Application of Proximal Algorithms to Three Dimensional Deconvolution Microscopy

Paroma Varma
Stanford University
paroma@stanford.edu

Abstract

In microscopy, shot noise dominates image formation, which can be modeled as a Poisson process. The Richardson-Lucy method tends to converge slowly for large problems and is not flexible to the addition of non-differentiable priors. In this project, proximal algorithms like ADMM and Chambolle-Pock are applied to three dimensional deconvolution and are shown to converge faster than Richardson-Lucy.

1. Introduction

Three dimensional (3D) deconvolution is used to reduce the out-of-focus light in 3D microscopy data. The pattern of the out-of-focus light is determined by the the microscope's point spread function (PSF), the image of a point on the sensor. Since the deconvolution is applied to a 3D dataset, the PSF is 3D as well.

1.1. Image Formation

The 3D captured dataset, $M(r)$, is created by recording optical sections of the three dimensional volume at different focal planes. Let $PSF(r)$ be the 3D PSF associated with the microscope system. Assuming linearity, the relationship between $M(r)$ and the 3D sharp object, $O(r)$, can be represented as a convolution operation:

$$M(r) = \int O(s)PSF(r-s)ds \tag{1}$$

$$= O(r) \otimes PSF(r) \tag{2}$$

1.2. Convolution as Linear Model

We represent each of $M(r), O(r), PSF(r)$ as 3D matrices $\mathbf{b}, \mathbf{x}, \mathbf{c}$, respectively, since the captured data is discrete. The convolution operation can now be represented as

$$\mathbf{b} = \mathbf{c} * \mathbf{x} \tag{3}$$

Furthermore, the convolution kernel, $\mathbf{c}$ can be represented as a block-circulant matrix $\mathbf{A}$ such that

$$\mathbf{b} = \mathbf{A}\mathbf{x} \tag{4}$$

In microscopy, the main sources of noise are detection, emission and re-emission, all of which can be modeled as a Poisson process. Therefore, the image formation can be modeled as

$$\mathbf{b} = \mathcal{P}(\mathbf{A}\mathbf{x}) \tag{5}$$

where $\mathcal{P}$ represents a Poisson process.

2. Proximal Methods

Proximal methods are a class of iterative algorithms to solve non-smooth, large scale, multi-dimensional problems [4]. Each iteration evaluates a proximal operator, which is a simple convex problem that can be solved quickly. The main advantages of using proximal methods for deconvolution are that (1) the problem can be solved with fewer iterations than Richardson-Lucy [3], [5] and (2) it is easier to include non-differentiable priors since its proximal operator is always convex.

2.1. Standard Form for Deconvolution

The standard form for proximal problems is:

$$\underset{x}{\text{minimize}} \quad f(\mathbf{x}) + g(\mathbf{z}) \tag{6}$$

$$\text{subject to} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c} \tag{7}$$

The deconvolution problem can be posed as a combination of minimizing the negative log-likelihood of the Poisson noise term and a non-negativity constraint. The constraints are added in to match the standard form of proximal problems:

$$\underset{x}{\text{minimize}} \quad -\log(p(\mathbf{b}|\mathbf{z_1})) + \mathcal{I}_{\mathbb{R}+}(\mathbf{z_2}) \tag{8}$$

$$\text{subject to} \quad \underbrace{\begin{bmatrix}\mathbf{A}\\ \mathbf{I}\end{bmatrix}}_{\mathbf{K}}\mathbf{x} - \underbrace{\begin{bmatrix}\mathbf{z_1}\\ \mathbf{z_2}\end{bmatrix}}_{\mathbf{z}} = 0 \tag{9}$$

The addition of the total variation (TV) prior can be represented in the above form as follows:

$$\underset{x}{\text{minimize}} \quad -\log(p(\mathbf{b}|\mathbf{z_1})) + \mathcal{I}_{\mathbb{R}+}(\mathbf{z_2}) \tag{10}$$

$$+ \quad \lambda||\mathbf{z_3}||_1 + \lambda||\mathbf{z_4}||_1 + \lambda||\mathbf{z_5}||_1 \tag{11}$$

$$\text{subject to} \quad \underbrace{\begin{bmatrix}\mathbf{A}\\ \mathbf{I}\\ \mathbf{D_x}\\ \mathbf{D_y}\\ \mathbf{D_z}\end{bmatrix}}_{\mathbf{K}}\mathbf{x} - \underbrace{\begin{bmatrix}\mathbf{z_1}\\ \mathbf{z_2}\\ \mathbf{z_3}\\ \mathbf{z_4}\\ \mathbf{z_5}\end{bmatrix}}_{\mathbf{z}} = 0 \tag{12}$$

$\mathbf{z_3}, \mathbf{z_4}, \mathbf{z_5}$ are used to represent the anisotropic gradient in the x,y,and z dimensions, respectively.

2.2. Alternate Direction of Multipliers Method

The ADMM algorithm [1], as discussed in class, can be applied to 3D data. The updates would remain the same, except the variables would have an additional dimension. To review, the iterative algorithm with no prior would be

$$\text{Initialization: } \mathbf{x}^0 = \mathbf{b}, \rho > 0 \tag{13}$$

$$\text{for k = 1 to MaxIter} \tag{14}$$

$$\mathbf{x} \leftarrow \mathbf{prox}_{quad}(\mathbf{v}), \mathbf{v} = \mathbf{z} - \mathbf{u} \tag{15}$$

$$\mathbf{z}_1 \leftarrow \mathbf{prox}_{\mathcal{P},\rho}(\mathbf{v_1}), \mathbf{v_1} = \mathbf{A}\mathbf{x} + \mathbf{u_1} \tag{16}$$

$$\mathbf{z}_2 \leftarrow \mathbf{prox}_{\mathcal{I},\rho}(\mathbf{v_2}), \mathbf{v_2} = \mathbf{x} + \mathbf{u_2} \tag{17}$$

$$\mathbf{u} \leftarrow \mathbf{u} + \mathbf{K}\mathbf{x} - \mathbf{z} \tag{18}$$

$$\text{end} \tag{19}$$

With the addition of the TV prior, the algorithm can be appended and written as

$$\text{Initialization: } \mathbf{x}^0 = \mathbf{b}, \rho, \lambda > 0 \tag{20}$$

$$\text{for k = 1 to MaxIter} \tag{21}$$

$$\mathbf{x} \leftarrow \mathbf{prox}_{quad}(\mathbf{v}), \mathbf{v} = \mathbf{z} - \mathbf{u} \tag{22}$$

$$\mathbf{z}_1 \leftarrow \mathbf{prox}_{\mathcal{P},\rho}(\mathbf{v_1}), \mathbf{v_1} = \mathbf{A}\mathbf{x} + \mathbf{u_1} \tag{23}$$

$$\mathbf{z}_2 \leftarrow \mathbf{prox}_{\mathcal{I},\rho}(\mathbf{v_2}), \mathbf{v_2} = \mathbf{x} + \mathbf{u_2} \tag{24}$$

$$\mathbf{z}_3 \leftarrow \mathbf{prox}_{||\cdot||_1,\rho}(\mathbf{v_3}), \mathbf{v_3} = \mathbf{D_x}\mathbf{x} + \mathbf{u_3} \tag{25}$$

$$\mathbf{z}_4 \leftarrow \mathbf{prox}_{||\cdot||_1,\rho}(\mathbf{v_4}), \mathbf{v_4} = \mathbf{D_y}\mathbf{x} + \mathbf{u_4} \tag{26}$$

$$\mathbf{z}_5 \leftarrow \mathbf{prox}_{||\cdot||_1,\rho}(\mathbf{v_5}), \mathbf{v_5} = \mathbf{D_z}\mathbf{x} + \mathbf{u_5} \tag{27}$$

$$\mathbf{u} \leftarrow \mathbf{u} + \mathbf{K}\mathbf{x} - \mathbf{z} \tag{28}$$

$$\text{end} \tag{29}$$

In both cases, each variable is updated by evaluating a proximal operator, which is a convex problems usually with an analytic solution. (proximal operators and their analytic solutions are discussed in section 2.5).

2.3. Chambolle-Pock Method

Chambolle-Pock is another algorithm based on the use of proximal operators that is well-suited for solving non-smooth convex problems [2]. The primal-dual algorithm is applicable to this problem as it has been shown to converge efficiently for image denoising and image deconvolution problems in 2D.

Given the deconvolution problem in standard proximal form, the iterative updates for Chambolle Pock are as follows:

$$\text{Initialization: } \bar{\mathbf{x}}^0 = \mathbf{x}^0 = \mathbf{b}, \sigma\tau < ||\mathbf{K}||_2^2 \tag{30}$$

$$\text{for k = 1 to MaxIter} \tag{31}$$

$$\mathbf{z}' \leftarrow \mathbf{z} + \sigma\mathbf{K}\bar{\mathbf{x}} \tag{32}$$

$$\mathbf{z}_1 \leftarrow \mathbf{z}'_1 - \sigma\mathbf{prox}_{\mathcal{P},\sigma}(\mathbf{z}'_1/\sigma) \tag{33}$$

$$\mathbf{z}_2 \leftarrow \mathbf{z}'_2 - \sigma\mathbf{prox}_{\mathcal{I},\sigma}(\mathbf{z}'_2/\sigma) \tag{34}$$

$$\mathbf{x} \leftarrow \mathbf{x} - \tau\mathbf{K}^T\mathbf{z} \tag{35}$$

$$\bar{\mathbf{x}} \leftarrow 2\mathbf{x} - \mathbf{x}_{prev} \tag{36}$$

$$\text{end} \tag{37}$$

The addition of the TV prior would lead to the following iterative updates:

$$\text{Initialization: } \bar{\mathbf{x}}^0 = \mathbf{x}^0 = \mathbf{b}, \sigma\tau < ||\mathbf{K}||_2^2, \lambda > 0 \tag{38}$$

$$\text{for k = 1 to MaxIter} \tag{39}$$

$$\mathbf{z}' \leftarrow \mathbf{z} + \sigma\mathbf{K}\bar{\mathbf{x}} \tag{40}$$

$$\mathbf{z}_1 \leftarrow \mathbf{z}'_1 - \mathbf{prox}_{\mathcal{P},\sigma}(\mathbf{z}'_1/\sigma) \tag{41}$$

$$\mathbf{z}_2 \leftarrow \mathbf{z}'_2 - \mathbf{prox}_{\mathcal{I},\sigma}(\mathbf{z}'_2/\sigma) \tag{42}$$

$$\mathbf{z}_{3,4,5} \leftarrow \mathbf{z}'_{3,4,5} - \mathbf{prox}_{||\cdot||_1,\sigma}(\mathbf{z}'_{3,4,5}/\sigma) \tag{43}$$

$$\bar{\mathbf{x}} \leftarrow \mathbf{x} - \tau\mathbf{K}^T\mathbf{z} \tag{44}$$

$$\bar{\mathbf{x}} \leftarrow 2\mathbf{x} - \mathbf{x}_{prev} \tag{45}$$

$$\text{end} \tag{46}$$

The original Chambolle-Pock method has another variable, θ that appears in the update for $\bar{\mathbf{x}}^{k+1}$ as

$$\bar{\mathbf{x}}^{k+1} \leftarrow \mathbf{x}^{k+1} - \theta(\mathbf{x}^k - \mathbf{x}^{k+1}) \tag{47}$$

However, it is shown that the problem converges for the choice of $\theta = 1$ [2], which is what is chosen in this case.

2.3.1 Estimating Spectral Norm

The spectral norm of $\mathbf{K}$ is needed to choose the parameters σ and τ. In the 2D case, the spectral norm is easily

calculated as the maximal singular value of the matrix $\mathbf{K}$. However, in the 3D case, calculating the spectral norm can be computationally expensive. Thus, an upper bound for $||\mathbf{K}||_2$ is calculated analytically. In the case with no priors, $\mathbf{K} = \begin{bmatrix}\mathbf{A}\\ \mathbf{I}\end{bmatrix}$. We can use the Cauchy-Schwartz inequality to attain an upper bound:

$$||\mathbf{K}||_2 \leq ||\mathbf{A}||_2 + ||\mathbf{I}||_2 \tag{48}$$

Trivially, $||\mathbf{I}||_2 = 1$. Since $\mathbf{A}$ is a circulant, symmetric matrix, it's eigendecomposition can be written as:

$$\mathbf{A} = \mathbf{F^{-1}DF} \tag{49}$$

where $\mathbf{D} = diag(\hat{\mathbf{c}})$. $\mathbf{F}$ and $\mathbf{F^{-1}}$ are the forward and inverse Fourier transform matrices, respectively. $\hat{\mathbf{c}}$ represents the coefficients of the Fourier transform of the PSF defined by $\mathbf{A}$. Since the PSF is normalized across the 3D volume, we know that the maximum value of $\hat{\mathbf{c}}$ is $\frac{1}{xDim \cdot yDim \cdot zDim}$, where $xDim, yDim, zDim$ are the x, y, and z dimensions of the PSF stack.

Combining the above information with Cauchy-Schwartz,

$$||\mathbf{K}||_2 \leq \frac{1}{xDim \cdot yDim} + 1 \tag{50}$$

One of the drawbacks of this method is that it is difficult to analytically estimate the spectral norm of all matrices in such a manner. As priors are added to the deconvolution objective function, it might not be possible to estimate an upper bound easily.

2.3.2 Choosing Parameters for Chambolle-Pock

The parameters σ and τ need to be chosen so that $\sigma\tau < ||\mathbf{K}||_2^2$. Given we are overestimating $||\mathbf{K}||_2$, we let $\sigma = \tau = 0.9/||\mathbf{K}||_2^2$, which ensures that the parameters guarantee convergence. This is possible only because $||\mathbf{K}||_2$ is overestimated - if the values for σ and τ are too large, the algorithm is no longer guaranteed to converge. On the other hand, if $||\mathbf{K}||_{2,est} >> ||\mathbf{K}||_2$, the algorithm will converge very slowly and will lose its advantage over iterative methods like Richardson-Lucy.

3. Proximal Operators

The proximal operators used in Chambolle-Pock are similar to those in ADMM, and all have analytic solutions as described in this section.

3.1. Maximum Likelihood Estimator Subproblem $\mathbf{prox}_{\mathcal{P},\sigma}(\mathbf{x})$

We can calculate the proximal operator for the MLE in the same manner as for ADMM. The analytic solution can be applied per pixel and therefore does not require solving a convex problem. In this case, the operator can be defined as

$$\mathbf{prox}_{\mathcal{P},\sigma}(\mathbf{x}) = -\frac{1-\sigma\mathbf{x}}{2\sigma} + \sqrt{(\frac{1-\sigma\mathbf{x}}{2\sigma})^2 + \frac{\mathbf{b}}{\sigma}} \tag{51}$$

3.2. Indicator Subproblem $\mathbf{prox}_{\mathcal{I},\sigma}$

Again, similar to the ADMM problem, this proxinal operator also has a analytic solution, which comprises of setting the negative values of the variable $\mathbf{x}$ to 0.

$$\mathbf{prox}_{\mathcal{I},\sigma}(\mathbf{x}) = \max\,(\mathbf{x}, 0) \tag{52}$$

3.3. l_1 Norm Subproblem $\mathbf{prox}_{||\cdot||_1,\sigma}$

The proximal operator for the l_1 norm can be analytically represented as an element-wise soft thresholding operator:

$$\mathbf{prox}_{||\cdot||_1,\sigma}(\mathbf{x}) = \mathcal{S}_{\lambda/\sigma}(\mathbf{x}) = (\mathbf{x} - \lambda/\sigma)_+ - (-\mathbf{x} - \lambda/\sigma)_+ \tag{53}$$

Like the previous proximal operators, there is no need to solve a convex optimization problem, which means the proximal expressions can be solved efficiently.

4. Simulated Data

The ground 3D volume was created by placing the same resolution bar pattern at different depths. The simulated captured stack was created by convolving this 3D volume with a theoretical PSF with the following parameters:

Parameter	Value
NA	0.5
Magnification	10
Sensor Size	256x256
Pixel Width	6.5um
Wavelength	515nm
z Range	-100um:10um:100um

500 iterations of each algorithm (Richardson Lucy, ADMM, and Chambolle-Pock) were used to deconvolve the simulated captured stack. Figure 1 shows the maximum intensity projection across different dimensions of the deconvolved stack for each of the three methods. After the same number of iterations, Chambolle Pock achieves the highest Peak Signal to Noise Ratio (PSNR) in this case. The larger resolution bars have fewer artifacts in the Chambolle-Pock solution than in the Richardson-Lucy deconvolution solution. This lack of artifacts can also be seen in the y-z and x-z maximum intensity projections, where the horizontal and vertical lines represent the z-slices where the resolution test chart was placed in the original volume.

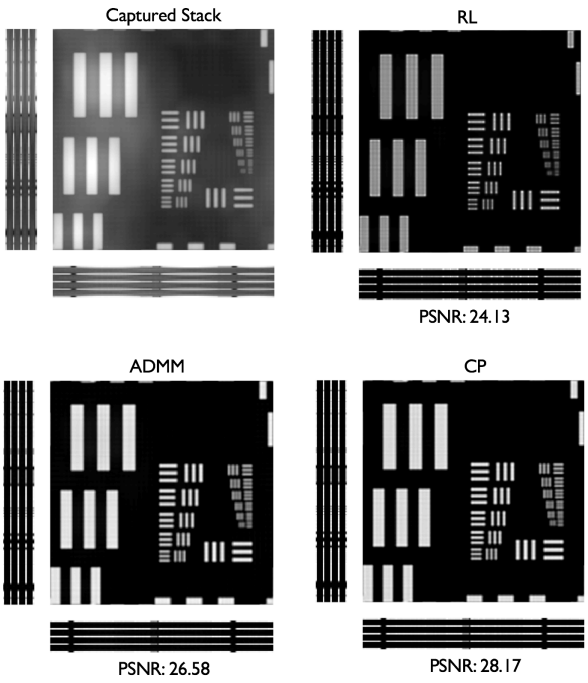

Figure 1. Maximum intensity projections of simulated captured stack, and stacks deconvolved using Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP) with final PSNR values after 500 iterations of each method.

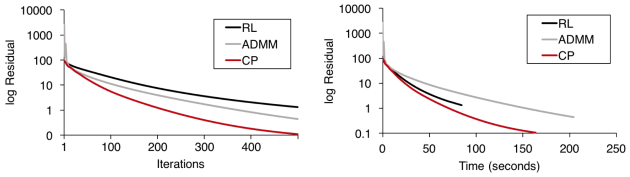

Figure 2. Convergence trends using log residuals ($\log(||Ax-b||_2^2)$) for Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP) against number of iterations and time in seconds for the simulated, no prior case.

Figure 2 shows the convergence trends for each of the algorithms for the log residual, $\log(||Ax - b||_2^2)$. When the log residuals are compared against number of iteration, Chambolle-Pock clearly performs better than both Richardson-Lucy and ADMM. After 500 iterations, the residual value for Chambolle-Pock is around 10x smaller compared to ADMM and 100x smaller compared to Richardson-Lucy.

When log residual is compared to time in seconds, Chambolle-Pock still converges to a smaller residual faster than Richardson Lucy and ADMM, even though Richardson Lucy takes less time to run per iteration. As this point, neither ADMM nor Chambolle-Pock were optimized for runtime.

4.1. Artifacts in ADMM Convergence

The ADMM algorithm shows some spiking in the log residual plot in the first few iterations. This spiking could be mitigated by choosing a larger ρ parameter. However, that makes the final residual value higher. Since the problem is a convex problem and each proximal operator analytically solves a convex problem as well, convergence is guaranteed after N iterations, which makes it safe to choose a small ρ parameter.

However, to determine the number N would require knowledge of the noise characteristics of the sample, which are usually unknown. It could be hypothesized that as noise increases, so does the number of iterations required to guarantee convergence.

5. Captured Data

The 3D captured volume was a focal stack of pollen grain, which was created by imaging the 3D sample at different depths using a conventional widefield setup. The microscope setup (and the generated theoretical PSF) had the following parameters:

Parameter	Value
NA	0.5
Magnification	10
Sensor Size	2048x2048
Pixel Width	6.5um
Wavelength	515nm
z Range	-39um:3um:39um

The images in the results are based on a 400 x 400 cropped section of the original image.

5.1. Deconvolution with No Prior

100 iterations of each algorithm (Richardson Lucy, ADMM, and Chambolle-Pock) were used to deconvolve the captured stack. Figure 3 shows the maximum intensity projection across different dimensions of the deconvolved stack for each of the three methods. The ADMM result looks like it preserves more details than the Richardson-Lucy result. It is difficult to estimate whether the details inside the pollen grain structure are high frequency structures or noise that was not removed during the deconvolution process.

In this case, the spikes in the residual value for ADMM are more apparent, which is due to the higher noise content in the images.

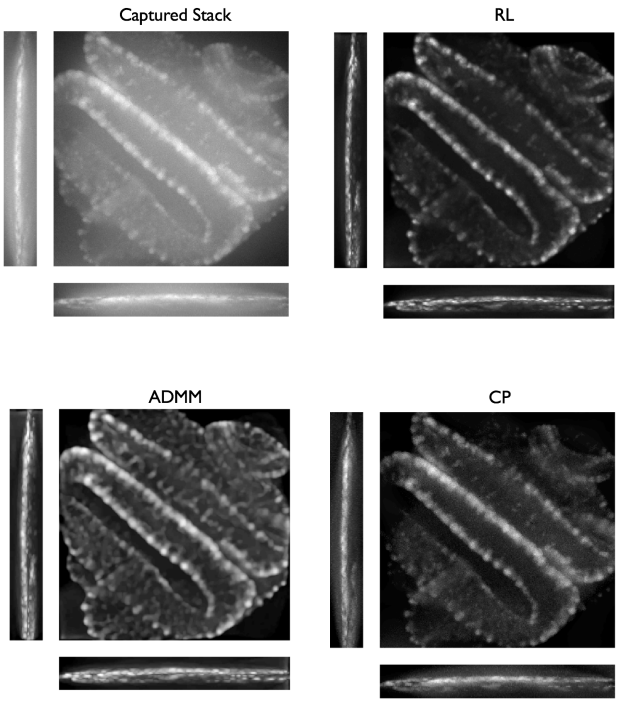

Figure 3. Maximum intensity projections of captured stack, and stacks deconvolved using Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP).

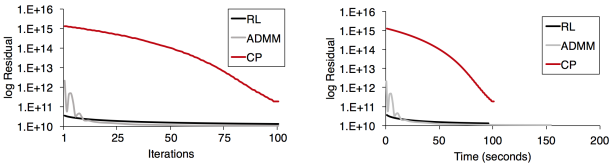

Figure 4. Convergence trends using log residuals ($\log(||Ax-b||_2^2)$) for Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP) against number of iterations and time in seconds for the simulated, no prior case.

5.2. Convergence with Chambolle-Pock

The Chambolle-Pock result looks noisier than the others and does not preserve the details inside the pollen grain. Looking at Figure 4, the Chambolle-Pock convergence behaves very differently from Richardson-Lucy and ADMM. Moreover, it does not reach a residual value close to that of ADMM, the fastest algorithm in this case.

A possible explanation for the slow convergence of Chambolle-Pock in this case is the overestimation of $||\mathbf{K}||_2$. Since the value of $||\mathbf{K}||_2$ used for the algorithm was significantly larger than the true value, the convergence rate of Chambolle-Pock was much slower than expected. A better estimate of $||\mathbf{K}||_2$ could be calculated computationally using the Lanczos method for each dataset and each prior used, since $\mathbf{K}$ would change.

The Chambolle-Pock method could also be more sensitive to errors in matrix $\mathbf{K}$ than Richardson-Lucy and ADMM. With the captured dataset, it is difficult to estimate the PSF perfectly, therefore introducing errors in the $\mathbf{A}$ matrix. The PSF could be better estimated by accounting for the aberrations in the system and correcting for them by changing the PSF accordingly.

5.3. Deconvolution with TV Prior

100 iterations of each algorithm (Richardson Lucy, ADMM, and Chambolle-Pock) were used with a TV prior ($\lambda = 0.005$) to deconvolve the captured stack. Figure 5 shows the maximum intensity projection across different dimensions of the deconvolved stack for each of the three methods.

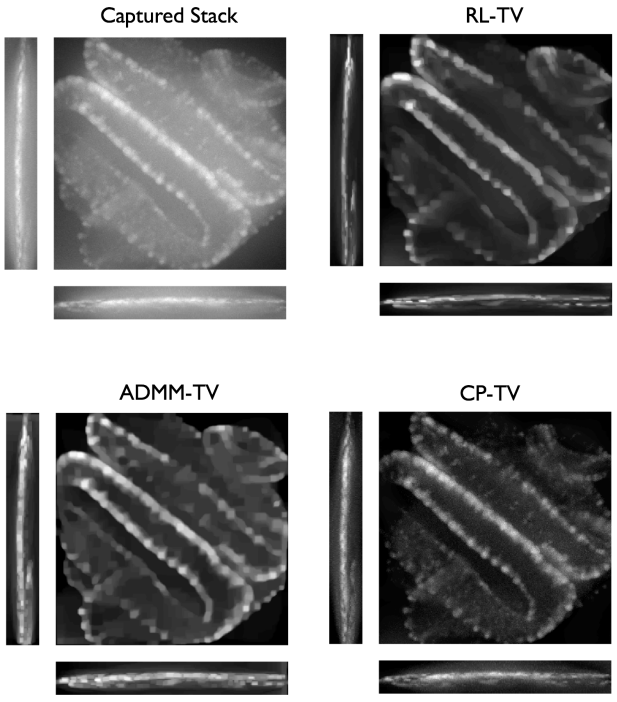

Figure 5. Maximum intensity projections of captured stack, and stacks deconvolved using Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP) with a TV prior.

With a sample like the pollen grain, addition of the TV prior makes the resulting image look patchy. If λ is reduced, the result matches the one without a prior. Therefore, with this sample, a TV prior was added to show that the proximal ADMM algorithm still performs better than Richardson-Lucy in terms of convergence.

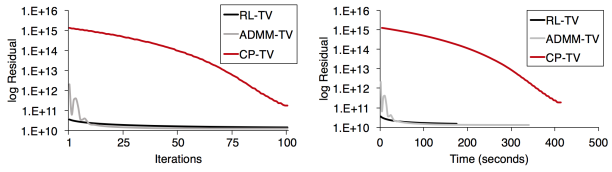

Figure 6. Convergence trends using log residuals ($\log(||Ax-b||_2^2)$) for Richardson-Lucy(RL), ADMM, and Chambolle-Pock(CP) with TV prior against number of iterations and time in seconds for the simulated.

Figure 6 shows a similar convergence pattern as with convergence without a prior, where ADMM performs the best across iterations as well as across time.

6. Discussion

Overall, proximal algorithms converge faster than Richardson-Lucy both in the case with and without a prior. However, these proximal methods can be better tuned per sample in order to improve their convergence.

For ADMM, the choices of the ρ parameter could be explored in order to find a value that performs well in terms of the final residual value and has a smooth convergence curve.

For Chambolle-Pock, there are various ways to improve the estimate of $||\mathbf{K}||_2$ and thus improve chosen values for σ and τ. Moreover, the σ and τ values were kept equal to each other, and optimizing those values could also improve the convergence for the algorithm. The θ value was also kept a constant, and it is possible to look further into the convergence analysis for different values of θ.

For both proximal algorithms, the variable updates could be implemented more efficiently to improve the runtime. There are multiple forward and inverse Fourier transforms that are implemented for the analytic solutions of the proximal operators, which could be implemented without computationally expensive operators for certain priors.

7. Acknowledgments

Thanks to Gordon Wetzstein for guiding this course project. I would also like to thank Julie Chang for captured focal stacks and advice regarding deconvolution and Steven Diamond and Felix Heide for helping with adapting the Chambolle-Pock method for deconvolution.

References

[1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[2] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[3] L. B. Lucy. An iterative technique for the rectification of observed distributions. *The astronomical journal*, 79:745, 1974.

[4] N. Parikh and S. P. Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 1(3):127–239, 2014.

[5] W. H. Richardson. Bayesian-based iterative method of image restoration*. *JOSA*, 62(1):55–59, 1972.