

ADMM in Imaging Inverse Problems: Non-Periodic and Blind Deconvolution

Mário A. T. Figueiredo

Instituto Superior Técnico,
University of Lisbon, Portugal

Instituto de Telecomunicações
Lisbon, Portugal



INSTITUTO SUPERIOR TÉCNICO



instituto de
telecomunicações

Joint work with:



Manya Afonso



José Bioucas-Dias



Mariana Almeida

Outline

1. Formulations and tools
2. The canonical ADMM and its extension for more than two functions
3. Linear-Gaussian observations: the SALSA algorithm.
4. Poisson observations: the PIDAL algorithm
5. Handling non periodic boundaries
6. Into the non-convex realm: blind deconvolution

Inference/Learning via Optimization

Many inference criteria (in signal processing, machine learning) have the form

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + \tau c(\mathbf{x})$$

$f : \mathbb{R}^n \rightarrow \mathbb{R}$ data fidelity, observation model, negative log-likelihood, loss, ...

... usually **smooth** and **convex**. Canonical example:

Canonical example: $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2$

$c : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ regularization/penalty function, negative log-prior, ...

... typically **convex**, often **non-differentiable** (e.g., for sparsity)

Examples: signal/image restoration/reconstruction, sparse representations, compressive sensing/imaging, linear regression, logistic regression, channel sensing, support vector machines, ...

Unconstrained Versus Constrained Optimization

Unconstrained optimization formulation

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} f(\mathbf{x}) + \tau c(\mathbf{x}) \quad (\text{Tikhonov regularization})$$

Constrained optimization formulations

$$\begin{aligned} \hat{\mathbf{x}} &\in \arg \min_{\mathbf{x}} c(\mathbf{x}) \\ &\text{s. t. } f(\mathbf{x}) \leq \varepsilon \end{aligned} \quad (\text{Morozov regularization})$$

$$\begin{aligned} \hat{\mathbf{x}} &\in \arg \min_{\mathbf{x}} f(\mathbf{x}) \\ &\text{s. t. } c(\mathbf{x}) \leq \delta \end{aligned} \quad (\text{Ivanov regularization})$$

“Equivalent”, under mild conditions; maybe not equally convenient/easy
[Lorenz, 2012]

A Fundamental Dichotomy: Analysis vs Synthesis

[Elad, Milanfar, Rubinstein, 2007], [Selesnick, F, 2010],

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + \tau c(\mathbf{x})$$

Synthesis regularization:

\mathbf{x} contains **representation** coefficients (not the signal/image itself)

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{A}\mathbf{x}) + \tau c(\mathbf{x})$$

$\mathbf{A} = \mathbf{B}\mathbf{W}$, where \mathbf{B} is the observation operator

\mathbf{W} is a synthesis operator; e.g., a *Parseval frame* $\mathbf{W}\mathbf{W}^* = \mathbf{I}$

\mathcal{L} depends on the noise model; e.g., $\mathcal{L}(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2$

typical (sparsity-inducing) regularizer: $c(\mathbf{x}) = \|\mathbf{x}\|_1$

proper, lower semi-continuous (lsc), convex (not strictly), coercive.

A Fundamental Dichotomy: Analysis vs Synthesis (II)

[Elad, Milanfar, Rubinstein, 2007], [Selesnick, F, 2010],

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{A}\mathbf{x}) + \tau c(\mathbf{x})$$

Analysis regularization

\mathbf{x} is the signal/image itself, \mathbf{A} is the observation operator

typical frame-based analysis regularizer:

$$c(\mathbf{x}) = \|\mathbf{P}\mathbf{x}\|_1$$

 analysis operator (e.g., of a Parseval frame, $\mathbf{P}^*\mathbf{P} = \mathbf{I}$)

proper, lsc, convex (not strictly), and coercive.

Total variation (TV) is also “analysis”; proper, lsc, convex (not strictly),
... but not coercive.

Typical Convex Data Terms

Let: $f(\mathbf{x}) = \mathcal{L}(\mathbf{Ax})$ where $\mathcal{L}(\mathbf{z}) \equiv \sum_{i=1}^m \xi(z_i, y_i)$

where ξ is one (e.g.) of these functions (log-likelihoods):

Gaussian observations: $\xi_G(z, y) = \frac{1}{2}(z - y)^2 \longrightarrow \mathcal{L}_G$

Poissonian observations: $\xi_P(z, y) = z + \iota_{\mathbb{R}_+}(z) - y \log(z_+) \longrightarrow \mathcal{L}_P$

Multiplicative noise: $\xi_M(z, y) = L(z + e^{y-z}) \longrightarrow \mathcal{L}_M$

...all proper, lower semi-continuous (lsc), coercive, convex.

\mathcal{L}_G and \mathcal{L}_M are strictly convex. \mathcal{L}_P is strictly convex if $y_i > 0, \forall_i$

A Key Tool: The Moreau Proximity Operator

The *Moreau proximity operator* [Moreau 62], [Combettes, Pesquet, Wajs, 01, 03, 05, 07, 10, 11].

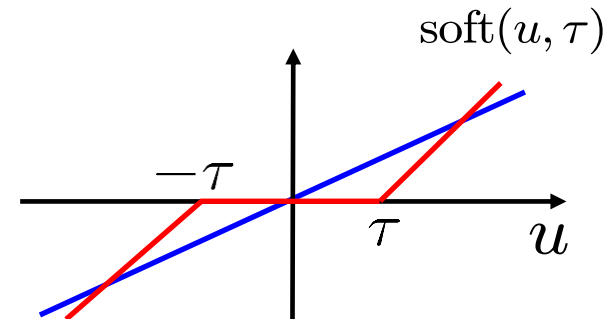
$$\text{prox}_{\tau c}(\mathbf{u}) = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{u}\|_2^2 + \tau c(\mathbf{x})$$

Classical cases:

$$c(\mathbf{z}) = \iota_{\mathcal{C}}(\mathbf{z}) = \begin{cases} 0 & \Leftrightarrow \mathbf{z} \in \mathcal{C} \\ +\infty & \Leftrightarrow \mathbf{z} \notin \mathcal{C} \end{cases} \Rightarrow \text{prox}_{\tau c}(\mathbf{u}) = \Pi_{\mathcal{C}}(\mathbf{u})$$

Euclidean projection on convex set \mathcal{C}

$$c(\mathbf{z}) = \frac{1}{2} \|\mathbf{z}\|_2^2 \Rightarrow \text{prox}_{\tau c}(\mathbf{u}) = \frac{\mathbf{u}}{1 + \tau}$$



$$c(\mathbf{z}) = \|\mathbf{z}\|_1 \Rightarrow \text{prox}_{\tau c}(\mathbf{u}) = \text{soft}(\mathbf{u}, \tau) = \text{sign}(\mathbf{u}) \odot \max(|\mathbf{u}| - \tau, 0)$$

Separability:

$$c(\mathbf{z}) = \sum_i c_i(z_i) \Rightarrow (\text{prox}_{\tau c}(\mathbf{u}))_i = \text{prox}_{\tau c_i}(u_i)$$

Moreau Proximity Operators

...many more!

[Combettes, Pesquet, 2010]

$\phi(x)$	$\text{prox}_{\phi}x$
i $\iota_{[\underline{\omega}, \bar{\omega}]}(x)$	$P_{[\underline{\omega}, \bar{\omega}]}x$
ii $\sigma_{[\underline{\omega}, \bar{\omega}]}(x) = \begin{cases} \underline{\omega}x & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ \bar{\omega}x & \text{otherwise} \end{cases}$	$\text{soft}_{[\underline{\omega}, \bar{\omega}]}(x) = \begin{cases} x - \underline{\omega} & \text{if } x < \underline{\omega} \\ 0 & \text{if } x \in [\underline{\omega}, \bar{\omega}] \\ x - \bar{\omega} & \text{if } x > \bar{\omega} \end{cases}$
iii $\begin{matrix} \psi(x) + \sigma_{[\underline{\omega}, \bar{\omega}]}(x) \\ \psi \in \Gamma_0(\mathbb{R}) \text{ differentiable at } 0 \\ \psi'(0) = 0 \end{matrix}$	$\text{prox}_{\psi}(\text{soft}_{[\underline{\omega}, \bar{\omega}]}(x))$
iv $\max\{ x - \omega, 0\}$	$\begin{cases} x & \text{if } x < \omega \\ \text{sign}(x)\omega & \text{if } \omega \leq x \leq 2\omega \\ \text{sign}(x)(x - \omega) & \text{if } x > 2\omega \end{cases}$
v $\kappa x ^q$	$\text{sign}(x)p$, where $p \geq 0$ and $p + q\kappa p^{q-1} = x $
vi $\begin{cases} \kappa x^2 & \text{if } x \leq \omega/\sqrt{2\kappa} \\ \omega\sqrt{2\kappa} x - \omega^2/2 & \text{otherwise} \end{cases}$	$\begin{cases} x/(2\kappa + 1) & \text{if } x \leq \omega(2\kappa + 1)/\sqrt{2\kappa} \\ x - \omega\sqrt{2\kappa}\text{sign}(x) & \text{otherwise} \end{cases}$
vii $\omega x + \tau x ^2 + \kappa x ^q$	$\text{sign}(x)\text{prox}_{\kappa \cdot ^q/(2\tau+1)}\frac{\max\{ x - \omega, 0\}}{2\tau + 1}$
viii $\omega x - \ln(1 + \omega x)$	$(2\omega)^{-1}\text{sign}(x)\left(\omega x - \omega^2 - 1 + \sqrt{ \omega x - \omega^2 - 1 ^2 + 4\omega x }\right)$
ix $\begin{cases} \omega x & \text{if } x \geq 0 \\ +\infty & \text{otherwise} \end{cases}$	$\begin{cases} x - \omega & \text{if } x \geq \omega \\ 0 & \text{otherwise} \end{cases}$
x $\begin{cases} -\omega x^{1/q} & \text{if } x \geq 0 \\ +\infty & \text{otherwise} \end{cases}$	$p^{1/q}$, where $p > 0$ and $p^{2q-1} - xp^{q-1} = q^{-1}\omega$
xi $\begin{cases} \omega x^{-q} & \text{if } x > 0 \\ +\infty & \text{otherwise} \end{cases}$	$p > 0$ such that $p^{q+2} - xp^{q+1} = \omega q$
xii $\begin{cases} x \ln(x) & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ +\infty & \text{otherwise} \end{cases}$	$W(e^{x-1})$, where W is the Lambert W-function
xiii $\begin{cases} -\ln(x - \underline{\omega}) + \ln(-\underline{\omega}) & \text{if } x \in]\underline{\omega}, 0[\\ -\ln(\bar{\omega} - x) + \ln(\bar{\omega}) & \text{if } x \in]0, \bar{\omega}[\\ +\infty & \text{otherwise} \end{cases}$ $\underline{\omega} < 0 < \bar{\omega}$	$\begin{cases} \frac{1}{2}(x + \underline{\omega} + \sqrt{ x - \underline{\omega} ^2 + 4}) & \text{if } x < 1/\underline{\omega} \\ \frac{1}{2}(x + \bar{\omega} - \sqrt{ x - \bar{\omega} ^2 + 4}) & \text{if } x > 1/\bar{\omega} \\ 0 & \text{otherwise} \end{cases}$ (see Figure 1)
xiv $\begin{cases} -\kappa \ln(x) + \tau x^2/2 + \alpha x & \text{if } x > 0 \\ +\infty & \text{otherwise} \end{cases}$	$\frac{1}{2(1 + \tau)}(x - \alpha + \sqrt{ x - \alpha ^2 + 4\kappa(1 + \tau)})$
xv $\begin{cases} -\kappa \ln(x) + \alpha x + \omega x^{-1} & \text{if } x > 0 \\ +\infty & \text{otherwise} \end{cases}$	$p > 0$ such that $p^3 + (\alpha - x)p^2 - \kappa p = \omega$
xvi $\begin{cases} -\kappa \ln(x) + \omega x^q & \text{if } x > 0 \\ +\infty & \text{otherwise} \end{cases}$	$p > 0$ such that $q\omega p^q + p^2 - xp = \kappa$
xvii $\begin{cases} -\underline{\kappa} \ln(x - \underline{\omega}) - \bar{\kappa} \ln(\bar{\omega} - x) & \text{if } x \in]\underline{\omega}, \bar{\omega}[\\ +\infty & \text{otherwise} \end{cases}$	$p \in]\underline{\omega}, \bar{\omega}[$ such that $p^3 - (\underline{\omega} + \bar{\omega} + x)p^2 + (\underline{\omega}\bar{\omega} - \underline{\kappa} - \bar{\kappa} + (\underline{\omega} + \bar{\omega})x)p = \underline{\omega}\bar{\omega}x - \underline{\omega}\bar{\kappa} - \bar{\omega}\kappa$

Iterative Shrinkage/Thresholding (IST)

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + \tau c(\mathbf{x})$$

$$\mathbf{x}_{k+1} = \text{prox}_{\tau c/\alpha} \left(\mathbf{x}_k - \frac{1}{\alpha} \nabla f(\mathbf{x}_k) \right)$$

Iterative shrinkage thresholding (IST)

a.k.a. forward-backward splitting

a.k.a. proximal gradient algorithm

[Bruck, 1977], [Passty, 1979], [Lions, Mercier, 1979],
[F, Nowak, 01, 03], [Daubechies, Defrise, De Mol, 02, 04],
[Combettes and Wajs, 03, 05], [Starck, Candés, Nguyen,
Murtagh, 03], [Combettes, Pesquet, Wajs, 03, 05, 07, 11],

Key condition in convergence proofs: ∇f is Lipschitz

...not true, e.g., with Poisson or multiplicative noise.

IST is usually **slow** (specially if τ is small); faster alternatives:

- Two-step IST (TwIST) [Bioucas-Dias, F, 07]
- Fast IST (FISTA) [Beck, Teboulle, 09], [Tseng, 08]
- Continuation [Hale, Yin, Zhang, 07], [Wright, Nowak, F, 07, 09]
- SpaRSA [Wright, Nowak, F, 08, 09]
- Proximal Newton [Becker, Fadili, 2012], [Lee, Sun, Saunders, 2012],
[Tran-Dinh, Kyrillidis, C, 2013], [Chouzenoux, Pesquet, Repetti, 2013].

Variable Splitting + Augmented Lagrangian

Unconstrained (convex) optimization problem: $\min_{\mathbf{z} \in \mathbb{R}^d} f_1(\mathbf{z}) + f_2(\mathbf{G} \mathbf{z})$

Equivalent constrained problem: $\min_{\mathbf{z} \in \mathbb{R}^d, \mathbf{u} \in \mathbb{R}^c} f_1(\mathbf{z}) + f_2(\mathbf{u})$
s.t. $\mathbf{u} - \mathbf{G} \mathbf{z} = 0$

Augmented Lagrangian (AL):

$$L_\mu(\mathbf{z}, \mathbf{u}, \lambda) = f_1(\mathbf{z}) + f_2(\mathbf{u}) + \lambda^T (\mathbf{G} \mathbf{z} - \mathbf{u}) + \frac{\mu}{2} \|\mathbf{G} \mathbf{z} - \mathbf{u}\|_2^2$$

AL, or method of multipliers [Hestenes, Powell, 1969]

$$(\mathbf{z}_{k+1}, \mathbf{u}_{k+1}) = \arg \min_{\mathbf{z}, \mathbf{u}} L_\mu(\mathbf{z}, \mathbf{u}, \lambda_k)$$

$$\lambda_{k+1} = \lambda_k + \mu(\mathbf{G} \mathbf{z}_{k+1} - \mathbf{u}_{k+1})$$

equivalent

$$(\mathbf{z}_{k+1}, \mathbf{u}_{k+1}) = \arg \min_{\mathbf{z}, \mathbf{u}} f_1(\mathbf{z}) + f_2(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{G} \mathbf{z} - \mathbf{u} - \mathbf{d}_k\|_2^2$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - (\mathbf{G} \mathbf{z}_{k+1} - \mathbf{u}_{k+1})$$

A Workhorse: the Alternating Direction Method of Multipliers (ADMM)

$$\text{Problem: } \min_{\mathbf{z} \in \mathbb{R}^d} f_1(\mathbf{z}) + f_2(\mathbf{G} \mathbf{z})$$

Method of multipliers (MM)

$$\begin{aligned} (\mathbf{z}_{k+1}, \mathbf{u}_{k+1}) &= \arg \min_{\mathbf{z}, \mathbf{u}} f_1(\mathbf{z}) + f_2(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{G} \mathbf{z} - \mathbf{u} - \mathbf{d}_k\|_2^2 \\ \mathbf{d}_{k+1} &= \mathbf{d}_k - (\mathbf{G} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}) \end{aligned}$$

ADMM [Glowinski, Marrocco, 75], [Gabay, Mercier, 76], [Gabay, 83], [Eckstein, Bertsekas, 92]

$$\begin{aligned} \mathbf{z}_{k+1} &= \arg \min_{\mathbf{z}} f_1(\mathbf{z}) + \frac{\mu}{2} \|\mathbf{G} \mathbf{z} - \mathbf{u}_k - \mathbf{d}_k\|_2^2 \\ \mathbf{u}_{k+1} &= \arg \min_{\mathbf{u}} f_2(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{G} \mathbf{z}_{k+1} - \mathbf{u} - \mathbf{d}_k\|_2^2 \\ \mathbf{d}_{k+1} &= \mathbf{d}_k - (\mathbf{G} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}) \end{aligned}$$

Interpretations: variable splitting + augmented Lagrangian + NLBGS;

Douglas-Rachford splitting on the dual [Eckstein, Bertsekas, 92]

split-Bregman approach [Goldstein, Osher, 08]

A Cornerstone Result on ADMM

[Eckstein, Bertsekas, 1992]

The problem $\min_{\mathbf{z} \in \mathbb{R}^d} f_1(\mathbf{z}) + f_2(\mathbf{G} \mathbf{z})$

f_1, f_2 closed, proper, convex; \mathbf{G} full column rank.

$\bar{\mathbf{z}}$ a solution of the problem

$(\mathbf{z}_k, k = 0, 1, 2, \dots)$ the ADMM sequence ($\mu > 0$)

$$\lim_{k \rightarrow \infty} \mathbf{z}_k = \bar{\mathbf{z}}$$

Inexact minimizations allowed, as long as the errors are absolutely summable.



Explosion of applications in signal processing, machine learning, statistics, ...

[Giovannelli, Coulais, 05], [Giannakis et al, 08, 09,...], [Tomioka et al, 09], [Boyd et al, 11], [Goldfarb, Ma, 10,...], [Fessler et al, 11, ...], [Mota et al, 10], [Jakovetić et al, 12], [Banerjee et al, 12], [Esser, 09], [Ng et al, 20], [Setzer, Steidl, Teuber, 09], [Yang, Zhang, 11], [Combettes, Pesquet, 10,...], [Chan, Yang, Yuan, 11],



More on the Convergence of ADMM

Convergence of ADMM is an active research topic

Dual objective $O(1/k)$ [Goldfarb, Ma 2009], [Goldstein et al, 2012], ...

...with Nesterov acceleration $O(1/k^2)$ [Goldstein et al, 2012], ...

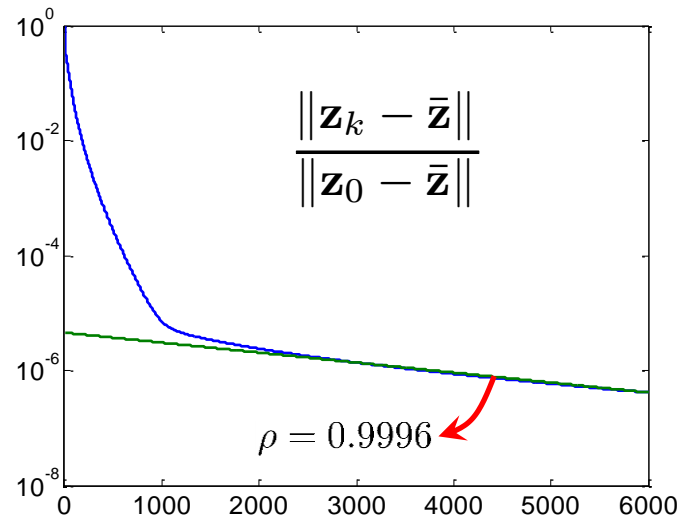
Linear convergence of iterates (under more conditions)

[Deng, Yin, 2012], [Luo, 2012], ...

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{z}_{k+1} - \bar{\mathbf{z}}\|}{\|\mathbf{z}_k - \bar{\mathbf{z}}\|} = \rho$$

Example: total-variation denoising $\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{x}\|_2^2 + \tau \text{TV}(\mathbf{x})$

100 iterations



6000 iterations



(The Art of) Applying ADMM

Synthesis formulation:

$$\min_{\mathbf{x}} \mathcal{L}(\mathbf{B}\mathbf{W}\mathbf{x}) + \tau c(\mathbf{x})$$

Template problem for ADMM

$$\min_{\mathbf{z}} f_1(\mathbf{z}) + f_2(\mathbf{G}\mathbf{z})$$

Naïve mapping: $\mathbf{G} = \mathbf{B}\mathbf{W}$, $f_1 = \tau c$, $f_2 = \mathcal{L}$

ADMM

$$\mathbf{z}_{k+1} = \arg \min_{\mathbf{z}} \tau c(\mathbf{z}) + \frac{\mu}{2} \|\mathbf{B}\mathbf{W}\mathbf{z} - \mathbf{u}_k - \mathbf{d}_k\|^2$$

usually hard!

$$\mathbf{u}_{k+1} = \arg \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{B}\mathbf{W}\mathbf{z}_{k+1} - \mathbf{u} - \mathbf{d}_k\|^2$$

usually easy
 $\text{prox}_{\mathcal{L}/\mu}$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - (\mathbf{B}\mathbf{W}\mathbf{z}_{k+1} - \mathbf{u}_{k+1})$$

Applying ADMM

Analysis formulation:

$$\min_{\mathbf{x}} \mathcal{L}(\mathbf{B}\mathbf{x}) + \tau c(\mathbf{P}\mathbf{x})$$

Template problem for ADMM

$$\min_{\mathbf{z}} f_1(\mathbf{z}) + f_2(\mathbf{G}\mathbf{z})$$

Naïve mapping: $\mathbf{G} = \mathbf{P}$, $f_1 = \mathcal{L} \circ \mathbf{B}$, $f_2 = \tau c$

$$\mathbf{z}_{k+1} = \arg \min_{\mathbf{z}} \mathcal{L}(\mathbf{B}\mathbf{z}) + \frac{\mu}{2} \|\mathbf{P}\mathbf{z} - \mathbf{u}_k - \mathbf{d}_k\|_2^2$$

$$\mathbf{u}_{k+1} = \arg \min_{\mathbf{u}} \tau c(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{P}\mathbf{z}_{k+1} - \mathbf{u} - \mathbf{d}_k\|_2^2$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - (\mathbf{P}\mathbf{z}_{k+1} - \mathbf{u}_{k+1})$$

usually easy
 $\text{prox}_{\tau c/\mu}$

Easy if: \mathcal{L} is quadratic and
 \mathbf{B} and \mathbf{P} diagonalized by common transform (e.g., DFT)
(split-Bregman [Goldstein, Osher, 08])

Applying ADMM

Analysis formulation:

$$\min_{\mathbf{x}} \mathcal{L}(\mathbf{B}\mathbf{x}) + \tau c(\mathbf{P}\mathbf{x})$$

Template problem for ADMM

$$\min_{\mathbf{z}} f_1(\mathbf{z}) + f_2(\mathbf{G}\mathbf{z})$$

Naïve mapping:

$$\mathbf{G} = \mathbf{B}, \quad f_1 = \tau c \circ \mathbf{P}, \quad f_2 = \mathcal{L}$$

$$\mathbf{z}_{k+1} = \arg \min_{\mathbf{z}} \tau c(\mathbf{P}\mathbf{z}) + \frac{\mu}{2} \|\mathbf{B}\mathbf{z} - \mathbf{u}_k - \mathbf{d}_k\|_2^2$$

$$\mathbf{u}_{k+1} = \arg \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{B}\mathbf{z}_{k+1} - \mathbf{u} - \mathbf{d}_k\|_2^2$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k - (\mathbf{B}\mathbf{z}_{k+1} - \mathbf{u}_{k+1})$$

usually easy
 $\text{prox}_{\mathcal{L}/\mu}$


Easy if: c is quadratic and
 \mathbf{B} and \mathbf{P} diagonalized by common transform (e.g., DFT)

General Template for ADMM with Two or More Functions

[F and Bioucas-Dias, 2009]

Consider a more general problem

$$\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z}) \quad (P)$$


$$g_j : \mathbb{R}^{p_j} \rightarrow \bar{\mathbb{R}}$$

Proper, closed, convex functions


$$\mathbf{H}^{(j)} \in \mathbb{R}^{p_j \times d}$$

Arbitrary matrices

There are many ways to write (P) as

$$\min_{\mathbf{z} \in \mathbb{R}^d} f_1(\mathbf{z}) + f_2(\mathbf{G} \mathbf{z})$$

We propose:

$$f_1(\mathbf{z}) = 0, \quad \mathbf{G} = \begin{bmatrix} \mathbf{H}^{(1)} \\ \vdots \\ \mathbf{H}^{(J)} \end{bmatrix}, \quad f_2 \left(\begin{bmatrix} \mathbf{u}^{(1)} \\ \vdots \\ \mathbf{u}^{(J)} \end{bmatrix} \right) = \sum_{j=1}^J g_j(\mathbf{u}^{(j)})$$

ADMM for Two or More Functions

$$\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z}), \quad \min_{\mathbf{z} \in \mathbb{R}^d} f_2(\mathbf{G} \mathbf{z}), \quad \mathbf{G} = \begin{bmatrix} \mathbf{H}^{(1)} \\ \vdots \\ \mathbf{H}^{(J)} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \mathbf{u}^{(1)} \\ \vdots \\ \mathbf{u}^{(J)} \end{bmatrix}$$

$$\mathbf{z}_{k+1} = \left(\sum_{j=1}^J (\mathbf{H}^{(j)})^* \mathbf{H}^{(j)} \right)^{-1} \sum_{j=1}^J (\mathbf{H}^{(j)})^* \left(\mathbf{u}_k^{(j)} + \mathbf{d}_k^{(j)} \right)$$

$$\mathbf{u}_{k+1}^{(1)} = \arg \min_{\mathbf{u}} g_1(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{u} - \mathbf{H}^{(1)} \mathbf{z}_{k+1} + \mathbf{d}_k^{(1)}\|^2 = \text{prox}_{g_1/\mu}(\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{d}_k^{(1)})$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$\mathbf{u}_{k+1}^{(J)} = \arg \min_{\mathbf{u}} g_J(\mathbf{u}) + \frac{\mu}{2} \|\mathbf{u} - \mathbf{H}^{(J)} \mathbf{z}_{k+1} + \mathbf{d}_k^{(J)}\|^2 = \text{prox}_{g_J/\mu}(\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{d}_k^{(J)})$$

$$\mathbf{d}_{k+1}^{(1)} = \mathbf{d}_k^{(1)} - (\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(1)})$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$\mathbf{d}_{k+1}^{(J)} = \mathbf{d}_k^{(J)} - (\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(J)})$$

ADMM for Two or More Functions

$$\mathbf{z}_{k+1} = \left(\sum_{j=1}^J (\mathbf{H}^{(j)})^* \mathbf{H}^{(j)} \right)^{-1} \sum_{j=1}^J (\mathbf{H}^{(j)})^* \left(\mathbf{u}_k^{(j)} + \mathbf{d}_k^{(j)} \right)$$

$$\mathbf{u}_{k+1}^{(1)} = \text{prox}_{g_1/\mu}(\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{d}_k^{(1)})$$

⋮

$$\mathbf{u}_{k+1}^{(J)} = \text{prox}_{g_1/\mu}(\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{d}_k^{(j)})$$

$$\mathbf{d}_{k+1}^{(1)} = \mathbf{d}_k^{(1)} - (\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(1)})$$

⋮ ⋮ ⋮

$$\mathbf{d}_{k+1}^{(J)} = \mathbf{d}_k^{(J)} - (\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(J)})$$

Conditions for easy applicability:

inexpensive proximity operators

inexpensive matrix inversion

...a cursing and a blessing!

ADMM for Two or More Functions

Applies to sum of convex terms

$$\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$$

Computation of proximity operators is parallelizable

Handling of matrices is isolated in a pure quadratic problem

Conditions for easy applicability: **inexpensive proximity operators**
inexpensive matrix inversion

Matrix inversion may be a ***curse or a blessing!*** (more later)

Similar algorithm: *simultaneous directions method of multipliers* (SDMM)

[Setzer, Steidl, Teuber, 2010], [Combettes, Pesquet, 2010]

Other ADMM versions for more than two functions

[Hong, Luo, 2012, 2013], [Ma, 2012]

Linear/Gaussian Observations: Frame-Based Analysis

Problem: $\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \tau \|\mathbf{P}\mathbf{x}\|_1$

Template: $\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$

Mapping: $J = 2, \quad g_1(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2, \quad g_2(\mathbf{z}) = \tau \|\mathbf{z}\|_1$

$$\mathbf{H}^{(1)} = \mathbf{A}, \quad \mathbf{H}^{(2)} = \mathbf{P},$$

Convergence conditions: g_1 and g_2 are closed, proper, and convex.

$$\mathbf{G} = \begin{bmatrix} \mathbf{A} \\ \mathbf{P} \end{bmatrix} \quad \text{has full column rank.}$$

Resulting algorithm: SALSA

(*split augmented Lagrangian shrinkage algorithm*) [Afonso, Bioucas-Dias, F, 2009, 2010]

ADMM for the Linear/Gaussian Problem: SALSA

Key steps of SALSA (both for analysis and synthesis):

Moreau proximity operator of $g_1(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2$,

$$\text{prox}_{g_1/\mu}(\mathbf{u}) = \arg \min_{\mathbf{z}} \frac{1}{2\mu} \|\mathbf{z} - \mathbf{y}\|_2^2 + \frac{1}{2} \|\mathbf{z} - \mathbf{u}\|_2^2 = \frac{\mathbf{y} + \mu \mathbf{u}}{1 + \mu}$$

Moreau proximity operator of $g_2(\mathbf{z}) = \tau \|\mathbf{z}\|_1$,

$$\text{prox}_{g_2/\mu}(\mathbf{u}) = \text{soft}\left(\mathbf{u}, \tau/\mu\right)$$

Matrix inversion:

$$\mathbf{z}_{k+1} = \left[\mathbf{A}^* \mathbf{A} + \mathbf{P}^* \mathbf{P} \right]^{-1} \left(\mathbf{A}^* \left(\mathbf{u}_k^{(1)} + \mathbf{d}_k^{(1)} \right) + \mathbf{P}^* \left(\mathbf{u}_k^{(2)} + \mathbf{d}_k^{(2)} \right) \right)$$

...next slide!

Handling the Matrix Inversion: Frame-Based Analysis

Frame-based analysis: $[\mathbf{A}^* \mathbf{A} + \mathbf{P}^* \mathbf{P}]^{-1} = [\mathbf{A}^* \mathbf{A} + \mathbf{I}]^{-1}$

$\mathbf{P}^* \mathbf{P} = \mathbf{I}$
Parseval frame

diagonal  DFT (FFT) 

Periodic deconvolution: $\mathbf{A} = \mathbf{U}^* \mathbf{D} \mathbf{U}$

$O(n \log n)$

$$[\mathbf{A}^* \mathbf{A} + \mathbf{I}]^{-1} = \mathbf{U}^* [|\mathbf{D}|^2 + \mathbf{I}]^{-1} \mathbf{U}$$

 subsampling matrix: $\mathbf{M} \mathbf{M}^* = \mathbf{I}$

Compressive imaging (MRI): $\mathbf{A} = \mathbf{M} \mathbf{U}$

$O(n \log n)$

$$[\mathbf{U}^* \mathbf{M}^* \mathbf{M} \mathbf{U} + \mathbf{I}]^{-1} = \mathbf{I} - \frac{1}{2} \mathbf{U}^* \mathbf{M}^* \mathbf{M} \mathbf{U}$$

matrix inversion lemma

 subsampling matrix: $\mathbf{S}^* \mathbf{S}$ is diagonal

Inpainting (recovery of lost pixels): $\mathbf{A} = \mathbf{S}$

$O(n)$

$$[\mathbf{S}^* \mathbf{S} + \mathbf{I}]^{-1} \text{ is a diagonal inversion}$$

SALSA for Frame-Based Synthesis

Problem: $\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \tau \|\mathbf{x}\|_1$

Template: $\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$

$\mathbf{A} = \mathbf{B}\mathbf{W}$

observation matrix

synthesis matrix

Mapping: $J = 2$, $g_1(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2$, $g_2(\mathbf{z}) = \tau \|\mathbf{z}\|_1$

$$\mathbf{H}^{(1)} = \mathbf{A} = \mathbf{B}\mathbf{W} \quad \mathbf{H}^{(2)} = \mathbf{I},$$

Convergence conditions: g_1 and g_2 are closed, proper, and convex.

$$\mathbf{G} = \begin{bmatrix} \mathbf{B}\mathbf{W} \\ \mathbf{I} \end{bmatrix} \text{ has full column rank.}$$

Handling the Matrix Inversion: Frame-Based Synthesis

Frame-based analysis: $\left[\sum_{j=1}^J (\mathbf{H}^{(j)})^* \mathbf{H}^{(j)} \right]^{-1} = \left[\mathbf{W}^* \mathbf{B}^* \mathbf{B} \mathbf{W} + \mathbf{I} \right]^{-1}$

DFT

Periodic deconvolution: $\mathbf{B} = \mathbf{U}^* \mathbf{D} \mathbf{U}$

diagonal matrix

$O(n \log n)$ $\left[\mathbf{W}^* \mathbf{B}^* \mathbf{B} \mathbf{W} + \mathbf{I} \right]^{-1} = \mathbf{I} - \mathbf{W}^* \mathbf{U}^* \mathbf{D}^* \left[|\mathbf{D}|^2 + \mathbf{I} \right]^{-1} \mathbf{D} \mathbf{U} \mathbf{W}$

matrix inversion lemma + $\mathbf{W} \mathbf{W}^* = \mathbf{I}$

subsampling matrix: $\mathbf{M} \mathbf{M}^* = \mathbf{I}$

Compressive imaging (MRI): $\mathbf{B} = \mathbf{M} \mathbf{U}$

$O(n \log n)$ $\left[\mathbf{W}^* \mathbf{U}^* \mathbf{M}^* \mathbf{M} \mathbf{U} \mathbf{W} + \mathbf{I} \right]^{-1} = \mathbf{I} - \frac{1}{2} \mathbf{W}^* \mathbf{U}^* \mathbf{M}^* \mathbf{M} \mathbf{U} \mathbf{W}$

subsampling matrix: $\mathbf{S} \mathbf{S}^* = \mathbf{I}$

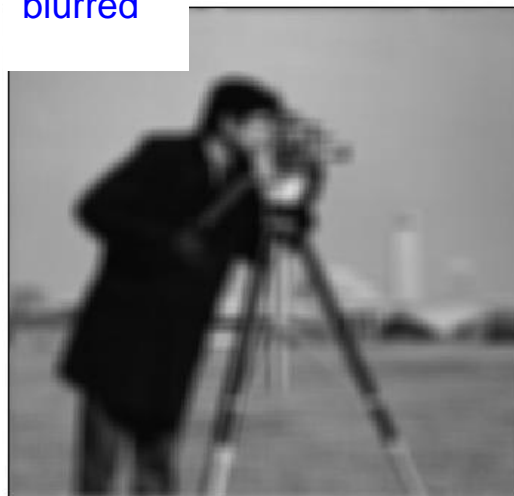
Inpainting (recovery of lost pixels): $\mathbf{B} = \mathbf{S}$

$O(n \log n)$ $\left[\mathbf{W}^* \mathbf{S}^* \mathbf{S} \mathbf{W} + \mathbf{I} \right]^{-1} = \mathbf{I} - \frac{1}{2} \mathbf{W}^* \mathbf{S}^* \mathbf{S} \mathbf{W}$

SALSA Experiments

9x9 uniform blur,
40dB BSNR

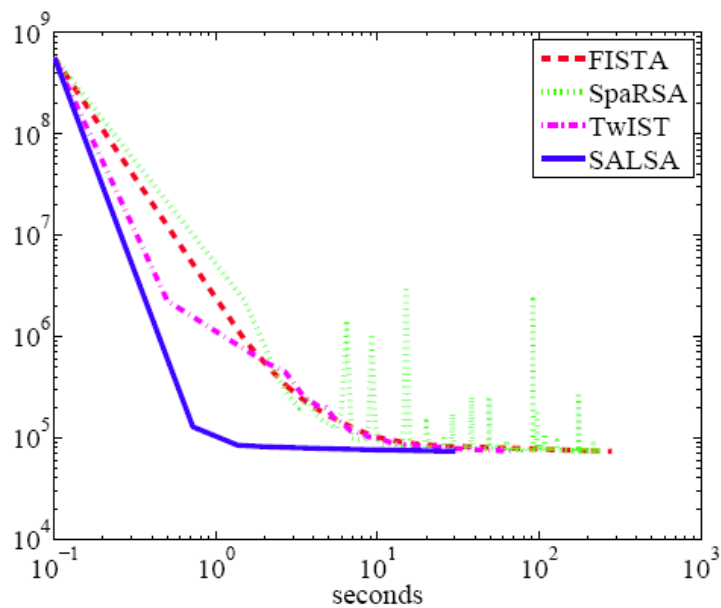
blurred



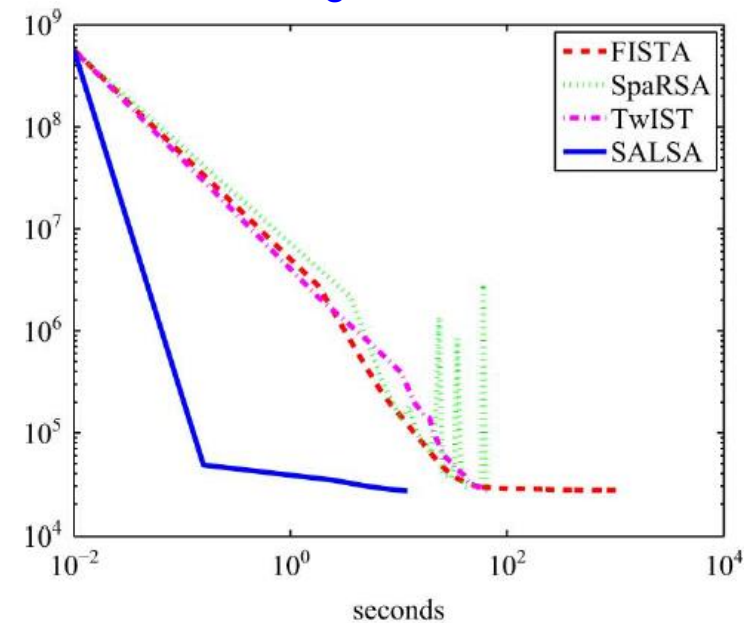
restored



undecimated Haar frame, ℓ_1 regularization.

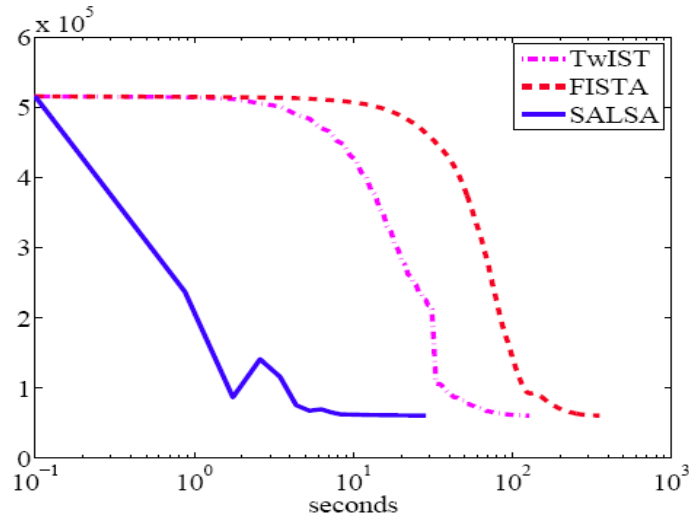


TV regularization



SALSA Experiments

Image inpainting
(50% missing)



Alg.	Calls to \mathbf{B}, \mathbf{B}^H	Iter.	CPU time (sec.)	MSE MSE	ISNR (dB)
FISTA	1022	340	263.8	92.01	18.96
TwIST	271	124	112.7	100.92	18.54
SALSA	84	28	20.88	77.61	19.68

Conjecture: SALSA is fast because it's *blessed* by the matrix inversion;
e.g., $\mathbf{A}^* \mathbf{A} + \mathbf{I}$ is the (regularized) Hessian of the data term;
...second-order (curvature) information (Newton, Levenberg-Maquardt)

Frame-Based Analysis Deconvolution of Poissonian Images

Problem template:
$$\min_{\mathbf{u} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{u}) \quad (P1)$$

positivity
constraint

Frame-analysis regularization:
$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \mathcal{L}_P(\mathbf{B} \mathbf{x}) + \lambda \|\mathbf{P} \mathbf{x}\|_1 + \iota_{\mathbb{R}_+^n}(\mathbf{x})$$

Same form as (P1) with: $J = 3$, $g_1 = \mathcal{L}_P$, $g_2 = \|\cdot\|_1$, $g_3 = \iota_{\mathbb{R}_+^n}$

Convergence conditions: g_1 , g_2 , and g_3 are closed, proper, and convex.

$$\mathbf{G} = \begin{bmatrix} \mathbf{B} \\ \mathbf{P} \\ \mathbf{I} \end{bmatrix} \quad \text{has full column rank}$$

Required inversion:
$$\left[\mathbf{B}^* \mathbf{B} + \mathbf{P}^* \mathbf{P} + \mathbf{I} \right]^{-1} = \left[\mathbf{B}^* \mathbf{B} + 2\mathbf{I} \right]^{-1}$$

...again, easy in periodic deconvolution, MRI, inpainting, ...

Proximity Operator of the Poisson Log-Likelihood

Proximity operator of the Poisson log-likelihood

$$\text{prox}_{\mathcal{L}/\mu}(\mathbf{u}) = \arg \min_{\mathbf{z}} \sum_i \xi(z_i, y_i) + \frac{\mu}{2} \|\mathbf{z} - \mathbf{u}\|_2^2$$

$$\xi(z, y) = z + \iota_{\mathbb{R}_+}(z) - y \log(z_+)$$

Separable problem with closed-form (non-negative) solution

[Combettes, Pesquet, 09, 11]:

$$\text{prox}_{\xi(\cdot, y)}(u) = \frac{1}{2} \left(u - \frac{1}{\mu} + \sqrt{\left(u - \frac{1}{\mu}\right)^2 + 4y/\mu} \right)$$

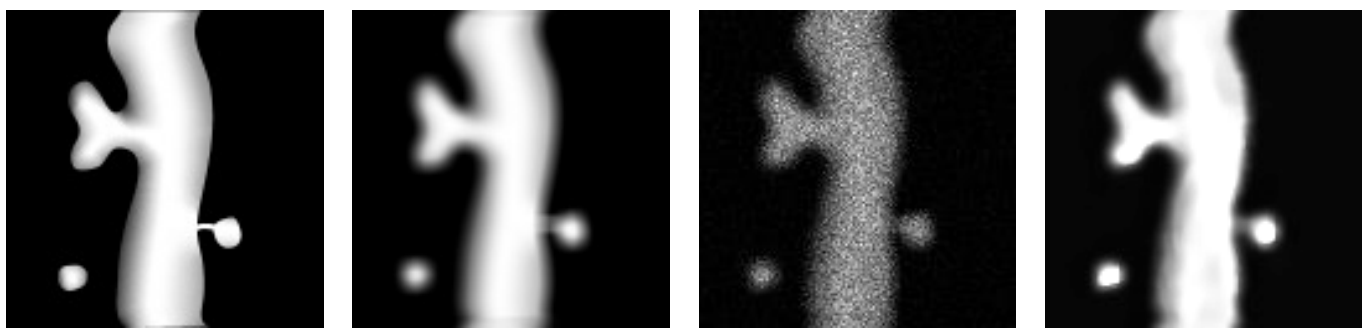
Proximity operator of $g_3 = \iota_{\mathbb{R}_+^n}$ is simply $\text{prox}_{\iota_{\mathbb{R}_+^n}}(\mathbf{x}) = (\mathbf{x})_+$

Experiments

PIDAL = Poisson image deconvolution by augmented Lagrangian
 [F and Bioucas-Dias, 2010]

Comparison with [Dupé, Fadili, Starck, 09] and [Starck, Bijaoui, Murtagh, 95]

Image	M	PIDAL-TV			PIDAL-FA			[Dupé, Fadili, Starck, 09]			[Starck et al, 95]
		MAE	iterations	time	MAE	iterations	time	MAE	iterations	time	MAE
Cameraman	5	0.27	120	22	0.26	70	13	0.35	6	4.5	0.37
Cameraman	30	1.29	51	9.1	1.22	39	7.4	1.47	98	75	2.06
Cameraman	100	3.99	33	6.0	3.63	36	6.8	4.31	426	318	5.58
Cameraman	255	8.99	32	5.8	8.45	37	7.0	10.26	480	358	12.3
Neuron	5	0.17	117	3.6	0.18	66	2.9	0.19	6	3.9	0.19
Neuron	30	0.68	54	1.8	0.77	44	2.0	0.82	161	77	0.95
Neuron	100	1.75	43	1.4	2.04	41	1.8	2.32	427	199	2.88
Neuron	255	3.52	43	1.4	3.47	42	1.9	5.25	202	97	6.31
Cell	5	0.12	56	10	0.11	36	7.6	0.12	6	4.5	0.12
Cell	30	0.57	31	6.5	0.54	39	8.2	0.56	85	64	0.47
Cell	100	1.71	85	15	1.46	31	6.4	1.72	215	162	1.37
Cell	255	3.77	89	17	3.33	34	7.0	5.45	410	308	3.10



$$MAE \equiv \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_1}{n}$$

Non-Periodic Deconvolution

Analysis formulation for deconvolution $\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \tau c(\mathbf{x})$

ADMM / SALSA easy (only?) if \mathbf{A} is circulant (periodic convolution via FFT)

Periodicity is an **artificial** assumption

...as are other boundary conditions (BC)

Neumann

Dirichlet



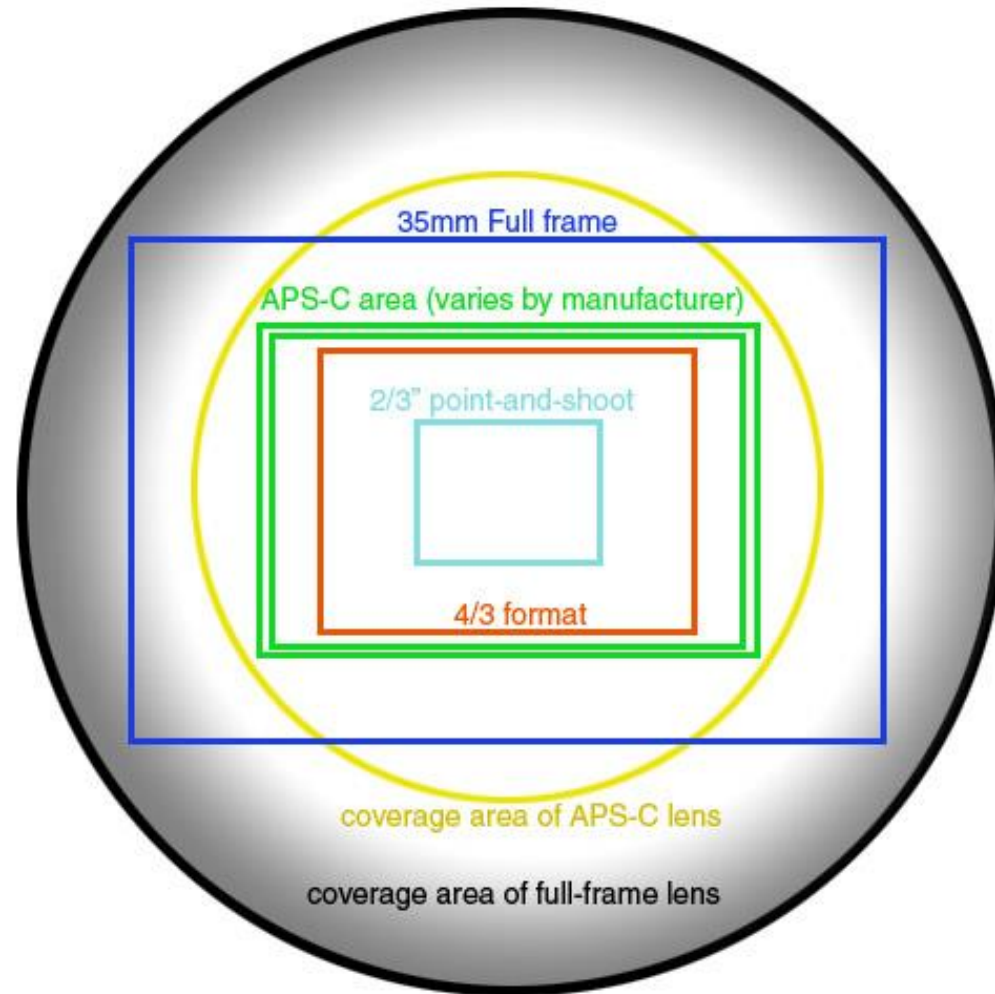
\mathbf{A} is (block) circulant

\mathbf{A} is (block) Toeplitz + Hankel

\mathbf{A} is (block) Toeplitz

[Ng, Chan, Tang, 1999]

Why Periodic, Neumann, Dirichlet Boundary Conditions are “wrong”



Non-Periodic Deconvolution

The natural choice: the boundary is unknown

[Chan, Yip, Park, 05], [Reeves, 05], [Sorel, 12], [Almeida, F, 12,13], [Matakos, Ramani, Fessler, 12, 13]

convolution, arbitrary BC

masking



 unknown values

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{M}\mathbf{B}\mathbf{x} - \mathbf{y}\|_2^2 + \tau c(\mathbf{x})$$

mask

periodic convolution

Non-Periodic Deconvolution (Frame-Analysis)

Problem: $\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{MBx} - \mathbf{y}\|_2^2 + \tau \|\mathbf{Px}\|_1$

Template: $\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$

Naïve mapping: $J = 2$, $g_1(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2$, $g_2(\mathbf{z}) = \tau \|\mathbf{z}\|_1$
 $\mathbf{H}^{(1)} = \mathbf{MB}$ $\mathbf{H}^{(2)} = \mathbf{P}$,

Difficulty: need to compute $\left[\mathbf{B}^* \mathbf{M}^* \mathbf{MB} + \mathbf{P}^* \mathbf{P} \right]^{-1} = \left[\mathbf{B}^* \mathbf{M}^* \mathbf{MB} + \mathbf{I} \right]^{-1}$

...the tricks above are no longer applicable.

Non-Periodic Deconvolution (Frame-Analysis)

Problem: $\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{M}\mathbf{B}\mathbf{x} - \mathbf{y}\|_2^2 + \tau \|\mathbf{P}\mathbf{x}\|_1$

Template: $\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$

Better mapping: $J = 2, \quad g_1(\mathbf{z}) = \frac{1}{2} \|\mathbf{M}\mathbf{z} - \mathbf{y}\|_2^2, \quad g_2(\mathbf{z}) = \tau \|\mathbf{z}\|_1$
 $\mathbf{H}^{(1)} = \mathbf{B} \qquad \mathbf{H}^{(2)} = \mathbf{P},$

$$\left[\mathbf{B}^* \mathbf{B} + \mathbf{P}^* \mathbf{P} \right]^{-1} = \left[\mathbf{B}^* \mathbf{B} + \mathbf{I} \right]^{-1} \quad \text{easy via FFT (B is circulant)}$$

$$\text{prox}_{g_1/\mu}(\mathbf{u}) = \arg \min_{\mathbf{z}} \frac{1}{2\mu} \|\mathbf{M}\mathbf{z} - \mathbf{y}\|_2^2 + \frac{1}{2} \|\mathbf{z} - \mathbf{u}\|_2^2$$

$$= \text{diagonal} \left(\mathbf{M}^T \mathbf{M} + \mu \mathbf{I} \right)^{-1} (\mathbf{M}^T \mathbf{y} + \mu \mathbf{u})$$

Non-Periodic Deconvolution: Example (19x19 uniform blur)

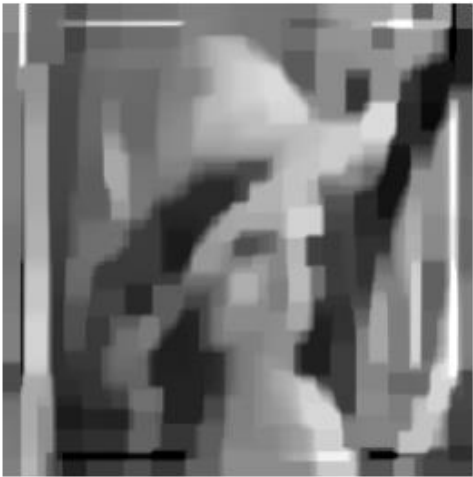


original (256 × 256)



observed (238 × 238)

Assuming periodic BC



FA-BC (ISNR = -2.52dB)

Edge tapering



FA-ET (ISNR = 3.06dB)

Proposed



FA-MD (ISNR = 10.63dB)

Non-Periodic Deconvolution: Example (19x19 motion blur)



original (256 × 256)



observed (238 × 238)

Assuming periodic BC



TV-BC (ISNR = 0.91dB)

Edge tapering



TV-ET (ISNR = 9.38dB)

Proposed



TV-MD (ISNR = 12.59dB)

Non-Periodic Deconvolution + Inpainting

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{MBx} - \mathbf{y}\|_2^2 + \tau c(\mathbf{x})$$

Mask the boundary
and the missing pixels

periodic convolution

Also applicable to super-resolution
(ongoing work)



original (256 × 256)



observed (238 × 238)



FA-CG (SNR = 20.58dB)



FA-MD (SNR = 20.57dB)

Non-Periodic Deconvolution via Accelerated IST

The synthesis formulation is easily handled by IST (or FISTA, TwIST, SpaRSA,...)
 [Matakos, Ramani, Fessler, 12, 13]

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{MBW}\mathbf{x} - \mathbf{y}\|_2^2 + \tau \|\mathbf{x}\|_1$$

periodic convolution
Parseval frame synthesis

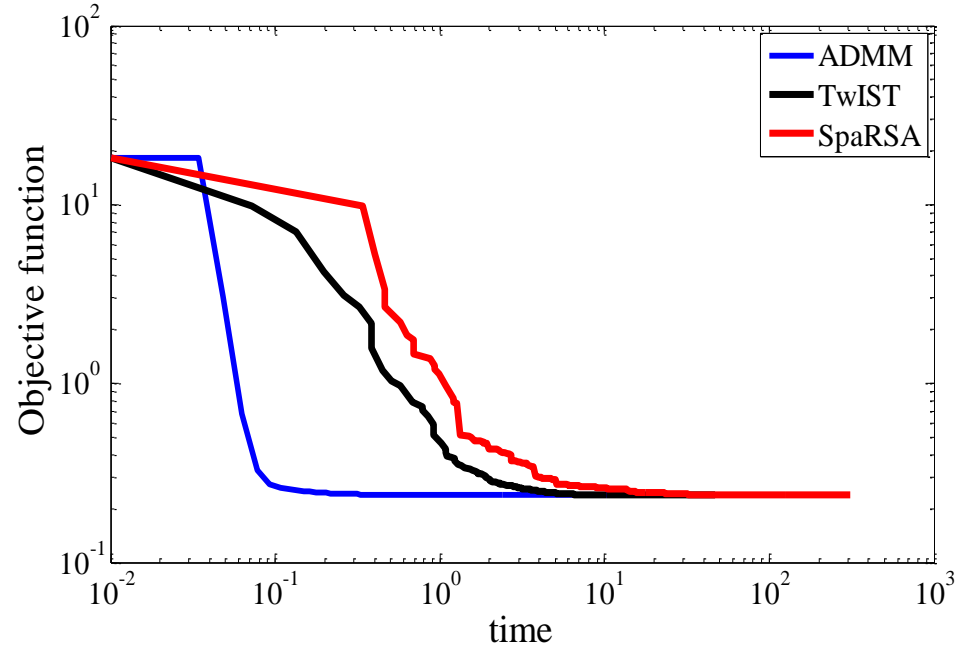
mask

Ingredients:

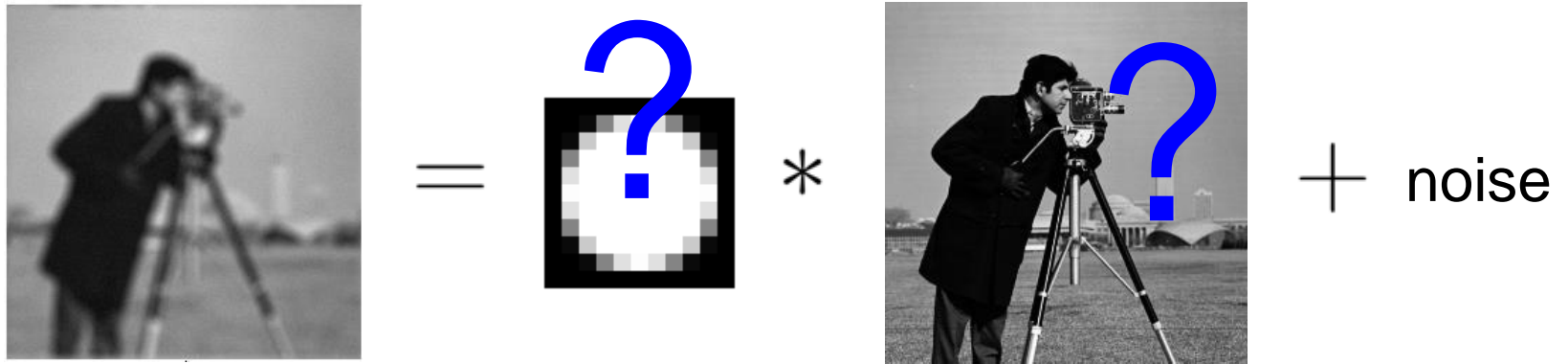
$$\text{prox}_{\tau \|\cdot\|_1}(\mathbf{u}) = \text{soft}(\mathbf{u}, \tau)$$

$$\nabla \frac{1}{2} \|\mathbf{MBW}\mathbf{x} - \mathbf{y}\|_2^2 = \mathbf{W}^* \mathbf{B}^* \mathbf{M}^* (\mathbf{MBW}\mathbf{x} - \mathbf{y})$$

(analysis formulation cannot be addressed by IST, FISTA, SpaRSA, TwIST,...)



Into the Non-convex Realm: Blind Image Deconvolution (BID)

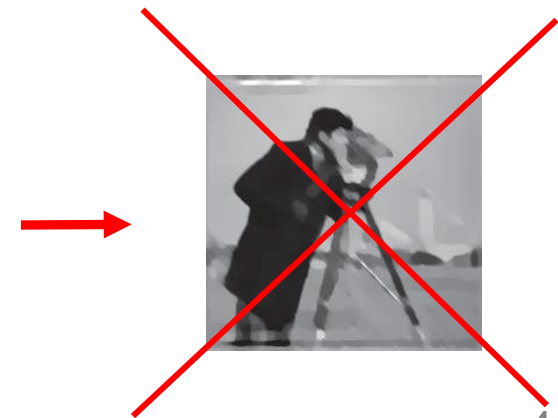


Degradation model: $y = \mathbf{h} * \mathbf{x} + \mathbf{n}$

Difficulties:

- Ill-posed*: - infinite number of solutions.
- ill-conditioned blurring operator.

Unknown boundaries (usually ignored)



BID Methods and Restrictions on the Blurring Filter

Hard restrictions:

(parameterized filters)

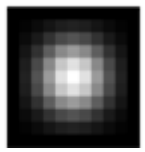
- **circular** blurs:
[Yin *et al*, 06]



- **linear** blurs:
[Krahmer *et al*, 06]
[Oliveira *et al*, 07]



- **Gaussian** blurs:
[Rooms *et al*, 04]
[Krylov *et al*, 09]



Soft restrictions:

(regularized filters)

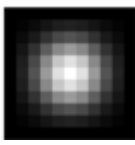
- **TV** regularization:
[Babacan *et al* 09],
[Amizic *et al* 10], [li, 12]



- **Sparse** regularization:
[Fregus *et al*, 06];
[Levin *et al*, 09, 11]
[Shan *et al*, 08], [Cho, 09]
[Krishnan, 11],[Xu, 11], [Cai, 12]



- **Smooth** regularization:
[Joshi *et al*, 08];
Babacan *et al*, 09]



Blind Image Deconvolution (BID): Formulation

$$\mathbf{y} = \mathbf{h} * \mathbf{x} + \mathbf{n} \quad \text{Both } \mathbf{x} \text{ and } \mathbf{h} \text{ are unknown}$$

Objective function (non-convex):

$$\mathbf{C}_\lambda(\mathbf{x}, \mathbf{h}) = \frac{1}{2} \|\mathbf{y} - \mathbf{M} \mathbf{B} \mathbf{x}\|_2^2 + \lambda \underbrace{\sum_{i=1}^m (\|\mathbf{F}_i \mathbf{x}\|_2)^q}_{\Phi(\mathbf{x})} + \iota_{\mathcal{S}^+}(\mathbf{h})$$

Support and positivity

Boundary mask

Matrix representation of the convolution with \mathbf{h}

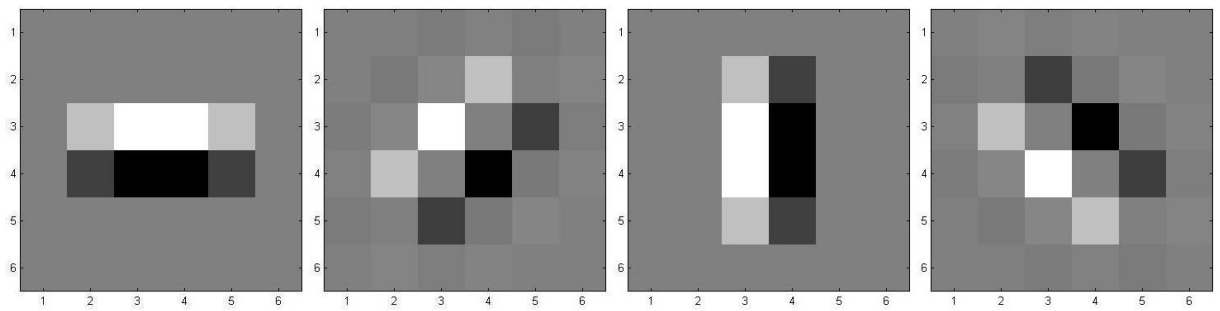
[Almeida and F, 13]

$\Phi(\mathbf{x})$ is “enhanced” TV; $q \in (0, 1]$ (typically 0.5);

\mathbf{F}_i is the convolution with four “edge filters” at location i

$$\mathbf{F}_i \in \mathbb{R}^{4 \times m}$$

$$\mathbf{F}_i \mathbf{x} \in \mathbb{R}^4$$



Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

- 1 Set $\hat{\mathbf{h}}$ to the identity filter, $\hat{\mathbf{x}} = \mathbf{y}$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
 - 2 **repeat**
 - 3 $\hat{\mathbf{x}} \leftarrow \arg \min_{\mathbf{x}} \mathbf{C}_\lambda(\mathbf{x}, \hat{\mathbf{h}})$ **update image estimate**
 - 4 $\hat{\mathbf{h}} \leftarrow \arg \min_{\mathbf{h}} \mathbf{C}_\lambda(\hat{\mathbf{x}}, \mathbf{h})$, **update blur estimate**
 - 5 $\lambda \leftarrow \alpha \lambda$
 - 6 **until** *stopping criterion is satisfied*
-

[Almeida et al, 2010, 2013]

Updating the image estimate

$$\hat{\mathbf{x}} \leftarrow \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{H}\mathbf{x}\|^2 + \lambda\Phi(\mathbf{x})$$

Standard image deconvolution, with unknown boundaries; ADMM as above.

Blind Image Deconvolution (BID)

Updating the image estimate

$$\hat{\mathbf{x}} \leftarrow \arg \min_{\mathbf{x} \in \mathbb{R}^m} \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{B}\mathbf{x}\|^2 + \lambda \sum_{i=1}^m (\|\mathbf{F}_i \mathbf{x}\|_2)^q$$

Template: $\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^J g_j(\mathbf{H}^{(j)} \mathbf{z})$

Mapping: $J = m + 1, \quad g_i(\mathbf{z}) = \|\mathbf{z}\|_2^q, \quad i = 1, \dots, m,$

$$\mathbf{H}^{(i)} = \mathbf{F}_i, \quad i = 1, \dots, m,$$

$$g_{m+1}(\mathbf{z}) = \frac{1}{2} \|\mathbf{M}\mathbf{z} - \mathbf{y}\|_2^2, \quad \mathbf{H}^{(m+1)} = \mathbf{B}$$

All the matrices are circulant: matrix inversion step in ADMM easy with FFT.

Also possible to compute $\text{prox}_{\tau \|\cdot\|_2^q}(\mathbf{u}) = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{u}\|_2^2 + \tau \|\mathbf{x}\|_2^q$

for $q \in \{0, \frac{1}{2}, \frac{2}{3}, 1, \frac{4}{3}, \frac{3}{2}, 2\}$

Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

- 1 Set $\hat{\mathbf{h}}$ to the identity filter, $\hat{\mathbf{x}} = \mathbf{y}$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
 - 2 **repeat**
 - 3 $\hat{\mathbf{x}} \leftarrow \arg \min_{\mathbf{x}} \mathbf{C}_\lambda(\mathbf{x}, \hat{\mathbf{h}})$ update image estimate
 - 4 $\hat{\mathbf{h}} \leftarrow \arg \min_{\mathbf{h}} \mathbf{C}_\lambda(\hat{\mathbf{x}}, \mathbf{h})$, update blur estimate
 - 5 $\lambda \leftarrow \alpha \lambda$
 - 6 **until** *stopping criterion is satisfied*
-

Updating the blur estimate: notice that $\mathbf{h} * \mathbf{x} = \mathbf{H}\mathbf{x} = \mathbf{X}\mathbf{h}$

$$\hat{\mathbf{h}} \leftarrow \arg \min_{\mathbf{h}} \frac{1}{2} \|\mathbf{y} - \mathbf{M}\mathbf{X}\mathbf{h}\|^2 + \iota_{\mathcal{S}^+}(\mathbf{h})$$

Like standard image deconvolution, with a support and positivity constraint.

Prox of support and positivity constraint is trivial: $\text{prox}_{\iota_{\mathcal{S}^+}}(\mathbf{h}) = \Pi_{\mathcal{S}^+}(\mathbf{h})$

Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

- 1 Set $\hat{\mathbf{h}}$ to the identity filter, $\hat{\mathbf{x}} = \mathbf{y}$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
 - 2 **repeat**
 - 3 $\hat{\mathbf{x}} \leftarrow \arg \min_{\mathbf{x}} \mathbf{C}_\lambda(\mathbf{x}, \hat{\mathbf{h}})$
 - 4 $\hat{\mathbf{h}} \leftarrow \arg \min_{\mathbf{h}} \mathbf{C}_\lambda(\hat{\mathbf{x}}, \mathbf{h}),$
 - 5 $\lambda \leftarrow \alpha \lambda$
 - 6 **until** *stopping criterion is satisfied*
-

Question: when to stop? What value of λ to choose?

For non-blind deconvolution, many approaches for choosing λ

generalized cross validation, L-curve, SURE and variants thereof

[Bertero, Poggio, Torre, 88], [Thomson, Brown, Kay, Titterington, 92], [Galatsanos, Kastagellos, 92],

[Hansen, O'Leary, 93], [Eldar, 09], [Giryas, Elad, Eldar 11], [Luisier, Blu, Unser 09], [Ramani, Blu, Unser, 10],

[Ramani, Rosen, Nielsen, Fessler, 12],...

Bayesian methods (some for BID)

[Babacan, Molina, Katsaggelos, 09], [Fergus et al, 06], [Amizic, Babacan, Molina, Katsaggelos, 10],

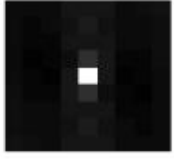
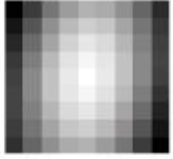
[Chantas, Galatsanos, Molina, Katsaggelos, 10], [Oliveira, Bioucas-Dias, F, 09]

No-reference quality measures [Lee, Lai, Chen, 07], [Zhu, Milanfar, 10]

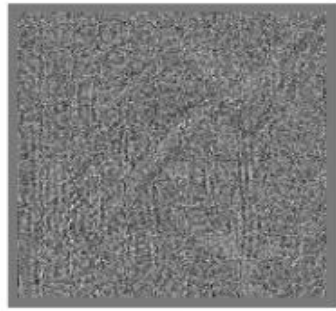
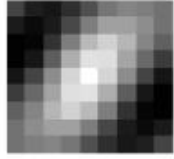
Blind Image Deconvolution: Stopping Criterion

Proposed rationale: if the blur kernel is well estimated, the residual is white.

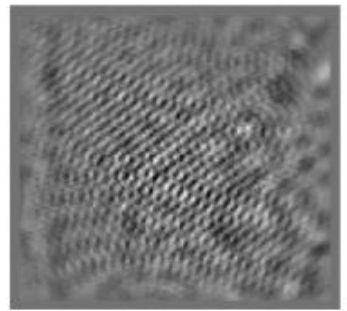
Autocorrelation:



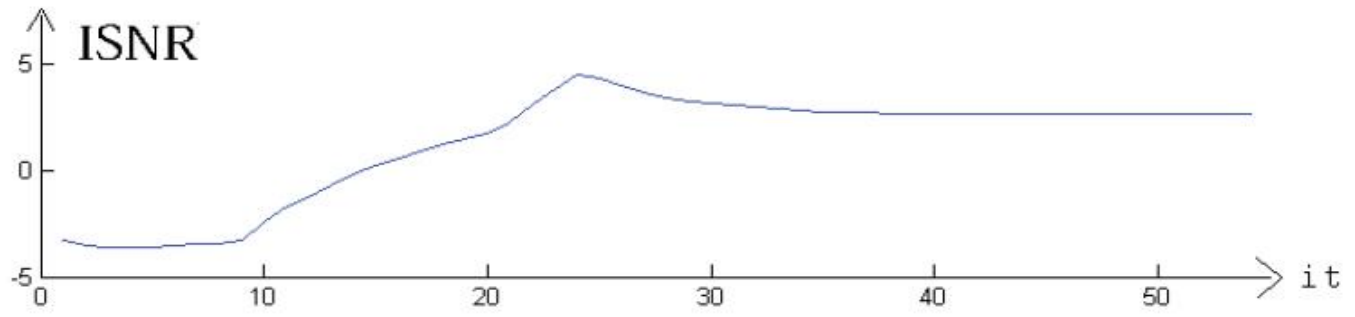
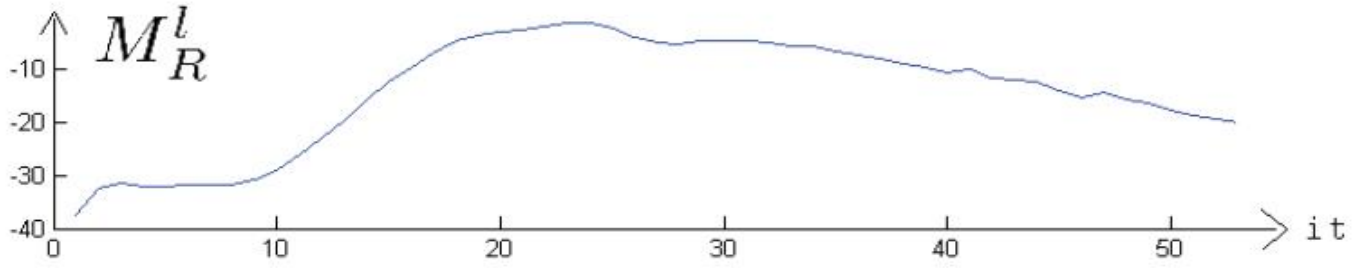
$$R_{rr}$$



Estimated residual



Whiteness:



DEMO

Blind Image Deconvolution (BID)

Experiment with real motion blurred photo



Blurred photo



[14], 70 seconds

[Krishnan et al, 2011]



[16], 100 seconds

[Levin et al, 2011]



Proposed method, 55 seconds

Blind Image Deconvolution (BID)

Experiment with real out-of-focus photo



Observed photo.



[Almeida et al, 2010]



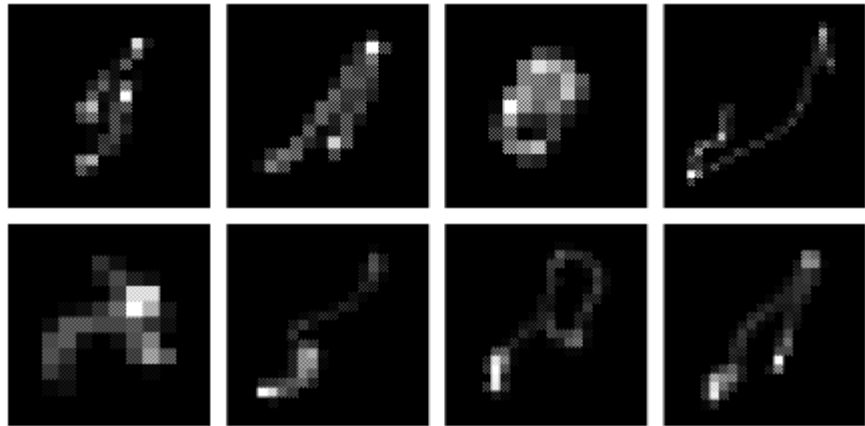
proposed

Blind Image Deconvolution (BID): Synthetic Results

Realistic motion blurs:

[Levin, Weiss, Durant, Freeman, 09]

Images: Lena, Cameraman



Average results over 2 images and 8 blurs:

	Method	∞ dB	40dB	30dB
ISNR* (dB)	[31]	6.14	5.90	4.91
	[35]	5.51	5.72	4.79
	[50]	4.70	4.70	4.30
	Ours	9.00	8.43	6.70
Time (s)	[31]	80	66	62
	[35]	399	399	399
	[50]	1.5^2	1.5^2	1.5^2
	Ours	70	55	45

[Krishnan et al, 11]

[Levin et al, 11]

[Xu, Jia, 10]

[Krishnan et al, 11]

[Levin et al, 11]

[Xu, Jia, 10] (GPU)

Blind Image Deconvolution (BID): Handling Saturations

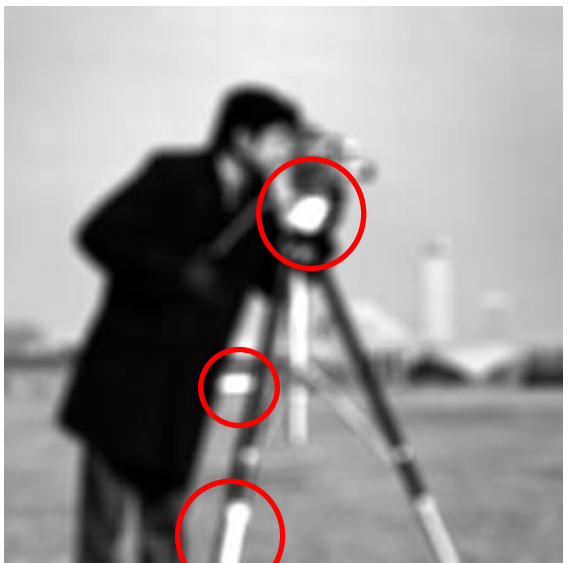
Several digital images have **saturated** pixels (at 0 or max): this impacts BID!

Easy to handle in our approach: **just mask them out**

$$\min(\alpha \mathbf{x} * \mathbf{h}, 255)$$

ignoring saturations

knowing saturations



out-of-focus (disk) blur



Summary:

- Alternating direction optimization (ADMM) is powerful, versatile, modular.
- Main hurdle: need to solve a linear system (invert a matrix) at each iteration...
- ...however, sometimes this turns out to be an advantage.
- State of the art results in several image/signal reconstruction problems.

Thanks!