Investigating Image Inpainting via the Alternating Direction Method of Multipliers (ADMM)

Jonathan Tuck
Stanford University

Motivation

- In many imaging applications, there exists potential for **corruption of the images** by sources of noise that **completely lose original pixel information**, such as degradation over time [1]:

Corrupted Image

Inpainted Image

- **Image inpainting** allows for **the estimation and restoration of missing pixels**
- The inpainting process leaves some artifacts behind and is usually not perfect
- Accuracy of the inpainting is related to the prior used

ADMM & Image Inpainting

- The Alternating Direction Method of Multipliers (ADMM) [2] solves the following optimization problem:

$$\underset{x}{\text{minimize}} \quad f(\mathbf{x}) + g(\mathbf{z})$$
$$\text{subject to} \quad \mathbf{Ax} + \mathbf{Bz} = \mathbf{c}$$

- ADMM is **general**, **simple**, and **parallelizable** [2]
- Solved using the following update procedure [2] [3]:

$$\mathbf{x}^{k+1} := \underset{x}{\text{argmin}}\, L_\rho(\mathbf{x}, \mathbf{z}^k, \mathbf{y}^k) = \underset{f,\rho}{\text{prox}}(\mathbf{z}^k - (1/\rho)\mathbf{y}^k) = \underset{f,\rho}{\text{prox}}(\mathbf{v})$$
$$\mathbf{z}^{k+1} := \underset{z}{\text{argmin}}\, L_\rho(\mathbf{x}^{k+1}, \mathbf{z}, \mathbf{y}^k) = \underset{g,\rho}{\text{prox}}(\mathbf{Ax}^{k+1} + (1/\rho)\mathbf{y}^k) = \underset{g,\rho}{\text{prox}}(\mathbf{v})$$
$$\mathbf{y}^{k+1} := \mathbf{y}^k + \rho(\mathbf{Ax}^{k+1} + \mathbf{Bz}^{k+1} - \mathbf{c})$$

- Image inpainting can be formulated in the ADMM framework:

$$\underset{x}{\text{minimize}} \quad \|\mathbf{Kx} - \mathbf{b}\|_2^2 + \lambda\Gamma(\mathbf{z})$$
$$\text{subject to} \quad \mathbf{x} - \mathbf{z} = \mathbf{0}$$

- **b** is observed, **K** is the diagonal inpainting mask, and **Γ(z) is the prior information on the image**

Prior Selection

- Total Variation [4] $\quad \Gamma(z) = \lambda\|Dz\|_1$
  - Used when image sparse gradients
  - Makes use of finite difference matrix **D**

$$Z(i) = \sum_j \exp\frac{\|v(N_i) - v(N_j)\|_{2,a}^2}{h^2}$$

- Non-Local Means [5]
  - Used when image has self-similar structure
  - Weighted sum of differences in neighborhoods

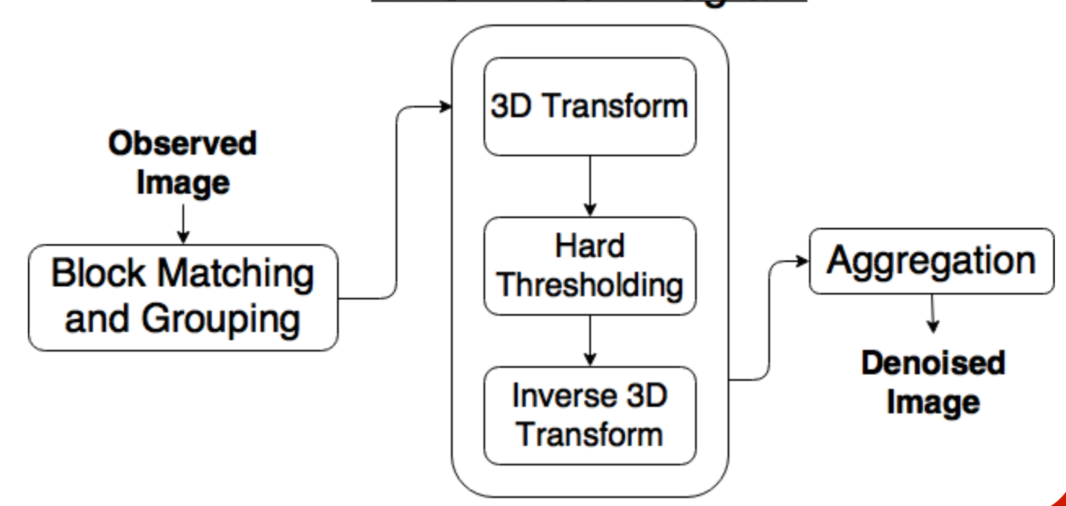
$$w(i,j) = \frac{1}{Z(i)}\exp\frac{\|v(N_i) - v(N_j)\|_{2,a}^2}{h^2}$$
$$\Gamma(i) = \sum_{j\in I} w(i,j)v(j)$$

- Recursive Filter [6]
  - Takes an image, transforms it into a new domain (right), filters the transformed signal with a Gaussian filter, and inverts the transformation.

$$T(u) = \int_0^u \left(1 + \frac{\sigma_s}{\sigma_r}\sum_{k=1}^{c}\left|\frac{dX_k}{dt}\right|\right) dt$$

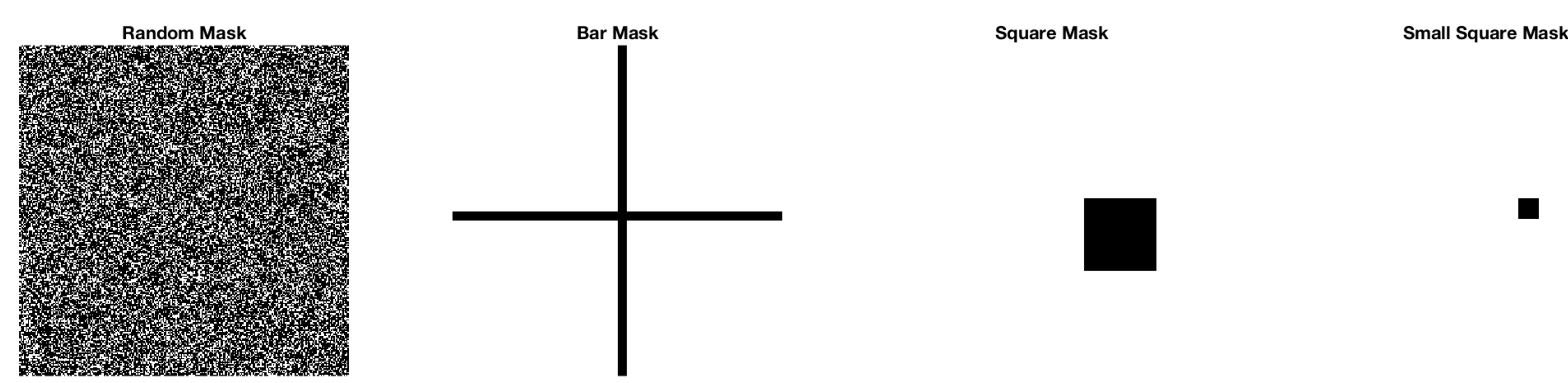
- BM3D [7]
  - Used when image has a locally sparse transform
  - Performs block matching, denoising in 3D transform domain

BM3D Block Diagram



Methodology and Results

- Formulate image inpainting as an ADMM problem and solve using ADMM
- Use "Plug-and-Play" ADMM [8] as an **open source ADMM solver**
- Measure PSNR and SSIM of inpainted images, using different **corruption masks**



- Example results:

Original Image

Corrupted Image
65.271 Percent of Data Missing

Recovered Image, TV Prior
$\lambda = 0.010$, $\rho = 1.00$
PSNR = 27.04 dB, SSIM = 0.8780

Results / Discussion

- In most cases, corruption mask determined which prior gave highest quality output
- Some images will perform better with a particular prior regardless of the mask type

PSNR	Random	Bars	Square	Small Square
Artwork	BM3D	NLM	RF	BM3D
Stanford Logo	BM3D	NLM	RF	BM3D
Basketball Players	BM3D/TV	TV	TV	TV

- Value of λ greatly affects performance: Optimal values for the test images ≈0.01 for TV and NLM priors, ≈0.005 for BM3D and RF priors.

Future Work

- Quantify relationship between mask type (corruption model) and priors, if any
- **Machine learning applications:**
  - An algorithm might be able to use other test images to determine best prior, λ to use for a given image.

Conclusion

- ADMM is an **efficient** and **effective** way of implementing image inpainting
- Not only does the image itself affect prior selection, but the **corruption model affects which prior to use**

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '00, pages 417–424, New York, NY, USA, 2000. ACMPress/Addison-Wesley Publishing Co.
[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. TrendsMach. Learn., 3(1):1–122, Jan. 2011.
[3] N. Parikh and S. Boyd. Proximal algorithms. Found. Trends Optim., 1(3):127–239, Jan. 2014.
[4] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. Phys. D, 60(1-4):259–268, Nov. 1992.
[5] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 2, pages 60–65 vol. 2, June 2005.
[6] E. S. L. Gastal and M. M. Oliveira. Domain transform for edge-aware image and video processing. In ACM SIGGRAPH 2011 Papers, SIGGRAPH '11, pages 69:1– 69:12, New York, NY, USA, 2011. ACM.
[7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising with block-matching and 3d filtering, 2006.
[8] S. H. Chan, X. Wang, and O. A. Elgendy. Plug-and-play ADMM for image restoration: Fixed point convergence and applications. CoRR, abs/1605.01710, 2016.