Recent Trends in Signal Representations and Their Role in Image Processing

Michael Elad

The CS Department

The Technion – Israel Institute of technology

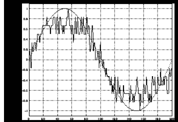
Haifa 32000, Israel

Research Day on Image Analysis and Processing

Industrial Affiliate Program (IAP)

The Computer Science Department – The Technion

January 27th, 2005



Today's Talk is About

Sparsity

and

Overcompleteness

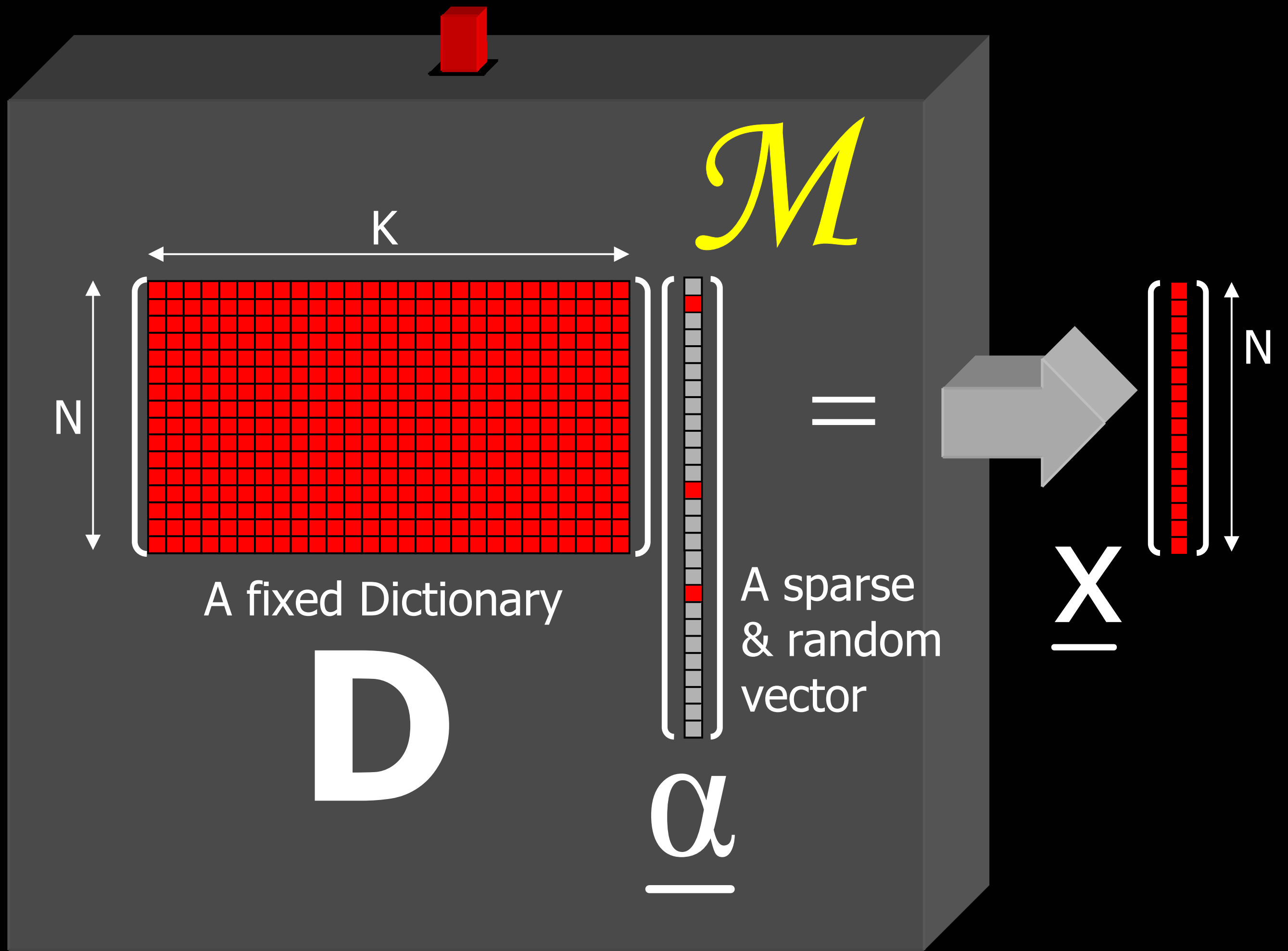
We will show today that

- Sparsity & Overcompleteness can & should be used to design new powerful signal/image processing tools (e.g., transforms, priors, models, …),
- The obtained machinery works very well – we will show these ideas deployed to **applications**.

Agenda

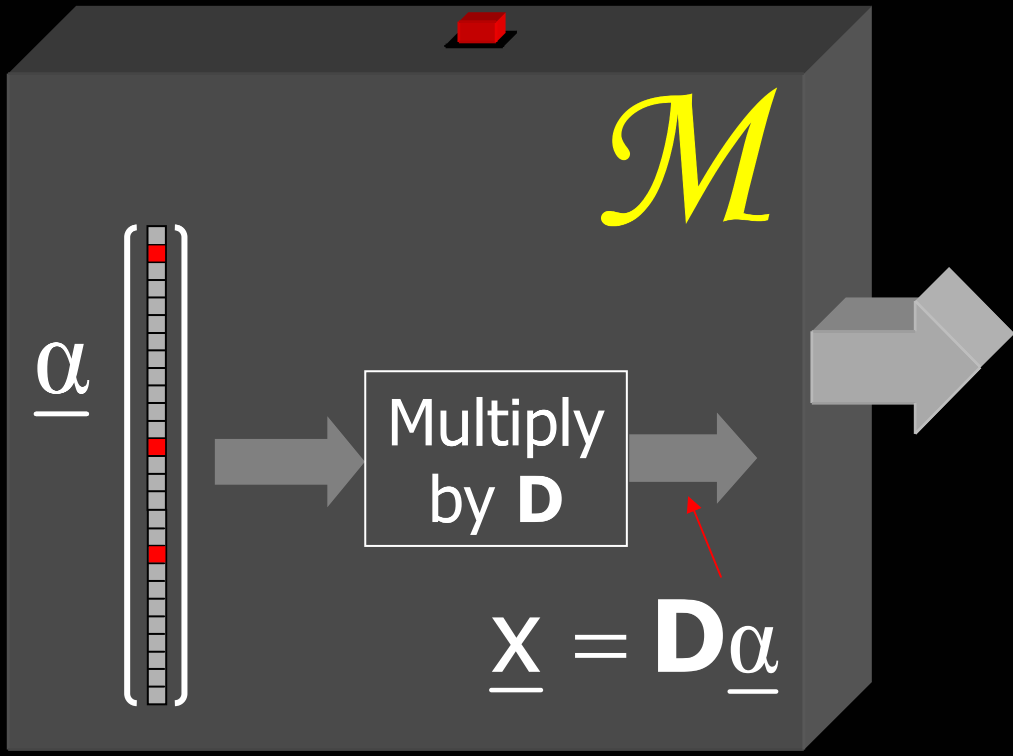
1. **A Visit to *Sparseland***
   Motivating Sparsity & Overcompleteness
2. Problem 1: Transforms & Regularizations
   How & why should this work?
3. Problem 2: What About **D**?
   The quest for the origin of signals
4. Problem 3: Applications
   Image filling in, denoising, separation, compression, …

Generating Signals in *Sparseland*



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

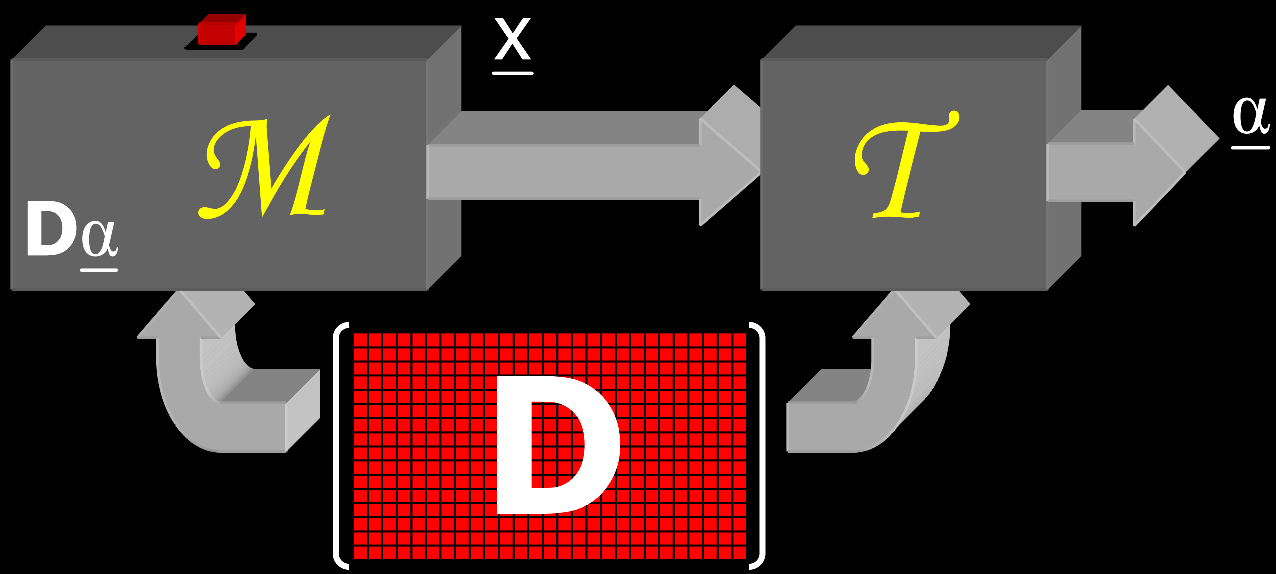
Sparseland Signals Are Special



- **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 special type mixture-of-Gaussians (or Laplacians).

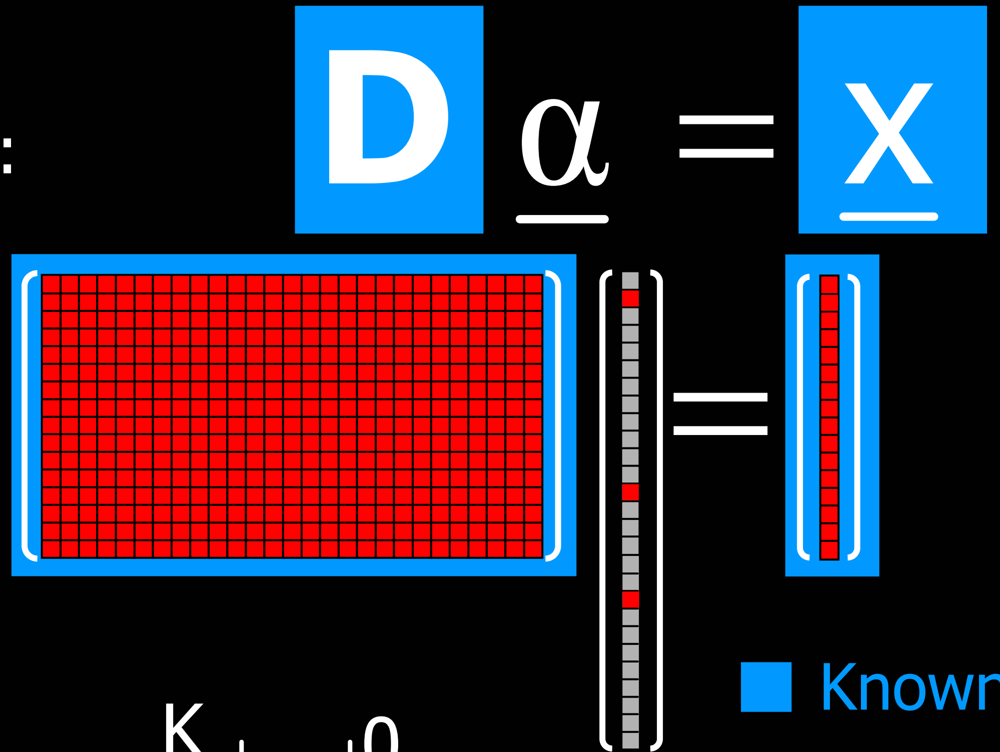
Transforms in *Sparseland* ?

- Assume that $\underline{x}$ is known to emerge from $\mathcal{M}$.
- We desire simplicity, independence, and expressiveness.
- How about "Given $\underline{x}$, find the $\underline{\alpha}$ that generated it in $\mathcal{M}$" ?



So, In Order to Transform …

We need to solve an under-determined linear system of equations:



- Among all (infinitely many) possible solutions we want the sparsest !!

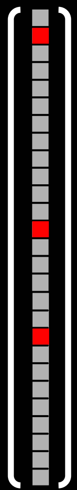
- Sparsity is measured using the L_0 norm:

$$\|\underline{\alpha}\|_0 = \sum_{j=1}^{K} |\alpha_j|^0$$

Signal's Transform in *Sparseland*

A sparse & random vector

$\underline{\alpha} =$



Multiply by **D**

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

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

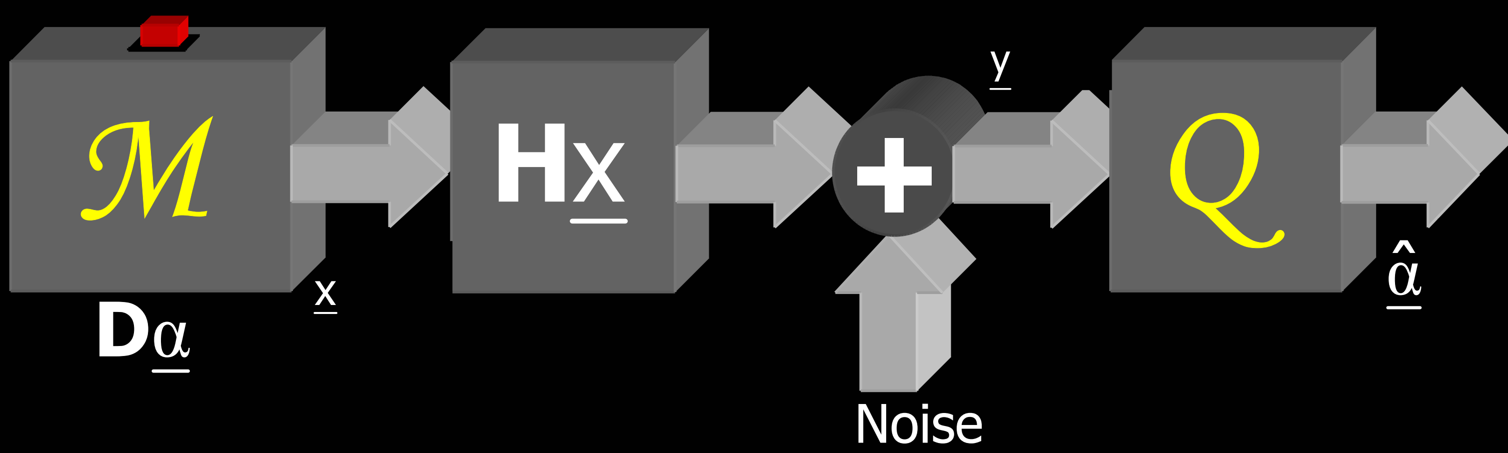
$\hat{\underline{\alpha}}$

4 Major Questions

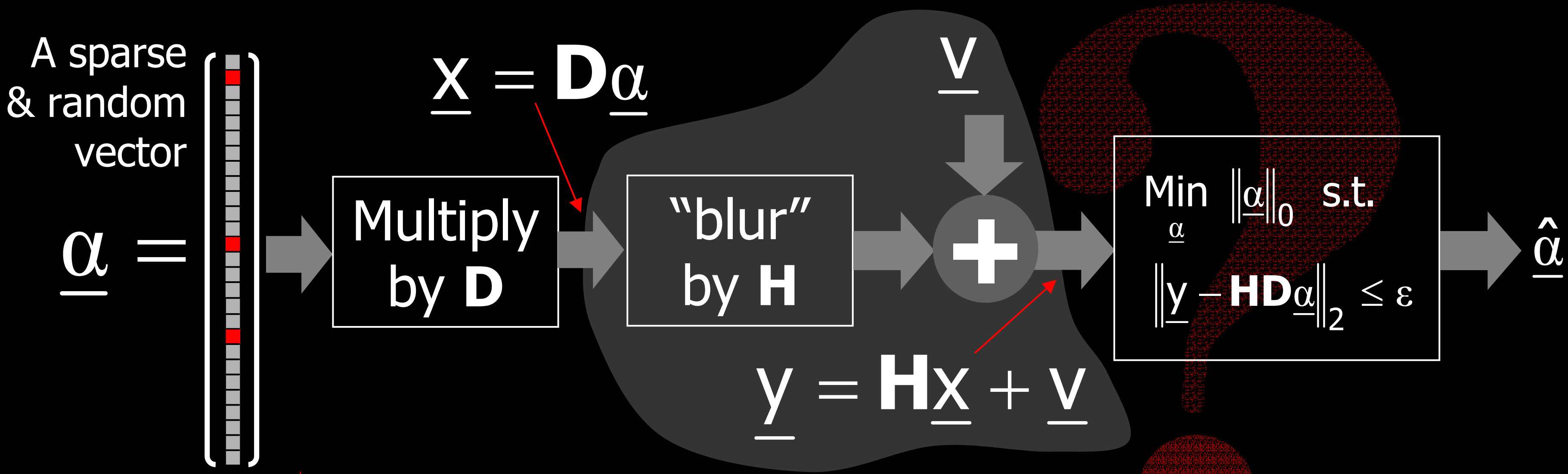
- Is $\hat{\underline{\alpha}} = \underline{\alpha}$? Under which conditions?
- Are there practical ways to get $\hat{\underline{\alpha}}$?
- How effective are those ways?
- How would we get **D**?

Inverse Problems in *Sparseland* ?

- Assume that $\underline{x}$ is known to emerge from $\mathcal{M}$.
- Suppose we observe $\underline{y} = \mathbf{H}\underline{x} + \underline{v}$, a "blurred" and noisy version of $\underline{x}$ with $\|\underline{v}\|_2 \le \varepsilon$. How will we recover $\underline{x}$?
- How about "find the $\underline{\alpha}$ that generated the $\underline{x}$ …" again?



Inverse Problems in *Sparseland* ?



4 Major Questions (again!)

- Is $\underline{\hat{\alpha}} = \underline{\alpha}$? How far can it go?
- Are there practical ways to get $\underline{\hat{\alpha}}$?
- How effective are those ways?
- How would we get **D**?

Back Home … Any Lessons?

Several recent trends worth looking at:

- JPEG to JPEG2000 - From (L … let and non-linear approximati…
- From … (Fourier) to L_1. … ➔ Sparsity.
- From u… …ift-invariar… …completeness.
- Approxi… …near approximation ➔ Spa… Overcompleteness.
- ICA and related models ➔ Independence and Sparsity.

Sparseland is HERE !

To Summarize so far …

The *Sparseland* model $\mathcal{M}$ for signals is interesting since it is rich and yet every signal has a simple description

Who cares?

We do! this model is relevant to us just as well.

So, what are the implications?

We need to solve (or approximate the solution of) a linear system with more unknowns than equations, while finding the sparsest possible solution

Sparsity? looks complicated

(a) Practical solvers?
(b) How will we get **D**?
(c) Applications?

Agenda

1. A Visit to *Sparseland*

   Motivating Sparsity & Overcompleteness

2. **Problem 1: Transforms & Regularizations**

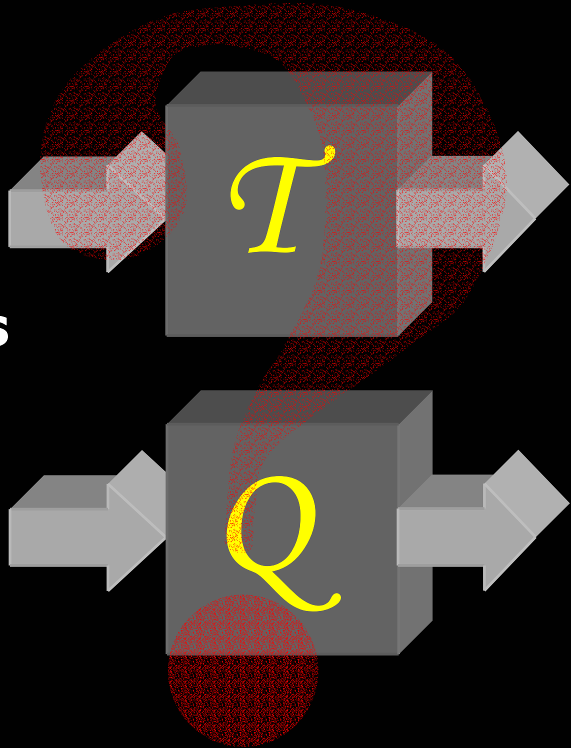
   How & why should this work?

3. Problem 2: What About **D**?

   The quest for the origin of signals

4. Problem 3: Applications

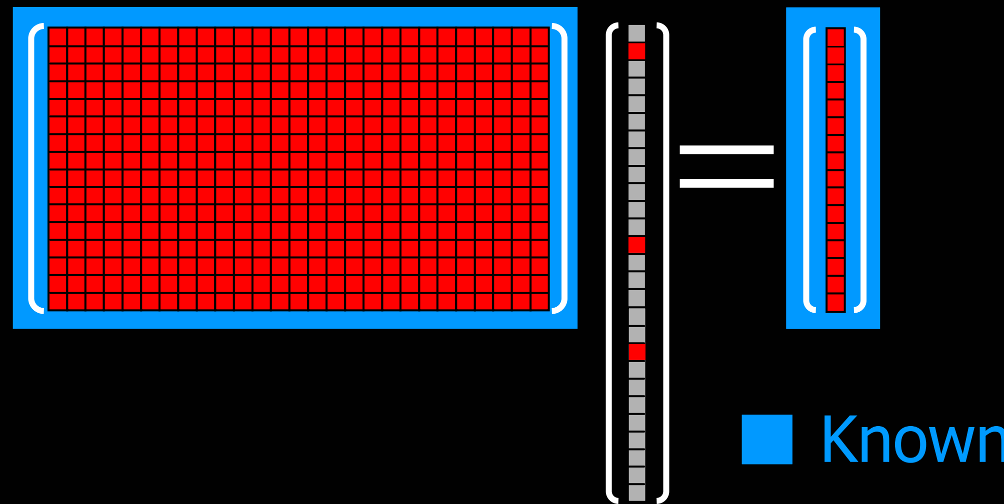
   Image filling in, denoising, separation, compression, …



Lets Start with the Transform …

Our dream for Now: Find the sparsest solution of

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



Put formally,

$$\min_{\underline{\alpha}} \left\|\underline{\alpha}\right\|_0 \quad \text{s.t.} \quad \boxed{\underline{x}} = \boxed{\mathbf{D}}\underline{\alpha}$$

known

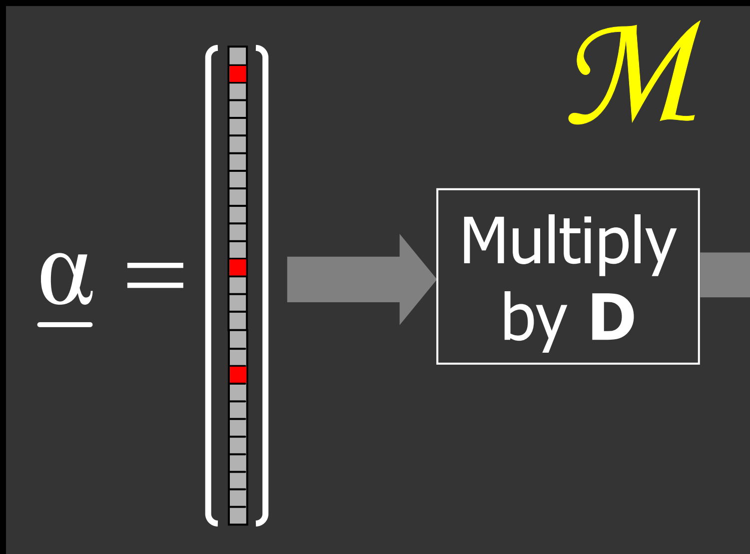
Questions to Address

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

4 Major Questions

- Is $\underline{\hat{\alpha}} = \underline{\alpha}$? Under which conditions?
- Are there practical ways to get $\underline{\hat{\alpha}}$?
- How effective are those ways?
- How would we get **D**?

Question 1 – Uniqueness?



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

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

$\hat{\underline{\alpha}}$

Suppose we can solve this exactly

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

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

Matrix "Spark"

Definition: Given a matrix **D**, σ=Spark{**D**} is the smallest number of columns that are linearly dependent.

Donoho & Elad ('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

Uniqueness Rule

Suppose this problem has been solved somehow

$$\min_{\underline{\alpha}} \left\|\underline{\alpha}\right\|_0 \quad \text{s.t.} \quad \underline{x} = \mathbf{D}\underline{\alpha}$$

Uniqueness

Donoho & Elad ('02)

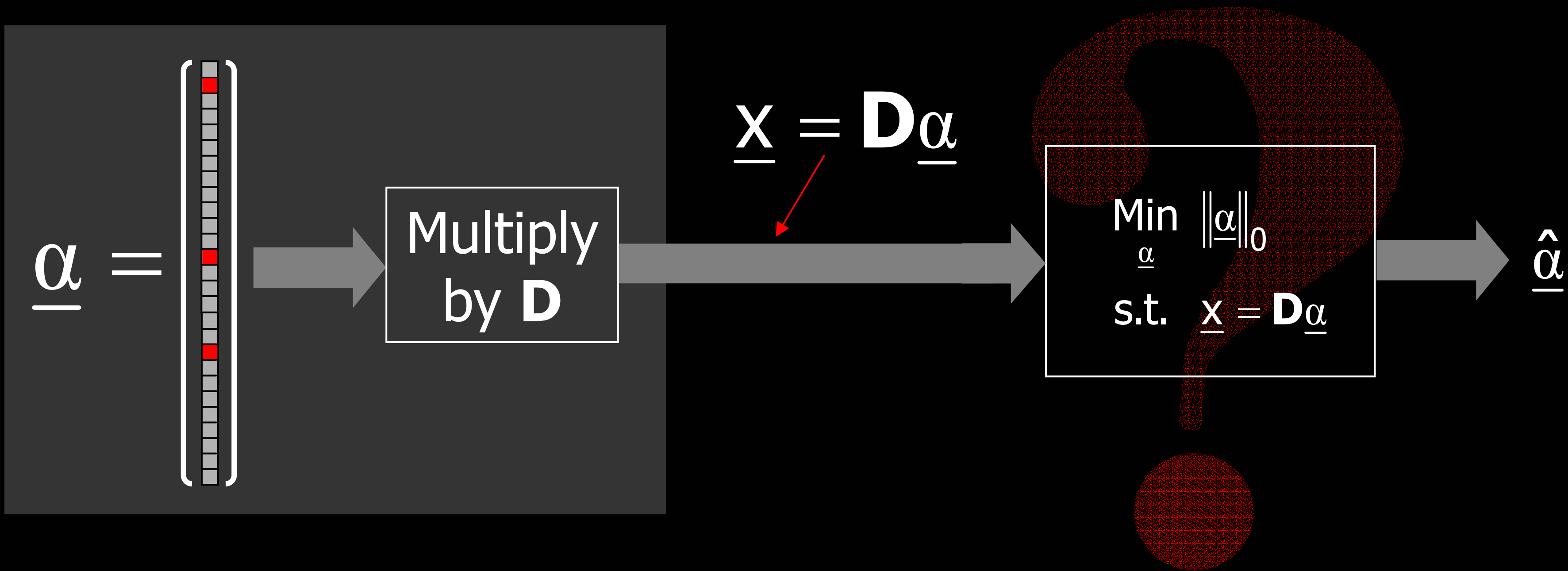
If we found a representation that satisfy

$$\frac{\sigma}{2} > \left\|\underline{\alpha}\right\|_0$$

Then necessarily it is unique (the sparsest).

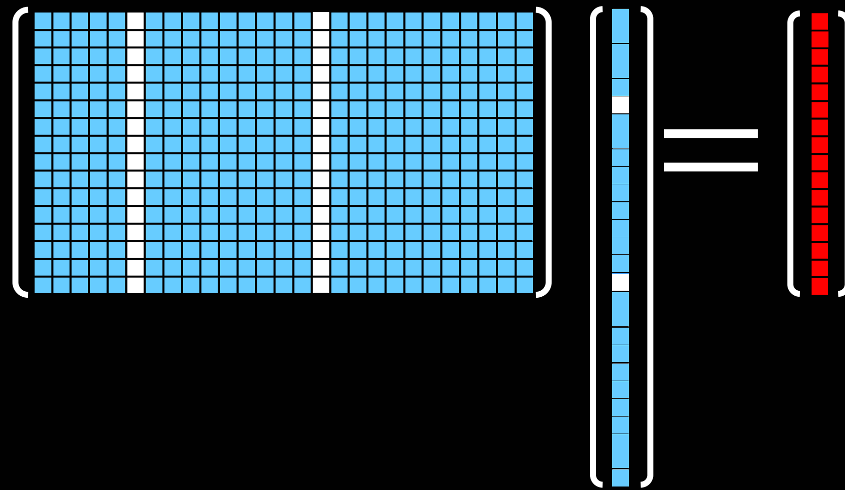
This result implies that if $\mathcal{M}$ generates signals using "sparse enough" $\underline{\alpha}$, the solution of the above will find it exactly.

Question 2 – Practical P_0 Solver?



Are there reasonable ways to find $\hat{\underline{\alpha}}$?

Matching Pursuit (MP) Mallat & Zhang (1993)



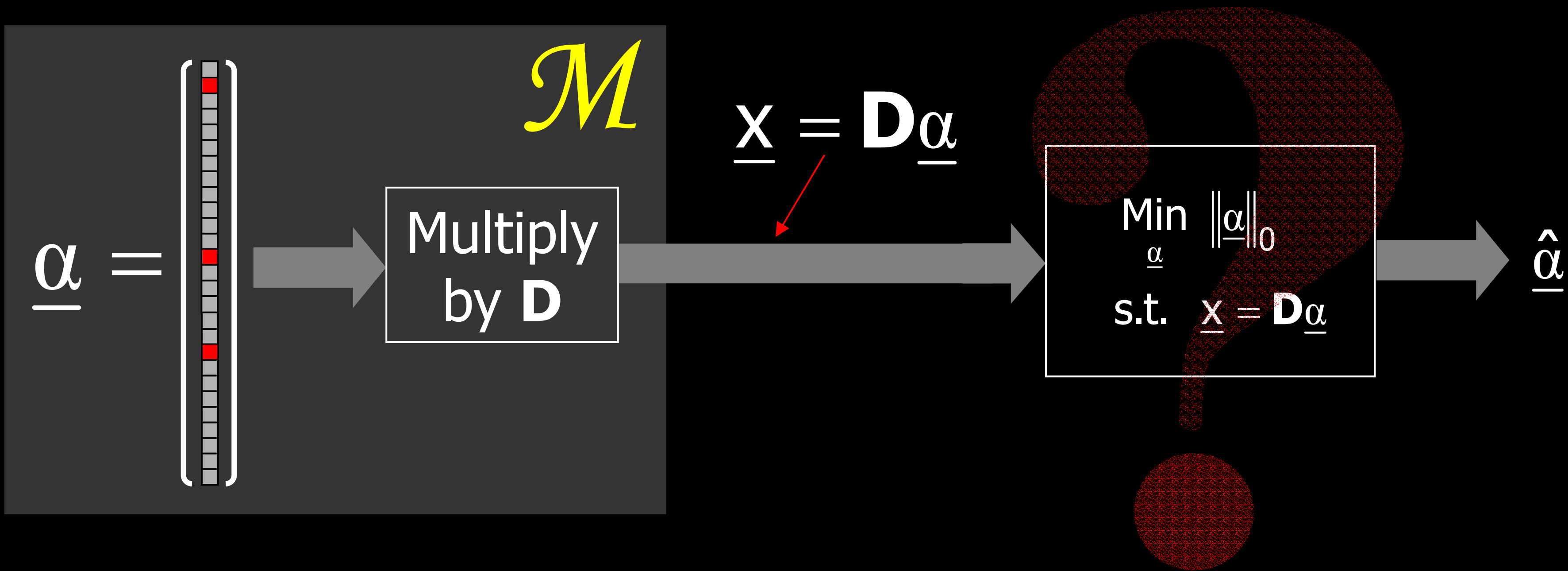
- The MP is a greedy algorithm that finds one atom at a time.
- Step 1: find the one atom that best matches the signal.
- Next steps: given the previously found atoms, find the next **one** to best fit …
- The Orthogonal MP (OMP) is an improved version that re-evaluates the coefficients after each round.

Basis Pursuit (BP) Chen, Donoho, & Saunders (1995)

Instead of solving		Solve Instead
$\min_{\underline{\alpha}} \left\|\underline{\alpha}\right\|_0 \quad \text{s.t.} \quad \underline{x} = \mathbf{D}\underline{\alpha}$	$\Rightarrow$	$\min_{\underline{\alpha}} \left\|\underline{\alpha}\right\|_1 \quad \text{s.t.} \quad \underline{x} = \mathbf{D}\underline{\alpha}$

- The newly defined problem is convex.
- It has a Linear Programming structure.
- Very efficient solvers can be deployed:
  - Interior point methods [Chen, Donoho, & Saunders (`95)] ,
  - Sequential shrinkage for union of ortho-bases [Bruce et.al. (`98)],
  - If computing $\mathbf{D}\underline{x}$ and $\mathbf{D}^T\underline{\alpha}$ are fast, based on shrinkage [Elad (`05)].

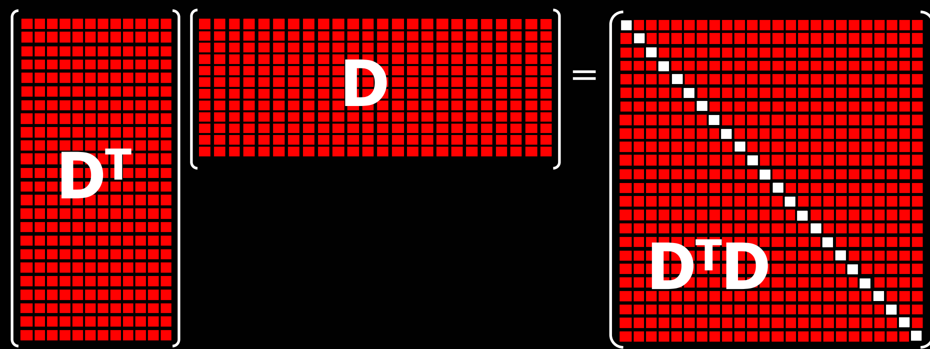
Question 3 – Approx. Quality?



How effective are the MP/BP in finding $\hat{\underline{\alpha}}$?

Evaluating the "Spark"

- Compute



Assume normalized columns

- The Mutual Incoherence M is the largest entry in absolute value outside the main diagonal of $\mathbf{D}^T\mathbf{D}$.

- The Mutual Incoherence is a property of the dictionary (just like the "Spark"). The smaller it is, the better the dictionary.

BP and MP Equivalence

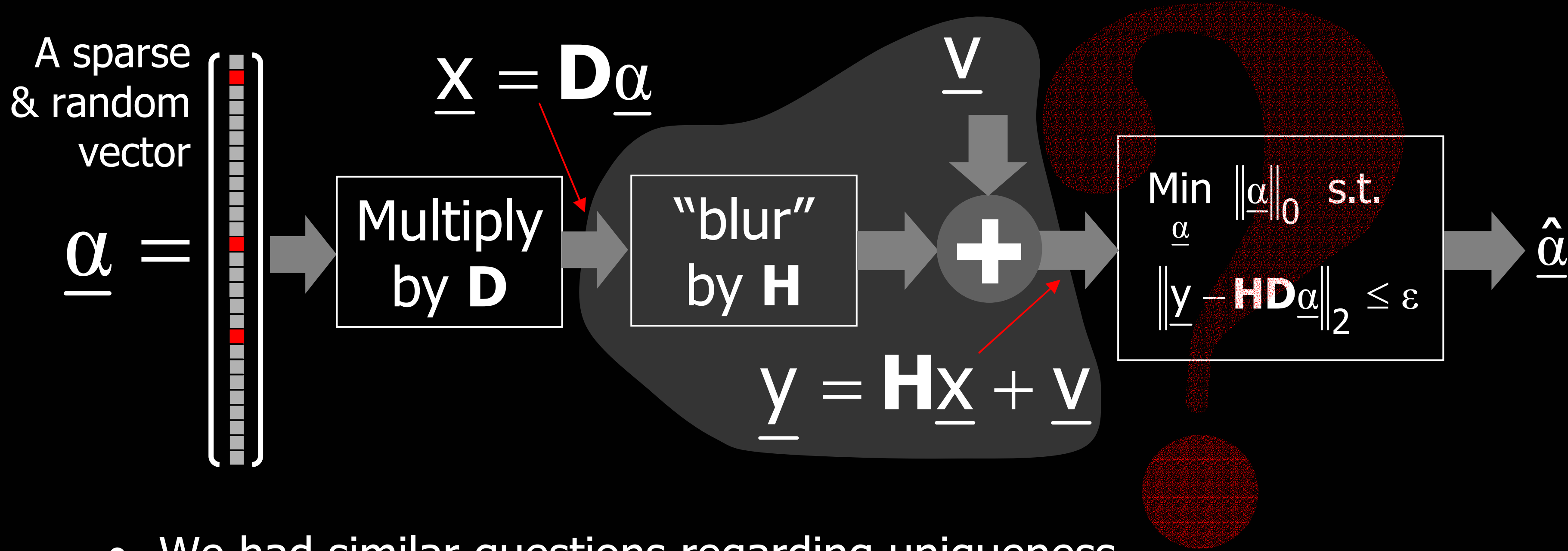
Equivalence

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

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

- MP is typically inferior to BP!
- The above result corresponds to the worst-case.
- Average performance results are available too, showing much better bounds [Donoho (`04), Candes et.al. (`04), Elad and Zibulevsky (`04)].

What About Inverse Problems?



- We had similar questions regarding uniqueness, practical solvers, and their efficiency.
- It turns out that similar answers are applicable here due to several recent works [Donoho, Elad, and Temlyakov (`04), Tropp (`04), Fuchs (`04)].

To Summarize so far …

The *Sparseland* model for signals is relevant to us. We can design transforms and priors based on it

Find the sparsest solution?

Use pursuit Algorithms

Why works so well?

A sequence of works during the past 3-4 years gives theoretic justifications for these tools behavior

What next?

(a) How shall we find **D**?

(b) Will this work for applications? Which?

Agenda

1. A Visit to *Sparseland*

   Motivating Sparsity & Overcompleteness

2. Problem 1: Transforms & Regularizations

   How & why should this work?

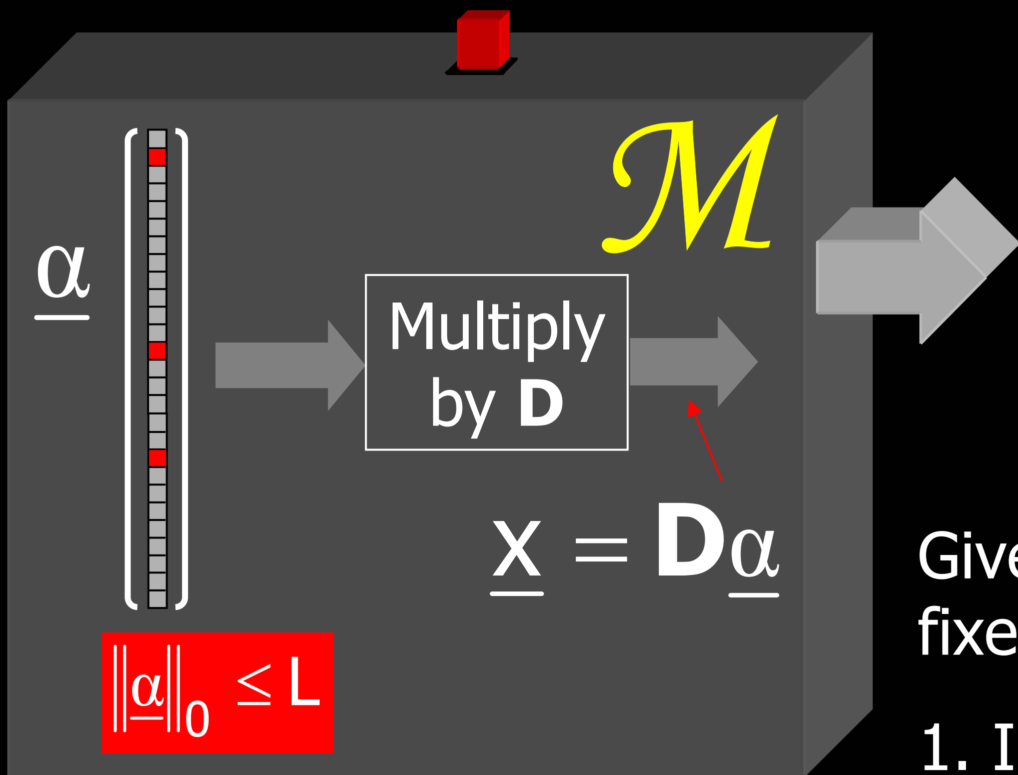
3. **Problem 2: What About D?**

   The quest for the origin of signals

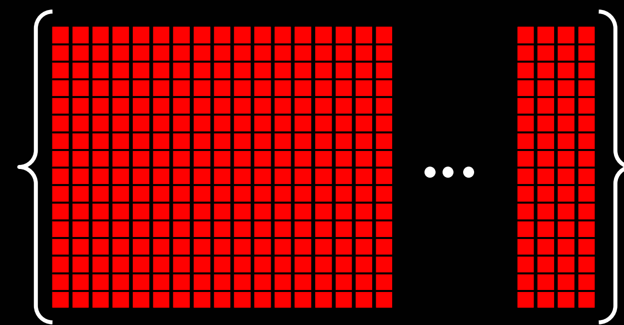
4. Problem 3: Applications

   Image filling in, denoising, separation, compression, …

Problem Setting



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



Given these P examples and a fixed size [N×K] dictionary **D**:

1. Is **D** unique?

2. How would we find **D**?

Uniqueness?

Uniqueness

Aharon, Elad, & Bruckstein (`05)

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

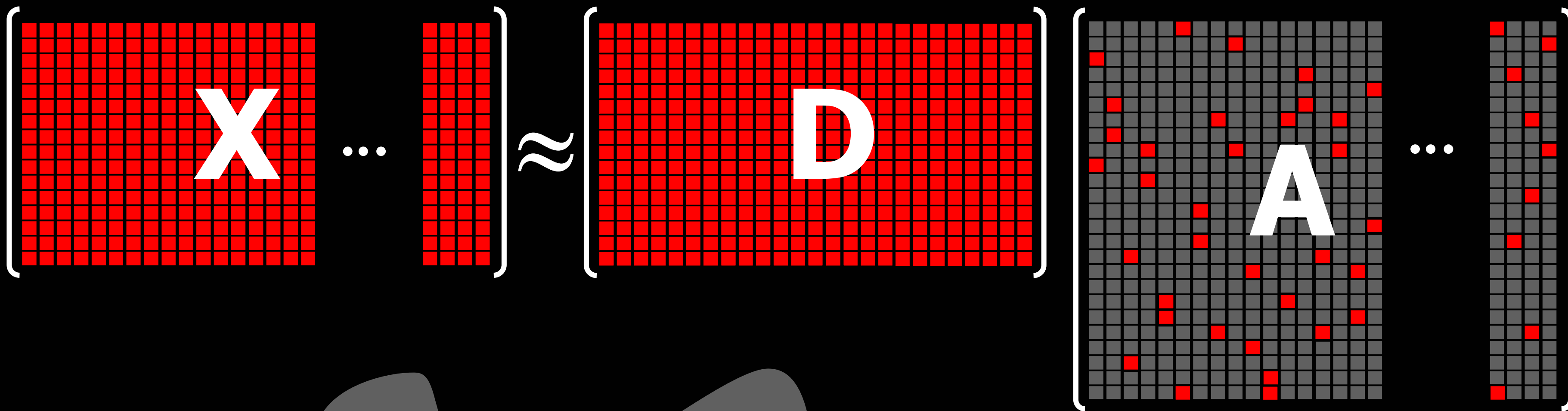
$$L < \frac{\text{Spark}\left\{\mathbf{D}\right\}}{2}$$

then **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.
- 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 can be constructed similarly.

Practical Approach – Objective



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

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

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

The K–SVD Algorithm – General

Aharon, Elad, & Bruckstein (`04)

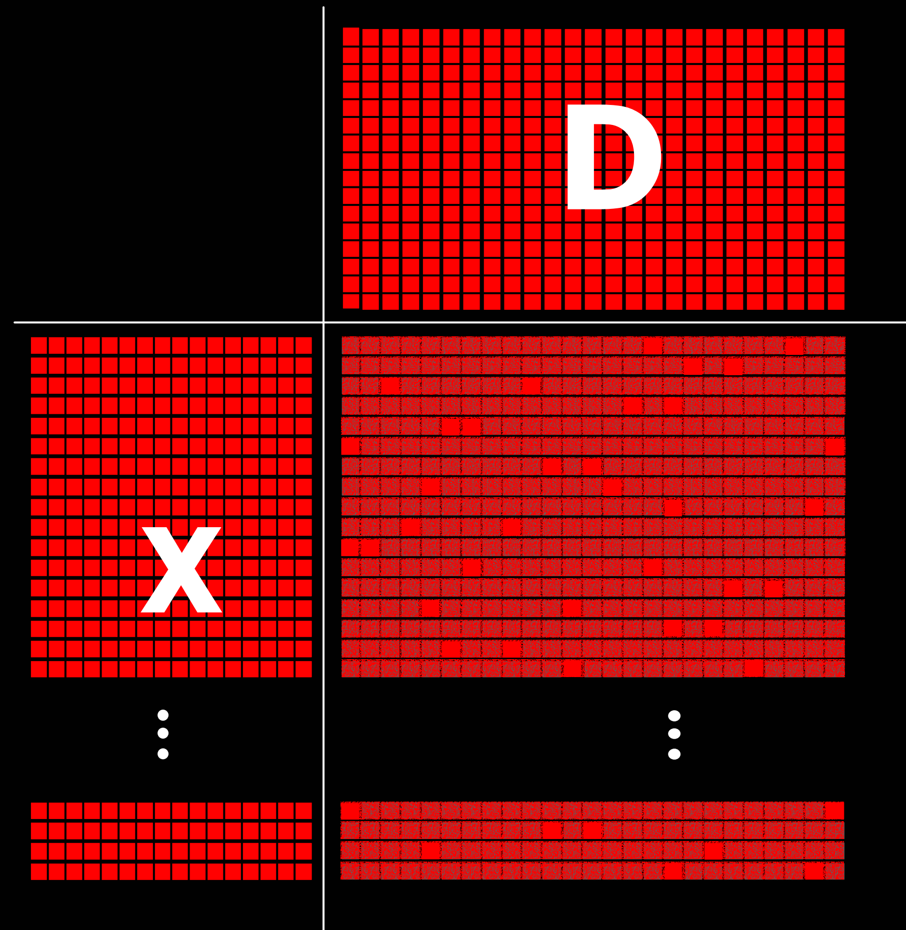
Initialize
D

Sparse Coding

Use MP or BP

Dictionary
Update

Column-by-Column by
SVD computation



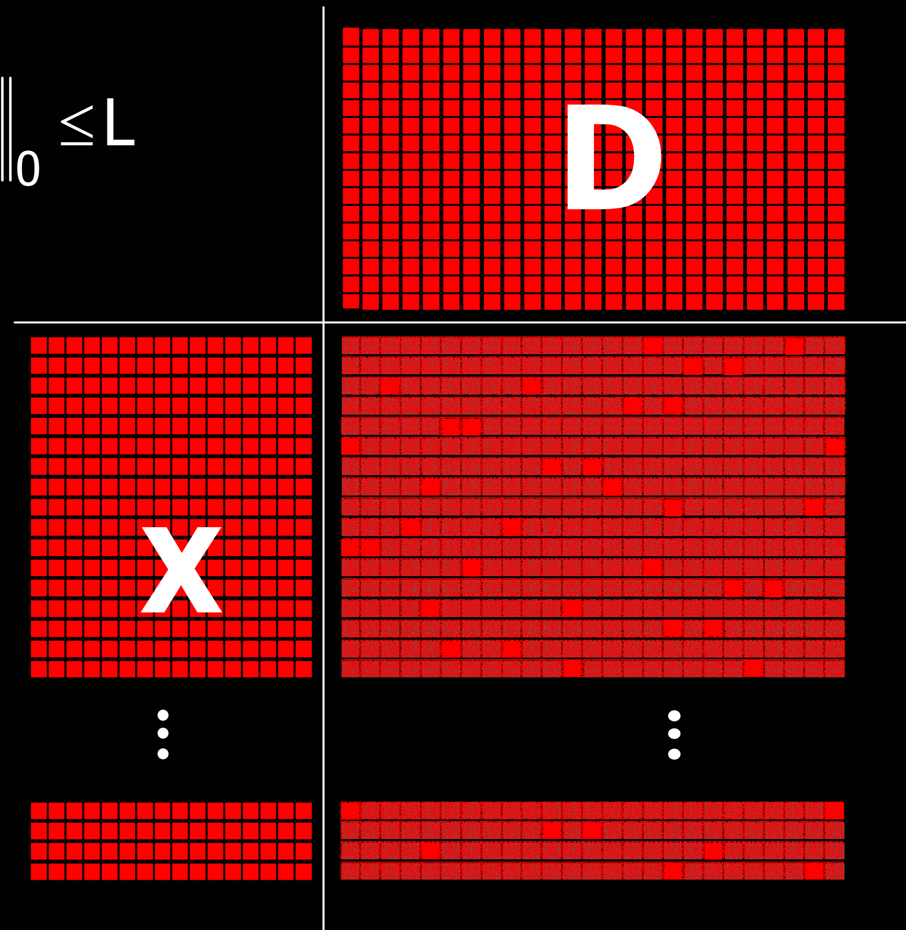
K–SVD Sparse Coding Stage

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

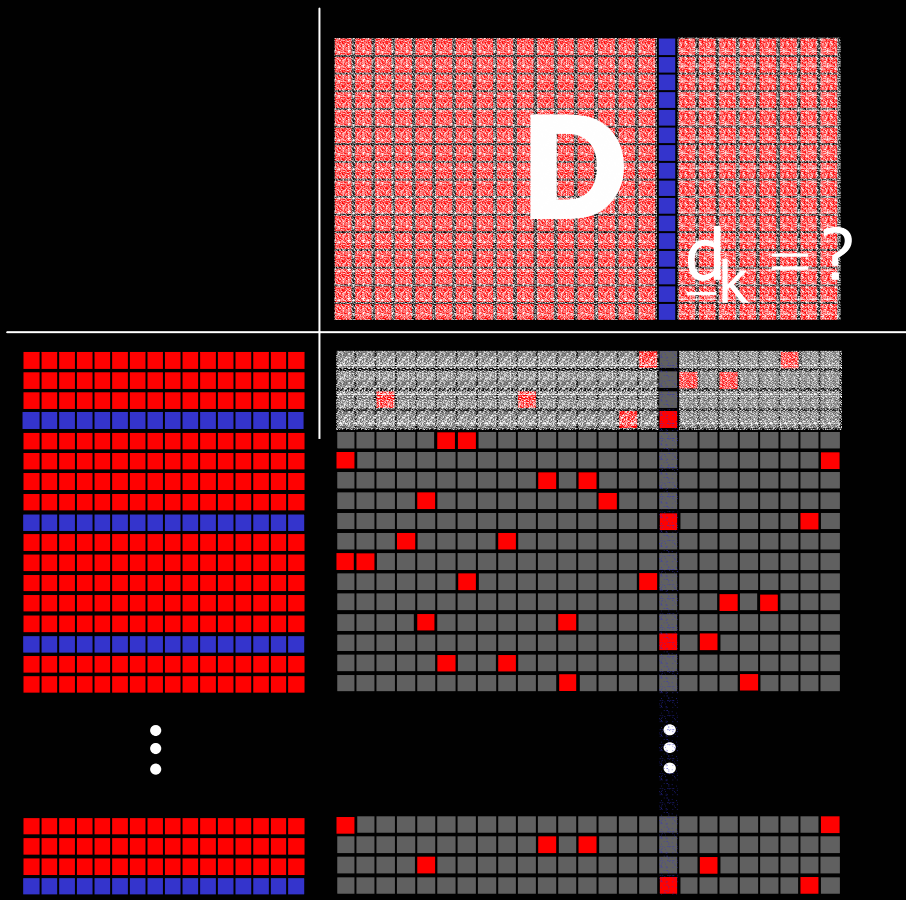
For the jth example we solve

$$\underset{\underline{\alpha}}{\text{Min}} \quad \left\| \mathbf{D}\underline{\alpha} - \underline{x}_j \right\|_2^2 \quad \text{s.t.} \quad \left\| \underline{\alpha} \right\|_0 \le L$$

Pursuit Problem !!!



K–SVD Dictionary Update Stage



G_k : The examples in $\{\underline{x}_j\}_{j=1}^{P}$ that use the column $\underline{d}_k$.

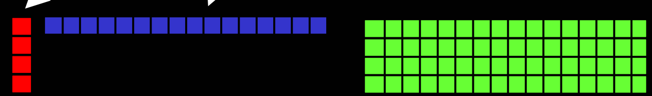
The content of $\underline{d}_k$ influences only the examples in G_k.

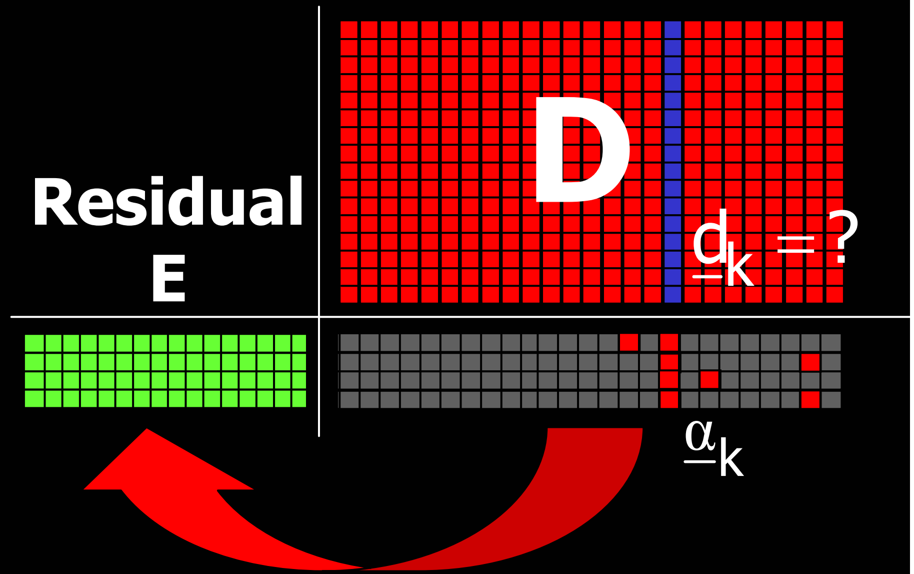
Let us fix all **A** apart from the k^{th} column and seek both $\underline{d}_k$ and the k^{th} column to better fit the **residual**!

K–SVD Dictionary Update Stage

We should solve:

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





$\underline{d}_k$ is obtained by **SVD** on the examples' residual in G_k.

To Summarize so far …

The *Sparseland* model for signals is relevant to us. In order to use it effectively we need to know **D**

How **D** can be found?

Use the K-SVD algorithm

Will it work well?

We have established a uniqueness result and shown how to practically train **D** using the K-SVD

What next?

(a) Deploy to applications

(b) Generalize in various ways: multiscale, non-negative factorization, speed-up, …

(c) Performance guarantees?

Agenda

1. A Visit to *Sparseland*

   Motivating Sparsity & Overcompleteness

2. Problem 1: Transforms & Regularizations

   How & why should this work?

3. Problem 2: What About D?

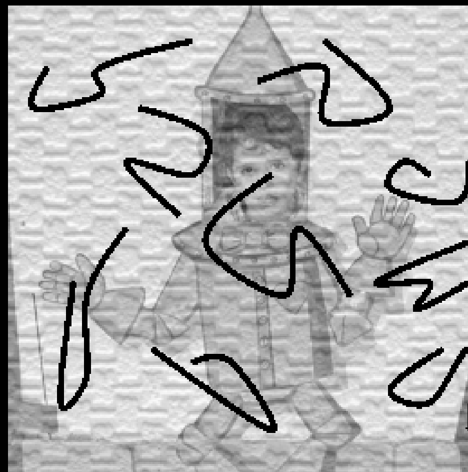
   The quest for the origin of signals

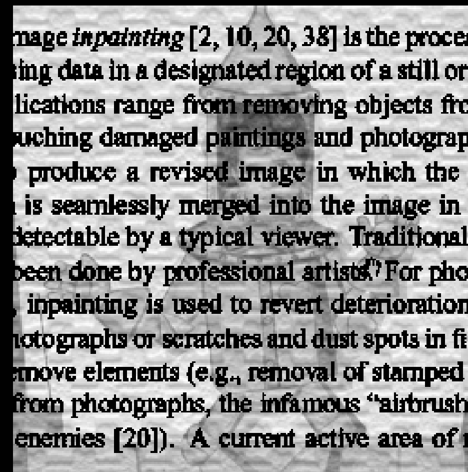
4. **Problem 3: Applications**

   Image filling in, denoising, separation, compression, …

Inpainting (1) [Elad, Starck, & Donoho (`04)]

Source



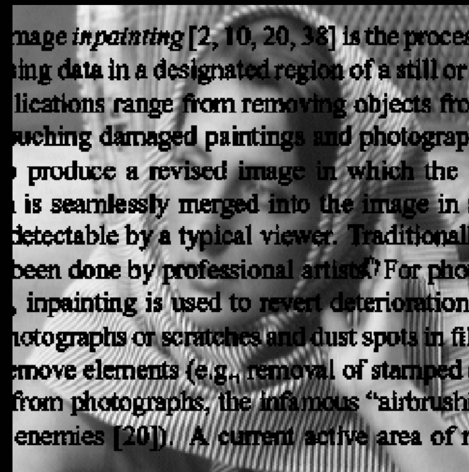


predetermined dictionary: **Curvelet+DCT**

Outcome

Inpainting (2)

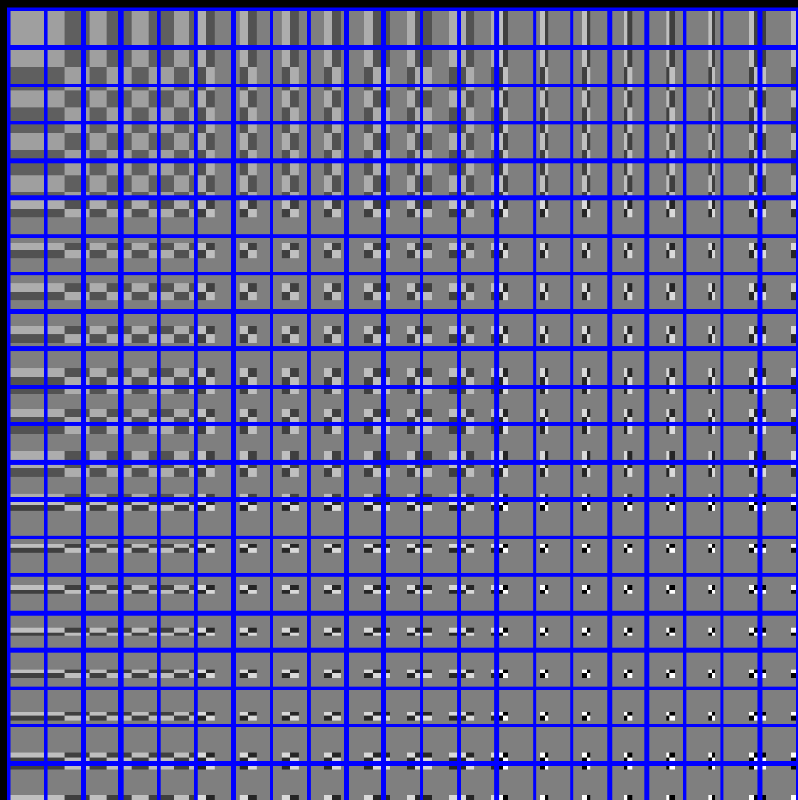
Source



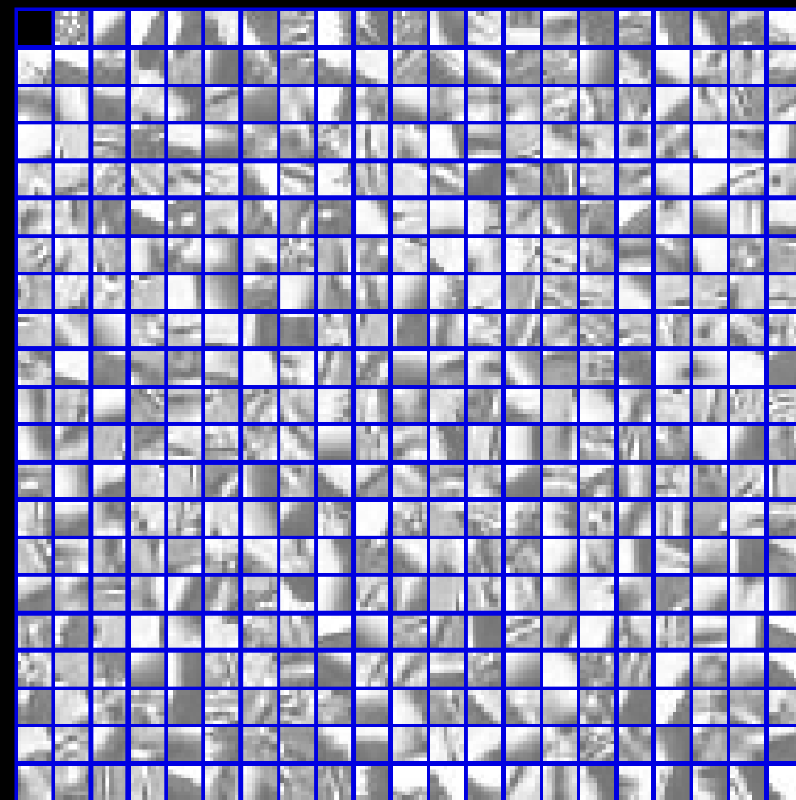
predetermined dictionary: **Curvelet+DCT**

Outcome

K-SVD on Images [Aharon, Elad, & Bruckstein (`04)]



Overcomplete Haar



K-SVD:

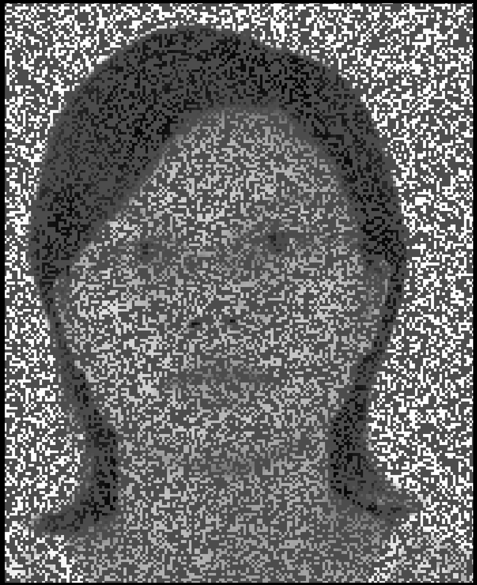
10,000 sample 8-by-8 images.

441 dictionary elements.

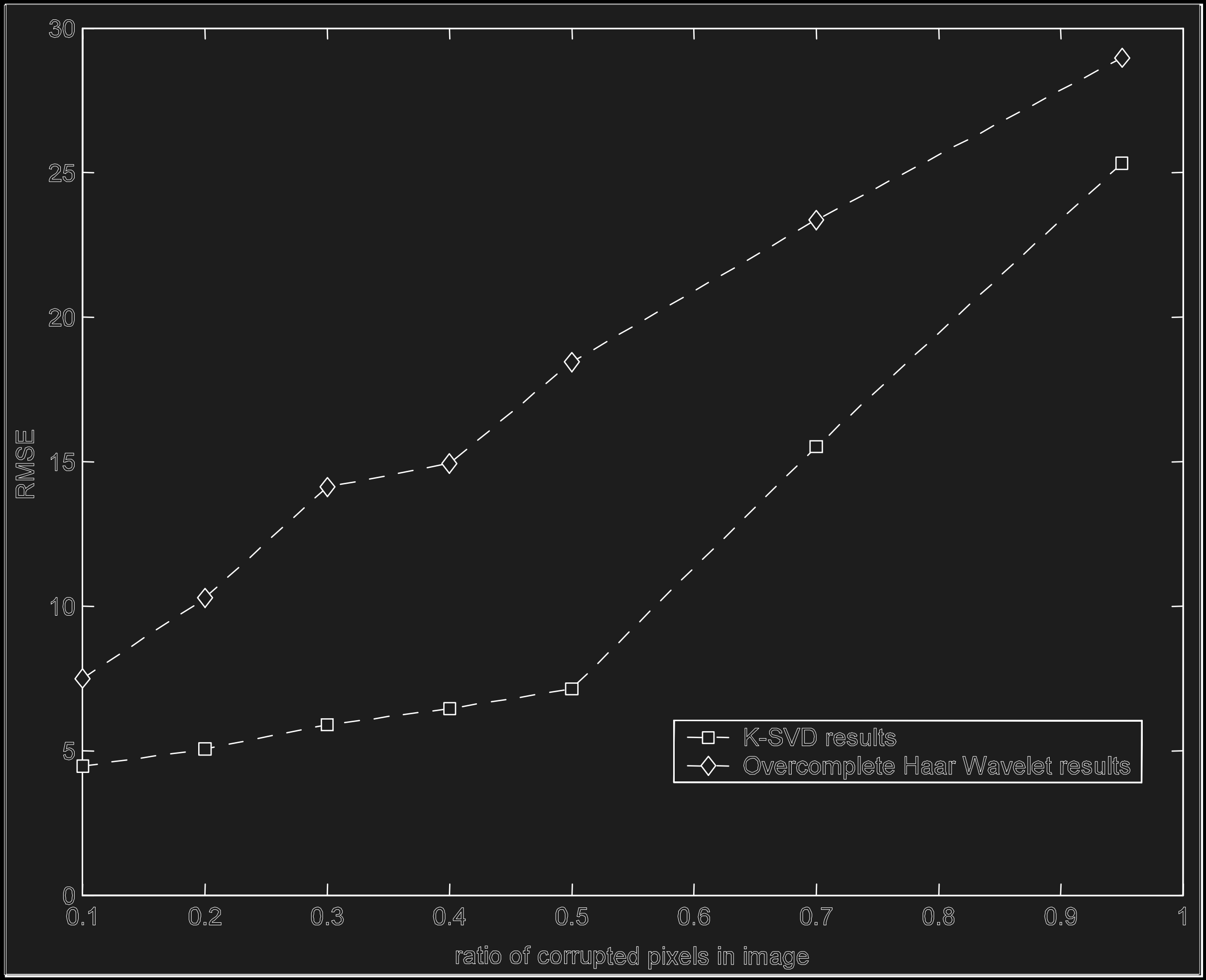
Approximation method: MP.

Filling-In Missing Pixels

90% missing pixels	K-SVD Results Average # coefficients 4.27 RMSE: 25.32	Haar Results Average # coefficients 4.48 RMSE: 28.97



Filling-In Missing Pixels



Compression

$$\mathrm{BPP} = \frac{C\left(10 + \log K\right)}{N}$$

$$\mathrm{BPP} = \frac{C \cdot 10}{N}$$

K-SVD dictionary

Haar dictionary

DCT dictionary

OMP with error bound

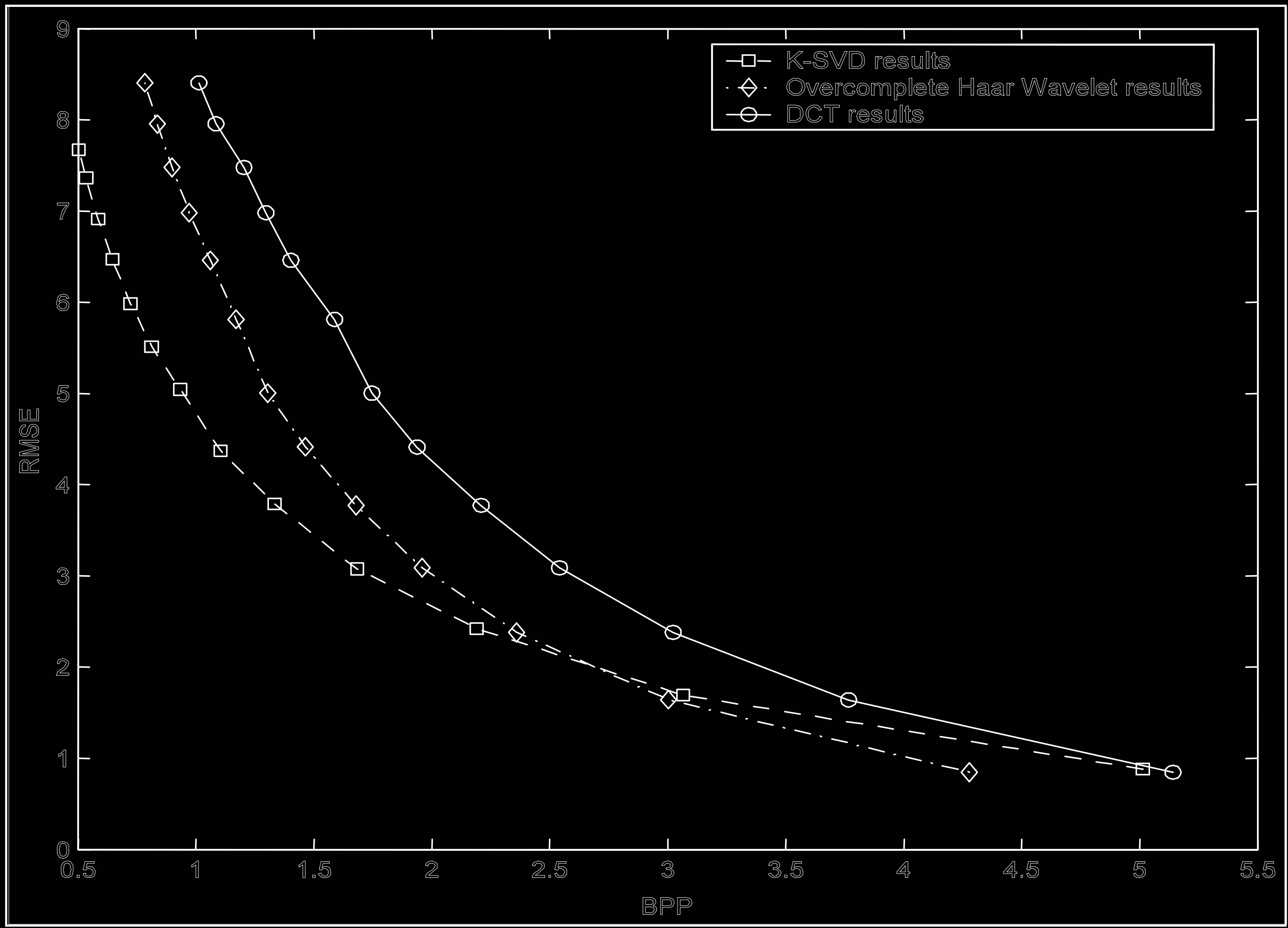
Compression

K-SVD Dictionary
BPP = 0.502
RMSE = 7.67

Haar Dictionary
BPP = 0.784
RMSE = 8.41

DCT Results
BPP = 1.01
RMSE = 8.14

Compression



Today We Have Discussed

1. A Visit to *Sparseland*

   Motivating Sparsity & Overcompleteness

2. Problem 1: Transforms & Regularizations

   How & why should this work?

3. Problem 2: What About **D**?

   The quest for the origin of signals

4. Problem 3: Applications

   Image filling in, denoising, separation, compression, …

Summary

Sparsity and Over-completeness are important ideas that can be used in designing better tools in signal/image processing

There are difficulties in using them!

We are working on resolving those difficulties:

- Performance of pursuit alg.
- Speedup of those methods,
- Training the dictionary,
- Demonstrating applications,
- …

The dream?

Future transforms and regularizations will be data-driven, non-linear, overcomplete, and promoting sparsity.