Image Denoising and Restoration by Constrained Regularization

G. Landi E. Loli Piccolomini F. Zama

Department of Mathematics, Bologna University
http://www.unibo.it

Genova,
February 2nd 2011

Image Deblurring and Denoising

Discrete Image Formation model

$$\mathbf{y}^\delta = \mathbf{H}\mathbf{x} + \delta$$

where

- $\mathbf{y}^\delta \in \mathbb{R}^N$ is the detected image of $N = n \times m$ pixels;
- $\mathbf{H} \in \mathbb{R}^{N\times N}$ $N = n \times m$ matrix:
  - **H** Identity matrix $\longrightarrow$ denoising
  - **H** Block-Toeplitz matrix describing the blur $\longrightarrow$ deblurring
- $\delta \in \mathbb{R}^N$ noise vector;
- $\mathbf{x} \in \mathbb{R}^N$ is the unknown image to be recovered

Ill Posed Problem $\longrightarrow$ Regularization

Regularization Methods

- Definition of a data fidelity functional:

$$\mathcal{J}(\mathbf{x}) = \|\mathbf{H}\mathbf{x} - \mathbf{y}^{\delta}\|_2^2$$

- Definition of regularization functional $\mathcal{R}(\mathbf{x})$:
  - Tikhonov: $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$. Where $\mathbf{L} = I$ of $\mathbf{L} = \nabla^2$ discrete Laplacian operator.
  - Total variation: $\mathcal{R}(\mathbf{x}) = TV_{\beta}(\mathbf{x})$
- Definition of minimization problem:
  - Constrained Least Squares (CLS)

$$\min_{\mathbf{x}} \mathcal{J}(\mathbf{x}), \quad \mathcal{R}(\mathbf{x}) \leq \gamma, \quad \gamma > 0,$$

  - Constrained minimization:

$$\min_{\mathbf{x}} \mathcal{R}(\mathbf{x}), \quad \mathcal{J}(\mathbf{x}) \leq \sigma, \quad \sigma > 0$$

Iterative Constrained Least Squares Method

Goals:

- Define an iterative algorithm that solves

$$\min_{\mathbf{x}} \|\mathbf{H}\mathbf{x} - \mathbf{y}^\delta\|_2^2, \quad s.t. \quad \mathcal{R}(\mathbf{x}) \leq \gamma \tag{1}$$

- Compute a smoothing parameter γ such that the solution of (1) is a good approximation of the solution $\mathbf{x}^*$ of the noiseless problem $\mathbf{H}\mathbf{x}^* = \mathbf{y}$ for the given regularization function: $\mathcal{R}(\mathbf{x})$.

> *If $\mathcal{R}(\mathbf{x})$ is a convex function then (1) is a convex optimization problem.*

Iterative Method

- Lagrangian function: $\mathcal{L}(\mathbf{x}, \lambda) \equiv \|\mathbf{Hx} - \mathbf{y}^\delta\|^2 + \lambda\left(\mathcal{R}(\mathbf{x}) - \gamma\right)$
- Dual problem:

$$\max_{\lambda} \left(\min_{x} \mathcal{L}(\mathbf{x}, \lambda) \right)$$

- Find $\hat{\lambda}$ s.t. $\nabla_\lambda \mathcal{L}_*(\hat{\lambda}) = 0$ where $\mathcal{L}_*(\hat{\lambda}) \equiv \min_x \mathcal{L}(\mathbf{x}, \hat{\lambda})$
- since $\nabla_\lambda \mathcal{L}_*(\lambda) \equiv \mathcal{R}(\mathbf{x}(\lambda)) - \gamma$

Find $\hat{\lambda}$ s.t. $\mathcal{R}(\mathbf{x}(\hat{\lambda})) - \gamma = 0$

where $\mathbf{x}(\hat{\lambda}) \equiv \hat{\mathbf{x}}$ s.t. $\mathbf{H}^t\mathbf{H}\hat{\mathbf{x}} + \hat{\lambda}\nabla_{\mathbf{x}}\mathcal{R}(\hat{\mathbf{x}}) - \mathbf{H}^t\mathbf{y}^\delta = 0$

The Iterative Method

- Solve the nonlinear equation $\mathcal{R}(\mathbf{x}(\lambda)) - \gamma = 0$ v.s. λ

$$\Downarrow$$

Define an iterative update of the lagrange multipliers.

- Case $\mathcal{R}(\mathbf{x}) \equiv \|x\|_2$

  T. F. Chan, J. A. Olkin, and D. W. Cooley, *Solving quadratically constrained least squares using black box solvers*, BIT **32** (1992), pp. 481–495.

- Case $\mathcal{R}(\mathbf{x}) \equiv \|x\|_p^p, \quad 1 < p < \infty$

  C. Cartis, N.I.M. Gould, and P.L. Toint. *Trust-Region and other regularizations of linear least squares problems*, BIT, **49(1)**, (2009) pp. 21-53.

- $\mathcal{R}(\mathbf{x}) \equiv \|D^\alpha(\mathbf{x})\|_2^2$, D^α discrete differential operator of order $\alpha = 0, 1, 2$

  E.Loli Piccolomini, F. Zama, An Iterative algorithm for large size least-Squares constrained regularization problems, submitted AMC

Nonlinear Solver

- Bisection Method
  - $\lambda_0 > 0$ given value s.t. $\mathcal{R}(\mathbf{x}(\lambda_0)) \leq \gamma$.
  - $\mathcal{R}(\mathbf{x})$ is a convex and twice continuosly differentiable function s.t.

$$0 < \mathcal{R}(\mathbf{x}(\lambda)) - \gamma \leq 0, \ \forall \lambda \geq \hat{\lambda}, \quad \mathcal{R}(\mathbf{x}(\lambda)) - \gamma > 0, \ \forall \lambda \in [0, \hat{\lambda}).$$

  - compute $(\mathbf{x}_k, \lambda_k)$ s.t.

$$\begin{array}{l} \mathbf{H}^t\mathbf{H}\mathbf{x}_k + \lambda_k \nabla_x \mathcal{R}(\mathbf{x}_k) - \mathbf{H}^t\mathbf{y}^\delta = 0 \\ \lambda_{k+1} = \lambda_k + sign(\mathcal{R}(\mathbf{x}_k) - \gamma)\frac{\lambda_0}{2^k} \end{array}, \quad k = 0, 1, 2, \ldots \qquad (2)$$

  - $(\mathbf{x}_k, \lambda_k)$ converge to $(\hat{\mathbf{x}}, \hat{\lambda})$ where $\hat{\mathbf{x}}$ solves problem (1) and $\hat{\lambda}$ is the relative Lagrange multiplier.

Nonlinear Solver

- Secant Method
  - $\lambda_0 > 0$ and $\lambda_1 > 0$ s.t. $\mathcal{R}(\mathbf{x}(\lambda_0)) - \gamma > 0$, $\mathcal{R}(\mathbf{x}(\lambda_1)) - \gamma > 0$.

$$\begin{array}{l} \mathbf{H}^t\mathbf{H}\mathbf{x}_k + \lambda_k \nabla_x \mathcal{R}(\mathbf{x}_k) - \mathbf{H}^t\mathbf{y}^\delta = 0 \\ \lambda_{k+1} = \lambda_k - \frac{\mathcal{R}(\mathbf{x}_k)-\gamma}{\mathcal{R}(\mathbf{x}_k)-\mathcal{R}(\mathbf{x}_{k-1})}(\lambda_k - \lambda_{k-1}) \end{array}, \quad k = 0, 1, 2, \ldots \qquad (3)$$

- Hybrid Method: Bisection $(k \geq 2)$ + Secant

Algorithm outline

1 Computation of the parameter γ:

2 Computation of the starting value λ_0.

3 Computation of the sequence $\{\mathbf{x}_k, \lambda_k\}$, $k = 1, 2 \ldots$ using the iterative procedure (2) + (3).

4 Stopping condition:

$$|\lambda_{k+1} - \lambda_k| \leq \tau_r |\lambda_k| + \tau_a, \quad \tau_a \approx 10^{-7}, \quad \tau_r \approx 10^{-4}$$

Parameter γ

- $\mathcal{R}(\mathbf{x}(\lambda))$ decreasing w.r. λ
- Let's define:

$$\hat{\gamma} \equiv argmin_{\gamma} \frac{\|\mathbf{x}_\gamma - \mathbf{x}^*\|}{\|\mathbf{x}^*\|}$$

  where

$$\mathbf{x}_\gamma \equiv argmin\{\|H\mathbf{x} - \mathbf{y}^\delta\|, \quad s.t. \;\; \mathcal{R}(\mathbf{x}) \leq \gamma\}$$

  Observations:

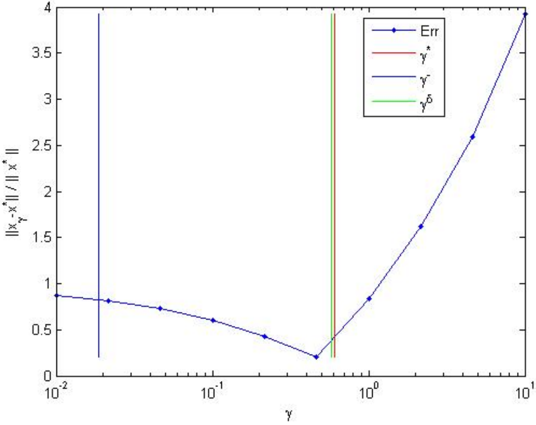
  1. $\tilde{\gamma} < \hat{\gamma}$ if $\tilde{\gamma} \equiv \mathcal{R}(\tilde{\mathbf{x}})$ where $\tilde{\mathbf{x}}$ smooth approximation by low pass filtering.
  2. $\hat{\gamma} \leq \gamma^\delta$ if $\gamma^\delta \equiv \mathcal{R}(\mathbf{y}^\delta)$

- Definition of γ:

$$\gamma = (1-\theta)\tilde{\gamma} + \theta\gamma^\delta, \quad \theta \in (0,1)$$

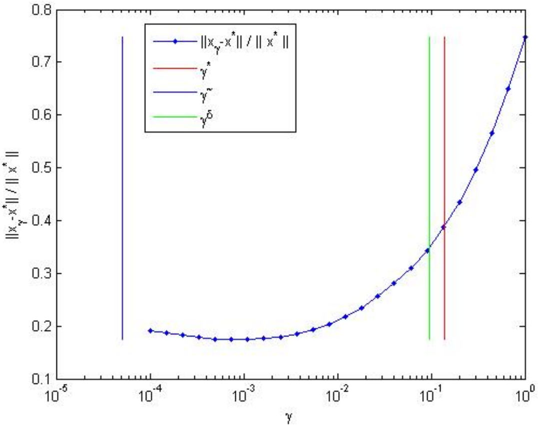
Examples

Deblurring problem: $\mathbf{L} = I$



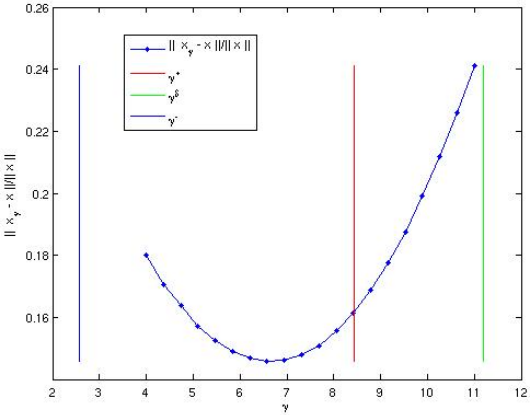
Examples

Deblurring problem: $\mathbf{L} = \nabla^2$



Examples

Denoising problem:



Starting value λ_0

1. Deblurring Problem
   - Symmetric boundary conditions $\Rightarrow$ **H** and **L** diagonalized by unitary discrete Fourier Transform Matrix [1]:
     $\mathbf{H} = \mathbf{F}^*\mathbf{D}\mathbf{F}, \quad \mathbf{L} = \mathbf{F}^*\mathbf{M}\mathbf{F}, \quad \mathbf{D} = diag(d_\ell), \quad \mathbf{M} = diag(\mu_\ell)$
   - Compute
     $$\lambda_0 = \min_\ell \left| \frac{(\mathbf{F}\mathbf{y}^\delta)_\ell}{d_\ell} \right|$$
2. Denoising Problem: Compute $\mathbf{x}_{Lw}$ by Gaussian lowpass filtering of y^δ.

$$\lambda_0 = \frac{\|\mathbf{x}_{Lw} - \mathbf{y}^\delta\|^2}{\mathcal{R}(\mathbf{x}_{Lw})}$$

[1] P.C. Hansen, J.G. Nagy, D.P. O'Leary, Deblurring Images, Siam, 2006

Step 3

1. Deblurring problem, solve the linear system:

$$\left(\mathbf{H}^t\mathbf{H} + \lambda_k \mathbf{L}^t\mathbf{L}\right)\mathbf{x}_k = \mathbf{H}^t\mathbf{y}^\delta \quad \Rightarrow \quad \mathbf{x}_k = \mathbf{F}^*\mathbf{\Phi}(\lambda_k)\mathbf{F}\mathbf{y}^\delta$$

$$\mathbf{x}_{\lambda_k} = \mathbf{F}^*\mathbf{\Phi}(\lambda_k)\mathbf{F}\mathbf{y}^\delta, \quad \mathbf{\Phi}(\lambda_k)_\ell = \frac{|d_\ell|^2}{|d_\ell|^2 + \lambda_k|\mu_\ell|^2}$$

$$\mathbf{H} = \mathbf{F}^*\mathbf{D}\mathbf{F}, \quad \mathbf{L} = \mathbf{F}^*\mathbf{M}\mathbf{F}, \quad \mathbf{D} = diag(d_\ell), \quad \mathbf{M} = diag(\mu_\ell)$$

Computational cost for k iterations: $(2 + k + 1)$ FFTs

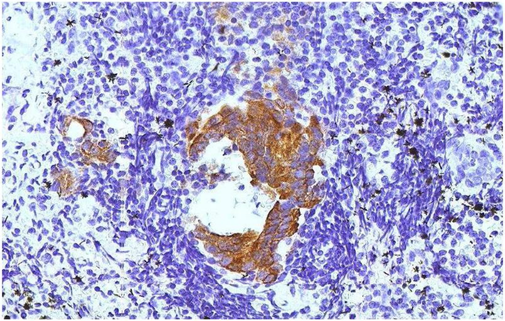
2. Denoising problem: $\mathbf{H} = \mathbf{I}$, $\mathcal{R}(\mathbf{x}) \equiv TV_\beta(\mathbf{x})$. Solve non linear equations system:

$$\mathbf{x} + \lambda\mathcal{L}(\mathbf{x})\mathbf{x} = \mathbf{H}^t\mathbf{y}^\delta, \quad \mathcal{L}(x)w = -\nabla\cdot\left(\frac{\nabla w}{\sqrt{|\nabla x|^2 + \beta^2}}\right)$$

Fixed Point + (Preconditioned) Conjugate Gradient Method. [2]

[2] C. R. Vogel and M. E. Oman, SIAM J. Sci. Comput., 17:227-238, 1996

Deblurring Experiment

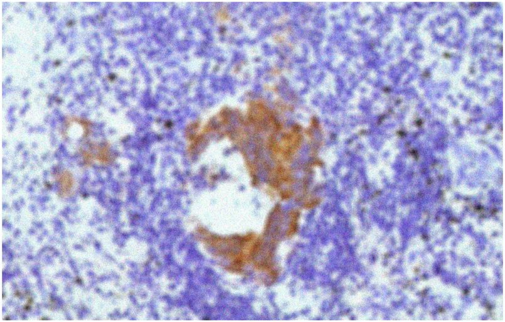


Original Image $\mathbf{x}^*$

506×800 RGB Image

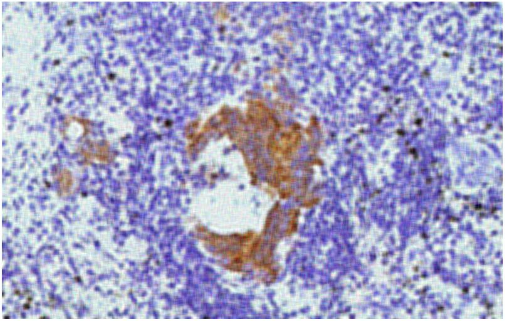
Deblurring Experiment

Blurred Noisy Image: $\mathbf{y}^\delta = \mathbf{y} + \delta, \quad \mathbf{y} = \mathbf{H}\mathbf{x}^*$ Gauss Blur, Gauss Noise



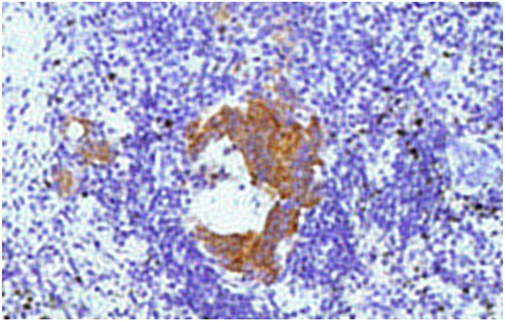
$$\frac{\|\mathbf{y}^\delta - \mathbf{x}^*\|}{\|\mathbf{x}^*\|} = 0.25, \quad SNR = 13$$

Deblurring Experiment



$\mathcal{R}$	λ	$\|x - x^*\|/\|x^*\|$ (SNR)	k	γ	θ
I	5.6e-2	1.87e-1 (15)	11	5.1e-1	9.e-1
$\nabla^2(x)$	1.35e-1	1.74e-1 (15)	9	1.02e-3	1.e-2

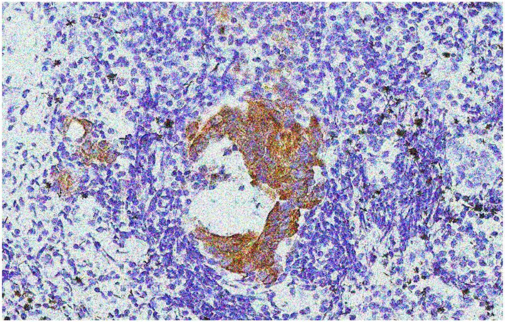
Deblurring Experiment



$\mathcal{R}$	λ	$\|x - x^*\|/\|x^*\|$ (SNR)	k	γ	θ
I	5.6e-2	1.87e-1 (15)	11	5.1e-1	9.e-1
$\nabla^2(\mathbf{x})$	1.35e-1	1.74e-1 (15)	9	1.02e-3	1.e-2

Denoising Experiment

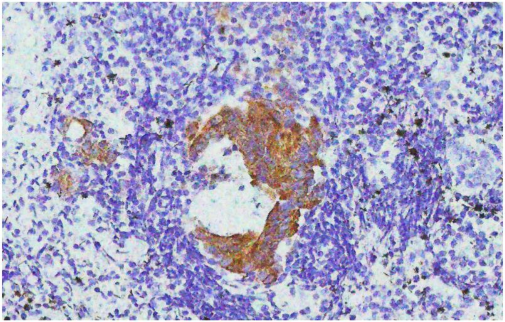
Noisy Image: $\mathbf{y}^\delta = \mathbf{x}^* + \delta,$



$$\frac{\|\mathbf{y}^\delta - \mathbf{x}^*\|}{\|\mathbf{x}^*\|} = 0.25, \quad SNR = 12$$

Denoising Experiment

$$\mathbf{TV}(\mathbf{x}) = \sqrt{\sum_{i=1}^{3} TV_\beta(\mathbf{x}_i)^2}, \quad \mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]$$ [3]

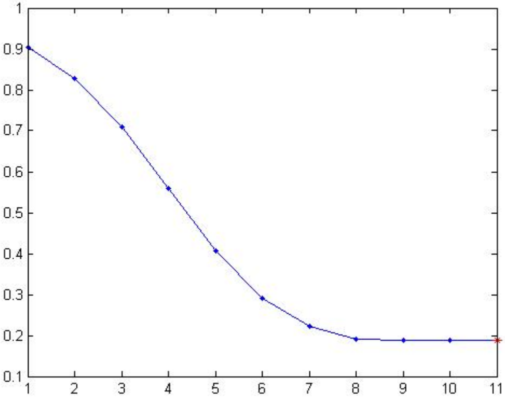


$\mathcal{R}$	λ	$\|\mathbf{x}-\mathbf{x}^*\|/\|\mathbf{x}^*\|$ (SNR)	k(it)	γ	θ
TV	**12.5**	**1.459e-1 (17)**	**5(80)**	**6.9**	**0.5**

[3] P. Blomgren, T. F. Chan, Color TV: Total Variation Mathods for Restoration of Vector Valued Images, IEEE Tr. on Im. Proc., vol.7, 1998.

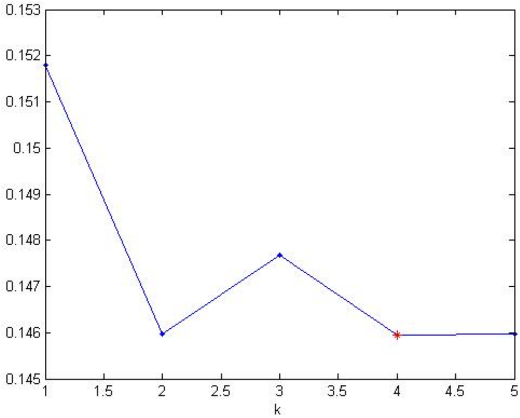
Convergence

Deblurring experiment: Relative Error



Convergence

Denoising experiment: Relative Error



Constrained regularized formulation

$f : \mathbb{R}^N \rightarrow \mathbb{R}$ continuous differentiable function

$$\text{minimize } \mathcal{R}(x) \quad \text{subject to} \quad x \in \Sigma = \{x \in \mathbb{R}^n \mid \|Hx - y\|^2 \leq \sigma^2\}$$

Advantages over the unconstrained formulation:

$$\text{minimize}\, \mathcal{R}(x) + \gamma \|Hx - y\|^2$$

- It doesn't require the choice of the regularization parameter
- It's easily extensible to different functions $f(x)$
- It prevents from too noisy solutions

Constrained regularized formulation

$f : \mathbb{R}^N \rightarrow \mathbb{R}$ continuous differentiable function

$$\text{minimize } \mathcal{R}(x) \quad \text{subject to} \quad x \in \Sigma = \{x \in \mathbb{R}^n \mid \|Hx - y\|^2 \leq \sigma^2\}$$

Advantages over the unconstrained formulation:

$$\text{minimize}\, \mathcal{R}(x) + \gamma \|Hx - y\|^2$$

- It doesn't require the choice of the regularization parameter
- It's easily extensible to different functions $f(x)$
- It prevents from too noisy solutions

Constrained regularized formulation

$f : \mathbb{R}^N \to \mathbb{R}$ continuous differentiable function

$$\text{minimize } \mathcal{R}(x) \quad \text{subject to} \quad x \in \Sigma = \{x \in \mathbb{R}^n \mid \|Hx - y\|^2 \leq \sigma^2\}$$

Advantages over the unconstrained formulation:

$$\text{minimize}\, \mathcal{R}(x) + \gamma \|Hx - y\|^2$$

- It doesn't require the choice of the regularization parameter
- It's easily extensible to different functions $f(x)$
- It prevents from too noisy solutions

Feasible directions Trust Region method (FDTR)

The method starts with $x_0 \in \Sigma$ and generates a sequence $\{x_k\}$ of strictly feasible iterates of the form

$$x_{k+1} = x_k + \lambda_k d_k$$

where:

- d_k is a feasible descent direction
- $\lambda_k \in (0, 1]$ is the steplength computed by a line search along d_k by the Armijo rule.

Feasible directions Trust Region method (FDTR)

The method starts with $x_0 \in \Sigma$ and generates a sequence $\{x_k\}$ of strictly feasible iterates of the form

$$x_{k+1} = x_k + \lambda_k d_k$$

where:

- d_k is a feasible descent direction
- $\lambda_k \in (0, 1]$ is the steplength computed by a line search along d_k by the Armijo rule.

d_k computation

The direction d_k is obtained as the solution of the Trust Region subproblem:

$$\min \varphi(d) = \nabla\mathcal{R}(x_k)^T d + \frac{1}{2} d^T M_k d \qquad \text{subject to} \qquad \|d\|_C \leq \rho_k$$

where:

- M_k is a symmetric and positive definite approximation of the Hessian $\nabla^2\mathcal{R}(x_k)$
- $\|d\|_C := (d^T C d)^{1/2}, \quad C = H^T H$
- the radius ρ_k is defined so that $x_k + d_k$ is strictly feasible:

$$\rho_k = (1-\epsilon)(\sigma - \|Hx_k - y\|), \quad \epsilon \in (0,1)$$

d_k computation

The direction d_k is obtained as the solution of the Trust Region subproblem:

$$\min \varphi(d) = \nabla \mathcal{R}(x_k)^T d + \frac{1}{2} d^T M_k d \qquad \text{subject to} \qquad \|d\|_C \leq \rho_k$$

where:

- $M_k$is a symmetric and positive definite approximation of the Hessian $\nabla^2 \mathcal{R}(x_k)$
- $\|d\|_C := (d^T C d)^{1/2}, \quad C = H^T H$
- the radius ρ_k is defined so that $x_k + d_k$ is strictly feasible:

$$\rho_k = (1 - \epsilon)(\sigma - \|Hx_k - y\|), \quad \epsilon \in (0, 1)$$

d_k computation

The direction d_k is obtained as the solution of the Trust Region subproblem:

$$\min \varphi(d) = \nabla\mathcal{R}(x_k)^T d + \frac{1}{2} d^T M_k d \qquad \text{subject to} \qquad \|d\|_C \leq \rho_k$$

where:

- $M_k$is a symmetric and positive definite approximation of the Hessian $\nabla^2\mathcal{R}(x_k)$
- $\|d\|_C := (d^T C d)^{1/2}, \quad C = H^T H$
- the radius ρ_k is defined so that $x_k + d_k$ is strictly feasible:

$$\rho_k = (1-\epsilon)(\sigma - \|Hx_k - y\|), \quad \epsilon \in (0,1)$$

d_k computation

- The minimization problem is solved by the a Newton-CG method, where the linear system:

$$M_k d = -\nabla \mathcal{R}(x_k)$$

  is inexactly solved by the Steihaug Truncated CG method.
- The CG iterations are stopped when the condition on the residuals: $\|r_k\|_C < \epsilon \|r_0\|_C$ is satisfied. The direction d is one of the followings:
  - the k-th CG iterate d_k if d_k is feasible
  - a scaled version of d_k lying on the border of the feasible region otherwise

FDTR algorithm

Choose $x_0 \in int(\Sigma)$ *and compute* M_0, *spd approximation of the Hessian matrix.*
Given $\varepsilon \in (0, 1)$ *and* $\eta \in (0, 1)$;
for $k = 0, 1, 2, \ldots$

1. Computation of the trust region radius ρ_k

Set $\rho_k = (1 - \varepsilon)(\sigma - \|Hx_k - z\|)$;

2. Computation of the search direction d_k

Compute d_k *by approximately solving* $M_k d = -\nabla\mathcal{R}(x_k)$ *via the Steihaug algorithm;*

if $d_k = 0$

return x_k;

3. Computation of the step-length λ_k

Find the smallest integer $i = 0, 1, 2, \ldots$ *satisfying*

$$\mathcal{R}(x_k + 2^{-i} d_k) \le \mathcal{R}(x_k) + \eta 2^{-i} d_k^T \nabla\mathcal{R}(x_k)$$

and set $\lambda_k = 2^{-i}$;

4. Updates

Compute a new symmetric positive definite approximation M_{k+1} *of the Hessian matrix;*

Set $x_{k+1} = x_k + \lambda_k d_k$ *and* $k = k + 1$;

Convergence properties

We proved that:

- The direction d_k is a descent direction and a feasible direction for the constraints
- The line search completes successfully
- In the case of finite iterations, the last computed iterate is a stationary point
- The FDTR method globally converges to stationary points

Convergence properties

We proved that:

- The direction d_k is a descent direction and a feasible direction for the constraints
- The line search completes successfully
- In the case of finite iterations, the last computed iterate is a stationary point
- The FDTR method globally converges to stationary points

Convergence properties

We proved that:

- The direction d_k is a descent direction and a feasible direction for the constraints
- The line search completes successfully
- In the case of finite iterations, the last computed iterate is a stationary point
- The FDTR method globally converges to stationary points

Convergence properties

We proved that:

- The direction d_k is a descent direction and a feasible direction for the constraints
- The line search completes successfully
- In the case of finite iterations, the last computed iterate is a stationary point
- The FDTR method globally converges to stationary points

Implementation details:Hessian approximation

The matrix M_k should be a positive definite approximation of the exact Hessian matrix $\nabla(f(x_k)$.

- Tikhonov regularization term

$$\mathcal{R}(x) = \|Dx\|^2$$

$$M(x_k) = \nabla^2 \mathcal{R}(x) = D^t D$$

Implementation details

The Hessian approximation $M(x_k)$

- Total Variation (TV) regularization term:

$$\mathcal{R}(x) = \frac{1}{N^2} \sum_{1 \le i,j \le N} \sqrt{|\nabla x_{i,j}|^2 + \beta}$$

$$\nabla^2 \mathcal{R}(x) = L(x) + L'(x)x$$

where, for $v \in \mathbb{R}^{N \times N}$, the operators $L(x)$ and $L'(x)$ are defined by

$$< L(x)v, v > := \frac{1}{N^2} \sum_{i,j=1}^{N} \frac{< \nabla v_{i,j}, \nabla v_{i,j} >}{\sqrt{|\nabla x_{i,j}|^2 + \beta}},$$

$$< (L'(x)x)v, v > := -\frac{1}{N^2} \sum_{i,j=1}^{m} \frac{< \nabla x_{i,j}, \nabla v_{i,j} >^2}{\sqrt{(|\nabla x_{i,j}|^2 + \beta)^3}}$$

Then:

$$M(x_k) = L(x_k), \quad \forall k = 0, 1, 2, \ldots .$$

Implementation details

Stopping criteria

- $$\Big| \|x_k - y\| - \sigma \Big| < \tau_1 \Big| \|x_0 - y\| - \sigma \Big|$$

  $\tau_1 > 0$

- $$\|\lambda_k d_k\| \leq \tau_2$$

  $\tau_2 > 0$

- $k > \mathrm{maxit}_{\mathrm{FDTR}}$

Implementation details

Stopping criteria

- $$\Big|\|x_k - y\| - \sigma\Big| < \tau_1\Big|\|x_0 - y\| - \sigma\Big|$$
  $\tau_1 > 0$
- $$\|\lambda_k d_k\| \leq \tau_2$$
  $\tau_2 > 0$
- $k > \mathrm{maxit}_{\mathrm{FDTR}}$

Implementation details

Stopping criteria

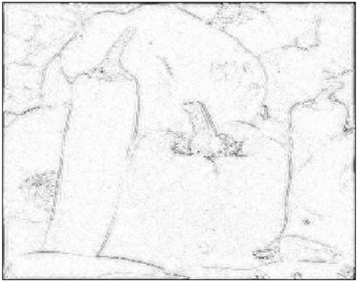
- $$\Big| \|x_k - y\| - \sigma \Big| < \tau_1 \Big| \|x_0 - y\| - \sigma \Big|$$
  $\tau_1 > 0$
- $$\|\lambda_k d_k\| \leq \tau_2$$
  $\tau_2 > 0$
- $k > \mathrm{maxit}_{\mathrm{FDTR}}$

Deblurring results

Blurry (psfgauss.m, $\varsigma = 2$) and noisy ($noiselevel = 10^{-3}$) observed image

True image

Deblurring results

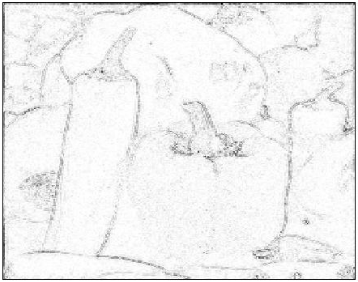


FDTR-Tikhonov reconstruction (rel. err.=0.087)

Difference image

Deblurring results

FDTR-TV reconstruction (rel. err.=0.082)



Difference image

Deblurring results

Reg.	Noise Level	Variance	Rel. Error	Inner It.	Outer It.
TV	1.0e-3	$\varsigma = 2$	8.2575e-002	13	8
TV	5.0e-3	$\varsigma = 2$	1.0076e-1	36	8
TV	1.0e-3	$\varsigma = 3$	1.0805e-1	17	11
Tikh	1.0e-3	$\varsigma = 2$	8.7267e-2	10	8
Tikh	5.0e-3	$\varsigma = 2$	1.1331e-1	24	7
Tikh	1.0e-3	$\varsigma = 3$	1.0922e-1	12	11

Tabella: Numerical results for the image deblurring test problems.

FDTR method with automatic noise estimate

Algorithm (Automatic Image Denoising Algorithm)

Input: z, $\varepsilon \in (0,1)$, $\eta \in (0,1)$, $\theta \in (0,1)$, $\epsilon > 0$ *and* $\tau_1 > 0$
Output: x, σ

Set $x_0 = z$ *and* $\sigma_0 = \|z\|$;
Set $update_\sigma = true$; *Set* $k = 0$

Repeat until convergence

Step 1: *image denoising.*
Compute the new iterate x_{k+1} *with FDTR algorithm .*

Step 2: *noise estimate.*
If $update_\sigma$

2.1 *Compute the new estimate* σ_{k+1}:
2.1.1 *Compute* $\delta_k = \|x_{k+1} - y\|$;
2.1.2 *Compute* $\sigma_{k+1} = \theta\delta_k + (1-\theta)\sigma_k$.
2.2 *If* $\sigma_k - \sigma_{k+1} < \tau_1$, *set* $update_\sigma = false$.

Set $k = k + 1$.

Set $x = x_k$ *and* $\sigma = \sigma_k$.

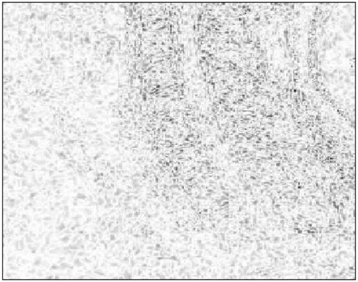
Denoising results

noisy ($noiselevel = 1.3 \cdot 10^{-1}$) observed image

True image

Denoising results

FDTR-TV reconstruction (rel. err.$=5.39 \cdot 10^{-2}$)



difference image

Denoising results

FDTR-TV automatic reconstruction (err.$=6.32 \cdot 10^{-2}$)

difference image

Denoising results

Reg.	Noise Level	Rel. Error	Inner It.	Outer It.	time (sec)
TV	1.3e-1	5.39e-2	11	3	1
TV(nogamma)	1.3e-1	6.32e-2	55	25	3.3
TV	2.5e-1	7.86e-2	13	3	1
TV(nogamma)	2.5e-1	8.13e-2	50	22	2.8

Tabella: Numerical results for image denoising.

Conclusions

- The discrete ill-posed problem $Hx = y$ has been reformulated as a constrained minimization regularized problem in two different forms.
- Iterative solution methods have been presented for the solution of the constrained minimization problems.
- The following regularization functions:
  - $\mathcal{R}(x) = \|Lx\|^2$
  - $\mathcal{R}(x) = \frac{1}{N^2} \sum_{1 \le i,j \le N} \sqrt{|\nabla x_{i,j}|^2 + \beta}$

  have been considered, but the methods are usable for any convex function $\mathcal{R}(x)$.
- The regularized solutions are computed on the basis of some parameters:
  - The noise on the recorded image
  - the smoothness of the solution

  that can be approximated by using only the information from the recorded image.
- Numerical results for image deblurring and denoising applications show that the methods have good precision and fast convergence, hence they are suitable for large size problems.