Sparse Modeling in Image Processing and Deep Learning

Michael Elad

Computer Science Department
The Technion - Israel Institute of Technology
Haifa 32000, Israel

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 Lecture

Sparseland
Sparse Representation Theory

CSC
Convolutional Sparse Coding

ML-CSC
Multi-Layered Convolutional Sparse Coding

Sparsity-Inspired Models  Deep-Learning

Another underlying idea that will accompany us

Generative modeling of data sources enables

- A systematic algorithm development, &
- A theoretical analysis of their performance
Multi-Layered Convolutional Sparse Modeling
Our Data is Structured

- We are surrounded by various diverse sources of massive information
- Each of these sources have an internal structure, which can be exploited
- This structure, when identified, is the engine behind our ability to process this data
Effective removal of noise (and many other tasks) relies on a proper modeling of the signal.

Fact 1: This signal contains AWGN $\mathbb{N}(0,1)$

Fact 2: The clean signal is believed to be PWC
Which Model to Choose?

- A model: a mathematical description of the underlying signal of interest, describing our beliefs regarding its structure.

- The following is a partial list of commonly used models for images:
  - Principal-Component-Analysis
  - Gaussian-Mixture
  - Markov Random Field
  - Laplacian Smoothness
  - DCT concentration
  - Wavelet Sparsity
  - Piece-Wise-Smoothness
  - C2-smoothness
  - Besov-Spaces
  - Total-Variation
  - Beltrami-Flow

- Good models should be simple while matching the signals.

- Simplicity ↔ Reliability

- Models are almost always imperfect.
An Example: JPEG and DCT

How & why does it work?

The model assumption: after DCT, the top left coefficients to be dominant and the rest zeros
The fields of signal & image processing are essentially built of an evolution of models and ways to use them for various tasks.

A New Research Work (and Paper) is Born

Michael Elad
The Computer-Science Department
The Technion
What This Talk is all About?

Data Models and Their Use

- Almost any task in data processing requires a model – true for denoising, deblurring, super-resolution, inpainting, compression, anomaly-detection, sampling, recognition, separation, and more

- Sparse and Redundant Representations offer a new and highly effective model – we call it Sparseland

- We shall describe this and descendant versions of it that lead all the way to ... deep-learning

Michael Elad
The Computer-Science Department
The Technion
Multi-Layered Convolutional Sparse Modeling
A New Emerging Model

Sparseland

Signal Processing
- Wavelet Theory
- Multi-Scale Analysis
- Signal Transforms

Machine Learning

Mathematics
- Approximation Theory
- Linear Algebra
- Optimization Theory

Semi-Supervised Learning
Compression
Recognition
Clustering
Identification
Interpolation
Inference (solving inverse problems)
Source-Separation
Prediction
Denoising
Segmentation
Classification
Sensor-Fusion
Summarizing
Synthesis
Anomaly detection
The Sparseland Model

- Task: model image patches of size $8 \times 8$ pixels
- We assume that a dictionary of such image patches is given, containing 256 atom images
- The Sparseland model assumption: every image patch can be described as a linear combination of few atoms

Michael Elad
The Computer-Science Department
The Technion
The *Sparseland* Model

Properties of this model:

**Sparsity and Redundancy**

- We start with a 8-by-8 pixels patch and represent it using 256 numbers
  - This is a redundant representation

- However, out of those 256 elements in the representation, only 3 are non-zeros
  - This is a sparse representation

- Bottom line in this case: 64 numbers representing the patch are replaced by 6 (3 for the indices of the non-zeros, and 3 for their entries)
Chemistry of Data

We could refer to the *Sparseland* model as the *chemistry* of information:

- Our dictionary stands for the **Periodic Table** containing all the elements
- Our model follows a similar rationale: Every molecule is built of **few** elements
\textbf{Sparseland}: A Formal Description

- Every column in $\mathbf{D}$ (dictionary) is a prototype signal (atom)
- The vector $\alpha$ is generated with few non-zeros at arbitrary locations and values
- This is a generative model that describes how (we believe) signals are created
Difficulties with *Sparseland*

- **Problem 1:** Given a signal, how can we find its atom decomposition?

- A simple example:
  - There are 2000 atoms in the dictionary
  - The signal is known to be built of 15 atoms

  \[
  \binom{2000}{15} \approx 2.4e+37 \text{ possibilities}
  \]
  - If each of these takes 1 nano-sec to test, will take \(~7.5e20\) years to finish !!!!!!!

- So, are we stuck?
Atom Decomposition Made Formal

\[ \min_\alpha \|\alpha\|_0 \quad \text{s.t.} \quad x = D\alpha \]

\[ \min_\alpha \|\alpha\|_0 \quad \text{s.t.} \quad \|D\alpha - y\|_2 \leq \varepsilon \]

Approximation Algorithms

- Relaxation methods
  - Basis-Pursuit
- Greedy methods
  - Thresholding/OMP

- \( L_0 \) – counting number of non-zeros in the vector
- This is a projection onto the *Sparseland* model
- These problems are known to be NP-Hard problem
Pursuit Algorithms

\[
\begin{align*}
\min_\alpha & \| \alpha \|_0 \quad \text{s.t.} \quad \| D \alpha - y \|_2 \leq \varepsilon \\
\min_\alpha & \| \alpha \|_1 \quad \text{s.t.} \quad \| D \alpha - y \|_2 \leq \varepsilon
\end{align*}
\]

Basis Pursuit

Matching Pursuit

Thresholding

Change the \( L_0 \) into \( L_1 \) and then the problem becomes convex and manageable

Find the support greedily, one element at a time

Multiply \( y \) by \( D^T \) and apply shrinkage:

\[ \hat{\alpha} = \mathcal{P}_\beta \{ D^T y \} \]
Difficulties with *Sparseland*

- There are various pursuit algorithms
- Here is an example using the Basis Pursuit (L\(_1\)):

Surprising fact: Many of these algorithms are often accompanied by *theoretical guarantees* for their success, if the unknown is sparse enough.
The Mutual Coherence

- Compute
  
  \[ \begin{bmatrix} D^T & D \end{bmatrix} = D^T D \]
  
  Assume normalized columns

- The **Mutual Coherence** \( \mu(D) \) is the largest off-diagonal entry in absolute value

- We will pose **all the theoretical results in this talk** using this property, due to its simplicity

- You may have heard of other ways to characterize the dictionary (Restricted Isometry Property - RIP, Exact Recovery Condition - ERC, Babel function, Spark, ...)

Michael Elad
The Computer-Science Department
The Technion
Basis-Pursuit Success

Theorem: **Given** a noisy signal $y = D\alpha + v$ where $\|v\|_2 \leq \varepsilon$ and $\alpha$ is sufficiently sparse, then

$\|\alpha\|_0 < \frac{1}{4} \left( 1 + \frac{1}{\mu} \right)$

leads to a stable result:

$\|\hat{\alpha} - \alpha\|_2^2 \leq \frac{4\varepsilon^2}{1 - \mu(4\|\alpha\|_0 - 1)}$

Comments:
- If $\varepsilon = 0 \rightarrow \hat{\alpha} = \alpha$
- This is a worst-case analysis – better bounds exist
- Similar theorems exist for many other pursuit algorithms

Donoho, Elad & Temlyakov ('06)
Difficulties with *Sparseland*

- Problem 2: Given a family of signals, how do we find the dictionary to represent it well?

- Solution: **Learn!** Gather a large set of signals (many thousands), and find the dictionary that sparsifies them.

- Such algorithms were developed in the past 10 years (e.g., K-SVD), and their performance is surprisingly good.

- We will not discuss this matter further in this talk due to lack of time.

Michael Elad
The Computer-Science Department
The Technion
Difficulties with *Sparseland*

- **Problem 3:** Why is this model suitable to describe various sources? e.g., Is it good for images? Audio? Stocks? ...

- **General answer:** Yes, this model is extremely effective in representing various sources
  - **Theoretical answer:** Clear connection to other models
  - **Empirical answer:** In a large variety of signal and image processing (and later machine learning), this model has been shown to lead to state-of-the-art results

Michael Elad
The Computer-Science Department
The Technion
Difficulties with *Sparseland*?

- **Problem 1**: Given an image patch, how can we find its atom decomposition?
- **Problem 2**: Given a family of signals, how do we find the dictionary to represent it well?
- **Problem 3**: Is this model flexible enough to describe various sources? E.g., Is it good for images? audio? ...

**ALL ANSWERED POSITIVELY AND CONSTRUCTIVELY**

Michael Elad  
The Computer-Science Department  
The Technion
Sparseland has a great success in signal & image processing and machine learning tasks.

In the past 8-9 years, many books were published on this and closely related fields.

This Field has been rapidly GROWING...
Coming Up: A Massive Open Online Course

Sparse Representations in Signal and Image Processing

Learn the theory, tools and algorithms of sparse representations and their impact on signal and image processing.

Start the Professional Certificate Program

Courses in the Professional Certificate Program

Sparse Representations in Signal and Image Processing: Fundamentals
Learn about the field of sparse representations by understanding its fundamental theoretical and algorithmic foundations.
Learn more

Sparse Representations in Image Processing: From Theory to Practice
Learn about the deployment of the sparse representation model to signal and image processing.
Learn more

Instructors

Yaniv Romano
Michael Elad
*Sparseland* for Image Processing

- When handling images, *Sparseland* is typically deployed on *small overlapping patches* due to the desire to *train the model* to fit the data better.

- The model assumption is: each patch in the image is believed to have a sparse representation w.r.t. a common local dictionary.

- What is the corresponding global model? This brings us to ... the Convolutional Sparse Coding (CSC).
Multi-Layered Convolutional Sparse Modeling

Joint work with

Yaniv Romano  Vardan Panyan  Jeremias Sulam
Convolutional Sparse Coding (CSC)

\[ [X] = \sum_{i=1}^{m} d_i \ast [\Gamma_i] \]

- \( m \) filters convolved with their sparse representations
- i-th feature-map: An image of the same size as \( X \) holding the sparse representation related to the i-filter
- An image with \( N \) pixels
- The i-th filter of small size \( n \)
CSC in Matrix Form

- Here is an alternative global sparsity-based model formulation

\[ x = \sum_{i=1}^{m} c^i \Gamma^i = [c^1 \ldots c^m] \begin{bmatrix} \Gamma^1 \\ \vdots \\ \Gamma^m \end{bmatrix} = D \Gamma \]

- $c^i \in \mathbb{R}^{N \times N}$ is a banded and Circulant matrix containing a single atom with all of its shifts

- $\Gamma^i \in \mathbb{R}^N$ are the corresponding coefficients ordered as column vectors

Michael Elad
The Computer-Science Department
The Technion
The CSC Dictionary

\[
\begin{bmatrix}
C^1 & C^2 & C^3
\end{bmatrix} =
\]

\[
D_L
\]

\[
D =
\]
Why CSC?

**Equation:**

\[ \mathbf{R}_i \mathbf{X} = \mathbf{\Omega} \mathbf{\gamma}_i \]

- **X = D\Gamma**
- **R\_i X = \Omega \gamma_i**

**Explanation:** Every patch has a sparse representation w.r.t. to the same local dictionary (\( \mathbf{\Omega} \)) just as assumed for images.

**Diagram Notes:**
- Stripe-dictionary
- Stripe vector
- \( (2n - 1)m \)
- \( n \)
Why CSC?

$\mathbf{R}_{i+1} \mathbf{X} = \mathbf{\Omega} \mathbf{\gamma}_i$

$\mathbf{R}_i \mathbf{X} = \mathbf{\Omega} \mathbf{\gamma}_i$

$\mathbf{R}_{i+1} \mathbf{X} = \mathbf{\Omega} \mathbf{\gamma}_{i+1}$

$\mathbf{X} = \mathbf{D} \mathbf{\Gamma}$

Every patch has a sparse representation w.r.t. to the same local dictionary ($\mathbf{\Omega}$) just as assumed for images.

stripe-vector

stripe-dictionary

$(2n - 1)m$

$n$
Classical Sparse Theory for CSC?

\[
\min_{\Gamma} \|\Gamma\|_0 \quad \text{s.t.} \quad \|Y - D\Gamma\|_2 \leq \varepsilon
\]

Theorem: **BP is guaranteed to “succeed” .... if**  \( \|\Gamma\|_0 < \frac{1}{4} \left( 1 + \frac{1}{\mu} \right) \)

- Assuming that \( m = 2 \) and \( n = 64 \) we have that [Welch, ’74]
  \[ \mu \geq 0.063 \]

- Success of pursuits is guaranteed as long as
  \[ \|\Gamma\|_0 < \frac{1}{4} \left( 1 + \frac{1}{\mu(D)} \right) \leq \frac{1}{2} \left( 1 + \frac{1}{0.063} \right) \approx 4.2 \]

- Only few (4) non-zeros GLOBALLY are allowed!!! This is a very pessimistic result!

- The classic *Sparseland* Theory does not cover well the CSC model
The main question we aim to address is this:

Can we generalize the vast theory of *Sparseland* to this new notion of local sparsity? For example, could we provide guarantees for success for pursuit algorithms?
Success of OMP

Theorem: If \( Y = DG + E \) where
\[
\| \Gamma \|_{0,\infty}^S < \frac{1}{2} \left( 1 + \frac{1}{\mu} \right) - \frac{1}{\mu} \cdot \frac{\| E \|_2^P}{\| \Gamma_{\text{min}} \|}
\]
then OMP run for \( \| \Gamma \|_0 \) iterations

1. Finds the correct support

2. \( \| \Gamma_{\text{OMP}} - \Gamma \|_2^2 \leq \frac{\| E \|_2^2}{1 - (\| \Gamma \|_{0,\infty}^S - 1)\mu} \)

This is a much better result – it allows few non-zeros locally in each stripe, implying a permitted \( O(N) \) non-zeros globally

Papyan, Sulam & Elad ('17)
Success of the Basis Pursuit

\[ \Gamma_{BP} = \min_{\Gamma} \frac{1}{2} \| Y - D\Gamma \|_2^2 + \lambda \| \Gamma \|_1 \]

Recent works tackling the convolutional sparse coding problem via BP

[Bristow, Eriksson & Lucey ‘13]
[Wohlberg ‘14]
[Kong & Fowlkes ‘14]
[Bristow & Lucey ‘14]
[Heide, Heidrich & Wetzstein ‘15]
[Šorel & Šroubek ‘16]
Success of the Basis Pursuit

\[ \Gamma_{BP} = \min_{\Gamma} \frac{1}{2} \| Y - D\Gamma \|_2^2 + \lambda \| \Gamma \|_1 \]

Theorem: For \( Y = D\Gamma + E \), if \( \lambda = 4 \| E \|_{2,\infty}^p \), if

\[ \| \Gamma \|_{0,\infty}^s < \frac{1}{3} \left( 1 + \frac{1}{\mu(D)} \right) \]

then Basis Pursuit performs very well:

1. The support of \( \Gamma_{BP} \) is contained in that of \( \Gamma \)
2. \( \| \Gamma_{BP} - \Gamma \|_{\infty} \leq 7.5 \| E \|_{2,\infty}^p \)
3. Every entry greater than \( 7.5 \| E \|_{2,\infty}^p \) is found
4. \( \Gamma_{BP} \) is unique

Papyan, Sulam & Elad (‘17)
Global Pursuit via Local Processing

- Could we suggest a solution of the global Basis Pursuit using only local (e.g. patch-based) operations?
- The answer is positive!!
- We define image slices:
  \[ s_i \equiv D_L \alpha_i \]

\[
\Gamma_{BP} = \min_{\Gamma} \frac{1}{2} \| Y - D\Gamma \|_2^2 + \lambda \| \Gamma \|_1
\]

\[ X = D\Gamma \]
Global Pursuit via Local Processing

\((P_1^c)\): \( \Gamma_{BP} = \min_{\Gamma} \frac{1}{2} \| Y - D\Gamma \|_2^2 + \lambda \| \Gamma \|_1 \)

Redefine this problem using \( s_i \) and \( \alpha_i \)

\[ \min_{\alpha_i,s_i} \frac{1}{2} \left\| Y - \sum_i R_i^T s_i \right\|_2^2 + \lambda \sum_i \| \alpha_i \|_1 \quad \text{s.t.} \quad \{s_i = D_L \alpha_i\}_i \]

These two are convex & equivalent

Update the \( \alpha_i \) by a local BP

Update the slices \( s_i \) by a simple LS & patch-averaging

If you apply the above two steps only once, you get a known patch-based denoising algorithm
Global Pursuit via Local Processing

\[(P_1^c):\quad \Gamma_{BP} = \min_{\Gamma} \frac{1}{2} \|Y - D\Gamma\|_2^2 + \lambda \|\Gamma\|_1\]

Redefine this problem using \(s_i\) and \(\alpha_i\)

\[
\min_{\alpha_i, s_i} \frac{1}{2} \left\| Y - \sum_i R_i^T s_i \right\|_2^2 + \lambda \sum_i \|\alpha_i\|_1 \quad \text{s.t.} \quad \{s_i = D_L \alpha_i\}_i
\]

These two are convex & equivalent

Update the \(\alpha_i\) by a local BP

Update the slices \(s_i\) by a simple LS & patch-averaging

This algorithm operates locally while guaranteeing to solve the global problem

If you apply these two steps only once, you get a known patch-based denoising algorithm.
Two Comments About this Scheme

We work with Slices and not Patches

Patches extracted from natural images, and their corresponding slices. Observe how the slices are far simpler, and contained by their corresponding patches.

The Proposed Scheme can be used for Dictionary ($D_L$) Learning

Slice-based DL algorithm using standard patch-based tools, leading to a faster and simpler method, compared to existing methods.

[Wohlberg, 2016]  Ours
Multi-Layered Convolutional Sparse Modeling
CSC and CNN

- There is a rough analogy between CSC and CNN:
  - Convolutional structure
  - Data driven models
  - ReLU is a sparsifying operator

- We shall now propose a principled way to analyze CNN

- But first, a brief review of CNN...
ReLU(z) = \text{max}(\text{Thr}, z)

[LeCun, Bottou, Bengio and Haffner ‘98]
[Krizhevsky, Sutskever & Hinton ‘12]
[Simonyan & Zisserman ‘14]
[He, Zhang, Ren & Sun ‘15]
CNN

\[ \text{ReLU}(z) = \max(\text{Thr}, z) \]

[LeCun, Bottou, Bengio and Haffner ‘98]
[Krizhevsky, Sutskever & Hinton ‘12]
[Simonyan & Zisserman ‘14]
[He, Zhang, Ren & Sun ‘15]
Mathematically...

\[ f(Y) = \text{ReLU}(b_2 + W_2^T \text{ReLU}(b_1 + W_1^T Y)) \]

\( Z_2 \in \mathbb{R}^{N m_2} \quad b_2 \in \mathbb{R}^{N m_2} \quad W_2^T \in \mathbb{R}^{N m_2 \times N m_1} \)

\( b_1 \in \mathbb{R}^{N m_1} \quad W_1^T \in \mathbb{R}^{N m_1 \times N} \)

\( Y \in \mathbb{R}^N \)
From CSC to Multi-Layered CSC

Convolutional sparsity (CSC) assumes an inherent structure is present in natural signals.

We propose to impose the same structure on the representations themselves.

\[ \mathbf{X} \in \mathbb{R}^N \quad \mathbf{D}_1 \in \mathbb{R}^{N \times Nm_1} \quad \mathbf{\Gamma}_1 \in \mathbb{R}^{Nm_1} \]

\[ m_1 \]

\[ n_0 \]

\[ \mathbf{\Gamma}_1 \in \mathbb{R}^{Nm_1} \quad \mathbf{D}_2 \in \mathbb{R}^{Nm_1 \times Nm_2} \quad \mathbf{\Gamma}_2 \in \mathbb{R}^{Nm_2} \]

\[ m_2 \]

\[ n_1 m_1 \]

\[ m_1 \}

Multi-Layer CSC (ML-CSC)
Intuition: From Atoms to Molecules

\[ X \in \mathbb{R}^N \quad D_1 \in \mathbb{R}^{N \times Nm_1} \quad \Gamma_1 \in \mathbb{R}^{Nm_1 \times Nm_2} \quad \Gamma_2 \in \mathbb{R}^{Nm_2} \]

- We can chain all the dictionaries into one effective dictionary
  \[ D_{\text{eff}} = D_1 D_2 D_3 \cdots D_K \rightarrow x = D_{\text{eff}} \Gamma_K \]
- This is a special *Sparseland* (indeed, a CSC) model

- However:
  - A key property in this model: sparsity of the intermediate representations
  - The effective atoms: atoms \( \rightarrow \) molecules \( \rightarrow \) cells \( \rightarrow \) tissue \( \rightarrow \) body-parts ...
A Small Taste: Model Training (MNIST)

MNIST Dictionary:
- $D_1$: 32 filters of size $7 \times 7$, stride of 2 (dense)
- $D_2$: 128 filters of size $5 \times 5 \times 32$, stride of 1 - 99.09% sparse
- $D_3$: 1024 filters of size $7 \times 7 \times 128$ – 99.89% sparse
A Small Taste: Model Training (CiFAR)

\[ D_1 \] (5×5×3)  \hspace{1cm} \[ D_1D_2 \] (13×13)  \hspace{1cm} \[ D_1D_2D_3 \] (32×32)

- \[ D_1 \]: 64 filters of size 5×5×3, stride of 2
- \[ D_2 \]: 256 filters of size 5×5×64, stride of 2

82.99% sparse
- \[ D_3 \]: 1024 filters of size 5×5×256

90.66% sparse
ML-CSC: Pursuit

- **Deep–Coding Problem** (DCP) (dictionaries are known):

  \[
  \begin{align*}
  X &= D_1 \Gamma_1 & \|\Gamma_1\|_{0,\infty}^{S} &\leq \lambda_1 \\
  \Gamma_1 &= D_2 \Gamma_2 & \|\Gamma_2\|_{0,\infty}^{S} &\leq \lambda_2 \\
  \vdots & & \vdots & \\
  \Gamma_{K-1} &= D_K \Gamma_K & \|\Gamma_K\|_{0,\infty}^{S} &\leq \lambda_K
  \end{align*}
  \]

- Or, more realistically for noisy signals,

  Find \( \{\Gamma_j\}_{j=1}^{K} \) s.t.

  \[
  \begin{align*}
  \|Y - D_1 \Gamma_1\|_2 &\leq \varepsilon & \|\Gamma_1\|_{0,\infty}^{S} &\leq \lambda_1 \\
  \Gamma_1 &= D_2 \Gamma_2 & \|\Gamma_2\|_{0,\infty}^{S} &\leq \lambda_2 \\
  \vdots & & \vdots & \\
  \Gamma_{K-1} &= D_K \Gamma_K & \|\Gamma_K\|_{0,\infty}^{S} &\leq \lambda_K
  \end{align*}
  \]
A Small Taste: Pursuit

\[ x = D_1 \Gamma_1 \]
\[ x = D_1 D_2 \Gamma_2 \]
\[ x = D_1 D_2 D_3 \Gamma_3 \]

\[ \Gamma_1 \]
94.51% sparse 
(213 nnz)

\[ \Gamma_2 \]
99.52% sparse 
(30 nnz)

\[ \Gamma_3 \]
99.51% sparse 
(5 nnz)
The simplest pursuit algorithm (single-layer case) is the THR algorithm, which operates on a given input signal $\mathbf{Y}$ by:

$$
\mathbf{Y} = \mathbf{D} \mathbf{\Gamma} + \mathbf{E}
$$

and $\mathbf{\Gamma}$ is sparse

$$
\hat{\mathbf{\Gamma}} = \mathcal{P}_\beta (\mathbf{D}^T \mathbf{Y})
$$
Consider this for Solving the DCP

- Layered thresholding (LT):
  - Estimate $\hat{\Gamma}_1$ via the THR algorithm
  - Estimate $\hat{\Gamma}_2$ via the THR algorithm

\[
\hat{\Gamma}_2 = \mathcal{P}_{\beta_2} \left( D_2^T \mathcal{P}_{\beta_1} (D_1^T Y) \right)
\]

- Now let’s take a look at how Conv. Neural Network operates:
  \[
f(Y) = \text{ReLU}(b_2 + W_2^T \text{ReLU}(b_1 + W_1^T Y))
\]

The layered (soft nonnegative) thresholding and the CNN forward pass algorithm are the very same thing !!!
Theoretical Path

\[ X = D_1 \Gamma_1 \]
\[ \Gamma_1 = D_2 \Gamma_2 \]
\[ \vdots \]
\[ \Gamma_{K-1} = D_K \Gamma_K \]
\[ \Gamma_i \text{ is } L_{0,\infty} \text{ sparse} \]

Armed with this view of a generative source model, we may ask new and daring questions.

Layered THR (Forward Pass)

Maybe other?
Theoretical Path: Possible Questions

- Having established the importance of the ML-CSC model and its associated pursuit, the DCP problem, we now turn to its analysis.

- The main questions we aim to address:
  
  I. Stability of the solution obtained via the hard layered THR algorithm (forward pass)?

  II. Limitations of this (very simple) algorithm and alternative pursuit?

  III. Algorithms for training the dictionaries \( \{D_i\}_{i=1}^{K} \) vs. CNN?

  IV. New insights on how to operate on signals via CNN?

... and here are questions we will not touch today:
Success of the Layered-THR

Theorem: If \( ||\Gamma_i||^s_{0,\infty} < \frac{1}{2} \left( 1 + \frac{1}{\mu(D_i)} \cdot \frac{|\Gamma_i^{\text{min}}|}{|\Gamma_i^{\text{max}}|} \right) - \frac{1}{\mu(D_i)} \cdot \frac{\epsilon_L^{i-1}}{|\Gamma_i^{\text{max}}|} \)

then the **Layered Hard THR** (with the proper thresholds) **finds the correct supports** and \( ||\Gamma_i^{LT} - \Gamma_i||^p_{2,\infty} \leq \epsilon_L^i \), where we have defined \( \epsilon_L^0 = ||E||^p_{2,\infty} \) and

\[
\epsilon_L^i = \sqrt{||\Gamma_i||^p_{0,\infty} \cdot (\epsilon_L^{i-1} + \mu(D_i) (||\Gamma_i||^s_{0,\infty} - 1) |\Gamma_i^{\text{max}}|))}
\]

The stability of the forward pass is guaranteed if the underlying representations are **locally** sparse and the noise is **locally** bounded.

Problems:
1. Contrast
2. Error growth
3. Error even if no noise

Papyan, Romano & Elad ('17)
Layered Basis Pursuit (BP)

- We chose the Thresholding algorithm due to its simplicity, but we do know that there are better pursuit methods – how about using them?
- Let's use the Basis Pursuit instead ...

\[
\Gamma_1^{\text{LBP}} = \min_{\Gamma_1} \frac{1}{2} \| \mathbf{Y} - \mathbf{D}_1 \Gamma_1 \|^2_2 + \lambda_1 \| \Gamma_1 \|_1
\]

\[
\Gamma_2^{\text{LBP}} = \min_{\Gamma_2} \frac{1}{2} \| \Gamma_1^{\text{LBP}} - \mathbf{D}_2 \Gamma_2 \|^2_2 + \lambda_2 \| \Gamma_2 \|_1
\]

\[
\vdots
\]

\[
(\text{DCP}^\mathcal{E}_\lambda): \text{Find } \{\Gamma_j\}_{j=1}^K \text{ s.t.}
\]

\[
\begin{cases}
\| \mathbf{Y} - \mathbf{D}_1 \Gamma_1 \|_2 \leq \mathcal{E} & \| \Gamma_1 \|_{0,\infty} \leq \lambda_1 \\
\Gamma_1 = \mathbf{D}_2 \Gamma_2 & \| \Gamma_2 \|_{0,\infty} \leq \lambda_2 \\
\vdots & \vdots \\
\Gamma_{K-1} = \mathbf{D}_K \Gamma_K & \| \Gamma_K \|_{0,\infty} \leq \lambda_K
\end{cases}
\]

Deconvolutional networks

[Zeiler, Krishnan, Taylor & Fergus '10]
Success of the Layered BP

Theorem: Assuming that $\|\Gamma_i\|_{0,\infty}^s < \frac{1}{3} \left( 1 + \frac{1}{\mu(D_i)} \right)$

then the Basis Pursuit performs very well:

1. The support of $\Gamma_i^{LBP}$ is contained in that of $\Gamma_i$
2. The error is bounded: $\|\Gamma_i^{LBP} - \Gamma_i\|^p_{2,\infty} \leq \varepsilon_i^L$, where
   $$\varepsilon_i^L = 7.5^i \|E\|^p_{2,\infty} \prod_{j=1}^i \sqrt{\|\Gamma_j\|^p_{0,\infty}}$$
3. Every entry in $\Gamma_i$ greater than $\varepsilon_i^L / \sqrt{\|\Gamma_i\|^p_{0,\infty}}$ will be found

Problems:
1. Contrast
2. Error growth
3. Error even if no noise

Papyan, Romano & Elad ('17)
Layered Iterative Thresholding:

Layered BP:

\[
\Gamma_j^{LBP} = \min_{\Gamma_j} \frac{1}{2} \|\Gamma_{j-1}^{LBP} - D_j \Gamma_j\|^2 + \xi_j \|\Gamma_j\|_1
\]

Layered Iterative Soft-Thresholding:

\[
\Gamma_j^t = S_{\xi_j/c_j} \left( \Gamma_{j-1}^{t-1} + D_j^T (\hat{\Gamma}_{j-1} - D_j \Gamma_{j-1}^{t-1}) \right)
\]

Note that our suggestion implies that groups of layers share the same dictionaries.

Can be seen as a very deep recurrent neural network

[Gregor & LeCun ‘10]
Time to Conclude
This Talk

A novel interpretation and theoretical understanding of CNN

Novel View of Convolutional Sparse Coding

Multi-Layer Convolutional Sparse Coding

Sparseland

The desire to model data

\[ X = D_1 \Gamma_1 \]
\[ \Gamma_1 = D_2 \Gamma_2 \]
\[ \vdots \]
\[ \Gamma_{K-1} = D_K \Gamma_K \]
\[ \Gamma_k \text{ is } L_{0,\infty} \text{ sparse} \]
This Talk

**Take Home Message 1:**
Generative modeling of data sources enables algorithm development along with theoretically analyzing algorithms’ performance

A novel interpretation and theoretical understanding of CNN

Sparseland

Multi-Layer Convolutional Sparse Coding

Novel View of Convolutional Sparse Coding

The desire to model data
This Talk

A novel interpretation and theoretical understanding of CNN

Sparseland

The desire to model data

Novel View of Convolutional Sparse Coding

Multi-Layer Convolutional Sparse Coding

Take Home Message 2: The Multi-Layer Convolutional Sparse Coding model could be a new platform for understanding and developing deep-learning solutions
More on these (including these slides and the relevant papers) can be found in http://www.cs.technion.ac.il/~elad