Solving Realistic Image Restoration and Reconstruction Problems Through Shearlet-based L1-Regularization

Bart Goossens, Jan Aelterman, Hiêp Luong, Aleksandra Pižurica and Wilfried Philips

Ghent University, Belgium
TELIN – IPI – IBBT

Dec. 2, 2010
Overview

- Introduction: image restoration & reconstruction
  - Problem statement and goal
  - Challenges
  - Applications

- Methodology for image restoration and reconstruction
  - Bregman-based optimization
  - Regularization in the discrete shearlet domain

- Results

- Conclusion
• Image reconstruction

- Demosaicing
- Color filter arrays
- MRI
- MRI spiral trajectory
- Sinogram
- CT

Introduction
• Image restoration

Why is this so useful?
Blur+noise
Compression
What we want
What we get: Noise + Blur + Compression
Goal:

- Recovery / reconstruction of “original images” from observed degraded images.
- Various degradations:
  - Noise
  - Blur
  - Compression artifacts
  - Missing or “destroyed” data
  - Discolorations
  - …
Digital photography: challenges

- More pixels = more noise
- Noise characteristics are significantly modified by the different post-processing steps
- White additive Gaussian noise model: ineffective!
Methodology for image restoration and reconstruction

- Image restoration in an iterative way:

  Degraded input image → 1. Restoration Algorithm → "Restored" image

  2. Degradation model

  3. Prior model
1. Restoration algorithm – Bregman iterations

- Restoration algorithm:
  - Solving the unconstrained problem

$$\min_x J(x), \quad \text{s.t.} \quad H(x; y) < \epsilon$$

- Image as sparse as possible
- And consistent with degraded image (according to degradation model)
1. Restoration algorithm – Bregman iterations

- Restoration algorithm:
  - Solution: Bregman iteration [Bregman, 1967]

\[
\min_x J(x), \quad \text{s.t.} \quad \|Ax - y\|_2^2 < \epsilon
\]

- Init: \( b_0 = y \)
- Repeat until \( \|Ax - y\|_2^2 < \epsilon \)
- \[
\begin{align*}
  x_{i+1} &= \arg\min_x J(x) + \frac{\lambda}{2} \|Ax - b_i\|_2^2 \\
  b_{i+1} &= b_i + (b_0 - Ax_{i+1})
\end{align*}
\]

Next:
- Split-Bregman [Goldstein and Osher, 2008]
- SALSA – split augmented Lagrangian [Afonso et al., 2010]
- Proximal theory [Combettes and Pesquet, 2010]

Very similar algorithms!
1. Restoration algorithm – Bregman iterations

• Split-Bregman method [Goldstein and Osher, 2009]:

Sparsity prior: \( J(x) = |Sx|_1 \)

1) Minimization problem to solve:

\[
x_{i+1} = \arg \min \ |Sx|_1 + \|Ax - b_i\|_2^2
\]

2) Introduce auxiliary variable \( d = Sx \) and apply the Bregman iteration

\[
(x_{i+1}, d_{i+1}) = \min_{(x,d)} |d|_1 + H(x; y) + \frac{\lambda}{2} \|Sx - d^i - b_i\|_2^2
\]

\[
b_{i+1} = b_i + (Sx_{i+1} - d_{i+1}).
\]
1. Restoration algorithm – Bregman iterations

- Split-Bregman method [Goldstein and Osher, 2009]:

  Sparsity prior: $J(x) = \|Sx\|_1$

  Split:

  $$(x_{i+1}, d_{i+1}) = \min_{(x,d)} |d|_1 + \|Ax - b_i\|_2^2 + \frac{\lambda}{2} \|Sx - d_i - b_i\|_2^2$$

  Into:

  $$\begin{aligned}
  x_{i+1} &= \arg\min_x \|Ax - b_i\|_2^2 + \frac{\lambda}{2} \|Sx - d_i - b_i\|_2^2, \\
  d_{i+1} &= \arg\min_d |d|_1 + \frac{\lambda}{2} \|Sx - d_i - b_i\|_2^2.
  \end{aligned}$$
1. Restoration algorithm – Bregman iterations

\[
\min_{x} J(x), \quad \text{s.t. } H(x; y) < \epsilon
\]

Data fitting function \[ \|y - Ax\|_2^2 \]

Sparsity prior: \[ J(x) = |Sx|_1 \]

initialize \[ x_1 = y, \quad d_1 = 0, \quad b_1 = 0, \quad y_1 = y \]

while \[ \|y_i - Ax_i\|_2 \geq \sigma \]

while \[ \|x_{i+1} - x_i\|_2 > \text{tolerance} \]

\[ x_{i+1} = \arg\min_x \|y_i - Ax\|_2^2 + \frac{1}{2} \|Sx - d_i - b_i\|_2^2, \quad \text{(step I)} \]

\[ d_{i+1} = \arg\min_d |d|_1 + \frac{1}{2} \|Sx - d - b_i\|_2^2, \quad \text{(step II)} \]

\[ b_{i+1} = b_i + (Sx_{i+1} - d_{i+1}). \quad \text{(step III)} \]

end

\[ y_{i+1} = y_i + y - Ax_i \]

Adding the error back to the image.
Image restoration using Bregman iterations:

1. Restoration Algorithm
   - Degraded input image
   - Degradation model
   - Prior model

“Restored” image
2. Degradation model for a digital still camera

- Degradation modeling: more general data fitting functions
  - Correlated Gaussian noise:
    \[ H(x; y) = \frac{1}{2} (y - x)^T C_n^{-1} (y - x) \]
  - Blurring operations (joint denoising & deblurring):
    \[ H(x; y) = \frac{1}{2} (y - Ax)^T C_n^{-1} (y - Ax) \]
    Block-Circulant or Block-Toeplitz blurring matrix
  - Gaussian modeling of signal-dependent noise:
    \[ H(x; y) = \frac{1}{2} \sum_{j=1}^{N} \left( \frac{y_j - \mu(x_j)}{\sigma^2(x_j)} \right)^2 \]
2. Degradation model for a digital still camera

- Clipped Poisson-distributed signal: \( y_j \sim \min(255, \text{Poisson}(x_j)) \)

Data fitting term:

\[
H(x; y) = \frac{1}{2} \sum_{j=1}^{N} \left( \frac{y_j - \mu(x_j)}{\sigma^2(x_j)} \right)^2
\]

where

\[
\begin{align*}
\mu(x) &= \mathbb{E}[y|x] = M_1(x) + \gamma(x) \\
\sigma^2(x) &= M_2(x) - 2(\mu(x) - \gamma(x))\mu(x) + \mu^2(x) - \gamma^2(x) \\
M_1(x) &= \sqrt{\frac{x}{2\pi}} \left( e^{-\frac{1}{2}|x|} - e^{-\frac{(x-255)^2}{2|x|}} \right) + \left( \frac{255}{2} - x \right) + \frac{255 - x}{2} \text{erf} \left( \frac{x - 255}{\sqrt{2x}} \right) + \frac{x}{2} \text{erf} \left( \frac{\sqrt{x}}{2} \right), \\
M_2(x) &= -\sqrt{\frac{x}{2\pi}} \left( xe^{-\frac{1}{2}|x|} + (255 - x)e^{-\frac{(x-255)^2}{2|x|}} \right) + \frac{(255 - x)^2}{2} + \frac{x^2}{2} + \frac{|x| - x^2}{2} \text{erf} \left( \frac{\sqrt{x}}{2} \right) - \frac{|x| - (x-1)^2}{2} \text{erf} \left( \frac{x - 1}{\sqrt{2x}} \right)
\end{align*}
\]
• Image restoration using Bregman iterations:
3. Prior model: Discrete Shearlet Transform domain

$$\min_x J(x), \quad \text{s.t. } H(x; y) < \epsilon$$

Sparsity prior:

$$J(x) = |Sx|_1$$

Shearlet transform

Mother shearlet function

Derived shearlet functions

Digital shearing

Dilation
3. Prior model: Discrete Shearlet Transform domain

\[
\min_{x} J(x), \quad \text{s.t.} \quad H(x; y) < \epsilon
\]

Sparsity prior:

\[
J(x) = |Sx|_1
\]
3. Prior model: Discrete Shearlet Transform domain

http://telin.ugent.be/~bgoossen/CplxWaveletDemo/ShearletDemo.html
Overview

• Image restoration
  - Problem statement and goal
  - Challenges
  - Applications

• Methodology for image restoration and reconstruction
  - Bregman-based optimization
  - Regularization in the discrete shearlet domain

• Results

• Conclusion
Results

• 1. Removal of signal-dependent noise and de-biasing (de-clipping)
Removal of signal-dependent noise + debiasing

Noisy input image
Bregman TV
Bregman SH
Removal of signal-dependent noise + debiasing

Denoised only

Denoised + debiased

Goossens et al., Solving Realistic Image Restoration and Reconstruction Problems Through Shearlet-based L1-Regularization, BIRS workshop, Nov. 28 - Dec. 3, Banff, Canada
Removal of signal-dependent noise + debiasing
2. Suppression of correlated noise from images
Removal of correlated Gaussian noise
3. Demosaicing
Demosaicing (preliminary result)
4. Parallel MRI Reconstruction
Subsampled MRI reconstruction:

• Non-Cartesian grids

• Sub-Nyquist Sampling/Noise: solve optimization problem

• Finding a sparsifying transform
Goossens et al., Solving Realistic Image Restoration and Reconstruction Problems Through Shearlet-based L1-Regularization, BIRS workshop, Nov. 28 - Dec. 3, Banff, Canada

Results: Parallel MRI reconstruction

- 8 coils, 25% sub-Nyquist sampled, 20dB SNR

1 (of 8) aliased single-coil images

SENSE
[Pruessman et al., 1999]
PSNR = 17.7dB

COMPASS
[Aelterman, 2010a]
PSNR = 29.1dB
Results: Parallel MRI reconstruction

- 4 coils, 10% sub-Nyquist sampled
Results: Parallel MRI reconstruction

- 4 coils, 10% sub-Nyquist sampled
Results: Parallel MRI reconstruction
Conclusion

• Adaptation to realistic problems requires an algorithm per problem!

• Generic restoration/reconstruction algorithm consisting of
  1. Degradation model (image domain)
  2. Image prior model (shearlet domain)
  3. Iterative solver for the minimization problem

• Shearlets: our “sparsification” tool

• Split-Bregman optimization:
  – Simple and elegant implementation (splitting)
  – Allows to solve several sub-problems **jointly** at once (e.g. demosaicing, noise, blur, …) – difficult with other techniques!
Thank you!

Questions?


