End-to-end Piece-wise Unwarping of Document Images
—
Supplementary Material

Sagnik Das[1,2] Kunwar Yashraj Singh[1] Jon Wu[1] Erhan Bas[1] Vijay Mahadevan[1]
Rahul Bhotika[1] Dimitris Samaras[2]

[1]Amazon AI [2]Stony Brook University
{sinkunwa, jonwu, erhanbas, vmahad, bhotikar}@amazon.com, {sadas, samaras}@cs.stonybrook.edu

In this supplementary material, we provide:

1. Discussion about the Local Distortion (LD) metric.
2. Discussion about the OCR metrics, CER and WER.
3. Details of the Global Stitching Variants.
4. Details of the Reconstruction Loss.
5. Training details.
6. Discussion about DocProj [4] trained on Doc3D.
7. A demo video of piece-wise unwarping.

We refer the main submission using *section X.X*.

1. Sensitivity of Local Distortion to Alignment

We would like highlight that the LD metric is sensitive to global misalignment. This is due to the fact that local shifts in unwarped documents increase the SIFT flow magnitude, and consequently LD. We demonstrate this phenomenon in figures 1 and 2 where our piece-wise approach produces better unwarping but incurs a higher LD than DewarpNet [2]. The LD map (middle column) shows the flow magnitude overlaid on the unwarped image. We can clearly notice high LD (denoted by blue) even at the regions where the unwarping is clearly better. This occurs frequently enough such that we obtain a higher mean LD 9.23 than that of DewarpNet [2], 8.98, which is reported in Table 1 of the main submission.

2. On the Relation Between OCR and Visual Quality

Though OCR and visual quality metrics such as LD, and MS-SSIM are typically correlated, there are occasionally examples, where the the OCR engine from Tesseract [5] can improve. We show such counter-intuitive examples in figure 3, where CER values are high, even when visual quality has improved or similar. In figure 3, column 1 shows an example where the proposed method obtains improved visual results (with straighter columns) than DewarpNet [2] but has a higher CER. We showcase another counter-intuitive example due to the OCR engine in column 2, where we have similar unwarping results, but a drastic 50% difference in CER. In terms of LD, our proposed method yields a lower score (5.30 to 4.27) in the first case but a higher score (7.26 to 8.06) in the second case. We posit that with improved OCR algorithms, such as with more modern deep learning based methods, the correlation between visual quality and downstream OCR accuracy will improve and our results will be more pronounced.

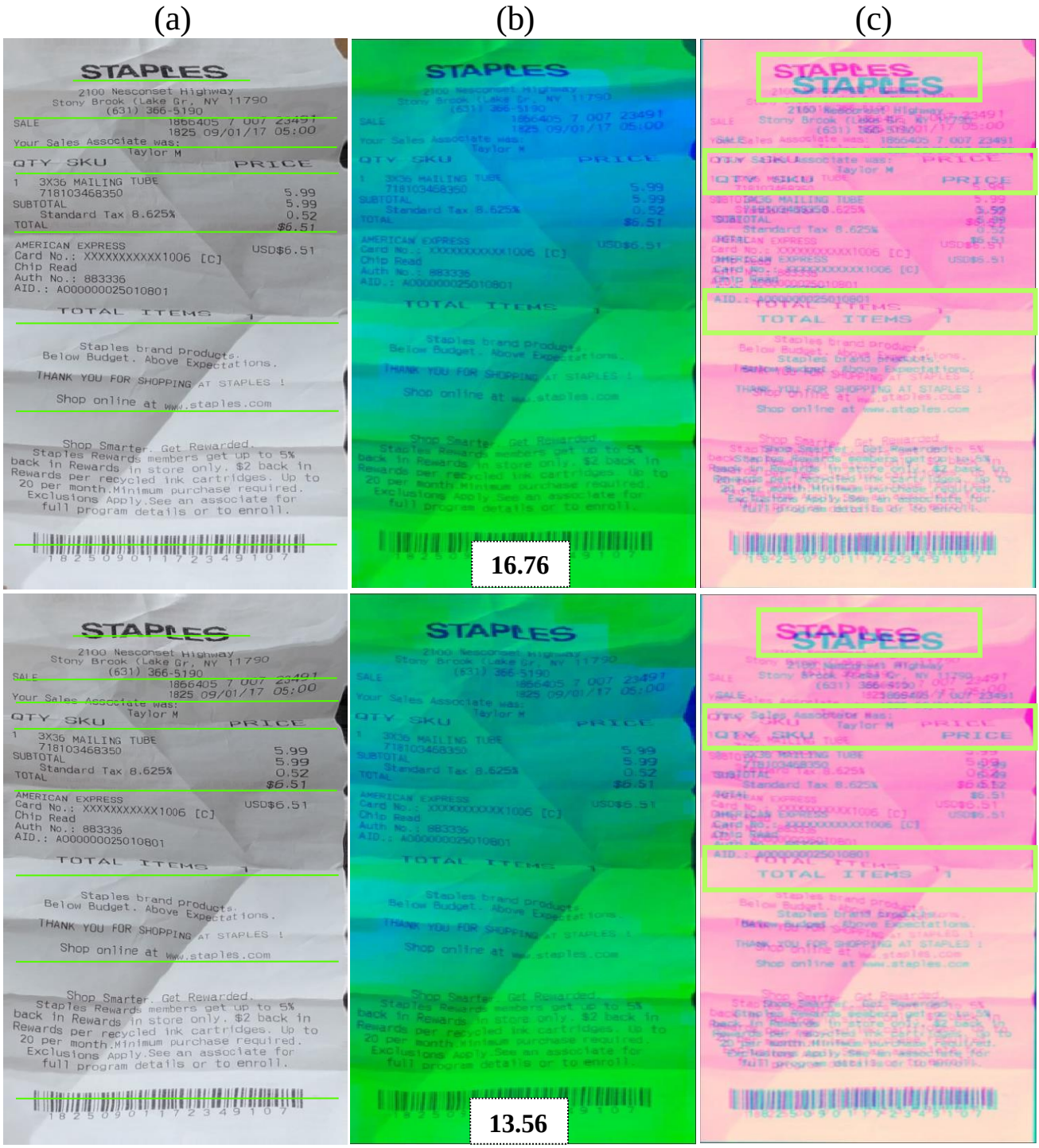

Figure 1. **Comparison of Local Distortion (LD)** Map of Proposed (top row middle) and DewarpNet (bottom row middle): (a) Unwarped Image (follow the green cue lines for a better visual comparison), (b) Unwarped image overlaid with the LD magnitude map (blue: higher LD). The corresponding LD value is shown at the bottom of each image, (c) Unwarped image (red channel) overlayed with the scan (blue channel). Green boxed regions highlight the misalignment of the unwarping algorithms with respect to ground truth. The proposed approach has better unwarping quality but global misalignment with the ground truth leads to 23% higher LD value.

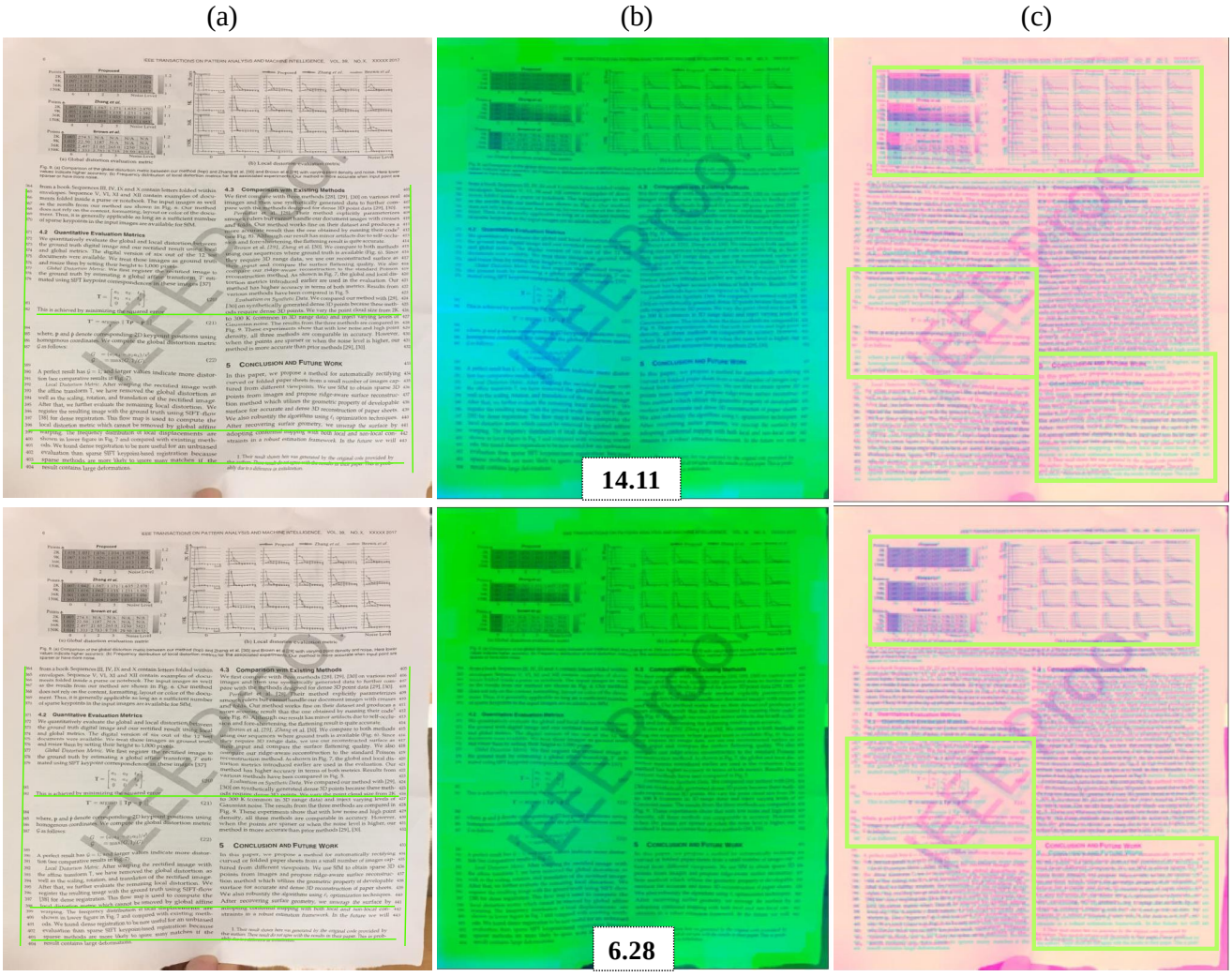

Figure 2. **Comparison of Local Distortion (LD)** Map of Proposed (top row middle) and DewarpNet (bottom row middle): (a) Unwarped Image (follow the green cue lines for a better visual comparison), (b) Unwarped image overlayed with the LD magnitude map (blue: higher LD). The corresponding LD value is shown at the bottom of each image, (c) Unwarped image (red channel) overlayed with the scan (blue channel). Green boxed regions highlight the misalignment of the unwarping algorithms with respect to ground truth. The proposed approach has better unwarping results (with straighter columns and better line alignment across two columns) but global misalignment with the ground truth leads to 51% higher LD value.

3. Global Stitching Variants

We have experimented with three variants of the global texture stitching module. The main difference among the variants is the long-skip connection [6] and the fusion block ($\mathcal{F}$) used to fuse the local pyramid features extracted from G-FPN and the global branch. Schematic diagrams of these variants are given in figure 4. In the baseline approach (Gl-C), the long skip is an identity function on the global backward map (BM). The fusion block is implemented as channel-wise concatenation of the global BM and the local G-FPN features. Gl-RF uses an identity long-skip connection of the global BM features from the penultimate layer of Global Warp decoder. The Gl-RF fusion block is implemented as element-wise addition of the local G-FPN and global BM features. On the other hand, Gl-R uses a shallow encoder to first encode the global BM for the long-skip connection. The quantitative comparison of these variants is presented in Table 1. In terms of MS-SSIM on real benchmark images and SSIM on synthetic validation images, the Gl-R variant performs the best, and is used in our proposed network. Gl-RF variant shows better LD due to the global features directly used as long skip. This demonstrates the fact that the unwarping network favors a global improvement rather than the local if stronger global features are provided. On

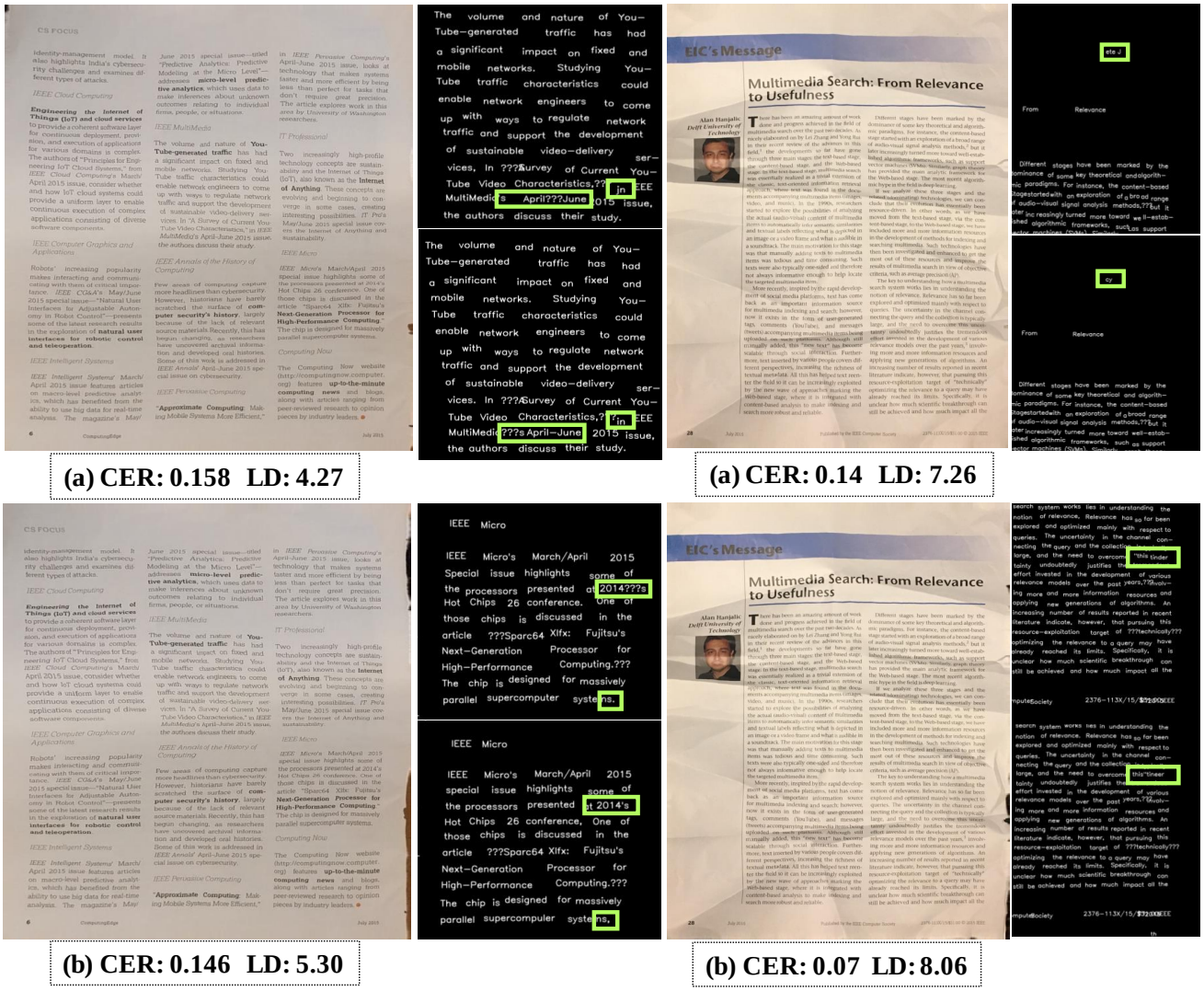

Figure 3. **Relation of OCR and Visual Quality**: **Column 1 and 3**: (a) Proposed and (b) DewarpNet [2] with the respective CER. **Column 2 and 4**: Enlarged detected words for visual comparison (green boxes show spurious character recognition), Proposed (row 1 and 3) and DewarpNet (row 2 and 4). Although, CER values are higher, unwarping results are better (column 1) or similar (column 3).

Stitching	MS-SSIM ↑	LD ↓	Val SSIM ↑
Gl-C	0.4530	10.69	0.8176
Gl-R	**0.4663**	10.14	**0.8266**
Gl-RF	0.4628	**10.07**	0.8157

Table 1. Comparison of different variants of the global texture stitching modules. See section 3.

the contrary when we use a shallow encoder to encode the low-level information available at the global branch, the local improvements become more prominent thus achieving better SSIM. There is an inherent trade-off between the local and global unwarping which is unexploited in single branch global approaches [2].

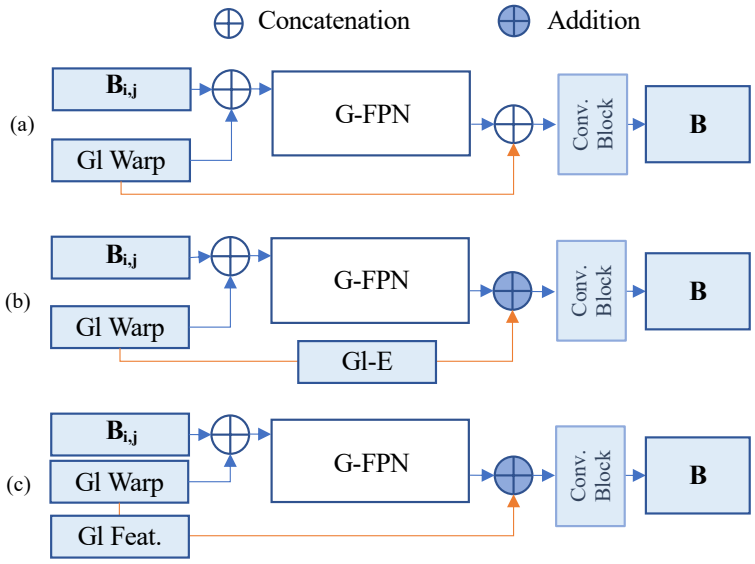

Figure 4. **Different variants of global stitching network**: (a) Gl-C, (b) Gl-R, (c) Gl-RF. $B_{i,j}$ denote the local BMs, Gl Warp denotes the global branch, Gl Feat. denotes the global features from the global decoder. Gl-E denotes a shallow convolution encoder for global BM. The orange arrow denotes the long-skip connection.

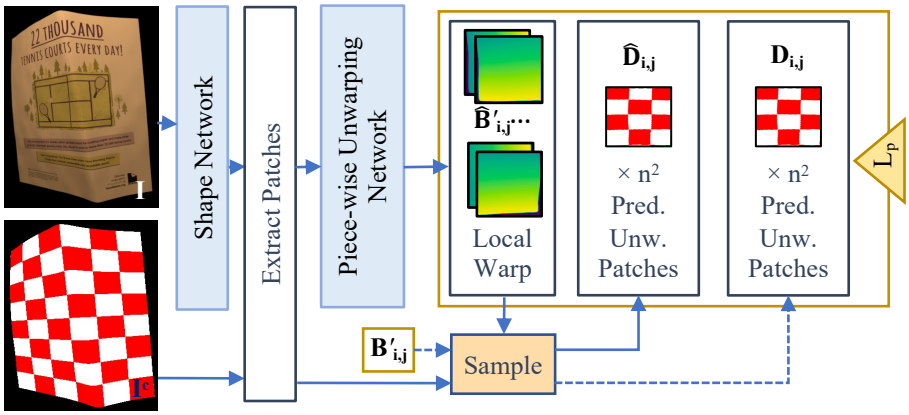

Figure 5. **PUNet training with checkerboard images for reconstruction loss**: Blue arrows denote training flow. Dashed arrows denote unwarping the checkerboard image patches ($I^c_{i,j}$) using ground-truth local backward map ($B'_{i,j}$). Triangle denotes the PUNet loss function, yellow rectangles enclose the variables used in loss calculation.

4. On Reconstruction Loss

In loss functions L_P and L_S of the main submission, we use an L2 reconstruction loss on the unwarped images D and $\hat{D}$. Documents are quite varied, they can have many variations in texture, as well as many regions that are texture-free (no-text).

In order to get a meaningful gradient of the reconstruction loss, we use a checkerboard texture which is kept consistent across a training batch. However, any other texture is also a viable choice as long as it is kept consistent across all the examples during training. This is done to avoid errors in areas of a document devoid of texture.

We use the available checkerboard textured images (I^c) in Doc3D [2] following the DewarpNet training procedure. Specifically, during the training of global stitching network, GSNet, we unwarp I^c using the predicted backward map ($\hat{B}$) and ground-truth backward map (B) to obtain $\hat{D}$ and D respectively. Similarly for the piece-wise unwarping network, PUNet training $I^c_{i,j}$ is unwarped using the predicted local backward map ($\hat{B}'_{i,j}$) and ground-truth local backward map ($B'_{i,j}$) to obtain $\hat{D}_{i,j}$ and $D_{i,j}$ respectively. Where $I^c_{i,j}$ denote the patches of the checkerboard image corresponding to the local backward map ($B'_{i,j}$). A schematic diagram of the PUNet training is presented in figure 5.

5. Training details.

5.1. Training schedule

In this section, we provide step-by-step details about the training of our proposed approach.

- First, we separately train the shape network (SNet), the piece-wise unwarping network (PUNet), and the Global Stitching Network (GSNet).

  The SNet is trained with tightly cropped images as input with variable padding size in the range of $[15, 20]$ pixels. We randomly replace the background and apply color jitter as training augmentations. We use L_c as the loss function to train SNet.

  For the PUNet training, we utilize the patches from the ground-truth 3D coordinate maps as input and train to regress the local BMs. Initially, PUNet is trained until convergence with random patches extracted from the ground-truth 3D coordinate maps. The input patch size is varied between [0.4,0.6] times of the image width. The loss defined as L_p is used to train PUNet. During this round of training, we set $\beta_2 = 0.0$.

  For GSNet training, we use ground-truth local BM patches from each image as input and train GSNet to stitch them with the loss function L_s.

  All three networks are trained until convergence.

- In the second round of training, we jointly train the SNet and PUNet by using the loss function L_p with $\beta_2 = 0.5$. The SNet and PUNet weights are initialized using the separately trained models, respectively. We utilize the same augmentations as the separate training. In this case, SNet- predicted 3D coordinate maps are used as PUNet. However, instead of random patches, we use all the patches extracted from each 3D coordinate map as input to the PUNet.

- In the final round of training, GSNet is trained with fixed SNet and PUNet using L_s as the loss function. Instead of ground-truth local BMs, predicted local BMs from PUNet are used as input to the GSNet. The SNet and PUNet models are initialized using the best models of the previous round. The GSNet is initialized with the best model of the first round of separate training.

5.2. Hyperparameters

SNet is trained with 256×256 sized images. For PUNet we set $n = 2$ and use 128×128 sized shape patches $C_{i,j}$ as input. PUNet outputs same-sized local BM predictions. Each local BM is then resized to $128/n$ and used as an input to CPM. Outputs of CPM and inputs to the global stitching module are 128×128. We use 5 Residual Channel Attention Blocks [6] to construct the feature pyramid network, and use 4 times feature reduction in the channel attention blocks. To train each network we use the Adam [3] optimizer with initial learning rate of $1e-5$. The learning rate is halved if the validation error doesn't decrease in 5 consecutive epochs.

Loss weights. In the first round of training, α is linearly increased from 0.1 to 0.5 every 20 epochs, $\beta_1 = 0.03$, $\beta_2 = 0.0$ and $\gamma = 0.03$. In the second round, β_2 is set to 0.5. In the final round, γ is set to 0.03. We found that using higher values for β_1 and γ results in artifacts on the unwarped image at test time.

Number of patches. Generally, patch sizes should have enough context to sufficiently infer and stitch the BM. We have experimented with patches of 2x2 (50%), and 4x4 (25%) of the image width. For 4x4 patches, we noticed approximately a 5% higher unwarping L2 error on the validation set. Therefore we perform the experiments with 2x2 patches.

6. Discussion about DocProj [4] trained on Doc3D

In this section, we discuss DocProj [4] training on Doc3D [2] dataset. [4] is trained using synthetically warped documents rendered using Blender [1]. Comparatively, Doc3d shapes are captured using a depth camera and rendered using Blender with numerous document textures. Therefore, Doc3D provides more challenging and realistic deformation cases in the training set. For a fair comparison with the proposed method, which is trained on Doc3D, we train DocProj [4] using Doc3D. However, we had to relax the assumption in [4] that local patches do not contain any background. This assumption is not applicable for Doc3D images because most of the paper shapes are moderately warped, and the camera view is not guaranteed to be aligned with the document boundary. Under this relaxed condition, we could not achieve a reasonable model for DocProj and replicate the pre-trained model's performance. As per our model, DocProj trained on Doc3D yields 0.2667 MS-SSIM and 25.34 LD. On the contrary, the pre-trained model released by the authors achieves 0.3832 MS-SSIM and 12.83 LD.

7. Demo Video

We provide a video (3515-supp-video.mp4) demonstrating our method and piece-wise unwarping of warped documents for different views across a video. Piece-wise and globally unwarped results are similar even under noticeable camera pose changes, demonstrating the proposed method's robustness.

References

[1] Blender - a 3D modelling and rendering package. 7

[2] Sagnik Das, Ke Ma, Zhixin Shu, Dimitris Samaras, and Roy Shilkrot. DewarpNet: Single-image document unwarping with stacked 3D and 2D regression networks. In *Int. Conf. Comput. Vis.*, 2019. 1, 4, 6, 7

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 6

[4] Xiaoyu Li, Bo Zhang, Jing Liao, and Pedro V. Sander. Document Rectification and Illumination Correction using a Patch-based CNN. *ACM Transactions on Graphics (TOG)*, 2019. 1, 7

[5] R. Smith. An Overview of the Tesseract OCR Engine. In *ICDAR*. IEEE, 2007. 1

[6] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018. 3, 6