Dictionary-Learning for the Analysis Sparse Model *

Michael Elad

The Computer Science Department

The Technion – Israel Institute of technology

Haifa 32000, Israel

Joint work with
Boaz Ophir,
Mark Plumbley, and
Nancy Bertin

Special Session on Processing and recovery using analysis and synthesis sparse models – September 1st 2011

This work was supported by the European Commission FP7-FET program SMALL

Sparse Models, Algorithms and Learning for Large-scale data

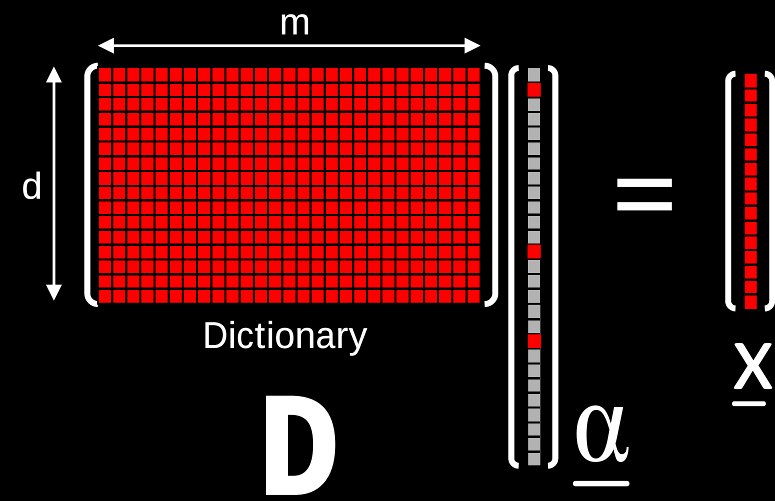
SMALL

Part I - Background

The Synthesis and Analysis Models

The Synthesis Model – Basics

- The synthesis representation is expected to be sparse: $\|\underline{\alpha}\|_0 = k << d$
- Adopting a Bayesian point of view:
  - Draw the support at random
  - Choose the non-zero coefficients randomly (e.g. iid Gaussians)
  - Multiply by **D** to get the synthesis signal



- Such synthesis signals belong to a Union-of-Subspaces (UoS):

$$\underline{x} \in \bigcap_{|T|=k} \text{span}\{\mathbf{D}_T\} \quad \text{where} \quad \mathbf{D}_T \underline{\alpha}_T = \underline{x}$$

- This union contains $\binom{m}{k}$ subspaces, each of dimension k.

The Analysis Model – Basics

- The analysis representation $\underline{z}$ is expected to be sparse

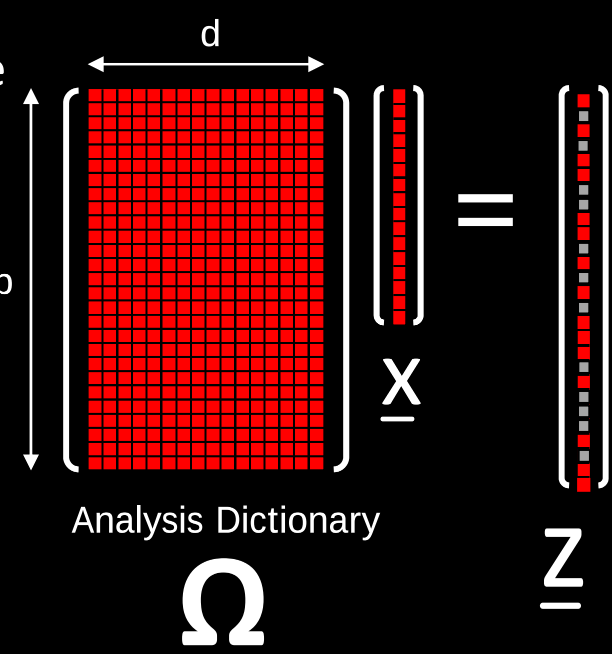
$$\|\Omega \underline{x}\|_0 = \|\underline{z}\|_0 = p - \ell$$

- Co-sparsity: ℓ - the number of zeros in $\underline{z}$.

- Co-Support: Λ - the rows that are orthogonal to $\underline{x}$

$$\Omega_\Lambda \underline{x} = \underline{0}$$

- If Ω is in general position*, then $0 \le \ell < d$ and thus we cannot expect to get a truly sparse analysis representation – Is this a problem? No!



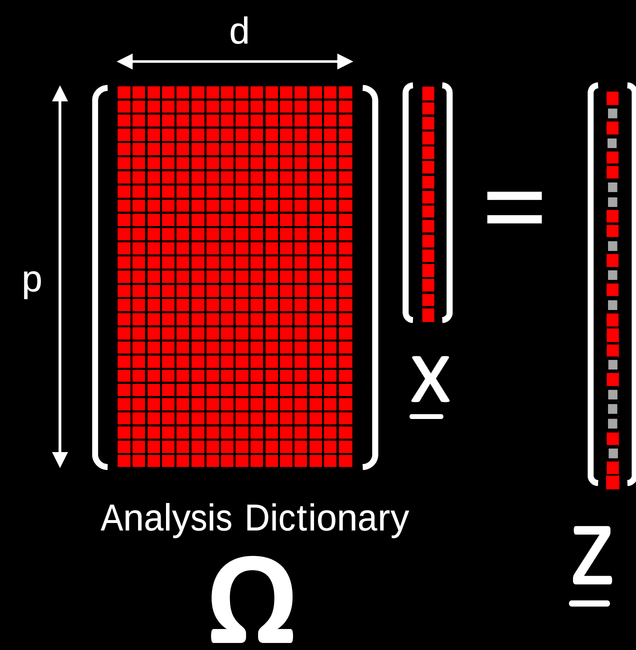
1. S. Nam, M.E. Davies, M. Elad, and R. Gribonval, "Co-sparse Analysis Modeling - Uniqueness and Algorithms" , ICASSP, May, 2011.
2. S. Nam, M.E. Davies, M. Elad, and R. Gribonval, "The Co-sparse Analysis Model and Algorithms" , Submitted to ACHA, June 2011.

* $\text{spark}\{\Omega^T\} = d + 1$

A Bayesian View of These Models

- Analysis signals, just like synthesis ones, can be generated in a systematic way:

	Synthesis Signals	Analysis Signals

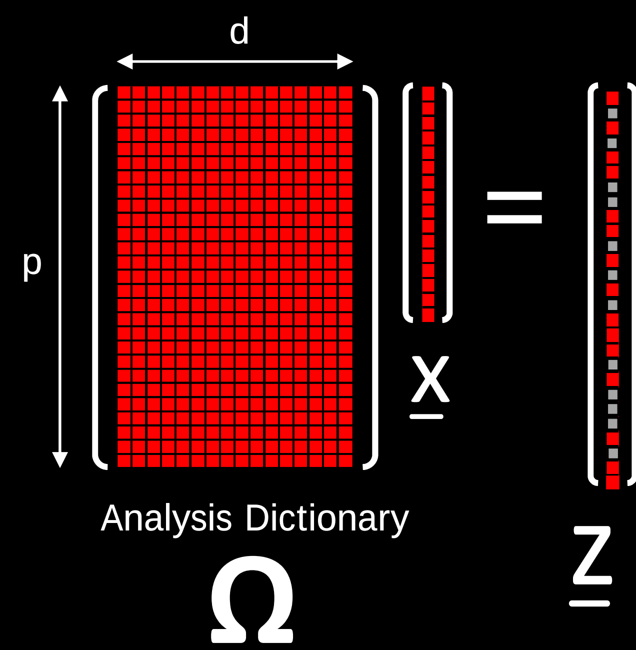


- Bottom line: an analysis signal $\underline{x}$ satisfies: $\exists \left|\Lambda\right| = \ell$ s.t. $\mathbf{\Omega}_{\Lambda}\underline{x} = \underline{0}$

Union-of-Subspaces

- Analysis signals, just like synthesis ones, belong to a union of subspaces:

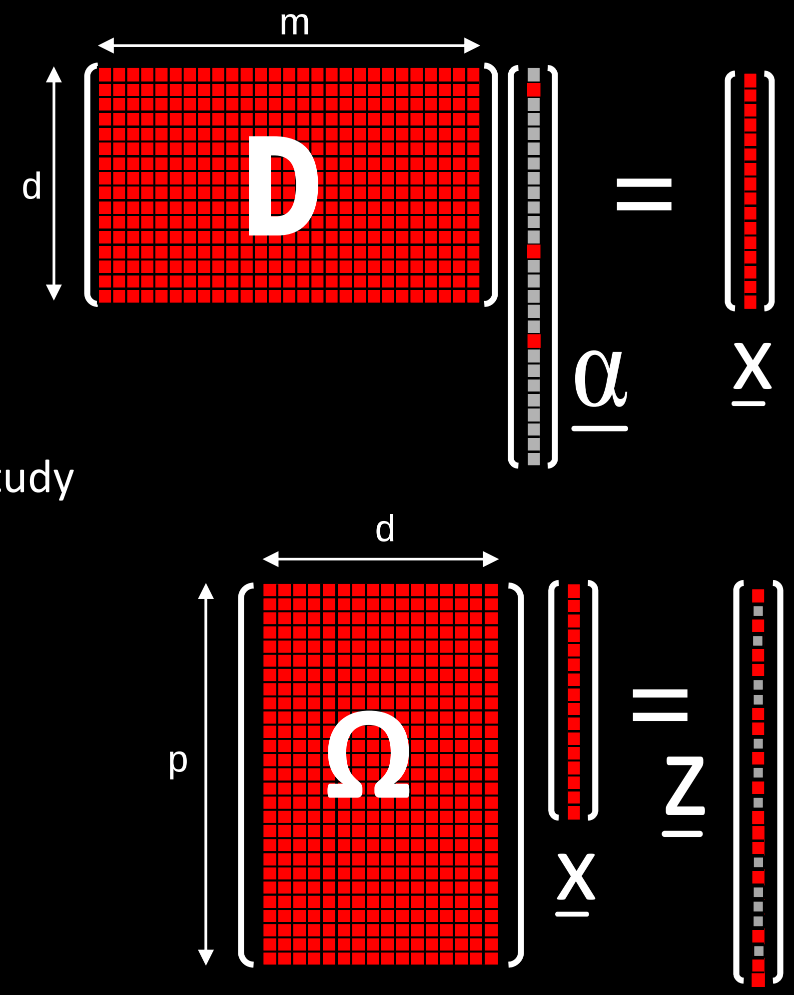
	Synthesis Signals	Analysis Signals



- Example: p=m=2d:
  - Synthesis: k=1 (one atom) – there are 2d subspaces of dimensionality 1
  - Analysis: ℓ=d-1 leads to $\binom{2d}{d-1} >> O(2^d)$ subspaces of dimensionality 1

The Analysis Model – Summary

- The analysis and the synthesis models are similar, and yet very different
- The two align for p=m=d : non-redundant
- Just as the synthesis, we should work on:
  - Pursuit algorithms (of all kinds) – Design
  - Pursuit algorithms (of all kinds) – Theoretical study
  - **Dictionary learning from example-signals**
  - Applications …
- Our experience on the analysis model:
  - Theoretical study is harder
  - Different applications should be considered

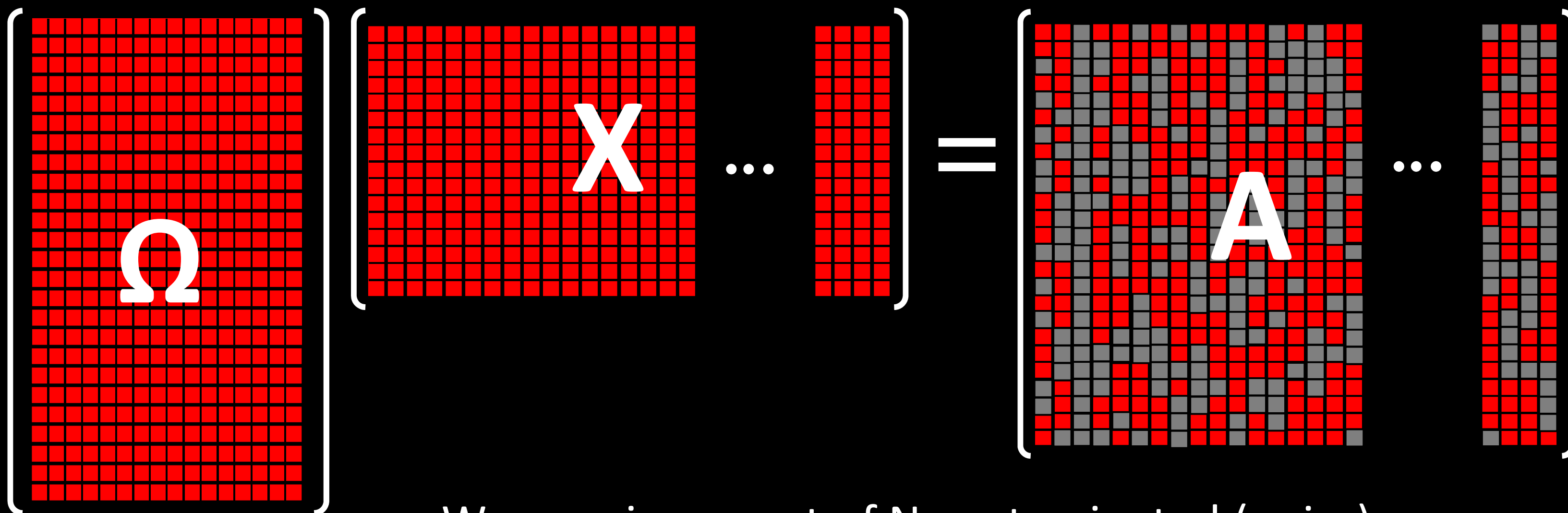


Part II – Dictionaries

Analysis Dictionary-Learning by Sequential Minimal Eigenvalues

1. B. Ophir, M. Elad, N. Bertin and M.D. Plumbley, "Sequential Minimal Eigenvalues - An Approach to Analysis Dictionary Learning", EUSIPCO, August 2011.
2. R. Rubinstein and M. Elad, "A k-SVD Dictionary Learning Algorithm for the Co-sparse Analysis Model" , will be submitted (very) soon to IEEE-TSP ....

Analysis Dictionary Learning – The Signals



We are given a set of N contaminated (noisy) analysis signals, and our goal is to recover their analysis dictionary, $\mathbf{\Omega}$

$$\left\{ \underline{y}_j = \underline{x}_j + \underline{v}_j, \quad \exists \left|\Lambda_j\right| = \ell \ \text{ s.t. } \ \mathbf{\Omega}_{\Lambda_j} \underline{x}_j = \underline{0}, \quad \underline{v} \sim \mathbf{N}\left\{\underline{0}, \sigma^2 \mathbf{I}\right\} \right\}_{j=1}^{N}$$

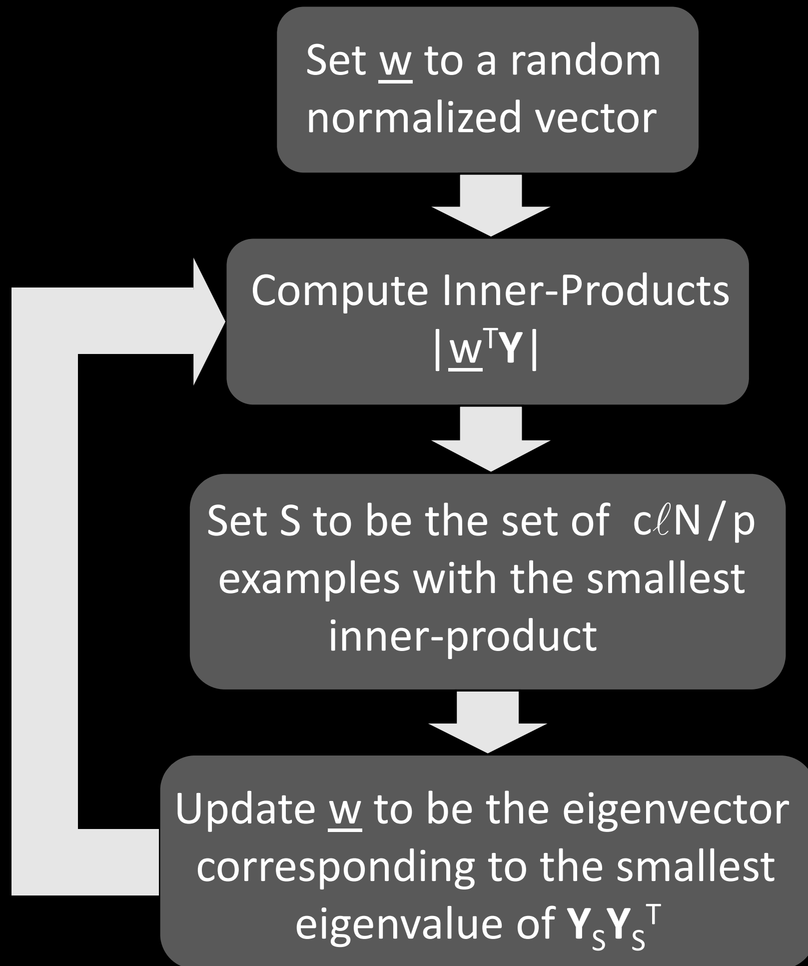
Lets Find a Single Row $\underline{w}$ from Ω

Observations:

1. Any given row is supposed to be orthogonal to $\sim \ell N/p$ signals.
2. If we knew S, the true set of examples that are orthogonal to a row $\underline{w}$, then we could approximate $\underline{w}$ as the solver of

$$\min_{\|\underline{w}\|_2=1} \sum_{j\in S} \left(\underline{w}^T \underline{y}_j\right)^2$$

We shall seek $\underline{w}$ by iteratively solving the above for $\underline{w}$ and S alternately.



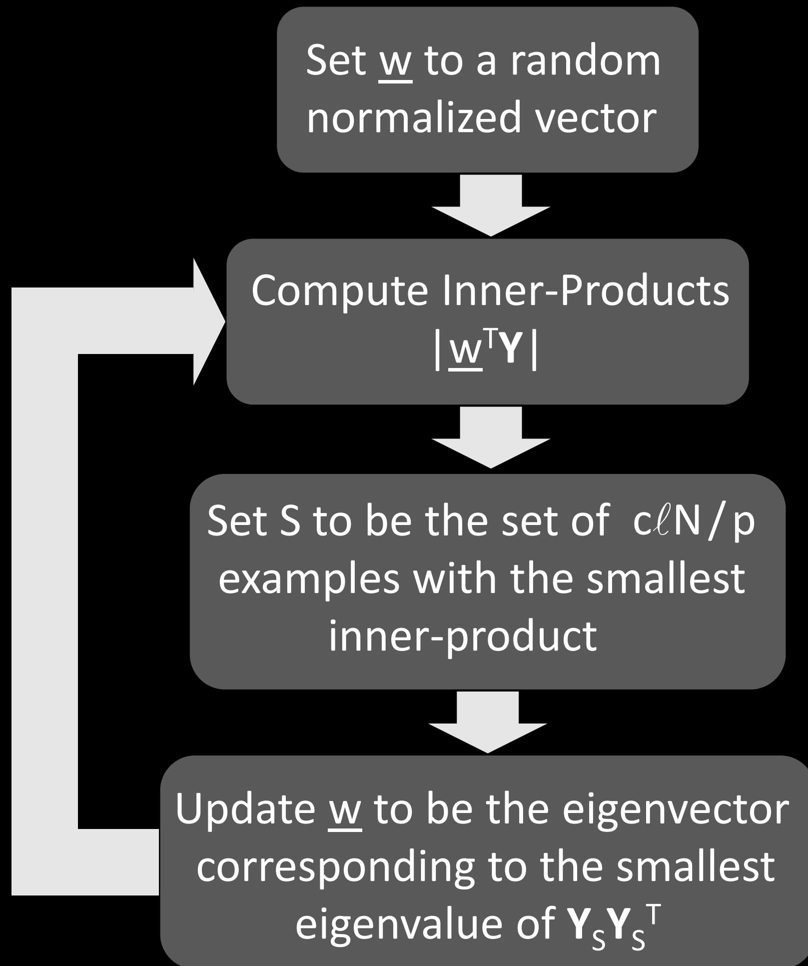
Lets Find a Single Row $\underline{w}$ from Ω

Note:

1. Having found a candidate row $\underline{w}$, it is not necessarily a row from Ω.
2. For a vector to represent a feasible solution, we should require

$$\frac{1}{|S|}\sum_{j\in S}\left(\underline{w}^T\underline{y}_j\right)^2 \approx d\sigma^2$$

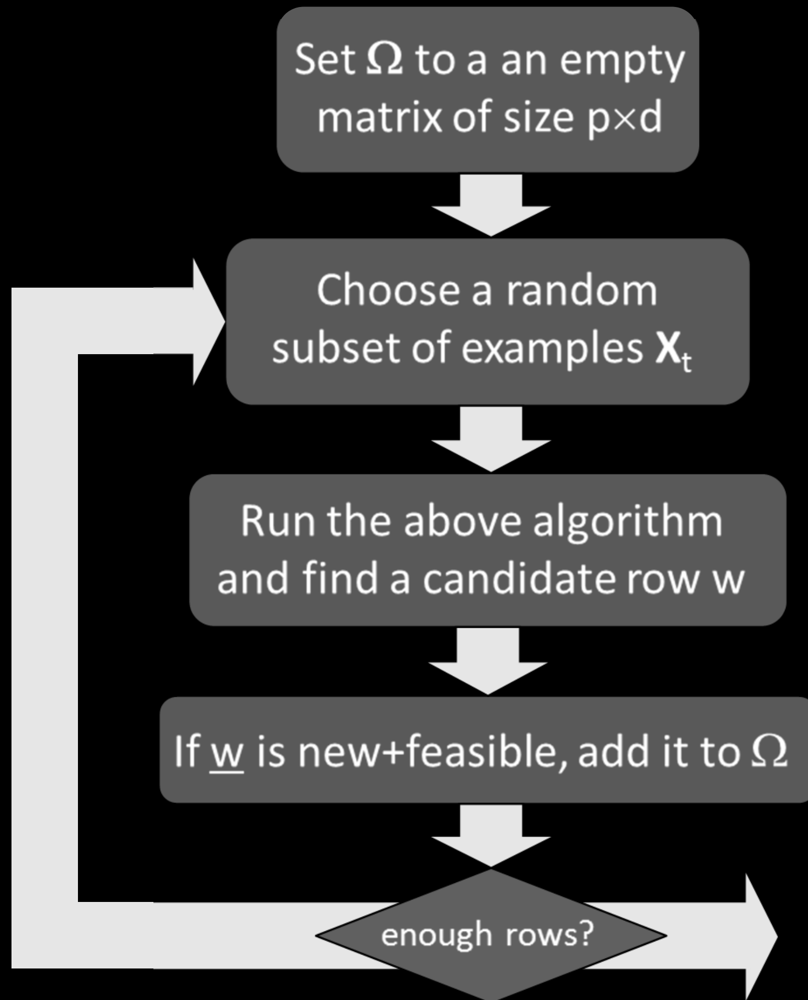
3. Thus, if, after the convergence of this algorithm, this condition is not met, we discard of this estimate.



Finding All Rows in Ω

Observations:

1. The previous algorithm can produce feasible rows from Ω.
2. Repeating the same process may result with different rows, due to the different random initialization.
3. We can increase chances of getting different rows by running the above procedure on a different (randomly selected) subset of the examples.
4. When a row is found several times, this is easily detected, and those repetitions can be pruned.



Results – Synthetic Experiment

Experiment Details:

1. We generate a random analysis dictionary Ω of size p=20, d=10.
2. We generate 10,000 analysis signal examples of dimension d=10 with co-sparsity=8.
3. We consider learning Ω from noiseless and noisy (σ=0.1) signals.

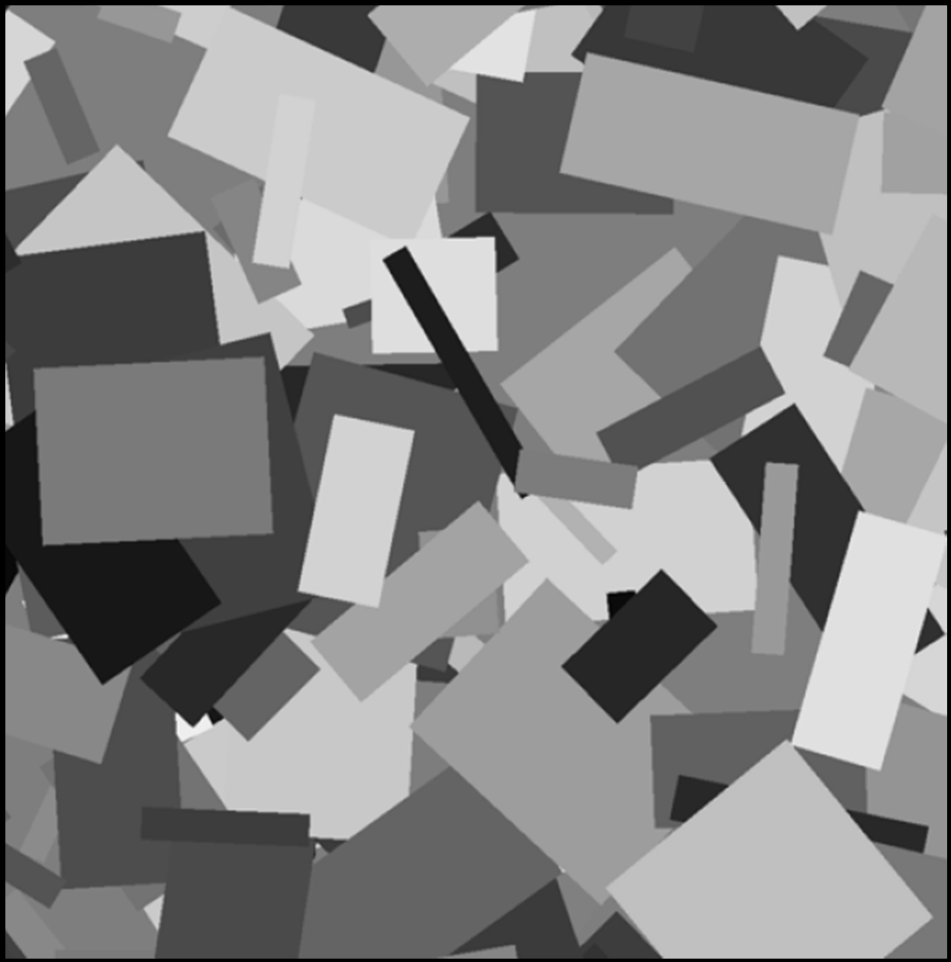
Experiment Results:

1. In the noiseless case, all rows (p=20) were detected to within 1e-8 error (per coefficient). This required ~100 row-estimates.
2. In the noisy case, all rows were detected with an error of 1e-4, this time requiring ~300 row-estimates.

Synthetic Image Data

Experiment Details:

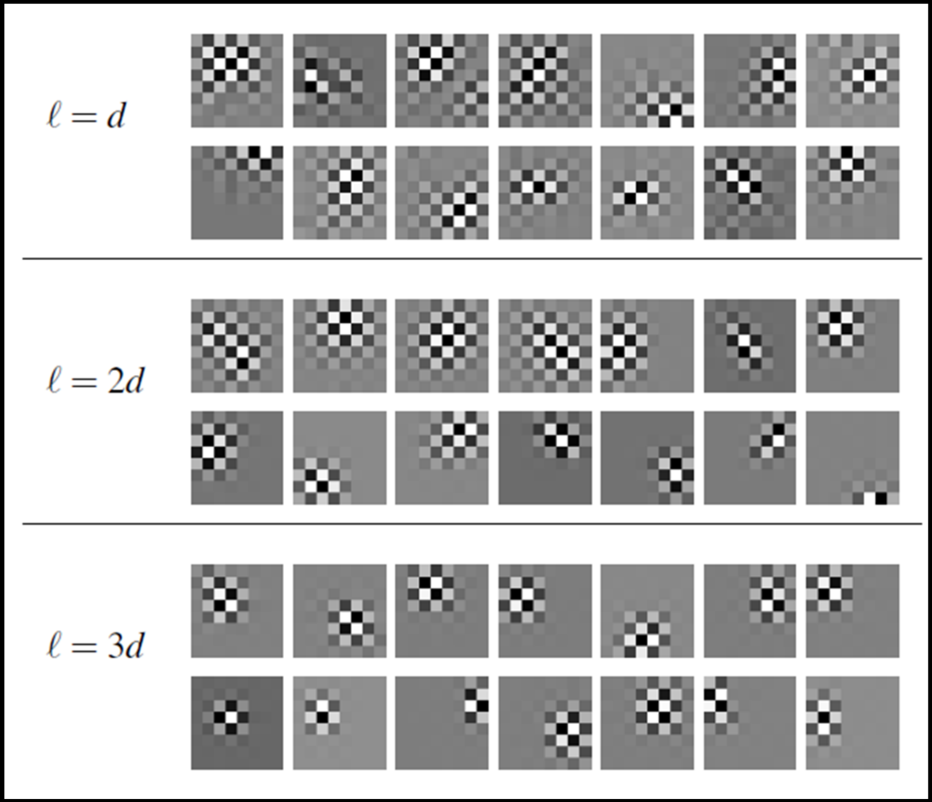
1. We generate a synthetic piece-wise constant image.
2. We consider 8×8 patches from this image as examples to train on. Thus, d=64.
3. We set p=256, and seek an analysis dictionary for these examples.
4. In our training, we stop when the error is σ=10 gray-values.



Synthetic Image Data

Experiment Results:

1. We have to choose the co-sparsity to work with, and it can go beyond d.
2. We see that the results are "random" patterns that are expected to be orthogonal to piece-wise constant patches. They become more localized when the co-sparsity is increased.



True Image Data

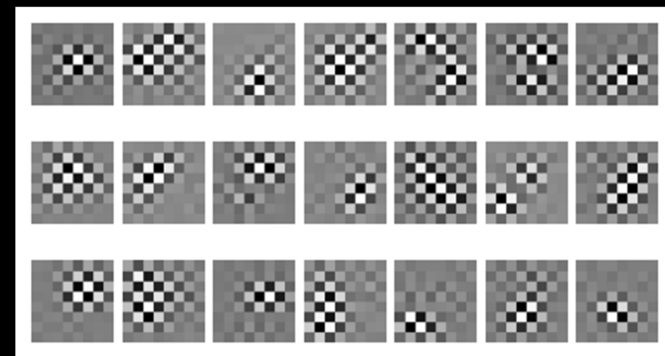
Experiment Details:

1. We take the image "Peppers" to work on.
2. We consider 8×8 patches from this image as examples to train on. Thus, d=64.
3. We set p=256, and seek an analysis dictionary for these examples.
4. In our training, we stop when the error is σ=10 gray-values.

True Image Data

Experiment Results:

1. Few of the found rows are shown here, and they are similar to the previous ones (co-sparsity=3d).
2. We also show the sorted projected values $|\Omega \mathbf{Y}|$ for the obtained dictionary (top) and a random one (bottom). White stands for zero – as can be seen, the trained dictionary leads to better sparsity.



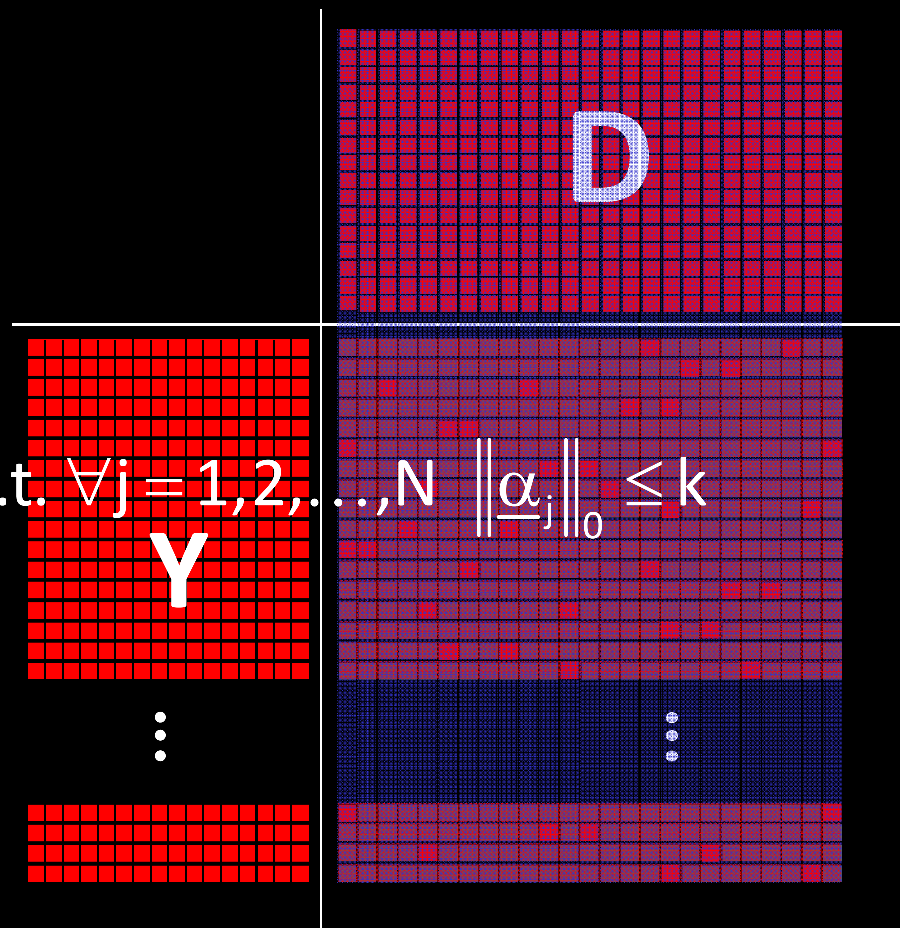
Recall the Synthesis K-SVD Aharon, Elad, & Bruckstein (`04)

Initialize D
e.g. choose a subset of the examples

Sparse Coding
Use OMP or BP

$$\min_{\mathbf{D},\mathbf{A}} \sum_{j=1}^{N} \left\| \mathbf{D}\underline{\alpha}_j - \underline{y}_j \right\|_2^2 \quad \text{s.t. } \forall j = 1,2,\ldots,N \ \ \left\| \underline{\alpha}_j \right\|_0 \le k$$

Dictionary Update
Column-by-Column by SVD computation



Analysis K-SVD

Rubinstein and Elad (`11)

Synthesis

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

Analysis

$$\underset{\boldsymbol{\Omega},\underline{\mathbf{X}}}{\text{Min}} \sum_{j=1}^{N} \left\| \underline{x}_j - \underline{y}_j \right\|_2^2 \quad \text{s.t. } \forall j=1,2,\ldots,N \quad \left\| \boldsymbol{\Omega}\underline{x}_j \right\|_0 \le p-\ell$$

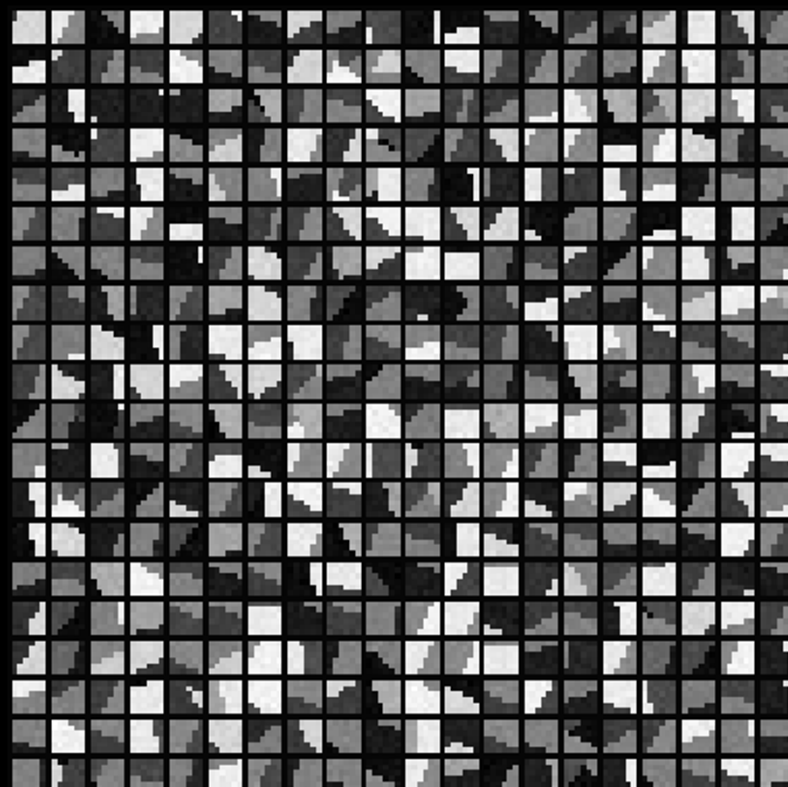
We adopt a similar approach to the K-SVD for approximating the minimization of the analysis goal, by iterating between the search for $\underline{x}_j$ and an update of the rows of Ω

Analysis Dictionary Learning – Results

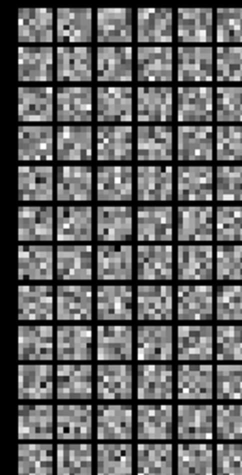
Experiment: Piece-Wise Constant Image

- ❑ We take 10,000 patches (+noise $\sigma=5$) to train on
- ❑ Here is what we got:

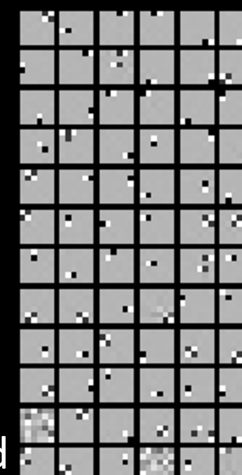
Original Image



Patches used for training



Initial Ω

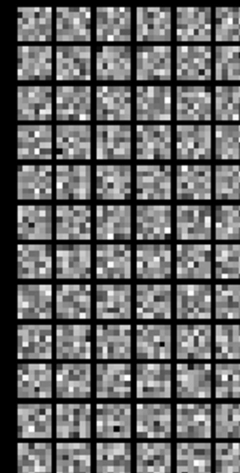


Trained (100 iterations) Ω

Analysis Dictionary Learning – Results

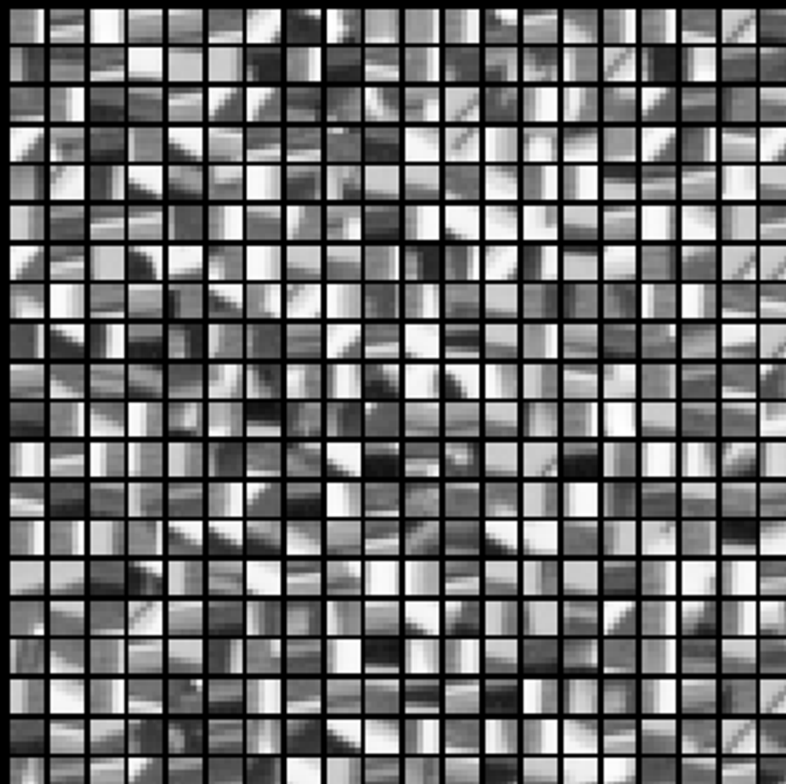
Experiment: The Image "House"

- We take 10,000 patches (+noise $\sigma=10$) to train on
- Here is what we got:

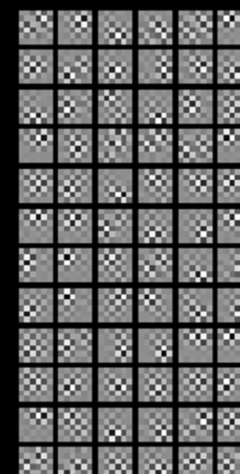


Initial Ω

Original Image



Patches used for training



Trained (100 iterations) Ω

Part III – We Are Done

Summary and Conclusions

Today We Have Seen that …

Sparsity and Redundancy are practiced mostly in the context of the synthesis model

Is there any other way?

Yes, the analysis model is a very appealing (and different) alternative, worth looking at

So, what to do?

In the past few years there is a growing interest in better defining this model, suggesting pursuit methods, analyzing them, etc.

What about Dictionary learning?

We propose new algorithms for this task. The next step is applications that will benefit from this

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