Recent Results on the Co-Sparse Analysis Model

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

The research leading to these results has received funding from:
- The European Council under the European union's Seventh Framework Programme (FP/2007-2013) ERC grant Agreement ERC-SPARSE- 320649
- Google Faculty Research Award

*Joint work with
Ron Rubinstein  Tomer Peleg
& Remi Gribonval, Sangnam Nam, Mark Plumbley, Mike Davies, Raja Giryes, Boaz Ophir, Nancy Bertin

NIPS Workshop – December 7th, 2012
It does not matter what is the data you are working on – if it carries information, it must have an inner **structure**.

This **structure** = **rules** the data complies with.

Signal/image processing relies on exploiting these “**rules**” by adopting **models**.

A **model** = mathematical construction describing the properties of the signal.

In the past decade, sparsity-based models has been drawing major attention.
The Co-Sparse Analysis Model: Recent Results
By: Michael Elad

Sparsity and Redundancy can be Practiced in (at least) two different ways

Synthesis

Analysis

The attention to sparsity-based models has been mostly given to the synthesis option, leaving the analysis almost untouched.

This Talk’s Message: The co-sparse analysis model is a very appealing alternative to the synthesis model, it has a great potential for signal modeling.
Agenda

Part I - Background
  Recalling the Synthesis Sparse Model

Part II - Analysis
  Turning to the Analysis Model

Part III – THR Performance
  Revealing Important Dictionary Properties

Part IV – Dictionaries
  Analysis Dictionary-Learning and Some Results

Part V – We Are Done
  Summary and Conclusions
Part I - Background
Recalling the Synthesis Sparse Model
The Sparsity-Based Synthesis Model

- We assume the existence of a synthesis dictionary $D \in \mathbb{R}^{d \times n}$ whose columns are the atom signals.
- Signals are modeled as sparse linear combinations of the dictionary atoms:
  \[ x = D\alpha \]
- We seek a sparsity of $\alpha$, meaning that it is assumed to contain mostly zeros.
- We typically assume that $n > d$: redundancy.
- This model is typically referred to as the synthesis sparse and redundant representation model for signals.
The Synthesis Model – Basics

- The synthesis representation is expected to be sparse: \( \| \alpha \|_0 = k << d \)

- Adopting a Bayesian point of view:
  - Draw the support \( T \) (with \( k \) non-zeroes) at random;
  - Choose the non-zero coefficients randomly (e.g. iid Gaussians); and
  - Multiply by \( D \) to get the synthesis signal.

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

  \[ x \in \bigcup_{|T|=k} \text{span}\{ D_T \} \quad \text{where} \quad D_T \alpha_T = x \]

- This union contains \( \binom{n}{k} \) subspaces, each of dimension \( k \).
The Synthesis Model – Pursuit

- Fundamental problem: Given the noisy measurements,
  \[ y = x + v = D\alpha + v, \quad v \sim N(0, \sigma^2 I) \]
  recover the clean signal \( x \) – This is a denoising task.

- This can be posed as:
  \[ \hat{\alpha} = \text{ArgMin}_{\alpha} \| y - D\alpha \|_2^2 \text{ s.t. } \|\alpha\|_0 = k \Rightarrow \hat{x} = D\hat{\alpha} \]

- While this is a (NP-) hard problem, its approximated solution can be obtained by
  - Use \( L_1 \) instead of \( L_0 \) (Basis-Pursuit)
  - Greedy methods (MP, OMP, LS-OMP)
  - Hybrid methods (IHT, SP, CoSaMP).

- Theoretical studies provide various guarantees for the success of these techniques, typically depending on \( k \) and properties of \( D \).
The Synthesis Model – Dictionary Learning

Given Signals: \( \{ y_j = x_j + v_j \} \), \( v_j \sim \mathcal{N}(0, \sigma^2 I) \), \( j = 1, 2, \ldots, N \)

Minimize
\[
\min_{D,A} \left\| DA - Y \right\|_F^2
\]
subject to
\[
\forall j = 1, 2, \ldots, N \quad \left\| \alpha_j \right\|_0 \leq k
\]

Example are linear combinations of atoms from \( D \)
Each example has a sparse representation with no more than \( k \) atoms

Field & Olshausen (’96)
Engan et. al. (’99)
... Gribonval et. al. (’04)
Aharon et. al. (’04)
...
Part II - Analysis

Turning to the Analysis Model

The Analysis Model – Basics

- The analysis representation $z$ is expected to be sparse
  \[ \| \Omega x \|_0 = \| z \|_0 = p - \ell \]

- Co-sparsity: $\ell$ - the number of zeros in $z$.

- Co-Support: $\Lambda$ - the rows that are orthogonal to $x$
  \[ \Omega_{\Lambda} x = 0 \]

- This model puts an emphasis on the zeros in $z$ for characterizing the signal, just like zero-crossings of wavelets used for defining a signal [Mallat (’91)].

- Co-Rank: \( \text{Rank}(\Omega_{\Lambda}) \leq \ell \) (strictly smaller if there are linear dependencies in $\Omega$).

- If $\Omega$ is in general position*, then the co-rank and the co-sparsity are the same, and \( 0 \leq \ell < d \), implying that we cannot expect to get a truly sparse analysis.

\* $\text{spark}\{\Omega^T\} = d + 1$
Analysis signals, just like synthesis ones, can be generated in a systematic way:

| Support: | Choose the support $T (|T| = k)$ at random |
| Coef.: | Choose $\alpha_T$ at random |
| Generate: | Synthesize by: $D_T \alpha_T = x$ |

Bottom line: an analysis signal $x$ satisfies: $\exists \Lambda \mid |\Lambda| = \ell$ s.t. $\Omega_{\Lambda}x = 0$. 
The Analysis Model – UoS

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

<table>
<thead>
<tr>
<th></th>
<th>Synthesis Signals</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the Subspace Dimension:</td>
<td>k</td>
</tr>
<tr>
<td>How Many Subspaces:</td>
<td>( \binom{n}{k} )</td>
</tr>
<tr>
<td>Who are those Subspaces:</td>
<td>( \text{span}{D_T} )</td>
</tr>
</tbody>
</table>

- The analysis and the synthesis models offer both a UoS construction, but these are very different from each other in general.
The Analysis Model – Count of Subspaces

- Example: \( p=n=2d \):
  - Synthesis: \( k=1 \) (one atom) – there are \( 2d \) subspaces of dimensionality 1.
  - Analysis: \( \ell = d-1 \) leads to \( \binom{2d}{d-1} \gg O(2^d) \) subspaces of dimensionality 1.

- In the general case, for \( d=40 \) and \( p=n=80 \), this graph shows the count of the number of subspaces. As can be seen, the two models are substantially different, the analysis model is much richer in low-dim., and the two complete each other.

- The analysis model tends to lead to a richer UoS. Are these good news?
The Low-Spark $\Omega$ Case

- What if $\text{spark}(\Omega^T) << d$?
- For example: a TV-like operator for image-patches of size $6 \times 6$ pixels ($\Omega$ size is $72 \times 36$).
- Here are analysis-signals generated for co-sparsity ($\ell$) of 32:
- Their true co-sparsity is higher – see graph:
- In such a case we may consider $\ell > d$, and thus...
  ... the number of subspaces is smaller.

$\Omega_{\text{DIF}} = \begin{bmatrix}
\text{Horizontal Derivative} \\
\text{Vertical Derivative}
\end{bmatrix}$
The Analysis Model – Pursuit

- Fundamental problem: Given the noisy measurements,
  \[ y = x + v, \quad \exists |\Lambda| = \ell \quad \text{s.t.} \quad \Omega_\Lambda x = 0, \quad v \sim \mathcal{N}\{0, \sigma^2 I\} \]
  recover the clean signal \( \hat{x} \) – This is a denoising task.

- This goal can be posed as:
  \[ \hat{x} = \text{ArgMin}_{x,\Lambda} \|y - x\|_2^2 \quad \text{s.t.} \quad \Omega_\Lambda x = 0 \quad \& \quad |\Lambda| = p - \ell \quad \text{or} \quad \text{rank}(\Omega_\Lambda) = d - r \]

- This is a (NP-) hard problem, just as in the synthesis case.

- We can approximate its solution by \( L_1 \) replacing \( L_0 \) (BP-analysis), Greedy methods (BG, OBG, GAP), and Hybrid methods (AIHT, ASP, ACoSaMP, ...).

- Theoretical study providing pursuit guarantees depending on the co-sparsity and properties of \( \Omega \). See [Candès, Eldar, Needell, & Randall (’10)], [Nam, Davies, Elad, & Gribonval, (’11)], [Vaiter, Peyré, Dossal, & Fadili, (’11)], [Peleg & Elad (’12)].
The Analysis Model – Backward Greedy

BG finds one row at a time from \( \Lambda \) for approximating the solution of

\[
\hat{x} = \text{argmin}_{x \in \Lambda} \left\| y - x \right\|_2^2 \quad \text{s.t.} \quad \{ \Omega_{\Lambda} x = 0 \quad \& \quad \text{Rank}(\Omega_{\Lambda}) = d - r \}
\]

i = 0, \( \hat{x}_0 = y \), \( \Lambda_0 = \{ \} \)

Stop condition?
(e.g. \( \text{Rank}(\Omega_{\Lambda}) = d - r \))

Yes
Output \( x_i \)

No

i = i + 1, \( \Lambda_i = \Lambda_{i-1} \cup \text{argmin}_{k \not\in \Lambda_{i-1}} \left| w_k^T \hat{x}_{i-1} \right| \)

\[
\hat{x}_i = \left[ I - \Omega_{\Lambda_i}^\dagger \Omega_{\Lambda_i} \right] y
\]
The Analysis Model – Backward Greedy

Is there a similarity to a synthesis pursuit algorithm?

Other options:
- Optimized BG pursuit (OBG) [Rubinstein, Peleg & Elad (‘12)]
- Greedy Analysis Pursuit (GAP) [Nam, Davies, Elad & Gribonval (‘11)]
- Iterative Cosparse Projection [Giryes, Nam, Gribonval & Davies (‘11)]
- $L_p$ relaxation using IRLS [Rubinstein (‘12)]
- CoSAMP/SP like algorithms [Giryes, et. al. (‘12)]
- Analysis-THR [Peleg & Elad (‘12)]
Synthesis vs. Analysis – Summary

- The two align for $p=n=d$: 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 work on the analysis model so far touched on all the above. In this talk we shall focus on:
  - A theoretical study of the simplest pursuit method: Analysis-THR.
  - Developing a K-SVD like dictionary learning method for the analysis model.
Part III – THR Performance

Revealing Important Dictionary Properties

The Analysis-THR Algorithm

Analysis-THR aims to find an approximation for the problem

\[ \hat{x} = \arg\min_{{x, \Lambda}} \|y - x\|_2^2 \text{ s.t. } \{\Omega_{\Lambda} x = 0 \land \text{Rank}(\Omega_{\Lambda}) = d-r\} \]

Compute \( z = |\Omega y| \) & sort (increasing) \( \rightarrow \{\gamma_k\}_{k=1}^p \)

Stop condition? \( \text{Rank}(\Omega_{\Lambda_i}) = d-r \)

Yes

Output
\[ \hat{x} = \left[ I - \Omega_{\Lambda_i}^+ \Omega_{\Lambda_i} \right] y \]

No

\( i = i + 1, \Lambda_i = \Lambda_{i-1} \cup \gamma_i \)

\( i = 0, \Lambda_0 = \{\} \)
The Restricted Ortho. Projection Property

$$\alpha_r = \min_{\Lambda,j} \left| \text{Rank}(\Omega_{\Lambda}) = d - r \right| \left\| (I - \Omega_{\Lambda}^+ \Omega_{\Lambda}) w_j \right\|_2$$

- ROPP aims to get near orthogonality of rows outside the co-support (i.e., $\alpha_r$ should be as close as possible to 1).
- This should remind of the (synthesis) ERC [Tropp ('04)]:

$$\max_{s,j \mid |S| = k \text{ and } j \not\in S} \left\| D_s^+ d_j \right\|_1 \leq 1$$
Theoretical Study of the THR Algorithm

Choose \( \Omega \in \mathbb{R}^{p \times d} \)
Choose \( \Lambda \)
Such that
\[
\text{Rank}(\Omega_\Lambda) = d - r \\
|\Lambda| = \ell
\]
Generate \( e \sim \mathcal{N}(0, \sigma^2 I) \in \mathbb{R}^d \)
Generate \( u \sim \mathcal{N}(0, I) \in \mathbb{R}^d \)
Project \( x = (I - \Omega_\Lambda^+ \Omega_\Lambda)u \)
Choose \( \alpha_r \)
\[
\text{Pr}\{\text{success (i.e. } \hat{\Lambda} = \Lambda)\}
\]

The Analysis THR Algorithm

Co-Rank \( r \)
\( \Omega \)
\( e \rightarrow y \)
\( x \rightarrow \hat{x} \)
\( \Lambda \rightarrow \hat{\Lambda} \)
As empirical tests show, the theoretical performance predicts an improvement for an $\Omega$ with strong linear dependencies, and high ROPP.
Part IV – Dictionaries

Analysis Dictionary-Learning and Some Results

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, $\Omega$

\[
\{y_j = x_j + v_j, \quad \exists |\Lambda_j| = \ell \quad \text{s.t.} \quad \Omega_{\Lambda_j} x_j = 0, \quad v \sim \mathcal{N}(0, \sigma^2) \}\}_{j=1}^N
\]
Analysis Dictionary Learning – Goal

We shall adopt a similar approach to the K-SVD for approximating the minimization of the analysis goal.
Analysis K-SVD – Outline [Rubinstein, Peleg & Elad (’12)]

\[ \begin{bmatrix} \Omega \\ \vdots \end{bmatrix} X = \begin{bmatrix} \Omega \\ \vdots \end{bmatrix} Z \]

- Initialize \( \Omega \)
- Sparse Code
- Dictionary Update
Analysis K-SVD – Sparse-Coding Stage

\[
\begin{align*}
\Omega & \quad X \quad \cdots \\
\Omega & \quad X \quad \cdots \\
\Omega & \quad X \quad \cdots \\
\end{align*}
\]

\[
\text{Min}_{\Omega, X} \left\| X - Y \right\|_F^2 \quad \text{s.t.} \quad \forall j = 1, 2, \ldots, N \quad \| \Omega x_j \|_0 \leq p - \ell
\]

Assuming that \( \Omega \) is fixed, we aim at updating \( X \)

\[
\left\{ \hat{x}_j = \text{ArgMin}_{x} \left\| x - y_j \right\|_2^2 \quad \text{s.t.} \quad \| \Omega x \|_0 \leq p - \ell \right\}_{j=1}^{N}
\]

These are \( N \) separate analysis-pursuit problems. We suggest to use the BG or the OBG algorithms.
Analysis K-SVD – Dictionary Update Stage

\[
\begin{pmatrix}
\Omega \\
\end{pmatrix}
\begin{pmatrix}
X \\
\end{pmatrix}
= 
\begin{pmatrix}
Z \\
\end{pmatrix}
\]

Min \( \Omega, x \) \( \| X - Y \|_F^2 \) s.t. \( \forall j = 1, 2, \ldots, N \) \( \| \Omega x_j \|_0 \leq p - \ell \)

- Only signals orthogonal to the atom should get to vote for its new value.
- The known supports should be preserved.
- Improved results for applications are obtained by promoting linear dependencies within \( \Omega \).
Experiment #1: Piece-Wise Constant Image

- We take 10,000 6x6 patches (+noise $\sigma=5$) to train on.

- Here is what we got (we promote sparse outcome):

  Initial $\Omega$

  Trained (100 iterations) $\Omega$

Original Image

Patches used for training
Experiment #2: denoising of the piece-wise constant image.

256×256

Non-flat patch examples
Analysis Dictionary Learning – Results (2)

Learned dictionaries for $\sigma=5$

**Analysis Dictionary**

- 38 atoms
- (again, promoting sparsity in $\Omega$)

**Synthesis Dictionary**

- 100 atoms
### Analysis Dictionary Learning – Results (2)

<table>
<thead>
<tr>
<th></th>
<th>BM3D</th>
<th>Synthesis K-SVD</th>
<th>Sparse Analysis K-SVD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average subspace dimension</td>
<td>n/a</td>
<td>2.42</td>
<td>2.03</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.79</td>
<td>1.69</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.79</td>
<td>1.69</td>
</tr>
<tr>
<td>Patch denoising: error per element</td>
<td>n/a</td>
<td>2.91</td>
<td>5.37</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.57</td>
<td>10.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6.81</td>
<td>9.62</td>
</tr>
<tr>
<td>Image PSNR [dB]</td>
<td>40.66</td>
<td>43.68</td>
<td>46.02</td>
</tr>
<tr>
<td></td>
<td>32.23</td>
<td>34.83</td>
<td>35.03</td>
</tr>
<tr>
<td></td>
<td>35.44</td>
<td>38.13</td>
<td>39.13</td>
</tr>
<tr>
<td></td>
<td>30.32</td>
<td>32.02</td>
<td>31.97</td>
</tr>
</tbody>
</table>

**Cell Legend:**

<table>
<thead>
<tr>
<th></th>
<th>(\sigma=5)</th>
<th>(\sigma=10)</th>
<th>(\sigma=15)</th>
<th>(\sigma=20)</th>
</tr>
</thead>
</table>

The Co-Sparse Analysis Model: Recent Results
By: Michael Elad
Experiment #3: denoising of natural images (with $\sigma=5$)

The following results were obtained by modifying the DL algorithm to improve the ROPP

<table>
<thead>
<tr>
<th>Method</th>
<th>Barbara</th>
<th>House</th>
<th>Peppers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields of Experts</td>
<td>37.19 dB</td>
<td>38.23 dB</td>
<td>27.63 dB</td>
</tr>
<tr>
<td>Synthesis K-SVD</td>
<td>38.08 dB</td>
<td>39.37 dB</td>
<td>37.78 dB</td>
</tr>
<tr>
<td>Analysis K-SVD</td>
<td>37.75 dB</td>
<td>39.15 dB</td>
<td>37.89 dB</td>
</tr>
</tbody>
</table>

An Open Problem: How to “Inject” linear dependencies into the learned dictionary?
Part V – We Are Done

Summary and Conclusions
Today ... 

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

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

Is there any other way?

In the past few years there is a growing interest in this model, deriving pursuit methods, analyzing them, designing dictionary-learning, etc.

So, what to do?

Today we discussed:

- The differences between the two models,
- A theoretical study of the THR algorithm, &
- Dictionary learning for the analysis model.

These slides and the relevant papers can be found in http://www.cs.technion.ac.il/~elad
Thank you for your time, and ... 

Thanks to the organizers: 
Martin Kleinsteuber
Francis Bach
Remi Gribonval
John Wright
Simon Hawe

Questions?
The Analysis Model – The Signature

Consider two possible dictionaries:

\[ \Omega_{\text{DIF}} \]

\[ \text{Spark}(\Omega^T) = 4 \]

\[ \text{Random } \Omega \]

\[ \text{Spark}(\Omega^T) = 37 \]

The Signature of a matrix is more informative than the Spark. Is it enough?

The Co-Sparse Analysis Model: Recent Results
By: Michael Elad