RED: Regularization by Denoising

The Little Engine that Could

Michael Elad

May 18, 2017

TECHNION
Israel Institute
of Technology

Joint work with:

Yaniv Romano

Peyman Milanfar

This work was done during the summer 2016 in Google-Research Mountain-View, CA

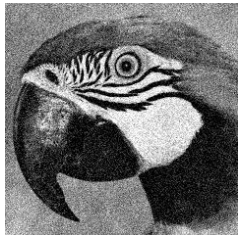
Background and Main Objective

Image Denoising: Definition

This is the “simplest” inverse problem

$$y = x + e$$

Measured Clean Noise (AWGN)



y

Filter

An output as close as possible to

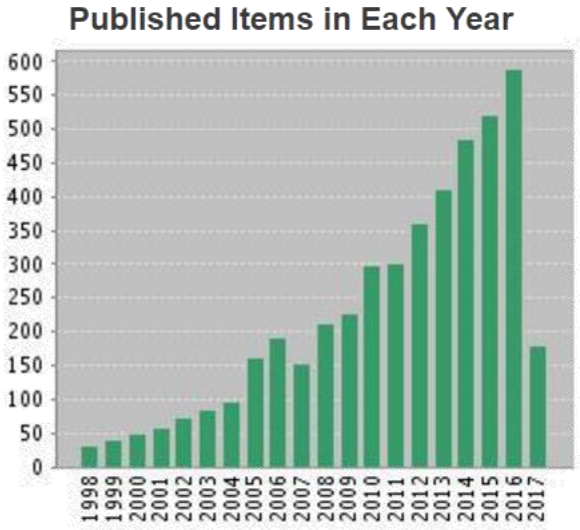
x

Image Denoising – Past Work

Searching Web-of-Science (May 4th, 2017) with

```
TOPIC = image and noise and (denoising or removal)
```

Leads to ~4700 papers



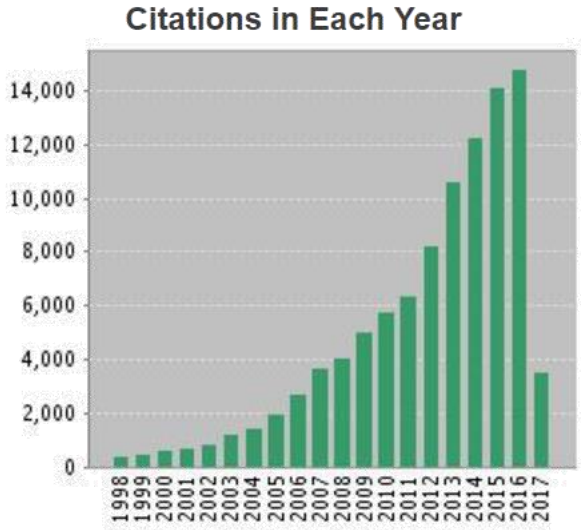
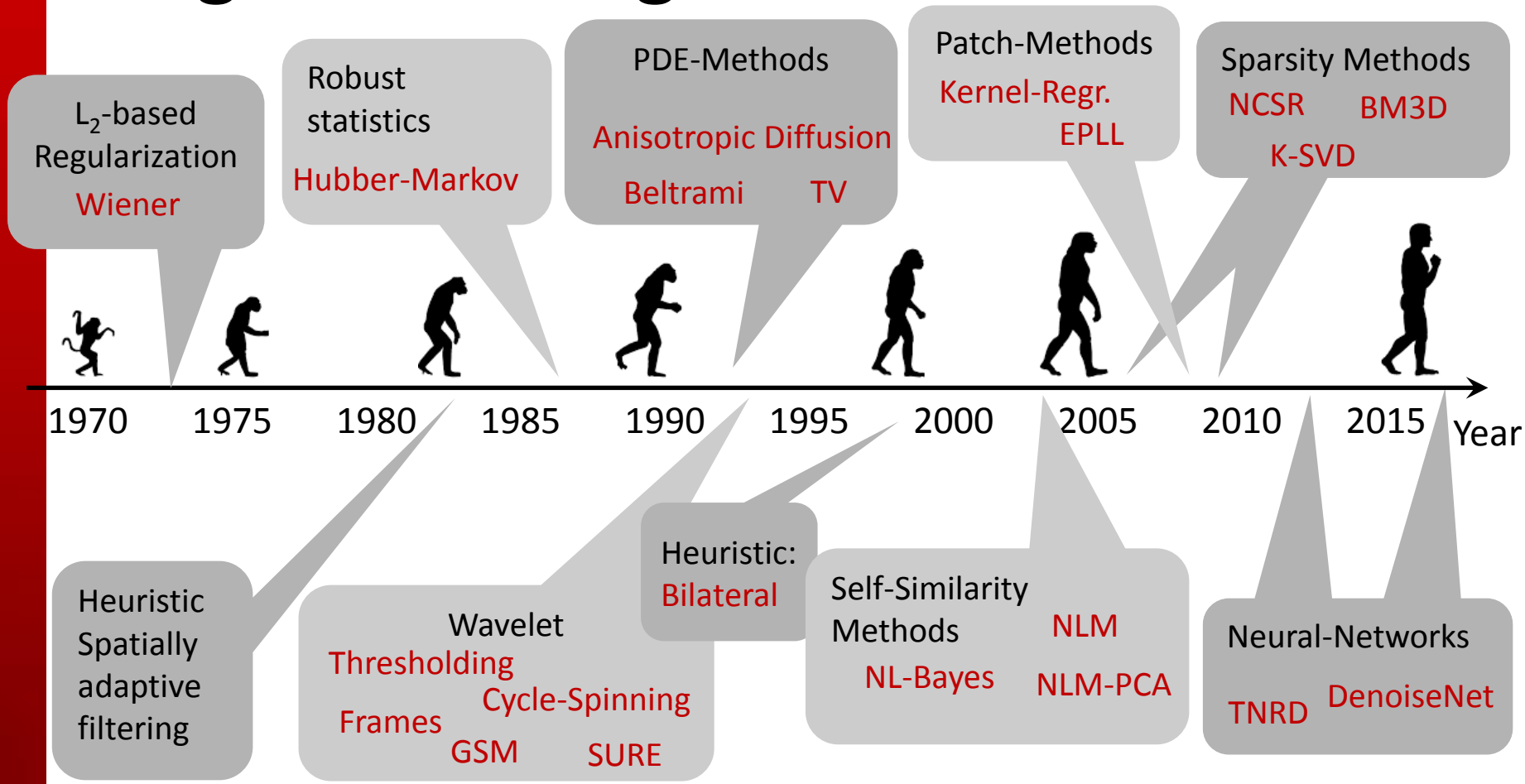


Image Denoising – What's Out There?



Is Denoising Dead?

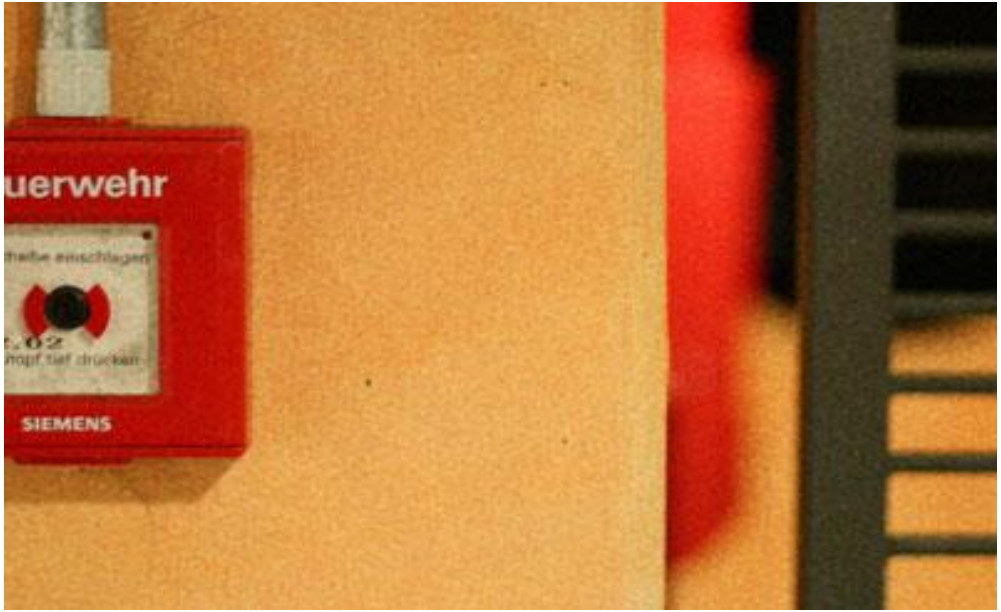
To a large extent, removal of additive noise from an image is a solved problem in image processing

This claim is based on several key observations:

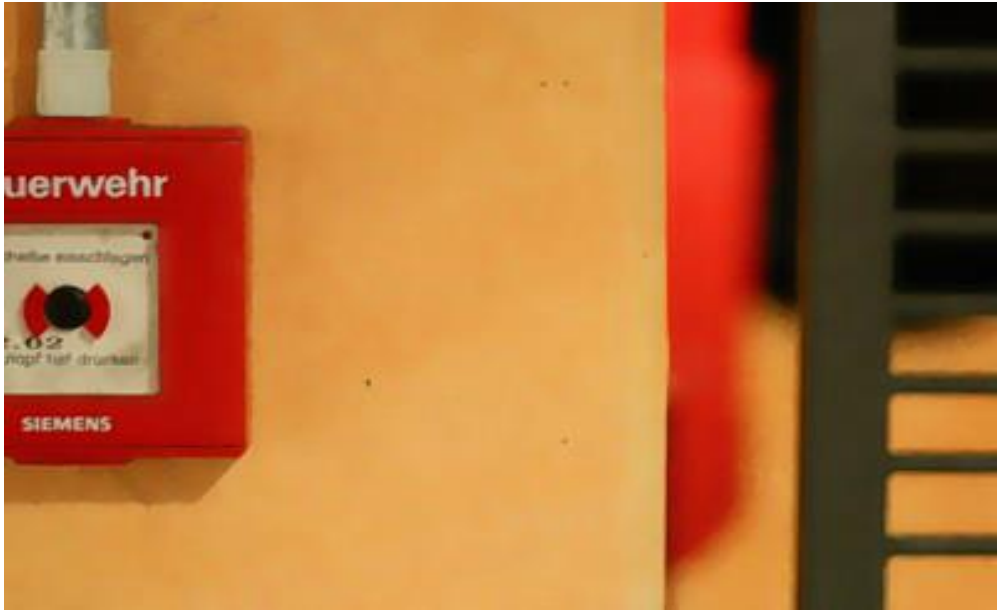
- ❑ There are more than 20,000 papers studying this problem
- ❑ The proposed methods in the past decade are highly impressive
- ❑ Very different algorithms tend to lead to quite similar output quality
- ❑ Work investigating performance bounds confirm this
  [Chatterjee & Milanfar, 2010], [Levin & Nadler, 2011]

Bottom line: Improving image denoising algorithms seems to lead to diminishing returns

Noise Removal, Sharpening



Noise Removal, Sharpening

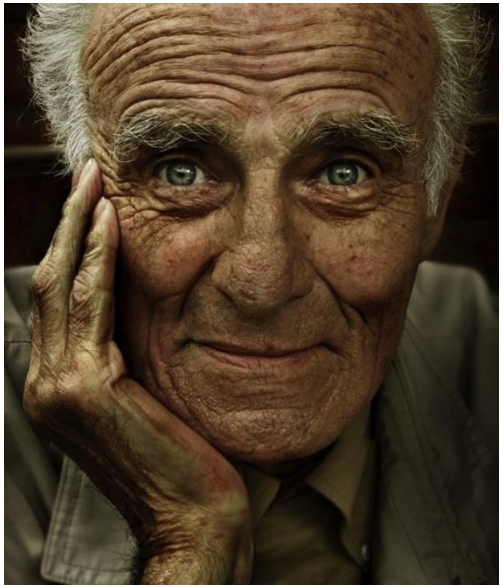


Compression Artifact Removal

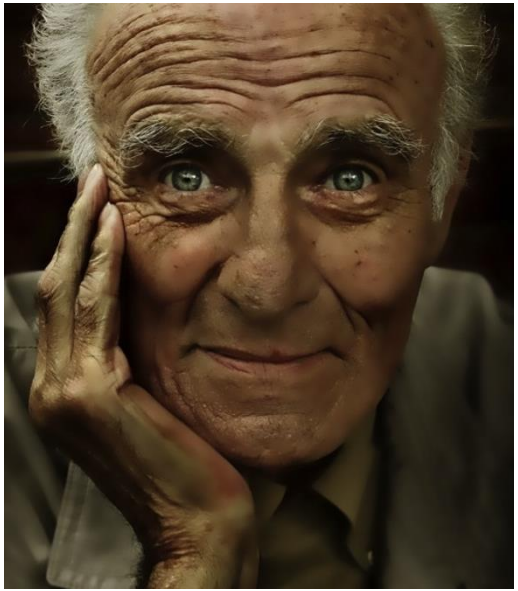
Compression Artifact Removal

Some Visual Effects

Old man



Younger



More fun

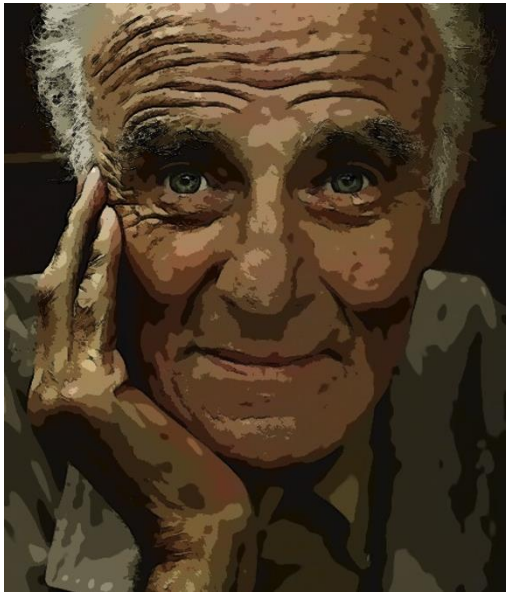


Image courtesy Andrzej Dragan

Image Denoising – Can we do More?

SO

if improving image denoising algorithms
is a dead-end avenue …

THEN

Lets seek ways to leverage existing denoising
algorithms in order to solve

OTHER (INVERSE) PROBLEMS

Inverse Problems ?

- Given the measurement y, recover its origin signal, x
- A statistical tie between x and y is given
- Examples: Image deblurring, inpainting, tomographic reconstruction, super-resolution, …
- Classic inverse problem: $y = \mathbf{H}x + v$ where $v \sim \mathbb{N}\left\{0, \sigma^2 \mathbf{I}\right\}$
- The Bayesian approach: use the conditional probability Prob(x|y) in order to estimate x. How?
  - Maximize the conditional probability (MAP)
  - Find the expected x given y
  - …

“The Little Engine that Could” ?

As we are about to see,
image denoisers
(we refer to these as “engines”)
are capable of far more than just
… denoising,
And thus this sub-title

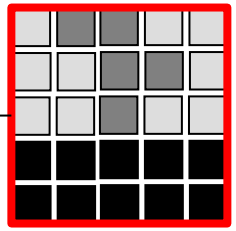
Prior-Art 1:

Laplacian Regularization

Pseudo-Linear Denoising

- Often times we can describe our denoiser as a pseudo-linear Filter

$$\text{Filter}\{x\} = \mathbf{W}(x)x$$

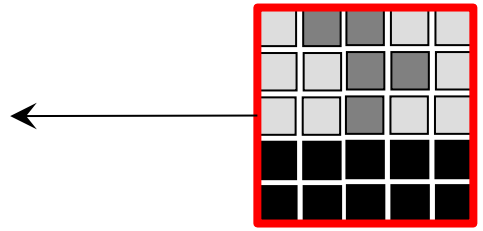


- True for K-SVD, EPLL, NLM, BM3D and other algorithms, where the overall processing is divided into a non-linear stage of decisions, followed by a linear filtering
- Typical properties of **W**:
  - Row stochastic: $\underline{1}$ is an eigenvector, spectral radius is 1
  - Fast decaying eigenvalues
  - Symmetry or Non-negativity of **W**? Not always …

The Image Laplacian

Given this form:

$$\text{Filter}\{x\} = \mathbf{W}(x)x$$



We may propose an image-adaptive (-)Laplacian:

$$\begin{aligned}\text{Laplacian}\{x\} &= x - \mathbf{W}(x)x \\ &= \left(\mathbf{I} - \mathbf{W}(x)\right)x \\ &= \mathbf{L}(x)x\end{aligned}$$

The “residual”

Laplacians as Regularization

Log-Likelihood Data Fidelity $\ell(x, y) + \frac{\lambda}{2} x^T \mathbf{L} x$ Regularization

$$\rho(x) = \frac{1}{2} x^T \mathbf{L} x = \frac{1}{2} x^T (x - \mathbf{W} x)$$

Signal Filter Residual

This idea appeared is a series of papers in several variations:

[Elmoataz, Lezoray, & Bougleux 2008] [Szlam, Maggioni, & Coifman 2008]
[Peyre, Bougleux & Cohen 2011] [Milanfar 2013] [Kheradmand & Milanfar 2014]
[Liu, Zhai, Zhao, Zhai, & Gao 2014] [Haque, Pai, & Govindu 2014] [Romano & Elad 2015]

Laplacians as Regularization

$$\ell(x,y)+\frac{\lambda}{2}x^{T}(x-\mathbf{W}x)$$

The problems with this line of work are that:

1. The regularization term is hard to work with since **L**/**W** is a function of x. This is circumvented by cheating and assuming a fixed **W** per each iteration
2. If so, what is really the underlying energy that is being minimized?
3. When the denoiser cannot admit a pseudo-linear interpretation of **W**(x)x, this term is not possible to use

Prior-Art 2:

The Plug-and-Play-Prior (P^3) Scheme

The P^3 Scheme

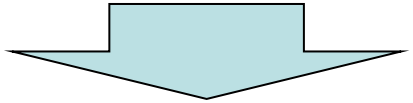
- The idea of using a denoiser to solve general denoising problems was proposed under the name "Plug-and-Play-Priors" (P^3) [Venkatakrishnan, Wohlberg & Bouman, 2013]
- Main idea: Use ADMM to minimize the MAP energy

$$\hat{x}_{MAP} = \min_x \ell(x,y) + \frac{\lambda}{2}\rho(x)$$

$$\min_{x,v} \ell(x,y) + \frac{\lambda}{2}\rho(v) \quad \text{s.t.} \quad x = v$$

The P^3 Scheme

$$\min_{x,v} \ell(x,y) + \frac{\lambda}{2}\rho(v) \quad \text{s.t.} \quad x = v$$

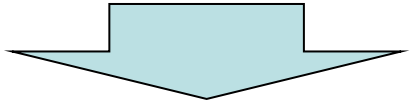


$$\min_{x,v} \ell(x,y) + \frac{\lambda}{2}\rho(v) + \frac{\beta}{2}\|x - v + u\|_2^2$$

- The above relies on a well-known concept in optimization called the augmented Lagrange algorithm, where u is the scaled Lagrange multiplier vector

The P³ Scheme

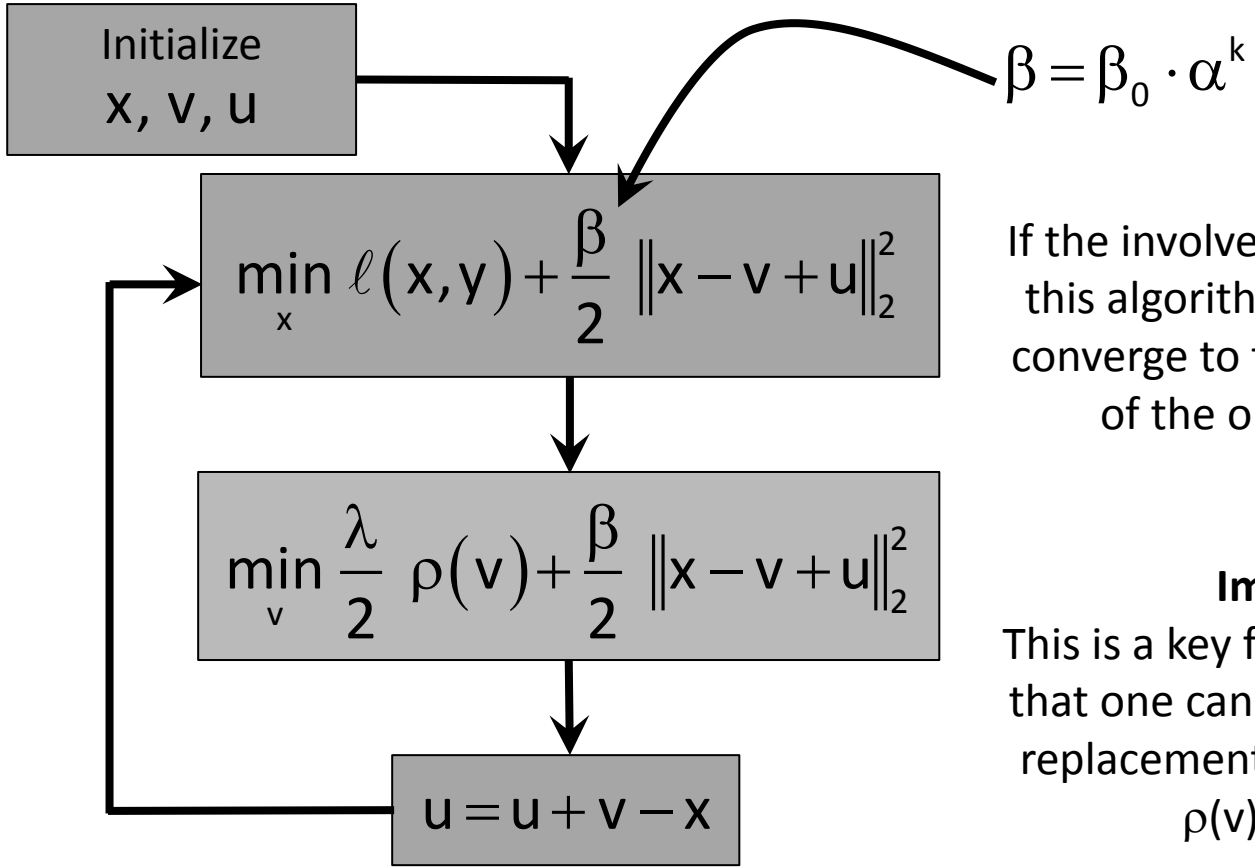
$$\min_{x,v} \ell(x,y) + \frac{\lambda}{2}\rho(v) + \frac{\beta}{2}\|x - v + u\|_2^2$$



- Minimize the above iteratively, in an alternating-direction fashion, w.r.t. the two unknowns (and update u=u-v+x):

1. $\min_{x} \ell(x,y) + \frac{\beta}{2}\|x - v + u\|_2^2$: Simple inverse problem
2. $\min_{v} \frac{\lambda}{2}\rho(v) + \frac{\beta}{2}\|x - v + u\|_2^2$

P^3 Algorithm

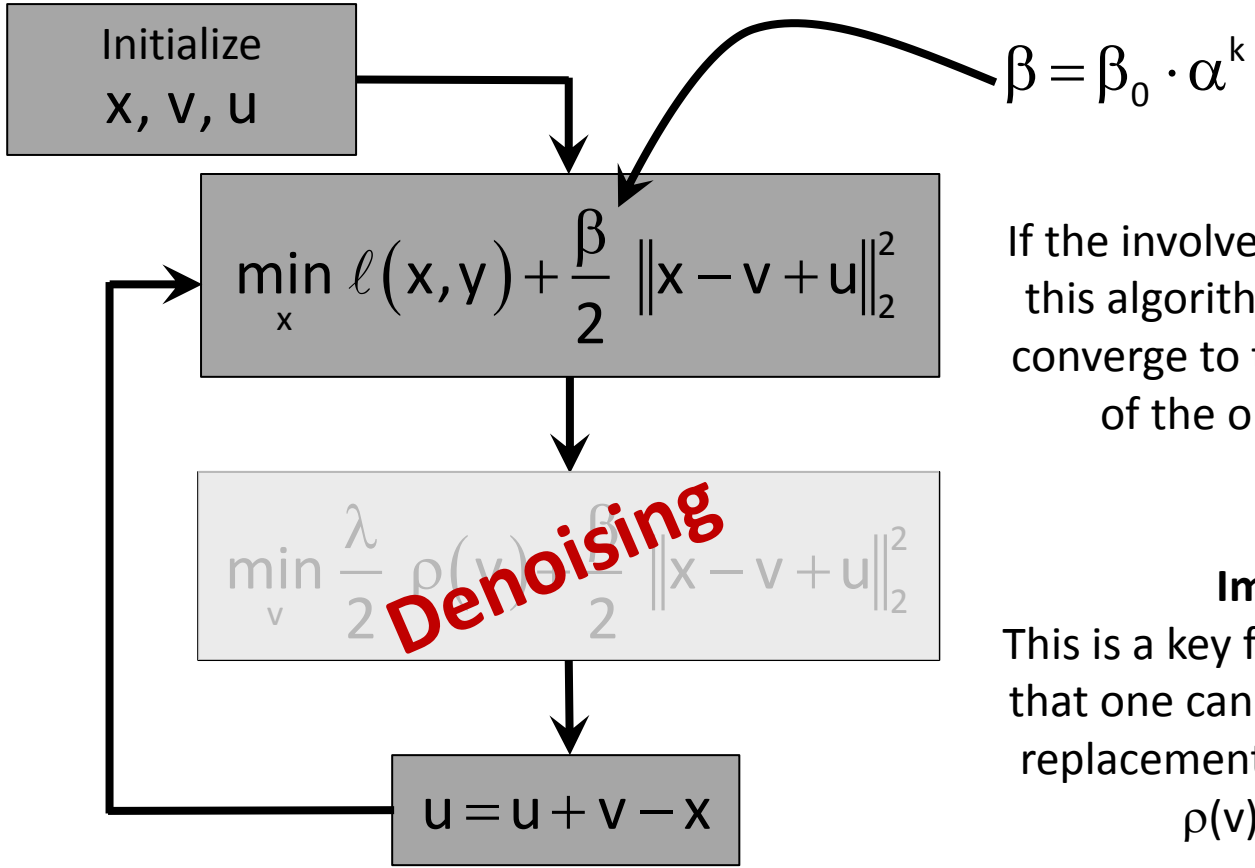


If the involved terms are convex, this algorithm is guaranteed to converge to the global optimum of the original function

Implicit Prior:

This is a key feature of P^3 – The idea that one can use ANY denoiser as a replacement for this stage, even if $\rho(v)$ is not known

P^3 Algorithm



If the involved terms are convex, this algorithm is guaranteed to converge to the global optimum of the original function

Implicit Prior:

This is a key feature of P^3 – The idea that one can use ANY denoiser as a replacement for this stage, even if $\rho(v)$ is not known

P^3 Shortcomings

- The P^3 scheme is an excellent idea, but it has few troubling shortcomings:
  - Parameter tuning is **TOUGH** when using a general denoiser
  - This method is tightly tied to **ADMM** without an option for changing this scheme
  - **CONVERGENCE** ? Unclear (steady-state at best)
  - For an arbitrarily denoiser, no underlying & consistent **COST FUNCTION**
- In this work we propose an alternative which is closely related to the above ideas (both Laplacian regularization and P^3) which overcomes the mentioned problems: RED

RED: First Steps

Regularization by Denoising [RED]

We suggest: $\rho(x)=\frac{1}{2}x^T(x-\mathbf{W}x)$

… for an arbitrary denoiser f(x)

$$\rho(x)=0 \Longrightarrow \begin{array}{l} 1.\ x=0 \\ 2.\ x=f(x) \\ 3.\ \text{Orthogonality} \end{array}$$

[Romano, Elad & Milanfar, 2016]

Regularization by Denoising [RED]

We suggest: $\rho(x)=\frac{1}{2}x^T\left(x-f(x)\right)$

… for an arbitrary denoiser f(x)

$$\rho(x)=0 \Longrightarrow \begin{array}{l} 1.\ x=0 \\ 2.\ x=f(x) \\ 3.\ \text{Orthogonality} \end{array}$$

[Romano, Elad & Milanfar, 2016]

Which f(x) to Use ?

Almost any algorithm you want may be used here, from the simplest Median (see later), all the way to the state-of-the-art CNN-like methods

We shall require f(x) to satisfy several properties as follows …

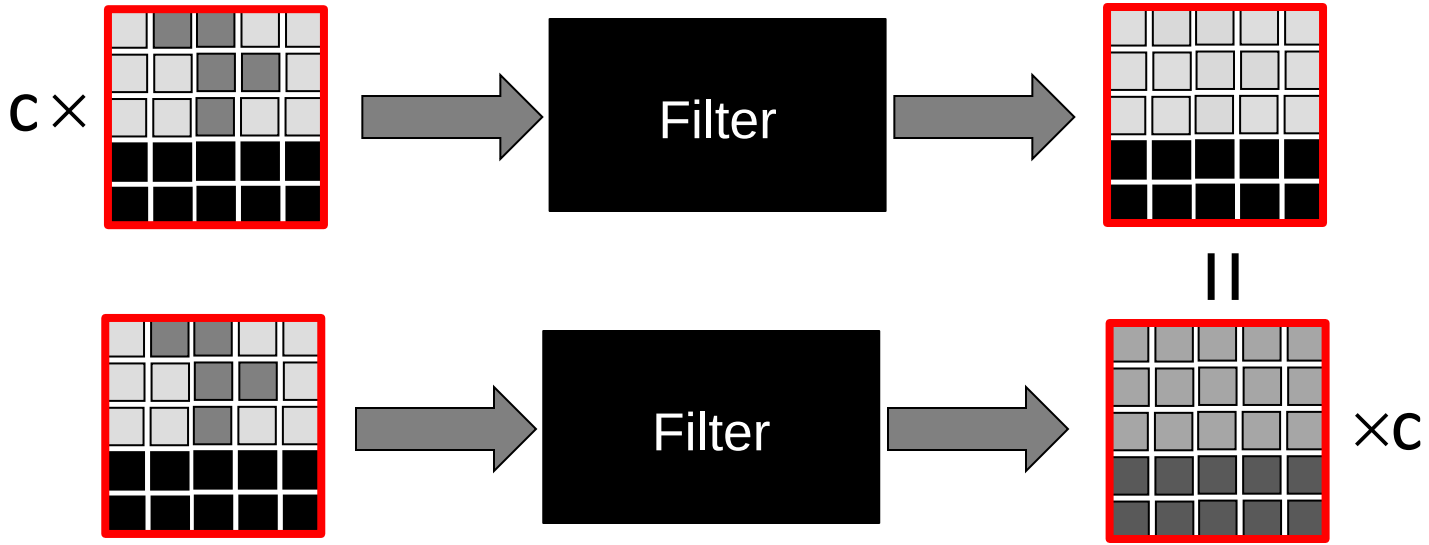
Denoising Filter Property I

$$f(x):[0,1]^n \rightarrow [0,1]^n$$

- Differentiability:
  - Some filters obey this requirement (NLM, Bilateral, Kernel Regression, TNRD)
  - Others can be ε-modified to satisfy this (Median, K-SVD, BM3D, EPLL, CNN, …)

Denoising Filter Property II

- Local Homogeneity: for $|c-1| \le \varepsilon << 1$, we have that $f(cx) = cf(x)$



Are Denoisers Homogenous ?

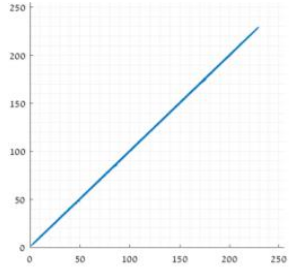
σ=5

Peppers

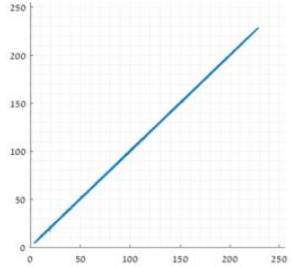
$$f((1+\varepsilon)x) = (1+\varepsilon)f(x)$$

0.01

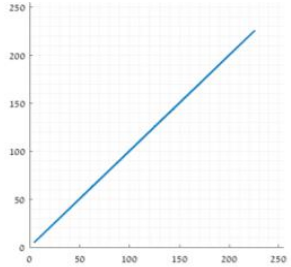
Scatter-plot of the two sides of this equation, along with the STD from the equality



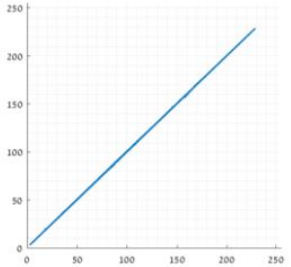
(a) K-SVD $(2.95e-4)$



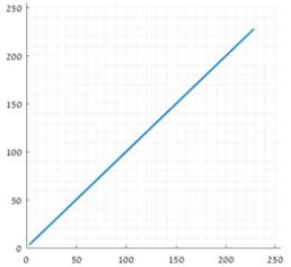
(b) BM3D $(3.38e-4)$



(c) NLM $(1.38e-4)$



(d) EPLL $(1.46e-4)$



(e) TNRD $(9.51e-5)$

Implication (1)

- Directional Derivative:

$$\nabla_x f(x)\cdot d = \lim_{\varepsilon\to 0}\frac{f(x+\varepsilon d)-f(x)}{\varepsilon}$$

$d = x$

$$\nabla_x f(x)\cdot x = \lim_{\varepsilon\to 0}\frac{f(x+\varepsilon x)-f(x)}{\varepsilon}$$

$$= \lim_{\varepsilon\to 0}\frac{(1+\varepsilon)f(x)-f(x)}{\varepsilon} = f(x)$$

Homogeneity

Looks Familiar ?

- We got the property $\nabla_x f(x) \cdot x = f(x)$

  ∇_x: n×n Matrix

- This is much more general than $f(x) = \mathbf{W}(x)x$ and applies to any denoiser satisfying the above conditions

Implication (2)

- For small $\|h\|_2$

$$f(x+h) \cong f(x) + \nabla_x f(x) \cdot h$$

Directional Derivative →

$$= \nabla_x f(x) \cdot x + \nabla_x f(x) \cdot h$$

$$= \nabla_x f(x) \cdot (x+h)$$

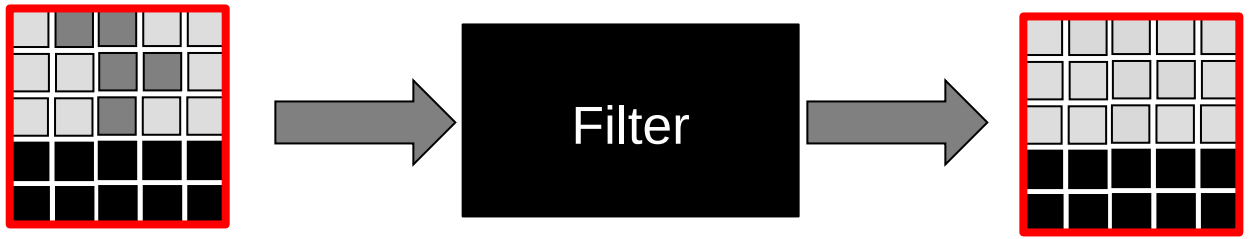
- Implication: Filter stability. Small additive perturbations of the input don't change the filter matrix

Denoising Filter Property III

- Passivity via the spectral radius:

$$r\left\{\nabla_x f(x)\right\} = \max\left|\lambda\left(\nabla_x f(x)\right)\right| \le 1$$

$$\Rightarrow \|x\| \ge r\left\{\nabla_x f(x)\right\}\cdot\|x\| \ge \left\|\nabla_x f(x)\,x\right\| = \|f(x)\|$$



Are Denoisers Passive ?

- A direct inspection of this property implies computing the Jacobian of f(x) and evaluating its spectral radius r by the Power-Method:

$$h_{k+1} = \frac{\nabla_x f(x) \cdot h_k}{\left\| \nabla_x f(x) \cdot h_k \right\|}$$

- After enough iterations, $r = \left| h_{k+1}^T \cdot h_k \right|$
- The problem is getting the Jacobian

Are Denoisers Passive ?

- The alternative is based on the Taylor expansion

$$f(x+h) \cong f(x) + \nabla_x f(x) \cdot h \Rightarrow \nabla_x f(x) \cdot h = f(x+h) - f(x)$$

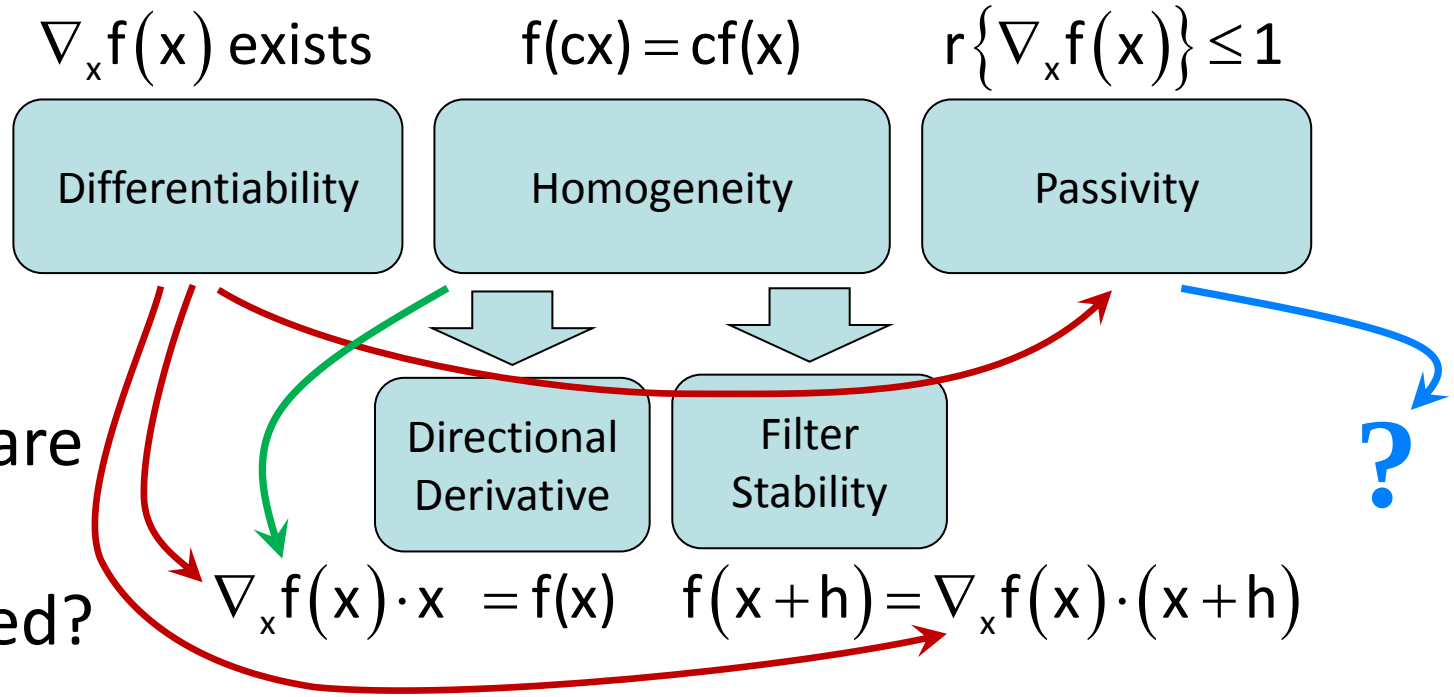
- The Power Method becomes:

$$h_{k+1} = \frac{\nabla_x f(x) \cdot h_k}{\left\| \nabla_x f(x) \cdot h_k \right\|} \Rightarrow h_{k+1} = \frac{f(x+h_k) - f(x)}{\left\| f(x+h_k) - f(x) \right\|}$$

- This leads to a value smaller than 1 for K-SVD, BM3D, NLM, EPLL, and the TNRD

Summary of Properties

- The 3 properties that f(x) should follow:



- Why are these needed?

RED: Advancing

Regularization by Denoising (RED)

$$\rho(x) = \frac{1}{2} x^T \left(x - f(x) \right)^*$$

Surprisingly, this expression is differentiable:

$$\nabla \rho(x) = x - \frac{1}{2} \nabla \left\{ x^T f(x) \right\}$$

$$= x - \frac{1}{2} \left(f(x) + \nabla f(x) x \right) = x - f(x) \text{ the residual}$$

$\nabla_x f(x) \cdot x = f(x)$ and Homogeneity

* Why not $\rho(x) = \frac{1}{2} \left\| x - f(x) \right\|_2^2$?

Regularization by Denoising (RED)

$$\rho(x) = \frac{1}{2} x^T \left(x - f(x) \right)$$

$$\nabla \rho(x) = x - f(x)$$

$$\nabla \left\{ \nabla \rho(x) \right\} = I - \nabla_x f(x) \succcurlyeq 0$$

Relying on the differentiability

$$r\left\{ \nabla_x f(x) \right\} \le 1$$

Passivity guarantees positive definiteness of the Hessian and hence convexity

RED for Inverse Problems

$$\min_{x} \ \ell(x,y) + \frac{\lambda}{2} x^{T}\left(x - f(x)\right)$$

Log-Likelihood
Data Fidelity

Regularization

Our regularization term is convex

and thus the whole expression is convex if the Log-likelihood term is convex as well

RED for Linear Inverse Problems

$$\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} x^T \left(x - f(x)\right)$$

L_2-based Data Fidelity Regularization

This energy-function is convex

Any reasonable optimization algorithm will get to the global minimum if applied correctly

Numerical Approach: Gradient Descent

$$\min_x \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2}x^T\left(x - f(x)\right)$$

$$x_{k+1} = \left(\mathbf{I} - \mu\left[\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right]\right)x_k + \mu\left(\mathbf{H}^T y + \lambda f(x_k)\right)$$

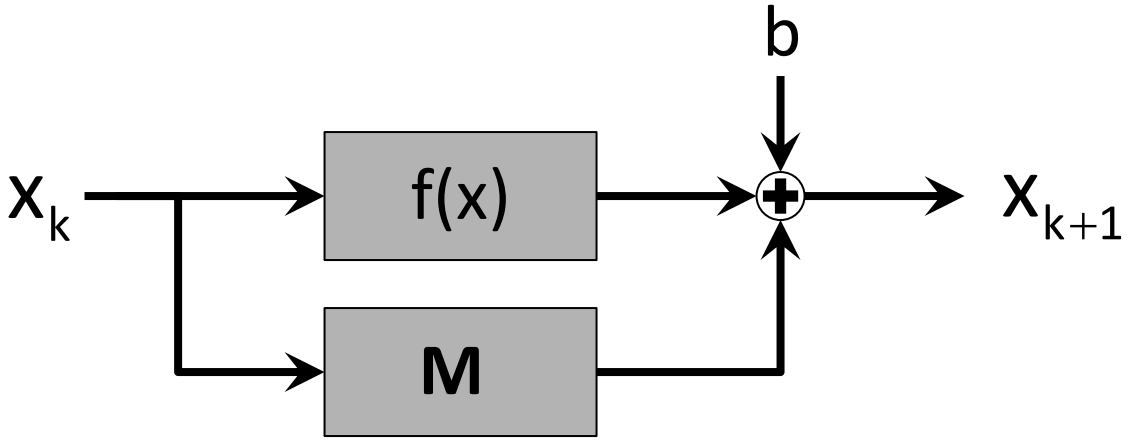
Guaranteed to converge for $0<\mu<1$, when **H** is stable

Drawback: Every iteration applies f(x) but mildly "inverting" the blur effect

Numerical Approach: Gradient Descent

$$x_{k+1} = \left(\mathbf{I} - \mu\left[\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right]\right)x_k + \mu\left(\mathbf{H}^T\mathbf{y} + \lambda f(x_k)\right)$$

$\mathbf{M}$ $\quad$ $\mathbf{b}$



Numerical Approach II: ADMM

$$\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} x^T \left(x - f(x)\right)$$

- First: apply variable splitting by setting x = v in the second term:

$$\min_{x,v} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} v^T \left(v - f(v)\right) \quad \text{s.t. } x = v$$

- Second: Use Augmented Lagrangian:

$$\min_{x,v} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} v^T \left(v - f(v)\right) + \frac{\beta}{2}\|x - v + u\|_2^2$$

Numerical Approach II: ADMM

- Third: Optimize w.r.t. x and v alternatingly
  - Update x: $\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\beta}{2}\|x - v + u\|_2^2$ Solve a simple linear system
  - Update v: $\min_{v} \frac{\lambda}{2} v^T\left(v - f(v)\right) + \frac{\beta}{2}\|x - v + u\|_2^2$

$$\Rightarrow \lambda\left(v - f(v)\right) - \beta\left(x - v + u\right) = 0$$ Solve by fixed point iteration

$$\lambda\left(v_{k+1} - f(v_k)\right) + \beta\left(x - v_{k+1} + u\right) = 0 \Rightarrow v_{k+1} = \frac{\lambda f(v_k) + \beta(x + u)}{\lambda + \beta}$$

The P^3 differs from this ADMM in this stage, offering to compute

$$v_{k+1} = f(x + u)$$

Numerical Approach II: ADMM

- Third: Optimize w.r.t. x and v alternatingly
  - Update x: $\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\beta}{2}\|x - v + u\|_2^2$ Solve a simple linear system
  - Update v: $\min_{v} \frac{\lambda}{2} v^T\left(v - f(v)\right) + \frac{\beta}{2}\|x - v + u\|_2^2$

$$\Rightarrow \lambda\left(v - f(v)\right) - \beta\left(x - v + u\right) = 0$$ Solve by fixed point iteration

$$\lambda\left(v_{k+1} - f(v_k)\right) + \beta\left(x - v_{k+1} + u\right) = 0 \Rightarrow v_{k+1} = \frac{\lambda f(v_k) + \beta(x+u)}{\lambda + \beta}$$

Guaranteed to Converge

The P^3 differs from this ADMM in this stage, offering to compute

$$v_{k+1} = f(x+u)$$

Numerical Approach: Fixed Point

$$\min_{x} \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} x^T \left(x - f(x)\right)$$

$$\mathbf{H}^T\left(\mathbf{H}x - y\right) + \lambda\left(x - f(x)\right) = 0$$

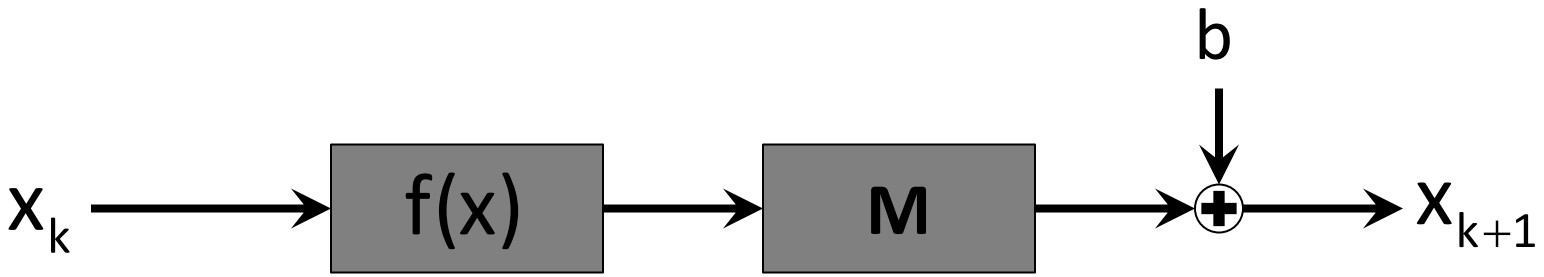
$$\mathbf{H}^T\left(\mathbf{H}x_{k+1} - y\right) + \lambda\left(x_{k+1} - f(x_k)\right) = 0$$

$$x_{k+1} = \left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}\left(\mathbf{H}^T y + \lambda f(x_k)\right)$$

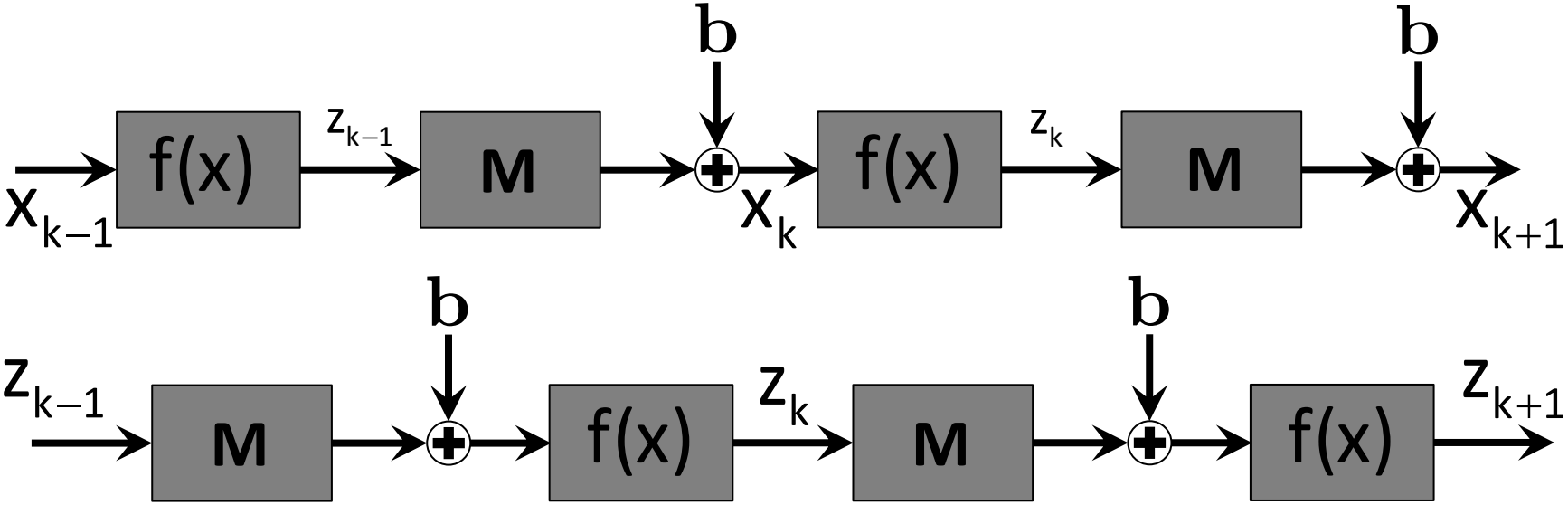
Guaranteed to converge due to the passivity of f(x)

Numerical Approach III: Fixed Point

$$x_{k+1} = \underbrace{\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}\mathbf{H}^T y}_{\mathbf{b}} + \underbrace{\lambda\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1}}_{\mathbf{M}} f\left(x_k\right)$$



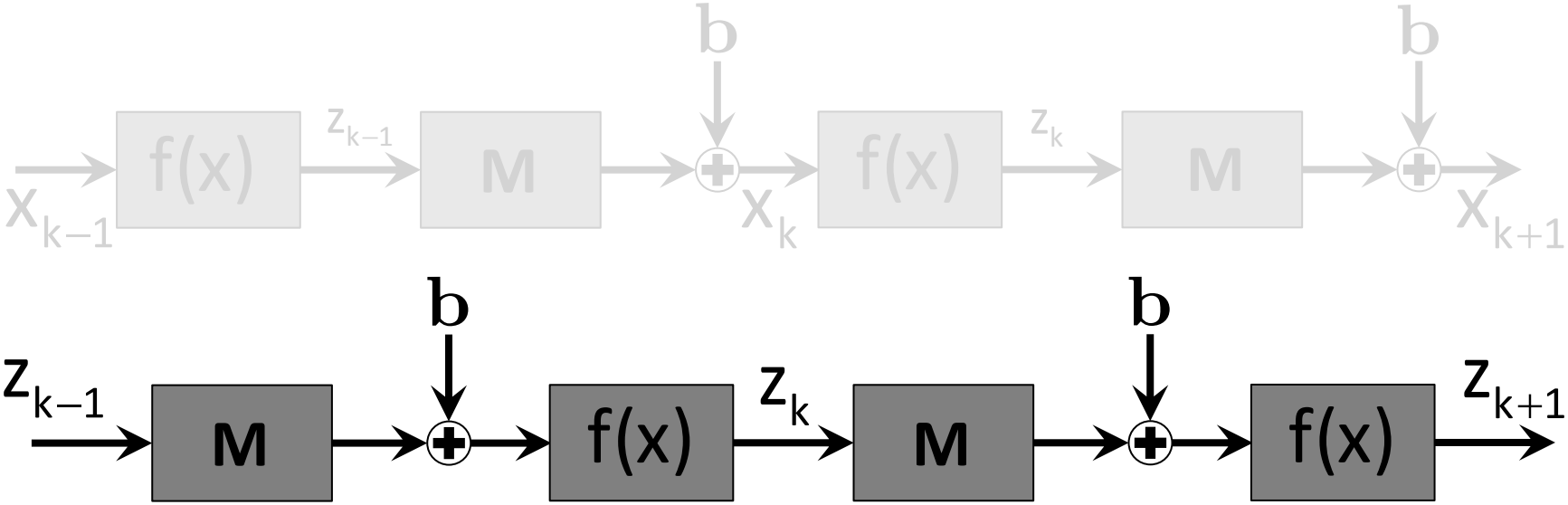
A connection to CNN



$$z_{k+1} = f\left(\mathbf{M}z_k + b\right)$$

- While CNN use a trivial and weak non-linearity f(●), we propose a very aggressive and image-aware denoiser
- Our scheme is guaranteed to minimize a clear and relevant objective function

A connection to CNN



$$z_{k+1} = f\left(\mathbf{M}z_k + b\right)$$

- While CNN use a trivial and weak non-linearity f(●), we propose a very aggressive and image-aware denoiser
- Our scheme is guaranteed to minimize a clear and relevant objective function

RED Underlying Model

RED assumes the following Prior

$$\text{Prob}(x) = C \cdot \exp\left\{-c \cdot x^T \left(x - f(x)\right)\right\}$$

Theorem: With respect to the above prior, f(x) is the MMSE estimator

Conjecture: With respect to this prior, minimizing

$$\min_x \frac{1}{2}\|\mathbf{H}x - y\|_2^2 + \frac{\lambda}{2} x^T \left(x - f(x)\right)$$

gives the optimal MMSE solution

So, again, Which f(x) to Use ?

- Almost any algorithm you want may be used here, from the simplest Median (see later), all the way to the state-of-the-art CNN-like methods

Comment: Our approach has one hidden parameter – the level of the noise (σ) the denoiser targets. We simply fix this parameter for now. But more work is required to investigate its effect

RED and P^3 Equivalence

RED vs. P^3 ?

- ADMM can be used for both P^3 and RED, leading to two similar (yet different) algorithms
- These algorithms differ in the update stage of v (assign y=x+u):
  - P^3: $v=f(y)$
  - RED: v is the solution of $\lambda(v-f(v))-\beta(y-v)=0$
- Question: Under which conditions on f(x) (and λ) would the two be the same?

RED for Denoising?

- Assume f(x)=**W**x in these equations, in order to simplify them:
  - P^3: v=f(y)=**W**y
  - RED: $\lambda(v-f(v))-\beta(y-v)=\lambda(v-\mathbf{W}v)-\beta(y-v)=0$
- Thus, for getting the same outcome we require

$$\lambda\left(v-\mathbf{W}v\right)-\beta\left(y-v\right)=0 \quad \& \quad v=\mathbf{W}y$$

$$\Longrightarrow \lambda\left(\mathbf{W}y-\mathbf{W}\mathbf{W}y\right)-\beta\left(y-\mathbf{W}y\right)=0$$

RED for Denoising?

- We got the equation

$$\lambda\left(\mathbf{W}\mathbf{y}-\mathbf{W}\mathbf{W}\mathbf{y}\right)-\beta\left(\mathbf{y}-\mathbf{W}\mathbf{y}\right)=0$$

$$\left[\beta\mathbf{I}-\left(\beta+\lambda\right)\mathbf{W}+\lambda\mathbf{W}^{2}\right]\mathbf{y}=\left(\beta\mathbf{I}-\lambda\mathbf{W}\right)\left(\mathbf{I}-\mathbf{W}\right)\mathbf{y}=0$$

- Answer: The two algorithms coincide if the eigenvalues of **W** are either 1 or β/λ, i.e. only when the denoiser is highly degenerated

Denoising via RED

RED for Denoising?

$$\min_{x} \frac{1}{2}\|x-y\|_2^2 + \frac{\lambda}{2} x^T\left(x - f(x)\right)$$

$$\Rightarrow \quad x - y + \lambda\left(x - f(x)\right) = 0$$

$$f(x) = \mathbf{W}x$$

$$x - y + \lambda\left(x - \mathbf{W}x\right) = 0 \Rightarrow x = \left(\mathbf{I} + \lambda\left(\mathbf{I} - \mathbf{W}\right)\right)^{-1} y$$

This is a “sharper” version of the original denoiser, known to modify the trading between bias and variance

[Elmoataz et al. ’08],[Bougleux et al. ‘09]

RED for Denoising?

Question: Could we claim this just gives **W**y?
(i.e. RED is equivalent to the denoiser it is built upon?)

Answer: Lets plug x=**W**y into the equation

$$x - y + \lambda\left(x - \mathbf{W}x\right) = 0 \Rightarrow \mathbf{W}y - y + \lambda\left(\mathbf{W}y - \mathbf{W}\mathbf{W}y\right) = 0$$

$$\left[\mathbf{I} - \left(1+\lambda\right)\mathbf{W} + \lambda\mathbf{W}^2\right]y = 0$$

The two filters are equivalent if the eigenvalues of **W** are either 1 or $1/\lambda$

Surprisingly: This condition is similar (equivalent for β=1) to the one leading to equivalence of RED and P[3]

RED in Practice

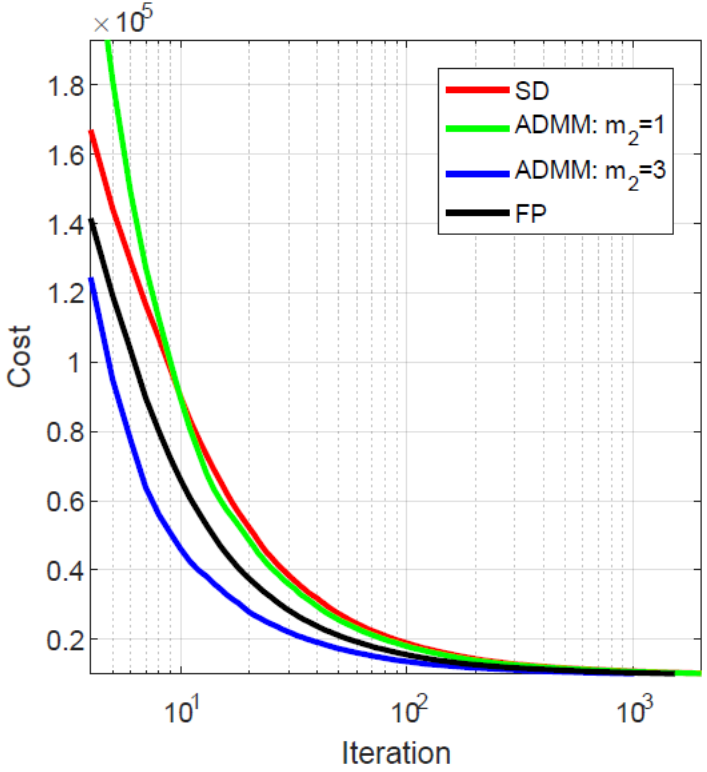
Examples: Deblurring

Uniform 9×9 kernel and WAGN with $\sigma^2=2$

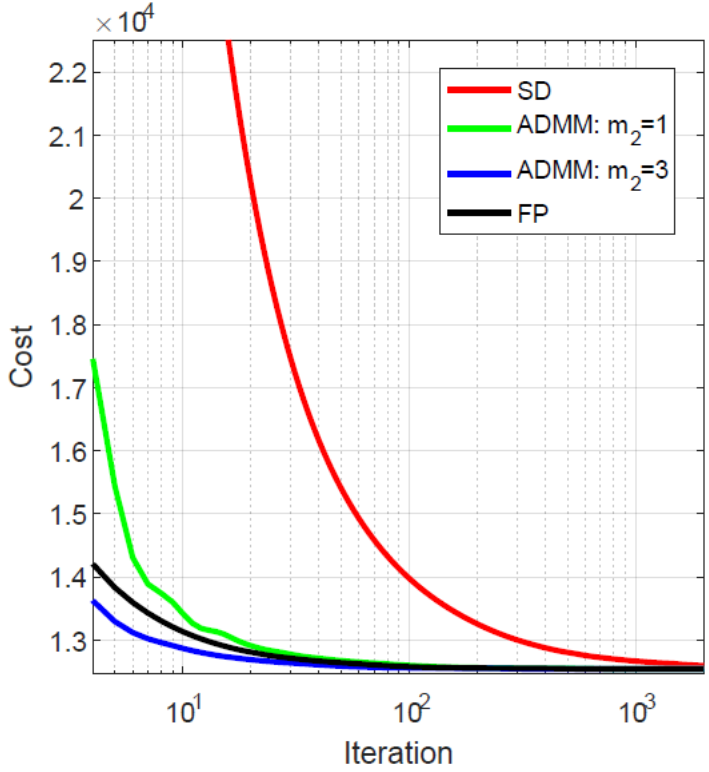
(a) Ground Truth (b) Input 21.40dB (c) RED:SD-Median filter 27.87dB

(d) NCSR 30.03dB (e) P^3-TNRD 30.36dB (f) RED: ADMM-TNRD 30.40dB

Examples: Convergence Comparison



(a) RED, Denoising Engine: Median Filter



(b) RED, Denoising Engine: TNRD

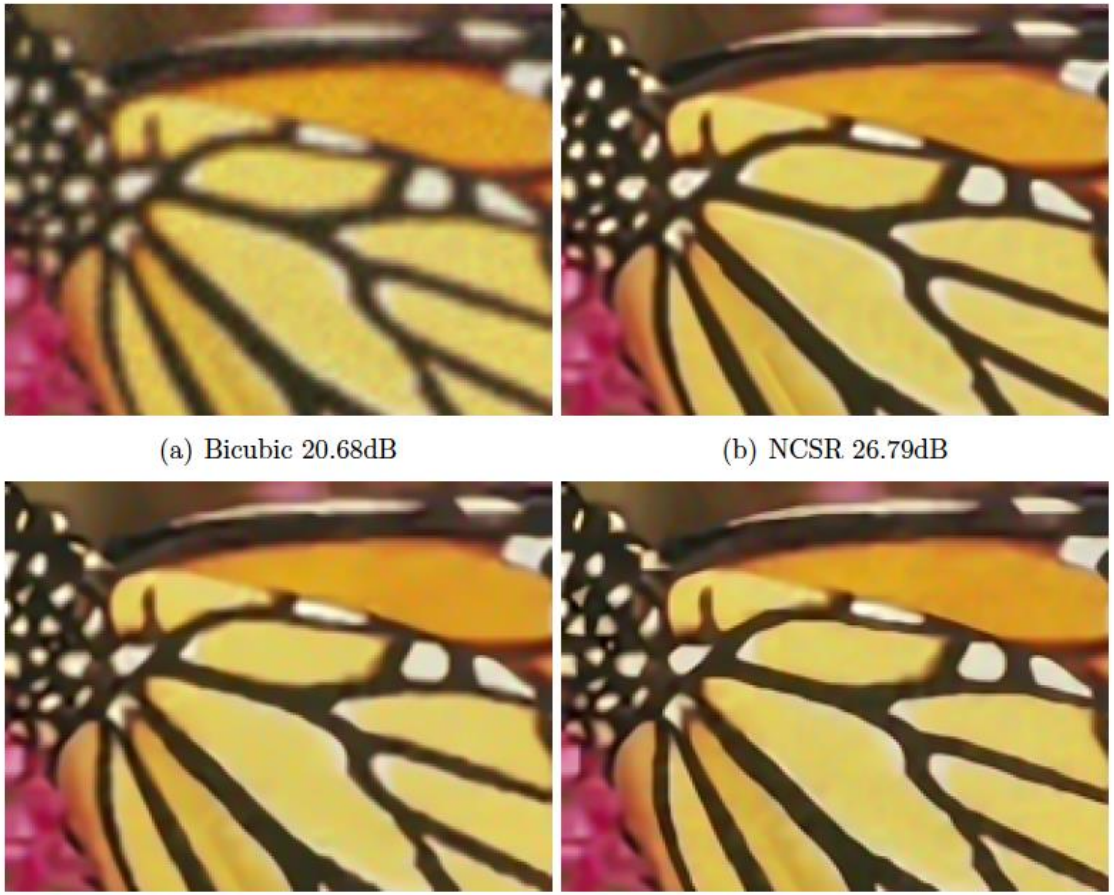
Examples: Deblurring

Uniform 9×9 kernel and WAGN with $\sigma^2=2$

(a) Ground Truth (b) Input 20.83dB (c) RED: SD-Median filter 25.87dB

(d) NCSR 28.39dB (e) P^3-TNRD 28.43dB (f) RED: FP-TNRD 28.82dB

Examples: 3x Super-Resolution



(a) Bicubic 20.68dB

(b) NCSR 26.79dB

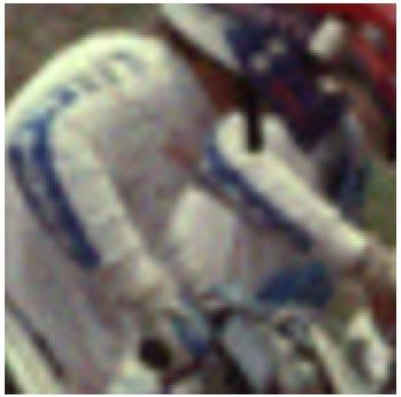
(c) P^3-TNRD 26.61dB

(d) Ours: SD-TNRD 27.39dB

Degradation:

- A Gaussian 7×7 blur with width 1.6
- A 3:1 down-sampling and
- WAGN with σ=5

Examples: 3x Super-Resolution



(a) Bicubic 20.44dB

(b) NCSR 22.97dB

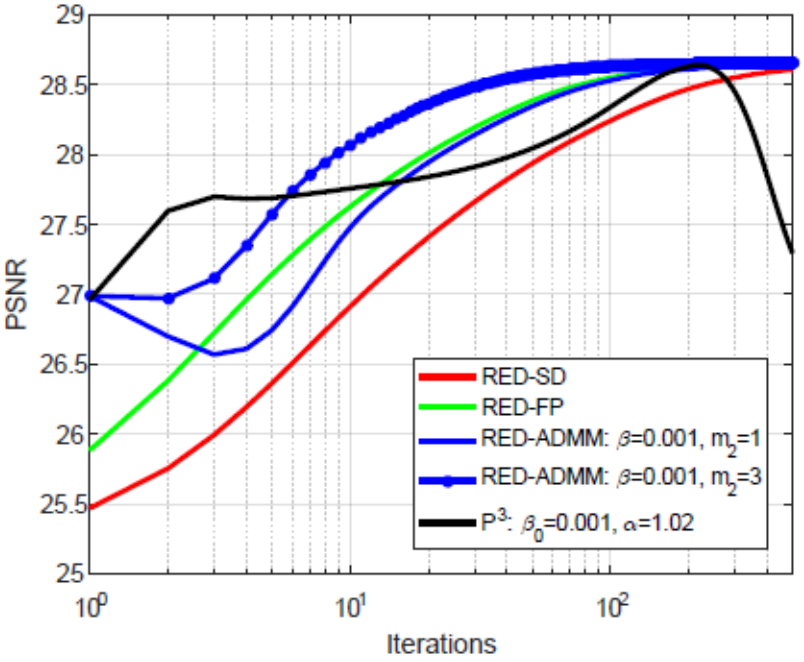
(c) P^3-TNRD 23.25dB

(d) Ours: ADMM-TNRD 23.28dB

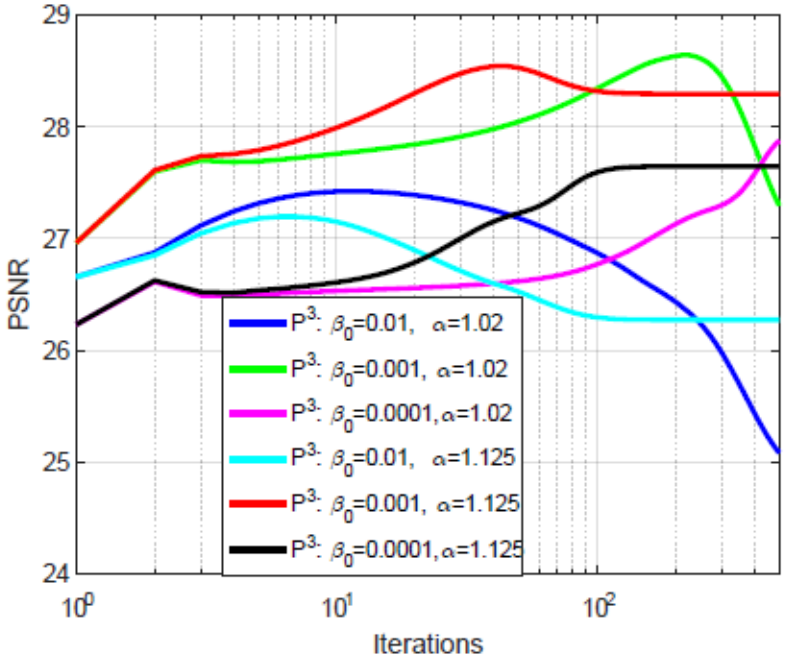
Degradation:

- A Gaussian 7×7 blur with width 1.6
- A 3:1 down-sampling and
- WAGN with σ=5

Sensitivity to Parameters

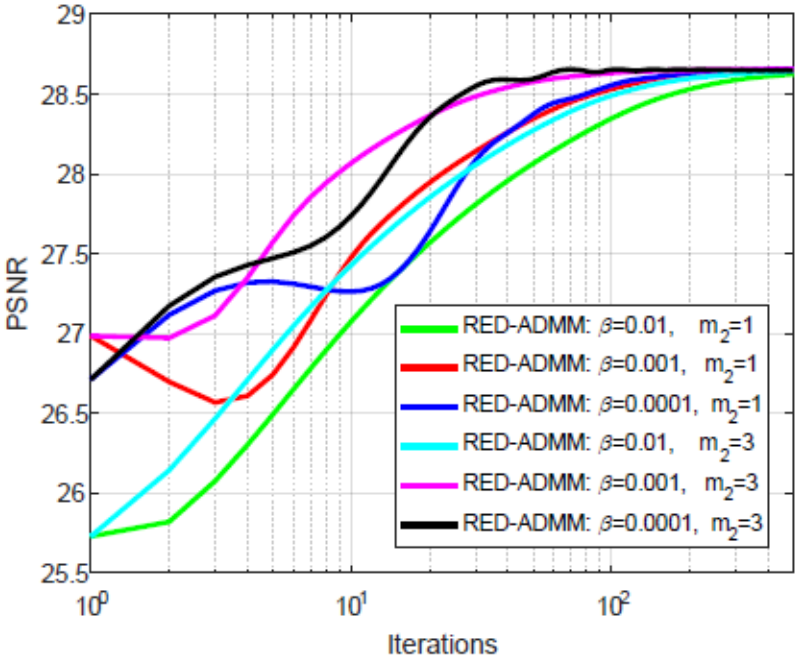


(a) Comparison between RED and PPP

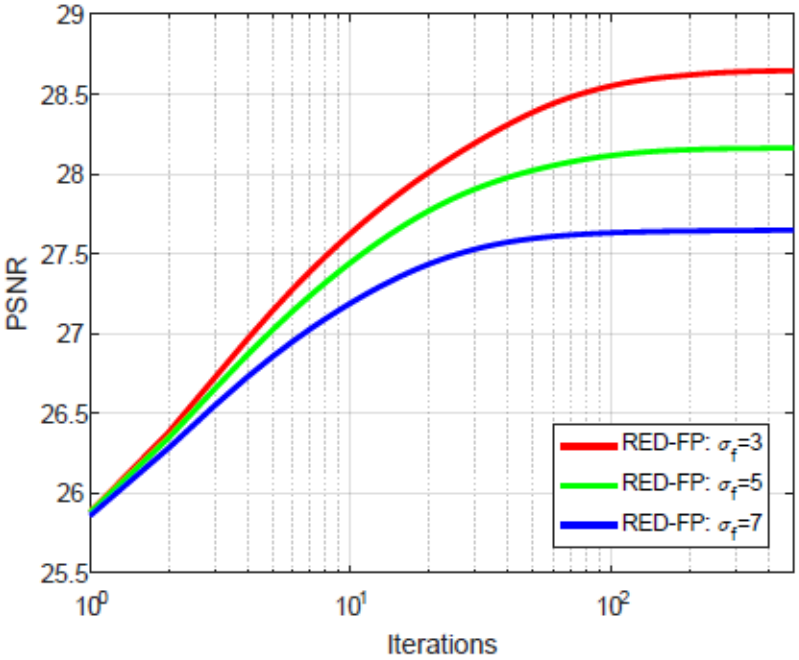


(b) P^3: Sensitivity to change in β_0 and α

Sensitivity to Parameters



(c) RED: Robustness to change in β and m_2



(d) RED: The effect of a change in σ_f

Conclusions

What have we Seen Today ?

- RED – a method to take a denoiser and use it sequentially for solving inverse problems
- Main benefits: Clear objective being minimized, Convexity, flexibility to use almost any denoiser and any optimization scheme
- One could refer to RED as a way to substantiate earlier methods (Laplacian-Regularization and the P^3) and fix them
- Challenges: Trainable version? Compression? MMSE conjecture?

Relevant Reading

1. "Regularization by Denoising", Y. Romano, M. Elad, and P. Milanfar, To appear, SIAM J. on Imaging Science.
2. "A Tour of Modern Image Filtering", P. Milanfar, IEEE Signal Processing Magazine, no. 30, pp. 106–128, Jan. 2013
3. "A General Framework for Regularized, Similarity-based Image Restoration", A. Kheradmand, and P. Milanfar, IEEE Trans on Image Processing, vol. 23, no. 12, Dec. 2014
4. "Boosting of Image Denoising Algorithms", Y. Romano, M. Elad, SIAM J. on Image Science, vol. 8, no. 2, 2015
5. "How to SAIF-ly Improve Denoising Performance", H. Talebi , X. Zhu, P. Milanfar, IEEE Trans. On Image Proc., vol. 22, no. 4, 2013
6. "Plug-and-Play Priors for Model-Based Reconstruction", S.V. Venkatakakrishnan, C.A. Bouman, B. Wohlberg, GlobalSIP, 2013
7. "BM3D-AMP: A New Image Recovery Algorithm Based on BM3D Denoising", C.A. Metzler, A. Maleki, and R.G. Baraniuk, ICIP 2015
8. "Is Denoising Dead?", P. Chatterjee, P. Milanfar, IEEE Trans. On Image Proc., vol. 19, no. 4, 2010

10. "Symmetrizing Smoothing Filters", P. Milanfar, SIAM Journal on Imaging Sciences, Vol. 6, No. 1, pp. 263–284
11. "What Regularized Auto-Encoders Learn from The Data-Generating Distribution", G. Alain, and Y. Bengio, JMLR, vol.15, 2014,
12. "Representation Learning: A Review and New Perspectives", Y. Bengio, A. Courville, and P. Vincent, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, Aug. 2013

Thank You