Image Deconvolution with the Half-quadratic Splitting (HQS) Method

EE367/CS448I: Computational Imaging

stanford.edu/class/ee367

Lecture 10

Gordon Wetzstein

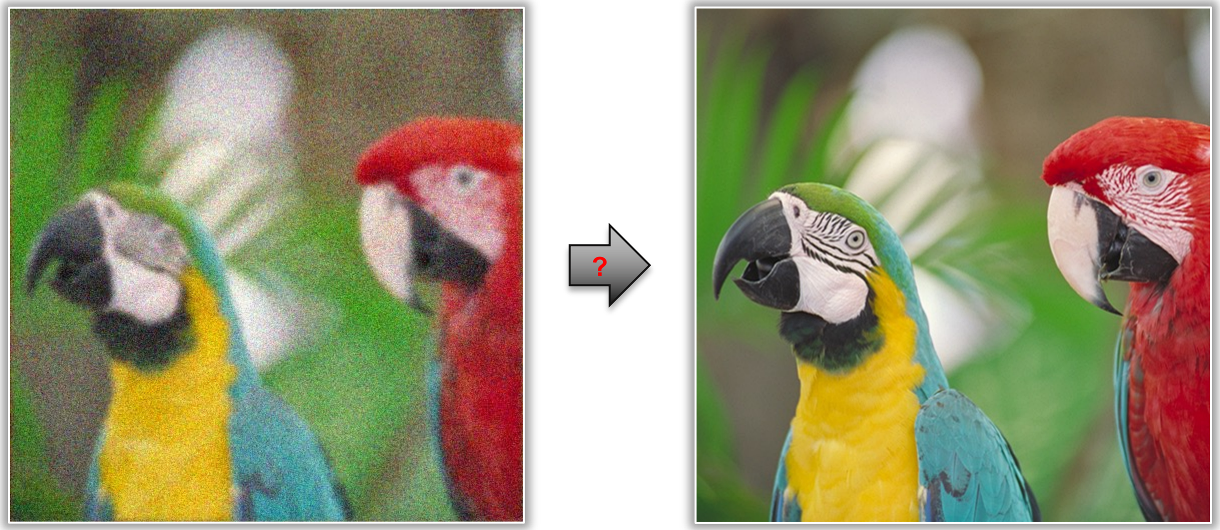
Stanford University

Overview

- Brief review of deconvolution with inverse/Wiener filtering
- A Bayesian perspective of inverse problems
- Image priors/regularization and total variation
- The half-quadratic splitting (HQS) method
- Image deconvolution with HQS
- Outlook on unrolled optimization with learned priors

Must read: course notes on Image Deconvolution with the Half-quadratic splitting method!

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

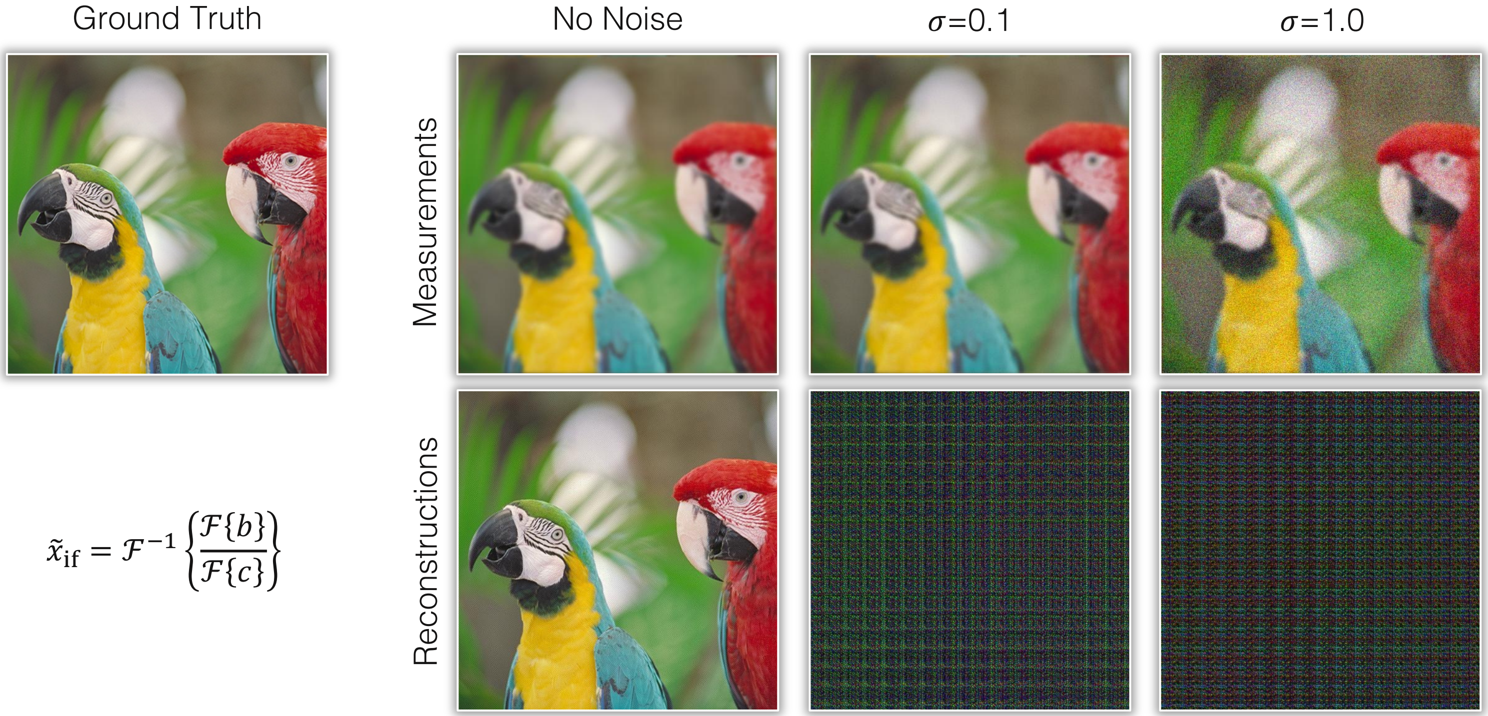
2D target image

Image Deconvolution – Brief Review

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

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

Image Deconvolution – Inverse Filtering



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

Image Deconvolution – Wiener Filtering

Ground Truth

No Noise | σ=0.1 | σ=1.0

Measurements

Reconstructions

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

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**

A Bayesian Perspective of Inverse Problems

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

- Interpret as random variables:
$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{x}_i, 0), \ \boldsymbol{\eta}_i \sim \mathcal{N}(0, \sigma^2)$$
$$\boldsymbol{b}_i \sim \mathcal{N}\left((\boldsymbol{A}\boldsymbol{x})_i, \sigma^2\right)$$

- Probability of observation i:
$$p(\boldsymbol{b}_i | \boldsymbol{x}_i, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\boldsymbol{b}_i - (\boldsymbol{A}\boldsymbol{x})_i)^2}{2\sigma^2}}$$

- Joint probability of all observations:
$$p(\boldsymbol{b} | \boldsymbol{x}, \sigma) = \prod_{i=1}^{M} p(\boldsymbol{b}_i | \boldsymbol{x}_i, \sigma) \propto e^{-\frac{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2}{2\sigma^2}}$$

A Bayesian Perspective of Inverse Problems

- Bayes' rule:

$$p(\boldsymbol{x}|\boldsymbol{b},\sigma) = \frac{p(\boldsymbol{b}|\boldsymbol{x},\sigma)\boldsymbol{p}(\boldsymbol{x})}{p(\boldsymbol{b})} \propto p(\boldsymbol{b}|\boldsymbol{x},\sigma)p(\boldsymbol{x})$$

posterior: $p(\boldsymbol{x}|\boldsymbol{b},\sigma)$

image formation model or likelihood: $p(\boldsymbol{b}|\boldsymbol{x},\sigma)$

prior: $p(\boldsymbol{x})$

- Maximum-a-posterior (MAP) solution:

$$\boldsymbol{x}_{MAP} = \arg\min_x \; -\log\big(p(\boldsymbol{x}|\boldsymbol{b},\sigma)\big)$$

$$= \arg\min_x \; -\log\big(p(\boldsymbol{b}|\boldsymbol{x},\sigma)\big) - \log\big(p(\boldsymbol{x})\big)$$

$$= \arg\min_x \; \frac{1}{2\sigma^2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \Psi(\boldsymbol{x})$$

A Bayesian Perspective of Inverse Problems

regularizer ↓ prior ↓

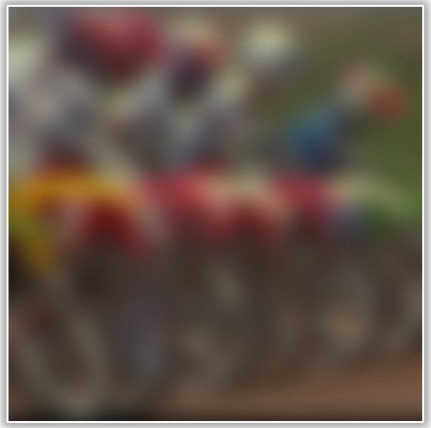
- Terminology: $\Psi(\boldsymbol{x}) = -\log\big(p(\boldsymbol{x})\big)$

data fidelity term ↓ regularization term ↓

$$\boldsymbol{x}_{MAP} = \arg\min_{\boldsymbol{x}} \frac{1}{2\sigma^2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \Psi(\boldsymbol{x})$$

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

$$\sqrt{(\boldsymbol{D}_x\boldsymbol{x})_i^2 + (\boldsymbol{D}_y\boldsymbol{x})_i^2}$$

easier: anisotropic

$$\sqrt{(\boldsymbol{D}_x\boldsymbol{x})_i^2} + \sqrt{(\boldsymbol{D}_y\boldsymbol{x})_i^2}$$

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:

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

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

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:

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

↑ 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
- The "fine print": convenient but doesn't always converge well

The Half-quadratic Splitting (HQS) Method

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

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

weight of regularizer

- Reformulate as:

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

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

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

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

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:
```

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

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

HQS for Image Deconvolution with TV

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

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

$\boldsymbol{x} \in \mathbb{R}^N$	unknown sharp image
$\boldsymbol{C} \in \mathbb{R}^{N \times N}$	circulant convolution matrix for known kernel $\boldsymbol{c}$
$\boldsymbol{z} \in \mathbb{R}^{2N}$	slack 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

HQS for Image Deconvolution with TV

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

```
while not converged:
```

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

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

HQS for Image Deconvolution with TV

<u>$\boldsymbol{x}$ – update:</u>

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

reformulate

$$= \frac{1}{2}(\boldsymbol{Cx}-\boldsymbol{b})^T(\boldsymbol{Cx}-\boldsymbol{b}) + \frac{\rho}{2}(\boldsymbol{Dx}-\boldsymbol{z})^T(\boldsymbol{Dx}-\boldsymbol{z})$$

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

find solution by setting gradient to 0

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

closed-form solution

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

HQS for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

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

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

Exploit duality of algebraic & signal processing interpretation

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

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

$$\boldsymbol{D}^T\boldsymbol{z} = \boldsymbol{D}_x^T\boldsymbol{z}_1 + \boldsymbol{D}_y^T\boldsymbol{z}_2 \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{d_x\} *\cdot \mathcal{F}\{z_1\} + \mathcal{F}\{d_y\} *\cdot \mathcal{F}\{z_2\}\right\}$$

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

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

$$\underline{\boldsymbol{C}^T\boldsymbol{b} + \rho\boldsymbol{D}^T\boldsymbol{z}} \Leftrightarrow \mathcal{F}^{-1}\left\{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{z_1\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{z_2\}\right)\right\}$$

HQS for Image Deconvolution with TV

$\boldsymbol{x}$ – update:

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

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

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

$$\text{prox}_{\|\cdot\|_2,\rho}(\boldsymbol{z}) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{z_1\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{z_2\}\right)}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho\left(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{d_x\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{d_y\}\right)}\right\}$$

↑ can pre-compute most parts

$z_1 = \boldsymbol{z}(1{:}\,N), z_2 = \boldsymbol{z}(N+1{:}\,2N)$

HQS for Image Deconvolution with TV

$\boldsymbol{z}$ – update:

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

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

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

$\kappa = \lambda/\rho$

$\boldsymbol{v} = \boldsymbol{Dx}$

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

HQS for Image Deconvolution with Denoiser

$\boldsymbol{x}$ – update:

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

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

no matrix $\boldsymbol{D}$!

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

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

HQS for Image Deconvolution with Denoiser

$\boldsymbol{z}$ – update:

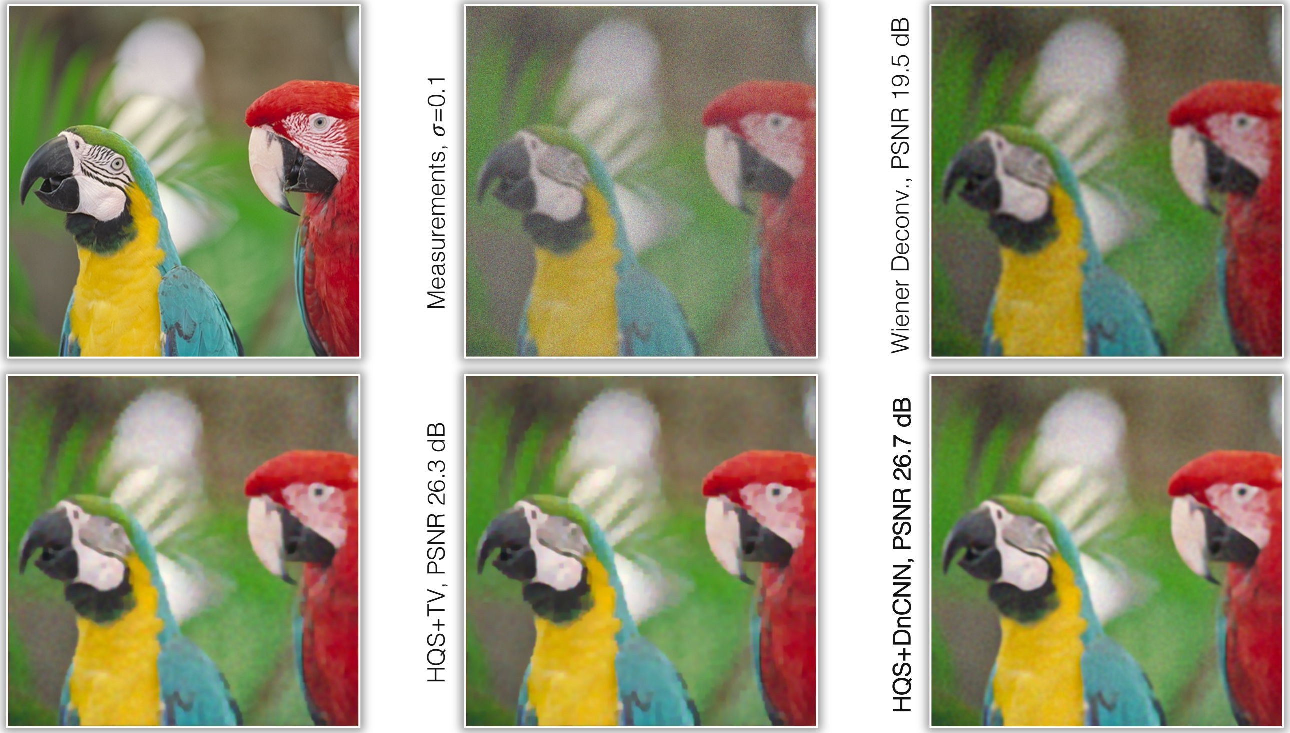
$$\boldsymbol{z} \leftarrow \text{prox}_{\mathcal{D},\rho}(\boldsymbol{x}) = \arg\min_{\boldsymbol{z}} \lambda\Psi(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

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

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

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

Image Deconvolution with HQS



Target Image

Measurements, σ=0.1

Wiener Deconv., PSNR 19.5 dB

Adam+TV, PSNR 26.1 dB

HQS+TV, PSNR 26.3 dB

HQS+DnCNN, PSNR 26.7 dB

Image Deconvolution with HQS

HQS for deconvolution with denoiser

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

HQS for deconvolution with TV

initialize ρ and λ
$x = zeros\left(W, H\right);$
$z = zeros\left(W, H\right);$
for $k = 1$ **to** max_iters **do**
    $x = \mathbf{prox}_{\|\cdot\|_2,\rho}\left(z\right) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{z_1\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{z_2\})}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho(\mathcal{F}\{d_x\}^* \cdot \mathcal{F}\{d_x\} + \mathcal{F}\{d_y\}^* \cdot \mathcal{F}\{d_y\})}\right\}$
    $z = \mathbf{prox}_{\|\cdot\|_1,\rho}\left(\mathbf{Dx}\right) = \mathcal{S}_{\lambda/\rho}\left(\mathbf{Dx}\right)$
end for

HQS - Convergence Criterion

- Run or “unroll” HQS for *K* iterations
- Run until change in residual between iterations is < threshold

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$\vdots$$

Outlook on Unrolled Optimization

- Run or “unroll” HQS for K iterations
- Interpret as unrolled feedforward network:

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

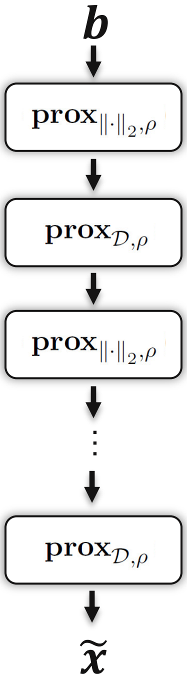
$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

$$x = \mathbf{prox}_{\|\cdot\|_2,\rho}(z) = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{b\} + \rho\mathcal{F}\{z\}}{\mathcal{F}\{c\}^* \cdot \mathcal{F}\{c\} + \rho}\right\}$$

$$z = \mathbf{prox}_{\mathcal{D},\rho}(\mathbf{x}) = \mathcal{D}\left(\mathbf{x}, \sigma^2 = \frac{\lambda}{\rho}\right)$$

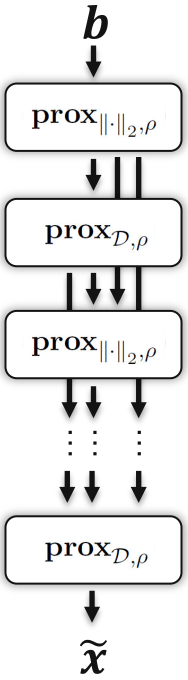
$$\vdots$$



[Diamond et al. 2017]

Outlook on Unrolled Optimization

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



Benefits over unrolled optimization

- Learnable parameters: $\lambda^{(k)}, \rho^{(k)}$, denoiser $\mathcal{D}^{(k)}$
- 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]

References and Further Reading

Must read: EE367 course notes on Image Deconvolution with the Half-quadratic splitting method!

Optional read: EE367 course notes on Noise, Denoising, and Image Reconstruction with Noise

Adam

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

HQS

- D. Geman and C. Yang “Nonlinear image recovery with half-quadratic regularization”, IEEE Transactions on Image Processing, 1995

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