceedings of the Asilomar Conference on Signals, Systems Computers. vol. 2, pp. 1398–1402 (2003)
40. Wu, C., Agam, G.: Document image de-warping for text/graphics recognition. In: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition. pp. 348–357. Springer (2002)
41. Xie, G.W., Yin, F., Zhang, X.Y., Liu, C.L.: Document dewarping with control points. In: International Conference on Document Analysis and Recognition. pp. 466–480. Springer (2021)
42. Xie, G., Yin, F., Zhang, X., Liu, C.: Dewarping document image by displacement flow estimation with fully convolutional network. In: International Workshop on Document Analysis Systems. pp. 131–144. Springer (2020)
43. Xue, C., Tian, Z., Zhan, F., Lu, S., Bai, S.: Fourier document restoration for robust document dewarping and recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4573–4582 (2022)
44. Yamashita, A., Kawarago, A., Kaneko, T., Miura, K.T.: Shape reconstruction and image restoration for non-flat surfaces of documents with a stereo vision system. In: Proceedings of the International Conference on Pattern Recognition. vol. 1, pp. 482–485 (2004)
45. You, S., Matsushita, Y., Sinha, S., Bou, Y., Ikeuchi, K.: Multiview rectification of folded documents. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(2), 505–511 (2018)
46. Zhang, L., Zhang, Y., Tan, C.: An improved physically-based method for geometric restoration of distorted document images. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(4), 728–734 (2008)