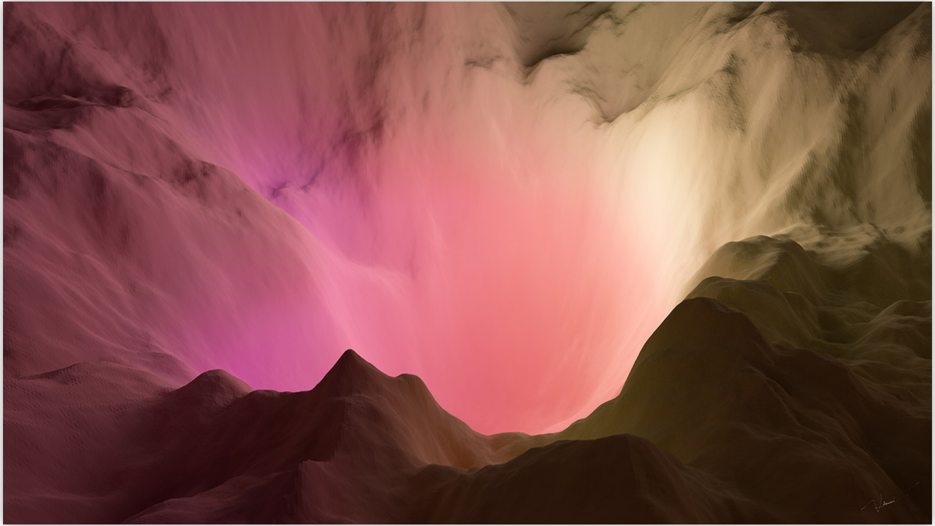
Solving Regularized Inverse Problems with ADMM



losslandscape.com

CSC2529
David Lindell
University of Toronto
cs.toronto.edu/~lindell/teaching/2529

*slides adapted from Gordon Wetzstein

Announcements

- HW5 due Friday 10/18
- HW6 is out (last one) & problem session tomorrow
- Project proposals due Friday!

Announcements

Final Project (50%)

The final project grade takes into account your poster presentation (organization of poster, clarify of presentation, ability to answer question), your source code submission (code organization and documentation), and your final project report (appropriate format and length, abstract, introduction, related work, description of your method, quantitative and qualitative evaluation of your method, results, discussion & conclusion, bibliography).

You can work in teams of up to 3 students for the project. Submit only one proposal and final report for each team. The expected amount of work is relative to the number of team members, so if two teams work on a similar project, we'd expect less work from a smaller team. Before you start to work on the proposal or the report, take a look at some of the past project proposals and reports to give you sense for what's expected (see link at the bottom of this page).

The project proposal is a 1-2 page document that should contain the following elements: clear motivation of your idea, a discussion of related work along at least 3 scientific references (i.e., scientific papers not blog articles or websites), an overview of what exactly your project is about and what the final goals are, milestones for your team with a timeline and intermediate goals. Once you send us your proposal, we may ask you to revise it and we will assign a project mentor to your team.

The final project report should look like a short (~6 pages) conference paper. We expect the following sections, which are standard practice for conference papers: abstract, introduction, related work, theory (i.e., your approach), analysis and evaluation, results, discussion and conclusion, references. To make your life easier, we provide an LaTex template that you can use to get started on your report (see schedule for link).

The project proposal is a 1-2 page document that should contain the following elements: clear motivation of your idea, a discussion of related work along at least 3 scientific references (i.e., scientific papers not blog articles or websites), an overview of what exactly your project is about and what the final goals are, milestones for your team with a timeline and intermediate goals. Once you send us your proposal, we may ask you to revise it and we will assign a project mentor to your team.

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The ADMM method
- Image deconvolution with ADMM
- Compressive imaging

Must read: course notes on Deconvolution and Compressive Imaging!

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The ADMM method
- Image deconvolution with ADMM
- Compressive imaging

Image Deconvolution – Brief Review

Given: blurry & noisy image

Desired: sharp & noise-free image

Image Deconvolution – Brief Review

- Image formation model: $b = c * x + \eta$

2D measurements

additive noise

known 2D convolution kernel

2D target image

Image Deconvolution – Brief Review

- Image formation model: $b = c * x + \eta$

- Convolution theorem:

Image Deconvolution – Brief Review

- Image formation model: $b = c * x + \eta$
- Convolution theorem: $b = \mathscr{F}^{-1}\left\{\mathscr{F}\{c\} \cdot \mathscr{F}\{x\}\right\} + \eta$

Image Deconvolution – Brief Review

- Image formation model: $b = c * x + \eta$
- Convolution theorem: $b = \mathscr{F}^{-1}\left\{\mathscr{F}\{c\} \cdot \mathscr{F}\{x\}\right\} + \eta$
- Inverse filtering:

Image Deconvolution – Brief Review

- Image formation model: $b = c * x + \eta$

- Convolution theorem: $b = \mathscr{F}^{-1}\left\{\mathscr{F}\{c\} \cdot \mathscr{F}\{x\}\right\} + \eta$

- Inverse filtering: $\tilde{x}_{\mathrm{if}} = \mathscr{F}^{-1}\left\{\dfrac{\mathscr{F}\{b\}}{\mathscr{F}\{c\}}\right\}$

Image Deconvolution – Brief Review

- Image formation model: $$b = c * x + \eta$$
- Convolution theorem: $$b = \mathcal{F}^{-1}\left\{ \mathcal{F}\{c\} \cdot \mathcal{F}\{x\} \right\} + \eta$$
- Inverse filtering: $$\tilde{x}_{\mathrm{if}} = \mathcal{F}^{-1}\left\{ \frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}} \right\}$$
- Wiener filtering:

Image Deconvolution – Brief Review

- Image formation model: $$b = c * x + \eta$$
- Convolution theorem: $$b = \mathcal{F}^{-1}\left\{\mathcal{F}\{c\} \cdot \mathcal{F}\{x\}\right\} + \eta$$
- Inverse filtering: $$\tilde{x}_{\text{if}} = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}}\right\}$$
- Wiener filtering: $$\tilde{x}_{\text{wf}} = \mathcal{F}^{-1}\left\{\frac{\left|\mathcal{F}\{c\}\right|^2}{\left|\mathcal{F}\{c\}\right|^2 + \frac{1}{SNR}} \cdot \frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}}\right\}$$

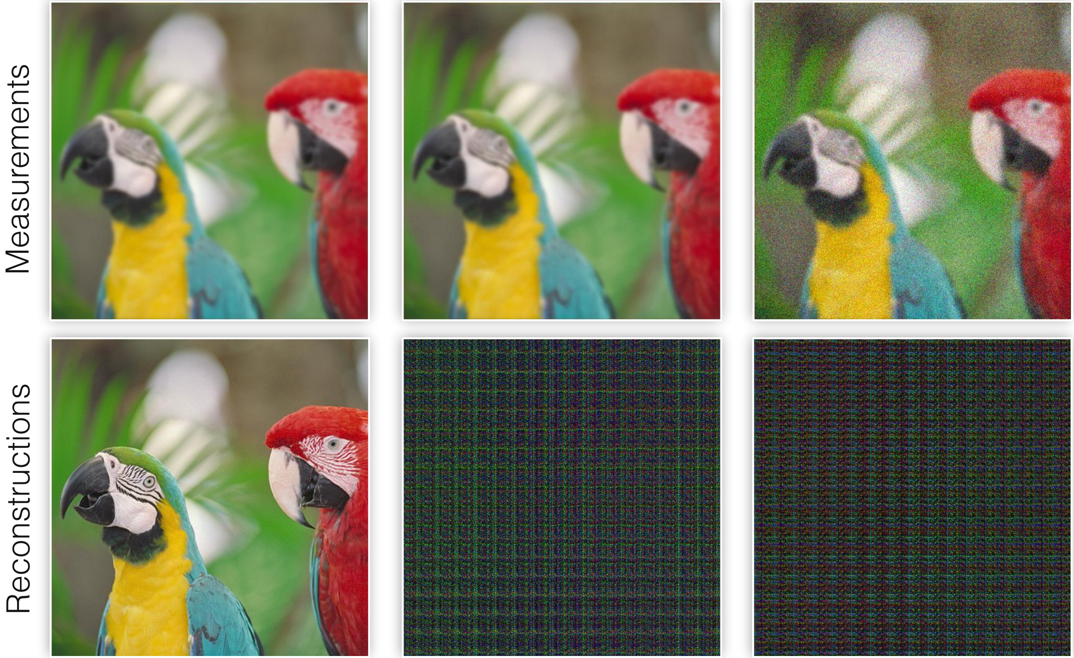
Image Deconvolution – Brief Review

- Image formation model: $$b = c * x + \eta$$
- Convolution theorem: $$b = \mathcal{F}^{-1}\left\{\mathcal{F}\{c\} \cdot \mathcal{F}\{x\}\right\} + \eta$$
- Inverse filtering: $$\tilde{x}_{\text{if}} = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}}\right\}$$
- Wiener filtering: $$\tilde{x}_{\text{wf}} = \mathcal{F}^{-1}\left\{\frac{\left|\mathcal{F}\{c\}\right|^2}{\left|\mathcal{F}\{c\}\right|^2 + \frac{1}{SNR}} \cdot \frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}}\right\}$$
- Duality of “signal processing” and “algebraic” interpretation:

$$b = c * x \;\Leftrightarrow\; \boldsymbol{b} = \boldsymbol{C}\boldsymbol{x} \qquad \boldsymbol{C} \in \mathbb{R}^{N \times N}, \quad \boldsymbol{b}, \boldsymbol{x} \in \mathbb{R}^N$$

Image Deconvolution – Inverse Filtering

Ground Truth



$$\tilde{x}_{\mathrm{if}} = \mathscr{F}^{-1}\left\{ \frac{\mathscr{F}\{b\}}{\mathscr{F}\{c\}} \right\}$$

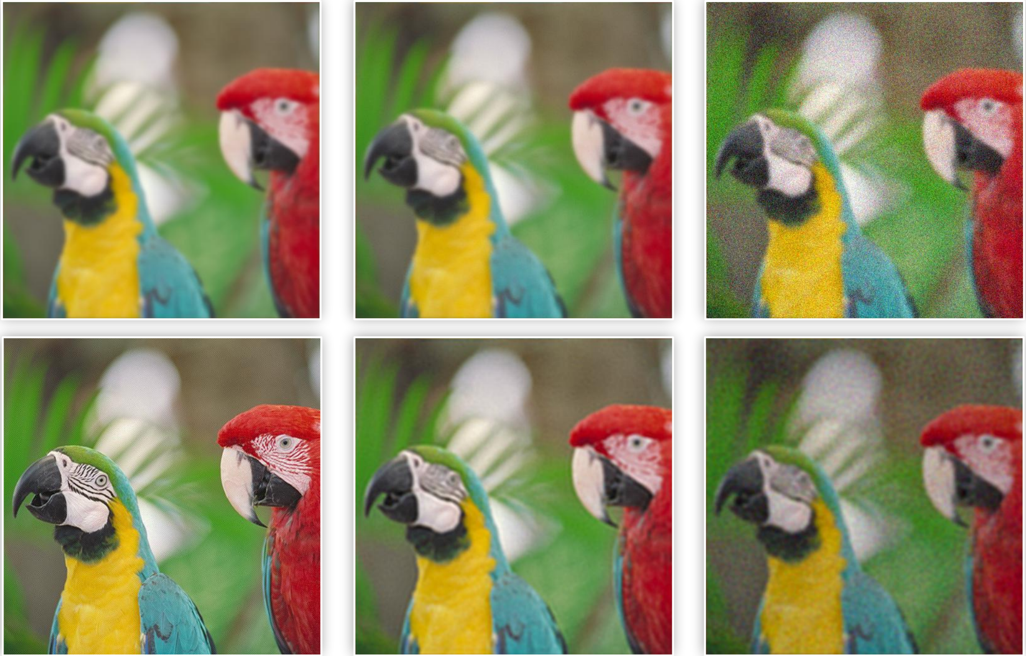
Image Deconvolution – Wiener Filtering

Ground Truth

No Noise | σ=0.1 | σ=1.0

Measurements

Reconstructions



$$x_{\mathrm{wf}} = \mathcal{F}^{-1}\left\{ \frac{\left|\mathcal{F}\{c\}\right|^2}{\left|\mathcal{F}\{c\}\right|^2 + \frac{1}{SNR}} \cdot \frac{\mathcal{F}\{b\}}{\mathcal{F}\{c\}} \right\}$$

Image Deconvolution

- Problem: this is an ill-posed inverse problem, i.e., there are infinitely many solutions that satisfy the measurements

- Need some way to determine how "desirable" any one of these feasible solutions is → need an image prior

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The ADMM method
- Image deconvolution with ADMM
- Compressive imaging

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

$$p(\boldsymbol{x} \mid \boldsymbol{b}) = ?$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M,\ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Bayes' rule: $$\underbrace{p(\boldsymbol{x} \mid \boldsymbol{b})}_{\text{posterior}} = \frac{p(\boldsymbol{b} \mid \boldsymbol{x})\, p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto \underbrace{p(\boldsymbol{b} \mid \boldsymbol{x})}_{\text{image formation model}}\, \underbrace{p(\boldsymbol{x})}_{\text{prior}}$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Bayes' rule: $$p(\boldsymbol{x} \mid \boldsymbol{b}) = \frac{p(\boldsymbol{b} \mid \boldsymbol{x})\, p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto p(\boldsymbol{b} \mid \boldsymbol{x})\, p(\boldsymbol{x})$$

posterior

image formation model

prior

$$\boldsymbol{\eta}_i \sim \mathcal{N}\left(0, \sigma^2\right) \qquad \boldsymbol{b} \mid \boldsymbol{x} \sim ?$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Bayes' rule: $$\underbrace{p(\boldsymbol{x} \mid \boldsymbol{b})}_{\text{posterior}} = \frac{p(\boldsymbol{b} \mid \boldsymbol{x})\, p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto \underbrace{p(\boldsymbol{b} \mid \boldsymbol{x})}_{\text{image formation model}}\, \underbrace{p(\boldsymbol{x})}_{\text{prior}}$$

$$\boldsymbol{\eta}_i \sim \mathcal{N}\left(0, \sigma^2\right) \qquad \boldsymbol{b} \mid \boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{x}, \sigma^2\right)$$

$$p(\boldsymbol{b} \mid \boldsymbol{x}) = \prod_{i=1}^{M} p(\boldsymbol{b}_i \mid \boldsymbol{x}_i) \propto e^{-\frac{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2}{2\sigma^2}}$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \; \boldsymbol{x} \in \mathbb{R}^N, \; \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Bayes' rule:

$$\underbrace{p(\boldsymbol{x} \mid \boldsymbol{b})}_{\text{posterior}} = \frac{p(\boldsymbol{b} \mid \boldsymbol{x})\, p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto \underbrace{p(\boldsymbol{b} \mid \boldsymbol{x})}_{\text{image formation model}} \underbrace{p(\boldsymbol{x})}_{\text{prior}}$$

- Maximum-a-posterior (MAP) solution:

$$\boldsymbol{x}_{MAP} = \arg\min_{x} \; -\log\Big(p(\boldsymbol{x} \mid \boldsymbol{b}) \Big)$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M,\ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M\times N}$

- Bayes' rule:

$$\underbrace{p(\boldsymbol{x}\,|\,\boldsymbol{b})}_{\text{posterior}} = \frac{p(\boldsymbol{b}\,|\,\boldsymbol{x})\,p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto \underbrace{p(\boldsymbol{b}\,|\,\boldsymbol{x})}_{\text{image formation model}}\,\underbrace{p(\boldsymbol{x})}_{\text{prior}}$$

- Maximum-a-posterior (MAP) solution:

$$\begin{aligned}\boldsymbol{x}_{MAP} &= \arg\min_x \ -\log\left(p(\boldsymbol{x}\,|\,\boldsymbol{b})\right) \\ &= \arg\min_x \ -\log\left(p(\boldsymbol{b}\,|\,\boldsymbol{x})\right) - \log\left(p(\boldsymbol{x})\right)\end{aligned}$$

A Bayesian Perspective of Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Bayes' rule:

$$\underbrace{p(\boldsymbol{x}\,|\,\boldsymbol{b})}_{\text{posterior}} = \frac{p(\boldsymbol{b}\,|\,\boldsymbol{x})\,p(\boldsymbol{x})}{p(\boldsymbol{b})} \propto \underbrace{p(\boldsymbol{b}\,|\,\boldsymbol{x})}_{\text{image formation model}}\,\underbrace{p(\boldsymbol{x})}_{\text{prior}}$$

- Maximum-a-posterior (MAP) solution:

$$\begin{aligned}\boldsymbol{x}_{MAP} &= \arg\min_x \ -\log\Big(p(\boldsymbol{x}\,|\,\boldsymbol{b})\Big) \\ &= \arg\min_x \ -\log\Big(p(\boldsymbol{b}\,|\,\boldsymbol{x})\Big) - \log\big(p(\boldsymbol{x})\big) \\ &= \arg\min_x \ \frac{1}{2\sigma^2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \Psi(\boldsymbol{x})\end{aligned}$$

A Bayesian Perspective of Inverse Problems

- Terminology:

$$\underbrace{\Psi(\boldsymbol{x})}_{\text{regularizer}} = -\log\big(\underbrace{p(\boldsymbol{x})}_{\text{prior}}\big)$$

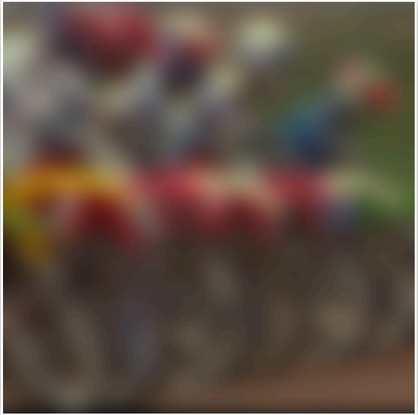
$$\boldsymbol{x}_{MAP} = \arg\min_{\boldsymbol{x}} \underbrace{\frac{1}{2\sigma^2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2}_{\text{data fidelity term}} + \underbrace{\Psi(\boldsymbol{x})}_{\text{regularization term}}$$

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The ADMM method
- Image deconvolution with ADMM
- Compressive imaging

Examples of Image Priors / Regularizers

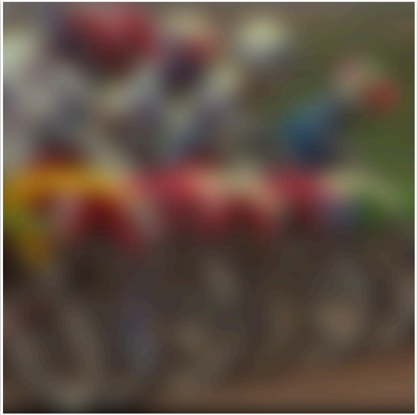
blurry stuff



Promote smoothness!

Examples of Image Priors / Regularizers

blurry stuff



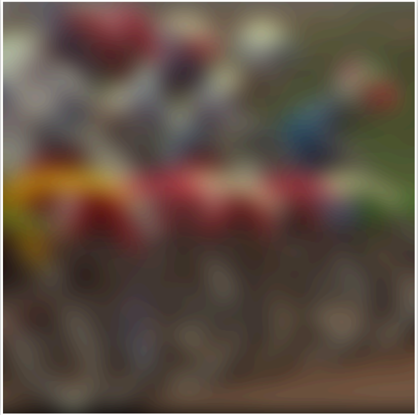
Promote smoothness!

$$\Psi(\boldsymbol{x}) = \| \Delta \boldsymbol{x} \|_2$$

Laplace operator

Examples of Image Priors / Regularizers

blurry stuff



Promote smoothness!

$$\Psi(\boldsymbol{x}) = \| \Delta \boldsymbol{x} \|_2$$

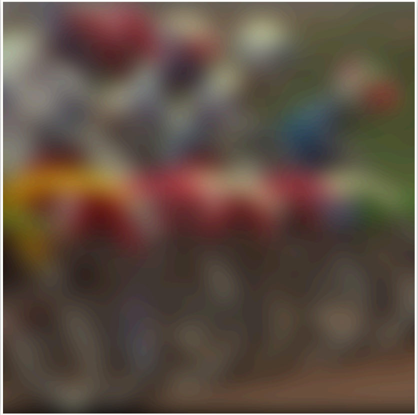
Laplace operator

stars

Promote sparsity!

Examples of Image Priors / Regularizers

blurry stuff



Promote smoothness!

$$\Psi(\boldsymbol{x}) = \| \Delta \boldsymbol{x} \|_2$$

Laplace operator

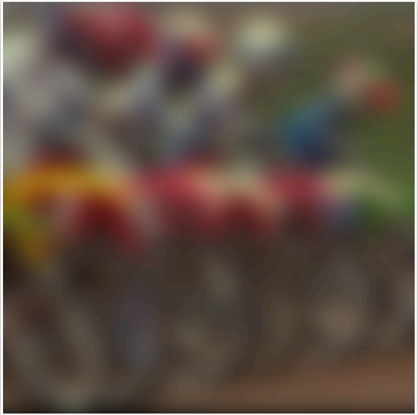
stars

Promote sparsity!

$$\Psi(\boldsymbol{x}) = \| \boldsymbol{x} \|_1$$

Examples of Image Priors / Regularizers

blurry stuff



Promote smoothness!

$$\Psi(\boldsymbol{x}) = \|\Delta \boldsymbol{x}\|_2$$

Laplace operator

stars

Promote sparsity!

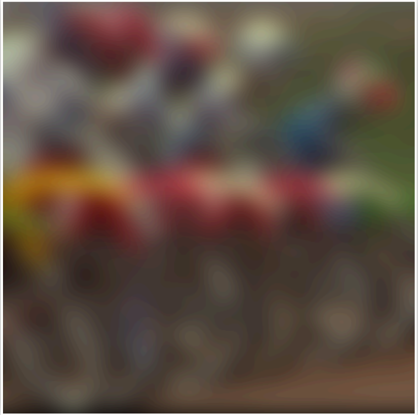
$$\Psi(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$$

"natural" image

Promote sparse gradients!

Examples of Image Priors / Regularizers

blurry stuff



Promote smoothness!

$$\Psi(\boldsymbol{x}) = \| \Delta \boldsymbol{x} \|_2$$

Laplace operator

stars

Promote sparsity!

$$\Psi(\boldsymbol{x}) = \| \boldsymbol{x} \|_1$$

"natural" image

Promote sparse gradients!

$$\Psi(\boldsymbol{x}) = \mathrm{TV}(\boldsymbol{x})$$

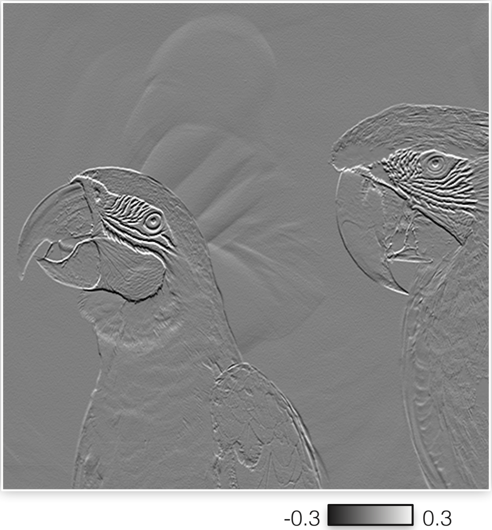
Total Variation (TV)

express (forward finite difference)
gradient as convolution

$\boldsymbol{x}$

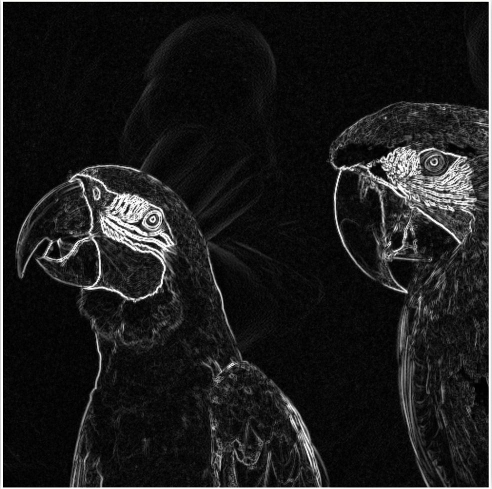
$$\boldsymbol{D}_x\boldsymbol{x} = d_x * x,\ d_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

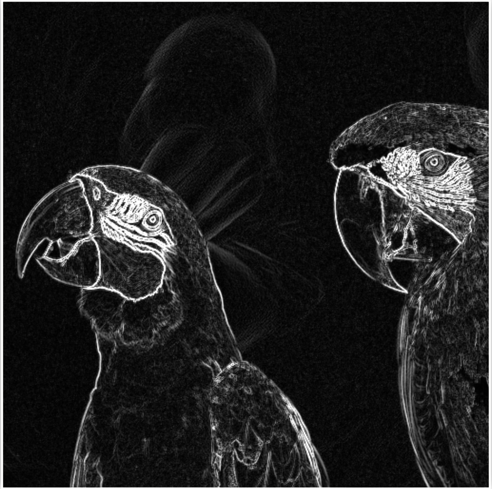
$$\boldsymbol{D}_y\boldsymbol{x} = d_y * x,\ d_y = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$



Total Variation (TV)

$\boldsymbol{x}$	better: isotropic $\sqrt{\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i^2 + \left(\boldsymbol{D}_y\boldsymbol{x}\right)_i^2}$	easier: anisotropic $\sqrt{\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i^2} + \sqrt{\left(\boldsymbol{D}_y\boldsymbol{x}\right)_i^2}$





Total Variation (TV)

- Examples are mostly black, indicating that gradient magnitudes are close to 0 → natural images have sparse gradients!

Total Variation (TV)

- Examples are mostly black, indicating that gradient magnitudes are close to 0 → natural images have sparse gradients!

- This intuition is well-captured by the TV pseudo-norm, either anisotropic or isotropic:

Total Variation (TV)

- Examples are mostly black, indicating that gradient magnitudes are close to 0 → natural images have sparse gradients!

- This intuition is well-captured by the TV pseudo-norm, either anisotropic or isotropic:

$$\mathrm{TV}_{\mathrm{anisotropic}}(\boldsymbol{x}) = \left\|\boldsymbol{D}_x\boldsymbol{x}\right\|_1 + \left\|\boldsymbol{D}_y\boldsymbol{x}\right\|_1 = \sum_{i=1}^{N} \left|\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i\right| + \left|\left(\boldsymbol{D}_y\boldsymbol{x}\right)_i\right| = \sum_{i=1}^{N} \sqrt{\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i^2} + \sqrt{\left(\boldsymbol{D}_y\boldsymbol{x}\right)_i^2}$$

Total Variation (TV)

- Examples are mostly black, indicating that gradient magnitudes are close to 0 → natural images have sparse gradients!

- This intuition is well-captured by the TV pseudo-norm, either anisotropic or isotropic:

$$\mathrm{TV}_{\mathrm{anisotropic}}(\boldsymbol{x}) = \left\|\boldsymbol{D}_x\boldsymbol{x}\right\|_1 + \left\|\boldsymbol{D}_y\boldsymbol{x}\right\|_1 = \sum_{i=1}^{N} \left|\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i\right| + \left|\left(\boldsymbol{D}_y\boldsymbol{x}\right)_i\right| = \sum_{i=1}^{N} \sqrt{\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i^2} + \sqrt{\left(\boldsymbol{D}_y\boldsymbol{x}\right)_i^2}$$

$$\mathrm{TV}_{\mathrm{isotropic}}(\boldsymbol{x}) = \left\|\boldsymbol{D}\boldsymbol{x}\right\|_{2,1} = \sum_{i=1}^{N} \left\|\begin{bmatrix}\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i \\ \left(\boldsymbol{D}_y\boldsymbol{x}\right)_i\end{bmatrix}\right\|_2 = \sum_{i=1}^{N} \sqrt{\left(\boldsymbol{D}_x\boldsymbol{x}\right)_i^2 + \left(\boldsymbol{D}_y\boldsymbol{x}\right)_i^2}$$

Total Variation (TV)

The TV pseudo-norm is one of the most popular regularization schemes for natural images!

Extensions to make it more general or applicable for other data:

- Hyper-Laplacian: Levin et al. 2009, Krishnan & Fergus 2009
- Total generalized variation: Bredies et al. 2009
- Frobenius norm of Hessian: Lefkimmiatis et al. 2003
- …

How to solve inverse problem that use these regularizers?

Solving Regularized Inverse Problem

- Objective or “loss” function of general inverse problem:

$$\text{minimize}_x \ \frac{1}{2}\left\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2+\lambda\Psi(\boldsymbol{x})$$

weight of regularizer

Solving Regularized Inverse Problem

- Objective or "loss" function of general inverse problem: $\text{minimize}_x \ \frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \lambda\Psi(\boldsymbol{x})$

weight of regularizer

- Practical #1 go-to solution: Adam solver implemented in PyTorch
- 3 simple steps, will explore in problem session & homework:
  1. Implement evaluation of loss function
  2. Set hyperparameters, including learning rate
  3. Run

Solving Regularized Inverse Problem

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_x \ \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \underset{\substack{\uparrow \\ \text{weight of regularizer}}}{\lambda}\Psi(\boldsymbol{x})$$

- Practical #1 go-to solution: Adam solver implemented in PyTorch
- 3 simple steps, will explore in problem session & homework:
  1. Implement evaluation of loss function
  2. Set hyperparameters, including learning rate
  3. Run
- The "fine print": convenient but doesn't always converge well

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The Alternating Direction Method of Multipliers (ADMM)
- Image deconvolution with ADMM
- Compressive imaging

Regularized Image Reconstruction

$$\underset{\{x\}}{\text{minimize}} \; \frac{1}{2}\left\|b - Ax\right\|_2^2 + \lambda\Gamma(x)$$

data fidelity term

some image prior, such as L1 norm or others

Regularized Image Reconstruction

$$\underset{\{x\}}{\text{minimize}} \; \frac{1}{2}\|b - Ax\|_2^2 + \lambda\Gamma(x)$$

$$\underset{\{x\}}{\text{minimize}} \; \underbrace{\frac{1}{2}\|b - Ax\|_2^2}_{f(x)} + \underbrace{\lambda\Gamma(z)}_{g(z)}$$

$$\text{subject to} \quad Kx - z = 0$$

- split into two parts → mathematically equivalent

Regularized Image Reconstruction

$$\underset{\{x\}}{\text{minimize}} \quad \underbrace{\frac{1}{2}\|b - Ax\|_2^2}_{f(x)} + \underbrace{\lambda\Gamma(z)}_{g(z)}$$

$$\text{subject to} \quad Kx - z = 0$$

- Lagrangian

$$L(x, z, y) = f(x) + g(z) + y^T(Kx - z)$$

Regularized Image Reconstruction

- Lagrangian

$$L(x, z, y) = f(x) + g(z) + y^T (Kx - z)$$

Optimal if all partial derivatives are zero!

$$\nabla_{x,z,y} L = 0$$

Regularized Image Reconstruction

- Lagrangian

$$L(x, z, y) = f(x) + g(z) + y^T(Kx - z)$$

Also implies:

$$\nabla_{x,z,y} L = 0 \Leftrightarrow \nabla_{x,z} f(x) + g(z) = y^T \nabla_{x,z}(Kx - z)$$

$$\boxed{Kx - z = 0}$$

Regularized Image Reconstruction

- Lagrangian

$$L(x,z,y) = f(x) + g(z) + y^T(Kx - z)$$

Also implies:

$$\nabla_{x,z,y} L = 0 \Leftrightarrow \nabla_{x,z} f(x) + g(z) = y^T \nabla_{x,z}(Kx - z)$$

$$\boxed{Kx - z = 0}$$

Regularized Image Reconstruction

- Lagrangian

$$L(x, z, y) = f(x) + g(z) + y^T(Kx - z)$$

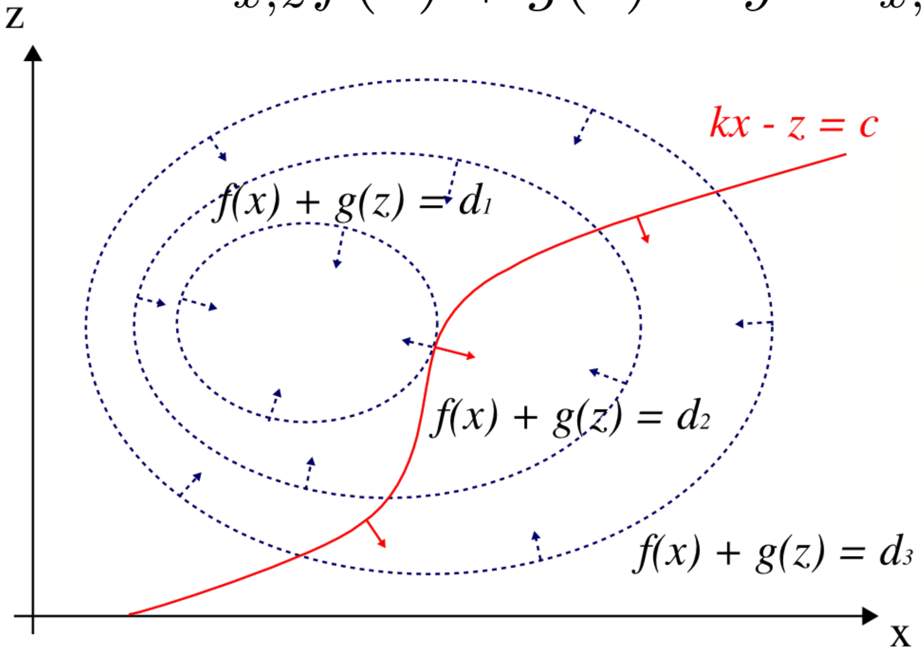
Also implies:

$$\nabla_{x,z,y} L = 0 \Leftrightarrow \boxed{\nabla_{x,z} f(x) + g(z) = y^T \nabla_{x,z}(Kx - z)}$$

$$Kx - z = 0$$

Regularized Image Reconstruction

$$\nabla_{x,z,y} L = 0 \Leftrightarrow \nabla_{x,z} f(x) + g(z) = y^T \nabla_{x,z}(Kx - z)$$



Regularized Image Reconstruction

- Augmented Lagrangian

$$L_\rho(x,z,y) = f(x) + g(z) + y^T(Kx - z) + \boxed{\frac{\rho}{2}\|Kx - z\|_2^2}$$

We add quadratic penalty: improves convergence properties compared to standard Lagrangian

ρ is called the "penalty parameter" (see Boyd 2011)

Regularized Image Reconstruction

- Scaled dual form of the augmented Lagrangian
  - Given by some algebraic manipulation (easier form to work with)

$$L_\rho(x, z, y) = f(x) + g(z) + \frac{\rho}{2}\|Kx - z + u\|^2 + \frac{\rho}{2}\|u\|^2$$

Where $u = y/\rho$ is the scaled dual variable

Regularized Image Reconstruction

- Recap
  - Split the objective function and enforce consistency with constraint
  - handle constraints using Lagrangian
  - Lagrangian -> Augmented Lagrangian -> scaled dual form

$$L_\rho(x,z,y) = f(x) + g(z) + \frac{\rho}{2}\|Kx - z + u\|^2 + \frac{\rho}{2}\|u\|^2$$

Regularized Image Reconstruction

- Recap
  - Split the objective function and enforce consistency with constraint
  - handle constraints using Lagrangian
  - Lagrangian -> Augmented Lagrangian -> scaled dual form
  - optimize using ADMM!

$$L_\rho(x, z, y) = f(x) + g(z) + \frac{\rho}{2}\|Kx - z + u\|^2 + \frac{\rho}{2}\|u\|^2$$

Regularized Image Reconstruction

$$L_\rho(x, z, y) = f(x) + g(z) + \frac{\rho}{2}\|Kx - z + u\|^2 + \frac{\rho}{2}\|u\|^2$$

repeat until converged

$$
\begin{aligned}
x &\leftarrow \quad \mathrm{prox}_{\|\cdot\|_2,\rho}(v) = \underset{\{x\}}{\arg\min}\, L_\rho(x,z,y) \\
z &\leftarrow \quad \mathrm{prox}_{\Gamma,\rho}(v) = \underset{\{z\}}{\arg\min}\, L_\rho(x,z,y) \\
u &\leftarrow \quad u + Kx - z
\end{aligned}
$$

- iterative updates - ADMM

Regularized Image Reconstruction

$$\mathbf{prox}_{\lambda f}(v) = \underset{x}{\mathrm{argmin}} \left(f(x) + (1/2\lambda)\|x - v\|_2^2 \right)$$

gives proximal point of v with respect to f

- finds a value of that is close to v and minimum of f, or moves to the domain of f

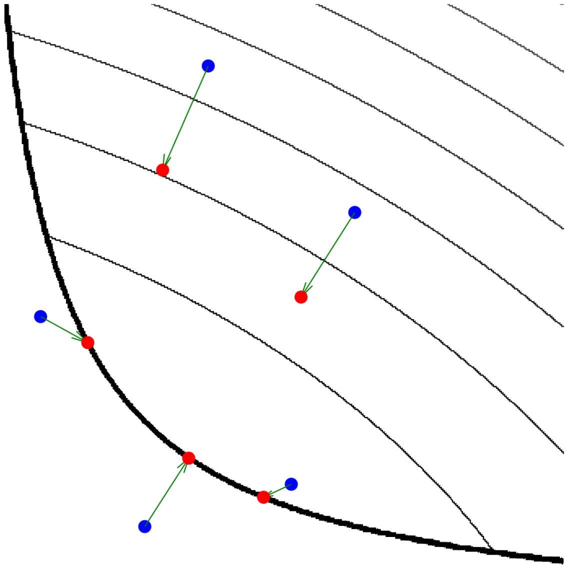


Figure 1.1: Evaluating a proximal operator at various points.

Regularized Image Reconstruction

$$L_\rho(x,z,y) = f(x) + g(z) + \frac{\rho}{2}\|Kx - z + u\|^2 + \frac{\rho}{2}\|u\|^2$$

repeat until converged

$$x \leftarrow \quad \mathrm{prox}_{\|\cdot\|_2,\rho}(v) = \underset{\{x\}}{\arg\min}\, L_\rho(x,z,y) = \underset{\{x\}}{\arg\min}\, \frac{1}{2}\|Ax-b\|_2^2 + \frac{\rho}{2}\|Kx-v\|,\ v = z-u$$

$$z \leftarrow \quad \mathrm{prox}_{\Gamma,\rho}(v) = \underset{\{z\}}{\arg\min}\, L_\rho(x,z,y) = \underset{\{z\}}{\arg\min}\, \lambda\Gamma(z) + \frac{\rho}{2}\|v-z\|,\ v = Kx+u$$

$$u \leftarrow \quad u + Kx - z$$

- iterative updates - ADMM

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The Alternating Direction Method of Multipliers (ADMM)
- Image deconvolution with ADMM
- Compressive imaging

ADMM for Image Deconvolution with TV

Generic: $$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\left\|\boldsymbol{K}\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{u}\right\|_2^2 + \frac{\rho}{2}\left\|\boldsymbol{u}\right\|_2^2$$

Deconv: $$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \frac{1}{2}\left\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\right\|_2^2 + \lambda\left\|\boldsymbol{z}\right\|_1 + \frac{\rho}{2}\left\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{u}\right\|_2^2 + \frac{\rho}{2}\left\|\boldsymbol{u}\right\|_2^2$$

$\boldsymbol{x} \in \mathbb{R}^N$ unknown sharp image

$\boldsymbol{C} \in \mathbb{R}^{N \times N}$ circulant convolution matrix for known kernel $\boldsymbol{c}$

$\boldsymbol{z},\ \boldsymbol{u} \in \mathbb{R}^{2N}$ slack/dual variable, twice the size of $\boldsymbol{x}$!

$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{D}_x \\ \boldsymbol{D}_y \end{bmatrix} \in \mathbb{R}^{2N \times N}$ finite difference gradients, horizontal & vertical

ADMM for Image Deconvolution with TV

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \ \lambda\|\boldsymbol{z}\|_1 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{u}\|_2^2 + \frac{\rho}{2}\ \|\boldsymbol{u}\|_2^2$$

```
while not converged:
```

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_x \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\|_2^2,\ \boldsymbol{v} = \boldsymbol{z} - \boldsymbol{u}$$

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\|\cdot\|_1,\rho}(\boldsymbol{v}) = \arg\min_z \lambda\|\boldsymbol{z}\|_1 + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2,\ \boldsymbol{v} = \ \boldsymbol{D}\boldsymbol{x} + \boldsymbol{u}$$

ADMM for Image Deconvolution with TV

$\underline{\boldsymbol{x} - \text{update:}}$

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\|_2^2$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v}\|_2^2$$

reformulate

$$= \frac{1}{2}\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right)^T\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right) + \frac{\rho}{2}(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})^T(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v}\|_2^2$$

reformulate

$$= \frac{1}{2}\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right)^T\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right) + \frac{\rho}{2}(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})^T(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})$$

$$= \frac{1}{2}\left(\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{b} + \boldsymbol{b}^T\boldsymbol{b}\right) + \frac{\rho}{2}\left(\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{v} + \boldsymbol{v}^T\boldsymbol{v}\right)$$

ADMM for Image Deconvolution with TV

$\underline{\boldsymbol{x} - \text{update:}}$

$$\boldsymbol{x} \leftarrow \text{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\|_2^2$$

reformulate

$$= \frac{1}{2}\left(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\right)^T\left(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\right) + \frac{\rho}{2}(\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v})^T(\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v})$$

$$= \frac{1}{2}\left(\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{b} + \boldsymbol{b}^T\boldsymbol{b}\right) + \frac{\rho}{2}\left(\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{v} + \boldsymbol{v}^T\boldsymbol{v}\right)$$

find solution by setting gradient to 0

$$0 = \nabla_x f(\boldsymbol{x}) = \boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} - \boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{x} - \rho\boldsymbol{D}^T\boldsymbol{v}$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v}\|_2^2$$

reformulate

$$= \frac{1}{2}\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right)^T\left(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right) + \frac{\rho}{2}(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})^T(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v})$$

$$= \frac{1}{2}\left(\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{C}^T\boldsymbol{b} + \boldsymbol{b}^T\boldsymbol{b}\right) + \frac{\rho}{2}\left(\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{x} - 2\boldsymbol{x}^T\boldsymbol{D}^T\boldsymbol{v} + \boldsymbol{v}^T\boldsymbol{v}\right)$$

find solution by setting gradient to 0

$$0 = \nabla_x f(\boldsymbol{x}) = \boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} - \boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{x} - \rho\boldsymbol{D}^T\boldsymbol{v}$$

closed-form solution

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\|_2^2$$

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \operatorname{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\left\|\boldsymbol{Cx}-\boldsymbol{b}\right\|_2^2 + \frac{\rho}{2}\left\|\boldsymbol{Dx}-\boldsymbol{v}\right\|_2^2$$

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)$$

Exploit duality of algebraic & signal processing interpretation

$$\boldsymbol{C}^T\boldsymbol{C} \Leftrightarrow \mathscr{F}^{-1}\left\{\mathscr{F}\{c\}^* \cdot \mathscr{F}\{c\}\right\}$$

$$\boldsymbol{D}^T\boldsymbol{D} \Leftrightarrow \mathscr{F}^{-1}\left\{\mathscr{F}\left\{d_x\right\}^* \cdot \mathscr{F}\left\{d_x\right\} + \mathscr{F}\left\{d_y\right\}^* \cdot \mathscr{F}\left\{d_y\right\}\right\}$$

$$\boldsymbol{D}^T\boldsymbol{z} = \boldsymbol{D}_{\boldsymbol{x}}^T\boldsymbol{v}_{\mathbf{1}} + \boldsymbol{D}_{\boldsymbol{y}}^T\boldsymbol{v}_{\mathbf{2}} \Leftrightarrow \mathscr{F}^{-1}\left\{\mathscr{F}\left\{d_x\right\} * \cdot \mathscr{F}\left\{v_1\right\} + \mathscr{F}\left\{d_y\right\} * \cdot \mathscr{F}\{\ldots\right.$$

$$\boldsymbol{C}^T\boldsymbol{b} \Leftrightarrow \mathscr{F}^{-1}\left\{\mathscr{F}\{c\} * \cdot \mathscr{F}\{b\}\right\}$$

ADMM for Image Deconvolution with TV

<u>$\boldsymbol{x}$ – update:</u>

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{Cx}-\boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{Dx}-\boldsymbol{v}\|_2^2$$

$$\boldsymbol{x} \leftarrow \underline{\left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}}\underline{\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)}$$

Exploit duality of algebraic & signal processing interpretation

$$\boldsymbol{C}^T\boldsymbol{C} \Leftrightarrow \mathcal{F}^{-1}\{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\}\}$$

$$\boldsymbol{D}^T\boldsymbol{D} \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{d_x\} + \mathcal{F}\left\{d_y\right\}^* \cdot \mathcal{F}\left\{d_y\right\}\right\}$$

$$\boldsymbol{D}^T\boldsymbol{z} = \boldsymbol{D}_{\boldsymbol{x}}^T\boldsymbol{v_1} + \boldsymbol{D}_{\boldsymbol{y}}^T\boldsymbol{v_2} \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{d_x\} * \cdot \mathcal{F}\{v_1\} + \mathcal{F}\left\{d_y\right\} * \cdot \mathcal{F}\{v_2\}\right\}$$

$$\boldsymbol{C}^T\boldsymbol{b} \Leftrightarrow \mathcal{F}^{-1}\{\mathcal{F}\{c\} * \cdot \mathcal{F}\{b\}\}$$

$$\underline{\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}} \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{d_x\} + \mathcal{F}\left\{d_y\right\}^* \cdot \mathcal{F}\left\{d_y\right\}\right)\right\}$$

$$\underline{\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}} \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{v_1\} + \mathcal{F}\left\{d_y\right\}^* \cdot \mathcal{F}\{v_2\}\right)\right\}$$

ADMM for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{v}\|_2^2$$

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)$$

- Efficient $\boldsymbol{x}$–update operates purely on 2D images with FFTs and element-wise multiplications and divisions:

$$\mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{z}) = \mathcal{F}^{-1}\left\{\frac{\boxed{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\}} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{v_1\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{v_2\}\right)}{\boxed{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{d_x\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{d_y\}\right)}}\right\}$$

can pre-compute most parts

$$v_1 = \boldsymbol{v}(1:N),\ v_2 = \boldsymbol{v}(N+1:2N)$$

ADMM for Image Deconvolution with TV

$\boldsymbol{z}$ – update:

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\|\cdot\|_1,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda \|\boldsymbol{z}\|_1 + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2$$

- Efficient $\boldsymbol{z}$–update uses element-wise soft thresholding operator $\mathcal{S}_\kappa(\cdot)$:

ADMM for Image Deconvolution with TV

<u>$\boldsymbol{z}$ – update:</u>

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\|\cdot\|_1,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda\|\boldsymbol{z}\|_1 + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2$$

- Efficient $\boldsymbol{z}$–update uses element-wise soft thresholding operator $\mathcal{S}_\kappa(\cdot)$:

$$\mathrm{prox}_{\|\cdot\|_1,\rho}(\boldsymbol{v}) = \mathcal{S}_\kappa(\boldsymbol{v}) = \begin{cases} v - \kappa & v > \kappa \\ 0 & |v| \leq \kappa \\ v + \kappa & v < -\kappa \end{cases} = (v - \kappa)_+ - (-v - \kappa)_+$$

$$\kappa = \frac{\lambda}{\rho}$$

$$\boldsymbol{v} = \boldsymbol{D}\boldsymbol{x} + \boldsymbol{u}$$

This element-wise soft thresholding is the proximal operator for anisotropic TV, see course notes on block soft thresholding for isotropic TV.

ADMM for Image Deconvolution with Denoiser

$$L_\rho(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\left\|\boldsymbol{K}\boldsymbol{x}-\boldsymbol{z}+\boldsymbol{u}\right\|_2^2 + \frac{\rho}{2}\ \|\boldsymbol{u}\|_2^2$$

<u>$\boldsymbol{x}$ – update:</u>

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\left\|\boldsymbol{C}\boldsymbol{x}-\boldsymbol{b}\right\|_2^2 + \frac{\rho}{2}\|\boldsymbol{x}-\boldsymbol{v}\|_2^2 \qquad \boldsymbol{v}\in\mathbb{R}^N$$

ADMM for Image Deconvolution with Denoiser

$\underline{\boldsymbol{x} - \text{update:}}$

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{x} - \boldsymbol{v}\|_2^2 \qquad \boldsymbol{v} \in \mathbb{R}^N$$

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho \boldsymbol{I}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{v}\right)$$

no matrix $\boldsymbol{D}$!

ADMM for Image Deconvolution with Denoiser

$\boldsymbol{x}$ – update:

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{x} - \boldsymbol{v}\|_2^2 \qquad \boldsymbol{v} \in \mathbb{R}^N$$

$$\boldsymbol{x} \leftarrow \left(\boldsymbol{C}^T\boldsymbol{C} + \rho\boldsymbol{I}\right)^{-1}\left(\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{v}\right)$$

no matrix $\boldsymbol{D}$!

- Efficient $\boldsymbol{x}$–update operates purely on 2D images with FFTs and element-wise multiplications and divisions:

$$\mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \mathscr{F}^{-1}\left\{ \frac{\mathscr{F}\{c\}^* \cdot \mathscr{F}\{b\} + \rho\mathscr{F}\{v\}}{\mathscr{F}\{c\}^* \cdot \mathscr{F}\{c\} + \rho} \right\}$$

ADMM for Image Deconvolution with Denoiser

$\underline{\boldsymbol{z} - \text{update:}}$

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\mathscr{D},\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda\Psi(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2 \quad \boldsymbol{v} = \boldsymbol{x} + \boldsymbol{u}$$

ADMM for Image Deconvolution with Denoiser

$\boldsymbol{z}$ – update:

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\mathscr{D},\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda\Psi(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2 \quad \boldsymbol{v} = \boldsymbol{x} + \boldsymbol{u}$$

$$= \arg\min_{\boldsymbol{z}} \Psi(\boldsymbol{z}) + \frac{\rho}{2\lambda}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2$$

This is a denoising problem with a regularizer that imposes a prior!

ADMM for Image Deconvolution with Denoiser

$\boldsymbol{z}$ – update:

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\mathscr{D},\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda\Psi(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2 \quad \boldsymbol{v} = \boldsymbol{x} + \boldsymbol{u}$$

$$= \arg\min_{\boldsymbol{z}} \Psi(\boldsymbol{z}) + \frac{\rho}{2\lambda}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2$$

- Efficient $\boldsymbol{z}$–update uses arbitrary denoiser $\mathscr{D}(\,\cdot\,)$, such as DnCNN and non-local means, using noise variance $\sigma^2 = \frac{\lambda}{\rho}$

$$\mathrm{prox}_{\mathscr{D},\rho}(\boldsymbol{x}) = \mathscr{D}\left(\boldsymbol{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

Image Deconvolution with ADMM

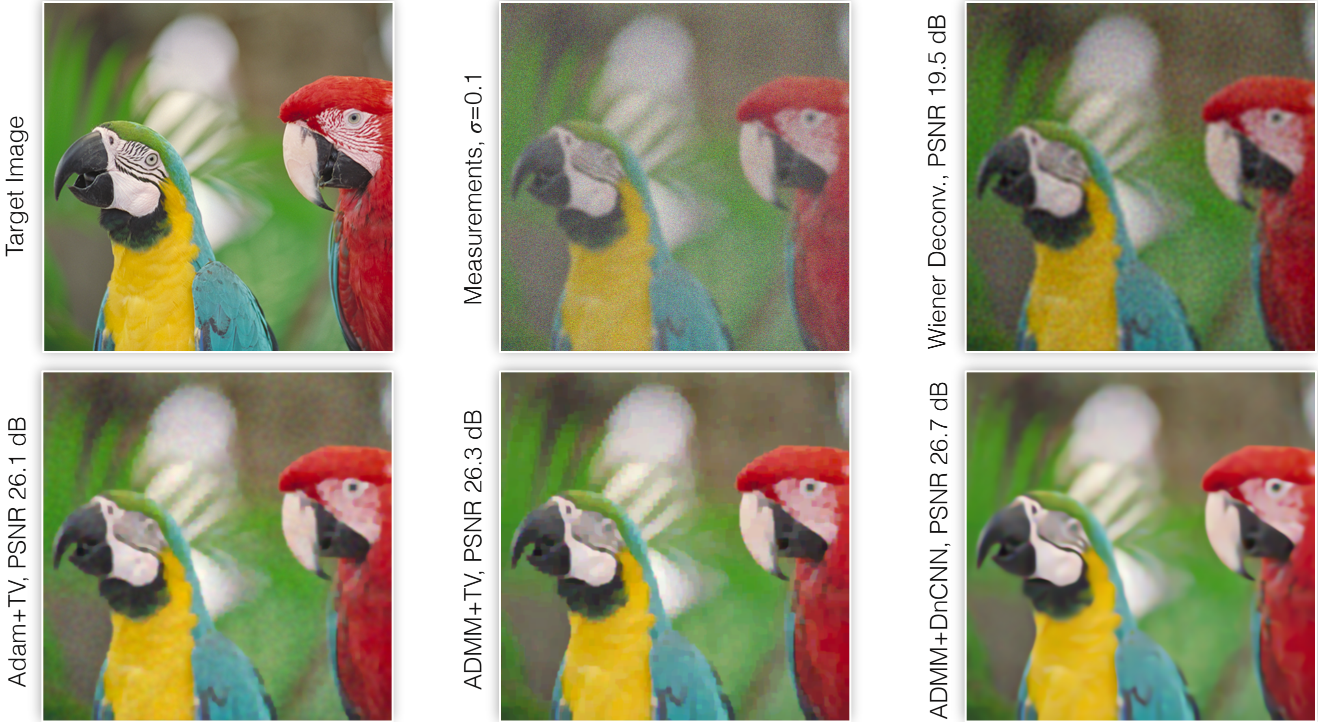


Image Deconvolution with ADMM

ADMM for deconvolution with TV

initialize ρ and λ
$x = zeros\,(W, H)\,;$
$z = zeros\,(W, H)\,;$
$u = zeros\,(W, H)\,;$
for $k = 1$ **to** max_iters **do**
    $v = z - u$
    $x = \mathbf{prox}_{\|\cdot\|_2,\rho}\,(v) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^*\cdot\mathcal{F}\{b\}+\rho(\mathcal{F}\{d_x\}^*\cdot\mathcal{F}\{v_1\}+\mathcal{F}\{d_y\}^*\cdot\mathcal{F}\{v_2\})}{\mathcal{F}\{c\}^*\cdot\mathcal{F}\{c\}+\rho(\mathcal{F}\{d_x\}^*\cdot\mathcal{F}\{d_x\}+\mathcal{F}\{d_y\}^*\cdot\mathcal{F}\{d_y\})}\right\}$
    $v = \mathbf{Dx} + \mathbf{u}$
    $z = \mathbf{prox}_{\|\cdot\|_1,\rho}\,(v) = \mathcal{S}_{\lambda/\rho}\,(v)$
    $u = u + \mathbf{Dx} - z$
end for

ADMM for deconvolution with denoiser

initialize ρ and λ
$x = zeros\,(W, H)\,;$
$z = zeros\,(W, H)\,;$
$u = zeros\,(W, H)\,;$
for $k = 1$ **to** max_iters **do**
    $v = z - u$
    $x = \mathbf{prox}_{\|\cdot\|_2,\rho}\,(v) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^*\cdot\mathcal{F}\{b\}+\rho\mathcal{F}\{v\}}{\mathcal{F}\{c\}^*\cdot\mathcal{F}\{c\}+\rho}\right\}$
    $v = x + u$
    $z = \mathbf{prox}_{\mathcal{D},\rho}\,(v) = \mathcal{D}\left(v, \sigma^2 = \frac{\lambda}{\rho}\right)$
    $u = u + x - z$
end for

ADMM - Convergence Criterion

- Run or "unroll" ADMM for *K* iterations
- Run until change in residual between iterations is < threshold

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\tfrac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho \mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \tfrac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\tfrac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho \mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \tfrac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\tfrac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho \mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \tfrac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

$$\vdots$$

Outlook on Unrolled Optimization

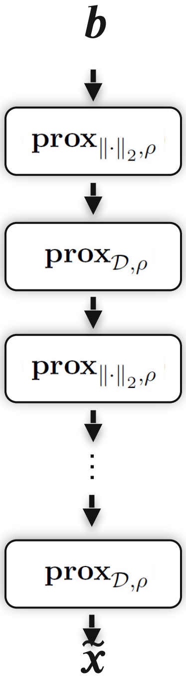
- Run or "unroll" ADMM for *K* iterations
- Interpret as unrolled feedforward network:

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \frac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \frac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

$$
\begin{aligned}
v &= z - u \\
x &= \mathbf{prox}_{\|\cdot\|_2,\rho}(v) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{v\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\} \\
v &= x + u \\
z &= \mathbf{prox}_{\mathcal{D},\rho}(v) = \mathcal{D}\left(v, \sigma^2 = \frac{\lambda}{\rho}\right) \\
u &= u + x - z
\end{aligned}
$$

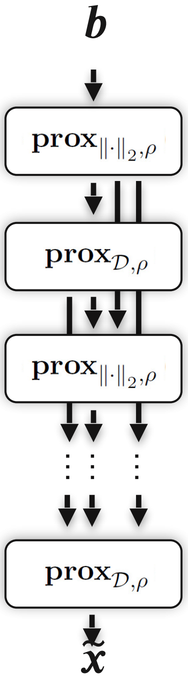
$$\vdots$$



[Diamond et al. 2017]

Outlook on Unrolled Optimization

- Run or "unroll" ADMM for K iterations
- Interpret as unrolled feedforward network:



Benefits over unrolled optimization

- Learnable parameters: $\lambda^{(k)}, \rho^{(k)},$ denoiser $\mathscr{D}^{(k)}$
- DenseNet-like skip connections
- Denoiser/regularizer can adapt to matrix $\boldsymbol{C}$
- Can train with advanced loss functions (perceptual, adversarial, other network, …)

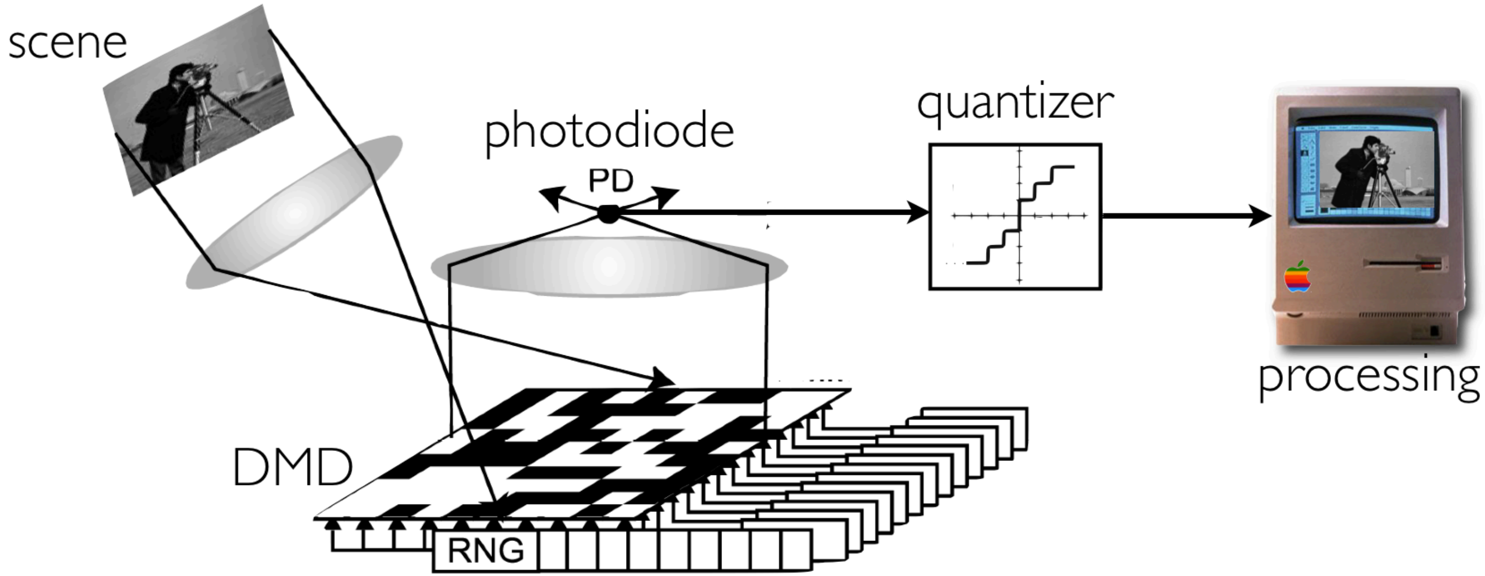
[Diamond et al. 2017]

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The Alternating Direction Method of Multipliers (ADMM)
- Image deconvolution with ADMM
- Compressive imaging

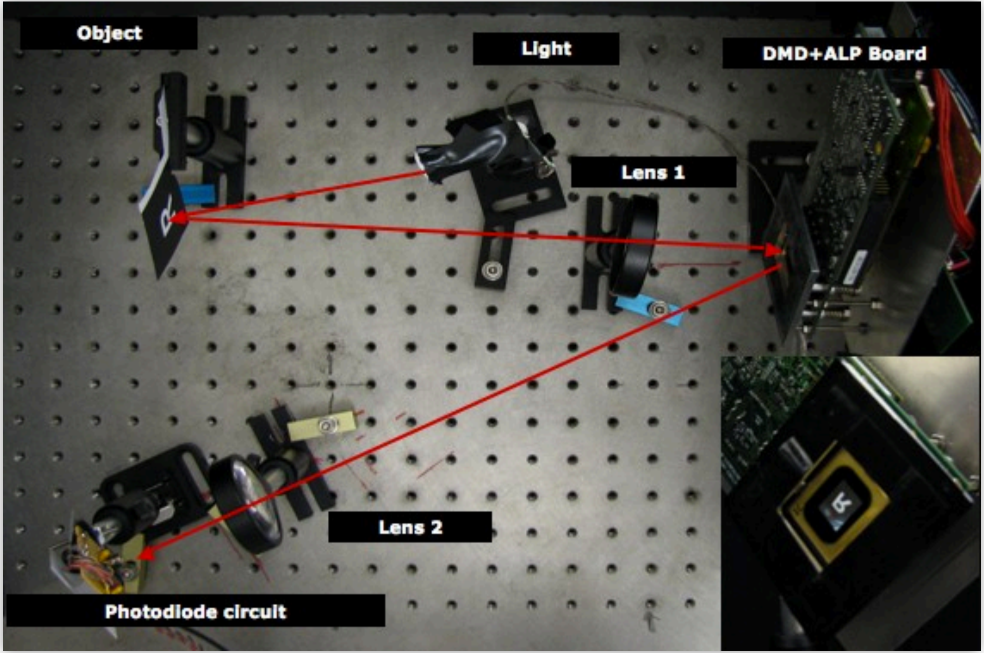
Compressive Imaging

Single-pixel Imaging



Duarte et al. 2008

Single-pixel Imaging



original

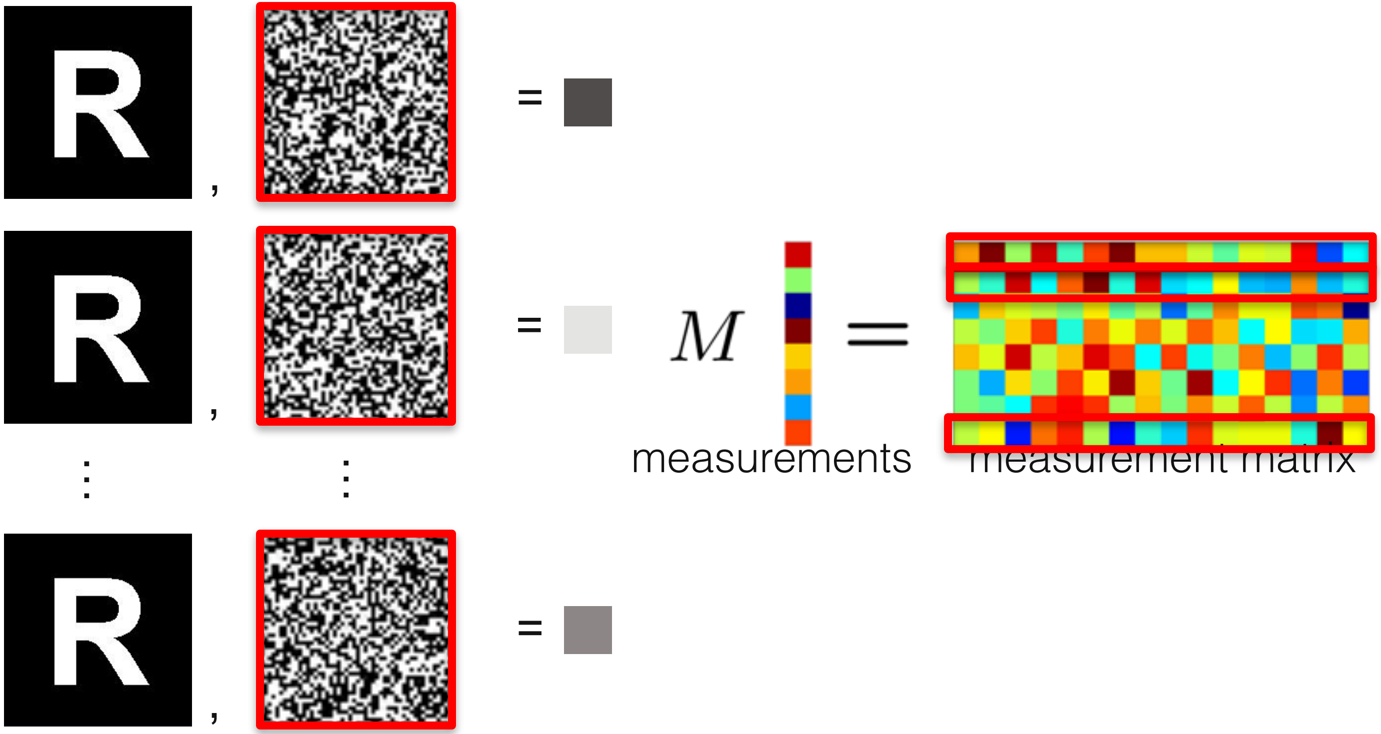
10%

5%

2%

Duarte et al. 2008

Single-pixel Imaging



Under-determined Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^{M}, \ \boldsymbol{x} \in \mathbb{R}^{N}, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- What makes it under-determined (or a compressive imaging problem):

Under-determined Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- What makes it under-determined (or a compressive imaging problem): $M < N$

Under-determined Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- What makes it under-determined (or a compressive imaging problem): $M < N$

- Problem: infinitely many solutions satisfy the observations!
  Same problem as ill-posed problems! → need image priors

Under-determined Inverse Problems

- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M, \ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}$

- Standard approach – the least-norm solution: $\tilde{\boldsymbol{x}}_{\text{ln}} = \boldsymbol{A}^T \left(\boldsymbol{A}\boldsymbol{A}^T \right) \boldsymbol{b}$

- This is the solution of optimization problem $$\begin{aligned} &\text{minimize}_x \ \|\boldsymbol{x}\|_2 \\ &\text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

Note: among the infinitely many solutions satisfying the observations, the least-norm solution is the one with the smallest L2 norm, thus equivalent to $\| \cdot \|_2$ regularizer

Under-determined Inverse Problems

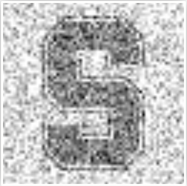
- Image formation model: $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \quad \boldsymbol{b} \in \mathbb{R}^M,\ \boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M\times N}$

- Standard approach – the least-norm solution: $\tilde{\boldsymbol{x}}_{\mathrm{ln}} = \boldsymbol{A}^T\left(\boldsymbol{A}\boldsymbol{A}^T\right)\boldsymbol{b}$

- Results (not great):

Compression Factor $\frac{\boldsymbol{N}}{\boldsymbol{M}}$

2x



PSNR 12.3

4x

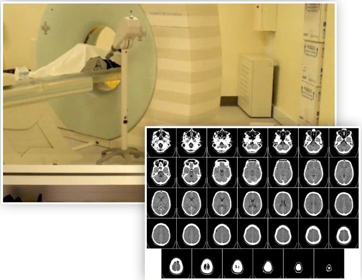
PSNR 10.4

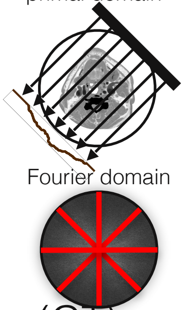
8x

PSNR 9.7

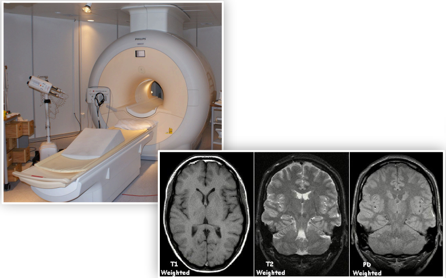
Other Inverse Problems in Imaging

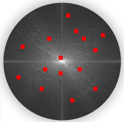
Images: Wikipedia



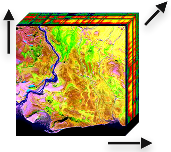


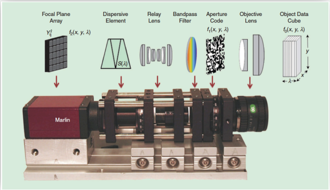
Computed Tomography (CT)



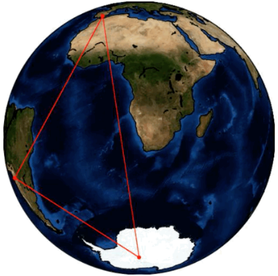


Magnetic Resonance Imaging (MRI)





Hyperspectral Imaging



Interferometry

Other Inverse Problems in Imaging

- All these inverse problems have important applications and are very different
- Yet, they all boil down to the same inverse problem, each with a different matrix $\boldsymbol{A}$: $$\text{minimize}_{\boldsymbol{x}} \ \frac{1}{2}\left\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2+\lambda\Psi(\boldsymbol{x})$$
- The methods derived here also apply to all those problems and applications; single-pixel imaging is a great example problem → "if you can solve this, you can solve anything"

Review of ADMM for General Inverse Problems

- Objective or "loss" function of general inverse problem: $$\text{minimize}_{\boldsymbol{x}} \ \frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \lambda\Psi(\boldsymbol{x})$$

weight of regularizer

Review of ADMM for General Inverse Problems

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_{\boldsymbol{x}} \ \frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \lambda\Psi(\boldsymbol{x})$$

weight of regularizer

- Reformulate as:

$$\text{minimize}_{\{x,z\}} \ \underbrace{\frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\Psi(\boldsymbol{z})}_{g(\boldsymbol{z})}$$

$$\text{subject to} \ \ \boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} = 0$$

Review of ADMM for General Inverse Problems

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_{\boldsymbol{x}} \ \frac{1}{2}\left\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \underbrace{\lambda}_{\text{weight of regularizer}}\Psi(\boldsymbol{x})$$

- Reformulate as:

$$\text{minimize}_{\{x,z\}} \ \underbrace{\frac{1}{2}\left\|\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\Psi(\boldsymbol{z})}_{g(\boldsymbol{z})}$$

$$\text{subject to} \ \ \boldsymbol{D}\boldsymbol{x}-\boldsymbol{z}=0$$

- Remove constraints using augmented Lagrangian

$$L_\rho(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\left\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{z}+\boldsymbol{u}\right\|_2^2 +$$

$$\left\|\boldsymbol{u}\right\|_2^2$$

Review of ADMM for General Inverse Problems

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\left\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} + \boldsymbol{u}\right\|_2^2 + \left\|\boldsymbol{u}\right\|_2^2$$

- Alternating gradient descent approach to solving penalty formulation leads to following iterative algorithm:

```
while not converged:
```

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{f,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \frac{\rho}{2}\left\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\right\|_2^2$$

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{g,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} L_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \arg\min_{\boldsymbol{z}} g(\boldsymbol{z}) + \frac{\rho}{2}\left\|\boldsymbol{v} - \boldsymbol{z}\right\|_2^2$$

Review of ADMM for General Inverse Problems

<u>$\boldsymbol{x}$ – update:</u>

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{v}\|_2^2$$

$$\boldsymbol{x} \leftarrow \underbrace{\left(\boldsymbol{A}^T\boldsymbol{A} + \rho\boldsymbol{D}^T\boldsymbol{D}\right)^{-1}}_{\tilde{\boldsymbol{A}}}\underbrace{\left(\boldsymbol{A}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{v}\right)}_{\tilde{\boldsymbol{b}}}$$

- For general inverse problems, we don't necessarily have an efficient closed-form solution for this problem, like we did for the deconvolution problem

- Use matrix-free iterative solver, such as the conjugate gradient method, to solve $\tilde{\boldsymbol{A}}\boldsymbol{x} = \tilde{\boldsymbol{b}}$ (e.g., `scipy.sparse.linalg.cg`)

Review of ADMM for General Inverse Problems

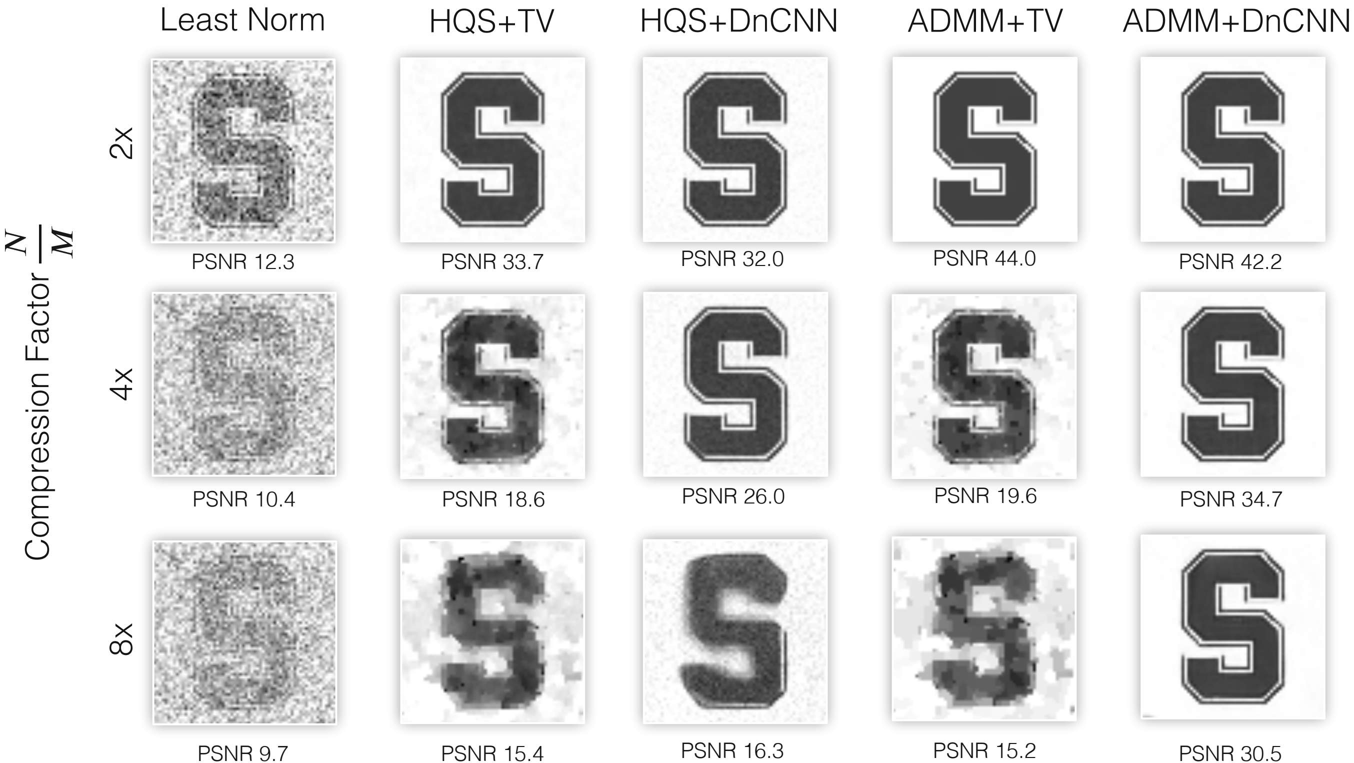
$\boldsymbol{z}$ – update for TV regularizer in closed form:

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\|\cdot\|_1,\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda \|\boldsymbol{z}\|_1 + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2 = \mathcal{S}_\kappa(\boldsymbol{v})$$

$\boldsymbol{z}$ – update for denoising-based regularizer in closed form:

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{\mathcal{D},\rho}(\boldsymbol{v}) = \arg\min_{\boldsymbol{z}} \lambda \Psi(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{v} - \boldsymbol{z}\|_2^2 = \mathcal{D}\left(\boldsymbol{v}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

ADMM – Results



Short tangent on Half Quadratic Splitting (HQS)
(Another solver for constrained optimization problems)

The Half-quadratic Splitting (HQS) Method

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_x \ \frac{1}{2}\left\| \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x} \right\|_2^2 + \lambda \Psi(\boldsymbol{x})$$

weight of regularizer

The Half-quadratic Splitting (HQS) Method

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_x \ \frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \lambda\Psi(\boldsymbol{x})$$

weight of regularizer

- Reformulate as:

$$\text{minimize}_{\{x,z\}} \ \underbrace{\frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\Psi(\boldsymbol{z})}_{g(\boldsymbol{z})}$$

$$\text{subject to} \ \ \boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} = 0$$

The Half-quadratic Splitting (HQS) Method

- Objective or "loss" function of general inverse problem:

$$\text{minimize}_x \; \frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2 + \underbrace{\lambda}_{\text{weight of regularizer}}\Psi(\boldsymbol{x})$$

- Reformulate as:

$$\text{minimize}_{\{x,z\}} \; \underbrace{\frac{1}{2}\left\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\right\|_2^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\Psi(\boldsymbol{z})}_{g(\boldsymbol{z})}$$

$$\text{subject to} \;\; \boldsymbol{D}\boldsymbol{x} - \boldsymbol{z} = 0$$

- Remove constraints using penalty term (equivalent for large ρ):

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \underbrace{\frac{\rho}{2}\left\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z}\right\|_2^2}_{\text{penalty term}}$$

The Half-quadratic Splitting (HQS) Method

$$L_\rho(\boldsymbol{x}, \boldsymbol{z}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

- Alternating gradient descent approach to solving penalty formulation leads to following iterative algorithm:

```
while not converged:
```

$$\boldsymbol{x} \leftarrow \mathrm{prox}_{f,\rho}(\boldsymbol{z}) = \arg\min_x L_\rho(\boldsymbol{x}, \boldsymbol{z}) = \arg\min_x f(\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{g,\rho}(\boldsymbol{D}\boldsymbol{x}) = \arg\min_z L_\rho(\boldsymbol{x}, \boldsymbol{z}) = \arg\min_z g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

The Half-quadratic Splitting (HQS) Method

$$L_\rho(\boldsymbol{x},\boldsymbol{z}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{z}\|_2^2$$

- Alternating gradient descent approach to solving penalty form

- Steps not tied together with dual variable

- Can be very sensitive to the penalty parameter, requiring more tuning than ADMM (technically, penalty needs to go to infinity)

`while n`

$\boldsymbol{x} \leftarrow$ p ... $- \boldsymbol{z}\|_2^2$

$$\boldsymbol{z} \leftarrow \mathrm{prox}_{g,\rho}(\boldsymbol{D}\boldsymbol{x}) = \arg\min_z L_\rho(\boldsymbol{x},\boldsymbol{z}) = \arg\min_z g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{z}\|_2^2$$

Applications of Compressive Imaging

Compressive Medical Imaging

- reduce acquisition time, radiation exposure, or allow for more patients in same time, …

- examples: x-ray computed tomography and MRI

Computed Tomography (CT)

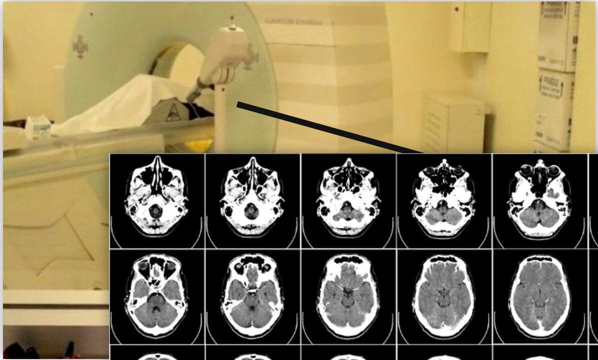
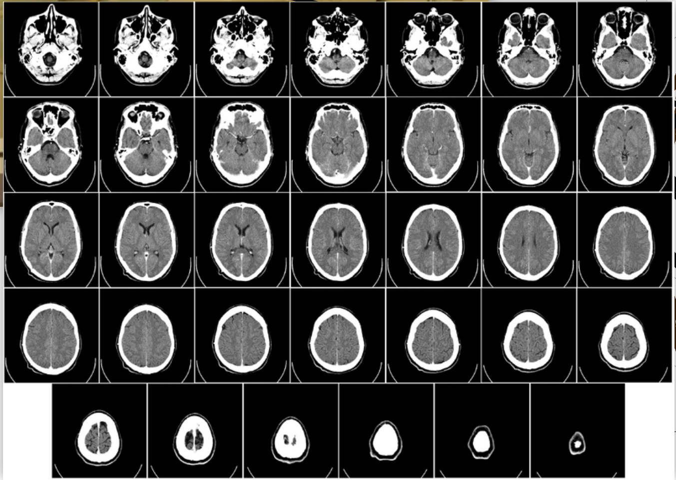
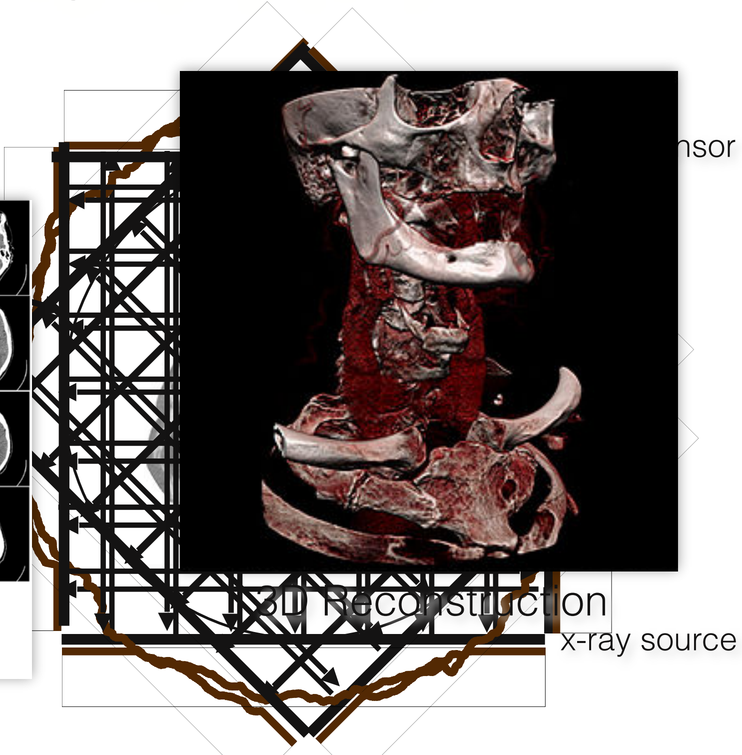


Image: Wikipedia



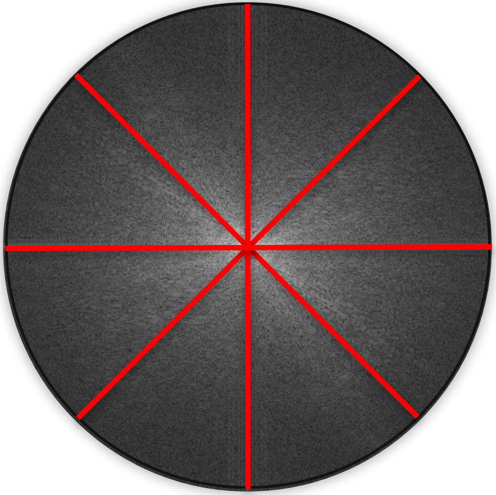
Reconstructed 2D Slices



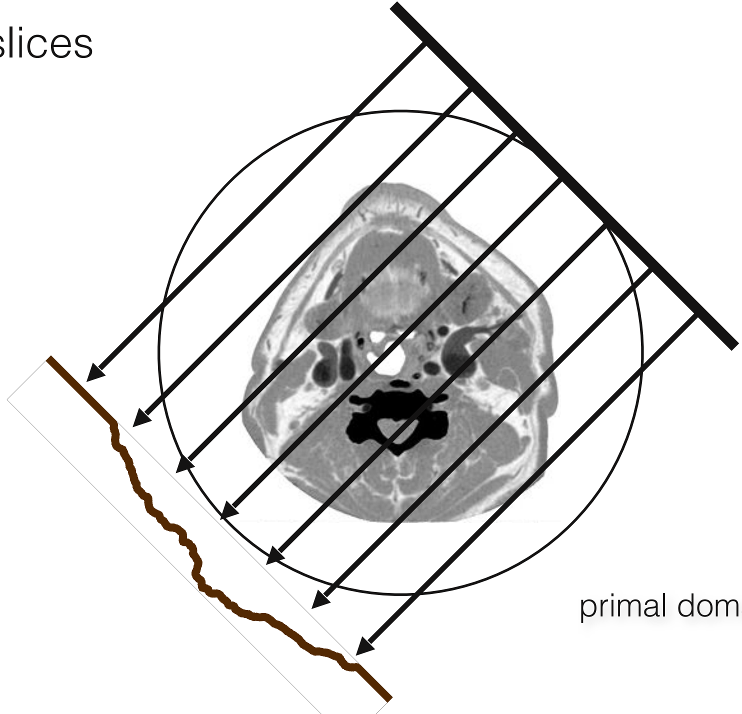
3D Reconstruction

Computed Tomography – Fourier Slice Theorem

- measurements = Fourier slices
- compressive CT: e.g. fewer slices



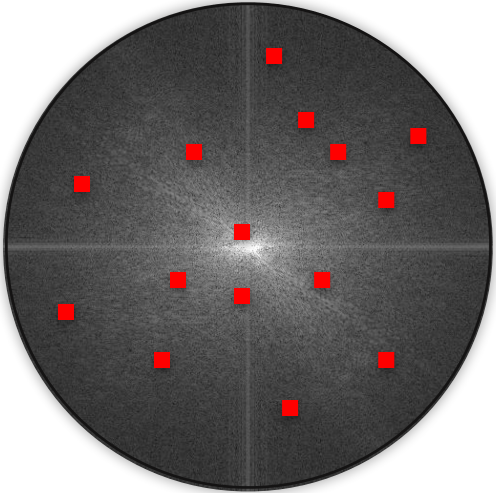
frequency domain



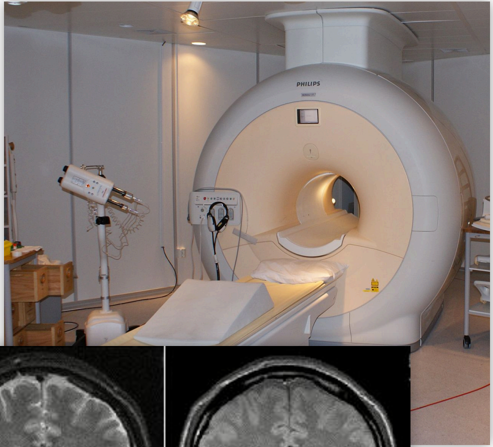
primal domain

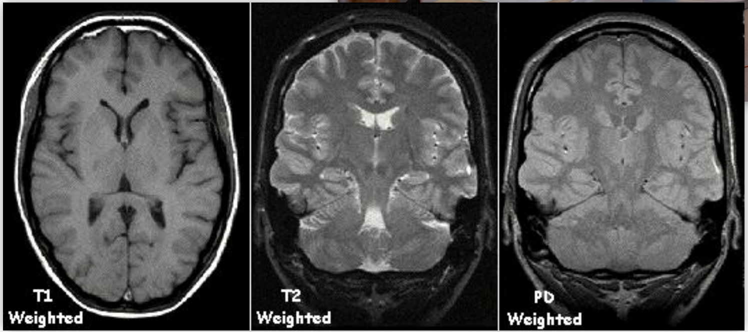
Magnetic Resonance Imaging

- measurements = (random) Fourier coefficients
- compressive MRI: fewer Fourier coefficients



frequency domain





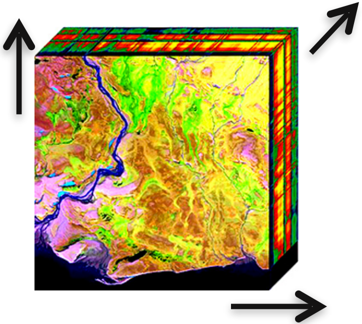
wikipedia

Compressive Imaging: CT & MRI

- people in bio-medical imaging often hesitant about priors:
  - few guarantees for success
  - if reconstruction breaks, not clear how exactly
  - is that feature a reconstruction artifact or the thing I'm looking for?

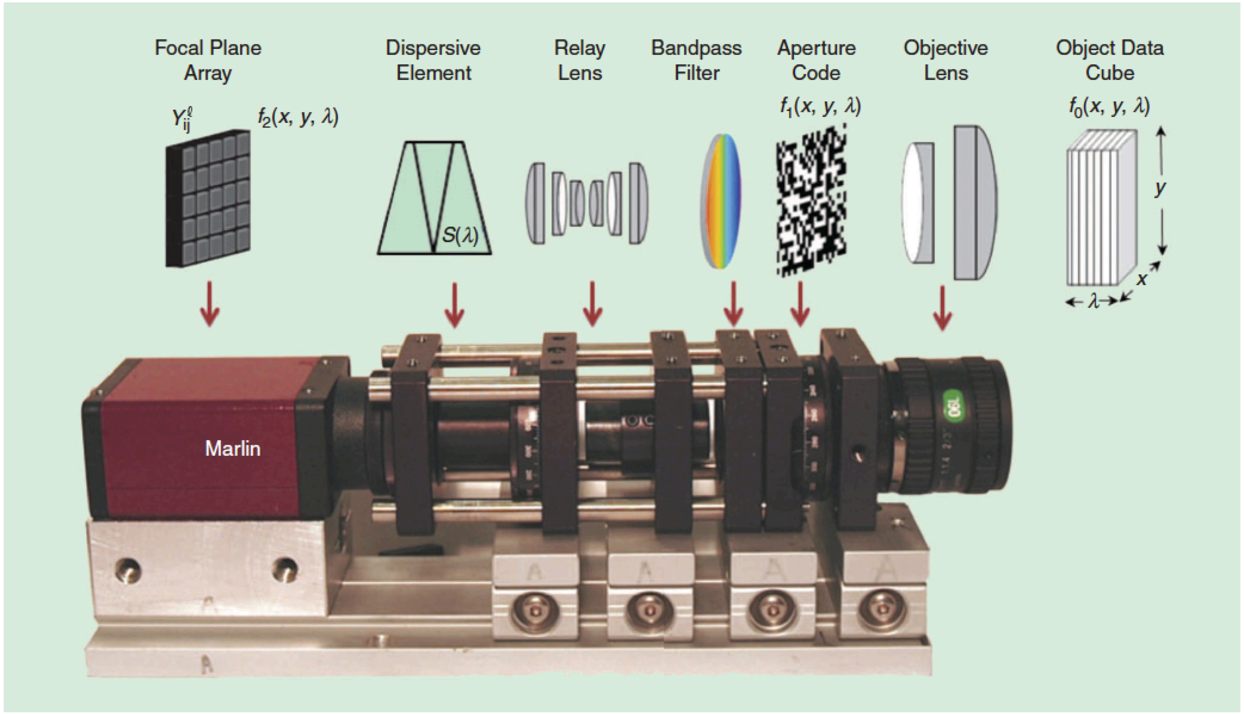
Compressive Hyperspectral Imaging

- motivation:



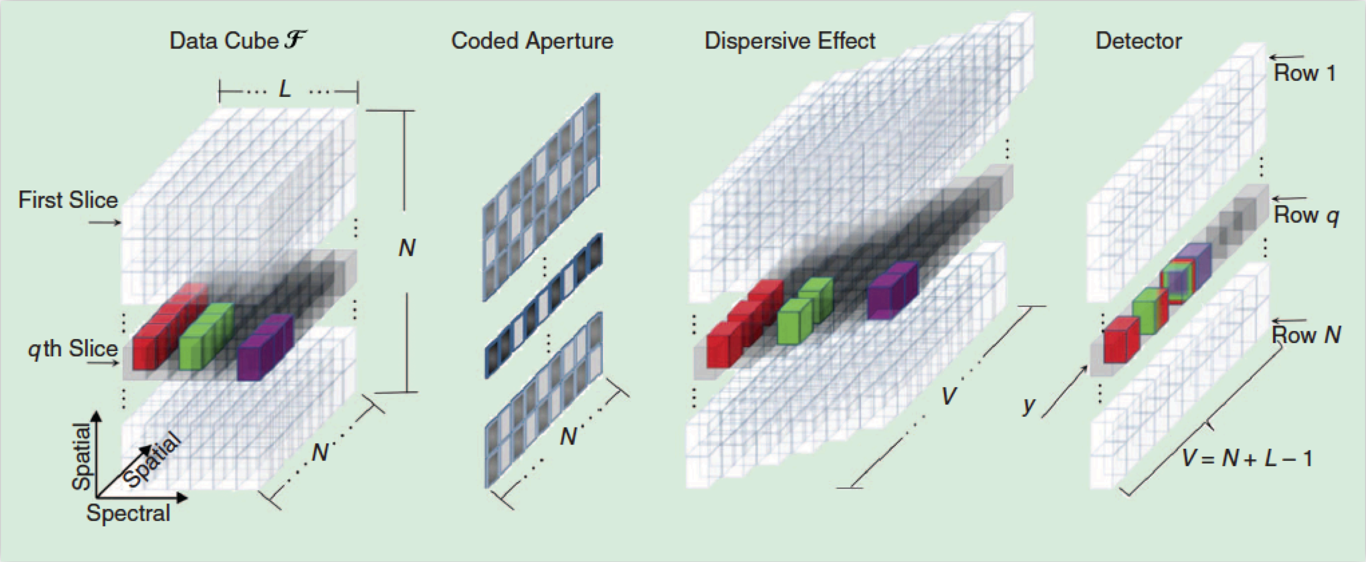
- conventional: either scan over xy or over lambda!
- idea: capture hyperspectral datacube with a single, coded image – use compressive sensing to reconstruct
- first approach: CASSI (coded aperture snapshot spectral imager), Wagadarikar 2008

Compressive Hyperspectral Imaging



Arce et al. 2014

Compressive Hyperspectral Imaging



Arce et al. 2014

Compressive Hyperspectral Imaging

- moderate quality for snapshot, but good quality for coded multi-shot
- applications: remote sensing, cultural heritage, …

Arce et al. 2014

Compressive Imaging Everywhere

- metamaterials
- THz imaging
- x-ray imaging
- thermal IR
- ultra-fast imaging
- not as much on compressive coherent imaging (could be interesting for course projects: OCT, holography, …)
- …

Notes

- compressive imaging is an exploding area: check COSI, ICCP, CVPR, ICCV conferences, other optics journals and conferences
- most variants of compressive imaging problems can be implemented with ADMM
- check lecture notes online to help with homework
- Increasingly we want to learn the sensing matrices, reconstruction using neural networks and datasets…

References and Further Reading

Must read: course notes on Image Deconvolution with ADMM & course notes on compressive imaging

Adam

- D. Kingma, J. Ba “Adam: A method for stochastic optimization”, ICLR 2015

ADMM

- S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein “Distributed optimization and statistical learning via the alternating direction method of multipliers”, Foundation and Trends in Machine Learning, 2001

Single-pixel Imaging

- M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, R. Baraniuk “Single-pixel imaging via compressive sampling”, IEEE Signal Processing Magazine 2008

TV Prior and Extensions

- L. Rudin, S. Osher, E. Fatemi “Nonlinear total variation-based noise removal algorithm”, Physica D, 1992
- A. Levin, Y. Weiss, F. Durand, W. Freeman "Understanding and evaluating blind deconvolution algorithms”, CVPR 2009
- D. Krishnan, R. Fergus “Fast Image Deconvolution using Hyper-Laplacian Priors”, NIPS 2009
- K. Bredies, K. Kunisch, T. Pock “Total Generalized Variation”, Technical Report 2009
- S. Lefkimmiatis, J. Ward, M. Unser "Hessian Schatten-Norm Regularization for Linear Inverse Problems", IEEE Transactions on Image Processing 2003

Unrolled Optimization

- S. Diamond, V. Sitzmann, F. Heide, G. Wetzstein “Unrolled optimization with deep priors”, arxiv, 2017