SRINKAGE FOR REDUNDANT REPRESENTATIONS ?

Michael Elad
The Computer Science Department
The Technion – Israel Institute of technology
Haifa 32000, Israel

SPARS'05
Signal Processing with Adaptive Sparse Structured Representations
November 16-18, 2005 – Rennes, France

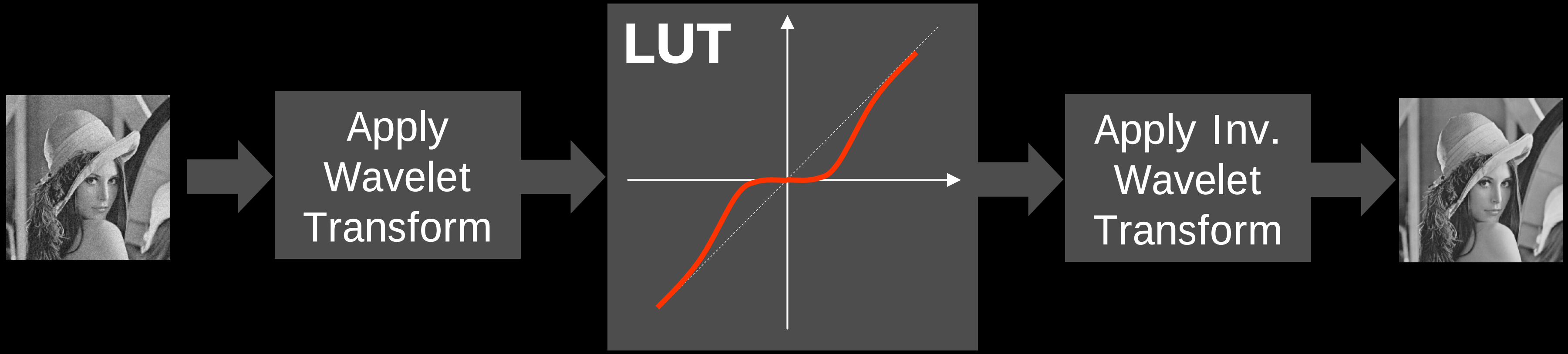
Noise Removal

Our story begins with signal/image denoising ...

- 100 years of activity – numerous algorithms.
- Considered Directions include: PDE, statistical estimators, adaptive filters, inverse problems & regularization, sparse representations, ...

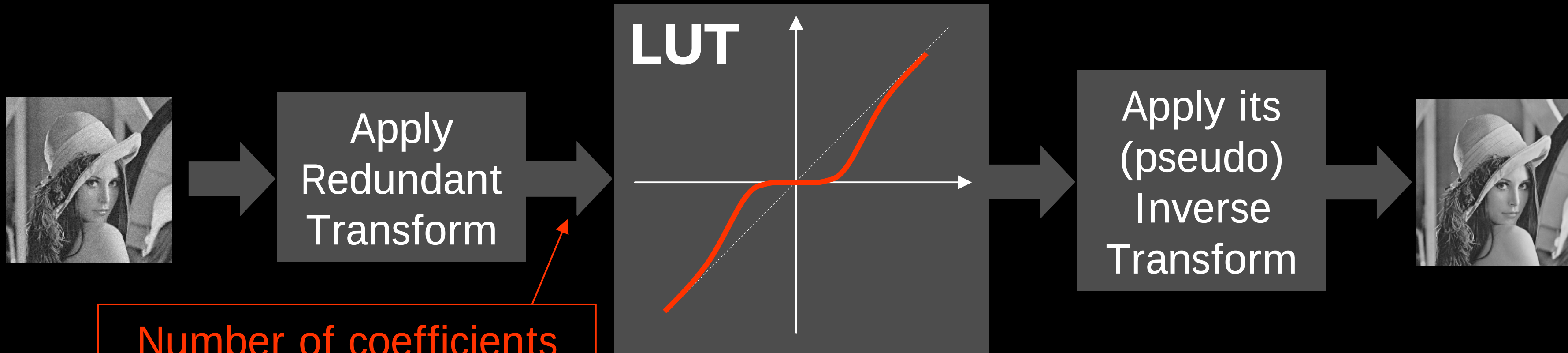
Shrinkage For Denoising

- Shrinkage is a simple yet effective denoising algorithm [Donoho & Johnstone, 1993].
- Justification 1: minimax near-optimal over the Besov (smoothness) signal space (complicated!!!!).



- Justification 2: Bayesian (MAP) optimal [Simoncelli & Adelson 1996, Moulin & Liu 1999].
- In both justifications, an additive Gaussian white noise and a unitary transform are crucial assumptions for the optimality claims.

Redundant Transforms?



Number of coefficients is (much) greater than the number of input samples (pixels)

- This scheme is still applicable, and it works fine (tested with curvelet, contourlet, undecimated wavelet, and more).
- However, it is no longer the optimal solution for the MAP criterion.

TODAY'S FOCUS:

IS SHRINKAGE STILL RELEVANT WHEN HANDLING REDUNDANT (OR NON-UNITARY) TRANSFORMS? HOW? WHY?

Agenda

1. **Bayesian Point of View – a Unitary Transform**
   Optimality of shrinkage

2. What About Redundant Representation?
   Is shrinkage is relevant? Why? How?

3. Conclusions

Thomas Bayes
1702 - 1761

The MAP Approach

Minimize the following function with respect to $\underline{x}$:

Log-Likelihood term

Prior or regularization

$$f(\underline{x}) = \frac{1}{2}\left\|\underline{x} - \underline{y}\right\|_2^2 + \lambda \cdot \Pr(\underline{x})$$

Unknown to be recovered

Given measurements

Image Prior?

During the past several decades we have made all sort of guesses about the prior Pr($\underline{x}$):

$$\Pr(\underline{x}) = \lambda\|\underline{x}\|_2^2 \qquad \Pr(\underline{x}) = \lambda\|\mathbf{L}\underline{x}\|_2^2 \qquad \Pr(\underline{x}) = \lambda\|\mathbf{L}\underline{x}\|_{\mathbf{W}}^2 \qquad \Pr(\underline{x}) = \lambda\rho\{\mathbf{L}\underline{x}\}$$


Energy


Smoothness


Adapt+ Smooth


Robust Statistics

$$\Pr(\underline{x}) = \lambda\|\nabla\underline{x}\|_1$$


Total-Variation

$$\Pr(\underline{x}) = \lambda\|\mathbf{W}\underline{x}\|_1 \qquad \Pr(\underline{x}) = \lambda\|\mathbf{T}\underline{x}\|_1$$

Today's Focus

- Mumford & Shah formulation,
- Compression algorithms as priors,
- …

(Unitary) Wavelet Sparsity

$$f(\underline{x}) = \frac{1}{2}\left\|\underline{x}-\underline{y}\right\|_2^2 + \lambda \cdot \left\|\mathbf{W}\underline{x}\right\|_1$$

Define
$$\underline{\hat{x}} = \mathbf{W}\underline{x}$$
$$\underline{x} = \mathbf{W}^H\underline{\hat{x}}$$

$$f(\underline{\hat{x}}) = \frac{1}{2}\left\|\mathbf{W}^H(\underline{\hat{x}}-\underline{\hat{y}})\right\|_2^2 + \lambda \cdot \left\|\underline{\hat{x}}\right\|_1$$

L_2 is unitarily invariant

$$f(\underline{x}) = \frac{1}{2}\left\|\underline{\hat{x}}-\underline{\hat{y}}\right\|_2^2 + \lambda \cdot \left\|\underline{\hat{x}}\right\|_1$$

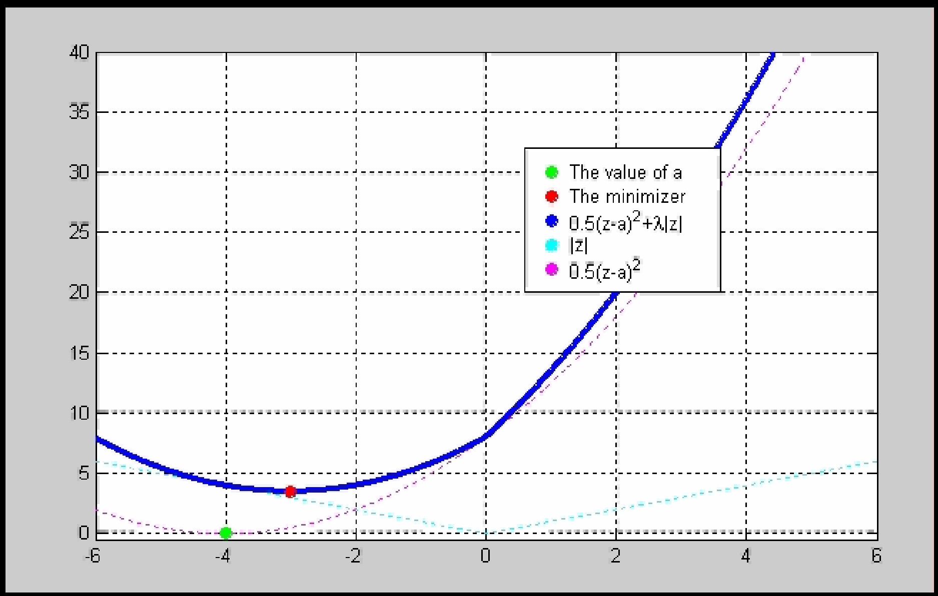
$$= \boxed{\sum_k \left[\frac{1}{2}\left(\hat{x}_k - \hat{y}_k\right)^2 + \lambda\left|\hat{x}_k\right|\right]}$$

We got a separable set of 1D optimization problems

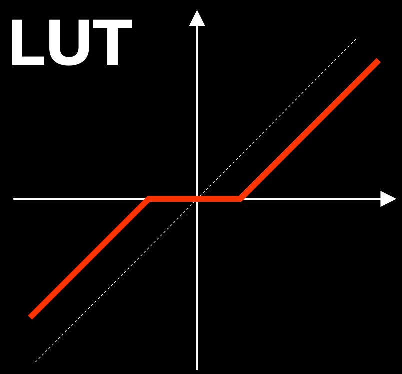
Why Shrinkage?

Want to minimize this 1-D function with respect to z

$$f(z) = \frac{1}{2}(z-a)^2 + \lambda|z|$$



LUT



$$z_{opt} = \begin{cases} a-\lambda & a \geq \lambda \\ 0 & |a| < \lambda \\ a+\lambda & a \leq -\lambda \end{cases}$$

A LUT can be built for any other robust function (replacing the |z|), including non-convex ones (e.g., L_0 norm)!!

Agenda

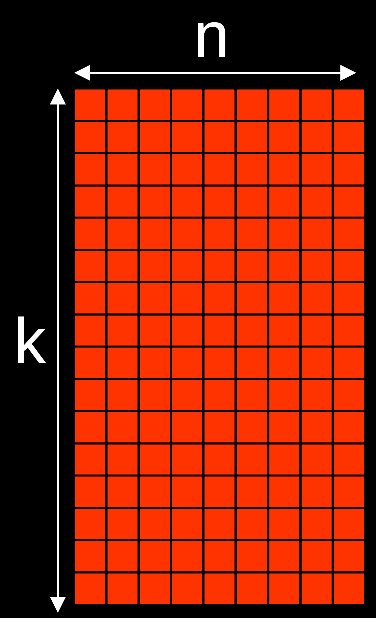
1. Bayesian Point of View – a Unitary Transform

   Optimality of shrinkage

2. **What About Redundant Representation?**

   Is shrinkage is relevant? Why? How?

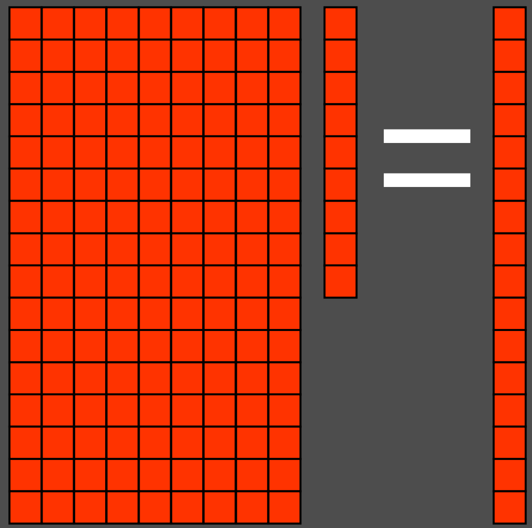
3. Conclusions



$$\mathbf{T} \in \Re^{k \times n}$$

An Overcomplete Transform

$$\mathbf{T}\underline{x} =$$



$$= \underline{\alpha}$$

$$f(\underline{x}) = \frac{1}{2}\left\|\underline{x} - \underline{y}\right\|_2^2 + \lambda \cdot \left\|\mathbf{T}\underline{x}\right\|_1$$

Redundant transforms are important because they can

(i) Lead to a shift-invariance property,

(ii) Represent images better (because of orientation/scale analysis),

(iii) Enable deeper sparsity (and thus give more structured prior).

Analysis versus Synthesis

Analysis Prior:

$$\boxed{f(\underline{x})} = \frac{1}{2}\left\|\underline{x}-\underline{y}\right\|_2^2 + \lambda \cdot \left\|\mathbf{T}\underline{x}\right\|_1$$

Define

$$\underline{\alpha} = \mathbf{T}\underline{x}$$

$$\underline{x} = \mathbf{T}^{+}\underline{\alpha}$$

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\|\mathbf{T}^{+}\underline{\alpha}-\underline{y}\right\|_2^2 + \lambda \cdot \left\|\underline{\alpha}\right\|_1$$

Synthesis Prior:

$$\boxed{\tilde{f}(\underline{\alpha})} = \frac{1}{2}\left\|\mathbf{D}\underline{\alpha}-\underline{y}\right\|_2^2 + \lambda \cdot \left\|\underline{\alpha}\right\|_1$$

Basis Pursuit

However

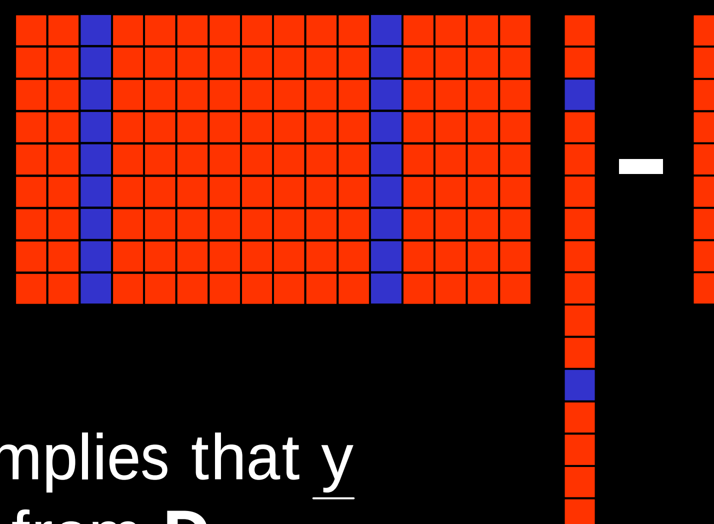
$$\mathbf{D} \cdot \underset{\underline{\alpha} = \mathbf{T}\mathbf{T}^{+}\underline{\alpha}}{\operatorname{Arg\,min}} \; \tilde{f}(\underline{\alpha}) = \underset{\underline{x}}{\operatorname{Arg\,min}} \; f(\underline{x})$$

!

Basis Pursuit As Objective

Our Objective: $\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\| \boxed{\mathbf{D}\underline{\alpha} - \underline{y}} \right\|_2^2 + \lambda \cdot \left\| \underline{\alpha} \right\|_1$

$$\mathbf{D}\underline{\alpha} - \underline{y} =$$



Getting a sparse solution implies that $\underline{y}$ is composed of few atoms from **D**

Sequential Coordinate Descent

Our objective

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\| \mathbf{D}\underline{\alpha} - \underline{y} \right\|_2^2 + \lambda \cdot \left\| \underline{\alpha} \right\|_1$$

- The unknown, $\underline{\alpha}$, has k entries.
- How about optimizing w.r.t. each of them sequentially?
- The objective per each becomes

$$\tilde{f}(z) = \frac{1}{2}\left\| z\underline{d}_j - \tilde{\underline{y}} \right\|_2^2 + \lambda |z|$$

Set j=1

↓

Fix all entries of $\underline{\alpha}$ apart from the j-th one

↓

Optimize with respect to α_j

↓

j=j+1 mod k

We Get Sequential Shrinkage

BEFORE: We had this 1-D function to minimize

$$f(z) = \frac{1}{2}(z-a)^2 + \lambda|z|$$

and the solution was

$$z_{opt} = \mathbf{S}\{a, \lambda\} = \begin{cases} a-\lambda & a \ge \lambda \\ 0 & |a| < \lambda \\ a+\lambda & a \le -\lambda \end{cases}$$

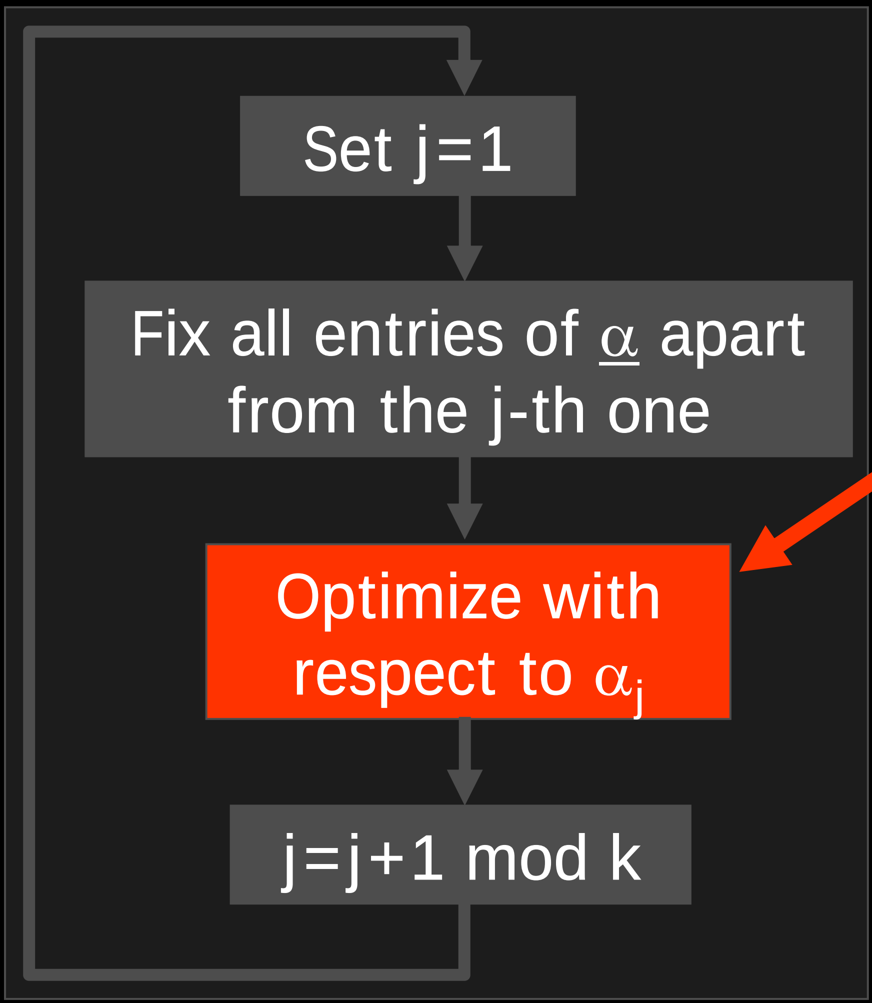
NOW: Our 1-D objective is

$$\tilde{f}(z) = \frac{1}{2}\left\| z\underline{d}_j - \underline{\tilde{y}} \right\|_2^2 + \lambda|z|$$

and the solution now is

$$z_{opt} = \mathbf{S}\left\{ \frac{\underline{d}_j^H \underline{\tilde{y}}}{\left\|\underline{d}_j\right\|_2^2}, \frac{\lambda}{\left\|\underline{d}_j\right\|_2^2} \right\}$$

Sequential? Not Good!!



$$\underline{\tilde{y}} = \underline{y} - \mathbf{D}\underline{\alpha} + \underline{d}_j\alpha_j \quad \text{and}$$

$$\alpha_j^{opt} = \mathbf{S}\left\{\frac{\underline{d}_j^H\underline{\tilde{y}}}{\left\|\underline{d}_j\right\|_2^2}, \frac{\lambda}{\left\|\underline{d}_j\right\|_2^2}\right\}$$

- ❑ This method requires drawing one column at a time from **D**.
- ❑ In most transforms this is not comfortable at all !!!

How About Parallel Shrinkage?

Our objective

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\|\mathbf{D}\underline{\alpha} - \underline{y}\right\|_2^2 + \lambda \cdot \left\|\underline{\alpha}\right\|_1$$

- Assume a current solution $\underline{\alpha}_n$.
- Using the previous method, we have k descent directions obtained by a simple shrinkage.
- How about taking all of them at once, with a proper relaxation?
- Little bit of math lead to ...

For j=1:k

Compute the descent direction per α_j : $\underline{v}_j$.

Update the solution by

$$\underline{\alpha}_{n+1} = \underline{\alpha}_n + \mu \sum_{j=1}^{k} \underline{v}_j$$

The Proposed Algorithm

Initialize $\underline{\alpha}_0 = \underline{0}$ & $k = 0.$

Compute

The synthesis error

Back-projection to the signal domain

Shrinkage operation

Update by line-search

At all stages, the dictionary is applied as a whole, either directly, or via its adjoint

$(*)\quad \mathbf{W} = \mathrm{diag}^{-1}\left\{\mathbf{D}^H\mathbf{D}\right\}$

The First Iteration – A Closer Look

Initialize $\underline{\alpha}_0 = \underline{0}$ & $k = 0.$

Compute

1. $\underline{e} = \underline{y} - \mathbf{D}\underline{\alpha}_k$
2. $\underline{e}_T = \mathbf{W}\mathbf{D}^H\underline{e}$ (*)
3. $\underline{e}_T^S = \mathbf{S}\left\{\underline{e}_T + \underline{\alpha}_k, \lambda\mathbf{W}\cdot\underline{1}\right\}$
4. $\underline{\alpha}_{k+1} = \underline{\alpha}_k + \mu\left(\underline{e}_T^S - \underline{\alpha}_k\right)$
5. $k = k + 1$

(*) $\mathbf{W} = \mathrm{diag}^{-1}\left\{\mathbf{D}^H\mathbf{D}\right\}$

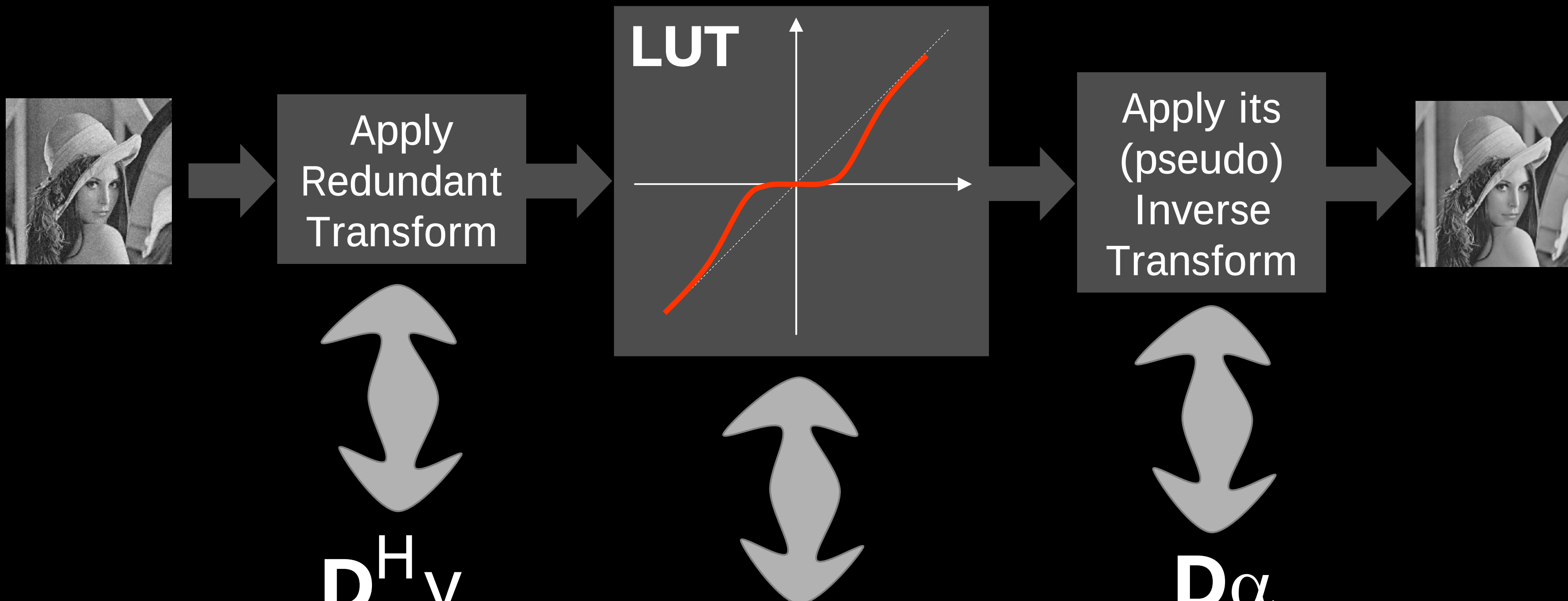
$$\underline{\alpha}_0 = \underline{0}$$

$$\underline{\alpha}_1 = \mu\mathbf{S}\left\{\mathbf{W}\mathbf{D}^H\underline{y}, \lambda\mathbf{W}\cdot\underline{1}\right\}$$

For Example:
Tight ($\mathbf{D}\mathbf{D}^H = c\cdot\mathbf{I}$)
and normalized
($\mathbf{W} = \mathbf{I}$) frame

$$\underline{\hat{x}}_1 = \mathbf{D}\cdot\mathbf{S}\left\{\mathbf{D}^H\underline{y}, \lambda\cdot\underline{1}\right\}$$

Relation to Simple Shrinkage



$$\mathbf{D}^H\underline{y}$$

$$\mathbf{D}\underline{\alpha}$$

$$\underline{\alpha} = \mathbf{S}\left\{\mathbf{D}^H\underline{y}, \lambda \cdot \underline{1}\right\}$$

A Simple Example

Minimize

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\| \mathbf{D}\underline{\alpha} - \underline{y} \right\|_2^2 + \lambda \cdot \left\| \underline{\alpha} \right\|_1$$

D: a 100×1000, union of 10 random unitary matrices,

$\underline{y}$: $\mathbf{D}\underline{\alpha}$, with $\underline{\alpha}$ having 15 non-zeros in random locations,

$\lambda=1$, $\underline{\alpha}_0=\underline{0}$,

Line-Search: Armijo

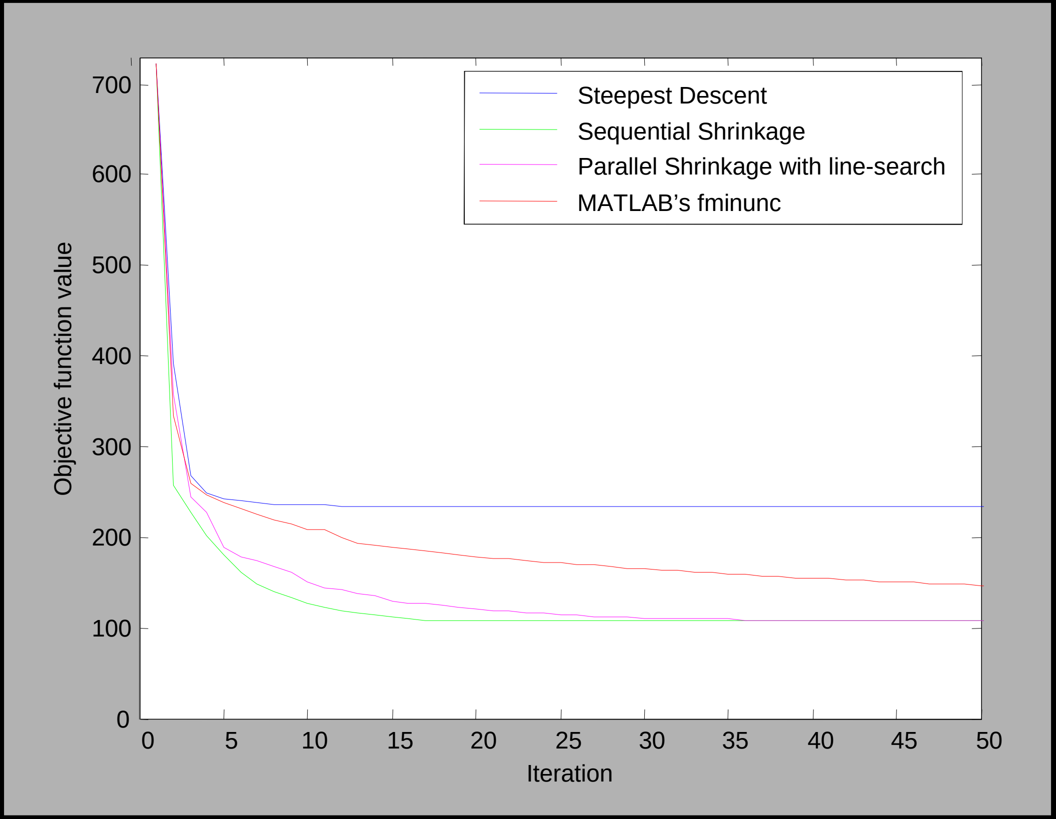


Image Denoising

Minimize

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\| \mathbf{D}\underline{\alpha} - \underline{y} \right\|_2^2 + \lambda \cdot \left\| \mathbf{W}\underline{\alpha} \right\|_1$$

- The Matrix **W** gives a variance per each coefficient, learned from the corrupted image.
- **D** is the contourlet transform (recent version).
- The length of $\underline{\alpha}$: ~1e+6.
- The Seq. Shrinkage algorithm can no longer be simulated

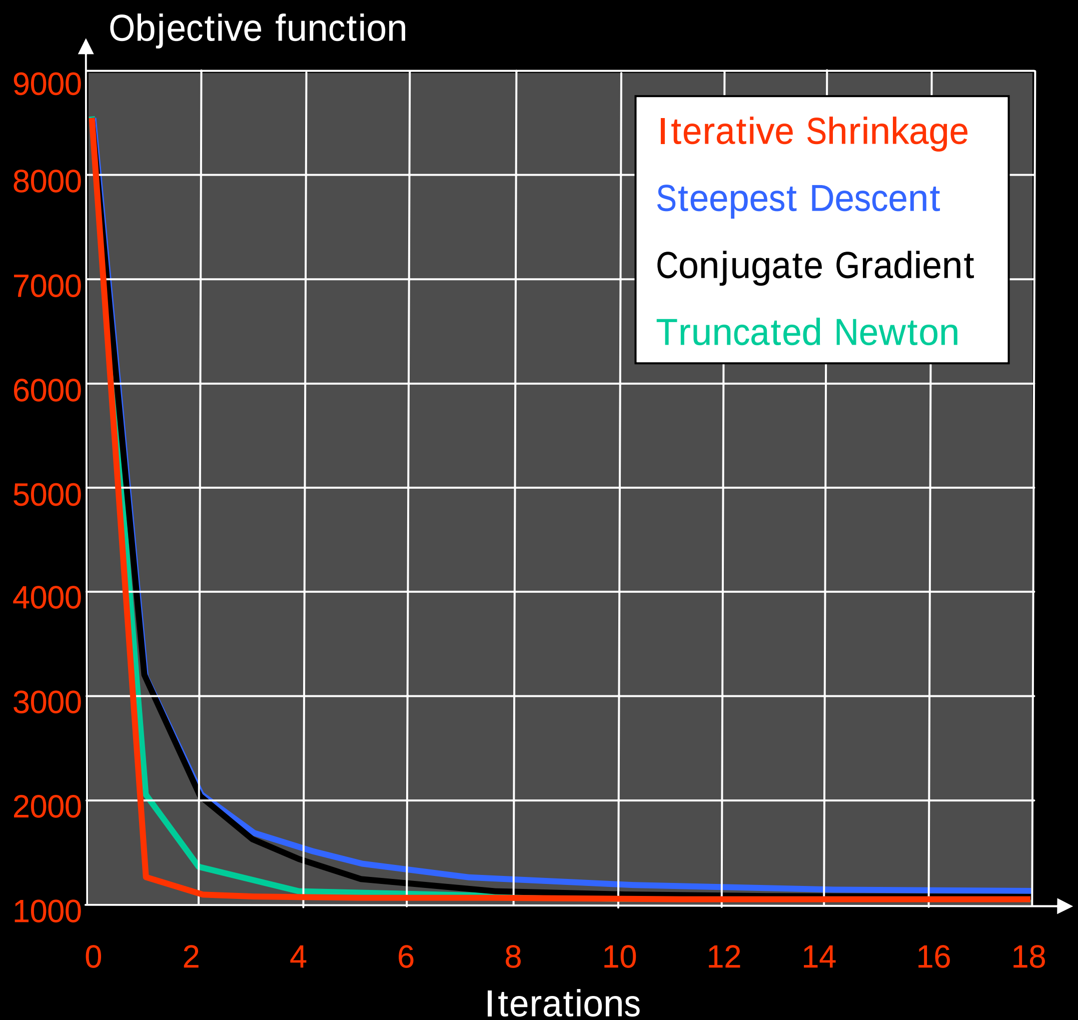


Image Denoising

Minimize

$$\tilde{f}(\underline{\alpha}) = \frac{1}{2}\left\| \mathbf{D}\underline{\alpha} - \underline{y} \right\|_2^2 + \lambda \cdot \left\| \mathbf{W}\underline{\alpha} \right\|_1$$

Evaluate $\left\| \mathbf{D}\hat{\underline{\alpha}} - \underline{x}_{\text{True}} \right\|_2^2$

Even though one iteration of our algorithm is equivalent in complexity to that of the SD, the performance is much better

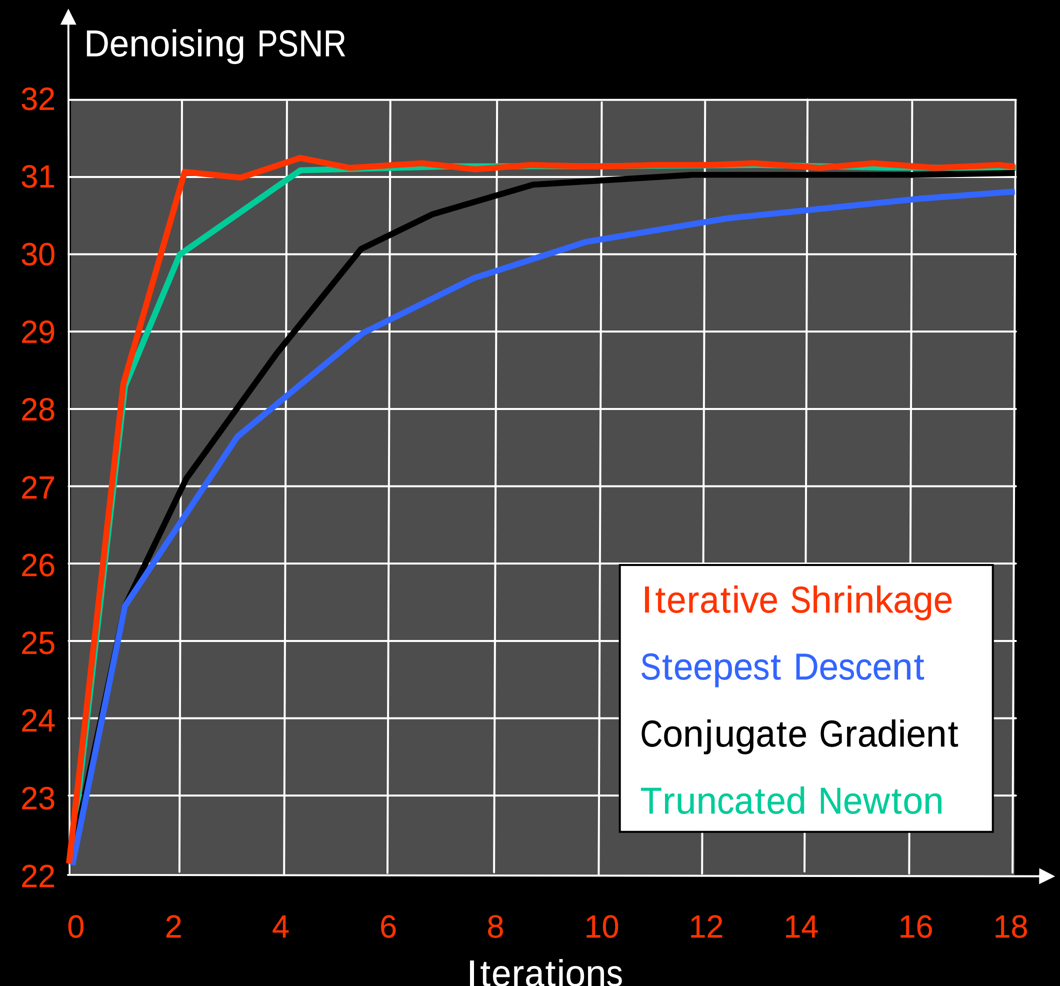
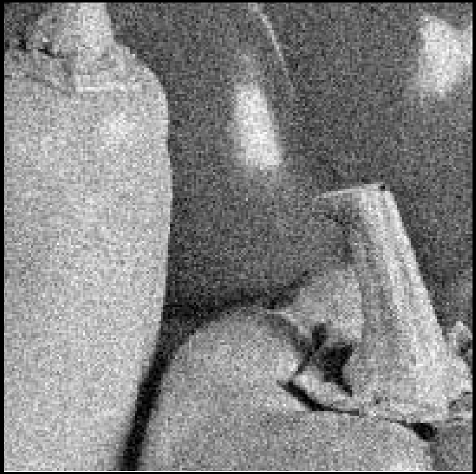


Image Denoising

Original Image

Noisy Image with $\sigma=20$



Iterated Shrinkage – First Iteration PSNR=28.30dB

Iterated Shrinkage – second iteration PSNR=31.05dB

Closely Related Work

- The "same" algorithm was derived in several other works:
  - Sparse representation over curvelet [Starck, Candes, Donoho, 2003].
  - E-M algorithm for image restoration [Figueiredo & Nowak 2003].
  - Iterated Shrinkage for problems of the form Minimize $\left\|\mathbf{K}\underline{x}-\underline{y}\right\|_2^2+\lambda\cdot\left\|\mathbf{W}\underline{x}\right\|_1$ [Daubechies, Defrise, & De-Mol, 2004].
- The work proposed here is different in several ways:
  - Motivation: Shrinkage for redundant representation, rather than general inverse problems.
  - Derivation: We used a parallel CD-algorithm, and others used the EM or a sequence of surrogate functions.
  - Algorithm: We obtain a slightly different algorithm, where the norms of the atoms are used differently, different thresholds are used, the choice of μ is different.

Agenda

1. Bayesian Point of View – a Unitary Transform
   Optimality of shrinkage
2. What About Redundant Representation?
   Is shrinkage is relevant? Why? How?
3. **Conclusions**

Conclusion

Shrinkage is an appealing signal denoising technique

When optimal?

For additive Gaussian noise and unitary transforms

What if the transform is redundant?

Option 1: apply sequential coordinate descent which leads to a sequential shrinkage algorithm

How to avoid the need to extract atoms?

Go Parallel

How?

Compute all the CD directions, and use the average

Getting what?

We obtain a very easy to implement parallel shrinkage algorithm that requires forward transform, scalar shrinkage, and inverse transform.