Image Denoising and Beyond via Learned Dictionaries and Sparse Representations*

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

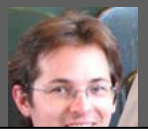
* Joint work with

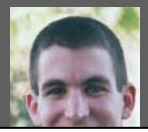

M. Aharon

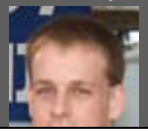

A.M. Bruckstein

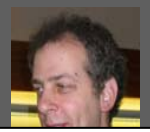

O. Bryt


D.L. Donoho









Tel-Aviv University
Applied Math Department
Approximation Seminar
June 26, 2008

Noise Removal?

Our story begins with image denoising …

- Important: (i) Practical application; (ii) A convenient platform (being the simplest inverse problem) for testing basic ideas in image processing, and then generalizing to more complex problems.
- Many Considered Directions: Partial differential equations, Statistical estimators, Adaptive filters, Inverse problems & regularization, Wavelets, Example-based techniques, Sparse representations, …

Agenda

Part I – Denoising by Sparse & Redundant Representations

Part II – Theoretical & Numerical Foundations

Part III – Dictionary Learning & The K-SVD Algorithm

Part IV – Back to Denoising … and Beyond

Part V – Interesting & Recent Advancement

Part VI – Summary & Conclusions

Our goal today is to show that

- A signal/image model based on sparsity & redundancy is important and well-founded.
- This model can be used in designing very effective tools in signal/image processing.

Part I
Denoising by Sparse & Redundant Representations

Denoising By Energy Minimization

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

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

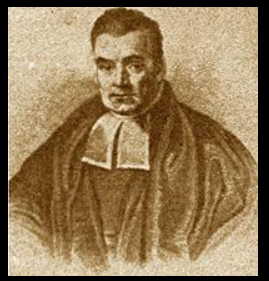
Relation to measurements

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 Pr($\underline{x}$)

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

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

Energy

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

Smoothness

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

Adapt+ Smooth

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

Robust Statistics

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

Total-Variation

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

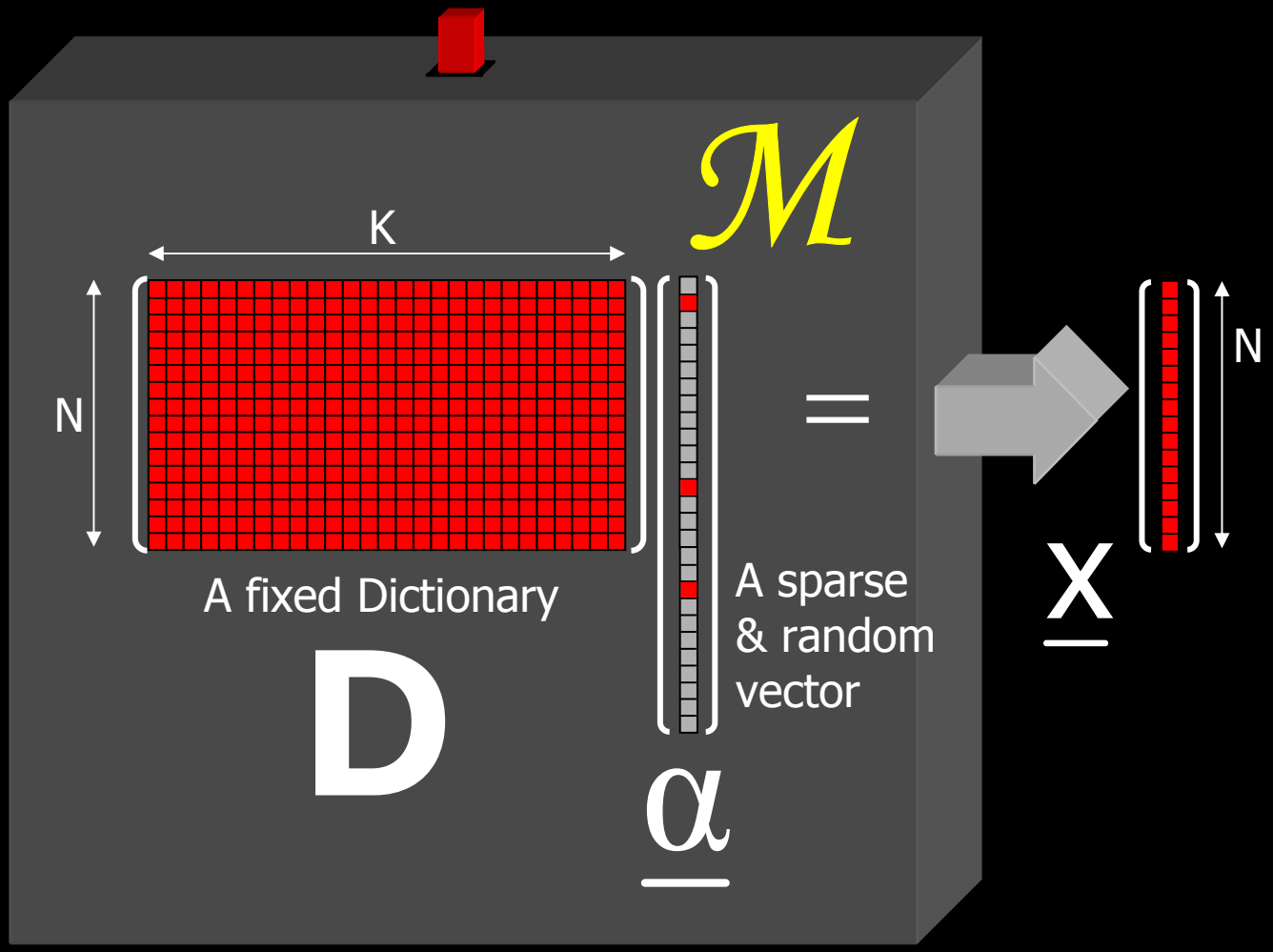
Wavelet Sparsity

$$\Pr(\underline{x}) = \lambda \|\underline{\alpha}\|_0^0 \quad \text{for} \quad \underline{x} = \mathbf{D}\underline{\alpha}$$

Sparse & Redundant

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

Sparse Modeling of Images



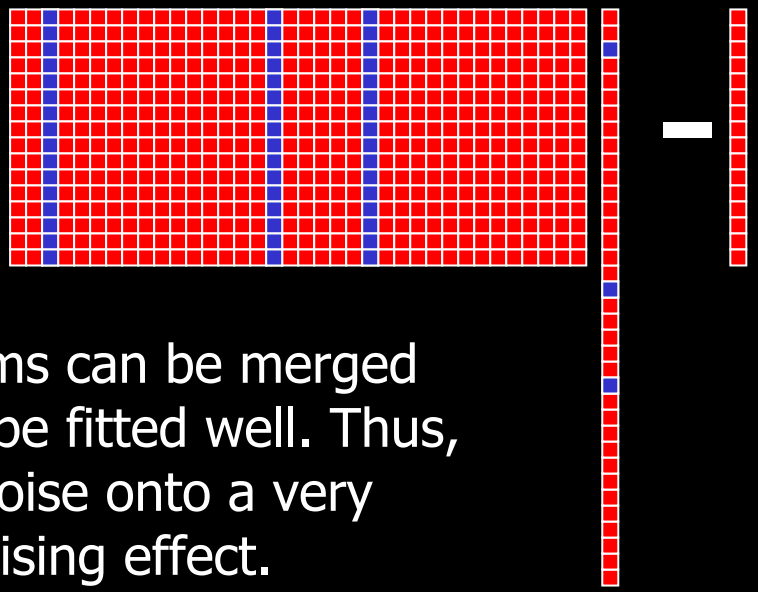
- Every column in **D** (dictionary) is a prototype signal (atom).
- The vector $\underline{\alpha}$ is generated randomly with few (say L) non-zeros at random locations and with random values.

Back to Our MAP Energy Function

- We L_0 norm is effectively counting the number of non-zeros in $\underline{\alpha}$.
- The vector $\underline{\alpha}$ is the representation (**sparse**/**redundant**).

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

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



- 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.

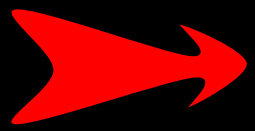
There are Some Issues ...

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

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

- **Theoretical Problem:** If we are to approximate the solution somehow, how close will we get?

- **Practical Problem:** What dictionary **D** should we use, such that all this leads to effective denoising?



These are the topics of the next 2 parts.

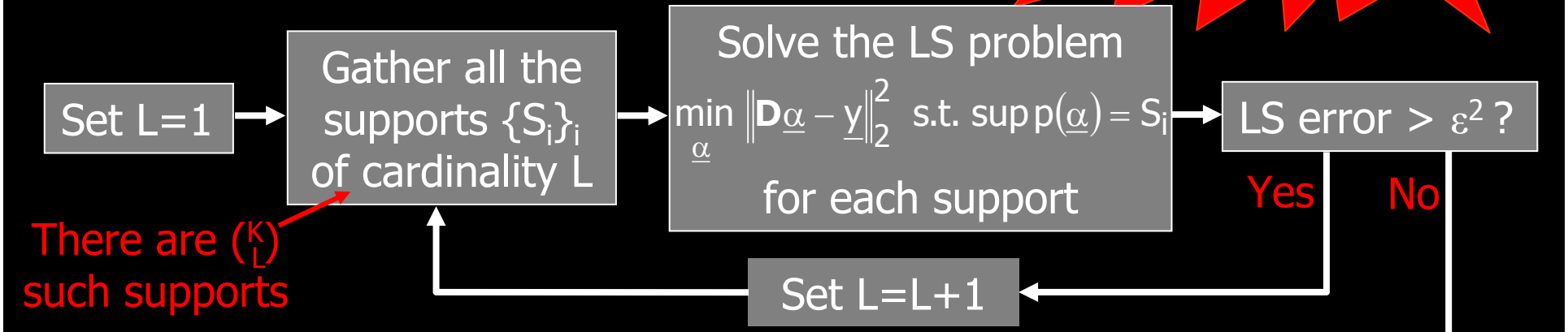
Part II
Theoretical & Numerical Foundations

Our Goal

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

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

Here is a recipe for solving this problem:



Assume: K=2000, L=10 (known!), 1 nano-sec per each LS

➡ We shall need ~8e+9 years to solve this problem !!!!!

Lets Approximate

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

Relaxation methods

Smooth the L_0 and use continuous optimization techniques

Greedy methods

Build the solution one non-zero element at a time

Relaxation: Basis Pursuit (BP)

Instead of solving

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

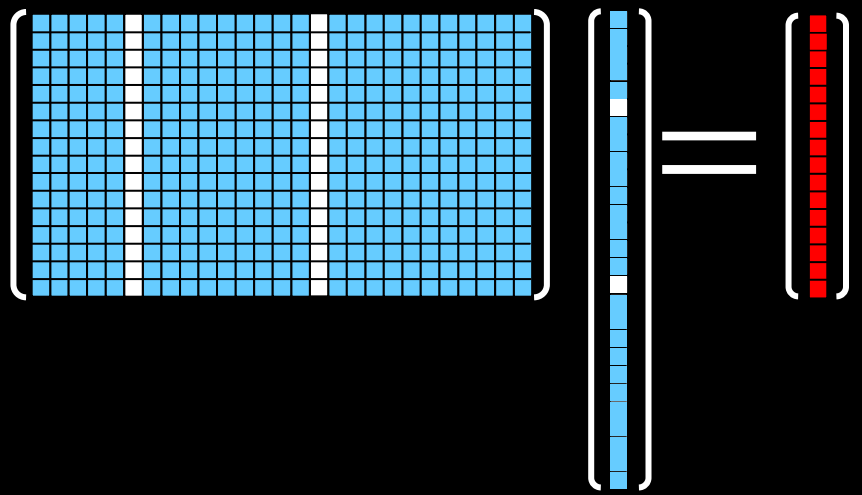
Solve Instead

$$\min_{\underline{\alpha}} \|\underline{\alpha}\|_1 \quad \text{s.t.} \quad \|\mathbf{D}\underline{\alpha} - \underline{y}\|_2 \le \varepsilon$$

- 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)].
  - Sequential shrinkage for union of ortho-bases [Bruce et.al. ('98)].
  - Iterated shrinkage [Figuerido & Nowak ('03)] [Daubechies, Defrise, & Demole ('04)] [Elad ('05)] [Elad, Matalon, & Zibulevsky ('06)].

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 rsidual.



- 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.

Equivalence For $\text{Min}\|\underline{\alpha}\|_0$ s.t. $\mathbf{D}\underline{\alpha} = \underline{y}$

Equivalence

[Donoho & Elad ('02)]
[Gribonval & Nielsen ('03)]
[Tropp ('03)]
[Temlyakov ('03)]

Given a signal $\underline{y}$ with a representation $\underline{y} = \mathbf{D}\underline{\alpha}$, assuming that $\|\underline{\alpha}\|_0 < \text{Thr}\{\mathbf{D}\}$, BP and OMP are **Guaranteed** to find the sparsest solution.

- 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)] [Elad ('06)] [Tropp et.al. ('06)].

BP Stability for $\text{Min}\|\underline{\alpha}\|_0$ s.t. $\|\mathbf{D}\underline{\alpha}-\underline{y}\|_2 \le \varepsilon$

Stability 1

Given a signal $\underline{y}=\mathbf{D}\underline{\alpha}+\underline{v}$ with a representation satisfying $\|\underline{\alpha}\|_0 < 0.5\text{Thr}\{\mathbf{D}\}$ and bounded noise $\|\underline{v}\|_2 \le \varepsilon$, BP will give stability, i.e.,

$$\left\|\underline{\hat{\alpha}}_{BP}-\underline{\alpha}\right\|_2^2 < \text{Const}_1\{\mathbf{D}\}\cdot\varepsilon^2$$

[Donoho, Elad & Temlyakov ('06)] [Tropp ('06)] [Donoho & Elad ('07)]

- For ε=0 we get a weaker version of the previous result.
- Surprising - the error is independent of the SNR.
- This result is useless for assessing denoising performance.
- Worst case versus average performance [Candes et. al. ('07)] [Donoho ('07)].

OMP Stability for $\text{Min}\|\underline{\alpha}\|_0$ s.t. $\|\mathbf{D}\underline{\alpha}-\underline{y}\|_2 \le \varepsilon$

Stability 2

Given a signal $\underline{y}=\mathbf{D}\underline{\alpha}+\underline{v}$ with bounded noise $\|\underline{v}\|_2 \le \varepsilon$, and a sparse representation,

$$\|\underline{\alpha}\|_0 < \text{Thr}\{\mathbf{D}\} - \frac{\varepsilon \cdot \text{Const}_2\{\mathbf{D}\}}{\min_k\{|\alpha(k)|\}}$$

OMP will give stability, i.e.,

$$\|\underline{\hat{\alpha}}_{MP}-\underline{\alpha}\|_2^2 < \text{Const}_3\{\mathbf{D}\}\cdot\varepsilon^2$$

[Donoho, Elad & Temlyakov ('06)] [Tropp ('06)]

- For ε=0 we get the results shown already.
- Here the error is dependent of the SNR, and
- There are additional results on the sparsity pattern recovery.

Part III
Dictionary Learning: The K-SVD Algorithm

What Should D Be?

$$\hat{\underline{\alpha}} = \arg\min_{\underline{\alpha}} \|\underline{\alpha}\|_0^0 \quad \text{s.t.} \quad \frac{1}{2}\|\mathbf{D}\underline{\alpha} - \underline{y}\|_2^2 \le \varepsilon^2 \Rightarrow \hat{\underline{x}} = \mathbf{D}\hat{\underline{\alpha}}$$

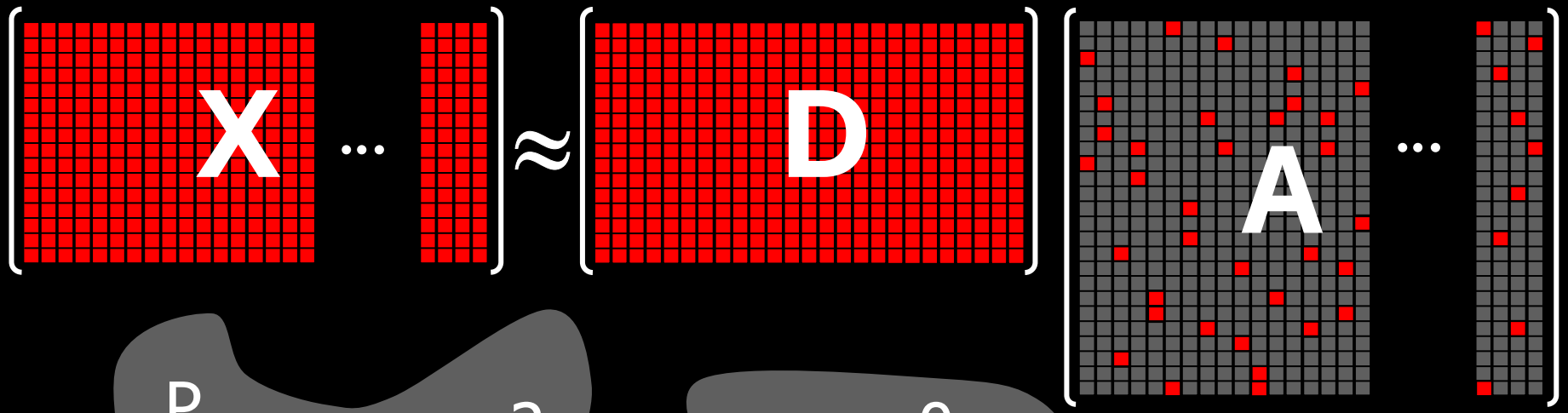
Our Assumption: Good-behaved Images have a sparse representation

D should be chosen such that it sparsifies the representations

One approach to choose **D** is from a known set of transforms (Steerable wavelet, Curvelet, Contourlets, Bandlets, …)

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

Measure of Quality for D



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

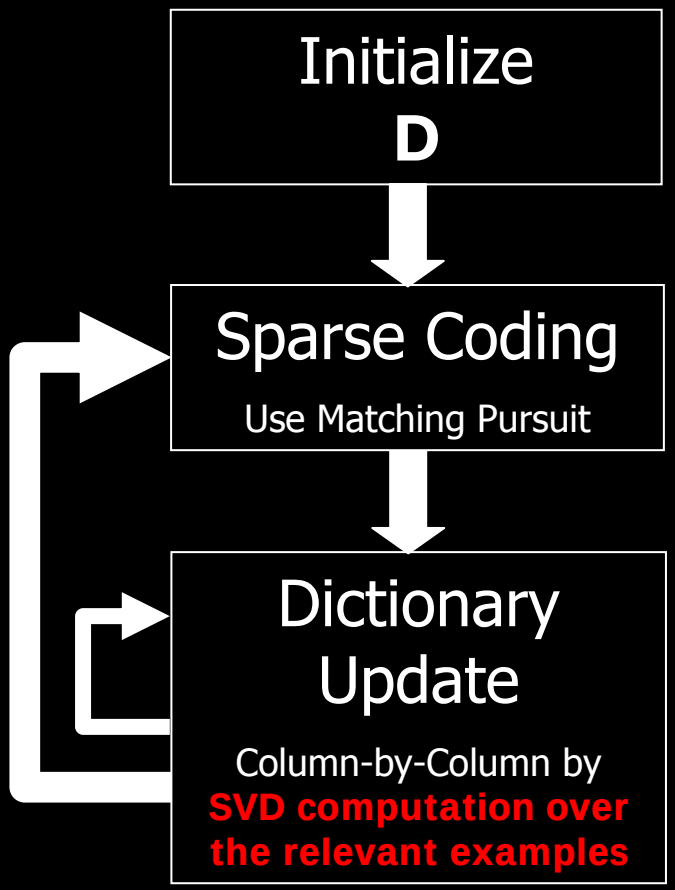
Each example is a linear combination of atoms from **D**

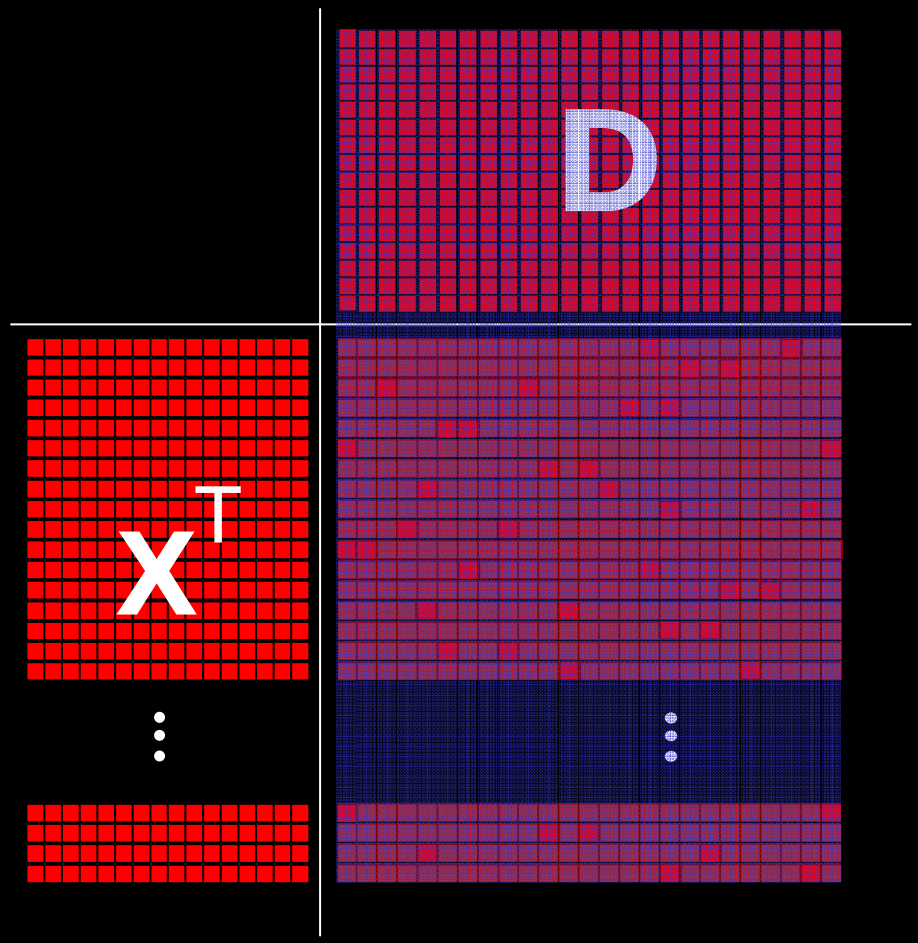
Each example has a sparse representation with no more than L atoms

[Field & Olshausen ('96)]
[Engan et. al. ('99)]
[Lewicki & Sejnowski ('00)]
[Cotter et. al. ('03)]
[Gribonval et. al. ('04)]
[Aharon, Elad & Bruckstein ('04)]
[Aharon, Elad & Bruckstein ('05)]

The K–SVD Algorithm – General

[Aharon, Elad & Bruckstein ('04,'05)]





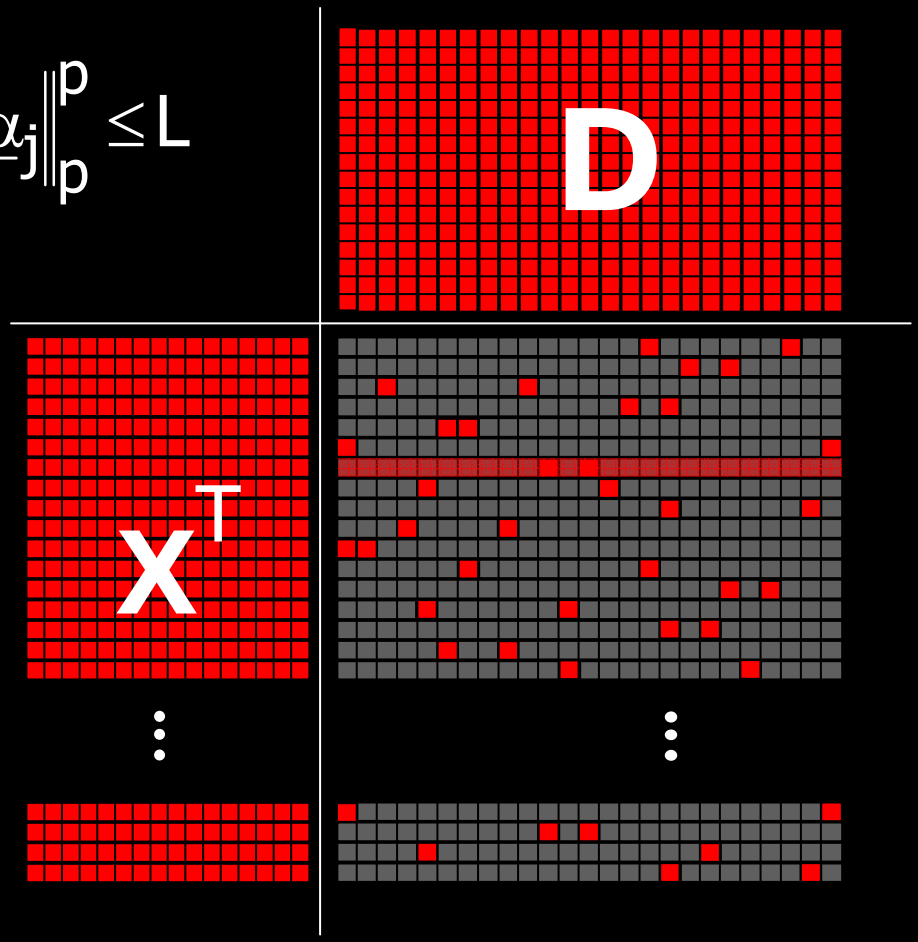
K–SVD: Sparse Coding Stage

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

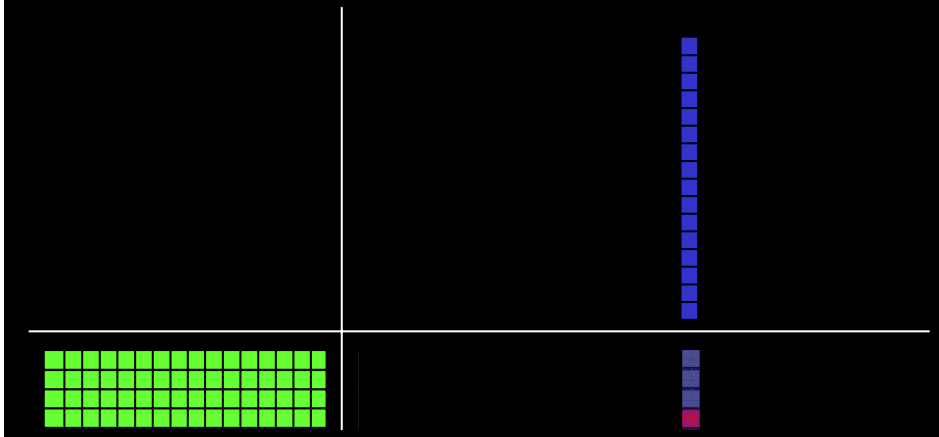
D is known!
For the j^{th} item we solve

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

Solved by
A Pursuit Algorithm



K–SVD: Dictionary Update Stage



We should solve:

$$\min_{\underline{d}_k, \alpha_k} \left\| \alpha_k \underline{d}_k^T - \mathbf{E} \right\|_F^2$$

SVD

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

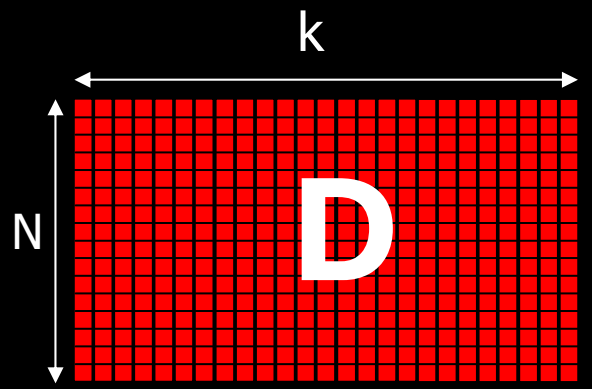
Fixing all **A** and **D** apart from the k^{th} column, and seek both $\underline{d}_k$ and the k^{th} column in **A** to better fit the **residual**!

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 [Roth & Black ('05)].

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

Extracts a patch in the ij location

Our prior

What Data to Train On?

Option 1:

- Use a database of images,
- We tried that, and it works fine (~0.5-1dB 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 ➡ $\sim 10^6$ examples to use – more than enough.
- This works much better!

Application 2: Image Denoising

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

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

Compute α_{ij} per patch

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

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

using the matching pursuit

$\underline{x}$ and α_{ij} known

Compute **D** to minimize

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

using SVD, updating one column at a time

D and α_{ij} known

Compute $\underline{x}$ by

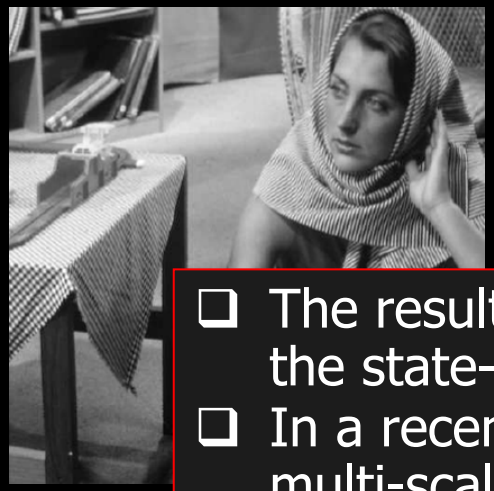
$$\underline{x} = \left[\mathrm{I}+\mu\sum_{ij}\mathbf{R}_{ij}^T\mathbf{R}_{ij}\right]^{-1}\left[\underline{y}+\mu\sum_{ij}\mathbf{R}_{ij}^T\mathbf{D}\underline{\alpha}_{ij}\right]$$

which is a simple averaging of shifted patches

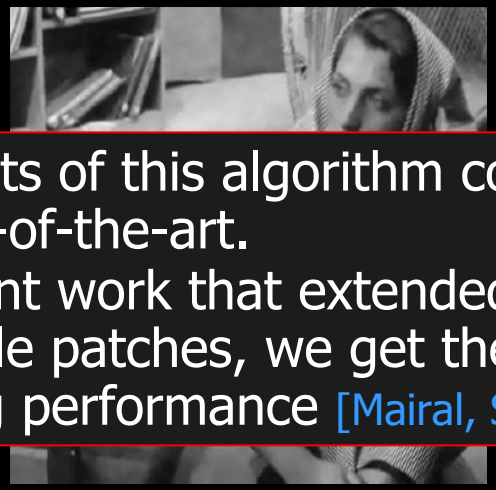
K-SVD

Complexity of this algorithm: $O(N^2\times K\times L$ flops

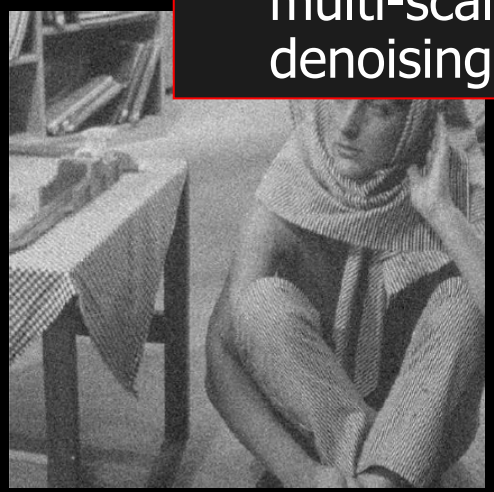
Image Denoising (Gray) [Elad & Aharon ('06)]



Source



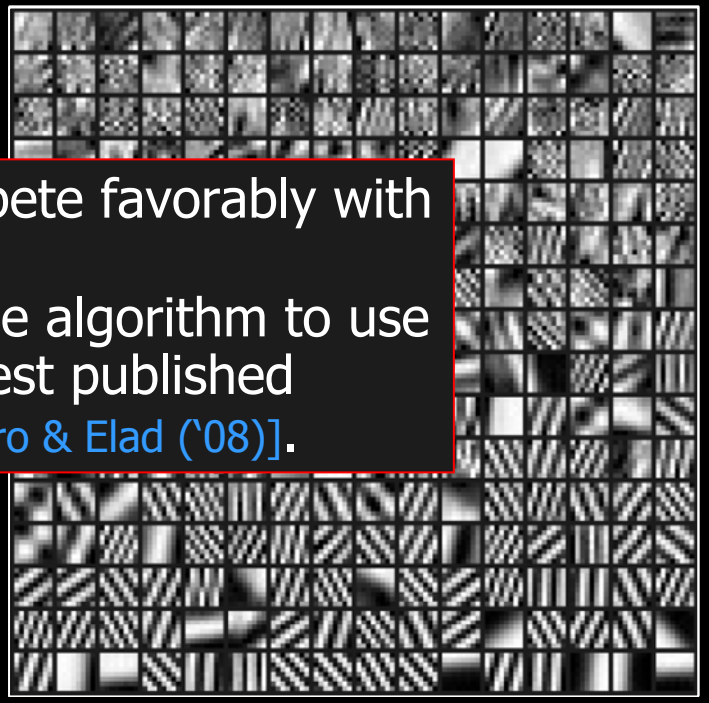
Result 30.829dB



Noisy image

$\sigma = 20$

- The results of this algorithm compete favorably with the state-of-the-art.
- In a recent work that extended the algorithm to use multi-scale patches, we get the best published denoising performance [Mairal, Sapiro & Elad ('08)].

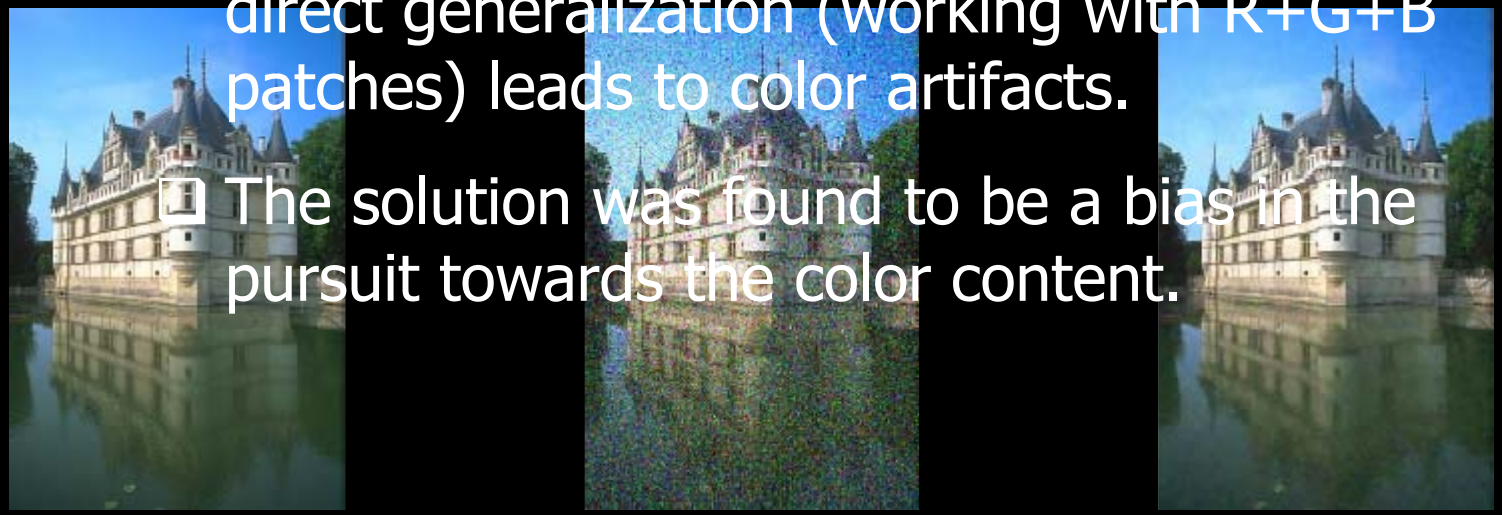


The obtained dictionary after 10 iterations

Denoising (Color) [Mairal, Elad & Sapiro ('06)]

Our experiments lead to state-of-the-art denoising results, giving ~1dB better results compared to [Mcauley et. al. ('06)] which implements a learned MRF model (Field-of-Experts)

- When turning to handle color images, the direct generalization (working with R+G+B patches) leads to color artifacts.
- The solution was found to be a bias in the pursuit towards the color content.



Original Noisy (12.77dB) Result (29.87dB)

Inpainting [Mairal, Elad & Sapiro ('06)]

Our experiments lead to state-of-the-art inpainting results.

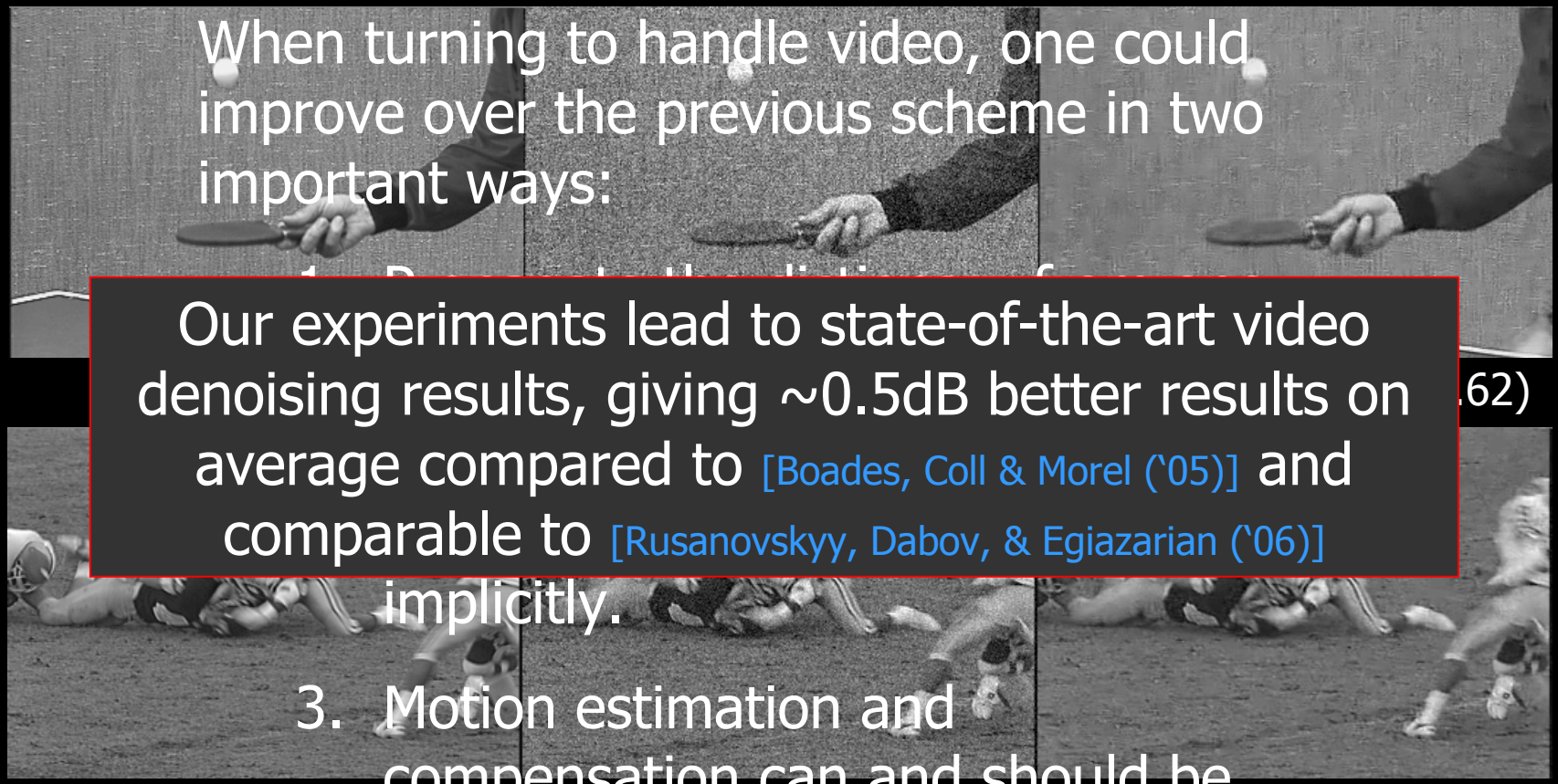
Video Denoising [Protter & Elad ('06)]

When turning to handle video, one could improve over the previous scheme in two important ways:

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

implicitly.

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



.62)

Original Noisy (σ=15) Denoised (PSNR=29.98)

Image Compression [Bryt and Elad ('06)]

- 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, & Elad ('05)].

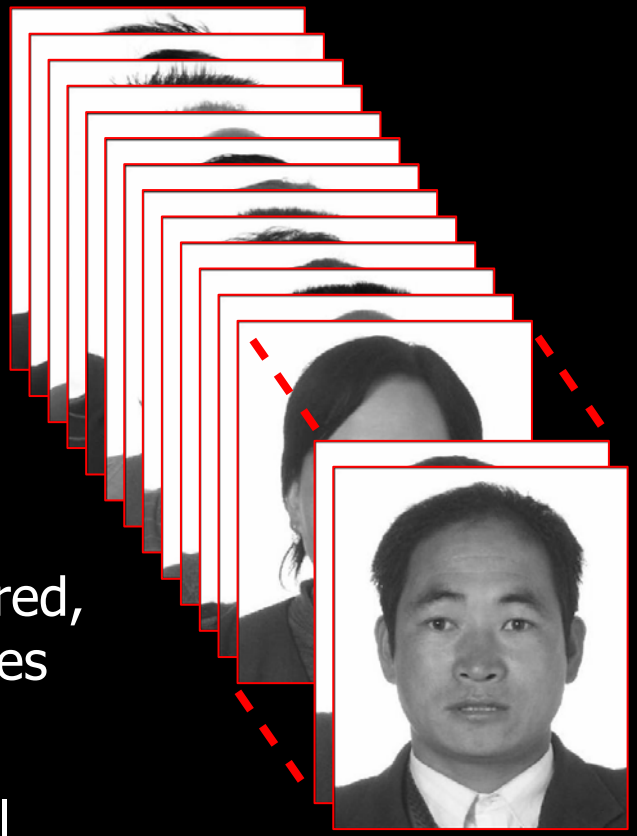


Image Compression Results

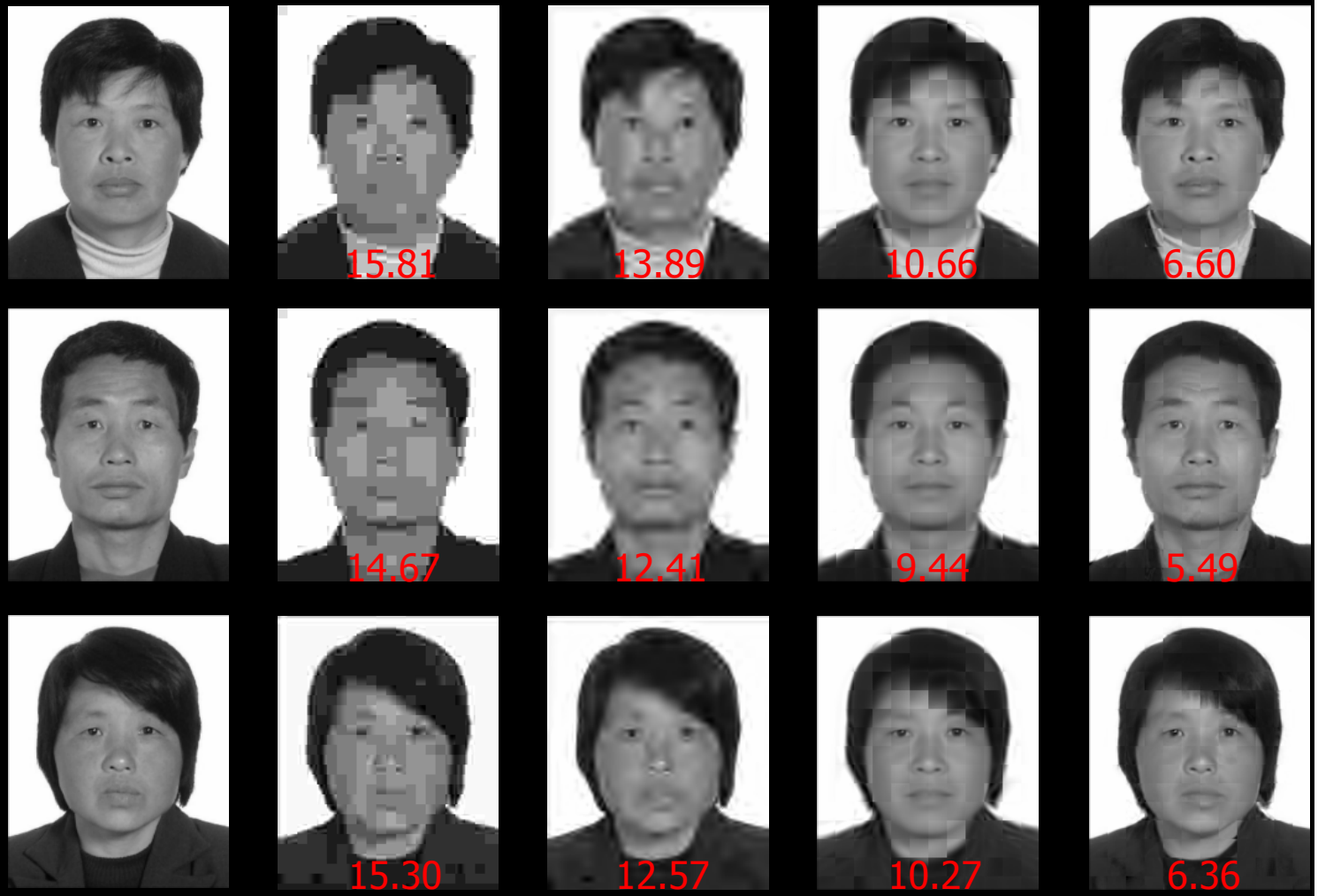


Image Compression Results

Original

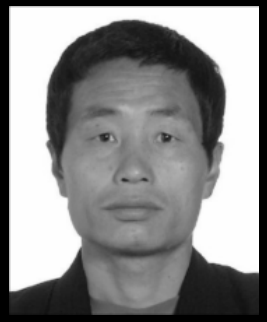
JPEG

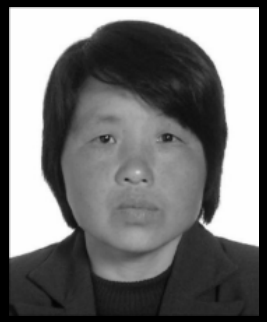
JPEG-2000

Local-PCA

K-SVD

Results for **400** Bytes per each file


Part V
Interesting & Recent Advancement

What If …

Consider the denoising problem

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

and suppose that we can find a group of J candidate solutions

$$\left\{\underline{\alpha}_j\right\}_{j=1}^{J}$$

such that

$$\forall j \quad \left\{ \begin{array}{c} \left\|\underline{\alpha}_j\right\|_0^0 << N \\ \left\|\mathbf{D}\underline{\alpha}_j - \underline{y}\right\|_2^2 \le \varepsilon^2 \end{array} \right\}$$

Basic Questions:

- **What** could we do with such a set of competing solutions in order to better denoise $\underline{y}$?
- **Why** should this work?
- **How** shall we practically find such a set of solutions?

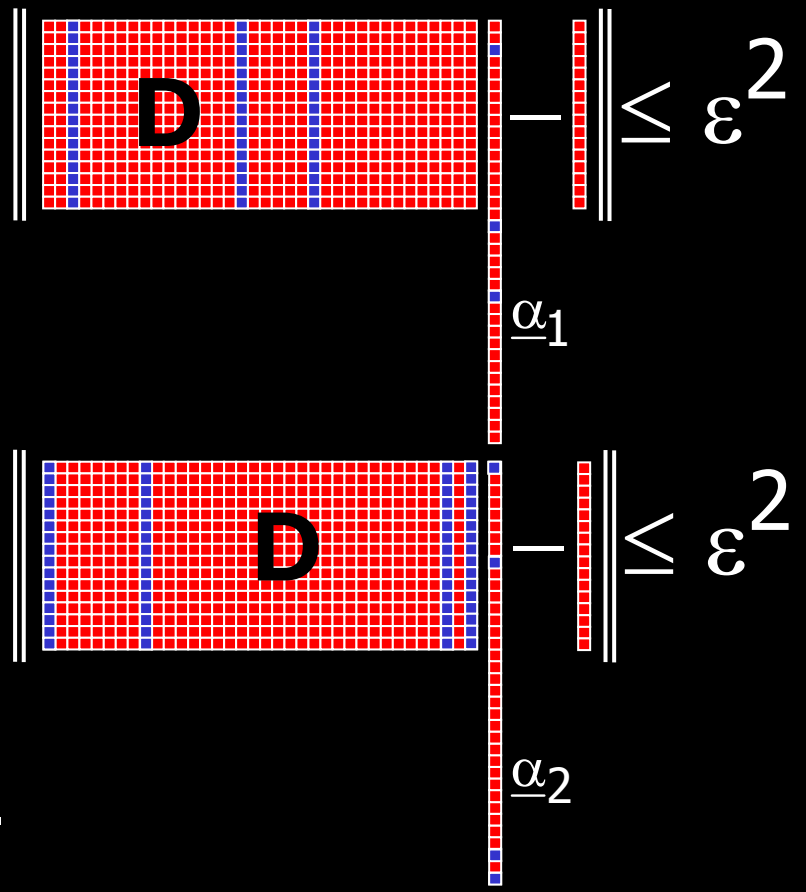
These questions were studied and answered recently

[Elad and Yavneh ('08)]

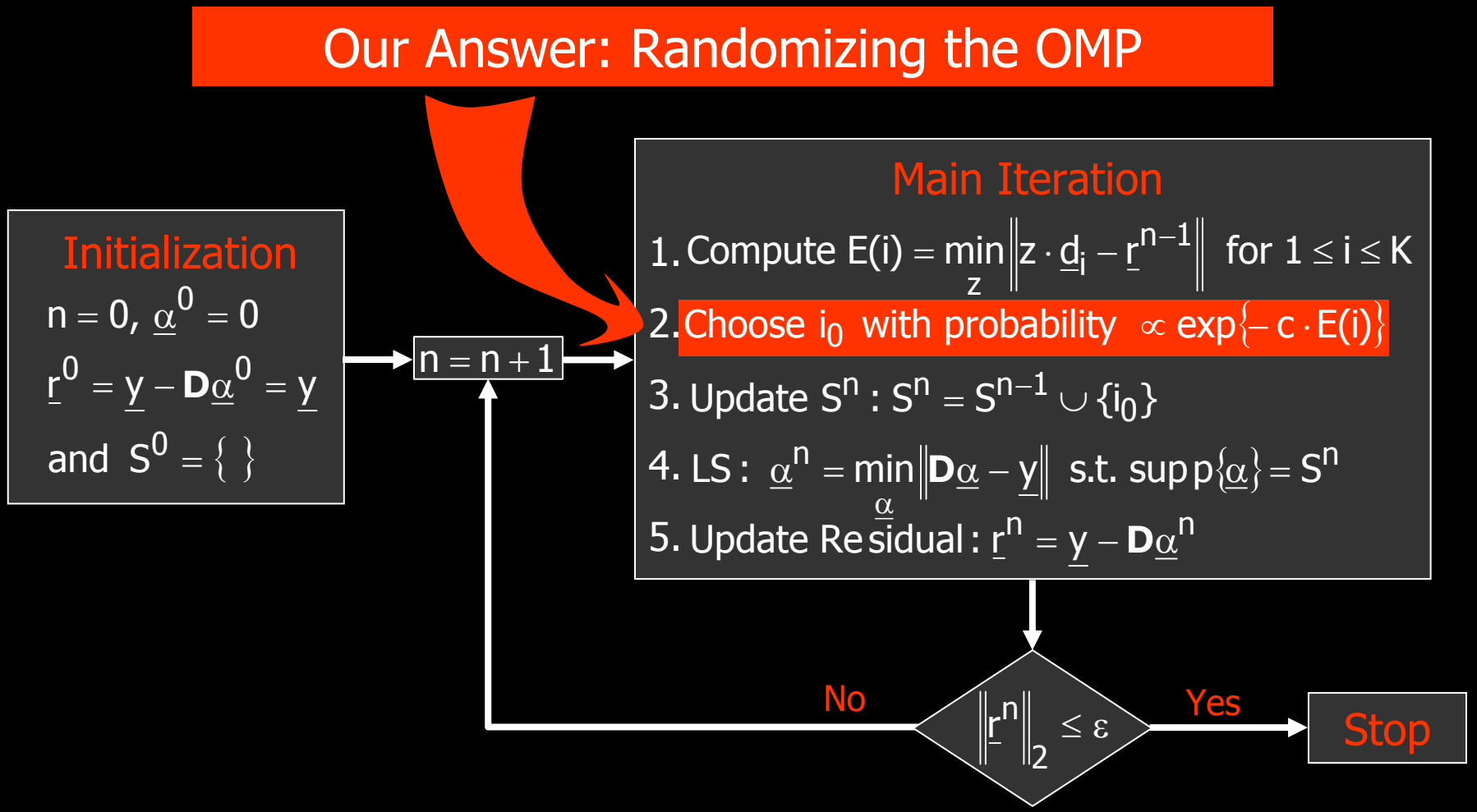
Motivation

Why bother with such a set?

- Because of the intriguing relation to example-based techniques, where several nearest-neighbors for the signal are used jointly.
- Because each representation conveys a different story about the desired signal.
- Because pursuit algorithms are often wrong in finding the sparsest representation, and then relying on their solution becomes too sensitive.
- … Maybe there are "deeper" reasons?



Generating Many Representations



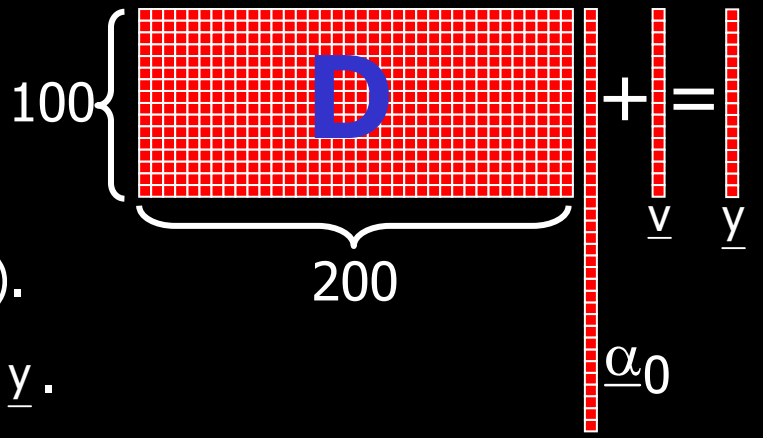
Lets Try

Proposed Experiment :

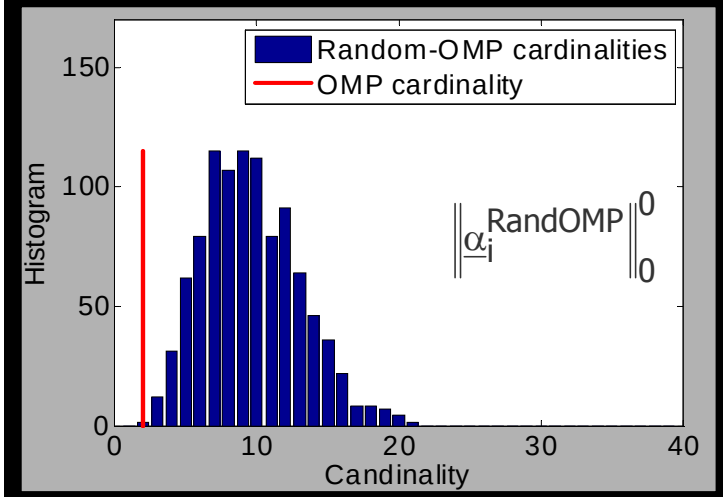
- Form a random **D**.
- Multiply by a sparse vector $\underline{\alpha}_0$ ($\left\|\underline{\alpha}_0\right\|_0^0 = 10$).
- Add Gaussian iid noise (σ=1) and obtain $\underline{y}$.
- Solve the problem

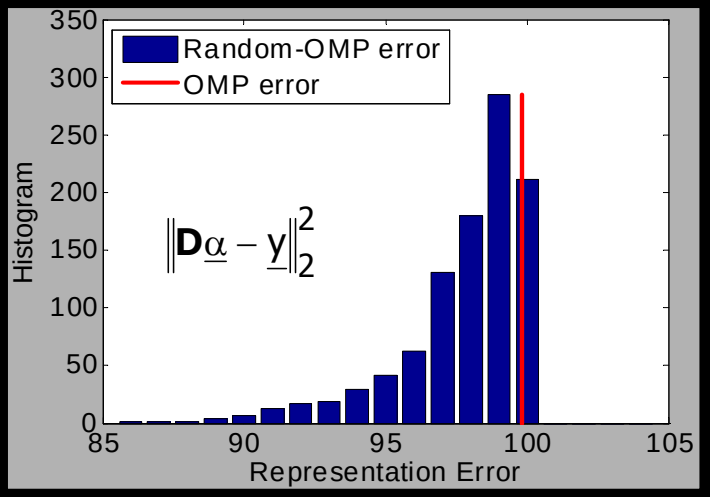
$$\min_{\underline{\alpha}} \left\|\underline{\alpha}\right\|_0^0 \quad \text{s.t.} \quad \left\|\mathbf{D}\underline{\alpha} - \underline{y}\right\|_2^2 \le 100$$

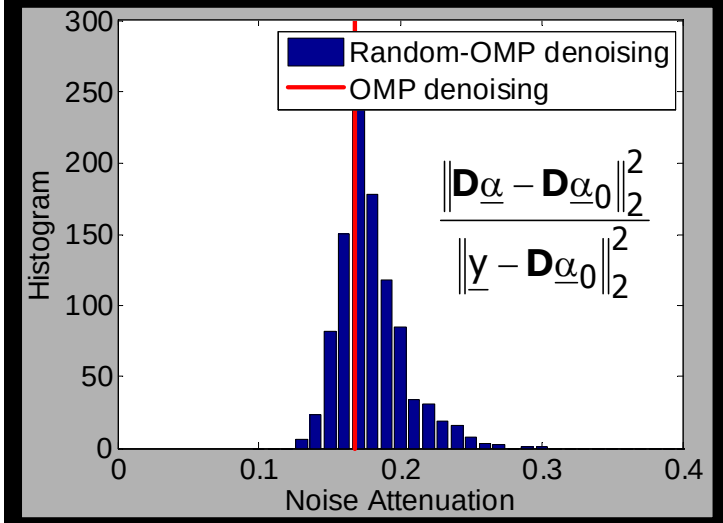
  using OMP, and obtain $\underline{\alpha}^{\text{OMP}}$.
- Use RandOMP and obtain $\left\{\underline{\alpha}_j^{\text{RandOMP}}\right\}_{j=1}^{1000}$.
- Lets look at the obtained representations …

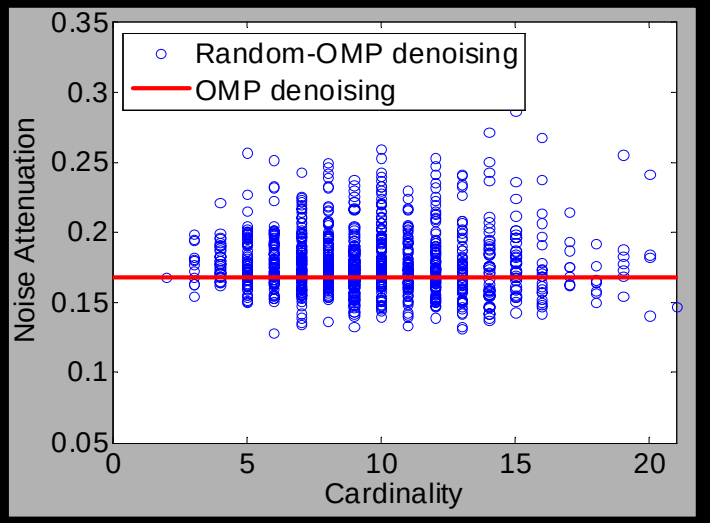


Some Observations









We see that

- The OMP gives the sparsest solution
- Nevertheless, it is not the most effective for denoising.
- The cardinality of a representation does not reveal its efficiency.

The Surprise ...

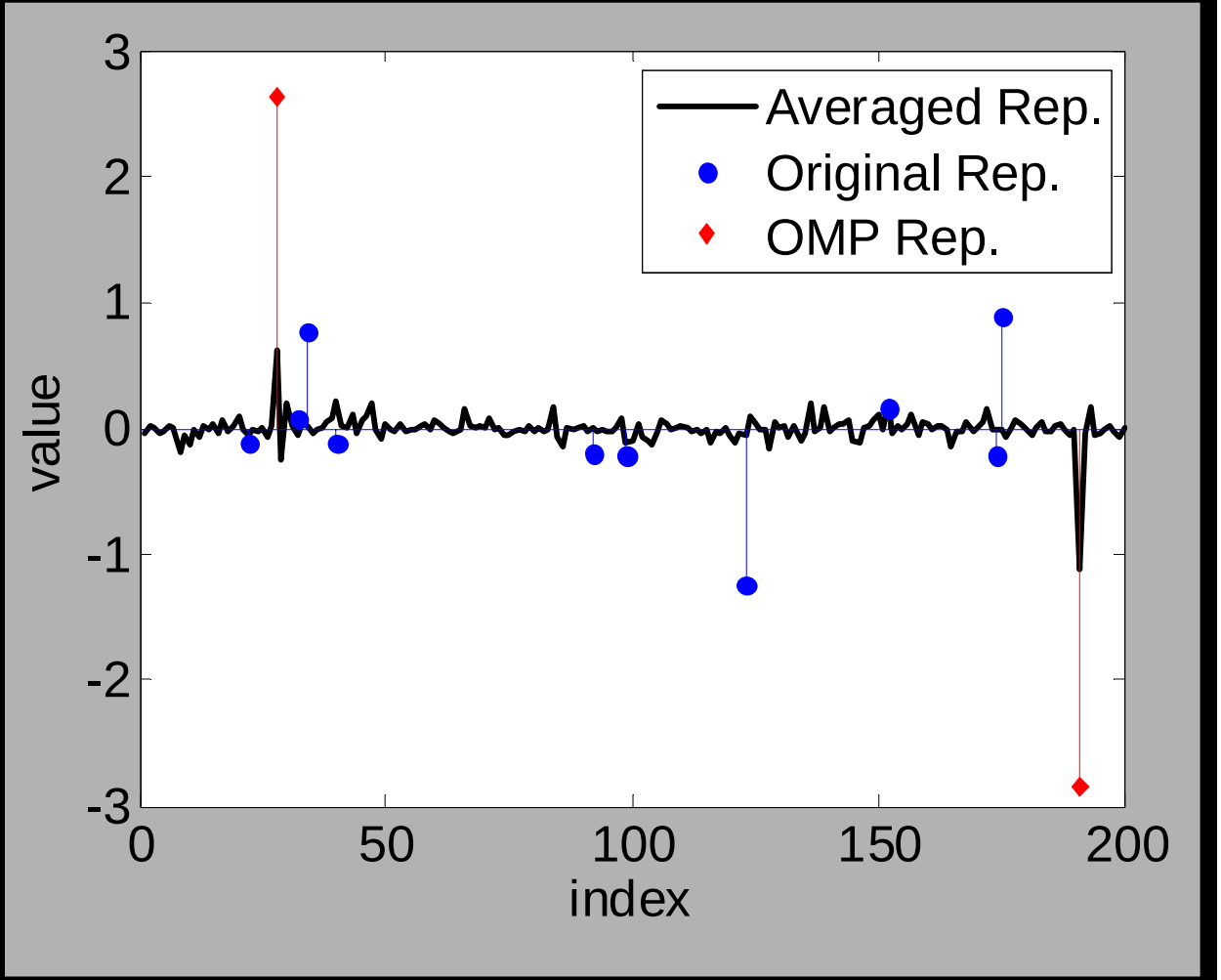
Lets propose the average

$$\hat{\underline{\alpha}} = \frac{1}{1000}\sum_{j=1}^{1000} \underline{\alpha}_j^{\text{RandOMP}}$$

as our representation

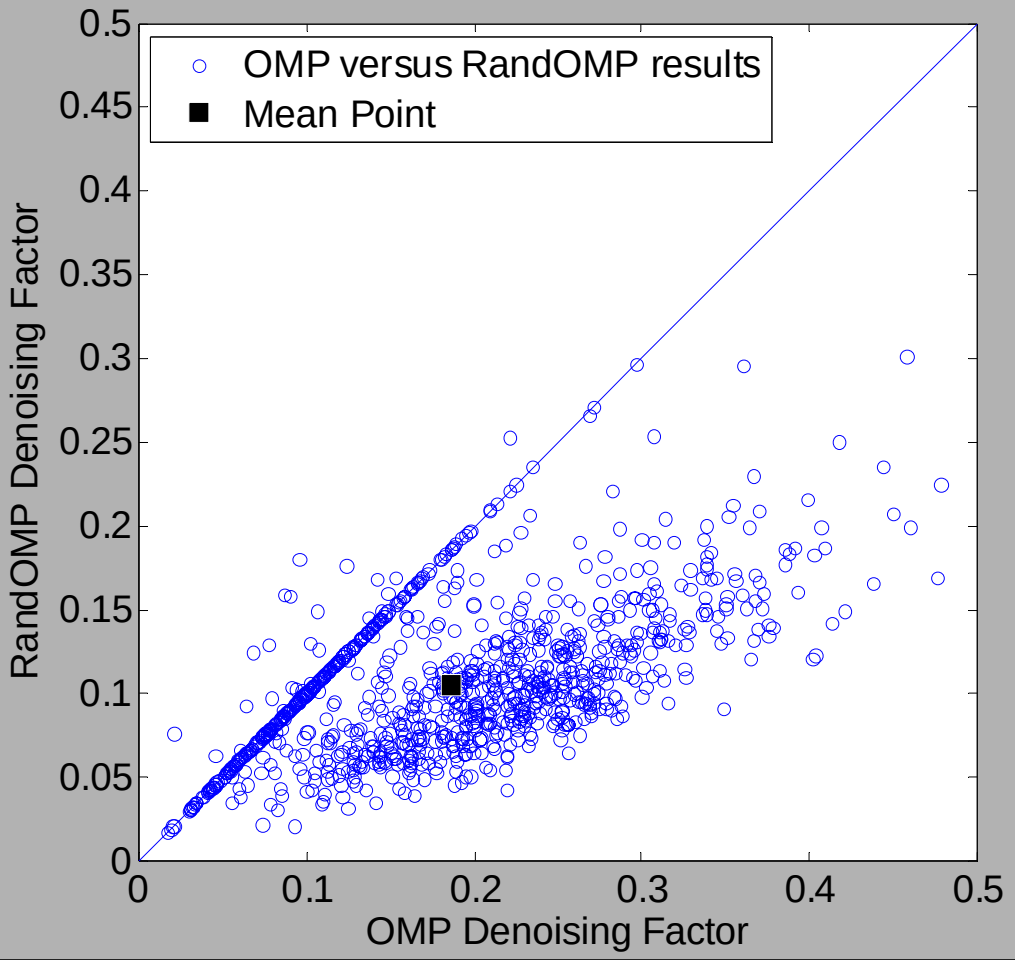
This representation IS NOT SPARSE AT ALL but its noise attenuation is:

0.06 (OMP gives 0.16)

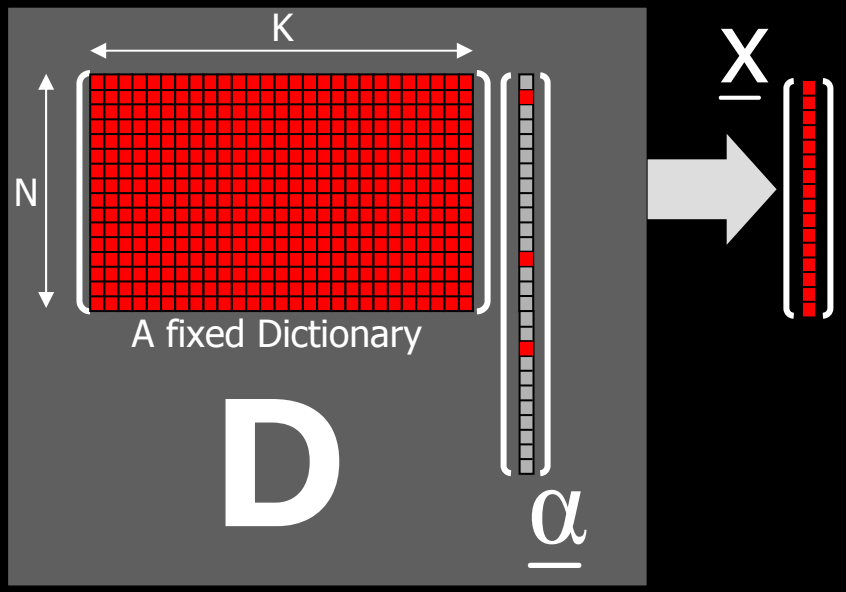


Is It Consistent? Yes!

Here are the results of 1000 trials with the same parameters …



The Explanation – Our Signal Model



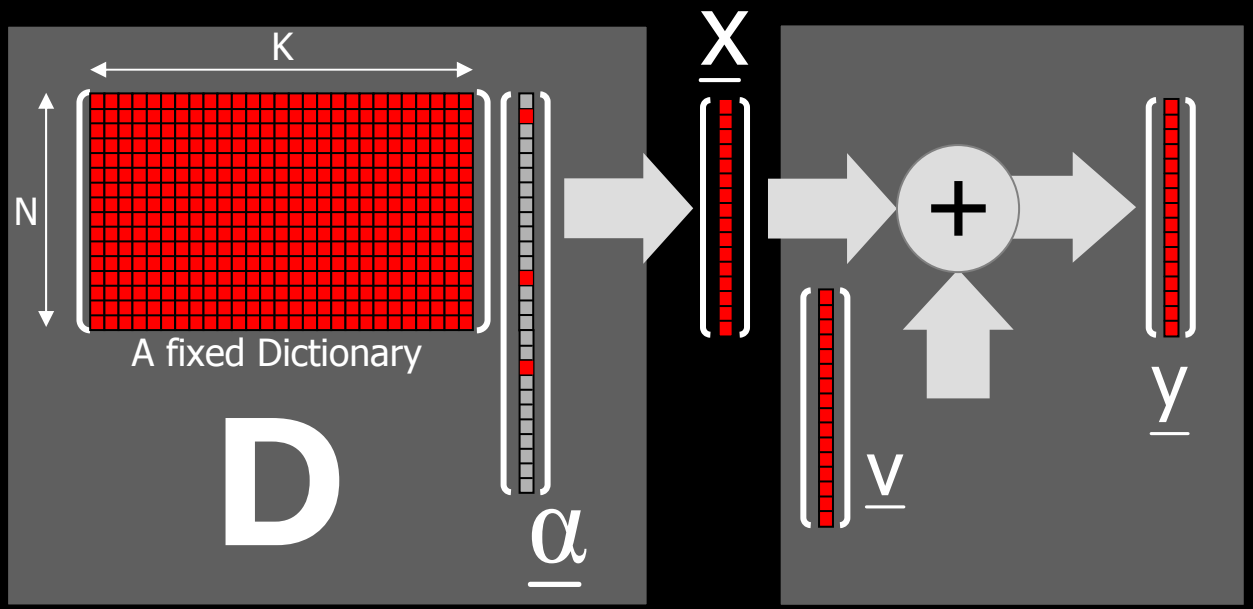
Signal Model Assumed

- **D** is fixed and known
- $\underline{\alpha}$ is built by:
  - Choosing the support S w.p. P(S) of all the 2^K possibilities Ω,
  - Choosing the coefficients using iid Gaussian entries* $N(0,\sigma_x)$: $P(\underline{\alpha}|S)$.
- The ideal signal is $\underline{x}=\mathbf{D}\underline{\alpha}$.

The p.d.f. of the signal $P(\underline{x})$ is: $P(\underline{x}) = \sum_{S\in\Omega} P(\underline{x}|S)P(S)$

* Not exactly, but this does not change our analysis.

The Explanation – Adding Noise



Noise Assumed:

The noise $\underline{v}$ is additive white Gaussian vector with probability $P_v(\underline{v})$

$$P(\underline{y}|\underline{x}) = C \cdot \exp\left\{-\frac{\|\underline{x}-\underline{y}\|^2}{2\sigma^2}\right\}$$

The p.d.f. of the noisy signal $P(\underline{y})$, and the conditionals $P(\underline{y}|S)$ and $P(S|\underline{y})$ are clear and well-defined (although nasty).

Some Algebra Leads To

$$\hat{\underline{x}}^{MMSE} = E\left\{\underline{x}\middle|\underline{y}\right\}$$

$$P\left(S\middle|\underline{y}\right) \propto \exp\left\{\frac{\sigma_x^2}{\sigma_x^2+\sigma^2}\cdot\frac{\left\|\underline{y}_S\right\|^2}{2\sigma^2}\right\}$$

Projection of the signal $\underline{y}$ onto the support S

$$\underline{y}_S = \mathbf{D}\cdot\left\{\underset{\underline{\alpha}}{\text{ArgMin}}\left\|\underline{y}-D\underline{\alpha}\right\| \text{ s.t. } \text{supp}\{\underline{\alpha}\}=S\right\}$$

Implications:

The best estimator (in terms of L_2 error) is a weighted average of many sparse representations!!!

As It Turns Out ...

- The MMSE estimation we got requires a sweep through all 2^K supports (i.e. combinatorial search) – impractical.
- Similarly, an explicit expression for $P(\underline{x}/\underline{y})$ can be derived and maximized – this is the MAP estimation, and it also requires a sweep through all possible supports – impractical too.
- The OMP is a (good) approximation for the MAP estimate.
- The RandOMP is a (good) approximation of the Minimum-Mean-Squared-Error (MMSE) estimate. It is close to the Gibbs sampler of the probability $P(S|\underline{y})P(S)$ from which we should draw the weights.

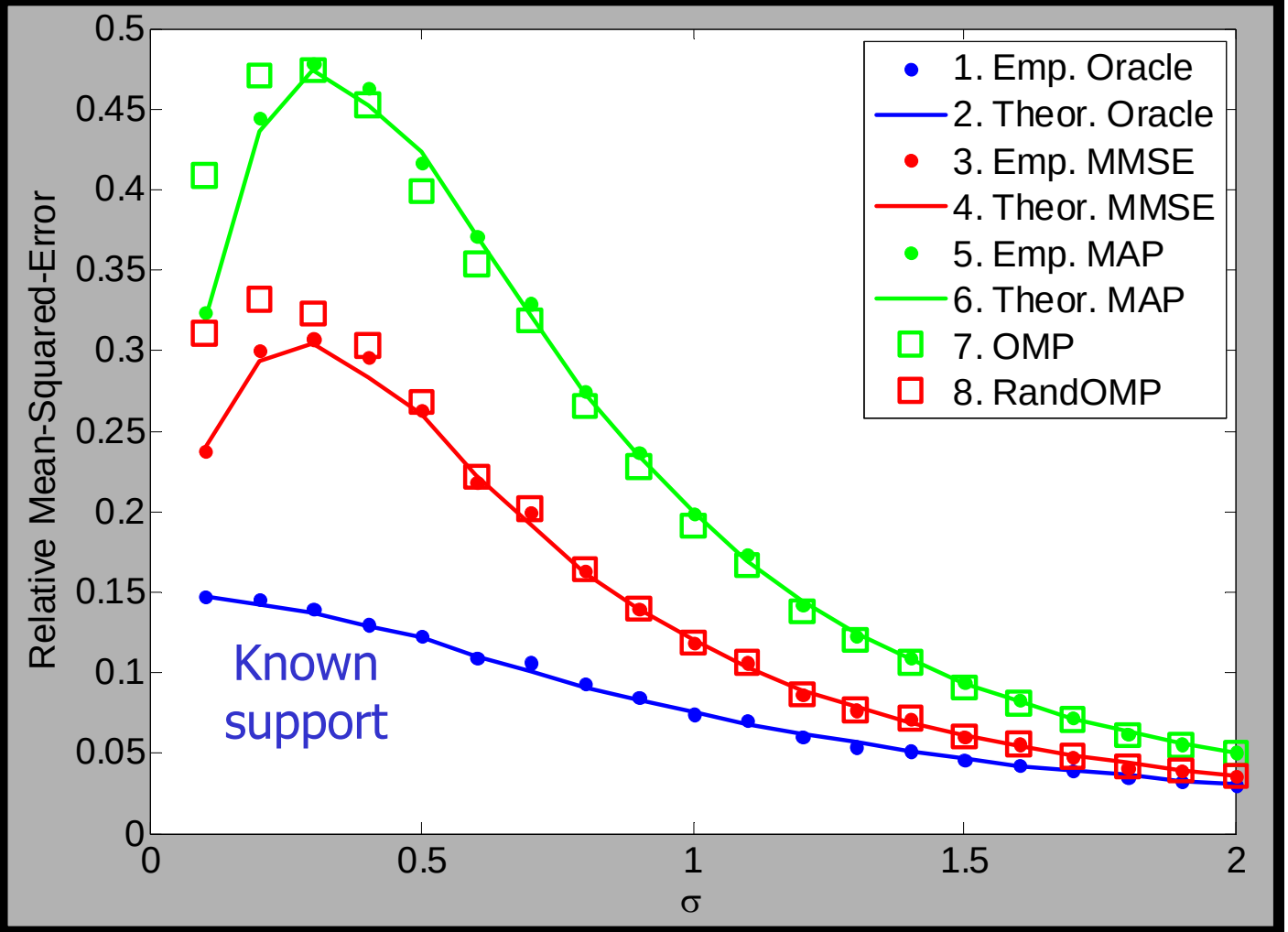
Back to the beginning: Why Use Several Representations?
Because their average leads to provable better noise suppression.

Example

The following results correspond to a small dictionary (20×30), where the combinatorial formulas can be evaluated as well.

Parameters:

- N=20, K=30
- True support=3
- σ_x=1
- J=10
- Averaged over 1000 experiments



Part VI
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.

What next?

- Improving the model
- Improving the dictionaries
- Demonstrating on other applications (graphics?)
- …

More on these (including the slides, the papers, and a Matlab toolbox) in http://www.cs.technion.ac.il/~elad

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

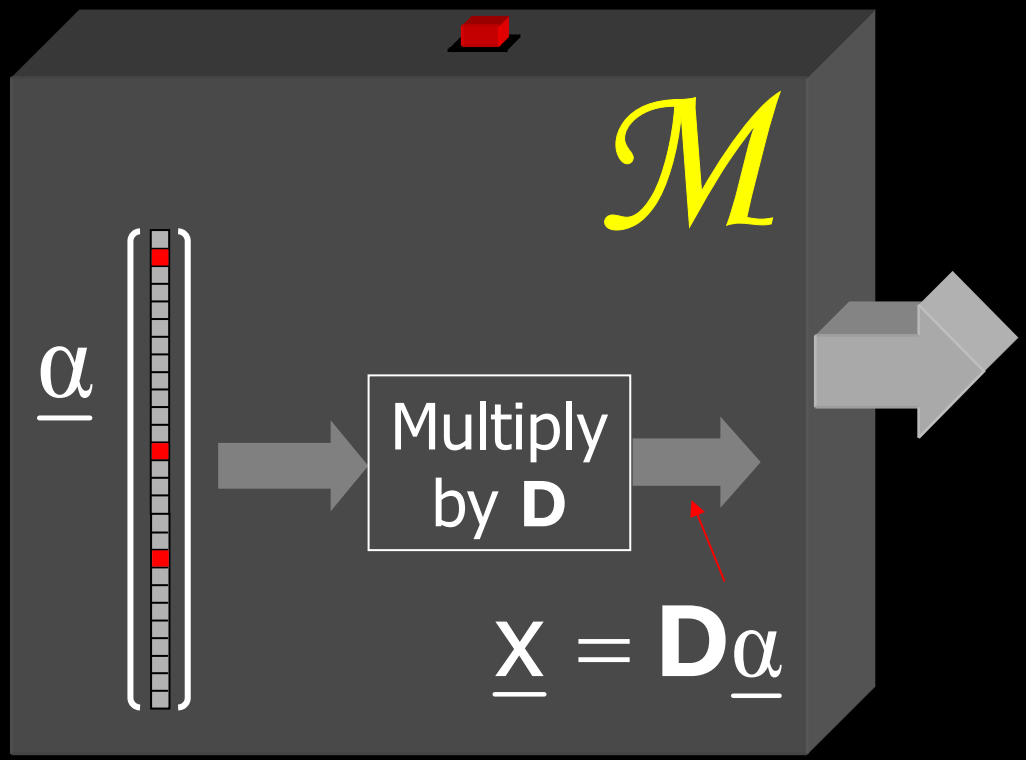

J. Mairal


M. Protter


R. Rubinstein


J. Shtok

Sparseland Signals Are Special

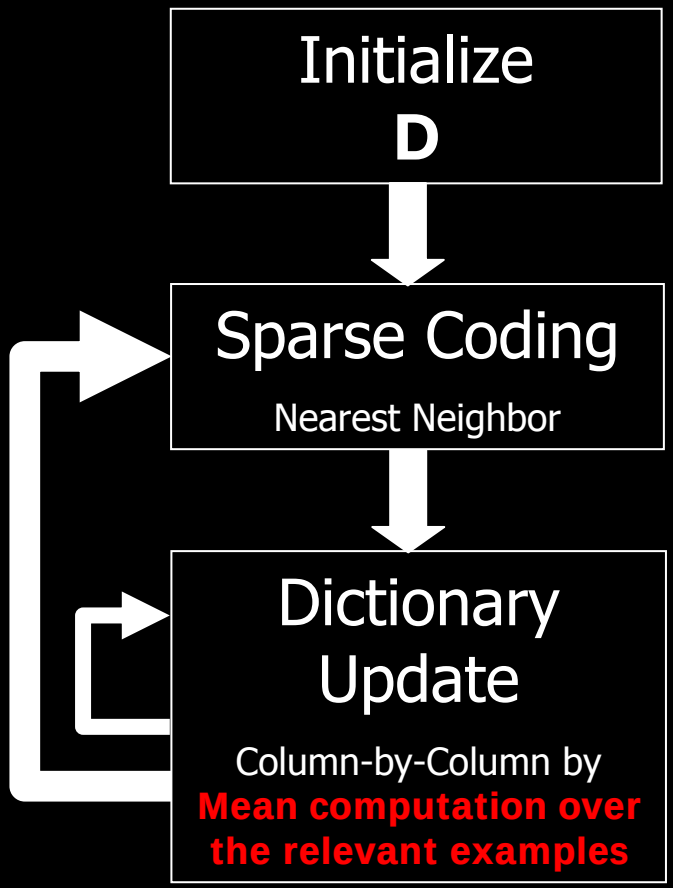


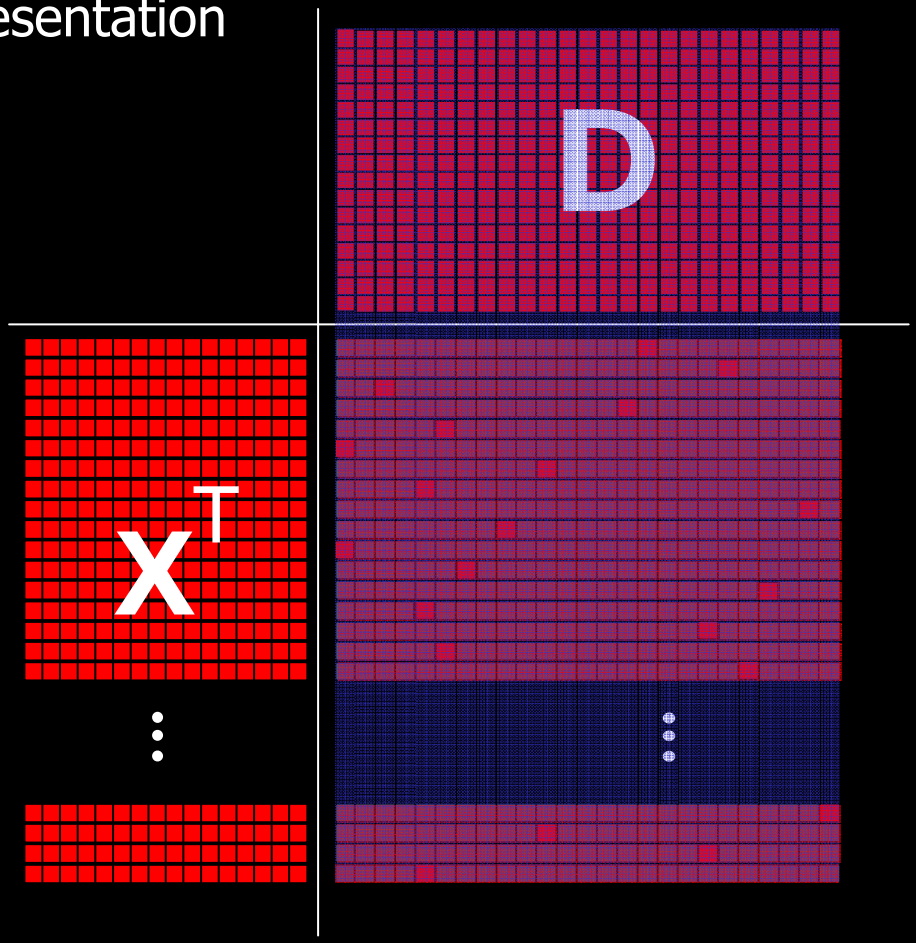
Interesting Model:

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

K–Means For Clustering

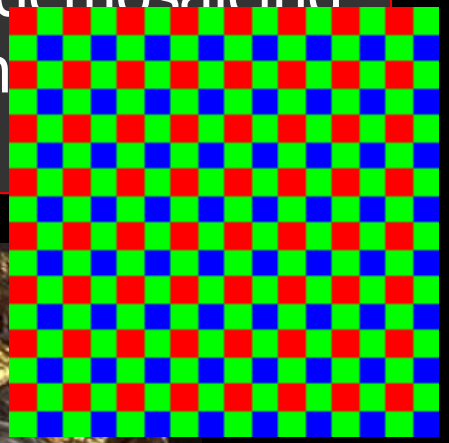
Clustering: An extreme sparse representation





Demosaicing [Mairal, Elad & Sapiro ('06)]

- Today's cameras are sensing only one color per pixel, leaving the rest to be interpolated.
- Our experiments lead to state-of-the-art demosaicing results, giving ~0.2dB better results on compared to [Chang & Chan ('06)]



- Generalizing the previous 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 have handled the demosaicing problem while forcing strong sparsity and only few iterations.
- The same concept can be deployed to inpainting.

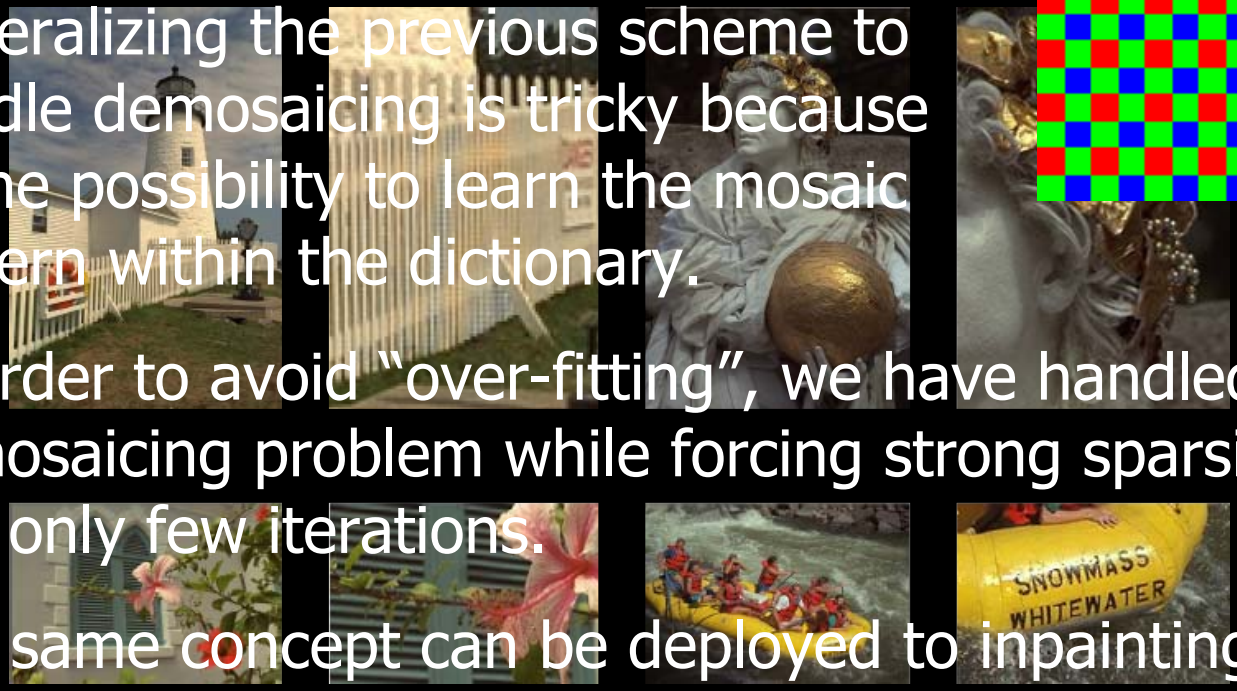


Image Compression Results

Image Compression

Detect main features and warp the images to a common reference (20 parameters)

↓

Divide the image into disjoint 15-by-15 patches. For each compute mean and dictionary

↓

Per each patch find the operating parameters (number of atoms L, quantization Q)

↓

Warp, remove the mean from each patch, sparse code using L atoms, apply Q, and dewarp

On the training set

On the test image

Training set (2500 images)