Trainable Regularization for Wiener Deconvolution

Artem Moskalev

Research advisors: prof. Anh Huy Phan, prof. S.Lefkimmiatis

Skoltech, MSc Program, Data Science

18 June 2019

State the problem I

Image restoration how it is

- Deep learning inspired image restoration approaches have established new state-of-the-art in the field
- Computational complexity limits deep NNs application in large-scale image processing
- Many problems like **video restoration** or **microscopy deconvolution** still rely on conservative (and very old) image processing algorithms [1]

[1] J. Kruse, C. Rother, U. Schmidt. "Learning to Push the Limits of Efficient FFT-Based Image Deconvolution". 2017. ICCV.

State the problem II

Microscopy deconvolution

- Such field as **microscopy deconvolution** heavily exploits non-blind deconvolution techniques (PSF could be estimated)
- For the non-blind problem formulation the following generic optimization setup is employed

$$\hat{x} = \underset{x}{argmin} \quad \underbrace{D(x, y)}_{\textbf{data fidelity}} + \underbrace{\alpha r(x)}_{\textbf{regularization}}$$

- Such approach called variational strategy and gives a rise to the most of the classical restoration algorithms

State the problem III

Generic optimization setup for deconvolution

$$\hat{x} = \underset{x}{argmin} \quad \overbrace{D(x, y)}^{\textbf{data fidelity}} + \overbrace{\alpha r(x)}^{\textbf{regularization}}$$

$$\text{where } r(x) = \sum_{d=1}^{D} \phi(G_d x)$$

- Regularization is **hand-crafted**. First or higher order differential operators for G are the common choice, and norms are used as potential function [2]
- Particular ϕ and G leads to computationally efficient restoration techniques like Wiener filter, thus suitable for large-scale image processing

State the problem IV

Generic form of Wiener filter

$$\hat{x} = \underset{x}{argmin}\ ||y - Kx||_2^2 + \frac{e^{\alpha}}{2}\sum_{d=1}^{D} ||G_d x||_2^2$$

$$= \left(K^T K + e^{\alpha}\sum_{d=1}^{D} G_d^T G_d\right)^{-1} K^T y = F^H\left(|D_K|^2 + e^{\alpha}\sum_{d=1}^{D} |D_{G_d}|^2\right)^{-1} D_K^* F y$$

Above we assumed that matrices **K** and **G** are circulant, hence diagonalizable in the DFT domain:

$$\boxed{K = F^H D_K F \qquad G_d = F^H D_{G_d} F} \qquad \boxed{D_{G_d} = F S_{G_d} P_{G_d} g_d}$$

OTF defined by kernel g

with F being the *DFT* matrix, S_{G_d}, P_{G_d} are shifting and padding operators

Stating the problem V

Generic form of Wiener filter

Generic form of Wiener filter:

$$\hat{x} = F^H \underbrace{\left(|D_K|^2 + e^{\alpha} \sum_{d=1}^{D} |D_{G_d}|^2\right)^{-1} D_K^* F y}_{\textbf{Frequency domain}}$$

- **Pros:** deconvolution in the Fourier Domain, super-fast, robust to noise
- **Cons:** prior knowledge about an image is required, α should be tuned manually, artifacts

Stating the problem V

Generic form of Wiener filter

Generic form of Wiener filter:

$$\hat{x} = F^H \underbrace{\left(|D_K|^2 + e^{\alpha} \sum_{d=1}^{D} |D_{G_d}|^2\right)^{-1} D_K^* F y}_{\textbf{Frequency domain}}$$

- **Pros:** deconvolution in the Fourier Domain, super-fast, robust to noise
- **Cons:** prior knowledge about an image is required, α should be tuned manually, artifacts
- **Proposal:** learn regularizers and α from the data using NN's **NOVELTY**

Skoltech Skolkovo Institute of Science and Technology

Aim

The overall purpose of the work is to:

- **Extend** classical **Wiener filter** with learnable regularization
- Apply it for 3D widefield **microscopy deconvolution**

Thus, we hope to combine the strength of NN's with efficiency of classical image restoration algorithms, giving the researchers a new powerful tool for large-scale image deconvolution.

Theory and algorithms I

Gradients

Derive gradients w.r.t. learned parameters to backpropagate:

$$\hat{x} = \underbrace{F^H\left(|D_K|^2 + e^{\alpha}\sum_{d=1}^{D}|D_{G_d}|^2\right)^{-1}D_K^* F y}_{\omega(y,k;\alpha,g)}$$

Recall that $D_{G_d} = F S_{G_d} P_{G_d} g_d$, hence we need to equate the following gradients:

$\boxed{\nabla_\alpha \omega(y,k;\alpha,g)}$	$\boxed{\nabla_g \omega(y,k;\alpha,g)}$	$\boxed{\nabla_y \omega(y,k;\alpha,g)}$
w.r.t. regularizing constant	**w.r.t. regularizing kernel**	**w.r.t. input**

Skoltech
Skolkovo Institute of Science and Technology

Theory and algorithms II

Gradients

Derive gradients w.r.t. learned parameters to backpropagate:

- $\frac{\partial \omega(y,k;\alpha,g)}{\partial \alpha} = -e^{\alpha} F^H \Lambda^{-2}(\alpha, g)\left[\sum_{d=1}^{D} |D_{G_d}|^2\right] D_K^* F y$
- $\frac{\partial \omega(y,k;\alpha,g)}{\partial g_j} p = -2e^{\alpha} P_j^T S_j^T F^H \left[D_{G_j} \odot Real\left(\frac{D_K^* F y}{\Lambda^2(\alpha,g)} \odot F p\right)\right]$
- $\frac{\partial \omega(y,k;\alpha,g)}{\partial y} p = F^H \Lambda^{-1}(\alpha, g) D_K F p$

where $\Lambda(\alpha, g) = |D_K|^2 + e^{\alpha} \sum_{d=1}^{D} |D_{G_d}|^2$

P_j, S_j are padding and shifting operators of j-th kernel and p is the gradient of upcoming NN's layer

Theory and algorithms III

Weights decoupling

Learned kernels should be zero-mean and fixed scale [2,3]. To ensure that we employ the following reparameterization:

$$g_j = s_j \frac{v_j - \bar{v}_j}{||v_j - \bar{v}_j||}$$

Kernel g_j consists of **magnitude** and **direction**. The gradients w.r.t. the new parameters s_j and v_j:

$$\boxed{\nabla_{s_j}\mathcal{L} = \langle \frac{g_j}{s_j}, \nabla_{g_j}\mathcal{L} \rangle} \qquad \boxed{\nabla_{v_j}\mathcal{L} = M_{v_j}\nabla_{g_j}\mathcal{L}}$$

$$M_{v_j} = \frac{s_j}{||v_j - \bar{v}_j||}\left(I - \frac{\mathbf{1}\mathbf{1}^T}{L}\right)\left(I - \frac{(v_j - \bar{v}_j)(v_j - \bar{v}_j)^T}{||v_j - \bar{v}_j||^2}\right)$$

Theory and algorithms IV

Baseline architecture of Trainable Wiener Block **(TWB)**

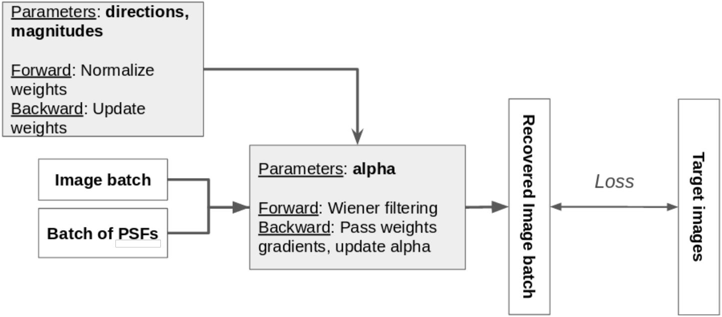
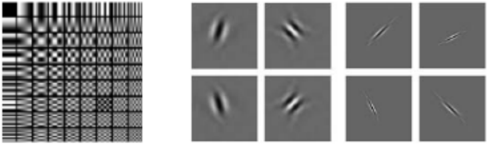


Figure: The architecture of the single Trainable Wiener Block (TWB) includes two modules (depicted by gray rectangles)

Theory and algorithms V

Weight initialization

- Random initialization:
  - **Xavier normal**: natural for NN's, not structured
- Initialization using pre-defined dictionaries:
  - **DCT**: initializing from Discrete Cosine Transform dictionary, excluding the mean part
  - **Wavelets**: specifically Gabor filters that correspond to Morlet wavelets. Used to extract frequencies in certain direction



DCT Gabor

Skoltech
Skolkovo Institute of Science and Technology

Experimental Set-Up

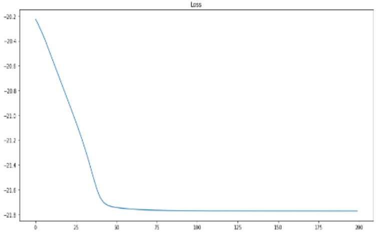
1. 2D: baseline initialization tests
2. 2D: extended model performance tests on DSBowl2018 dataset for deconvolution of noisy images
3. 2D: comparison against poplar microscopy deconvolution techniques

4. 3D: baseline initialization tests
5. 3D: extended model performance tests on Murphy's Hela dataset for deconvolution of noisy images
6. 3D: comparison against poplar microscopy deconvolution techniques

Results of the Experiment and Discussion

2D **toy** experiment: dictionary of optimal regularizers

Let's overfit the network to restore a tiny batch of images to see if it works at all:

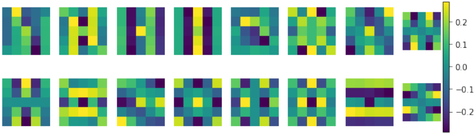


Training setup: 200 epochs, Adam, lr=0.001, NPSNR loss. Learning 24 5x5 regularizers

Results of the Experiment and Discussion

2D **toy** experiment: dictionary of optimal regularizers

The dictionary of optimal regularizers for Lena:



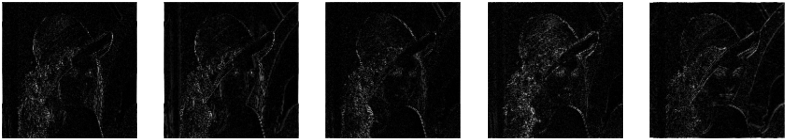


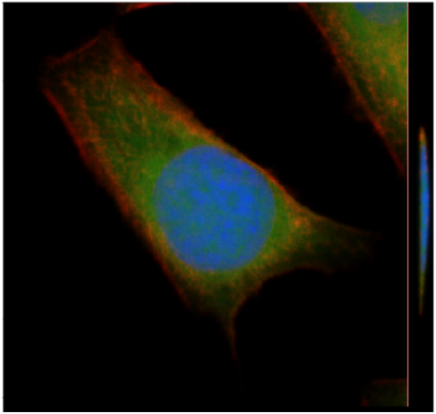
Figure: Magnitude of the Lena convolved with different learned regularizing kernels

Results of the Experiment and Discussion

3D experiment: training and validation data

Initialization & Performance test:

- Hela cells 3D fluorescence CLS microscopy images of different spatial extents, with DAPI fluorescence protocol
- Downscaled to 512x512 xy-extent, depth on z axis varies from 13 to 27
- 96 images for training, 6*5 images for validation



Skoltech
Skolkovo Institute of Science and Technology

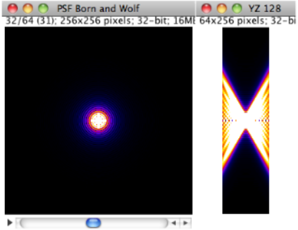
Results of the Experiment and Discussion

3D experiment: plausible PSFs

How to obtain plausible PSF for training?

Wavelength	NA	Sim. model	Spatial extent
450	1.4	GL	27,27,13
520	1.2	GL	33,33,13
630	1.4	RW	23,23,13
580	1.2	RW	19,19,13
670	1.4	BW	33,33,13
450	1.2	BW	21,21,13

EPFL PSF generator:



- GL - *Gibson&Lani* model
- RW - *Richards&Wolf* model
- BW - *Born&Wolf* model

5 kernels for training

5 kernels for validation

*Theoretical PSFs are pretty reliable and widely used by community [8]

Skoltech
Skolkovo Institute of Science and Technology

Results of the Experiment and Discussion

3D experiment: initialization

Different initialization setups for TWB on Hela3D:

Spatial extent	**Number of kernels**
3x3x3	1/15/26
5x5x3	1/15/74
5x5x5	1/15/124

Table: Top 5 initializations

rank	**setup**	**PSNR**	**CPSNR**
1	DCT 5x5x3:15	31.147	35.385
2	Gabor 3x3x3:26	31.124	35.399
3	Gabor 5x5x5:15	31.121	35.41
4	Rand 3x3x3:26	31.101	35.44
5	DCT 5x5x3:74	31.088	35.523

mean: 31.116

Table: Worst 5 initializations

rank	**setup**	**PSNR**	**CPSNR**
23	Gabor 5x5x3:1	28.477	28.5
24	Gabor 3x3x3:1	28.332	27.847
25	DCT 5x5x3:1	24.41	21.684
26	DCT 5x5x5:1	23.72	21.09
27	DCT 3x3x3:1	22.8	20.742

mean: 25.510

Results of the Experiment and Discussion

3D experiment: deconvolution of images with 1% AWGN

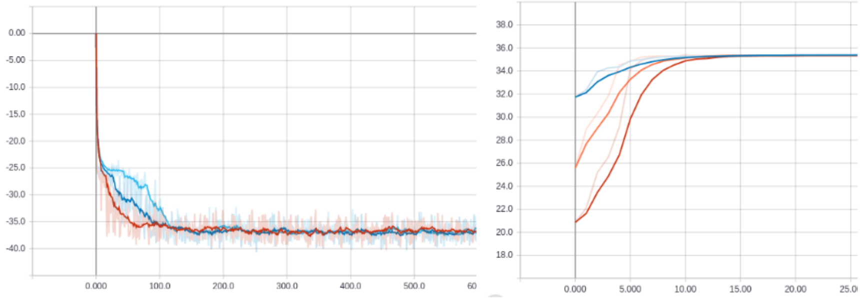


Figure: **Left:** Training with NPSNR loss, **Right:** CPSNR for validation

- Losses and validation CPSNR plateaus after 15-th epoch
- Network contains only **3421** parameters

Results of the Experiment and Discussion

3D experiment: deconvolution of images with 1% AWGN

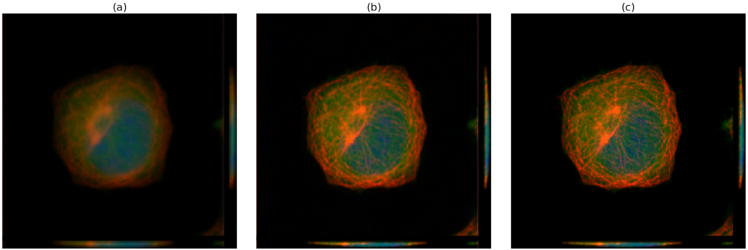


Figure: One 2D Z-slice of 3D degraded/restored/target images. (a) Degraded image SNR=3.14, **PSNR=25.67**; (b) image restored by the TWB **PSNR=34.71**; (c) undegraded reference image (target)

Results of the Experiment and Discussion

3D experiment: deconvolution of images with 3% AWGN

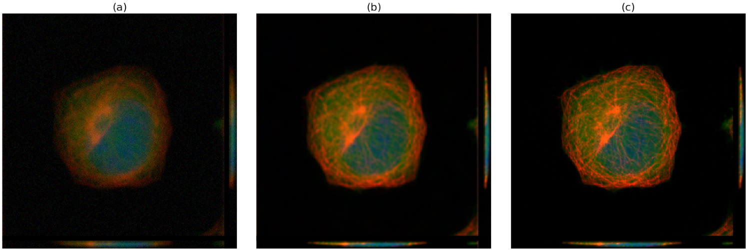


Figure: One 2D Z-slice of 3D degraded/restored/target images. (a) Degraded image SNR=1.25, **PSNR=24.93**; (b) image restored by the TWB **PSNR=30.22**; (c) undegraded reference image (target)

Results of the Experiment and Discussion

Comparison against other methods for 3D microscopy deconvolution

Table: TWB compared with other deconvolution algorithms on Hela3D dataset corrupted with 1% AWGN

model	PSNR (dB)	CPSNR (dB)	ISNR (dB)	CISNR (dB)	time (sec.)
TWB	**31.308**	35.394	**4.258**	**9.352**	3.186
Tikhonov-Miller	31.168	32.64	4.118	6.599	113.726
TWB_C	31.114	35.366	4.064	9.325	3.2
TWB_M	31.099	**35.403**	4.048	9.361	3.2
Isotropic TV	30.732	32.771	3.682	6.729	223.1
RIF	30.336	32.886	3.286	6.845	0.239
Wavelet Thresholding	29.779	30.759	2.728	4.718	171.9

- **trainable regularization** (proposed method)
- Second best method (Tikhonov-Miller) is 37x timex slower

Skoltech
Skolkovo Institute of Science and Technology

Results of the Experiment and Discussion

Comparison against other methods for 3D microscopy deconvolution

Table: TWB compared with other deconvolution algorithms on Hela3D dataset corrupted with 3% AWGN

model	**PSNR (dB)**	**CPSNR (dB)**	**ISNR (dB)**	**CISNR (dB)**	**time (sec.)**
Isotropic TV	**29.969**	31.034	**3.945**	6.202	179.05
TWB_C	29.784	32.665	3.761	7.833	3.191
TWB	29.773	**32.713**	3.749	**7.881**	3.17
TWB_M	29.712	32.584	3.688	7.752	3.17
RIF	29.612	31.654	3.588	6.823	0.168
Tikhonov-Miller	28.987	29.6	2.963	4.769	74.71
Wavelet Thresholding	28.813	28.823	2.789	3.992	134.44

- **trainable regularization** **pre-defined regularization**
- Isotropic TV is almost 60x times slower

Results of the Experiment and Discussion

Conclusions

- Proposed method performs competitively in comparison with other popular deconvolution techniques
- The proposed method is one-shot and relatively fast in comparison with iterative deconvolution approaches
- Proposed method contains few trainable parameters in comparison with deep neural networks (thousands vs millions)
- Due to backpropagation in the Fourier domain, the network is computationally efficient and suits for large-scale image processing (much faster than deep NNs)
- In contrast to the classical deconvolution approaches proposed method requires no parameter tuning

Results of the Experiment and Discussion

Why not compare with recent DL based methods?

- 3D multichannel microscopy deconvolution is rather untouched area for most of the DL based methods
- No 3D multichannel deconvolution benchmarks exist
- Most of the DL based approaches tackle only the problem of 2D grayscale image restoration
- 3D convolution is expensive!

Relatively lightweight DL models for non-blind & blind deconvolution:

1. Fourier Deconvolutional Network (FDN) [1]
2. Super-resolution CNN (SRCNN) [9]

Memory error at Tesla K80 during inference with batch size of 1

References

[1] Kruse, C. Rother, U. Schmidt. "Learning to Push the Limits of Efficient FFT-Based Image Deconvolution". 2017. ICCV.
[2] S. Lefkimmiatis. "Universal Denoising Networks : A Novel CNN Architecture for Image Denoising". 2017. CVPR.
[3] Tim Salimans, Diederik P. Kingma. "Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks". 2016. arXiv preprint arXiv:1602.07868.
[4] D. Martin and C. Fowlkes and D. Tal and J. Malik. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. 2001. Proc. 8th Int'l Conf. Computer Vision.
[5] Rolf Köhler et al. "Recording and playback of camera shake: benchmarking blind deconvolution with a real-world database". 2012. ECCV.
[6] 2018 Data Science Bowl: Find the nuclei in divergent images to advance medical discovery. kaggle.com/c/data-science-bowl-2018
[7] M. Velliste and R.F. Murphy. Automated Determination of Protein Subcellular Locations from 3D Fluorescence Microscope Images. Proceedings of the 2002 IEEE International Symposium on Biomedical Imaging (ISBI 2002), pp. 867-870.
[8] H. Kirshner, F. Aguet, D. Sage, M. Unser. 3-D PSF Fitting for Fluorescence Microscopy. Journal of Microscopy, vol. 249, 2013.

[9] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. ECCV pages 184–199. Springer. 2014.