Research Article | Vol. 27, No. 11 | 27 May 2019 | OPTICS EXPRESS 15880
Optics EXPRESS

Fiber bundle imaging resolution enhancement using deep learning

JIANBO SHAO,[1,2] JUNCHAO ZHANG,[1,3] RONGGUANG LIANG,[1,*] AND KOBUS BARNARD[2,4]

[1]*College of Optical Sciences, University of Arizona, Tucson, Arizona 85721, USA*
[2]*Department of Electrical and Computer Engineering, University of Arizona, Tucson, Arizona 85721, USA*
[3]*Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, Liaoning Province 110016, China*
[4]*Department of Computer Science, University of Arizona, Tucson, Arizona 85721, USA*
[*]*rliang@optics.arizona.edu*

Abstract: We propose a deep learning based method to estimate high-resolution images from multiple fiber bundle images. Our approach first aligns raw fiber bundle image sequences with a motion estimation neural network and then applies a 3D convolution neural network to learn a mapping from aligned fiber bundle image sequences to their ground truth images. Evaluations on lens tissue samples and a 1951 USAF resolution target suggest that our proposed method can significantly improve spatial resolution for fiber bundle imaging systems.

1. Introduction

Fiber bundle (FB) imaging technique has demonstrated its success in biomedical endoscopy [1] by providing cellular-level spatial resolution. Due to the irregular layouts of fiber cores in FBs, images taken by such systems have well known honeycomb-like fixed pattern noises. Moreover, the spatial resolution of FB imaging systems are limited by diameter and density of fiber cores. To address these two issues, many methods have been proposed in the past decades and literature survey in this field can be found in our two previous works [2, 3].

Previously, we proposed a deep learning based FB image restoration method [3]. We built a dual-sensor FB imaging system to capture well registered pairs of FB images and corresponding ground truth (GT) data. We then developed a generative adversarial restoration neural network (GARNN) to learn a direct mapping from FB to GT images. Evaluation of our trained model shows that it can remove fixed pattern noises completely from a single FB image as direct input and the spatial resolution is increased for samples that are the same as ones in the training set. However, restoration accuracy of this method is still limited and small object details are hardly recovered, mostly due to the limited information from one single FB image. Moreover, there is no apparent resolution enhancement across different samples (cross-type).

Recent developments in deep learning based video resolution enhancement methods [4, 5] show that image restoration and reconstruction accuracy can benefit from exploiting information from multiple consecutive input frames. Also, we have demonstrated improvement of spatial resolution of FB imaging systems from multiple frames with a classical (non deep learning) approach [2].

In this paper, we present a deep learning based approach to estimate fixed-pattern-free and high-resolution (HR) image from multiple input FB frames. First, we train a motion estimation network by minimizing photometric losses between input FB image pairs' corresponding ground truth (GT) images. This motion estimation network estimates unknown motions represented by homographies among FB image sequences, despite the prominent fixed pattern noise in the input frames. We then align FB images by warping them to reference frames using the estimated homographies and then train a 3D convolution neural network to learn a direct mapping from

#363190 https://doi.org/10.1364/OE.27.015880
Journal © 2019 Received 25 Mar 2019; revised 2 May 2019; accepted 3 May 2019; published 20 May 2019

input aligned FB image sequences to their corresponding GT images. We test our method on a dataset obtained from the lens tissue samples and a 1951 USAF resolution target and we demonstrate significant spatial resolution enhancements.

2. Hardware setup

We use the same dual-sensor FB imaging system that we used in our FB image restoration method [3]. This imaging system provides well-registered "one-to-one" pairs of FB images and their corresponding GT data. However, instead of capturing one pair for each sample region, we capture multiple pairs of FB images for their corresponding GT images by randomly generating small random transverse displacements between the FB probe and the test sample using a programmable XY motion stage. We capture all the images using a white LED light source.

3. Method pipeline

Figure 1 shows the pipeline of our proposed method, which consists of a motion estimation network and a 3D convolution network. The input FB image sequence passes through the motion estimation network for estimating homographies. The spatial transformer then aligns the input FB image sequence using estimated homographies and passes the aligned sequences into the second 3D convolution network, which will finally reconstruct a HR image. The architecture details of these two networks are discussed next.

3.1. Architecture of the motion estimation network

Due to fixed pattern noise in FB images, aligning FB image sequences is not trivial. When we capture FB images by randomly shifting the sample, only signals that come from the sample will move accordingly, and signals from fixed patterns will remain stationary. Therefore, traditional motion estimation methods, such as ECC [6], cannot directly align FB image sequences because of the mixed signals in FB images. They would require a preprocessing technique to remove fixed pattern noises first to correctly align FB images.

Our design of the motion estimation network comes from an unsupervised deep homography network [7]. For the same reason above, this network cannot directly align raw FB image sequences. However by changing the data source for the loss function of this deep homography approach, our motion estimation network can estimate unknown motions with raw FB image pairs as direct inputs. In particular, we calculate the loss function as the difference between input FB image pair's corresponding GT image pair, rather than the FB ones. Since the GT images are displaced identically, but have no fixed pattern noise, the motion estimation network is rewarded for finding evidence of the true motion and learns to ignore the fixed pattern. Hence, once trained, the motion estimation network can directly estimate the unknown motions for input FB image pairs.

Given a pair of FB images F_i and F_j , we choose F_j as the reference frame and F_i is the target image. We feed this pair into a neural network model that has the similar architecture to the VGGNet [8]. This VGGNet-like model has demonstrated good performance in several deep learning based homography estimation tasks [7, 9]. In this VGGNet-like model, there are 8 convolution layers, the first four layers have 64 filters with size 3×3 and the last four layers have 128 filters, also with size 3×3. A max pooling layer is inserted between every two convolution layers. The output layers of this model are two fully connected layers. The first fully connected layers has 128 units, which is different from the designs in [7, 9] that used 1024 units. We use fewer units to allow our network to accept large input image sizes under common memory restrictions. The second fully connected layer has 8 units, which represents homography with 8 parameters. In addition, we use dropout with probability 0.5 after the last convolution layer and

Optics EXPRESS

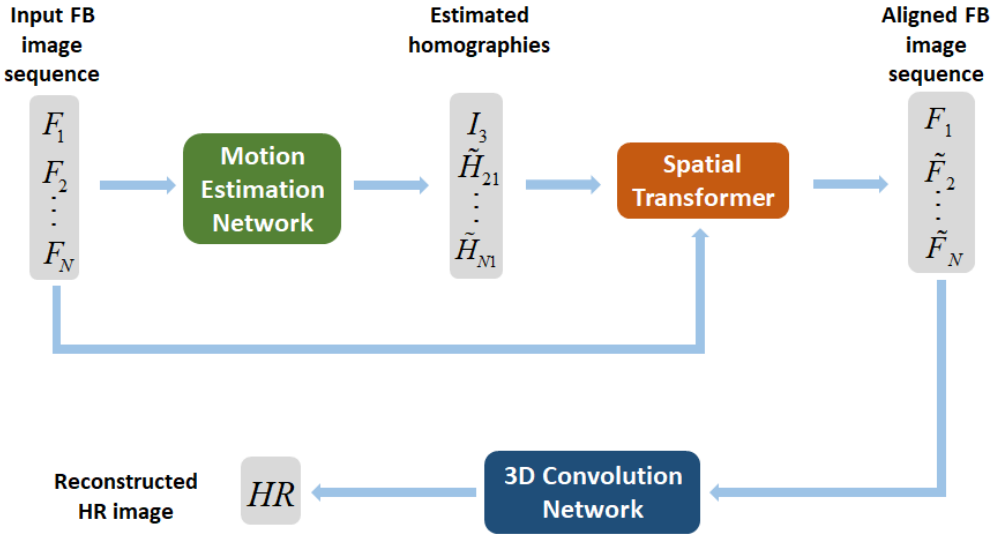

Fig. 1. The pipeline of our proposed resolution enhancement method with multi-frame FB images input using deep learning. We choose the first FB frame F_1 as the reference image for motion estimation. The raw FB image sequence is aligned by the spatial transformer with estimated homographies from the motion estimation network. Our 3D convolution network then takes the aligned FB image sequence as input and finally output one HR image.

the first fully connected layer.

This VGGNet-like model outputs an estimated 4-point homography parameterization [10]. The 4-point homography parameterization uses corner offsets with 8 parameters to represent the geometric relationship, and it has a one-to-one mapping with the traditional 3×3 homography matrix. As suggested in the literature [7,9], the rotation and shear components in 3×3 homography have much smaller magnitudes than other components (translation, scale). Therefore, a small error in rotation and shear components will have great impact for the homography and potentially causes the homography to become singular during loss minimization. Thus we choose the 4-point homography parameterization to avoid such issues for our image alignment task. We then use a Tensor Direct linear Transform that is described by Nguyen et al. [7], to obtain a traditional 3×3 homography matrix $\tilde{H}_{ij}$, which represents estimated geometric mapping from F_i to F_j.

Next, we pass the image G_i , which is the corresponding GT data of the input target FB image F_i , and the estimated 3×3 homography matrix $\tilde{H}_{ij}$ into a spatial transformer block. This block estimates the warped coordinates of the input image G_i based on $\tilde{H}_{ij}$ and outputs the aligned image $\tilde{G}_i$ using bilinear interpolation. More details of this spatial transformer can be found in Nguyen et al. [7].

Finally, we calculate the loss function of this motion estimation network as the L1 pixel-wise difference among G_j , which is the corresponding GT image of the input FB reference image F_j , and the warped image $\tilde{G}_i$. This is different from the loss function used in Nguyen et al. [7], we calculate the loss in this way so that our motion estimation network can estimate the motions among input FB image pairs directly. It is given by

$$L_{\text{motion}} = \|G_j - T_{\tilde{H}_{ij}}(G_i)\|_1, \tag{1}$$

where $T_{\tilde{H}_{ij}}(\cdot)$ denotes the warping operation using the spatial transformer that is determined by the estimated homography $\tilde{H}_{ij}$, and $\tilde{G}_i = T_{\tilde{H}_{ij}}(G_i)$.

Because image pair $\{F_i, G_i\}$ and image pair $\{F_j, G_j\}$ are both well-registered, the geometric

mappings from G_i to G_j and from F_i to F_j are essentially the same. Therefore, by calculating the loss function as described in Eq. 1, we can force this VGGNet-like model to directly estimate geometric mappings between input FB image pairs, despite fixed pattern noises in our input FB image pairs. During training, we update the trainable parameters in this motion estimation network with the Adam Optimizer [11].

We use the trained motion estimation network to directly estimate motions for raw FB image sequences. Given a raw FB image sequence $\mathbf{F} = \{F_1, F_2, ..., F_N\}$, we choose the first image F_1 to be the reference frame, $\{F_2..., F_N\}$ to be the target images, and estimate homographies $\{I_3, \tilde{H}_{21}, ..., \tilde{H}_{N1}\}$ for this entire sequence with respect to the reference. Because F_1 is the reference frame, the homography for F_1 is a 3×3 identity matrix I_3. By applying the same spatial transformer as described above, we obtain the aligned FB image sequence $\tilde{\mathbf{F}} = \{F_1, \tilde{F}_2, ..., \tilde{F}_N\}$ and feed them into the second 3D convolution network.

3.2. Architecture of the 3D convolution network

Our design of the 3D convolution network is inspired by [12]. Instead of taking a single frame image input, the 3D convolution network can take multi-frame images as the input by adding a depth dimension. Compared to 2D convolution networks, the 3D convolution network uses 3D convolution filters and it shows better performance in exploiting spatial-temporal information features for input image sequences [5, 12].

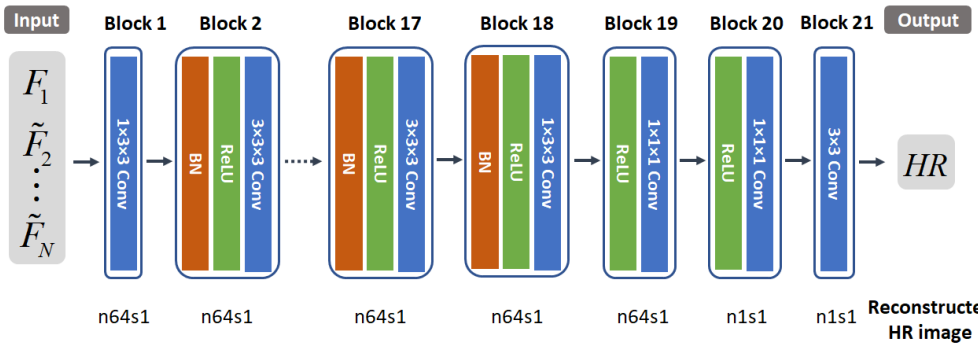

Fig. 2. Architecture of our 3D convolution network. n represents number of filters and s is stride size for each convolutional layer.

The proposed 3D convolution network is shown in Fig. 2. Block 1 serves as the input layer of aligned FB image sequence $\tilde{\mathbf{F}}$. The input data first passes through 64 3D convolution filters with size $1 \times 3 \times 3$. Block 2 to 17 contains the same structure, with batch normalization [13], ReLU regularizer [14], and 64 filters with size $3 \times 3 \times 3$. Block 18 has batch normalization, ReLU, and 64 filters with size $1 \times 3 \times 3$. Block 19 has ReLU and 64 filters with size $1 \times 1 \times 1$. Block 20 has ReLU and 1 filter with size $1 \times 1 \times 1$. The last Block 21 is the output layer, which has one 2D filter with size 3×3 that applies along the depth dimension to reconstruct one estimated HR image frame. We formulate the loss function of our 3D convolution network as

$$L_{3D} = \|G_1 - \mathcal{N}_{3D}(\tilde{\mathbf{F}})\|_2 , \tag{2}$$

where $\|\cdot\|_2$ denotes $L2$ pixel-wise norm, G_1 is the input reference FB frame's corresponding GT image, and $\mathcal{N}_{3D}(\tilde{\mathbf{F}})$ represents 3D convolution network's output HR image with aligned FB image sequences $\tilde{\mathbf{F}}$ as input. Similarly with training our motion estimation network, we use the Adam optimizer [11] to update the parameters in our 3D convolution network.

Optics EXPRESS

4. Implementation

We run and test our neural networks on Nvidia graphics processing units (GPUs), using the TensorFlow library [15] with Python platform. For testing the traditional image alignment method ECC, we use a MATLAB toolbox for image alignment and registration [16] and run it on a platform with two six-core Intel Xeon E5-2620 v3 processors. The motion estimation network and the ECC method use different spatial transformers (Python implementation versus MATLAB implementation) with different warping and interpolation schemes, which will potentially cause small differences during the objective evaluation of their image alignment performances.

5. Dataset

To evaluate our proposed method, we capture data from 390 unique lens tissue regions. We fix the sample onto a programmable XY motion stage and for each sample region, we capture 10 pairs of FB and GT images by introducing small unknown transverse displacements. All the captured original data are RGB (red, green and blue) color images and have the same pixel dimensions of 700×700. We convert the raw images into grayscale ones for all our experiments described in this paper.

We allocate 180 sample regions for the motion estimation network, with 18 for validation and 162 for training. For the training dataset, we iteratively assign each of the 10 FB images that are from one same sample region as the references, to generate 90 pairs that represent 90 different homographies, between each of the 10 choices for the reference and the 9 remaining images. Thus there are $14,580$ (90×162) image pairs allocated for training the motion estimation network. The input FB images of this motion estimation data are 640×640 pixels because we crop out 30 pixels near each border to avoid border effects. Similarly during the loss calculation (see Eq. 1), we also crop out 30 pixels for the warped image and the reference image. For each training epoch, we use a batch size of 8 and we train for 50 epochs.

For the 3D convolution network, we provide it with 180 sample regions, 21 for validation and 159 for training. Due to the border pixel cropping from the motion estimation network, all the images for the 3D convolution network are 640×640 pixels. We train this 3D convolution network for 50 epochs, with batch size of 32 for each epoch. The remaining 30 out of total 390 regions are used as the test dataset for evaluating both motion estimation network and 3D convolution network.

To compare our motion estimation network with the ECC image alignment method, we use the same training dataset for the 3D convolution network to train a GARNN FB image restoration neural network [3]. Such processing is essential because the ECC method requires preprocessing to remove fixed pattern noise to align FB images correctly.

6. Experimental results

6.1. Evaluation on the motion estimation network

We first evaluate the motion estimation network using the mean absolute value (MAE) between the reference frames and the warped target images, as the quality measure. Due to the fixed structural patterns in FB images, we choose their corresponding GT images for the MAE calculations, since input FB image pairs and their corresponding GT image pairs share the same geometric mapping relationships.

In our motion estimation network, larger unit numbers, N_{fc1}, of the first fully connected layer require more memory to compute and store more weights. Therefore, it is important to strike a balance between the hardware usage and the performance in choosing the unit number. We compare two different unit numbers, $N_{fc1} = 128$ and $N_{fc1} = 512$. We plot the curves of the average MAEs against training epoch for these two numbers in Fig. 3, using the assigned validation dataset. Both curves drop quickly and achieve comparable MAE performance. N_{fc1}

Optics EXPRESS

could potentially be further decreased. However, $N_{fc1} = 128$ achieves reasonable memory space usage using modern PC hardware configurations, and so we settled on $N_{fc1} = 128$ for our implementation.

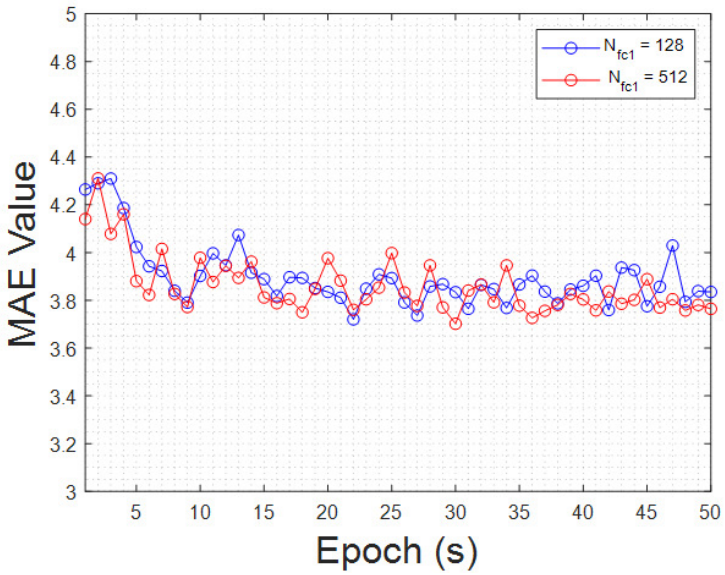

Fig. 3. Evaluating the motion estimation network: the average MAE values against the training epoch on the validation dataset with two different unit numbers (128 and 512) of the first fully connected layer.

Next we compare the motion estimation network to the traditional image alignment method ECC. We evaluate their image alignment performances on the lens tissue test dataset using MAE values. The unaligned raw FB sequences has the MAE value of 9.09, the the aligned FB sequences using the motion estimation network has the MAE value of 3.58, and the aligned FB sequences using the ECC method has the MAE value of 2.72. For the unaligned FB sequences, the MAE values are calculated between the reference frames and the raw target images. For the motion estimation network and the ECC method, their MAE values are calculated between the reference frames and the warped target images. The ECC method achieves the best image alignment performance while our motion estimation network achieves fairly good performance, while running much faster.

The predication time for our 3D convolution network (discussed next) is less than one second per output frame, and as we decrease this further, faster motion estimation will be critical for achieving real time performance. For registering one image pair, the motion estimation network takes approximately 0.076 second (on GPU) and the ECC method takes roughly 11.4 seconds (on CPU). While GPU and CPU times are not directly comparable, the nature of the ECC algorithm suggests that even a good GPU implementation will not match the neural network motion estimation time.

6.2. Evaluation on the 3D convolution network

Next, we use the trained motion estimation model to align both training and a validation dataset that are assigned for the 3D convolution network. Following Jo et al. [5], we choose the input FB image sequence length to be 7 for training our 3D convolution network. We plot the average peak signal-to-noise ratio (PSNR) value for the validation dataset against the training epoch number

in Fig. 4. This curve oscillates substantively, but becomes somewhat stable after more training epochs. We used 50 epochs of training for subsequent experiments.

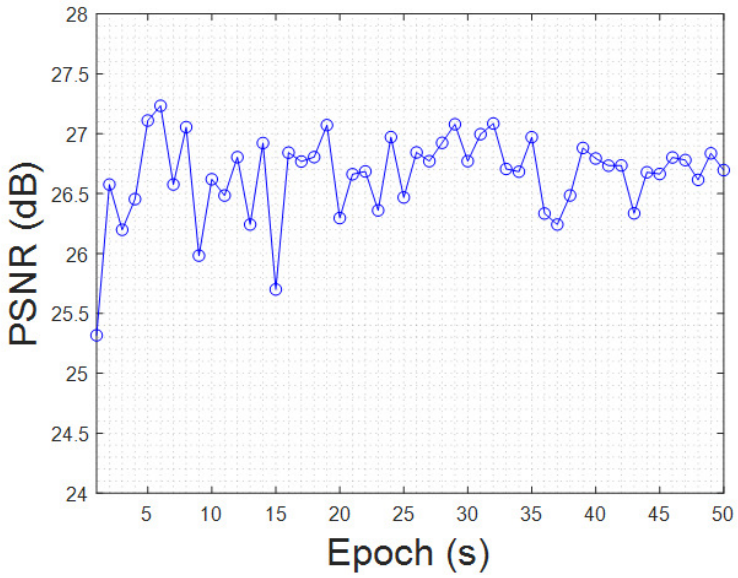

Fig. 4. Evaluating the 3D convolution network (input sequence length 7): the average PSNR values of the validation dataset against the number of training epochs.

Table 1. Quantitative Measures on the Lens Tissue Sample Experimental Results

	FB	GARNN	3D Convolution (3)	3D Convolution (7)
PSNR/dB	14.6	**26.6**	**26.6**	26.4
SSIM	0.082	0.238	0.276	**0.285**
IFC	0.275	0.700	0.875	**0.896**
HAARPSI	0.228	0.540	0.566	**0.569**

We then evaluate the performance of our 3D convolution network on the resolution enhancement task on the test dataset. Here we also study the input sequence length affects the resolution enhancement by training and testing two 3D convolution networks using input sequences of lengths 3 and 7. In addition, we train a GARNN model [3] using the same aligned training dataset that is used by the 3D convolution network, to compare our proposed method with that method.

We use four measures in our evaluation, including the commonly used PSNR measure, which is based on pixel-wise intensity difference between two images. However, it is well-known that PSNR fails to describe the perceptual quality of images. Therefore, we also use SSIM [17], IFC [18] and HAARPSI [19]. These perceptual quality indexes range from 0 to 1, with higher values representing better results. The PSNR, SSIM, IFC and HAARPSI results are shown in Table 1. We see that the GARNN and the 3D convolution networks achieve about the same PSNR gain, with the 3D convolution network for 7 images being a bit less than the other two by about 0.2dB. However, both 3D convolution networks achieve higher perceptual indexes (SSIM, IFC and HAARPSI) than the GARNN method, and the 3D convolution network yields better

Optics EXPRESS

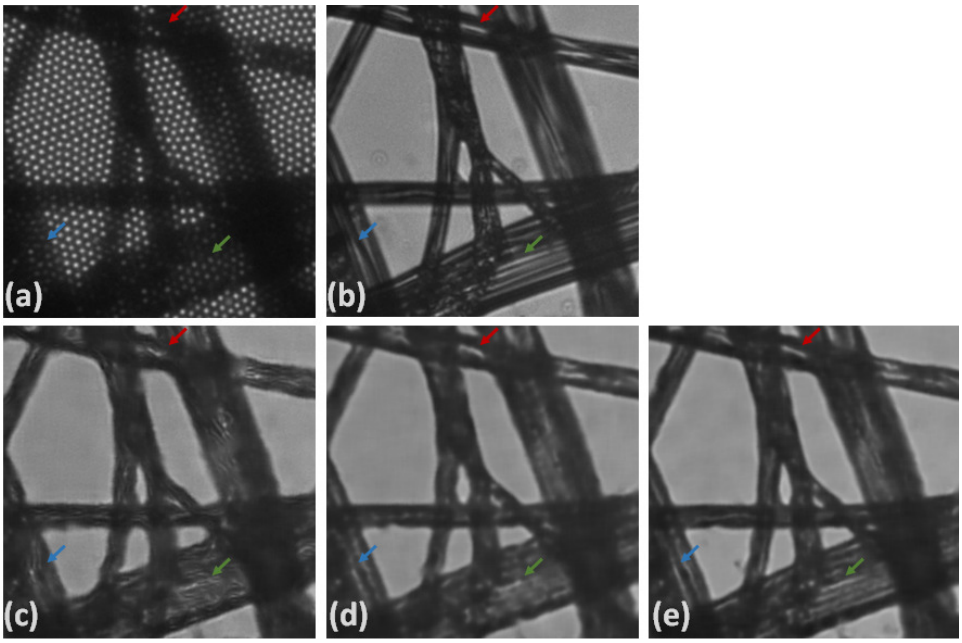

Fig. 5. Experimental results with the lens tissue sample: (a) raw FB image, (b) ground truth FB image, (c) result from GARNN, (d) result from 3D convolution network with input sequence length 3, and (e) result from 3D convolution network with input sequence length 7. We draw the arrows of three different colors (red, green and blue) to mark three small representative areas, which show that 3D convolution network with input sequence length 7 can recover the finest details.

perceptual indexes using 7 images instead of 3.

Figure 5 shows qualitative results from one lens tissue region. The GARNN method and two 3D convolution networks all completely remove the structural pattern noises. However, the 3D convolution networks are able to recover finer details, with an input FB image sequence length 7, giving the best visual result. We draw several arrows in Fig. 5 to mark three representative small areas that demonstrate the advantages of the 3D convolution methods for recovering finer details (arrows with the same color represent the same small regions). For example, in the small area marked by the green arrow, there is a bright stripe in the GT image. The GARNN method and the 3D convolution network with input FB image sequence length 3 fail to recover this, while the 3D convolution method with input sequence length 7 reconstructs this narrow bright stripe reasonably well.

We also evaluate the performance of our methods in the cross-type dataset. We capture 10 pairs of FB and GT images from a 1951 USAF resolution target by introducing random transverse displacements. We again use the MAE values to evaluate the image alignment performances of our motion estimation network on cross-type dataset, and we similarly compare it to the ECC method. For the unaligned raw sequence, the aligned sequence using the motion estimation network and the aligned sequence using the ECC method, their MAE values are 9.64, 6.41 and 3.81 respectively. Compared to the within-type image alignment performance on lens tissue test dataset shown above, our motion estimation network has worse performance on aligning cross-type dataset, which makes sense given that it is a learning based method.

We then feed the first 7 pairs of the aligned sequences using both motion estimation network and ECC into the 3D convolution network that is trained on lens tissue dataset. We also test the

Optics EXPRESS

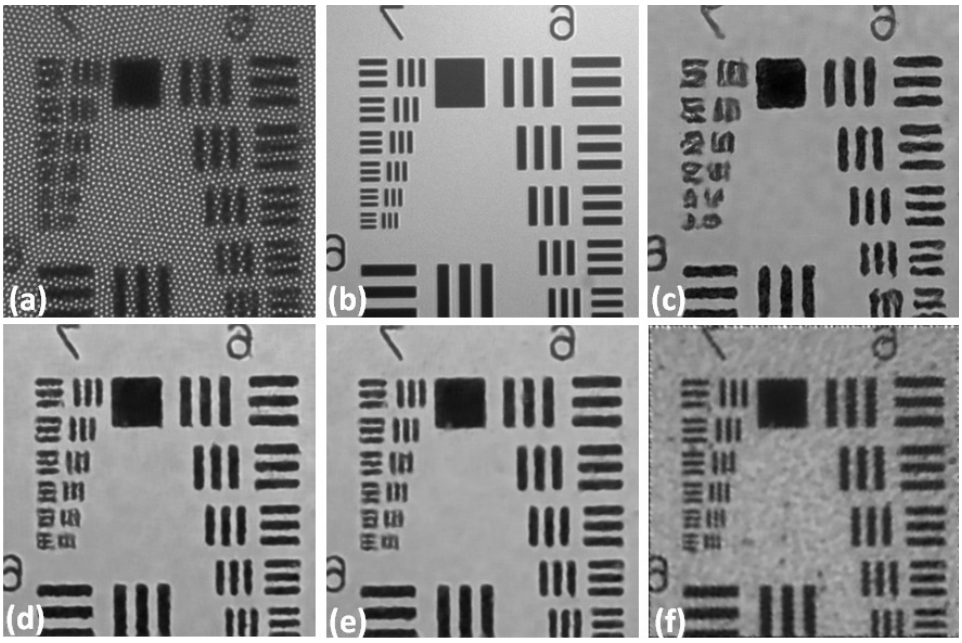

Fig. 6. Experimental results with a 1951 USAF resolution target (input sequence length 7): (a) raw FB image, (b) GT image, (c) result from GARNN, (d) result from the motion estimation network and the 3D convolution network, (e) result from the ECC image alignment and the 3D convolution network and (f) result from the MAP method.

same dataset with our GARNN method and MAP method. We present the results in Fig. 6. For the raw FB image, we can resolve up to the sixth element in the Group 6, which equals to $4.38\mu m$. For the GARNN method, there is no apparent resolution gain. For the result from the motion estimation and the 3D convolution, we can resolve up to the fourth element in the Group 7, or $2.76\mu m$. For the result from the ECC and the 3D convolution network, we can similarly resolve up to the fourth element in the Group 7, or $2.76\mu m$. For the result from the MAP method, we can resolve the sixth element in the Group 7, which equals to $2.19\mu m$. For the three neural network methods, the GARNN has no apparent resolution gain; both 3D convolution methods (motion estimation network and ECC) achieve the same resolution gains (1.6 times), which indicates that our 3D convolution network can improve the spatial resolution, with some image alignment errors. The MAP method achieves the highest resolution gain (2 times), though it has significant noise on its reconstructed image. Moreover, the MAP method is iterative and has much lower speed than the neural network methods.

Finally, we capture another USAF resolution target dataset under a scenario that is closer to practical applications. We separate the FB imaging system (Path 2) from our dual-sensor FB imaging system [3] and attach the FB probe directly to the USAF resolution target. We randomly shift the sample and capture 7 raw FB frames. We show the results from our GARNN method, 3D convolution with motion estimation network, 3D convolution with ECC and MAP method in Fig. 7. The MAP method achieves the best resolution gain (2.0 times), although with high image noise. Both 3D convolution methods (motion estimation network and ECC) also show good resolution enhancement (1.6 times), while the GARNN shows no apparent resolution gain. Based on this experiment, we suggest that our trained neural network models can be used for real applications.

Optics EXPRESS

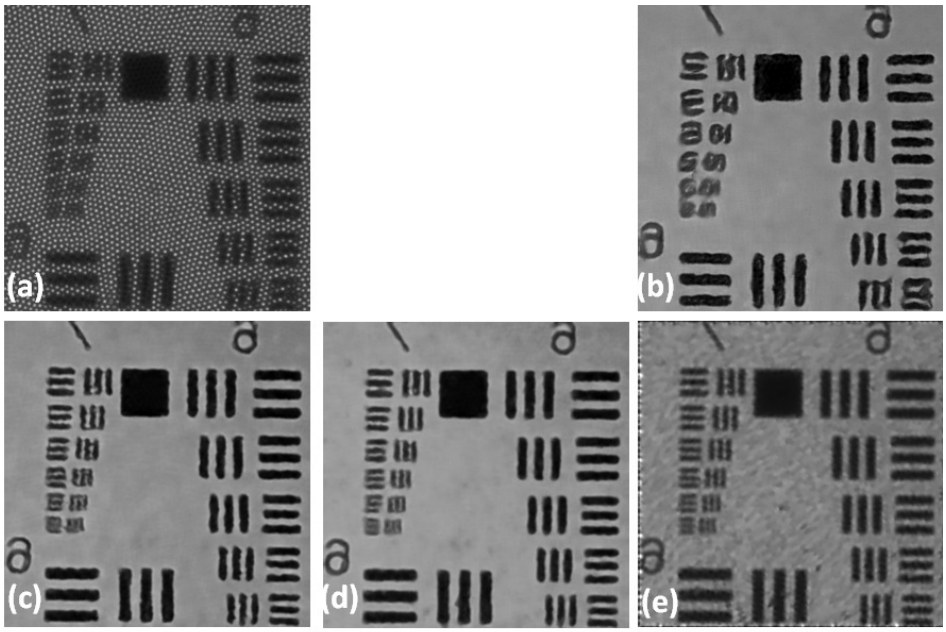

Fig. 7. Experimental results with a 1951 USAF resolution target captured by directly attaching the FB probe to the sample (input sequence length 7): (a) raw FB image, (b) result from GARNN, (c) result from the motion estimation network and the 3D convolution network, (d) result from the ECC method and the 3D convolution network and (e) result from MAP method.

7. Conclusion and future work

In conclusion, we propose a deep learning based resolution enhancement method using multi-frame FB images input. Our proposed approach first uses a motion estimation network to directly estimate unknown motions among input raw FB sequences and then uses a 3D convolution network to estimate HR images from input aligned FB images. Experiments on the lens tissue dataset and the USAF target sample show that our proposed method can significantly improve spatial resolution. Based on our previous results [3] on single FB image restoration task, we expect that our resolution enhancement method can have similar performance on biological samples.

In the future, we will continue to evaluate and optimize this method. We will conduct fine tuning by training the motion estimation network and the 3D convolution network together, for potential better performances. Also, we will collect more data from a greater variety of sample types to better train our models with the goal of transitioning this prototype method into real applications.

Funding

National Institute of Biomedical Imaging and Bioengineering (NIBIB) (R21EB022378).

References

1. M. Pierce, D. Yu, and R. Kortum, "High-resolution fiber-optic microendoscopy for in situ cellular imaging," J. Vis. Exp. **47**, 2306 (2011).
2. J. Shao, W.-C. Liao, R. Liang, and K. Barnard, "Resolution enhancement for fiber bundle imaging using maximum a posteriori estimation," Opt. Lett. **43**, 1906–1909 (2018).

Optics EXPRESS

3. J. Shao, J. Zhang, X. Huang, R. Liang, and K. Barnard, "Fiber bundle image restoration using deep learning," Opt. Lett. **44**, 1080–1083 (2019).
4. X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV),* (2017), pp. 4472–4480.
5. Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (2018), pp. 3224–3232.
6. G. D. Evangelidis and E. Z. Psarakis, "Parametric image alignment using enhanced correlation coefficient maximization," IEEE Trans. Pattern Anal. Mach. Intell. **30**, 1858–1865 (2008).
7. T. Nguyen, S. W. Chen, S. S. Shivakumar, C. J. Taylor, and V. Kumar, "Unsupervised Deep Homography: A Fast and Robust Homography Estimation Model," IEEE Robot. Autom. Lett. **3**, 2346–2353 (2018).
8. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint **arXiv:1409.1556** (2014).
9. D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," arVix preprint **arXiv:1606.03798** (2016).
10. S. Baker, A. Datta, and T. Kanade, "Parameterizing homographies," Tech. Rep. CMU-RI-TR-06-11, Carnegie Mellon University, Pittsburgh, PA (2006).
11. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint **arXiv:1412.6980** (2014).
12. J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* (2017), pp. 4778–4787.
13. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv preprint **arXiv:1502.03167** (2015).
14. V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10),* (2010), pp. 807–814.
15. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI),* (2016), pp. 265–283.
16. G. Evangelidis, "Iat: A matlab toolbox for image alignment," https://sites.google.com/site/imagealignment (2013).
17. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Image Process. **13**, 600–612 (2004).
18. H. R. Sheikh, A. C. Bovik, and G. D. Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," IEEE Trans. Image Process. **14**, 2117–2128 (2005).
19. R. Reisenhofer, S. Bosse, G. Kutyniok, and T. Wiegand, "A haar wavelet-based perceptual similarity index for image quality assessment," Signal Process. Image Commun. **61**, 33–43 (2018).