

Sparse & Redundant Representation Modeling of Images: Theory and Applications



Michael Elad

The Computer Science Department
The Technion
Haifa 32000, Israel



School of ICASSP Wednesday 22nd of April 2015



The research leading to these results has been received funding
from the European union's Seventh Framework Program
(FP/2007-2013) ERC grant Agreement ERC-SPARSE- 320649

This Talk Gives an **Overview** On ...

15 years of tremendous progress in the field of

Sparse and Redundant
Representations

Theory

Numerical
Problems

Applications



Agenda

**Part I – Denoising
by Sparse &
Redundant
Representations**

**Part II – Theoretical &
Numerical Foundations**

**Part III – Dictionary Learning
& The K-SVD Algorithm**

**Part V –
Summary &
Conclusions**

**Part IV – Back to Denoising ... and Beyond –
handling stills and video denoising & inpainting,
demosaicing, super-res., and compression**

**Today we will
show that**

- Sparsity and Redundancy are valuable and well-founded tools for modeling data.
- When used in image processing, they lead to state-of-the-art results.



Part I

Denoising by Sparse & Redundant Representations



Noise Removal?

Our story begins with image denoising ...



- **Important:** (i) Practical application; (ii) A convenient platform (the simplest) for testing basic ideas in image processing; (iii) Given a good denoising algorithm, one could solve many other problems.
- **Many Considered Directions:** Partial differential equations, Statistical estimators, Adaptive filters, Inverse problems & regularization, Wavelets, Example-based techniques, **Sparse representations**, ...



Denoising By Energy Minimization

Many of the proposed image denoising algorithms are related to the minimization of an energy function of the form

$$f(\underline{x}) = \frac{1}{2} \|\underline{x} - \underline{y}\|_2^2$$

Relation to
measurements

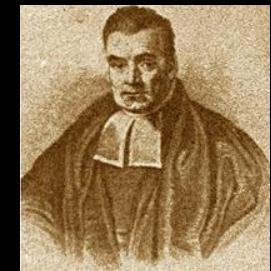
$$+ G(\underline{x})$$

Prior or regularization

\underline{y} : Given measurements

\underline{x} : Unknown to be recovered

- This is in-fact a Bayesian point of view, adopting the Maximum-A-posteriori Probability (MAP) estimation.
- Clearly, the wisdom in such an approach is within the choice of the prior – **modeling the images** of interest.



Thomas Bayes
1702 - 1761



The Evolution of $G(\underline{x})$

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

$$G(\underline{x}) = \lambda \|\underline{x}\|_2^2$$



Energy

$$G(\underline{x}) = \lambda \|\mathbf{L}\underline{x}\|_2^2$$



Smoothness

$$G(\underline{x}) = \lambda \|\mathbf{L}\underline{x}\|_{\mathbf{W}}^2$$



**Adapt+
Smooth**

$$G(\underline{x}) = \lambda \rho \{\mathbf{L}\underline{x}\}$$



**Robust
Statistics**

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



**Total-
Variation**

$$G(\underline{x}) = \lambda \|\mathbf{W}\underline{x}\|_1$$



**Wavelet
Sparsity**

$$G(\underline{x}) = \lambda \|\underline{\alpha}\|^0$$

for $\underline{x} = \mathbf{D}\underline{\alpha}$

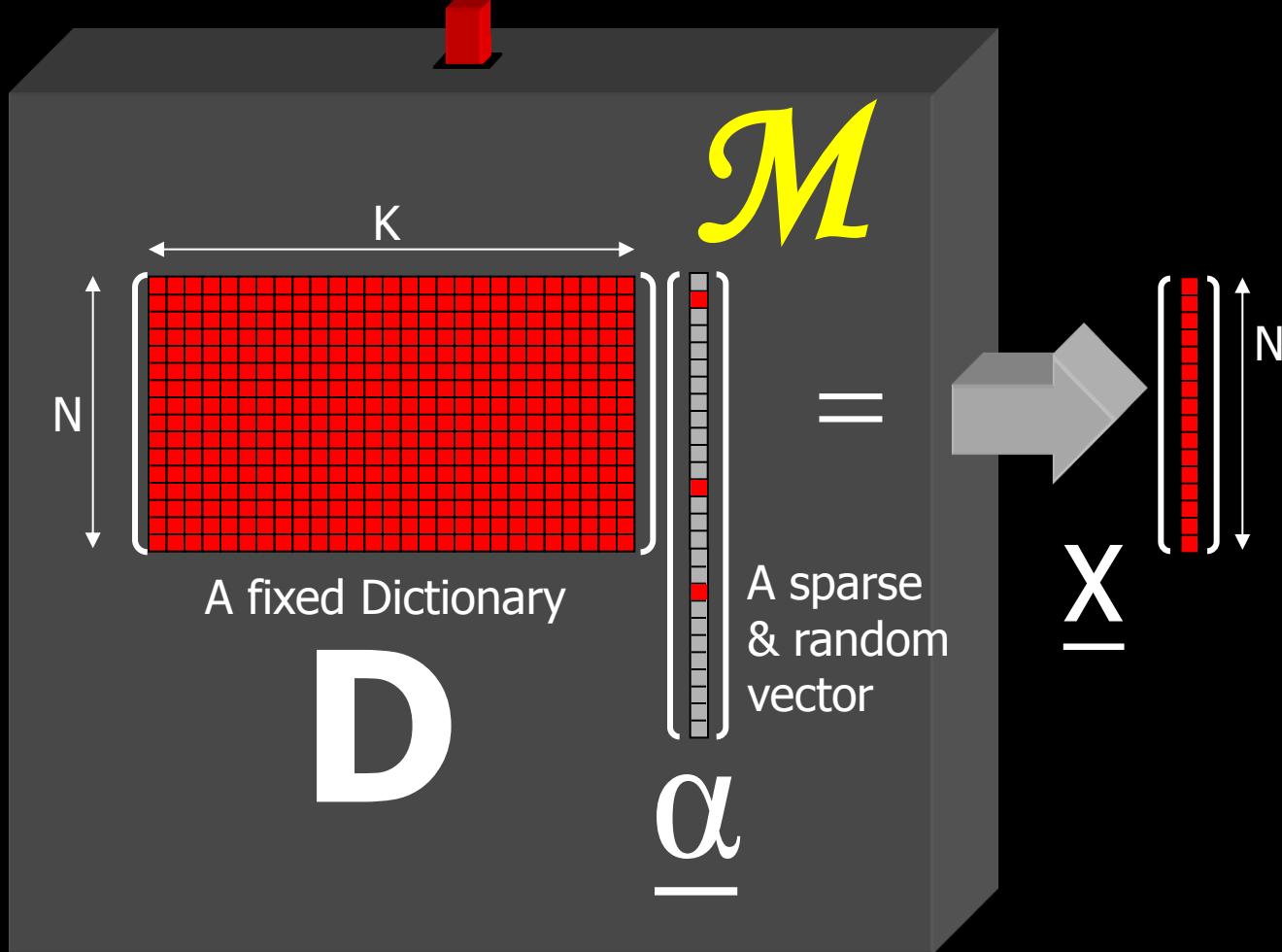


**Sparse &
Redundant**

- Hidden Markov Models,
- Compression algorithms as priors,
- ...



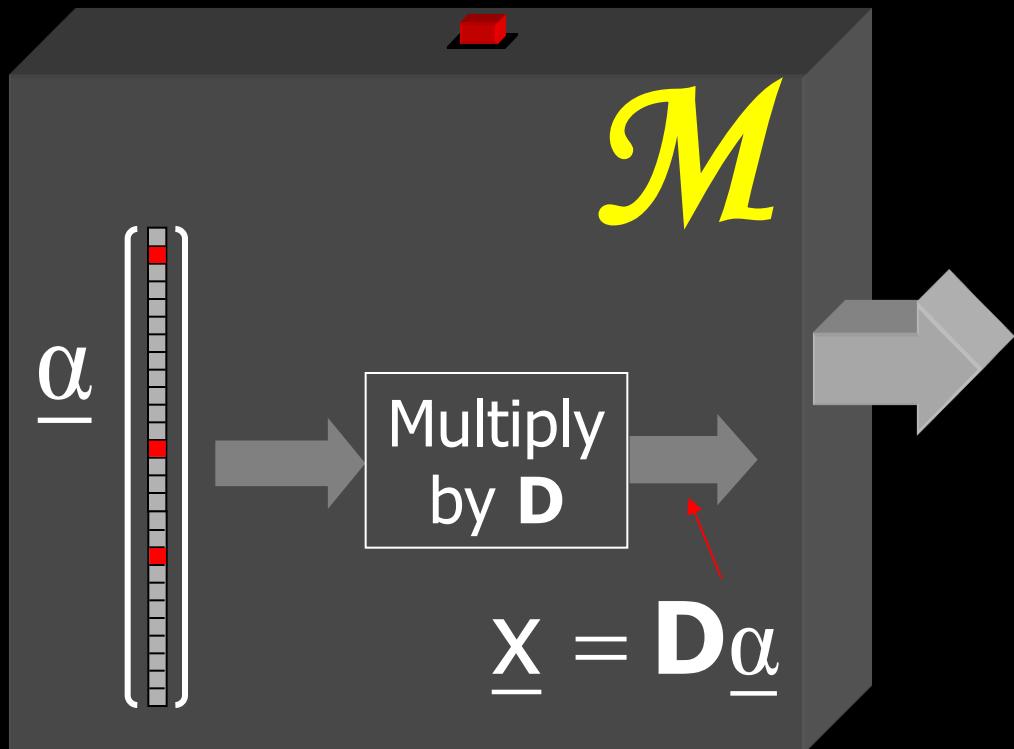
Sparse Modeling of Signals



- Every column in D (**dictionary**) is a prototype signal (**atom**).
- The vector $\underline{\alpha}$ is generated randomly with few (say L) non-zeros at random locations and with random values.
- We shall refer to this model as *Sparseland*



Sparseland Signals are Special



Interesting Model:

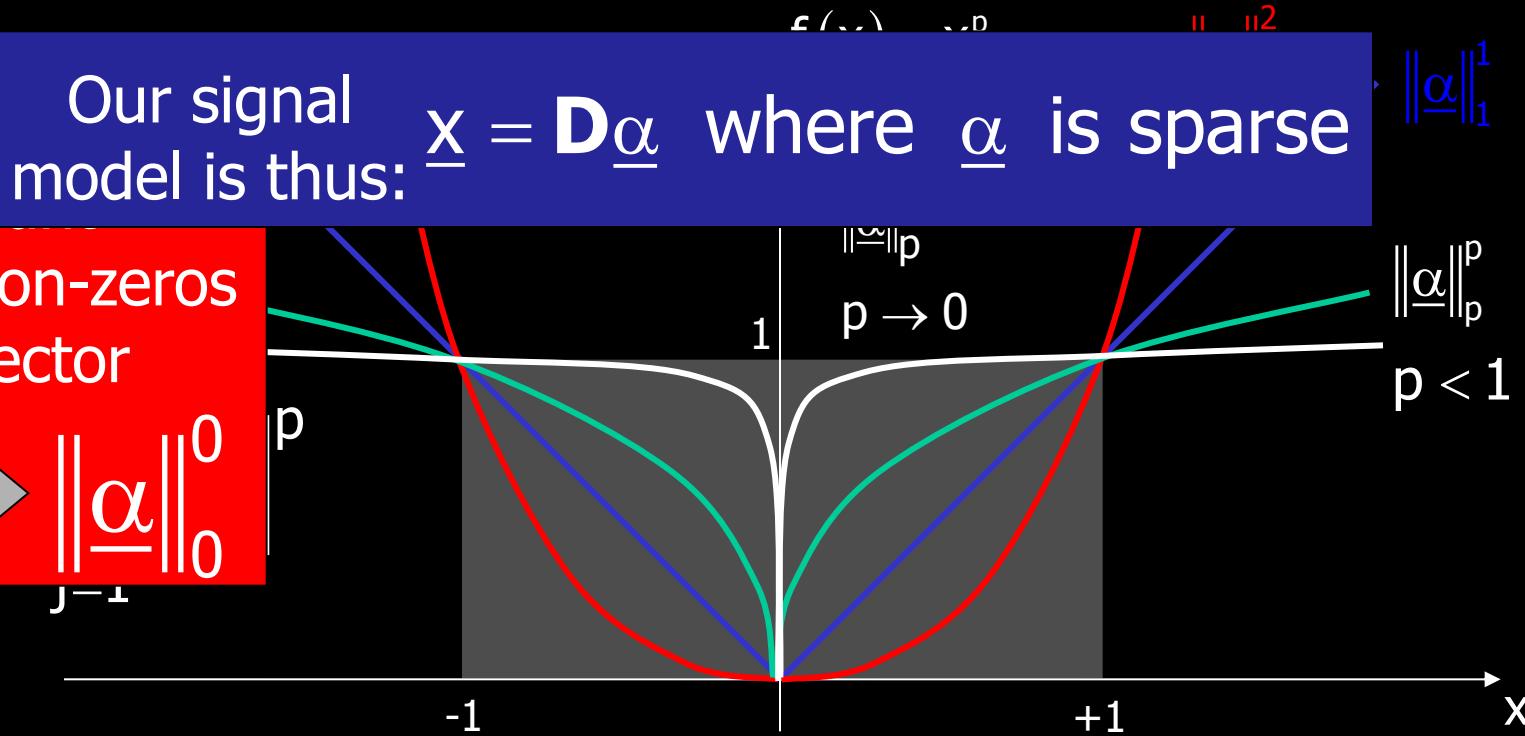
- **Simple:** Every generated signal is built as a linear combination of few atoms from our **dictionary \mathbf{D}**
- **Rich:** A general model: the obtained signals are a **union of many low-dimensional Gaussians**.
- **Familiar:** We have been using this model in other context for a while now (wavelet, JPEG, ...).



Sparse & Redundant Rep. Modeling?

As $p \rightarrow \infty$,
get a convex hull
of the non-zeros
in the vector

$$\|\underline{\alpha}\|_0^0$$



$$\underline{x} = \mathbf{D}\underline{\alpha} \text{ where } \|\underline{\alpha}\|_0^0 \leq L$$



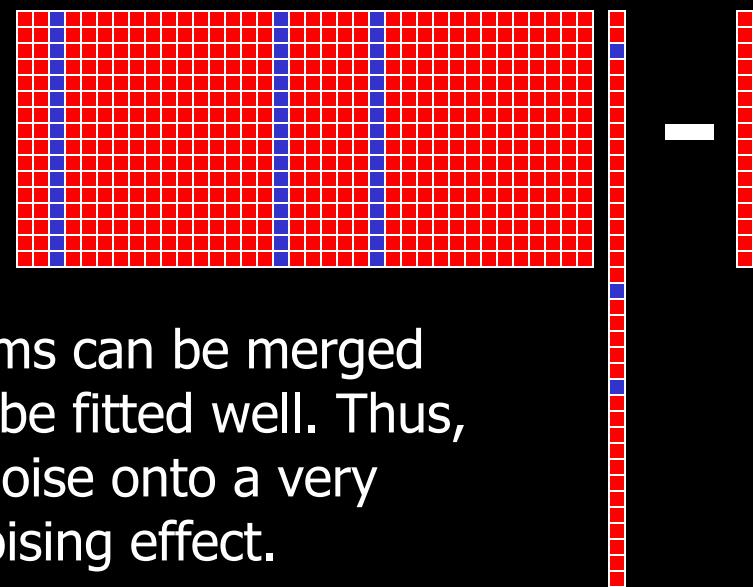
Back to Our MAP Energy Function

- We L_0 norm is effectively counting the number of non-zeros in α .

$$\frac{1}{2} \|\underline{x} - \underline{y}\|_2^2$$

- The vector $\underline{\alpha}$ is the representation (**sparse/redundant**) of the desired signal x .

Dα-y =



- The core idea: while few (L out of K) atoms can be merged to form the true signal, the noise cannot be fitted well. Thus, we obtain an effective projection of the noise onto a very low-dimensional space, thus getting denoising effect.

Wait! There are Some Issues

- **Numerical Problems:** How should we solve or approximate the solution of the problem

$$\min_{\underline{\alpha}} \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \text{ s.t. } \|\underline{\alpha}\|_0^0 \leq L \quad \text{or} \quad \min_{\underline{\alpha}} \|\underline{\alpha}\|_0^0 \text{ s.t. } \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \leq \varepsilon^2$$

$$\text{or } \min_{\underline{\alpha}} \lambda \|\underline{\alpha}\|_0^0 + \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \quad ?$$

- **Theoretical Problems:** Is there a unique sparse representation? If we are to approximate the solution somehow, how close will we get?
- **Practical Problems:** What dictionary \mathbf{D} should we use, such that all this leads to effective denoising? Will all this work in applications?



To Summarize So Far ...

Image denoising
(and many other
problems in image
processing) requires
a model for the
desired image

What do
we do?

We proposed a
model for
signals/images
based on sparse
and redundant
representations

There are some issues:
1. Theoretical
2. How to approximate?
3. What about **D**?

Great!
No?



Part II

Theoretical & Numerical Foundations



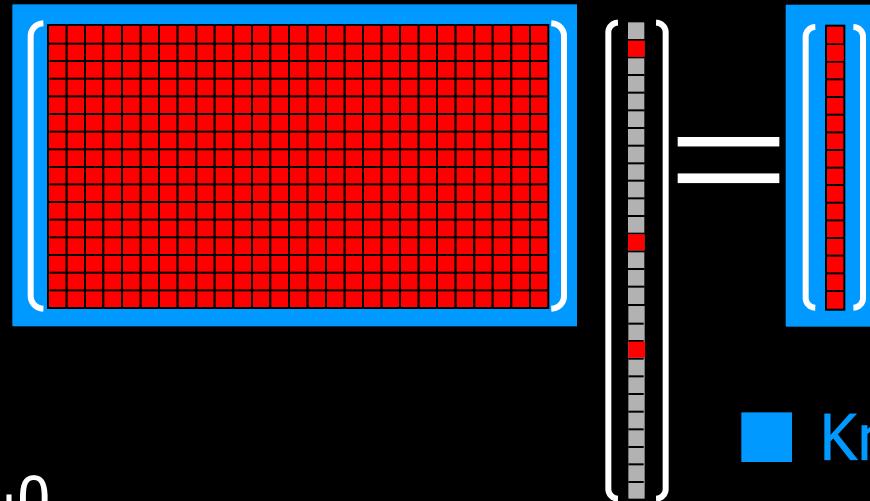
Lets Start with the Noiseless Problem

Suppose we build a signal by the relation

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

We aim to find the signal's representation:

$$\hat{\underline{\alpha}} = \underset{\underline{\alpha}}{\text{ArgMin}} \|\underline{\alpha}\|_0^0 \quad \text{s.t.} \quad \underline{x} = \mathbf{D}\underline{\alpha}$$



■ Known

Uniqueness

Why should we necessarily get $\hat{\underline{\alpha}} = \underline{\alpha}$?

It might happen that eventually $\|\hat{\underline{\alpha}}\|_0^0 < \|\underline{\alpha}\|_0^0$.



Matrix “Spark”

Definition: Given a matrix \mathbf{D} , $\sigma = \text{Spark}\{\mathbf{D}\}$ is the smallest number of columns that are linearly dependent.

*

Donoho & E. ('02)

Example:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Rank = 4

Spark = 3

* In tensor decomposition,
Kruskal defined something
similar already in 1989.



Uniqueness Rule

Suppose this problem has been solved somehow

$$\hat{\underline{\alpha}} = \underset{\underline{\alpha}}{\text{ArgMin}} \|\underline{\alpha}\|_0^0 \quad \text{s.t. } \underline{x} = \mathbf{D}\underline{\alpha}$$

Uniqueness

Donoho & E. ('02)

If we found a representation that satisfy

$$\|\hat{\underline{\alpha}}\|_0 < \frac{\sigma}{2}$$

Then necessarily it is unique (the sparsest).

This result implies that if \mathcal{M} generates signals using “sparse enough” $\underline{\alpha}$, the solution of the above will find it exactly.

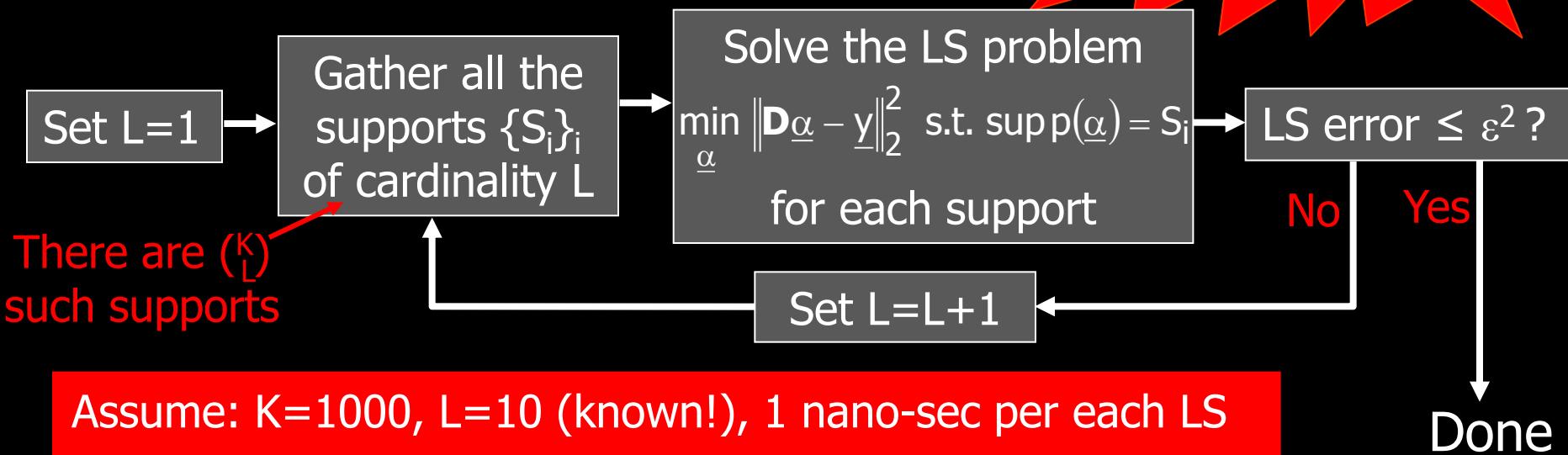


Our Goal

$$\min_{\underline{\alpha}} \|\underline{\alpha}\|_0^0 \text{ s.t. } \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \leq \varepsilon^2$$

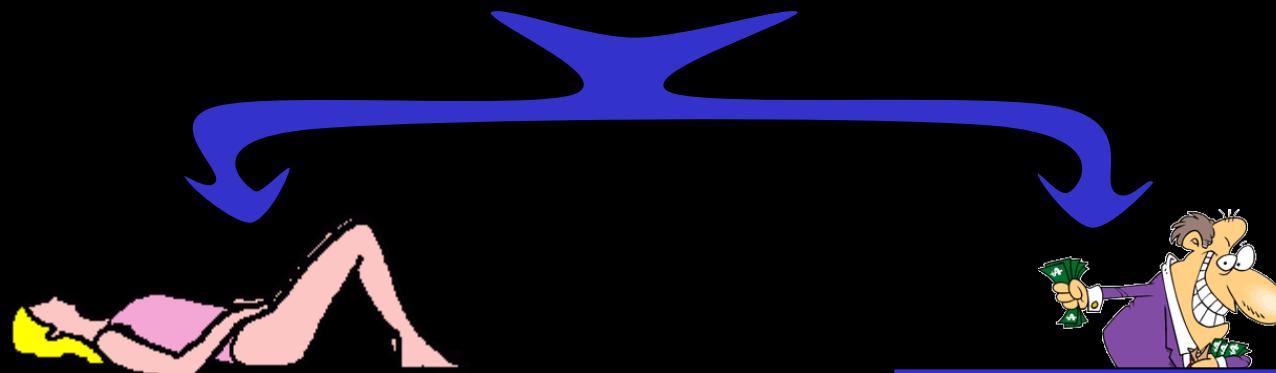
This is a combinatorial problem, proven to be NP-Hard!

Here is a recipe for solving this problem:



Lets Approximate

$$\min_{\underline{\alpha}} \|\underline{\alpha}\|_0^0 \text{ s.t. } \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \leq \varepsilon^2$$



Relaxation methods

Smooth the L_0 and use
continuous optimization
techniques

Greedy methods

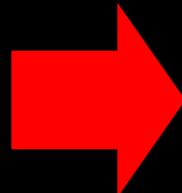
Build the solution
one non-zero
element at a time



Relaxation – The Basis Pursuit (BP)

Instead of solving

$$\underset{\underline{\alpha}}{\text{Min}} \quad \|\underline{\alpha}\|_0^0 \quad \text{s.t.} \quad \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2 \leq \varepsilon$$



Solve Instead

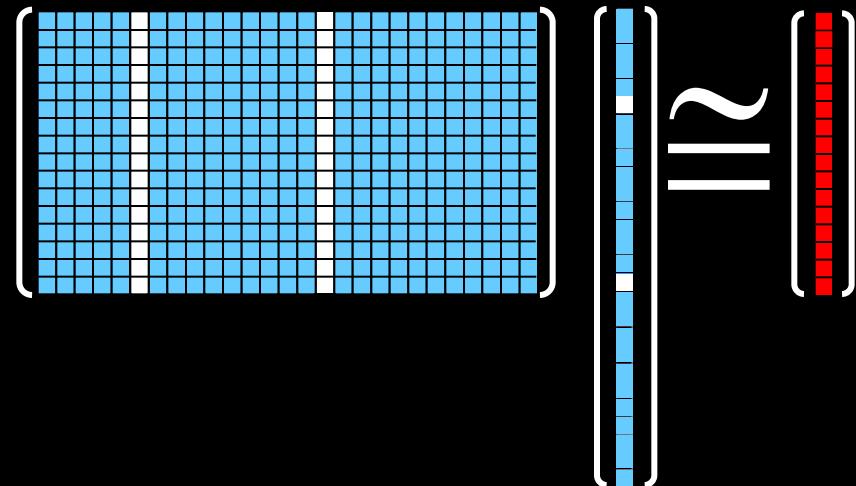
$$\underset{\underline{\alpha}}{\text{Min}} \quad \|\underline{\alpha}\|_1 \quad \text{s.t.} \quad \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2 \leq \varepsilon$$

- This is known as the Basis-Pursuit (BP) [Chen, Donoho & Saunders ('95)].
- The newly defined problem is convex (quad. programming).
- Very efficient solvers can be deployed:
 - Interior point methods [Chen, Donoho, & Saunders ('95)] [Kim, Koh, Lustig, Boyd, & D. Gorinevsky ('07)].
 - Sequential shrinkage for union of ortho-bases [Bruce et.al. ('98)].
 - Iterative shrinkage [Figueiredo & Nowak ('03)] [Daubechies, Defrise, & De-Mole ('04)] [E. ('05)] [E., Matalon, & Zibulevsky ('06)] [Beck & Teboulle ('09)] ...



Go Greedy: Matching Pursuit (MP)

- The MP is one of the greedy algorithms that finds one atom at a time [Mallat & Zhang ('93)].
- Step 1: find the one atom that **best matches** the signal.
- Next steps: given the previously found atoms, find the next **one** to **best fit** the residual.
- The algorithm stops when the error $\|\mathbf{D}\underline{\alpha} - \underline{y}\|_2$ is below the destination threshold.
- The Orthogonal MP (OMP) is an improved version that re-evaluates the coefficients by Least-Squares after each round.



Pursuit Algorithms

$$\min_{\underline{\alpha}} \|\underline{\alpha}\|_0^0 \text{ s.t. } \|\mathbf{D}\underline{\alpha} - \mathbf{y}\|_2^2 \leq \varepsilon^2$$

There are various algorithms designed for approximating the solution of

- Greedy Algorithms (OMP), L₁-Pursuit [1995]
- Relaxation & numerical methods [1995]
- Hybrid Algorithms: Thresholding [2007-today].
- ...

Why should they work?

unit
ing
ector
lard-



The Mutual Coherence

- Compute

$$\begin{bmatrix} \mathbf{D}^T \\ \vdots \end{bmatrix} \begin{bmatrix} \mathbf{D} & & \\ & \ddots & \\ & & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{D}^T \mathbf{D} & & \\ & \ddots & \\ & & \mathbf{D}^T \mathbf{D} \end{bmatrix}$$

Assume
normalized
columns

- The **Mutual Coherence** μ is the largest off-diagonal entry in absolute value.
- The Mutual Coherence is a property of the dictionary (just like the “Spark”). In fact, the following relation can be shown:

$$\sigma \geq 1 + \frac{1}{\mu}$$



BP and MP Equivalence (No Noise)

Equivalence

Donoho & E. ('02)

Gribonval & Nielsen ('03)

Tropp ('03)

Temlyakov ('03)

Given a signal \underline{x} with a representation $\underline{x} = \mathbf{D}\underline{\alpha}$, assuming that $\|\underline{\alpha}\|_0^0 < 0.5(1 + 1/\mu)$, BP and MP are guaranteed to find the sparsest solution.

$$\hat{\underline{\alpha}} = \underset{\underline{\alpha}}{\text{ArgMin}} \|\underline{\alpha}\|_0^0 \quad \text{s.t. } \underline{x} = \mathbf{D}\underline{\alpha}$$

- MP and BP are different in general (hard to say which is better).
- The above result corresponds to the worst-case, and as such, it is too pessimistic.
- Average performance results are available too, showing much better bounds [Donoho ('04)] [Candes et.al. ('04)] [Tanner et.al. ('05)] [E. ('06)] [Tropp et.al. ('06)] ... [Candes et. al. ('09)].



BP Stability for the Noisy Case

Stability

Given a signal $\underline{y} = \mathbf{D}\underline{\alpha} + \underline{v}$ with a representation satisfying $\|\underline{\alpha}\|_0^0 < 1 / 3\mu$ and a white Gaussian noise $\underline{v} \sim N(0, \sigma^2 \mathbf{I})$, BP will show* stability, i.e.,

$$\|\hat{\underline{\alpha}}_{\text{BP}} - \underline{\alpha}\|_2^2 < \text{Const}(\lambda) \cdot \log K \cdot \|\underline{\alpha}\|_0^0 \cdot \sigma^2$$

Ben-Haim, Eldar & E. (09)

* With very high probability

- For $\sigma=0$ we get
- This result is true for any vector,
- Similar results for Orthogonal Matrices

$$\min_{\underline{\alpha}} \lambda \|\underline{\alpha}\|_1 + \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2$$



To Summarize So Far ...

Image denoising
(and many other
problems in image
processing) requires
a model for the
desired image

What do
we do?

We proposed a
model for
signals/images
based on sparse
and redundant
representations

Problems?

The
Dictionary **D**
should be
found
somehow !!!

What
next?

We have seen that there are
approximation methods to
find the sparsest solution,
and there are theoretical
results that guarantee their
success.



Part III

Dictionary Learning: The K-SVD Algorithm



What Should \mathbf{D} Be?

$$\hat{\underline{\alpha}} = \underset{\underline{\alpha}}{\operatorname{arg\,min}} \|\underline{\alpha}\|_0^0 \quad \text{s.t. } \frac{1}{2} \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \leq \varepsilon^2 \rightarrow \hat{\underline{x}} = \mathbf{D}\hat{\underline{\alpha}}$$

Our Assumption: Good-behaved Images
have a sparse representation



\mathbf{D} should be chosen such that it sparsifies the representations

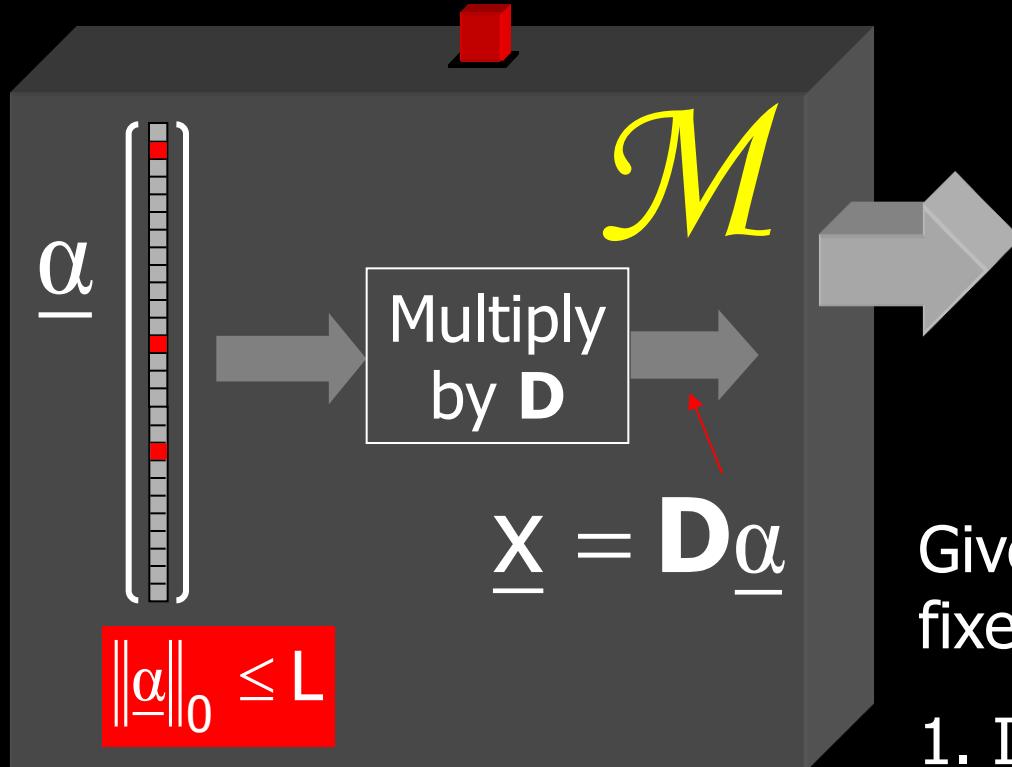


One approach to choose \mathbf{D} is from
a known set of transforms
(Steerable wavelet, Curvelet,
Contourlets, Bandlets, Shearlets ...)

The approach we will take for
building \mathbf{D} is training it,
based on **Learning** from
Image Examples



Dictionary Learning: Problem Setting



$$\left\{ \underline{X}_j \right\}_{j=1}^P$$

$$\left\{ \begin{array}{c} \text{Red grid} \\ \dots \\ \text{Red grid} \end{array} \right\}$$

Given these P examples and a fixed size $[N \times K]$ dictionary \mathbf{D} :

1. Is \mathbf{D} unique?
2. How would we find \mathbf{D} ?



Measure of Quality for \mathbf{D}

$$\left[\begin{array}{c|c|c} \text{X} & \cdots & \end{array} \right] \approx \left[\begin{array}{c|c|c} \mathbf{D} & & \end{array} \right] \left[\begin{array}{c|c|c} \mathbf{A} & \cdots & \end{array} \right]$$

$$\underset{\mathbf{D}, \mathbf{A}}{\text{Min}} \sum_{j=1}^P \|\mathbf{D}\underline{\alpha}_j - \underline{x}_j\|_2^2$$

Each example is
a linear combination
of atoms from \mathbf{D}

$$\text{s.t. } \forall j, \|\underline{\alpha}_j\|_0^0 \leq L$$

Each example has a
sparse representation with
no more than L atoms

[Field & Olshausen ('96)]

[Engan et. al. ('99)]

[Lewicki & Sejnowski ('00)]

[Cotter et. al. ('03)]

[Gribonval et. al. ('04)]

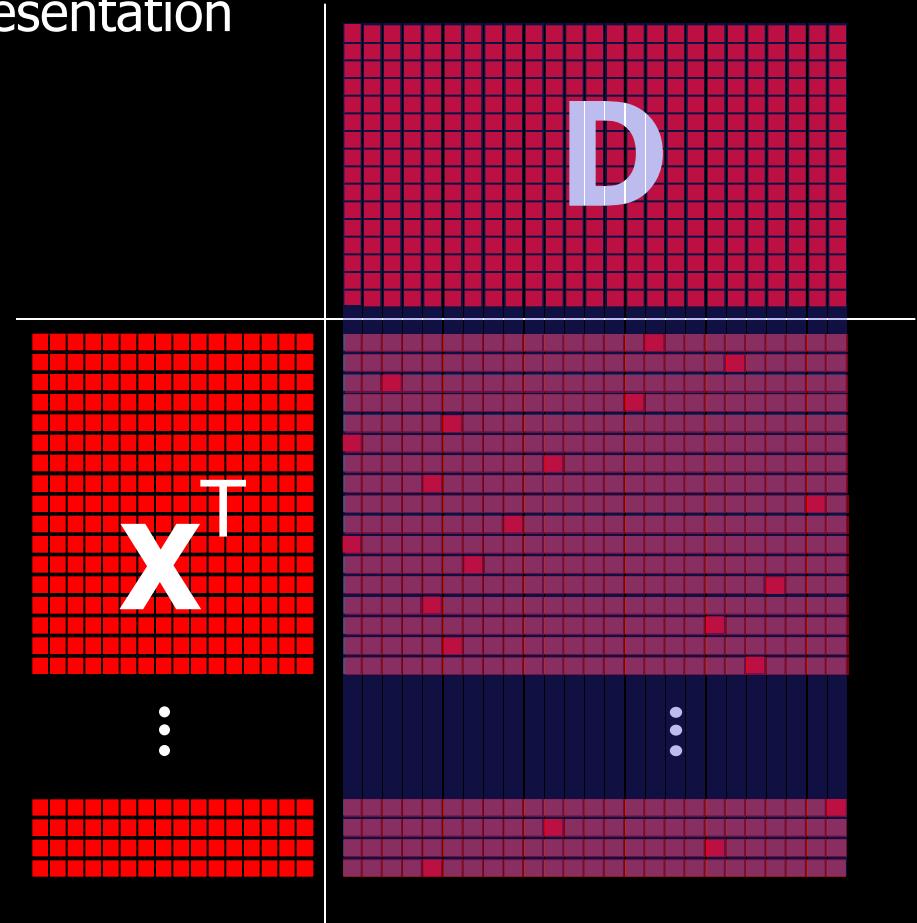
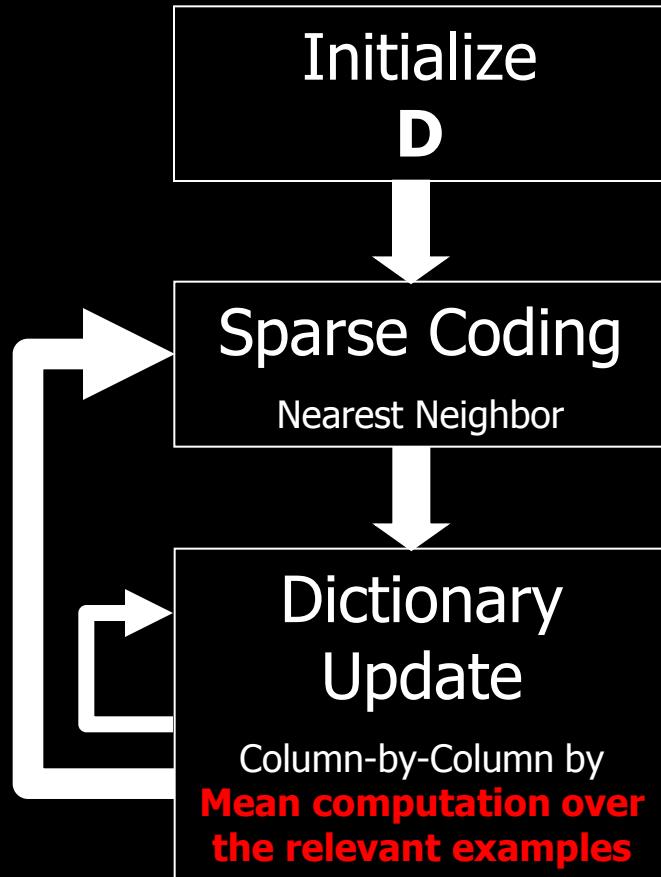
[Aharon, E. & Bruckstein ('04)]

[Aharon, E. & Bruckstein ('05)]



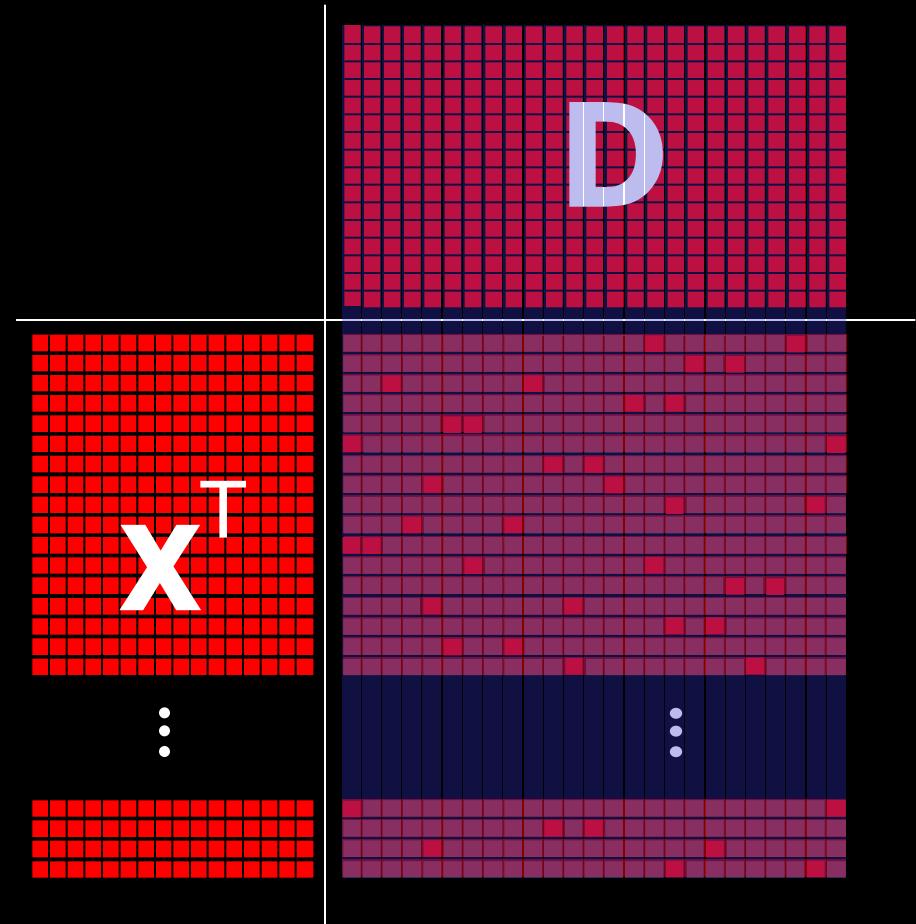
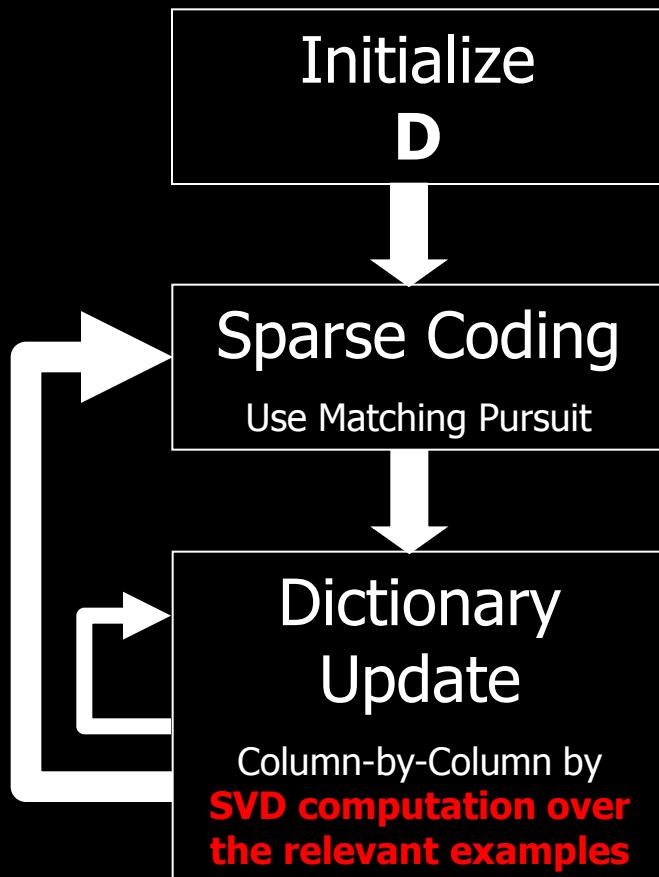
K-Means For Clustering

Clustering: An extreme sparse representation



The K-SVD Algorithm – General

[Aharon, E. & Bruckstein ('04,'05)]



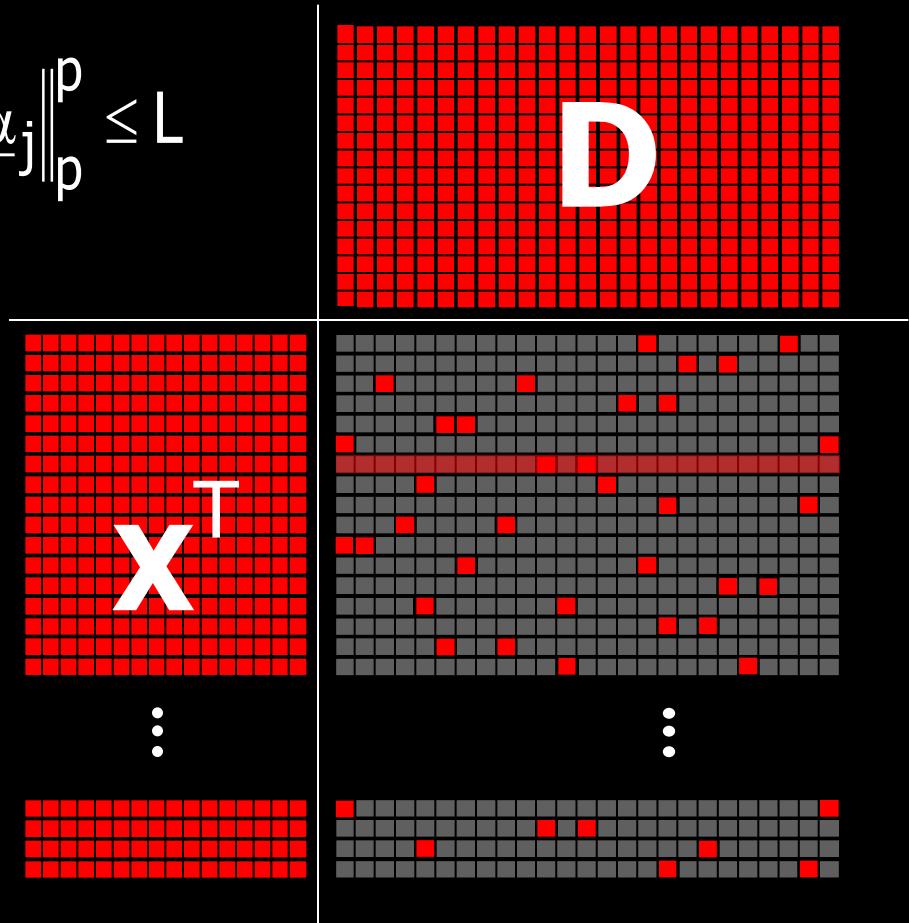
K-SVD: Sparse Coding Stage

$$\underset{\mathbf{A}}{\text{Min}} \quad \sum_{j=1}^P \left\| \mathbf{D}\underline{\alpha}_j - \underline{x}_j \right\|_2^2 \quad \text{s.t.} \quad \forall j, \left\| \underline{\alpha}_j \right\|_p^p \leq L$$

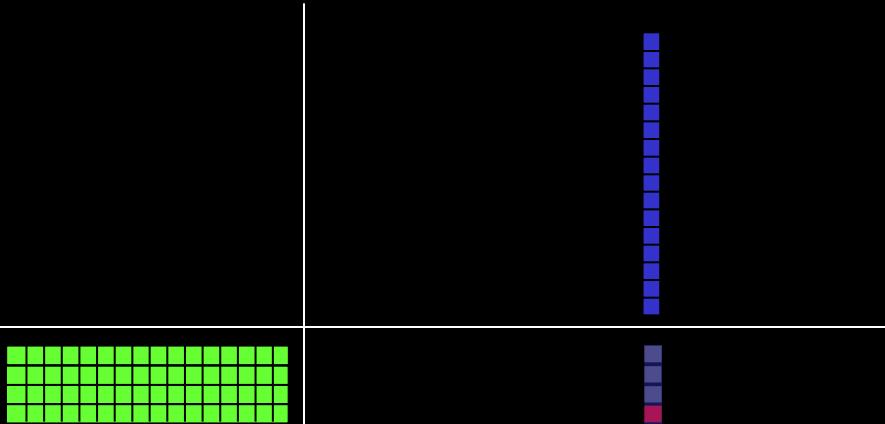
D is known!
For the j^{th} item
we solve

$$\underset{\underline{\alpha}}{\text{Min}} \quad \left\| \mathbf{D}\underline{\alpha} - \underline{x}_j \right\|_2^2 \quad \text{s.t.} \quad \left\| \underline{\alpha} \right\|_p^p \leq L$$

Solved by
A Pursuit Algorithm



K-SVD: Dictionary Update Stage

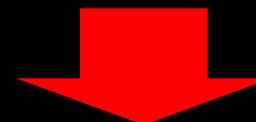


We should solve:

$$\text{Min}_{\underline{\mathbf{d}}_k, \alpha} \left\| \alpha_k \underline{\mathbf{d}}_k - \mathbf{E} \right\|_F^2$$

The word "SVD" is written in large red letters across the equation.

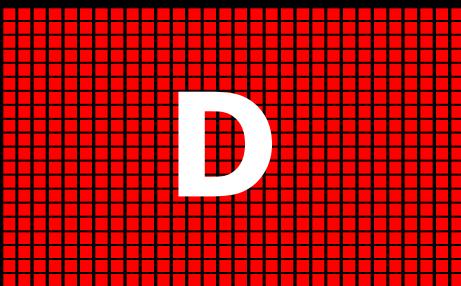
We refer only to the examples that use the column $\underline{\mathbf{d}}_k$



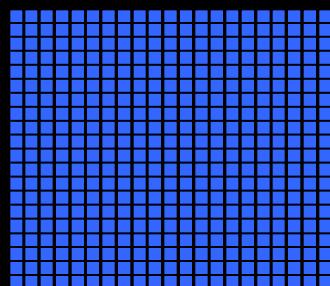
Fixing all \mathbf{A} and \mathbf{D} apart from the k^{th} column, and seek both $\underline{\mathbf{d}}_k$ and the k^{th} column in \mathbf{A} to better fit the **residual!**

A Synthetic Experiment

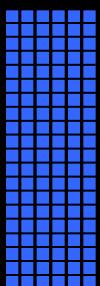
Create A 20×30 random dictionary
with normalized columns



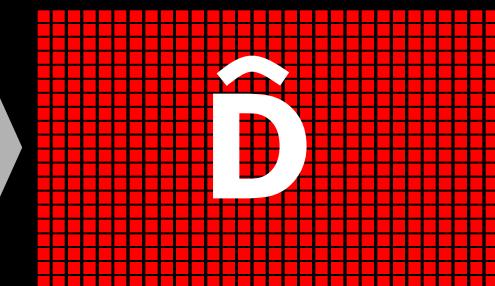
Generate 2000 signal examples with
3 atoms per each and add noise



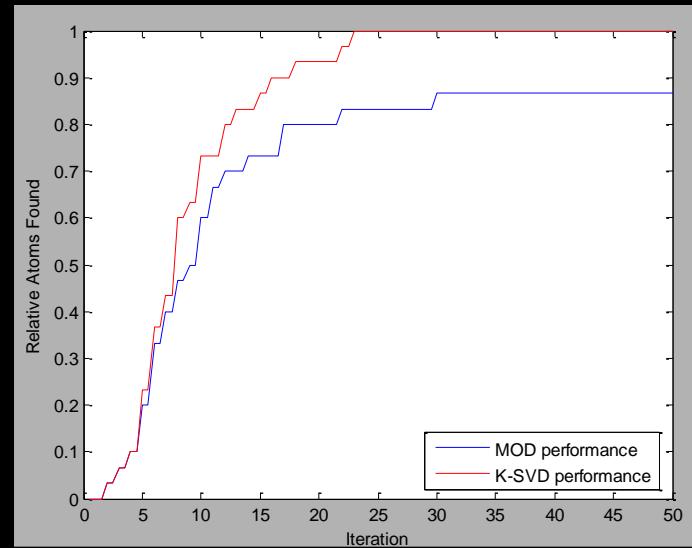
...



Train a dictionary using the KSVD
and MOD and compare



Results



To Summarize So Far ...

Image denoising
(and many other
problems in image
processing) requires
a model for the
desired image

What do
we do?

We proposed a
model for
signals/images
based on sparse
and redundant
representations

Problems?

Will it all
work in
applications?

What
next?

We have seen approximation
methods that find the
sparsest solution, and
theoretical results that
guarantee their success. We
also saw a way to learn **D**



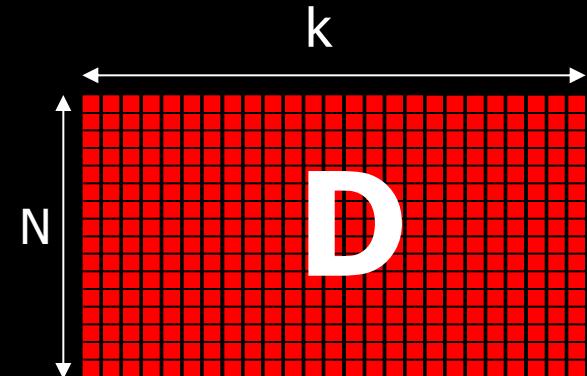
Part IV

Back to Denoising ... and Beyond – Combining it All



From Local to Global Treatment

- The K-SVD algorithm is reasonable for low-dimension signals (N in the range 10-400). As N grows, the complexity and the memory requirements of the K-SVD become prohibitive.
- So, how should large images be handled?
- **The solution:** Force shift-invariant sparsity - on each patch of size N -by- N ($N=8$) in the image, including overlaps.



$$\hat{\underline{x}} = \underset{\underline{x}, \{\alpha_{ij}\}_{ij}}{\text{ArgMin}} \quad \frac{1}{2} \|\underline{x} - \underline{y}\|_2^2 + \mu \sum_{ij} \left\| \mathbf{R}_{ij} \underline{x} - \mathbf{D}_{\alpha_{ij}} \right\|_2^2$$

Extracts a patch
in the ij location

$$\text{s.t. } \left\| \alpha_{ij} \right\|_0^0 \leq L$$

Our prior



What Data to Train On?

Option 1:

- Use a database of images,
- We tried that, and it works fine ($\sim 0.5\text{-}1\text{dB}$ below the state-of-the-art).

Option 2:

- Use the corrupted image itself !!
- Simply sweep through all patches of size N-by-N (overlapping blocks),
- Image of size 1000^2 pixels $\rightarrow \sim 10^6$ examples to use – more than enough.
- This works much better!



K-SVD Image Denoising

$$\hat{\underline{x}} = \underset{\underline{x}, \{\underline{\alpha}_{ij}\}_{ij}}{\text{ArgMin}} \frac{1}{2} \|\underline{x} - \underline{y}\|_2^2 + \mu \sum_{ij} \|\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}_{ij}\|_2^2 \text{ s.t. } \|\underline{\alpha}_{ij}\|_0^0 \leq L$$

$\underline{x} = \underline{y}$ and \mathbf{D} known

Compute α_{ij} per patch

$$\underline{\alpha}_{ij} = \underset{\underline{\alpha}}{\text{Min}} \|\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}\|_2^2$$

$$\text{s.t. } \|\underline{\alpha}\|_0^0 \leq L$$

using the matching pursuit

\underline{x} and α_{ij} known

Compute \mathbf{D} to minimize

$$\underset{\underline{\alpha}}{\text{Min}} \sum_{ij} \|\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}\|_2^2$$

using SVD, updating one column at a time

\mathbf{D} and α_{ij} known

Compute \underline{x} by

$$\underline{x} = \left[I + \mu \sum_{ij} \mathbf{R}_{ij}^T \mathbf{R}_{ij} \right]^{-1} \left[\underline{y} + \mu \sum_{ij} \mathbf{R}_{ij}^T \mathbf{D} \underline{\alpha}_{ij} \right]$$

which is a simple averaging of shifted patches

K-SVD



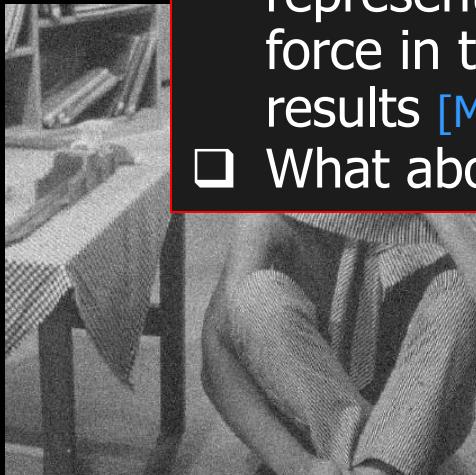
Image Denoising (Gray) [E. & Aharon ('06)]



Source



- The results of this algorithm compete favorably with the state-of-the-art.
- This algorithm can be extended by using joint sparse representation on the patches, introducing a non-local force in the denoising, thus leading to improved results [[Mairal, Bach, Ponce, Sapiro & Zisserman \('09\)](#)].
- What about EPLL ? ...



Noisy image
 $\sigma = 20$



The obtained dictionary after
10 iterations

Denoising (Color) [Mairal, E. & Sapiro ('08)]

- When turning to handle color images, the



Original



Noisy (20.43dB)



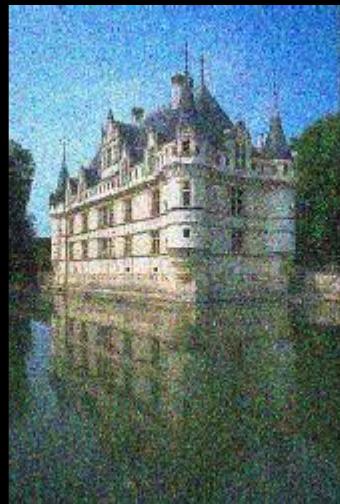
Result (30.75dB)

Denoising (Color) [Mairal, E. & Sapiro ('08)]

Our experiments lead to state-of-the-art denoising results, giving $\sim 1\text{dB}$ better results compared to [McAuley et. al. ('06)] which implements a learned MRF model (Field-of-Experts)



Original



Noisy (12.77dB)



Result (29.87dB)

Video Denoising [Protter & E. ('09)]



Our experiments lead to state-of-the-art video denoising results, giving $\sim 0.5\text{dB}$ better results on average compared to [Boades, Coll & Morel ('05)] and comparable to [Rusanovskyy, Dabov, & Egiazarian ('06)]

Original

compensation can and should be avoided [Buades, Col, and Morel ('06)].

Noisy ($\sigma=15$)

Denoised (PSNR=29.98)



Low-Dosage Tomography [Shtok, Zibulevsky & E. (10)]

- In Computer-Tomography (CT) reconstruction, an image is recovered from a set of its projections.
- In medicine, CT projections are obtained by X-ray, and it typically requires a high dosage of radiation in order to obtain a good quality reconstruction.
- A lower-dosage projection implies a stronger noise (Poisson distributed) in data to work with.
- Armed with sparse and redundant representation modeling, we can denoise the data and the final reconstruction ... enabling CT with lower dosage.



Image Inpainting – The Basics

- Assume: the signal \underline{x} has been created by $\underline{x} = \mathbf{D}\underline{\alpha}_0$ with very sparse $\underline{\alpha}_0$.

- Missing values in \underline{x} imply missing rows in this linear system.

- By removing these rows, we get

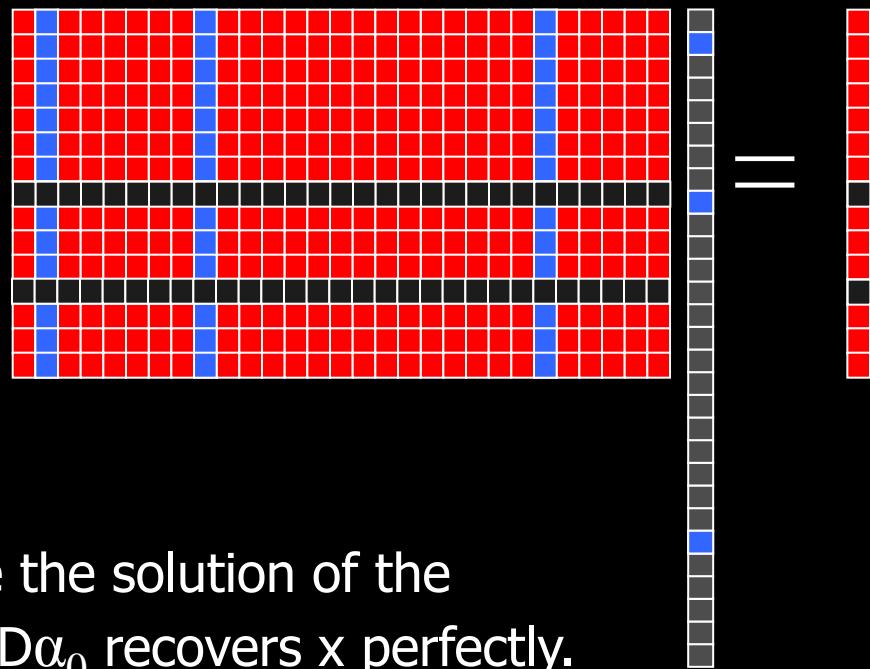
$$\tilde{\mathbf{D}}\underline{\alpha} = \tilde{\underline{x}}$$

- Now solve

$$\underset{\underline{\alpha}}{\text{Min}} \|\underline{\alpha}\|_0 \quad \text{s.t. } \tilde{\underline{x}} = \tilde{\mathbf{D}}\underline{\alpha}$$

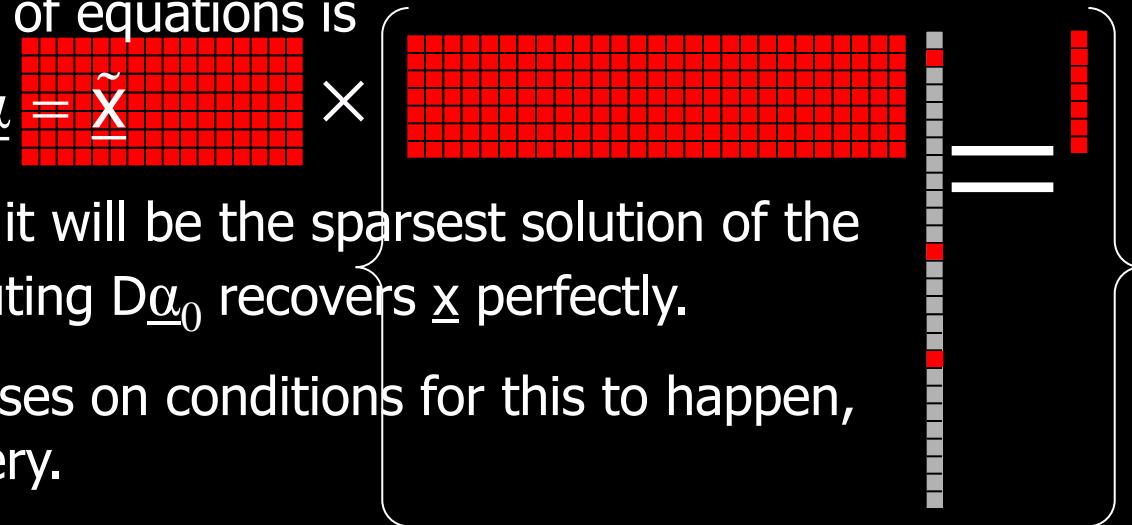
- If $\underline{\alpha}_0$ was sparse enough, it will be the solution of the above problem! Thus, computing $\mathbf{D}\underline{\alpha}_0$ recovers \underline{x} perfectly.

$$\mathbf{D} \underline{\alpha}_0 = \underline{x}$$

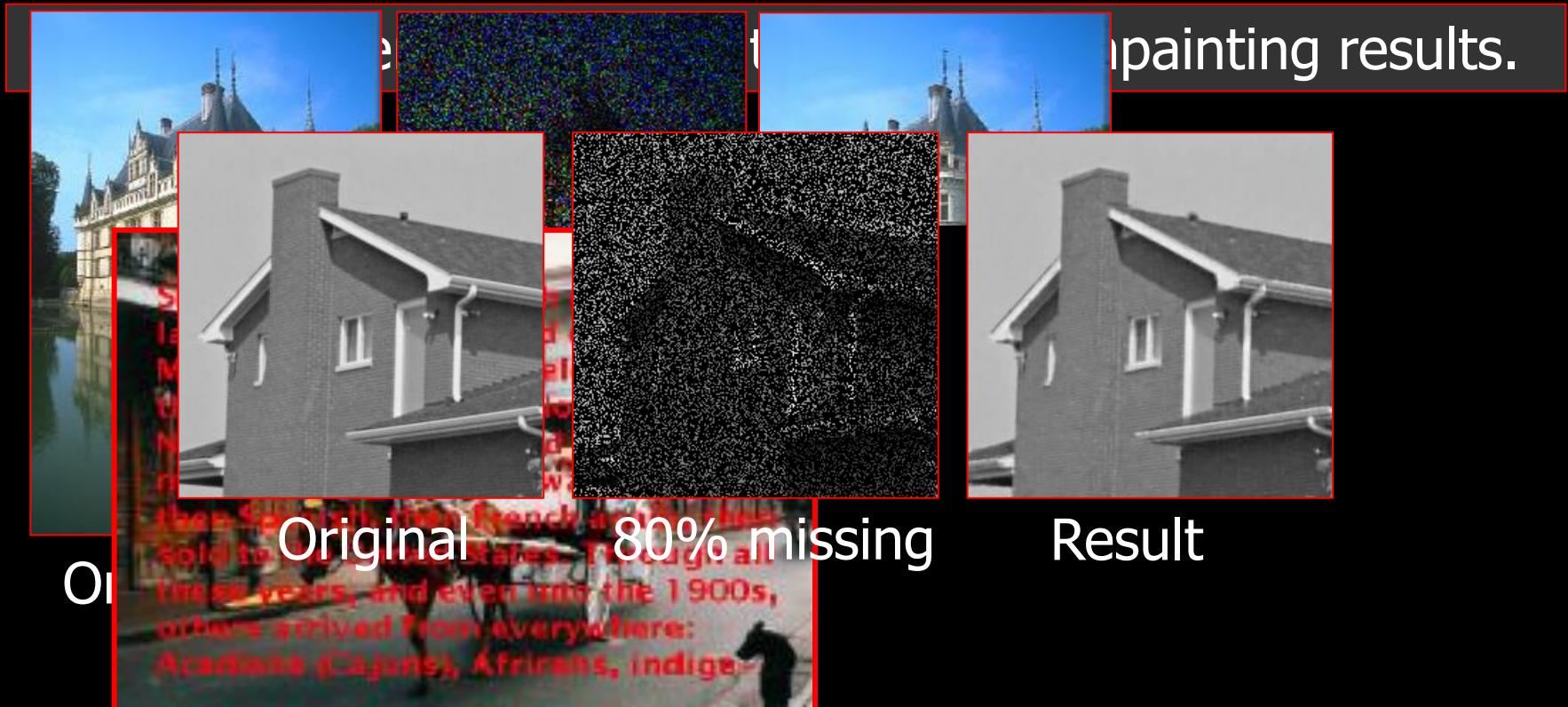


Side Note: Compressed-Sensing

- Compressed Sensing is leaning on the very same principal, leading to alternative sampling theorems.
- Assume: the signal \underline{x} has been created by $\underline{x} = D\underline{a}_0$ with very sparse \underline{a}_0 .
- Multiply this set of equations by the matrix \mathbf{Q} which reduces the number of rows.
- The new, smaller, system of equations is
$$\mathbf{Q}\mathbf{D}\underline{a} = \mathbf{Q}\underline{x} \rightarrow \tilde{\mathbf{D}}\underline{a} = \tilde{\underline{x}}$$
- If \underline{a}_0 was sparse enough, it will be the sparsest solution of the new system, thus, computing $D\underline{a}_0$ recovers \underline{x} perfectly.
- Compressed sensing focuses on conditions for this to happen, guaranteeing such recovery.

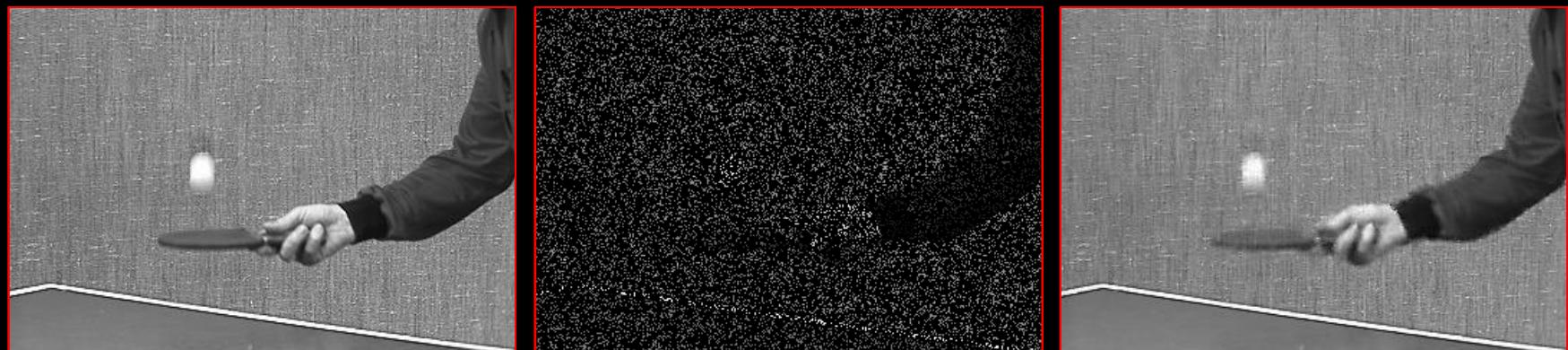


Inpainting [Mairal, E. & Sapiro ('08)]



Inpainting [Mairal, E. & Sapiro ('08)]

The same can be done for video, very much like the denoising treatment: (i) 3D patches, (ii) no need to compute the dictionary from scratch for each frame, and (iii) no need for explicit motion estimation



Original

80% missing

Result



Demosaicing [Mairal, E. & Sapiro ('08)]

- Our experiments lead to state-of-the-art demosaicing
- Today's cameras are sensing only one color per pixel, leaving the rest for interpolation.

- Generalizing the inpainting scheme to handle demosaicing is tricky because of the possibility to learn the mosaic pattern within the dictionary.

- In order to avoid “over-fitting”, we handle the demosaicing problem while forcing strong sparsity and applying only few iterations.

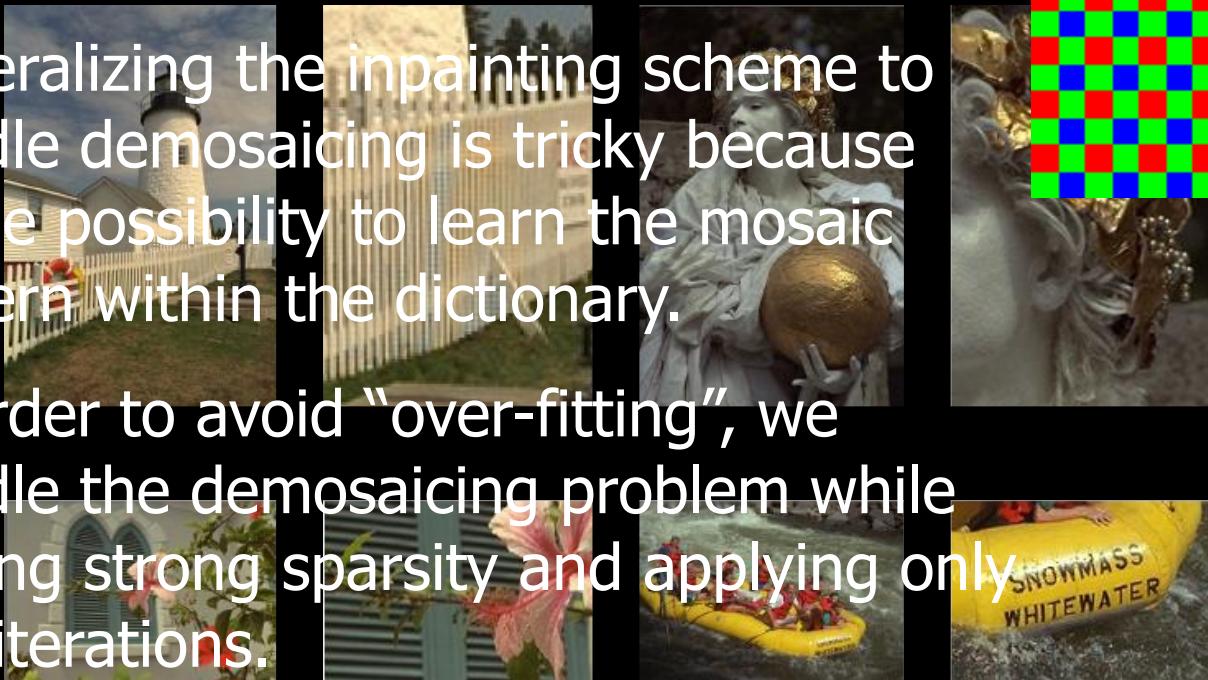


Image Compression [Bryt and E. ('08)]

- The problem: Compressing photo-ID images.
- General purpose methods (JPEG, JPEG2000) do not take into account the specific family.
- By adapting to the image-content (PCA/K-SVD), better results could be obtained.
- For these techniques to operate well, train dictionaries locally (per patch) using a training set of images is required.
- In PCA, only the (quantized) coefficients are stored, whereas the K-SVD requires storage of the indices as well.
- Geometric alignment of the image is very helpful and should be done [Goldenberg, Kimmel, & E. ('05)].

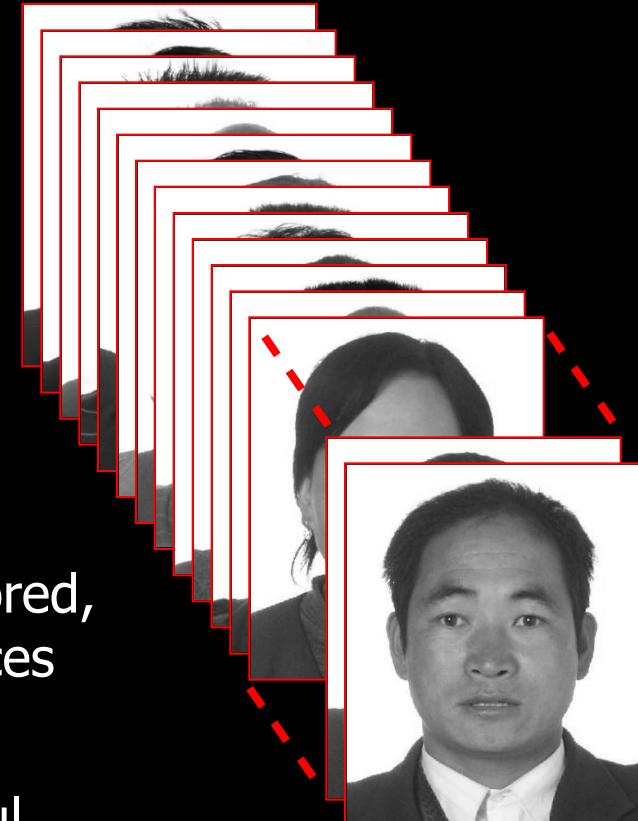
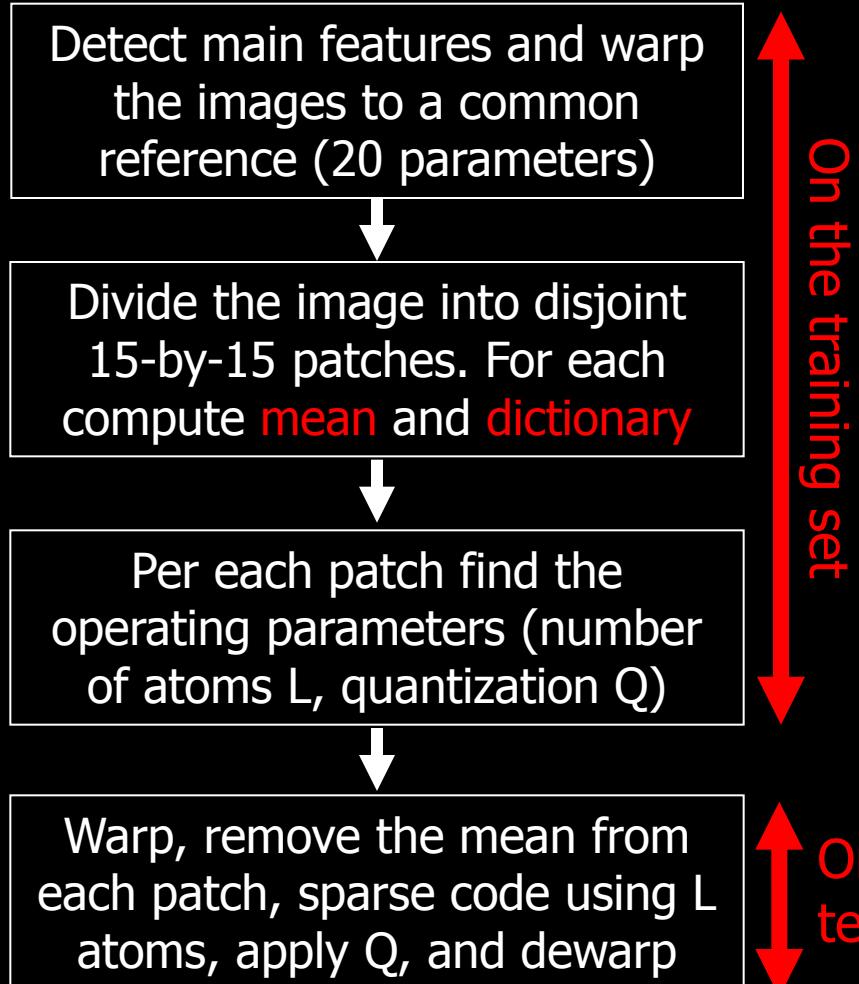


Image Compression



Training set (2500 images)

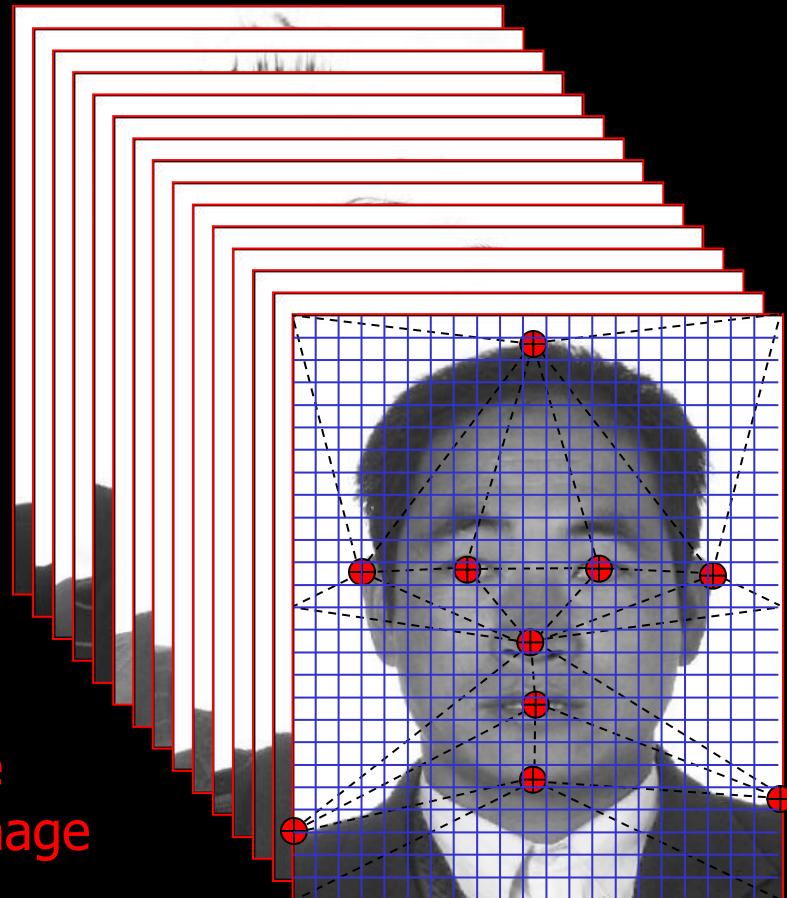
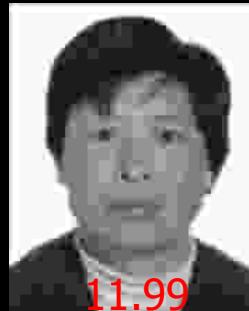


Image Compression Results

Original



JPEG



11.99

JPEG-2000



10.49

Local-PCA



8.81



5.56

K-SVD



10.83



8.92



7.89



4.82

Results
for **820**
Bytes per
each file



10.93



8.71



8.61



5.58

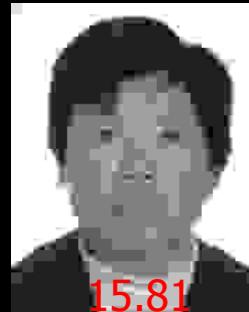


Image Compression Results

Original



JPEG



JPEG-2000



Local-PCA



K-SVD



Results
for **550**
Bytes per
each file



Image Compression Results

Original



JPEG

?

JPEG-2000



18.62

Local-PCA



12.30

K-SVD



?



16.12



6.31

Results
for **400**
Bytes per
each file



?



16.81



12.54



7.20



Deblocking the Results [Bryt and E. ('09)]

550 bytes
K-SVD
results with
and without
deblocking



K-SVD (6.60)



K-SVD (5.49)



K-SVD (6.45)



K-SVD (11.67)



Deblock (6.24)



Deblock (5.27)



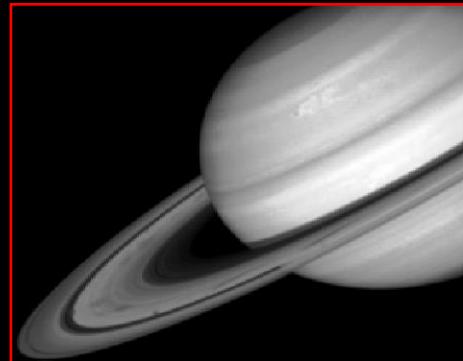
Deblock (6.03)



Deblock (11.32)



Poisson Denoising



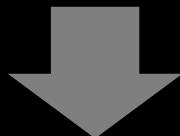
+



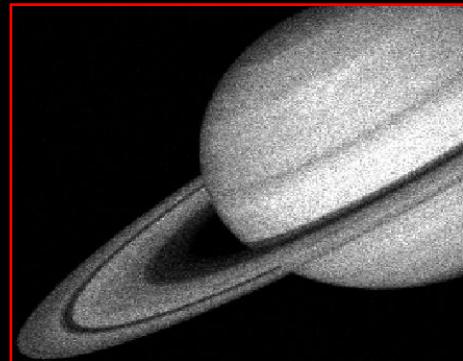
=



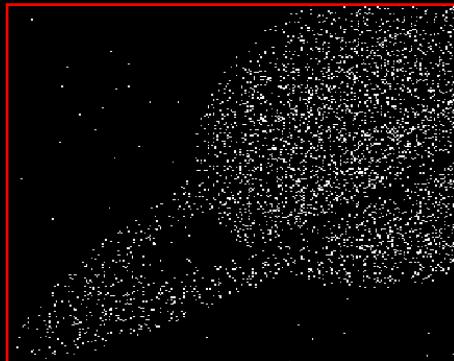
$$\underline{Y} = \underline{X} + \underline{V}$$
$$\underline{V} \sim \mathbb{N}(\underline{0}, \sigma^2 \mathbf{I})$$



peak = 100



peak = 0.1



$$P(y | x) = \frac{x^y}{y!} e^{-x}$$

$$\text{peak} \triangleq \max_{i,j} \{x_{i,j}\}$$



Poisson Denoising

[Salmon et. al., 2011] [Giryes et. al., 2013]

- Anscombe transform converts Poisson distributed noise into an approximately Gaussian one, with variance 1 using the following formula [Anscombe, 1948]:

$$f_{\text{Anscombe}}(y) = 2\sqrt{y + \frac{3}{8}}$$

- However, this is of reasonable accuracy only if $\text{peak} > 4$.
- For lower peaks (poor illumination), we use the patch-based approach with dictionary learning, BUT ... in the exponent domain:

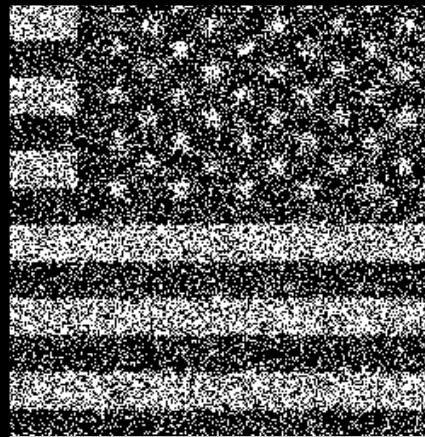
$$\left\{ \begin{array}{l} \underline{x} = \mathbf{D}\underline{\alpha} \\ \text{where } \|\underline{\alpha}\|_0 \leq L \end{array} \right\} \rightarrow \left\{ \begin{array}{l} \underline{x} = \exp\{\mathbf{D}\underline{\alpha}\} \\ \text{where } \|\underline{\alpha}\|_0 \leq L \end{array} \right\}$$



Poisson Denoising – Results (1)



Original

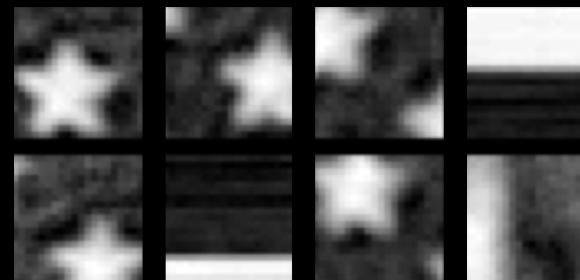


Noisy (peak=1)

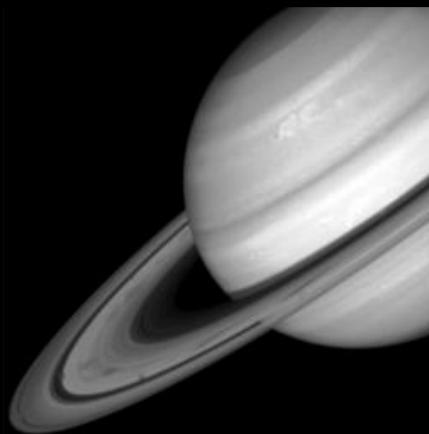


Result (PSNR=22.59dB)

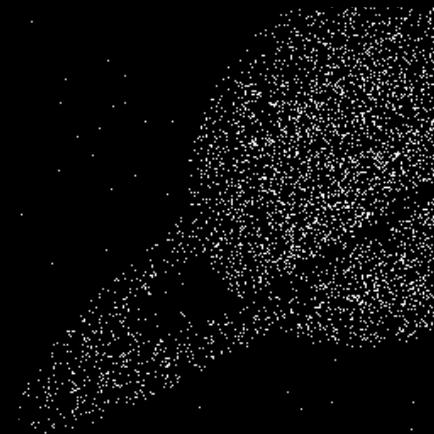
Dictionary learned atoms:



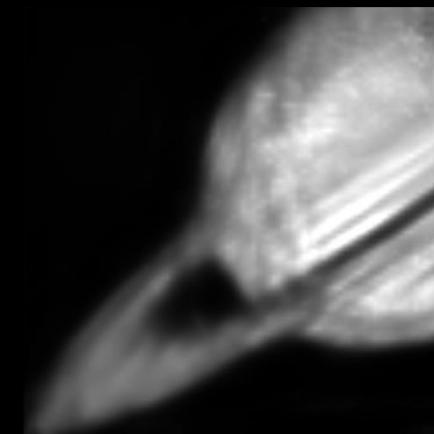
Poisson Denoising – Results (2)



Original



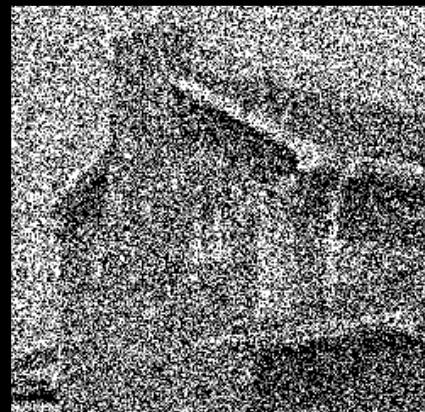
Noisy (peak=0.2)



Result (PSNR=24.16dB)



Original



Noisy (peak=2)



Result (PSNR=24.76dB)



Other Applications?

- Poisson Inpainting
- Super-Resolution
- Blind deblurring
- Audio inpainting
- Dynamic MRI reconstruction
- Clutter reduction in Ultrasound
- Single image interpolation
- Anomaly detection
- ...



To Summarize So Far ...

Image denoising
(and many other
problems in image
processing) requires
a model for the
desired image

What do
we do?

We proposed a
model for
signals/images
based on sparse
and redundant
representations

Well, does
this work?

Well, many
more things ...

So, what
next?

Yes! We have seen a group of
applications where this model is
showing very good results:
denoising of bw/color stills/video,
CT improvement, inpainting,
super-resolution, and
compression



Part V

Summary and Conclusion



Today We Have Seen that ...

Sparsity, Redundancy, and the use of examples are important ideas that can be used in designing better tools in signal/image processing

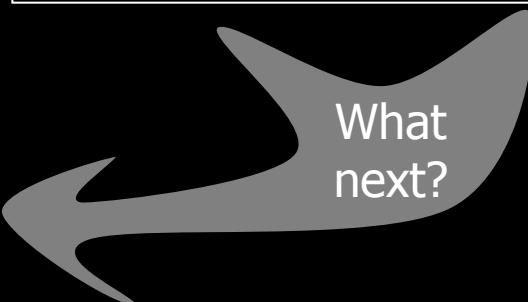


What do we do?

In our work on we cover theoretical, numerical, and applicative issues related to this model and its use in practice.

We keep working on:

- Improving the model
- Improving the dictionaries
- Demonstrating on other applications
- ...



What next?



Thank You

All this Work is Made Possible Due to

my teachers and mentors



A.M. Bruckstein D.L. Donoho

colleagues & friends collaborating with me



G. Sapiro J.L. Starck I. Yavneh M. Zibulevsky

and my students



M. Aharon

O. Bryt

J. Mairal

M. Protter

R. Rubinstein

J. Shtok

R. Giryes

Z. Ben-Haim

J. Turek

R. Zeyde

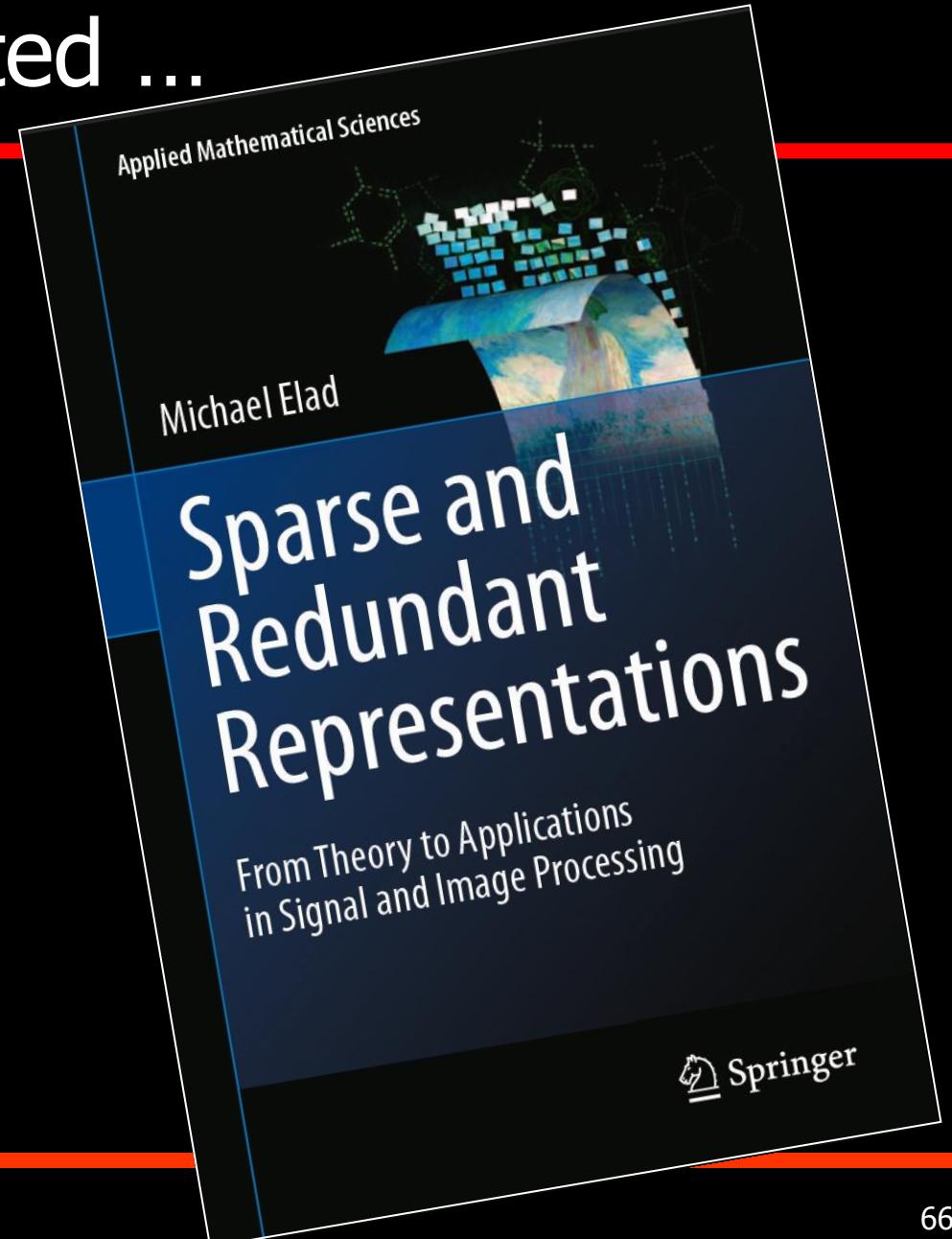


If you are Interested ...

More on this topic (including the slides, the papers, and Matlab toolboxes) can be found in my webpage:

<http://www.cs.technion.ac.il/~elad>

A book on these topics was published in August 2010.



Thank You all !



Questions?

More on these (including the slides and the relevant papers) can be found in
<http://www.cs.technion.ac.il/~elad>



Dictionary Learning: Uniqueness?

Uniqueness

Aharon, E., & Bruckstein ('05)

If $\{\underline{x}_j\}_{j=1}^P$ is rich enough* and if

$$L < \frac{\text{Spark}\{\mathbf{D}\}}{2}$$

then \mathbf{D} is unique.

Comments:

- “Rich Enough”: The signals from \mathcal{M} could be clustered to $\binom{K}{L}$ groups that share the same support. At least $L+1$ examples per each are needed. More recent results (see Schnass and Wright’s work) improve this dramatically.
- This result is proved constructively, but the number of examples needed to pull this off is huge – we will show a far better method next.
- A parallel result that takes into account noise is yet to be constructed.



Improved Dictionary Learning

$$\underset{\mathbf{D}, \mathbf{A}}{\text{Min}} \sum_{j=1}^P \left\| \mathbf{D} \underline{\alpha}_j - \underline{x}_j \right\|_2^2 \quad \text{s.t. } \forall j, \left\| \underline{\alpha}_j \right\|_0^0 \leq L$$

MOD Algorithm

Fix \mathbf{D} and update \mathbf{A}

Fix \mathbf{A} and update \mathbf{D}

K-SVD Algorithm

Fix \mathbf{D} and update \mathbf{A}

for $j=1:1:K$

- Fix \mathbf{A} & \mathbf{D} apart from the j -th atom its coefficients
 - Update \underline{d}_j and its coef. in \mathbf{A}
- end



Improved Dictionary Learning

$$\underset{\mathbf{D}, \mathbf{A}}{\text{Min}} \sum_{j=1}^P \left\| \mathbf{D}\underline{\alpha}_j - \underline{x}_j \right\|_2^2 \quad \text{s.t. } \forall j, \|\underline{\alpha}_j\|^0 \leq L$$

Improved Algorithm

MOD and K-SVD can be considered as crude approximation of this method

Fix D and update A
Fix the supports
in A and update
D until $\|\mathbf{D}\mathbf{A} - \mathbf{X}\|_F^2 \leq \epsilon$
 $\mathbf{A} \odot \mathbf{M} = 0$

and the non-zeros

This can be done in two ways:

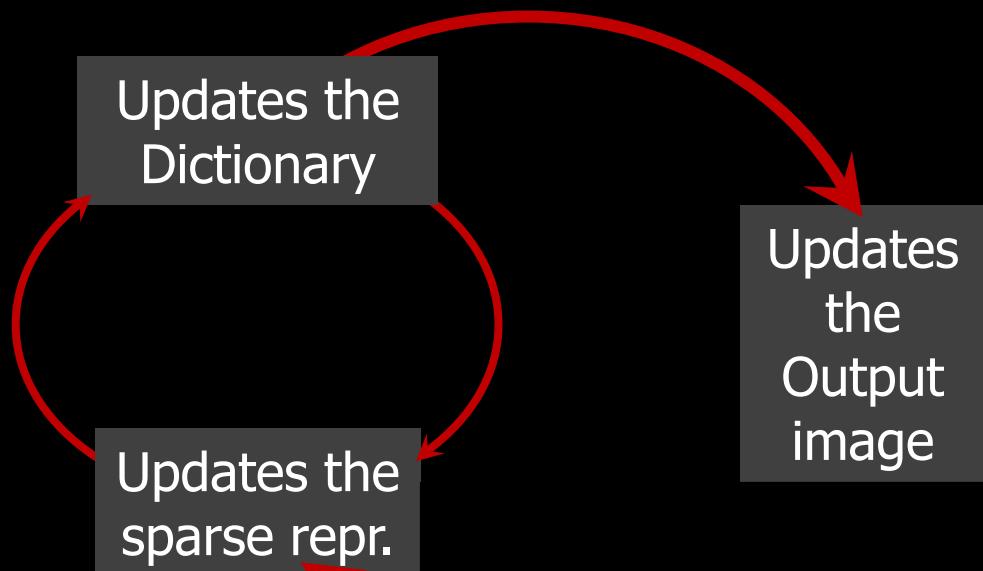
1. Apply several rounds of the atoms' update in the K-SVD, or
2. Extend the MOD to update the non-zero elements in **A**



EPPLL Improvement [Sulam and E. (15)]

$$\hat{\underline{x}} = \underset{\underline{x}, \{\underline{\alpha}_{ij}\}_{ij}, \mathbf{D}}{\text{ArgMin}} \frac{1}{2} \|\underline{x} - \underline{y}\|_2^2 + \mu \sum_{ij} \|\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}_{ij}\|_2^2 \text{ s.t. } \|\underline{\alpha}_{ij}\|_0^0 \leq L$$

- The algorithm we proposed updates \underline{x} only once at the end.
- Why not repeat the whole process several times?
- The rationale: The sparse representation model should be imposed on the patches of the FINAL image. After averaging, this is ruined.



EPLL Improvement [Sulam and E. ('15)]

- Expected Patch Log Likelihood (EPLL) is an algorithm that came to fix this problem [Zoran and Weiss, ('11)] in the context of a GMM prior.
- An extension of EPLL to Spars-Land is proposed in [Sulam and E. ('15)].
The core idea is:
 - After the image has been computed, we proceed the iterative process, and apply several such overall rounds of updates.
 - Sparse coding must be done with a new threshold, based on the remaining noise in the image. This is done by evaluating the noise level based on the linear projections (disregarding the support detection by the OMP).
 - This algorithm leads to state-of-the-art results, with 0.5-1dB improvement over the regular K-SVD algorithm shown before.

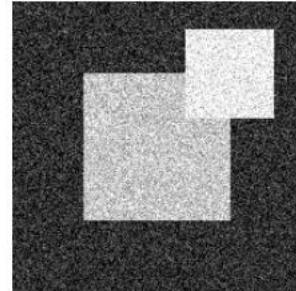


EPPLL Improvement [Sulam and E. ('15)]

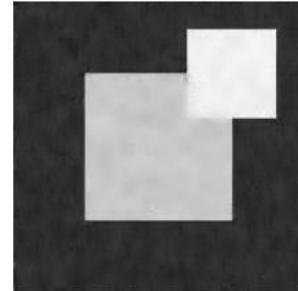
Original Image



Noisy Image. PSNR = 18.59 dB



K-SVD. PSNR = 34.45 dB



Noisy image
has $\sigma=25$

KSVD PSNR
31.42 dB

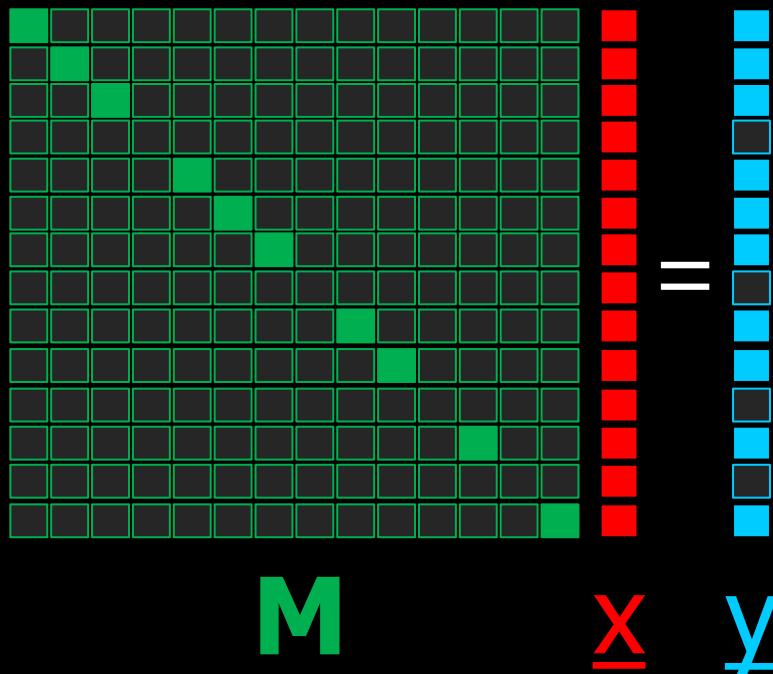
EPPLL PSNR
31.83 dB



Inpainting Formulation [Mairal, E. & Sapiro ('08)]

$$\hat{\underline{x}} = \underset{\underline{x}, \{\underline{\alpha}_{ij}\}_{ij}, \mathbf{D}}{\text{ArgMin}} \frac{1}{2} \left\| \mathbf{M} \underline{x} - \underline{y} \right\|_2^2 + \mu \sum_{ij} \left\| \mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}_{ij} \right\|_2^2 \text{ s.t. } \left\| \underline{\alpha}_{ij} \right\|_0^0 \leq L$$

The matrix \mathbf{M} is a mask matrix, obtained by the identity matrix with some of its rows omitted, corresponding to the missing samples



Inpainting Formulation [Mairal, E. & Sapiro ('08)]

$$\hat{\underline{x}} = \underset{\underline{x}, \{\underline{\alpha}_{ij}\}_{ij}, \mathbf{D}}{\text{ArgMin}} \frac{1}{2} \left\| \mathbf{M} \underline{x} - \underline{y} \right\|_2^2 + \mu \sum_{ij} \left\| \mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}_{ij} \right\|_2^2 \text{ s.t. } \left\| \underline{\alpha}_{ij} \right\|_0^0 \leq L$$

$\underline{x} = \underline{y}$ and \mathbf{D} known

\underline{x} and $\underline{\alpha}_{ij}$ known

\mathbf{D} and $\underline{\alpha}_{ij}$ known

Compute $\underline{\alpha}_{ij}$ per patch

$$\underline{\alpha}_{ij} = \underset{\underline{\alpha}}{\text{Min}} \left\| \mathbf{M}_{ij} (\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}) \right\|_2^2$$

$$\text{s.t. } \left\| \underline{\alpha} \right\|_0^0 \leq L$$

using the matching pursuit

Compute \mathbf{D} to minimize

$$\underset{\underline{\alpha}}{\text{Min}} \sum_{ij} \left\| \mathbf{M}_{ij} (\mathbf{R}_{ij} \underline{x} - \mathbf{D} \underline{\alpha}) \right\|_2^2$$

using SVD, updating one column at a time

\sim K-SVD

Compute \underline{x} by

$$\underline{x} = \left[\mathbf{M}^T \mathbf{M} + \mu \sum_{ij} \mathbf{R}_{ij}^T \mathbf{R}_{ij} \right]^{-1}$$

$$\cdot \left[\mathbf{M}^T \underline{y} + \mu \sum_{ij} \mathbf{R}_{ij}^T \mathbf{D} \underline{\alpha}_{ij} \right]$$

which is again a simple averaging of patches



Inpainting [Mairal, E. & Sapiro ('08)]

For the Peppers image

Alg.	RMSE for 25% missing	RMSE for 50% missing	RMSE for 75% missing
No-overlap	14.55	19.61	29.70
Overlap	9.00	11.55	18.18
K-SVD	8.1	10.05	17.74

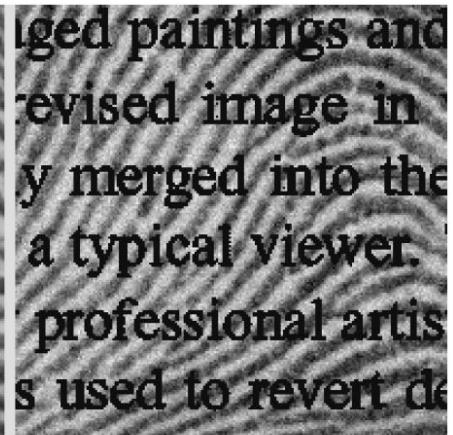
This is a more challenging case, where the DCT is not a suitable dictionary.

- For Redundant DCT we get RMSE=16.13, and
- For K-SVD (15 iterations) we get RMSE=12.74

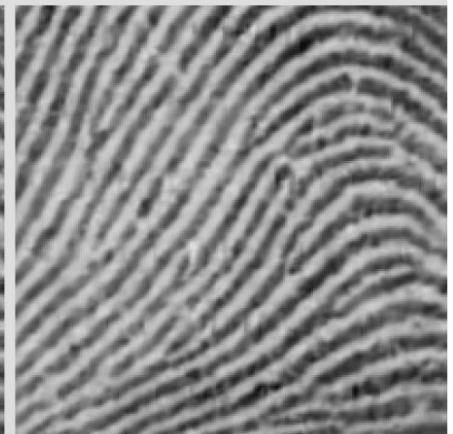
Original Image



Masked Image



DCT Result



K-SVD Result

Super-Resolution [Zeyde, Protter, & Elad (11)]

- Given a low-resolution image, we desire to enlarge it while producing a sharp looking result. This problem is referred to as “Single-Image Super-Resolution”.
- Image scale-up using bicubic interpolation is far from being satisfactory for this task.
- Recently, a sparse and redundant representation technique was proposed [Yang, Wright, Huang, and Ma ('08)] for solving this problem, by training a coupled-dictionaries for the low- and high res. images.
- We extended and improved their algorithms and results.



Super-Resolution – Results (1)

This book is about *convex optimization*, a special class of mathematical optimization problems, which includes least-squares and linear programming problems. It is well known that least-squares and linear programming problems have a fairly complete theory, arise in a variety of applications, and can be solved numerically very efficiently. The basic point of this book is that the same can be said for the larger class of convex optimization problems.

While the mathematics of convex optimization has been studied for about a century, several related recent developments have stimulated new interest in the topic. The first is the recognition that interior-point methods, developed in the 1980s to solve linear programming problems, can be used to solve convex optimization problems as well. These new methods allow us to solve certain new classes of convex optimization problems, such as semidefinite programs and second-order cone programs, almost as easily as linear programs.

The second development is the discovery that convex optimization problems (beyond least-squares and linear programs) are more prevalent in practice than was previously thought. Since 1990 many applications have been discovered in areas such as automatic control systems, estimation and signal processing, communications and networks, electronic circuit design, data analysis and modeling, statistics, and finance. Convex optimization has also found wide application in combinatorial optimization and global optimization, where it is used to find bounds on the optimal value, as well as approximate solutions. We believe that many other applications of convex optimization are still waiting to be discovered.

There are great advantages to recognizing or formulating a problem as a convex optimization problem. The most basic advantage is that the problem can then be solved, very reliably and efficiently, using interior-point methods or other special methods for convex optimization. These solution methods are reliable enough to be embedded in a computer-aided design or analysis tool, or even a real-time reactive or automatic control system. There are also theoretical or conceptual advantages of formulating a problem as a convex optimization problem. The associated dual

Ideal Image

The training set consists of 717 images, providing 289 training image-pairs.

An amazing variety of practical problems (design, analysis, and operation) can be cast as convex optimization problems, or some variation such as integer programming. Indeed, mathematical optimization has been used in a variety of applications. It is widely used in engineering, in electrical and aerospace engineering. Optimization is used in design and operation, finance, supply chain management, and other areas. The list of applications is still growing.

For most of these applications, mathematical optimization is used by a human decision maker, system designer, or engineer. This human decision maker processes, checks the results, and modifies the optimization problem when necessary. This human decision maker may be the optimization problem, e.g., buying a portfolio.

An amazing variety of practical problems (design, analysis, and operation) can be cast as convex optimization problems, or some variation such as integer programming. Indeed, mathematical optimization has been used in a variety of applications. It is widely used in engineering, in electrical and aerospace engineering. Optimization is used in design and operation, finance, supply chain management, and other areas. The list of applications is still growing.

Given Image

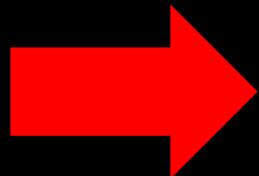
FORK-IT.UGG portfolio.



Super-Resolution – Results (2)



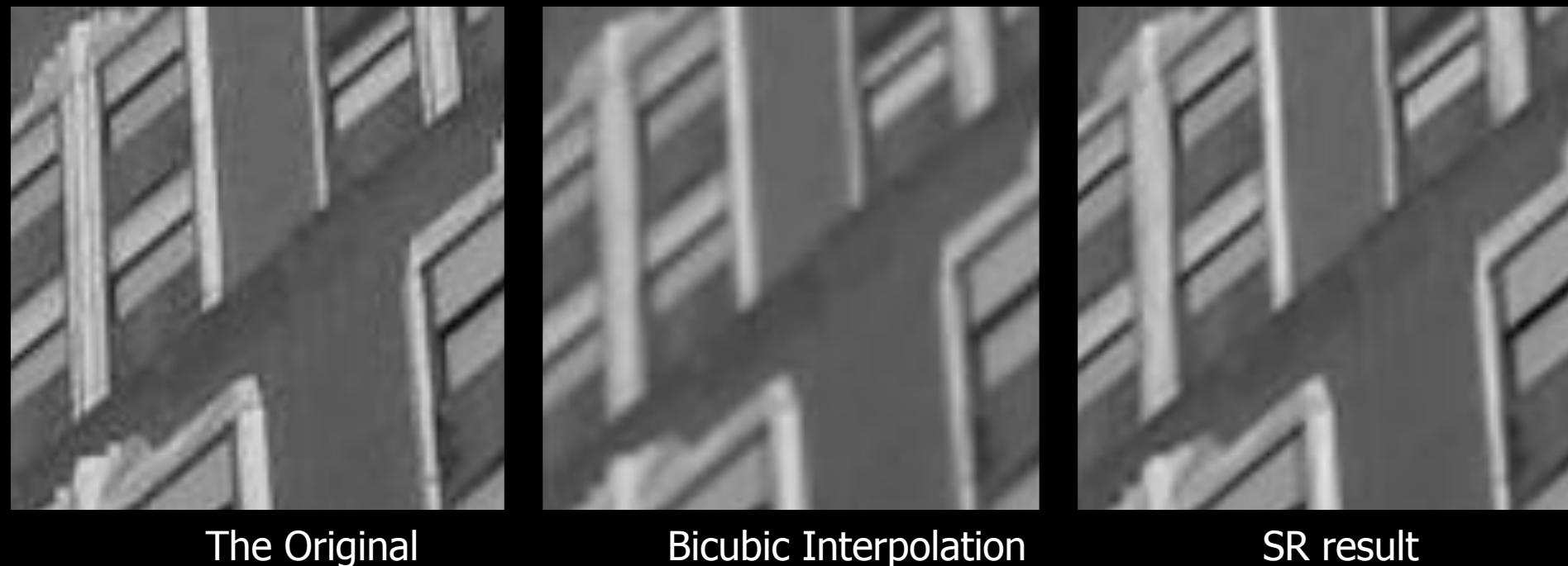
Given image



Scaled-Up (factor 2:1) using the proposed algorithm,
PSNR=29.32dB (3.32dB improvement over bicubic)



Super-Resolution – Results (2)



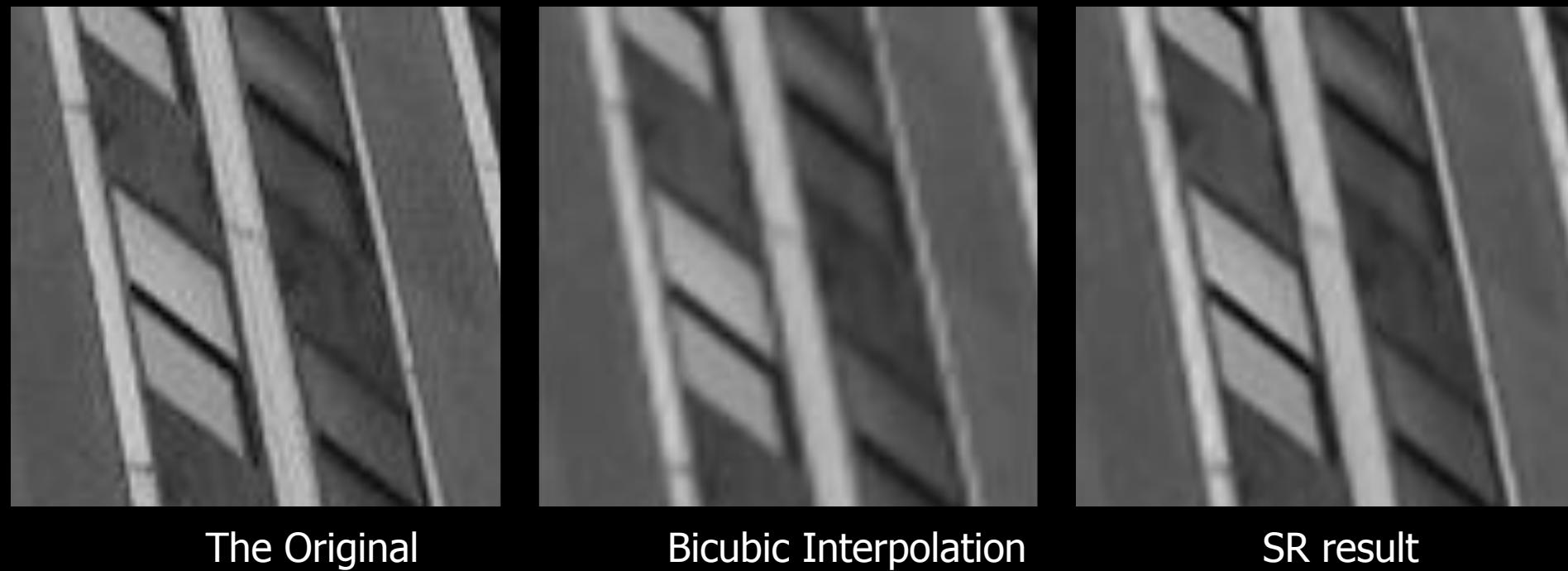
The Original

Bicubic Interpolation

SR result



Super-Resolution – Results (2)



The Original

Bicubic Interpolation

SR result

