

DocTr: Document Image Transformer for Geometric Unwarping and Illumination Correction

Hao Feng^{1*}, Yuechen Wang^{1*}, Wengang Zhou^{1,2†}, Jiajun Deng¹, Houqiang Li^{1,2†}

¹CAS Key Laboratory of Technology in GIPAS, EEIS Department,

University of Science and Technology of China

²Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

haof@mail.ustc.edu.cn, wyc9725@mail.ustc.edu.cn, zhwg@ustc.edu.cn, dengjj@mail.ustc.edu.cn, lihq@ustc.edu.cn

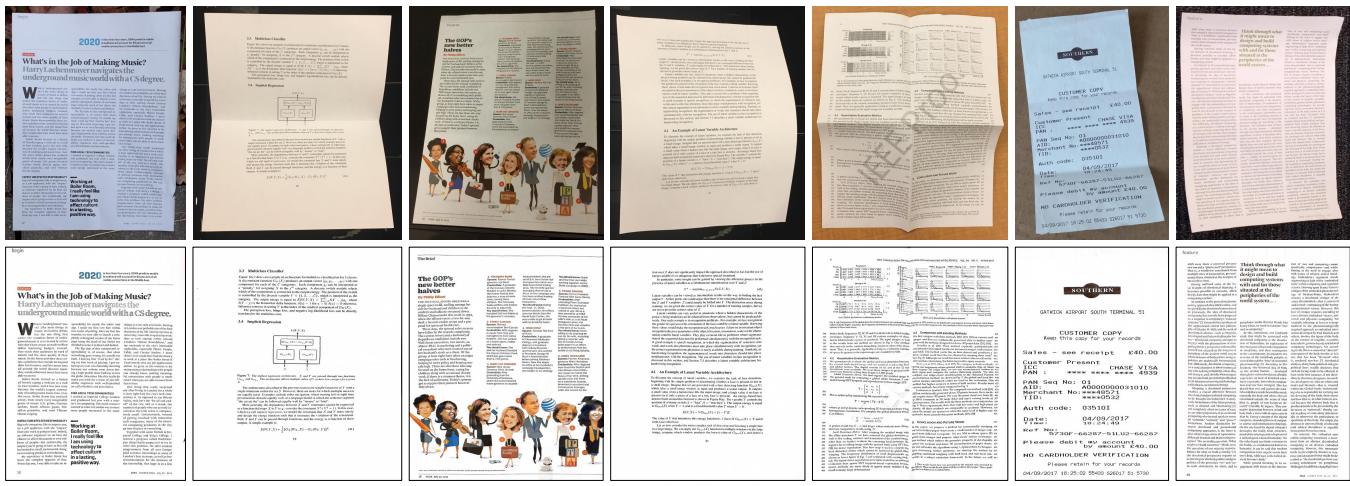


Figure 1: Qualitative rectified results of our Document Image Transformer (DocTr). The top row shows the distorted document images. The second row shows the rectified results after geometric unwarping and illumination correction.

ABSTRACT

In this work, we propose a new framework, called Document Image Transformer (DocTr), to address the issue of geometry and illumination distortion of the document images. Specifically, DocTr consists of a geometric unwarping transformer and an illumination correction transformer. By setting a set of learned query embedding, the geometric unwarping transformer captures the global context of the document image by self-attention mechanism and decodes the pixel-wise displacement solution to correct the geometric distortion. After geometric unwarping, our illumination correction transformer further removes the shading artifacts to improve the visual quality and OCR accuracy. Extensive evaluations are conducted on several datasets, and superior results are reported against the

*The first two authors contribute equally to this work.

†Corresponding authors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '21, October 20–24, 2021, Virtual Event, China

© 2021 Association for Computing Machinery.

ACM ISBN 978-1-4503-8651-7/21/10...\$15.00

<https://doi.org/10.1145/3474085.3475388>

state-of-the-art methods. Remarkably, our DocTr achieves 20.02% Character Error Rate (CER), a 15% absolute improvement over the state-of-the-art methods. Moreover, it also shows high efficiency on running time and parameter count. Our code and results are available at <https://github.com/fh2019ustc/DocTr>.

CCS CONCEPTS

- Theory of computation → Scheduling algorithms; • Computing methodologies → Reconstruction.

KEYWORDS

Transformer, Document Unwarping, Illumination Correction, OCR

ACM Reference Format:

Hao Feng, Yuechen Wang, Wengang Zhou, Jiajun Deng, Houqiang Li. 2021. DocTr: Document Image Transformer for Geometric Unwarping and Illumination Correction. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3474085.3475388>

1 INTRODUCTION

In our daily life, we have to digitize physical documents from time to time. With the advance of smartphone and thanks to its convenience, many people turn to take photos of documents with the camera of smartphones. However, such captured images often suffer from uncontrolled geometry and illumination distortion due to

various physical deformation of the paper (*i.e.*, folded, curved, or crumpled), camera positions, and uneven illumination conditions. Therefore, they often show bad visual quality and readability. What is more, such distorted document images also cast a negative influence on downstream multimedia applications, such as automatic text recognition, multi-modal retrieval, content analysis, editing, and preservation. This leads to a growing demand for an efficient approach to correct such geometry and illumination distortions.

Early works on geometric unwarping for document images resort to 3D reconstruction and often rely on auxiliary hardware [1, 24, 44] or multi-view images [15, 43] to recover the 3D shape of the paper sheet. However, the involvement of costly hardware or extra shooting requirement unavoidably limits the applicability. Some other methods assume a parametric model on the document surface and optimize the model by extracting specific features such as shading [5], boundaries [11], or text lines [16, 40]. Besides the non-trivial cost of the optimization process, their problem formulation is oversimplified that leads to sub-optimal performance.

Recently, deep learning has been introduced to address the issue of geometry and illumination distortion, and shown as an alternative to traditional approaches. For geometric unwarping, deep learning methods formulate the task as a pixel-wise displacement regression problem. However, current methods commonly ignore the fact that the physical deformation of the paper is an integral whole in which the parts are interrelated. Specifically, Ma *et al.* [22] directly regress a dense 2D coordinate mapping field with a stacked UNet [30]. Das *et al.* [7] explicitly model the 3D shape of the paper sheet with a U-structure network. These methods adopt the practice of stacking convolutional modules to perform the regression, while CNN is incapable to capture the long-range relationship to model the deformation of paper. As a result, the rectified documents still exist curved text regions, which carry some important information of the documents. Additionally, Li *et al.* [18] propose to unwarped a distorted document by stitching the unwarped patches. However, it is difficult and ambiguous for a network to understand the original shape of a figure when it is incomplete in a patch image. Hence, it is insufficient to recover the geometry distortion.

For illumination correction, Das *et al.* [7] regress a low-resolution shading map, which is upsampled to the original size of the high-resolution document. The document image is corrected by matrix multiplication with the high-resolution shading map. However, such an operation will cause blur and artifacts due to the upsampling process. Recently, Li *et al.* [18] stack several residual blocks to directly regress the corrected document patches which are then stitched. However, simply stacking the convolutional layers can not capture the global context that is important to model the illumination variation, as reflected by the regions of shadow remaining in the corrected document images.

To address the aforementioned problems, we propose a new framework called Document Image Transformer (DocTr) to simultaneously address the issue of geometry and illumination distortions in document images. Inspired by recent success of Transformer [35] in computer vision [2, 3, 10], we integrate the transformer structure into the document image rectification pipeline. Specifically, DocTr consists of a Geometric Unwarping Transformer and an illumination Correction Transformer, both of which are the transformer encoder-decoder architecture with task-specific heads and tails.

Different from the standard transformer, the input of the transformer encoder and decoder are flattened 2D image features and a task-specific learnable embedding, respectively. Given a distorted document image, the geometric unwarping transformer first encodes the features of images and decodes the coordinate mapping field for unwarping. By applying the predicted coordinate mapping to the original image, we can obtain the unwarped image. Then, we crop the unwarped image into patches with a predefined overlap. The patch images are fed into the illumination correction transformer for illumination rectification. Finally, the high-resolution document image can be obtained by stitching the corrected patches.

The main contribution of this paper is the proposal of DocTr, which is the first transformer-based framework for document image rectification. Thanks to the attention mechanism of transformer, DocTr is able to capture global information for pixel-level geometry and illumination distortion. Extensive experiments on several datasets, *i.e.*, Doc3D, DRIC, and DocUNet dataset, demonstrate the effectiveness and superiority of our DocTr over the existing state-of-the-art methods on both tasks. Notably, on DocUNet benchmark [22], we achieve significant improvement on OCR results (absolutely 15.32% Character Error Rate (CER) reduced compared to the state-of-the-art method [7]). Furthermore, our method shows high efficiency on inference time and parameter count.

2 RELATED WORK

2.1 Geometric Unwarping

Rectification by 3D shape reconstruction. Some methods utilize auxiliary hardware to reconstruct the 3D shape of the deformed paper sheet. Brown *et al.* [1] acquire the 3D representation of shape using a structured light 3D acquisition system. Based on physical modeling technique, Zhang *et al.* [44] use a laser range scanner to perform the modeling. Some other methods utilize multiview images or single image for 3D shape reconstruction. By using multiview images, You *et al.* [43] propose a ridge-aware 3D reconstruction method, and Tan *et al.* [5] utilize the shape from shading technique for geometric unwarping. Cao *et al.* [12] assume a general cylindrical surface on the surface of document. Das *et al.* [7] explicitly model the 3D shape with a convolutional network.

Rectification from low-level features. The low-level features from a single image also show a useful clue for correcting the geometric distortion. In early works, many algorithms aim to correct the curved text lines to be horizontal and straight. The detected text lines are modeled as cubic B-splines by Lavialle *et al.* [16], nonlinear curve by Wu and Agam [40], polynomial approximation by Mischke and Luther [25]. Ma *et al.* [22] first introduce the neural network to the task and directly regress the pixel-wise displacement with a stacked UNet [30]. Li *et al.* [18] propose to perform the unwarping by stitching the displacement field of the image patches. Xie *et al.* [41] estimate pixel-wise displacement using a fully convolutional network with a smooth constraint. Amir *et al.* [23] localize the words and their orientation in the networks.

2.2 Illumination Correction

By feeding a low-resolution document image to a stacked UNet [30], Das *et al.* [7] regress a low-resolution shadow map, typically 256 ×

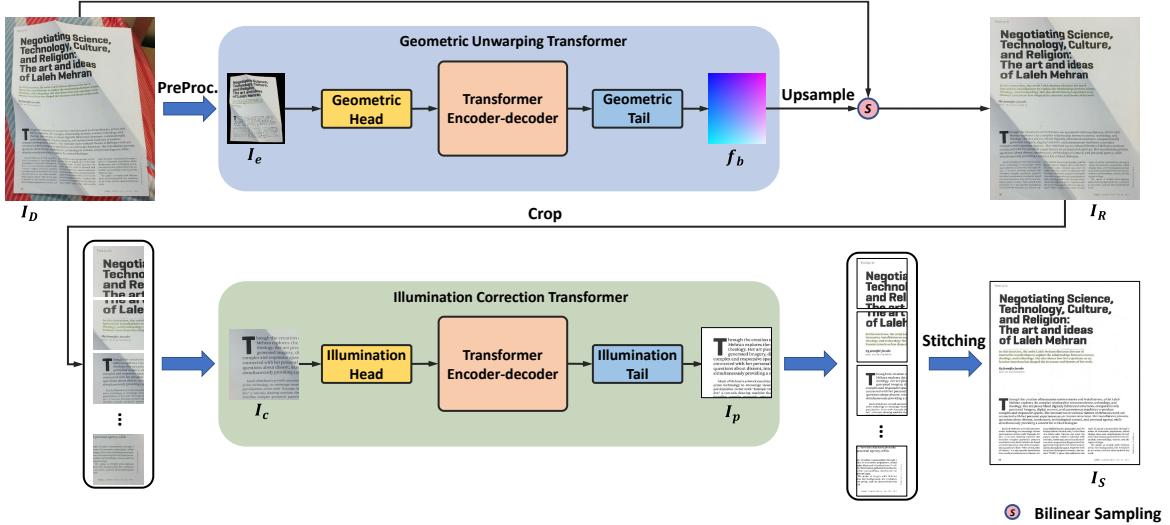


Figure 2: An overview of Document Image Transformer (DocTr). It consists of two main components: a geometric unwarping transformer and an illumination correction transformer.

256. Then, the shadow map is upsampled to the shape of the original high-resolution document and conducts matrix multiplication with the document image to output the corrected document. This approach is limited by the size of the shadow map as the upsampling process will cause blur. Differently, Li *et al.* [18] stack several residual blocks to regress the corrected document patches which are then stitched. However, simply stacking the convolutional layers is hard to capture the global context to model the illumination variation, causing the remaining shadow regions on the stitched patches. Therefore, we propose a transformer-based network to capture the global information as well as retain local features.

2.3 Transformer in Language and Vision

Since it is first proposed by Vaswani *et al.* [35] for machine translation, transformer has become a prevailing architecture in NLP. The basic block of transformer is the multi-head attention module, which aggregates information from the whole input in both transformer encoder and decoder module. Transformer demonstrates superior performance in language model pretraining methods [9, 29, 42], and achieves competitive performance on diverse NLP problems. Recently, transformer has been introduced to various computer vision tasks, such as image classification [4], image generation [26], object detection [2], semantic segmentation [36], tracking [37], etc. Comparing to CNN, the attention mechanism learns more global dependencies, therefore, transformer also shows great performance in low-level tasks [3]. Transformer has also been proved effectiveness in multi-modal area, including multi-modal representations [45] and applications [13, 19, 31]. Inspired by the extensive applications of transformer, we integrate the transformer encoder-decoder into the document image rectification problem.

3 APPROACH

In this section, we propose a novel framework called DocTr for document unwarping and illumination correction. As shown in

Figure 2, DocTr consists of a geometric unwarping transformer and an illumination correction transformer.

3.1 Geometric Unwarping Transformer

For the geometric unwarping problem, a photographed distorted document image with arbitrary resolution is given as input, the goal is to predict pixel-wise displacement for unwarping. Specifically, as shown in Figure 2, given an image $I_D \in \mathbb{R}^{H \times W \times 3}$, we first downsample it and get the image $I_d \in \mathbb{R}^{H_0 \times W_0 \times C_0}$, where $H_0 = W_0 = 288$ in our method and $C_0 = 3$ is the number of RGB channels. Then, I_d is fed into the preprocessing module to get the background-excluded image I_e which is injected into geometric unwarping transformer to predict a backward mapping field $f_b = (f_b^u, f_b^v)$, where $f_b^u, f_b^v \in \mathbb{R}^{H_0 \times W_0}$ denote the horizontal and vertical coordinate mapping map, respectively. Finally, we upsample f_b to the original size $H \times W$ of I_D and get $f_B \in \mathbb{R}^{H \times W \times 2}$. With f_B , the rectified document image $I_R \in \mathbb{R}^{H \times W \times 3}$ can be obtained by warping operation based on the bilinear interpolation as follows,

$$I_R(u_0, v_0) = I_D(f_B^u(u_0, v_0), f_B^v(u_0, v_0)), \quad (1)$$

where (u_0, v_0) is the pixel position. In the following, we elaborate the key components of our geometric unwarping transformer.

Preprocessing. Given a downsampled distorted document image $I_d \in \mathbb{R}^{H_0 \times W_0 \times 3}$, a light semantic segmentation network [28] is utilized to predict the confidence map of the foreground document, which is further binarized with a threshold τ to obtain the binary document region mask $M_{I_d} \in \mathbb{R}^{H_0 \times W_0}$. Then, the background of I_d can be removed by element-wise matrix multiplication with broadcasting along the channels of I_d . The preprocessing network is trained with a binary cross-entropy loss [8] as follows,

$$\mathcal{L}_{seg} = - \sum_{i=1}^{N_p} [y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i)], \quad (2)$$

where N_p is the number of the pixels, and y_i and \hat{p}_i denote the ground-truth and predicted confidence, respectively.

Head. Given the preprocessed background-excluded document image $I_e \in \mathbb{R}^{H_0 \times W_0 \times 3}$, features are then extracted from I_e using a convolutional module G_θ that consists of 6 residual blocks. G_θ downsamples the feature maps at $\frac{1}{2}$ resolution every two blocks and output features $f_g = G_\theta(I_e) \in \mathbb{R}^{\frac{H_0}{8} \times \frac{W_0}{8} \times c_g}$, where we set $c_g = 512$. To adapt to the sequence input form of the subsequent transformers, we flatten f_g into a sequence of 2D features, i.e., $f_s \in \mathbb{R}^{N_g \times c_g}$, where $N_g = \frac{H_0}{8} \times \frac{W_0}{8}$ is the number of patches.

Transformer Encoder. We use a transformer encoder [35] to encode the global relationship among patches. During the flattening operation, the flattened image feature f_s losses 2D position information, which is essential to learn local and global relationships. Also, since the transformer encoder is permutation-invariant, it is necessary to add position representation explicitly. Therefore, to maintain the position information in the process, we add learnable 2D position embedding $E_p \in \mathbb{R}^{N_g \times c_g}$ following [10], which is consistent to different input images. The resulting 2D position-aware feature $f_p = f_s + E_p$ is then feed into K transformer encoder layers. As shown in Figure 3, each of the encoder layers contains a multi-head self-attention module and a feed-forward network. The output representation can be calculated as follows,

$$\begin{aligned} F_0 &= [f_{pi}], i \in \{0, 1, \dots, N_g - 1\}, \\ Q_i &= W_e^Q F_{i-1}, K_i = W_e^K F_{i-1}, V_i = W_e^V F_{i-1}, \\ F'_i &= LN(MA(Q_i, K_i, V_i) + F_{i-1}), \\ F_i &= LN(FFN(F'_i) + F'_i), \end{aligned} \quad (3)$$

where $W_e^Q, W_e^K, W_e^V \in \mathbb{R}^{M \times c_g \times c_w}$, M is the number of attention heads, i denotes the i^{th} layer of the transformer encoder, $MA(\cdot)$, $FFN(\cdot)$, $LN(\cdot)$ denote the multi-head attention, feed-forward network, and layer normalization, respectively. F_i denotes the output feature of the i^{th} encoder layer.

Transformer Decoder. After obtaining the global-aware representations F_K , we adopt K transformer decoder layers to generate pixel-level predictions. Each of the decoder layers contains a multi-head self-attention module, an encoder-decoder multi-head attention modules, and a feed-forward network. The encoded feature F_K is used as attention value in the encoder-decoder attention computation in each decoder layer. Different from the original transformer decoder, which performs decoding in an iterative way, we decode the feature of each position in parallel. Therefore, a learnable embedding $E_d \in \mathbb{R}^{N_g \times c_g}$ as well as position embedding E_p are taken as the input of transformer decoder. Formally,

$$\begin{aligned} Y_0 &= [E_{pi} + E_{di}], i \in \{0, 1, \dots, N_g - 1\}, \\ Q_i &= W_d^Q Y_{i-1}, K_i = W_d^K Y_{i-1}, V_i = W_d^V Y_{i-1}, \\ Y'_i &= LN(MA(Q_i, K_i, V_i) + Y_{i-1}), \\ Q'_i &= W_c^Q Y'_i, K'_i = W_c^K F_K, V'_i = W_c^V F_K, \\ Y''_i &= LN(MA(Q'_i, K'_i, V'_i) + Y_{i-1}), \\ Y_i &= LN(FFN(Y''_i) + Y''_i), \end{aligned} \quad (4)$$

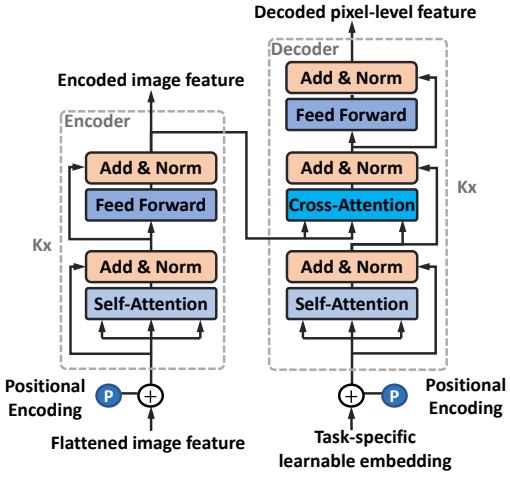


Figure 3: Architecture of the transformer encoder-decoder.

where $W_d^Q, W_d^K, W_d^V, W_c^Q, W_c^K, W_c^V \in \mathbb{R}^{M \times c_g \times c_w}$. The final output of the transformer decoder $Y_K \in \mathbb{R}^{N_g \times c_g}$ is reshaped to $f_d \in \mathbb{R}^{\frac{H_0}{8} \times \frac{W_0}{8} \times c_g}$ and then used to make pixel-level predictions.

Tail. As shown in Fig 4, we train a learnable module to perform upsampling on the decoded features f_d and obtain high-resolution predictions. Specifically, we first use a two-layer convolutional network to reduce the channel dimension c_g to 2, i.e., $f_o \in \mathbb{R}^{\frac{H_0}{8} \times \frac{W_0}{8} \times 2}$, where the two channels represent horizontal and vertical displacement, respectively. Then, an additional two-layer convolutional network predicts a weight mask $f_{m,ij} \in \mathbb{R}^{3 \times 3 \times 64}$ for each pixel (i, j) on f_d , where a 2-dimensional softmax is performed on the first two dimensions of $f_{m,ij}$. Finally, we calculate the convolution of each 3×3 patch on f_o and the corresponding pixel-wise weight mask. Formally,

$$\begin{aligned} f_{oh}^{i,j} &= f_o^{i-1:i+1, j-1:j+1} \otimes f_{m,ij} \\ &= \sum_{u=-1}^1 \sum_{v=-1}^1 f_o^{i-u, j-v} f_{m,ij}^{u+1, v+1}, \end{aligned} \quad (5)$$

where $f_{oh}^{i,j} \in \mathbb{R}^{2 \times 64}$. The high resolution 2-dimensional coordinate mapping map $f_b \in \mathbb{R}^{H_0 \times W_0 \times 2}$ is obtained by reshaping the last dimension of output f_{oh} and flattening into 8×8 patches.

Loss Function. The training loss for our geometric unwarping transformer is defined as the L_1 distance between the predicted backward mapping f_b and ground truth f_{gt} as follows,

$$L_{geo} = \|f_{gt} - f_b\|_1. \quad (6)$$

3.2 Illumination Correction Transformer

After geometric unwarping, the unwarped document image still suffers from sampling and shading artifacts due to the rectification process and lighting conditions. Hence, we further propose an illumination correction transformer to remove the shadow and improve the OCR accuracy. As shown in Figure 2, given the unwarped image $I_R \in \mathbb{R}^{H \times W \times 3}$, we first crop I_R into patches with a overlap of 12.5% (each patch at 128×128 resolution). All patches are fed into our illumination correction transformer for illumination correction,

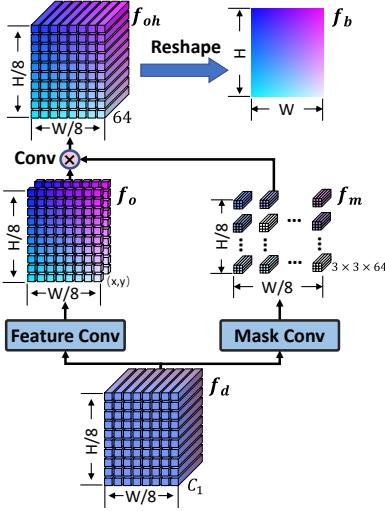


Figure 4: Illustration of the tail network for unwarping.

and the results are then stitched to get the complete corrected image I_S . We elaborate correction process in the following.

Head. Unlike the smooth, continuous backward mapping output of the geometric unwarping network, the illumination correction network outputs a high-frequency image. Hence, we use the architecture without lowering feature resolution. Specifically, given a cropped patch image $I_c \in \mathbb{R}^{H_p \times W_p \times 3}$, we first leverage a 3-layer convolutional module F_θ to extract features $f_i = F_\theta(I_c) \in \mathbb{R}^{H_p \times W_p \times c_i}$. Next, we further reshape the patch feature f_i into a sequence of flattened $P \times P$ mini-patches, i.e., $f_{sp} \in \mathbb{R}^{N_i \times c'_i}$, where $N_i = \frac{H_p \times W_p}{P^2}$ is the number of mini-patches on the input image patch, $c'_i = c_i \times P^2$ is the number of channels of transformer input. Note that in our method, c_i is set to 16, and P is set to 4.

Transformer Encoder-Decoder. Similar to the body of the geometric unwarping transformer, we devise an encoder-decoder architecture to encode features of patch images and generate pixel-level illumination correction predictions. To be concrete, the pre-extracted feature f_{sp} , added by position embedding $E_p \in \mathbb{R}^{N_i \times c'_i}$, is fed into a K-layer transformer encoder. After being processed by multiple multi-head self-attention and feed-forward layers in the same way as the computation in Equation (3), the output feature F_K^i aggregates global relationship. We subsequently use a transformer decoder to decode the illumination distribution and predict corrected pixels. The decoder takes a learnable embedding $E_d^i \in \mathbb{R}^{N_i \times c'_i}$ and a position embedding as input, and perform multi-head attention to query and aggregate features of different pixels, as in Equation (4).

Tail. The tail of the illumination correction transformer is a simple convolutional network with one layer. Given the decoded features $f_d^i \in \mathbb{R}^{N_i \times c'_i}$ from the decoder, we first reshape it back to the original shape $H_p \times W_p \times c_i$ by restoring the 2D mini-patches as well as the image patch. Then, we use a convolutional layer to estimate the corrected patch image $I_p \in \mathbb{R}^{H_p \times W_p \times 3}$.

Loss Function. We optimize the illumination correction transformer by minimizing the L_1 distance and the VGG loss between

the estimated patch image I_p and ground truth image I_{gt} as follows,

$$\mathcal{L}_{ill} = \|I_{gt} - I_p\|_1 + \alpha \|V(I_{gt}) - V(I_p)\|_1, \quad (7)$$

where α is the weight of the VGG loss and $V(\cdot)$ denotes the VGG Network [32]. The VGG loss is defined as the L_1 distance between the output of the ReLU activation layers of estimated I_p and ground truth image I_{gt} . It is based on the pre-trained 19-layer VGG network, and also known as perceptual loss or content loss.

4 DATASET

We evaluate our proposed DocTr on several datasets. To be specific, we train the geometric unwarping transformer on Doc3D dataset, following [7]. Then, our illumination correction transformer is optimized using the DRIC dataset [18]. Finally, We evaluate the performance of our DocTr on DocUNet benchmark [22] as previous works [7, 18, 22, 41] suggest. Note that due to the image resolution limit of the Doc3D dataset (*i.e.*, 448 × 448) which is not suitable for the patch-based illumination correction method, we only use Doc3D for the geometric unwarping task. Besides, the number of document images of DRIC dataset (*i.e.*, 2700) is also not enough for the geometric unwarping task, especially for our transformer-based method. We elaborate individual dataset in the following.

Doc3D. The Doc3D dataset [7] is the largest dataset to date for the document image rectification task. Created by real document data and rendering software, the dataset consists of 100k distorted document images. For each distorted document image, there are corresponding 3D world coordinate map, albedo map, normals map, depth map, UV map, and backward mapping map.

DRIC. The DRIC dataset [18] consists of 2700 distorted document images, each at 2400 × 1800 resolution. For each distorted document image, there are corresponding backward mapping map and scanned PDF image. They are also rendered by the rendering software in which many rendering techniques are used.

DocUNet Benchmark. The DocUNet benchmark [22] is a widely used dataset for document image rectification. It consists of 130 photos of real paper documents captured by mobile cameras. The documents include various types, such as receipts, letters, fliers, magazines, academic papers, and books, *etc*. Besides, their distortion and background are various to cover different levels of difficulty.

5 EXPERIMENTS

5.1 Evaluation Metrics

For geometric unwarping, we use two evaluation schemes based on pixel alignment and Optical Character Recognition (OCR) accuracy. Then, we use image similarity and OCR accuracy to evaluate the performance of illumination correction. To be specific, for pixel alignment, we use Local Distortion (LD) [43] as recommended in [7, 22, 41] to evaluate the geometric distortion of rectified images. For image similarity, we use Multi-Scale Structural SIMilarity (MS-SSIM) [39] as previous works [7, 22, 41] suggest. For OCR, following [7, 22], we choose Edit Distance (ED) [17] and Character Error Rate (CER) to evaluate the capacity on text recognition.

Local Distortion. Local Distortion (LD) [43] measures the average deformation of each pixel, and is defined as the mean displacement error based on the SIFT flow [20] ($\Delta x, \Delta y$) from the ground truth scanned image to the rectified image. The SIFT flow is a 2D

displacement field that maps each pixel from the scanned image to the rectified image. Then, LD is calculated as the mean value of L_2 distance between all matched pixels.

MS-SSIM. The Structural SIMilarity (SSIM) [38] measures the similarity between two images and is calculated on various windows of an image. However, the perceived quality of an image usually depends on the sampling density. For this reason, Multi-Scale Structural SIMilarity (MS-SSIM) [39] propose to build a Gaussian pyramid of the two images and is calculated as the weighted summation of SSIM across multiple resolutions.

ED and CER. Edit Distance (ED) [17] is a way of quantifying how dissimilar two strings are to one another, which is defined as the minimum number of operations required to transform one string into the reference string. The involved edit operations include deletions (d), insertions (i) and substitutions (s). Then, Character Error Rate (CER) can be calculated: $(d + i + s)/N$, where N is the character number of the reference string. Following [7, 22], we use Tesseract (v3.02.02) [34] as the OCR engine.

5.2 Implementation Details

We implement our DocTr in Pytorch [27]. We train the geometric unwarping transformer and illumination correction transformer independently on the Doc3D dataset [7] and DRIC dataset [18].

Geometric Unwarping Transformer. We first train the pre-processing segmentation module using Doc3D dataset [7]. During training, we randomly replace the background of the distorted document images with the texture images from Describable Texture Dataset [6] for argumentation. It is trained for 45 epochs with a batch size of 32. The Adam optimizer [14] is employed with a learning rate of 1×10^{-4} that is reduced by a factor of 0.1 after 30 epochs. We binarize the confidence map with threshold τ of 0.5.

Then, we use the Doc3D dataset [7] to train the geometric unwarping transformer. During training, we remove the background of distorted images and resize the images to 288×288 . We use AdamW optimizer [21] with a batch size of 8. It is trained for 500k iterations. The learning rate reaches the maximum 1×10^{-4} after 700 iterations and is reduced based on the One-Cycle policy [33].

Illumination Correction Transformer. During training, we randomly crop scanned PDF images and the document images in DRIC dataset [17] which are unwarped by the ground truth backward mapping. The size of the obtained image patches is 128×128 . We train the network for 35 epochs based on the AdamW optimizer [21] with a batch size of 24. The initial learning rate is set as 1×10^{-4} and reduced by a factor of 0.3 after 20 epochs. We set the hyperparameter $\alpha = 1 \times 10^{-5}$ in Equation (7).

5.3 Experimental Results

In the following experiments, we denote the proposed Geometric Unwarping Transformer and Illumination Correction Transformer as **GeoTr** and **IIITr**, respectively.

Geometric Unwarping. We compare our GeoTr with all previous methods on DocUNet benchmark [22] by quantitative and qualitative evaluation. For quantitative evaluation, we evaluate the pixel alignment and OCR accuracy performance. Note that for OCR accuracy evaluation, we select 30 images from the benchmark in

Method	LD↓	MS-SSIM↑	ED↓	CER↓
Distorted Image	-	-	2051.4	0.68
DocUNet [22]	14.08	-	-	-
DRIC [18]	18.19	-	1840.9	0.61
+ DRIC-ILL [18]	-	0.2381	1547.9	0.52
DewarpNet [7]	8.98	-	1121.1	0.38
+ DewarpNet-ILL [7]	-	0.4735	1007.4	0.35
FCN based [41]	8.50	-	1167.3	0.46
GeoTr	8.38	-	935.2	0.31
GeoTr + IIITr	-	0.4970	576.4	0.20

Table 1: Comparison with all previous works on DocUNet benchmark. The suffix “-ILL” denotes illumination correction method. “↑” indicates the higher the better and “↓” means the opposite.

GEO. Method	DewarpNet [7]			GeoTr		
	(a)	(b)	(c)	(d)	(e)	(f)
+ DewarpNet-ILL [7]	✓			✓		
+ DRIC-ILL [18]		✓			✓	
+ IIITr			✓			✓
MS-SSIM ↑	0.47	0.44	0.42	-	0.49	0.50
ED ↓	1007.4	939.6	869.1	-	669.8	576.4
CER ↓	0.36	0.32	0.30	-	0.23	0.20

Table 2: Comparison with previous illumination correction methods on DocUNet benchmark. “GEO.” denotes the task geometric unwarping. “↑” indicates the higher the better and “↓” means the opposite.

which the text makes up the majority of content, following DewarpNet [7]. As shown in Table 1, our GeoTr achieves state-of-the-art performance on all metrics. Additionally, after further illumination correction by our IIITr, we achieve a CER performance of 20.22%, which achieves an absolute improvement of 15.32% over the state-of-the-art result by DewarpNet [7]. These results demonstrate that our method can effectively rectify the structure and the content of the distorted documents. For qualitative evaluation, we compare the results with state-of-the-art methods as shown in Figure 5. As we can see, the rectified text lines of our GeoTr are much more straight. Besides, the incomplete and redundant boundaries phenomena that existing in other methods are relieved in our GeoTr.

Illumination Correction. We compare our IIITr with two existing methods denoted as “DewarpNet-ILL” [7] and “DRIC-ILL” [18]. For quantitative evaluation, we first evaluate the three methods by correcting the geometric rectified results of DewarpNet [7], as shown in method (a), (b) and (c) in Table 2. Then, considering that the code of DewarpNet-ILL [7] is not available, we correct the geometric unwarped results of GeoTr with DRIC-ILL [18] and our IIITr. The results are shown in method (e) and (f). In both comparisons, our IIITr shows much more performance gain. We further visualize the comparison with DewarpNet-ILL [7] and DRIC-ILL [18], as shown in Figure 5. Note that we visualize the DRIC-ILL based on the geometric rectified results of our GeoTr, because the geometric unwarping method of DRIC [18] is incapable to rectify the document images with background. It can be seen that our method shows less blur and shading than the other methods.

Task	Method	Time (s)	Parameters (M)
GEO.	DocUNet [22] *	17.5	69.1
	DRIC [18]	8.74	47.8
	FCN based [41] *	0.67	-
	DewarpNet [7]	0.14	86.9
	GeoTr	0.13	26.9
ILL.	DewarpNet-ILL [7]	-	-
	DRIC-ILL [18]	3.84	0.5
	IIITr	2.79	11.6

Table 3: Comparison of running time and model size with previous methods. The task geometric unwarping and illumination correction are denoted as "GEO." and "ILL.", respectively. "*" denotes the reported results in the original paper.

Base model	GeoTr				
	(a)	(b)	(c)	(d)	(e)
+ Preprocessing		✓	✓	✓	✓
+ Encoder	✓		✓	✓	✓
+ Decoder	✓	✓		✓	✓
+ Upsampling Tail	✓	✓	✓		✓
+ Bilinear Up.				✓	
MS-SSIM ↑	0.46	0.50	0.49	0.50	0.50
ED ↓	839.4	590.6	602.3	599.3	576.4
CER ↓	0.27	0.21	0.21	0.21	0.20

Table 4: Ablation experiments on GeoTr. The setting used in our final model is underlined. “↑” indicates the higher the better and “↓” means the opposite.

Base model	IIITr				
Setting	(a)	(b)	(c)	(d)	(e)
+ 64 × 64	✓				
+ 128 × 128		✓			
+ 256 × 256			✓		
+ Encoder	✓	✓	✓		✓
+ Decoder	✓	✓	✓	✓	
MS-SSIM ↑	0.49	0.50	0.50	0.50	0.49
ED ↓	651.5	576.4	551.4	590.6	602.3
CER ↓	0.23	0.20	0.19	0.21	0.21

Table 5: Ablation experiments on IIITr. The setting used in our final model is underlined. “↑” indicates the higher the better and “↓” means the opposite.

Efficiency Comparison. We compare the inference time and parameter numbers of our GeoTr and IIITr with previous works on processing a 1080P resolution document image. The evaluation is conducted on an NVIDIA GTX 1080Ti GPU. As shown in Table 3, our GeoTr and IIITr both show higher and comparable efficiency, respectively. Note that the running time of DRIC-ILL [18] is limited by the bilateral mean preprocessing on document images.

Robustness. We further evaluate the robustness of our method in terms of viewpoint and illumination change. The results are shown in Figure 7 and Figure 8, respectively. It can be seen that our

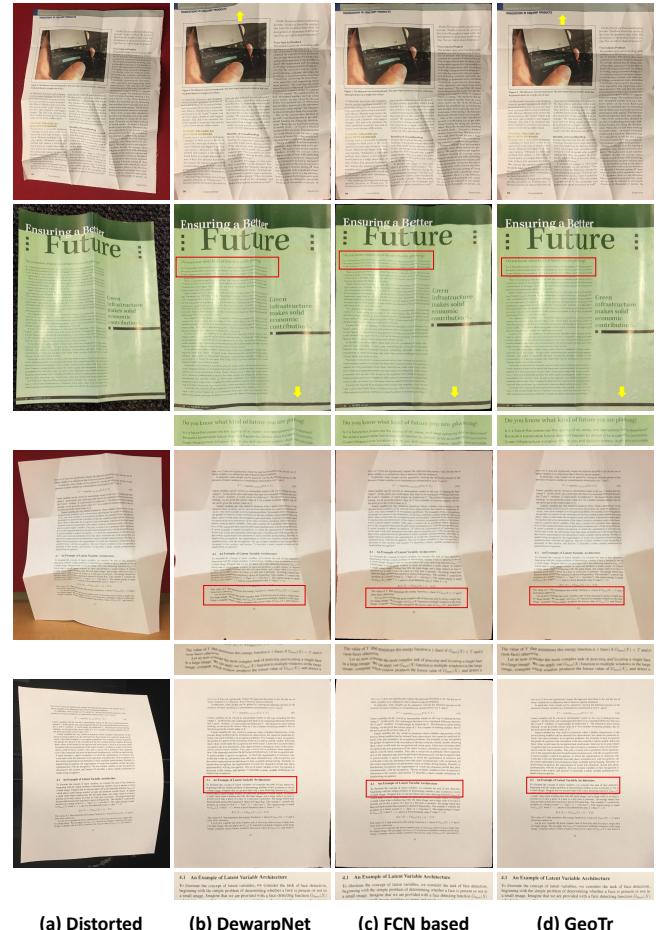


Figure 5: Qualitative comparison with state-of-the-art methods. Column 1: original distorted images, column 2: results of DewarpNet [7], column 3: results of [41], column 4: ours. The comparison of rectified boundaries and text lines are highlighted by yellow arrows and cropped text, respectively.

method shows high robustness in various viewpoints, illumination conditions, and background environments.

5.4 Ablation Studies

In this section, we validate the contribution of the key components of our proposed GeoTr and IIITr, respectively. As shown in Table 4, to verify the effectiveness of the preprocessing module, encoder, decoder, and upsampling tail, we evaluate the performance of DocTr under different settings of GeoTr.

Preprocessing. In Table 4, for setting (a) and (e), performance is improved by adding the preprocessing segmentation stage. This is because the various backgrounds of distorted document images involve an extra implicit learning burden for the network to localize the foreground document. As a result, the rectified documents often struggle with incomplete or redundant boundaries, and geometric distortion further spreads the boundary issue to nearby regions. In contrast, given the localized document, the network can focus on the content correction with an integrated document boundary.

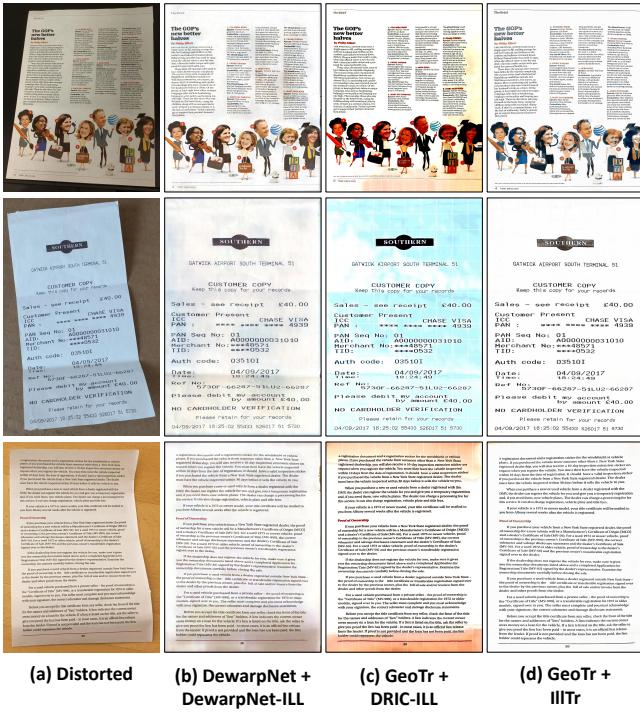


Figure 6: Qualitative comparison with illumination correction methods DewarpNet-ILL [7] and DRIC-ILL [18] on unwarped results of DewarpNet [7] and GeoTr, respectively.

Encoder and Decoder. We independently verify the encoder and decoder module of GeoTr. For setting (b) in Table 4, without any encoder block, we directly feed the flatten image features to the decoder. Then, in setting (c), we feed the encoded image features to the upsampling module without any decoder block. As we can see, our GeoTr (e) achieves better performance than the above two settings. This result demonstrates the essential of both the transformer encoder and decoder. Specifically, the self-attention mechanism in encoder block helps to process the global information of the document images. And the decoder, as well as the learned task-specific embeddings, decode the global-aware features into pixel-wise displacement information for tail end prediction.

Upsampling Tail. The decoder block output features at $\frac{1}{8}$ resolution of input document images. We compare bilinear upsampling with our learned upsampling module. As shown in setting (d) and (e), our upsampling tail produces better results.

For IllTr, we verify the setting of the patch size for stitching, as well as the effectiveness of the encoder and decoder module.

Patch size. We compare the performance of different sizes for stitching. As shown in Table 5, setting (c) shows slight improvement than setting (a) and (b). Following DRIC-ILL [18], we use setting (b) (*i.e.*, patch size 128×128), in our DocTr, to stride a balance between accuracy and inference memory.

Encoder and Decoder. Similar to the experiments for GeoTr, we validate the encoder and decoder module in IllTr. The results are shown in setting (b), (d), and (e) in Table 5. It can be seen that our IllTr (b) outperforms the setting (d) and (e), in which the encoder and decoder blocks are removed, respectively.

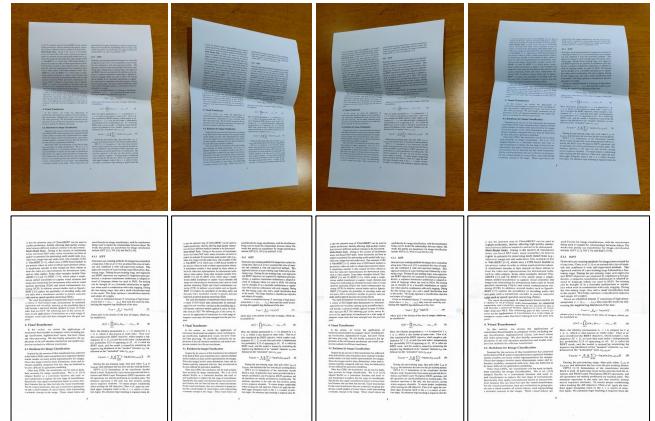


Figure 7: Robustness of DocTr in terms of viewpoint. The top row shows the distorted document images taken from different viewpoints. The second row shows the corresponding rectified document images by DocTr.



Figure 8: Robustness of DocTr in terms of illumination variation and background environment. The top row shows the distorted document images captured in different scenes. The second row shows the rectified document images by DocTr.

6 CONCLUSION

In this work, we present DocTr, a Transformer-based framework, to address the geometry and illumination distortion in document images. Thanks to the processing of global information performed by the self-attention, it achieves state-of-the-art performance on both tasks and significantly relieves the curved text lines and artifacts in rectified results of previous approaches. In addition, it shows comparable or state-of-the-art efficiency. In the future, we will utilize the text content in images to further improve the visual quality and OCR performance.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Contract 61836011 and 61632019, and in part by the Youth Innovation Promotion Association CAS under Grant 2018497. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

REFERENCES

- [1] Michael S Brown and W Brent Seales. 2001. 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. 367–374.
- [2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-End Object Detection with Transformers. In *Proceedings of the European Conference on Computer Vision*. Springer International Publishing, 213–229.
- [3] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. 2020. Pre-Trained Image Processing Transformer. arXiv:2012.00364 [cs.CV]
- [4] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. 2020. Generative pretraining from pixels. In *Proceedings of the International Conference on Machine Learning*. 1691–1703.
- [5] Chew Lim Tan, Li Zhang, Zheng Zhang, and Tao Xia. 2006. Restoring warped document images through 3D shape modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 2 (2006), 195–208.
- [6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing Textures in the Wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3606–3613.
- [7] Sagnik Das, Ke Ma, Zhixin Shu, Dimitris Samaras, and Roy Shilkrot. 2019. DewarpNet: Single-Image Document Unwarping With Stacked 3D and 2D Regression Networks. In *Proceedings of the International Conference on Computer Vision*. 131–140.
- [8] Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. 2005. A tutorial on the cross-entropy method. *Annals of Operations Research* 134, 1 (2005), 19–67.
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *Proceedings of the International Conference on Learning Representations*.
- [11] Yuan He, Pan Pan, Shufu Xie, Jun Sun, and Satoshi Naoi. 2013. A Book Dewarping System by Boundary-Based 3D Surface Reconstruction. In *Proceedings of the International Conference on Document Analysis and Recognition*. 403–407.
- [12] Huaiyu Cao, Xiaoqing Ding, and Changsong Liu. 2003. 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. 71–75.
- [13] Xincheng Ju, Dong Zhang, Junhui Li, and Guodong Zhou. 2020. Transformer-Based Label Set Generation for Multi-Modal Multi-Label Emotion Detection. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) (MM ’20). Association for Computing Machinery, New York, NY, USA, 512–520.
- [14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2015).
- [15] Hyung Il Koo, Jinho Kim, and Nam Ik Cho. 2009. Composition of a Dewarped and Enhanced Document Image From Two View Images. *IEEE Transactions on Image Processing* 18, 7 (2009), 1551–1562.
- [16] Olivier Lavialle, X Molines, Franck Angella, and Pierre Baylou. 2001. Active contours network to straighten distorted text lines. In *Proceedings of the International Conference on Image Processing*, Vol. 3. 748–751.
- [17] Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. 10 (1966), 707–710.
- [18] Xiaoyu Li, Bo Zhang, Jing Liao, and Pedro V Sander. 2019. Document rectification and illumination correction using a patch-based CNN. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–11.
- [19] Jingjun Liang, Ruichen Li, and Qin Jin. 2020. Semi-Supervised Multi-Modal Emotion Recognition with Cross-Modal Distribution Matching. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) (MM ’20). Association for Computing Machinery, New York, NY, USA, 2852–2861.
- [20] Ce Liu, Jenny Yuen, and Antonio Torralba. 2011. SIFT Flow: Dense Correspondence across Scenes and Its Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 5 (2011), 978–994.
- [21] I. Loshchilov and F. Hutter. 2019. Decoupled Weight Decay Regularization. In *Proceedings of the International Conference on Learning Representations*.
- [22] Ke Ma, Zhixin Shu, Xue Bai, Jue Wang, and Dimitris Samaras. 2018. DocUNet: Document Image Unwarping via a Stacked U-Net. In *Proceedings of the IEEE International Conference on Computer Vision*. 4700–4709.
- [23] Amir Markovitz, Inbal Lavi, Or Perel, Shai Mazor, and Roei Litman. 2020. Can You Read Me Now? Content Aware Rectification Using Angle Supervision. In *Proceedings of the European Conference on Computer Vision*. Springer, 208–223.
- [24] Gaofeng Meng, Ying Wang, Shenquan Qu, Shiming Xiang, and Chunhong Pan. 2014. Active Flattening of Curved Document Images via Two Structured Beams. In *Proceedings of the IEEE International Conference on Computer Vision*. 3890–3897.
- [25] Lothar Mischke and Wolfram Luther. 2005. Document image de-warping based on detection of distorted text lines. In *Proceedings of the International Conference on Image Analysis and Processing*. Springer, 1068–1075.
- [26] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. 2018. Image Transformer. In *Proceedings of the 35th International Conference on Machine Learning*, Vol. 80. 4055–4064.
- [27] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. (2017).
- [28] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R. Zaiane, and Martin Jaggersand. 2020. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognition* 106 (Oct 2020).
- [29] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. Technical report, OpenAI.
- [30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 234–241.
- [31] Botian Shi, Lei Ji, Zhendong Niu, Nan Duan, Ming Zhou, and Xilin Chen. 2020. Learning Semantic Concepts and Temporal Alignment for Narrated Video Procedural Captioning. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) (MM ’20). Association for Computing Machinery, New York, NY, USA, 4355–4363.
- [32] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs.CV]
- [33] Leslie N Smith and Nicholay Topin. 2019. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, Vol. 11006. International Society for Optics and Photonics, 1100612.
- [34] Ray Smith. 2007. An Overview of the Tesseract OCR Engine. In *Proceedings of the International Conference on Document Analysis and Recognition*, Vol. 2. 629–633.
- [35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. *Proceedings of the Neural Information Processing Systems*.
- [36] Huiyi Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. 2021. MaX-DeepLab: End-to-End Panoptic Segmentation with Mask Transformers. In *Proceedings of the European Conference on Computer Vision*.
- [37] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. 2021. Transformer Meets Tracker: Exploiting Temporal Context for Robust Visual Tracking. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
- [38] Zhou Wang, Alan C Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.
- [39] Zhou Wang, Eero P. Simoncelli, and Alan C. Bovik. 2003. Multiscale structural similarity for image quality assessment. In *Proceedings of the Asilomar Conference on Signals, Systems Computers*, Vol. 2. 1398–1402.
- [40] Changhua Wu and Gady Agam. 2002. Document image de-warping for text/images recognition. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 348–357.
- [41] Guowang Xie, Fei Yin, Xuyao Zhang, and Chenglin Liu. 2020. Dewarping Document Image by Displacement Flow Estimation with Fully Convolutional Network. In *International Workshop on Document Analysis Systems*. Springer, 131–144.
- [42] Zhihui Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Proceedings of the Neural Information Processing Systems*.
- [43] Shaodi You, Yasuyuki Matsushita, Sudipta Sinha, Yusuke Bou, and Katsushi Ikeuchi. 2018. Multiview Rectification of Folded Documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 2 (2018), 505–511.
- [44] Li Zhang, Yu Zhang, and Chew Tan. 2008. An Improved Physically-Based Method for Geometric Restoration of Distorted Document Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 4 (2008), 728–734.
- [45] Shengyu Zhang, Tan Jiang, Tan Wang, Kun Kuang, Zhou Zhao, Jianke Zhu, Jin Yu, Hongxia Yang, and Fei Wu. 2020. DeVLBert: Learning Deconfounded Visio-Linguistic Representations. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) (MM ’20). Association for Computing Machinery, New York, NY, USA, 4373–4382.