A Tale of Signal Modeling Evolution

SparseLand → CSC → CNN

Michael Elad

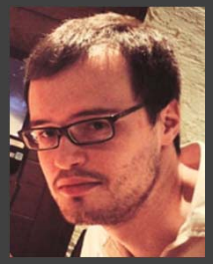
The Computer Science Department

Technion – Israel Institute of Technology

Workshop on Frame Theory and
Sparse Representation for Complex Data
Institute for Mathematical Sciences
May 29th – June 2nd

Joint work with

Yaniv Romano



Vardan Papyan

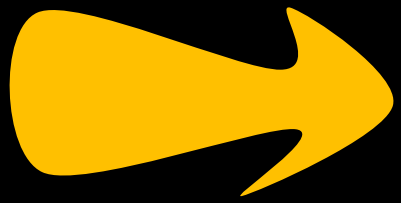
Jeremias Sulam

Technion Israel Institute of Technology

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

In This Talk

SparseLand
Sparse Representation Theory



CNN*
Convolutional Neural Networks

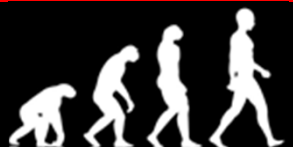
The Underlying Idea

Modeling

data sources enables a theoretical analysis of algorithms' performance

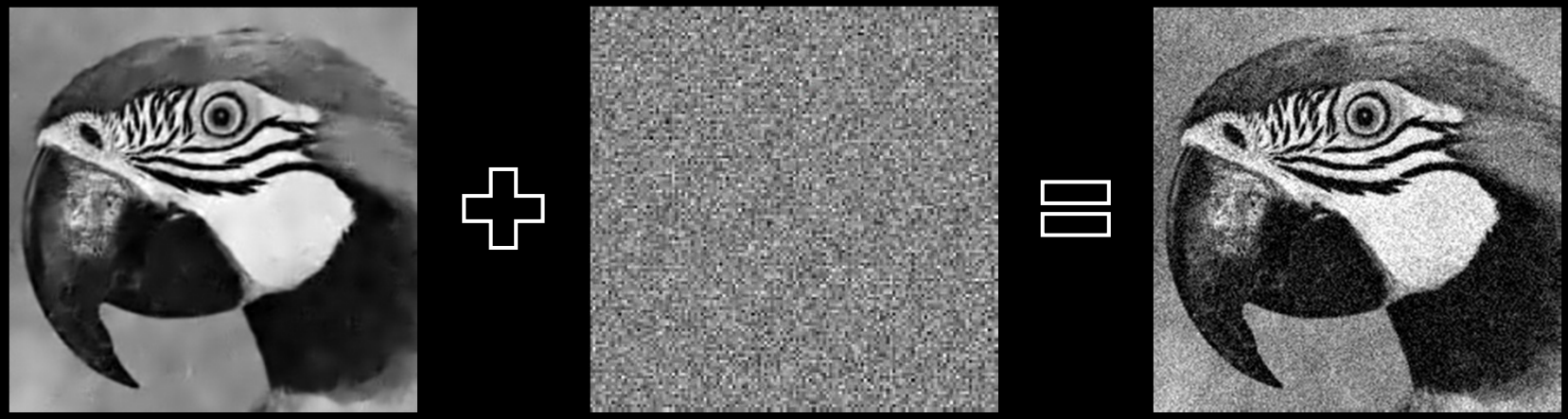
* Only CNN? What about other architectures ?

Part I
Motivation and Background



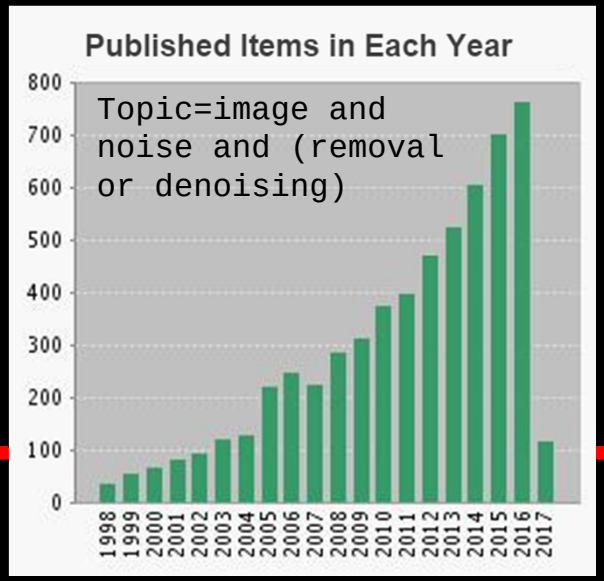
Our Starting Point: Image Denoising

Original Image $\mathbf{X}$ + White Gaussian Noise $\mathbf{E}$ = Noisy Image $\mathbf{Y}$



Many (thousands) image denoising algorithms have been proposed over the years, some of which are extremely effective

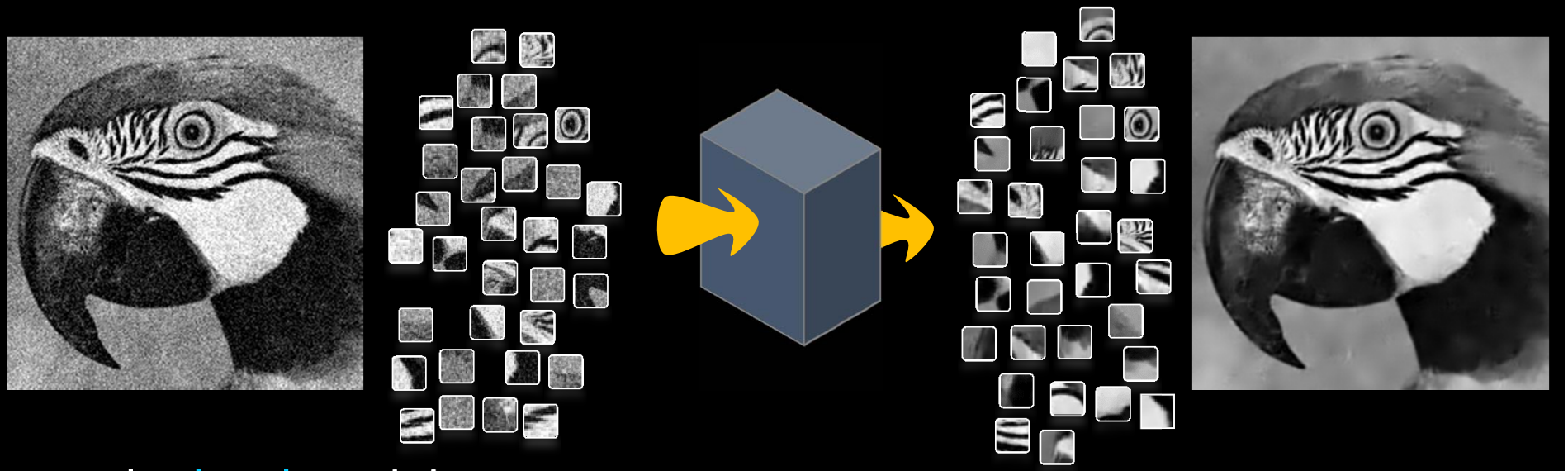
$\mathbf{Y}$ → Denoising Algorithm → $\hat{\mathbf{X}}$



Technion
Israel Institute of Technology

Leading Image Denoising Methods…

are built upon powerful patch-based local models:



Popular local models: GMM
Sparse-Representation
Example-based
Low-rank
Field-of-Experts &
Neural networks

Patch-Based Image Denoising

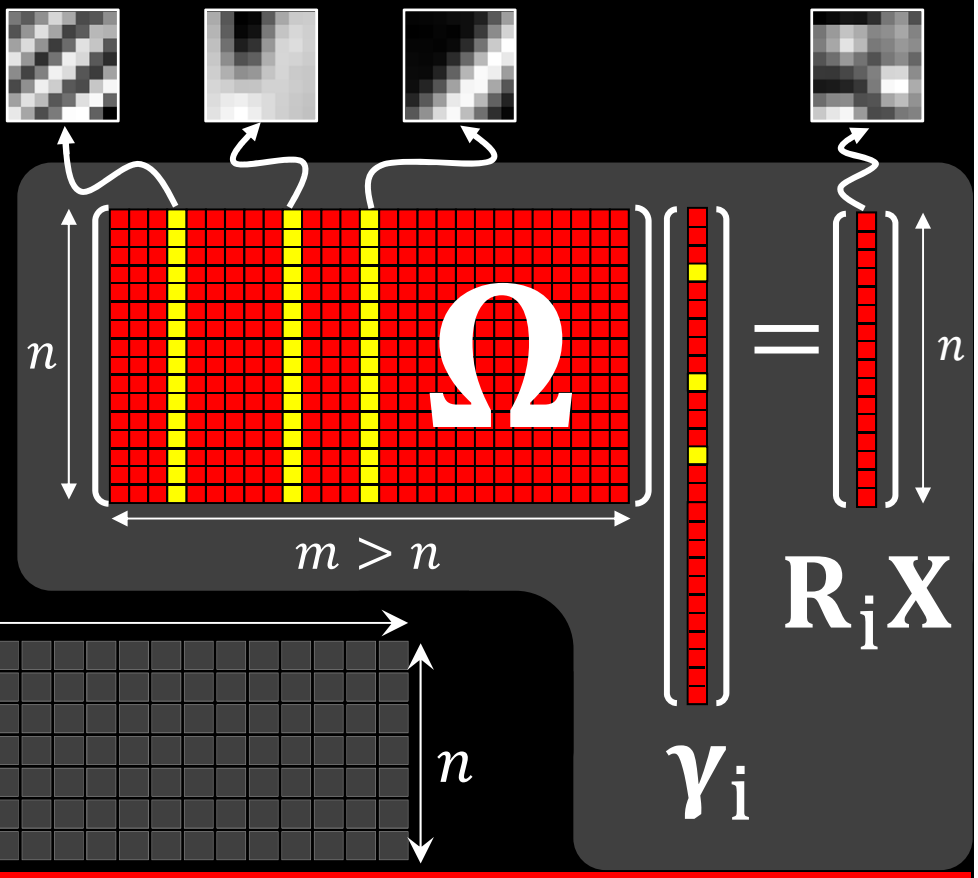
- K-SVD: sparse representation modeling of image patches [Elad & Aharon, ‘06]
- BM3D: combines sparsity and self-similarity [Dabov, Foi, Katkovnik & Egiazarian ‘07]
- EPLL: uses GMM of the image patches [Zoran & Weiss ‘11]
- MLP: multi-layer perceptron [Burger, Schuler & Harmeling ‘12]
- NCSR: non-local sparsity with centralized coefficients [Dong, Zhang, Shi & Li ‘13]
- WNNM: weighted nuclear norm of image patches [Gu, Zhang, Zuo & Feng ‘14]
- SSC–GSM: nonlocal sparsity with a GSM coefficient model [Dong, Shi, Ma & Li ‘15]

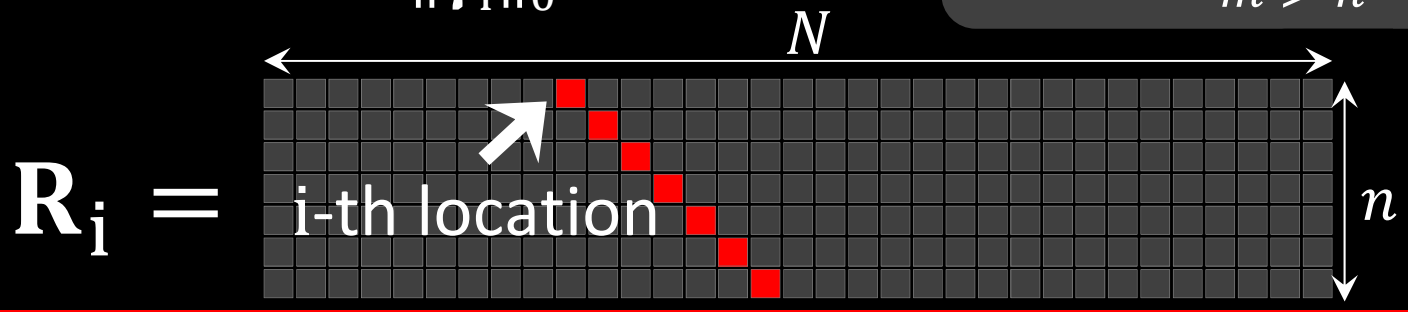
The SparseLand Model for Patches

- Assumes that every patch is a linear combination of a few atoms, from a dictionary
- The operator $\mathbf{R}_i$ extracts the i-th n-dimensional patch from $\mathbf{X} \in \mathbb{R}^N$
- Model assumption:

$$\forall i\ ,\ \mathbf{R}_i\mathbf{X} = \mathbf{\Omega}\boldsymbol{\gamma}_i$$

where $\|\boldsymbol{\gamma}_i\|_0 \ll n$





* $\mathbf{R}_i$ for 1D signals

Patch Denoising

Given a noisy patch $\mathbf{R}_i\mathbf{Y}$, solve $(\mathbf{P}_0^\epsilon)$: $\hat{\boldsymbol{\gamma}}_i = \underset{\boldsymbol{\gamma}_i}{\operatorname{argmin}} \|\boldsymbol{\gamma}_i\|_0$ s.t. $\|\mathbf{R}_i\mathbf{Y} - \boldsymbol{\Omega}\boldsymbol{\gamma}_i\|_2 \leq \epsilon$

Clean patch: $\boldsymbol{\Omega}\hat{\boldsymbol{\gamma}}_i$

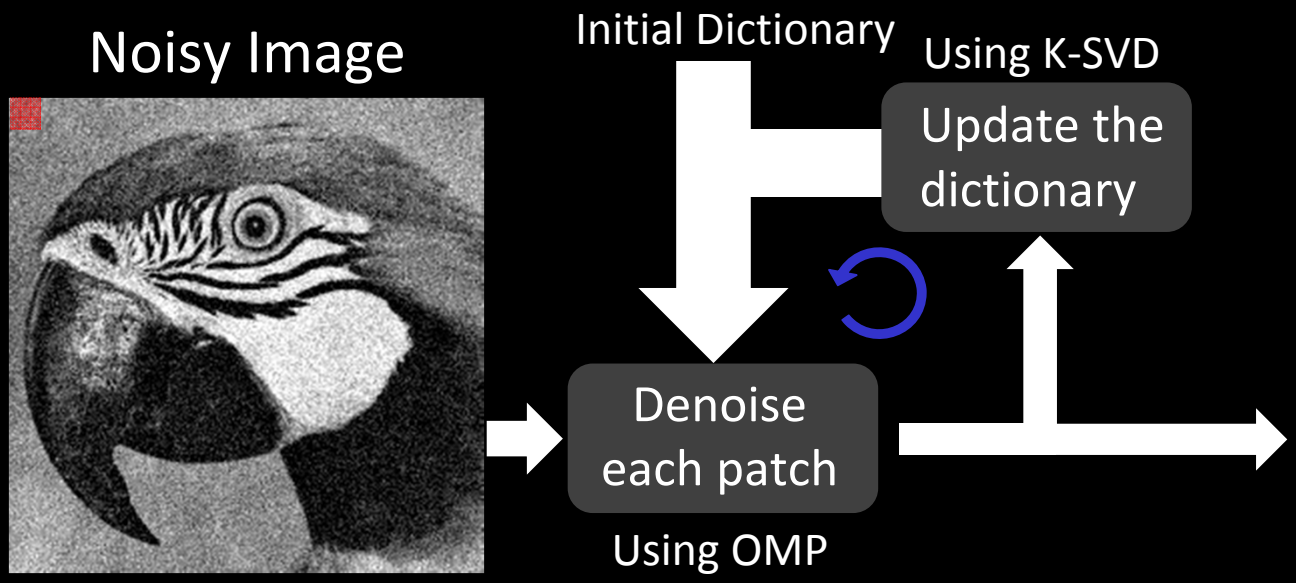
$(\mathbf{P}_0^\epsilon)$ is hard to solve

Greedy methods such as Orthogonal Matching Pursuit (OMP) or Thresholding

Convex relaxations such as Basis Pursuit (BP)

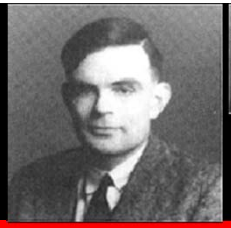
$(\mathbf{P}_1^\epsilon)$: $\min_{\boldsymbol{\gamma}_i} \|\boldsymbol{\gamma}_i\|_1 + \xi\|\mathbf{R}_i\mathbf{Y} - \boldsymbol{\Omega}\boldsymbol{\gamma}_i\|_2^2$

Recall K-SVD Denoising [Elad & Aharon, '06]



- Despite its simplicity, this is a very well-performing algorithm
- Its origins can be traced back to Guleryuz's local DCT recovery
- A small modification of this method leads to state-of-the-art results [Mairal, Bach, Ponce, Spairo, Zisserman, `09]

What is Missing?



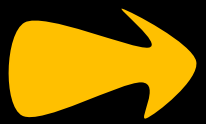
- Over the years, many kept revisiting this algorithm and its line of thinking, with a clear feeling that key features are still lacking
- What is missing? Here is what **WE** thought of…
  - A multi-scale treatment [Ophir, Lustig & Elad '11] [Sulam, Ophir & Elad '14] [Papyan & Elad '15]
  - Exploiting self-similarities [Ram & Elad '13] [Romano, Protter & Elad '14]
  - Pushing to better agreement on the overlaps [Romano & Elad '13] [Romano & Elad '15]
  - Enforcing the local model on the final patches (EPLL) [Sulam & Elad '15]
- Eventually, we realized that the key part that is missing is

A Theoretical Backbone

Missing Theoretical Backbone?

- The core global-local model assumption on $\mathbf{X} \in \mathbb{R}^N$:

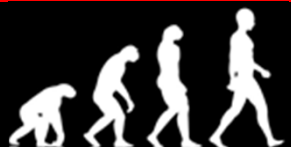
$$\forall i \quad \mathbf{R}_i \mathbf{X} = \boldsymbol{\Omega}\boldsymbol{\gamma}_i \quad \text{where} \quad \|\boldsymbol{\gamma}_i\|_0 \leq k$$



Every patch in the unknown signal is expected to have a sparse representation w.r.t. the same dictionary $\boldsymbol{\Omega}$

- Questions to consider:
  - Who are the signals belonging to this model? Do they exist?
  - How should we project a signal on this model (pursuit)?
  - Could we offer theoretical guarantees for this model/algorithms?
  - Could we offer a global pursuit algorithm that operates locally?
  - How should we learn $\boldsymbol{\Omega}$ if this is indeed the model?

- As we will see, all these questions are very relevant to recent developments in signal processing and machine learning



Coming Up

Limitations of patch averaging

Convolutional Sparse Coding (CSC) model

Theoretical study of CSC

Multi-Layer Convolutional Sparse Coding (ML-CSC)

Convolutional neural networks (CNN)

Fresh view of CNN through the eyes of sparsity

Part II
Convolutional Sparse Coding

Working Locally Thinking Globally:
Theoretical Guarantees for Convolutional Sparse Coding
Vardan Papyan, Jeremias Sulam and Michael Elad

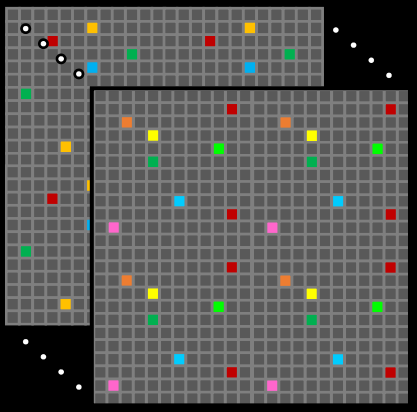
Convolutional Dictionary Learning via Local Processing
Vardan Papyan, Yaniv Romano, Jeremias Sulam, and Michael Elad

Convolutional Sparse Coding (CSC)

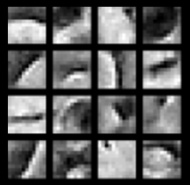
m filters convolved with their sparse representations

$$\mathbf{X} = \sum_{i=1}^{m} \mathbf{d}_i * \mathbf{z}_i$$

i-th feature-map: An image of the same size as $\mathbf{X}$ holding the sparse representation related to the i-filter

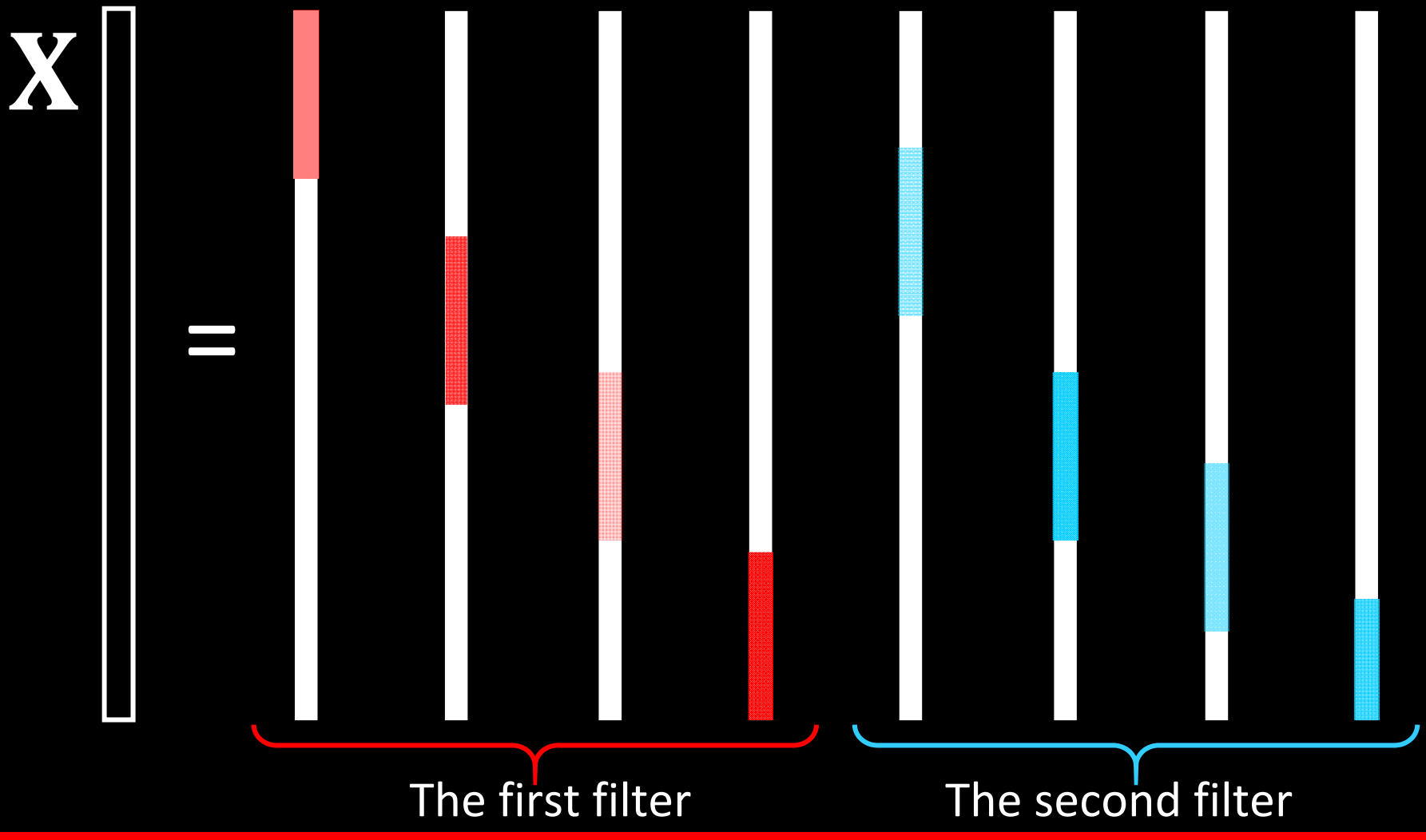


An image held as a column vector of length N



The j-th filter of small support n

Intuitively …

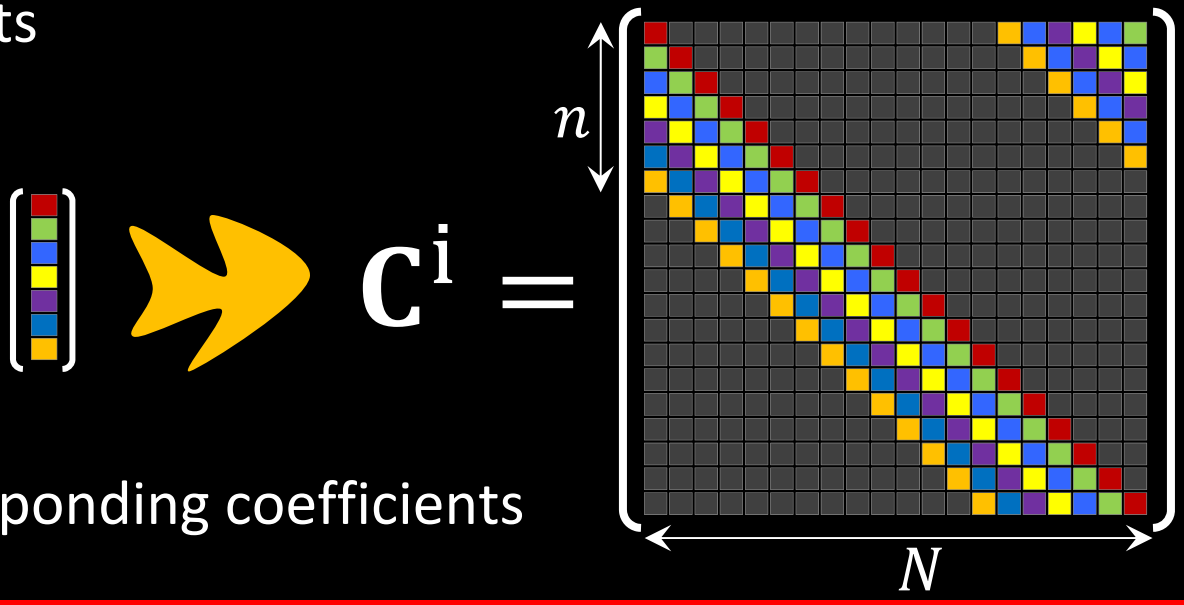


CSC in Matrix Form

- Here is an alternative global sparsity-based model formulation

$$\mathbf{X} = \sum_{i=1}^{m} \mathbf{C}^i \boldsymbol{\Gamma}^i = \mathbf{D}\boldsymbol{\Gamma}$$

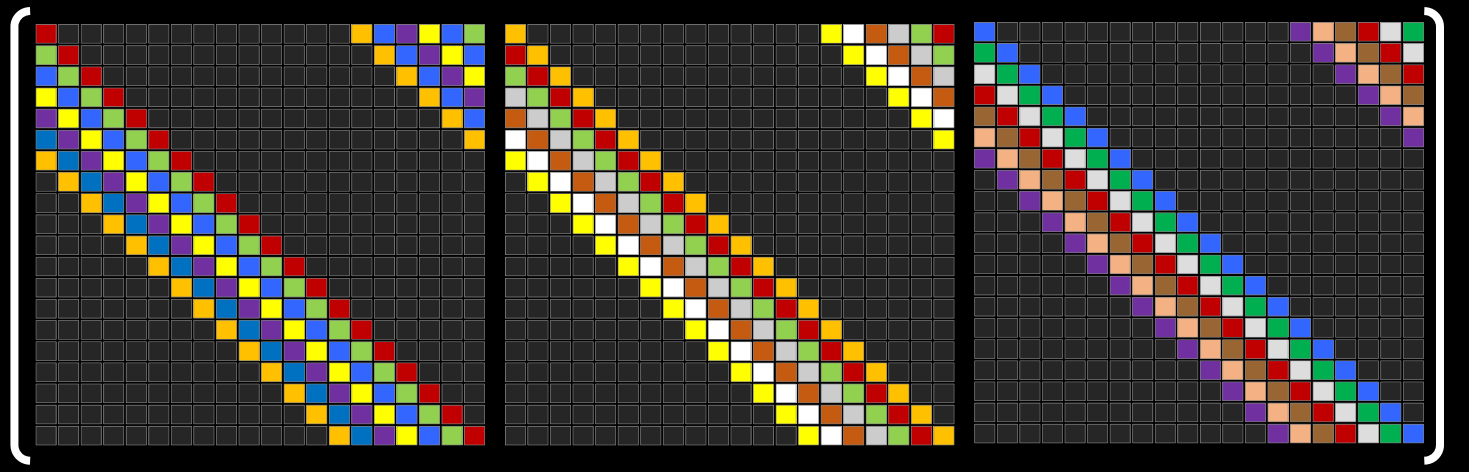
- $\mathbf{C}^i \in \mathbb{R}^{N \times N}$ is a banded and Circulant matrix containing a single atom with all of its shifts



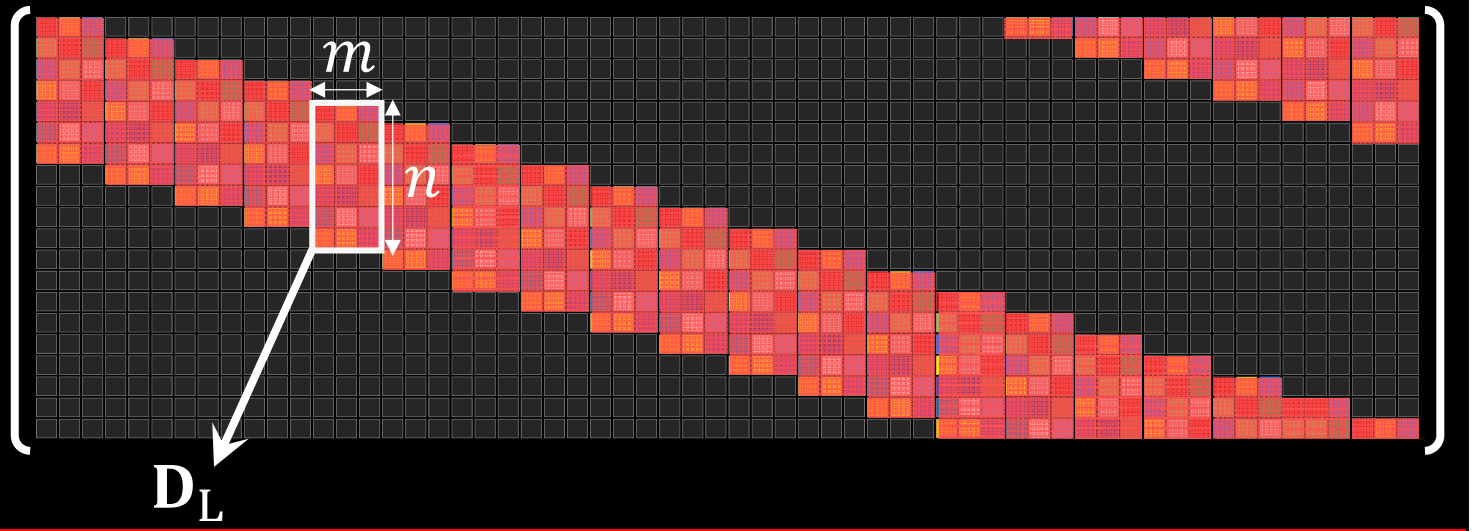
- $\boldsymbol{\Gamma}^i \in \mathbb{R}^N$ are the corresponding coefficients

Two Interpretations

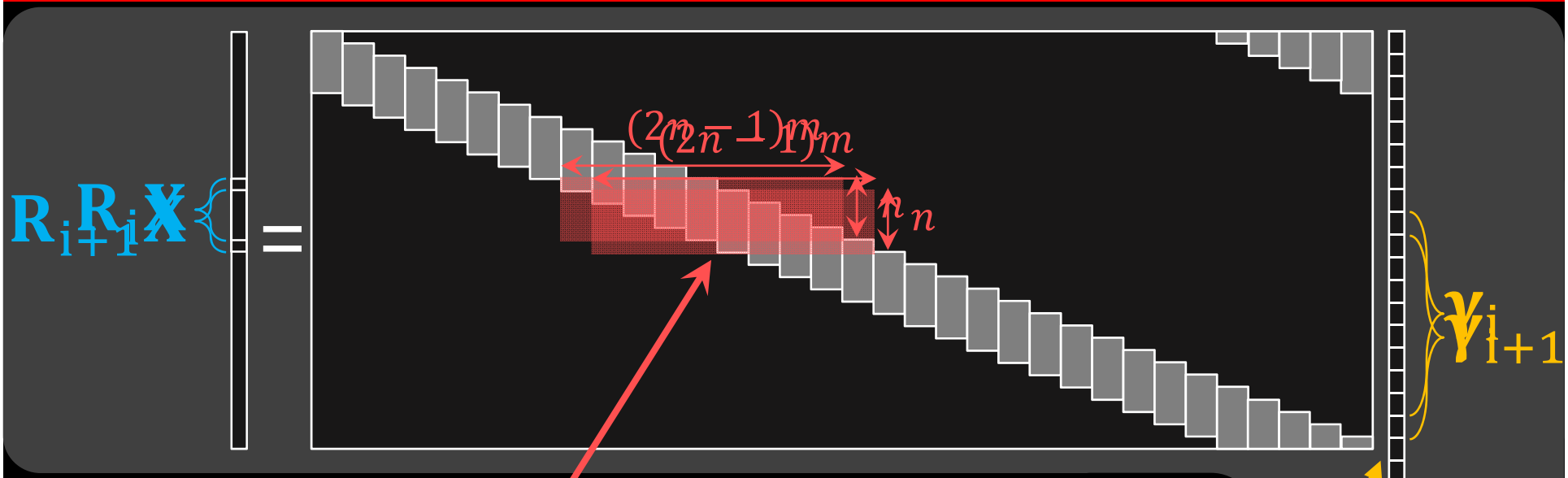
$[\mathbf{C}^1\ \mathbf{C}^2\ \mathbf{C}^3] =$



$\mathbf{D} =$



Why CSC?



stripe-dictionary

stripe vector

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

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

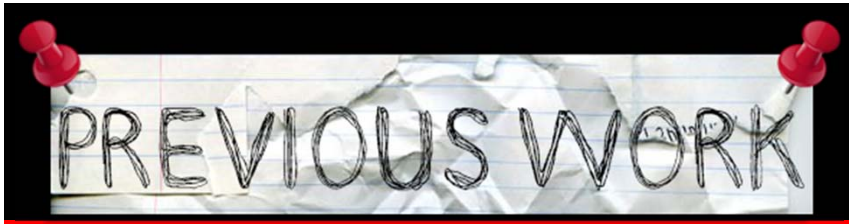
Every patch has a sparse representation w.r.t. to the same local dictionary $\boldsymbol{\Omega}$, just as we have assumed

CSC Relation to Our Story

- A clear global model: every patch has a sparse representation w.r.t. to the same local dictionary $\boldsymbol{\Omega}$, just as we have assumed
- No notion of disagreement on the patch overlaps
- Related to the current common practice of patch averaging ($\mathbf{R}_i^T$ - put the patch $\boldsymbol{\Omega}\boldsymbol{\gamma}_i$ back in the i-th location of the global vector)

$$\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma} = \frac{1}{n}\sum_i \mathbf{R}_i^T \boldsymbol{\Omega}\boldsymbol{\gamma}_i$$

- What about the Pursuit?
  - "Patch averaging": independent sparse coding for each patch
  - CSC: should seek all the representations together
- Is there a bridge between the two? We'll come back to this later …



- This model has been used in the past [Lewicki & Sejnowski '99] [Hashimoto & Kurata, '00]
- Most works have focused on solving *efficiently* its associated pursuit, called **convolutional sparse coding**, using the BP algorithm

$$(\mathbf{P}_1^{\epsilon}): \quad \min_{\boldsymbol{\Gamma}} \quad \|\boldsymbol{\Gamma}\|_1 + \lambda\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2$$

Convolutional dictionary

- Several applications were demonstrated:
  - Pattern detection in images and the analysis of instruments in music signals [Mørup, Schmidt & Hansen '08]
  - Inpainting [Heide, Heidrich & Wetzstein '15]
  - Super-resolution [Gu, Zuo, Xie, Meng, Feng & Zhang '15]
- However, little is known regrading its theoretical aspects. Why? Perhaps because the regular SparsLand theory is sufficient?

Classical Sparse Theory (Noiseless)

$$(\mathbf{P}_0): \quad \min_{\boldsymbol{\Gamma}} \quad \|\boldsymbol{\Gamma}\|_0 \quad \text{s.t.} \quad \mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$$

Definition: Mutual Coherence: $\mu(\mathbf{D}) = \max_{i \neq j} |d_i^T d_j|$ [Donoho & Elad '03]

Theorem: For a signal $\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$, if $\|\boldsymbol{\Gamma}\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$ then this solution is necessarily the sparsest

[Donoho & Elad '03]

Theorem: The OMP and BP are guaranteed to recover the true sparse code assuming that $\|\boldsymbol{\Gamma}\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$

[Tropp '04], [Donoho & Elad '03]

The Need for a Theoretical Study

- Assuming that $m = 2$ and $n = 64$ we have that [Welch, '74]

$$\mu(\mathbf{D}) \geq 0.063$$

- As a result, uniqueness and success of pursuits is guaranteed as long as

$$\|\boldsymbol{\Gamma}\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right) \leq \frac{1}{2}\left(1 + \frac{1}{0.063}\right) \approx 8$$

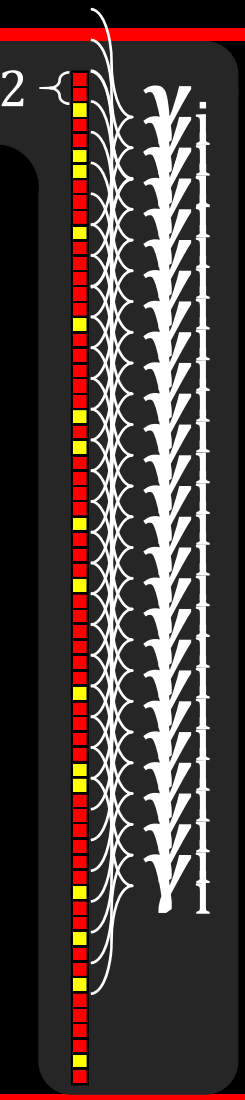
- Less than 8 non-zeros GLOBALLY are allowed!!! This is a very pessimistic result!
- Repeating the above for the noisy case leads to even worse performance predictions
- Bottom line: Classic SparseLand Theory cannot provide good explanations for the CSC model

Moving to Local Sparsity: Stripes

$\ell_{0,\infty}$ Norm: $\|\boldsymbol{\Gamma}\|_{0,\infty}^{s} = \max_{i} \|\boldsymbol{\gamma}_i\|_0$

$$\left(\mathbf{P}_{0,\infty}\right): \quad \min_{\boldsymbol{\Gamma}} \|\boldsymbol{\Gamma}\|_{0,\infty}^{s} \ \text{s.t.} \ \mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$$

$\|\boldsymbol{\Gamma}\|_{0,\infty}^{s}$ is low $\rightarrow$ all $\boldsymbol{\gamma}_i$ are sparse $\rightarrow$ every patch has a sparse representation over $\boldsymbol{\Omega}$



The Main Questions we Aim to Address:

I. Is the solution to this problem unique ?

II. Can we recover the solution via a global OMP/BP ?

Stripe-Spark and Uniqueness

$$(\mathbf{P}_{0,\infty}): \quad \min_{\boldsymbol{\Gamma}} \ \|\boldsymbol{\Gamma}\|_{0,\infty}^{s} \ \text{s.t.} \ \mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$$

Definition: Stripe Spark $\eta_\infty(\mathbf{D}) = \min_{\boldsymbol{\Delta}} \ \|\boldsymbol{\Delta}\|_{0,\infty}^{s} \ \text{s.t.} \ \begin{Bmatrix} \mathbf{D}\boldsymbol{\Delta} = 0 \\ \boldsymbol{\Delta} \neq 0 \end{Bmatrix}$

Theorem: If a solution $\boldsymbol{\Gamma}$ is found for $(\mathbf{P}_{0,\infty})$ such that:

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{s} < \frac{1}{2}\eta_\infty$$

then it is necessarily the optimal solution to this problem

Theorem: The relation between the Stripe-Spark and the Mutual Coherence is:

$$\eta_\infty(\mathbf{D}) \geq 1 + \frac{1}{\mu(\mathbf{D})}$$

Uniqueness via Mutual Coherence

$$(\mathbf{P}_{0,\infty}): \quad \min_{\boldsymbol{\Gamma}} \ \|\boldsymbol{\Gamma}\|_{0,\infty}^{s} \ \text{s.t.} \ \mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$$

Theorem: If a solution $\boldsymbol{\Gamma}$ is found for $(\mathbf{P}_{0,\infty})$ such that:

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{s} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$$

then this is necessarily the unique optimal solution to this problem

This result is exciting: This and later results pose a **local** constraint for a **global** guarantee, and as such, they are far more optimistic compared to the **global** guarantees

For k non-zeros per stripe, and filters of length n, we get

$$\|\boldsymbol{\Gamma}\|_0 \cong \frac{k}{2n-1} \cdot N$$

non-zeros globally

Recovery Guarantees

$$\left(\mathbf{P}_{0,\infty}\right): \quad \min_{\boldsymbol{\Gamma}} \ \|\boldsymbol{\Gamma}\|_{0,\infty}^{s} \ \text{s.t.} \ \mathbf{X} = \mathbf{D}\boldsymbol{\Gamma}$$

Lets solve this problem via OMP or BP*, applied globally

Theorem: If a solution $\boldsymbol{\Gamma}$ of $\left(\mathbf{P}_{0,\infty}\right)$ satisfies:

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{s} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right)$$

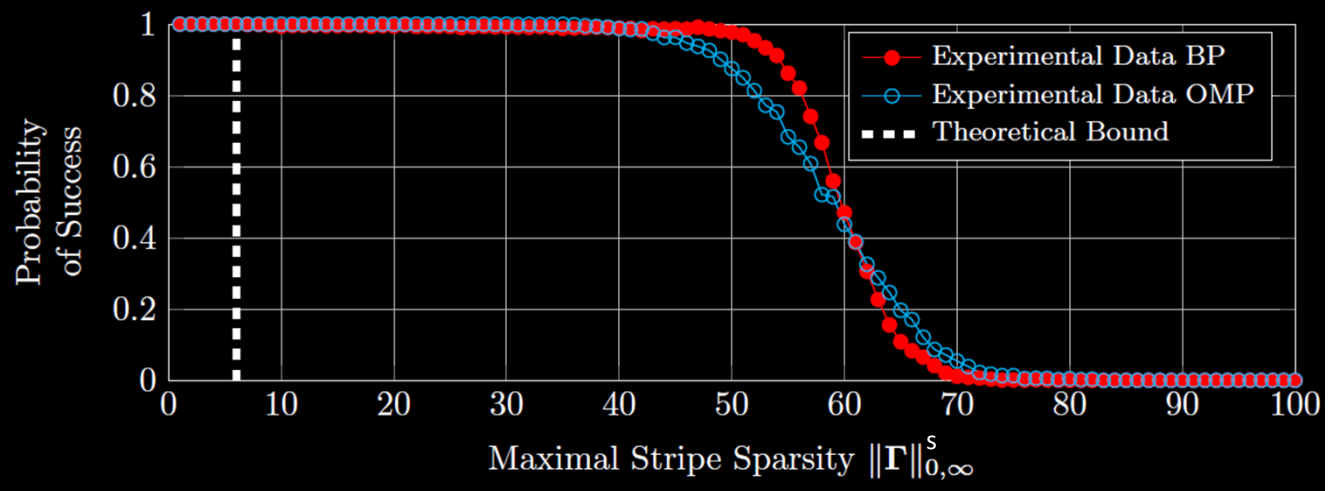
then **global** OMP and BP are guaranteed to find it

Both OMP and BP do not assume **local** sparsity but still guaranteed to succeed. One could propose algorithms that rely on this assumption

* How about variants that would exploit the local sparsity?

Phase Transition Experiment

- We construct a dictionary with a low mutual coherence:
$$m = 2, n = 64, N = 640$$
- We generate random sparse vectors in which the non-zero entries are drawn as random i.i.d Gaussians
- Given a sparse vector, we compute its global signal and attempt to recover it using the global OMP and BP
- The theoretical bound allows $\approx 0.05 \cdot N$ global non-zeros



From Ideal to Noisy Signals

- So far, we have assumed an ideal signal $\mathbf{X} = \mathbf{D\Gamma}$
- However, in practice we usually have $\mathbf{Y} = \mathbf{D\Gamma} + \mathbf{E}$ where $\mathbf{E}$ is due to noise or model deviations
- To handle this, we redefine our problem as:

$$\left(\mathbf{P}_{0,\infty}^{\epsilon}\right): \quad \min_{\mathbf{\Gamma}} \quad \|\mathbf{\Gamma}\|_{0,\infty}^{s} \quad \text{s.t.} \quad \|\mathbf{Y} - \mathbf{D\Gamma}\|_2 \leq \epsilon$$

- **The Main Questions We Aim to Address:**
  I. Stability of the solution to this problem ?
  II. Stability of the solution obtained via global OMP/BP ?
  III. Could the same recovery be done via local (patch) operations ?

[Candes & Tao '05]

Stability of via Stripe-RIP

$$\left(\mathbf{P}_{0,\infty}^{\epsilon}\right): \quad \min_{\boldsymbol{\Gamma}} \ \|\boldsymbol{\Gamma}\|_{0,\infty}^{s} \ \text{s.t.} \ \|\mathbf{Y}-\mathbf{D}\boldsymbol{\Gamma}\|_2 \le \epsilon \ \Rightarrow \ \widehat{\boldsymbol{\Gamma}}$$

Definition: $\mathbf{D}$ is said to satisfy Stripe-RIP with constant δ_k if

$$(1-\delta_k)\|\boldsymbol{\Delta}\|_2^2 \le \|\mathbf{D}\boldsymbol{\Delta}\|_2^2 \le (1+\delta_k)\|\boldsymbol{\Delta}\|_2^2$$

for any vector $\boldsymbol{\Delta}$ with $\|\boldsymbol{\Delta}\|_{0,\infty}^{s} = k$

Theorem: If the true representation $\boldsymbol{\Gamma}$ satisfies

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{s} = k < \frac{1}{2}\left(1+\frac{1}{\mu(\mathbf{D})}\right)$$

then a solution $\widehat{\boldsymbol{\Gamma}}$ for $\left(\mathbf{P}_{0,\infty}^{\epsilon}\right)$ must be close to it

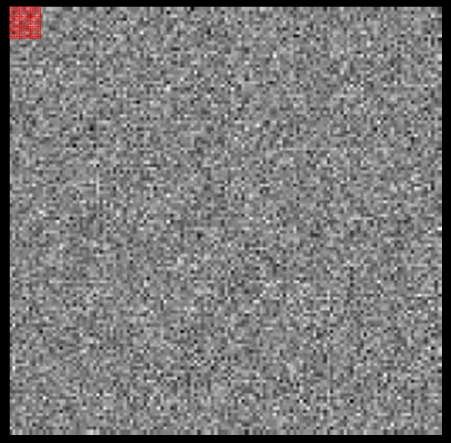
$$\left\|\widehat{\boldsymbol{\Gamma}}-\boldsymbol{\Gamma}\right\|_2^2 \le \frac{4\epsilon^2}{1-\delta_{2k}} \le \frac{4\epsilon^2}{1-(2k-1)\mu(\mathbf{D})}$$

$$\delta_k \le (k-1)\mu(\mathbf{D})$$

Local Noise Assumption

- Thus far, our analysis relied on the local sparsity of the underlying solution $\boldsymbol{\Gamma}$, which was enforced through the $\ell_{0,\infty}$ norm
- In what follows, we present stability guarantees for both OMP and BP that will also depend on the local energy in the noise vector $\mathbf{E}$
- This will be enforced via the $\ell_{2,\infty}$ norm, defined as:

$$\|\mathbf{E}\|_{2,\infty}^{\mathrm{p}} = \max_{\mathrm{i}} \|\mathbf{R}_{\mathrm{i}}\mathbf{E}\|_2$$



Stability of OMP

Theorem: If $\mathbf{Y} = \mathbf{D\Gamma} + \mathbf{E}$ where

$$\|\mathbf{\Gamma}\|_{0,\infty}^{s} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D})}\right) - \frac{1}{\mu(\mathbf{D})} \cdot \frac{\|\mathbf{E}\|_{2,\infty}^{p}}{|\Gamma_{\min}|}$$

then OMP run for $\|\mathbf{\Gamma}\|_0$ iterations will

1. Find the correct support
2. $\|\mathbf{\Gamma}_{\mathrm{OMP}} - \mathbf{\Gamma}\|_2^2 \leq \frac{\|\mathbf{E}\|_2^2}{1-\left(\|\mathbf{\Gamma}\|_{0,\infty}^{s}-1\right)\mu(\mathbf{D})}$

Stability of Lagrangian BP

$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{\mathrm{BP}} = \min_{\boldsymbol{\Gamma}} \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \lambda\|\boldsymbol{\Gamma}\|_1$$

Theorem: For $\mathbf{Y} = \mathbf{D}\boldsymbol{\Gamma} + \mathbf{E}$, if $\lambda = 4\|\mathbf{E}\|^{\mathrm{p}}$ and

$$\|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathrm{s}} < \frac{1}{3}\left(1 + \frac{}{}\right.$$

Then we are guaranteed that

1. The support of $\boldsymbol{\Gamma}_{\mathrm{BP}}$ is contain
2. $\|\boldsymbol{\Gamma}_{\mathrm{BP}} - \boldsymbol{\Gamma}\|_{\infty} \leq 7.5\|\mathbf{E}\|_{2,\infty}^{\mathrm{p}}$
3. Every entry greater than $7.5\|$
4. $\boldsymbol{\Gamma}_{\mathrm{BP}}$ is unique

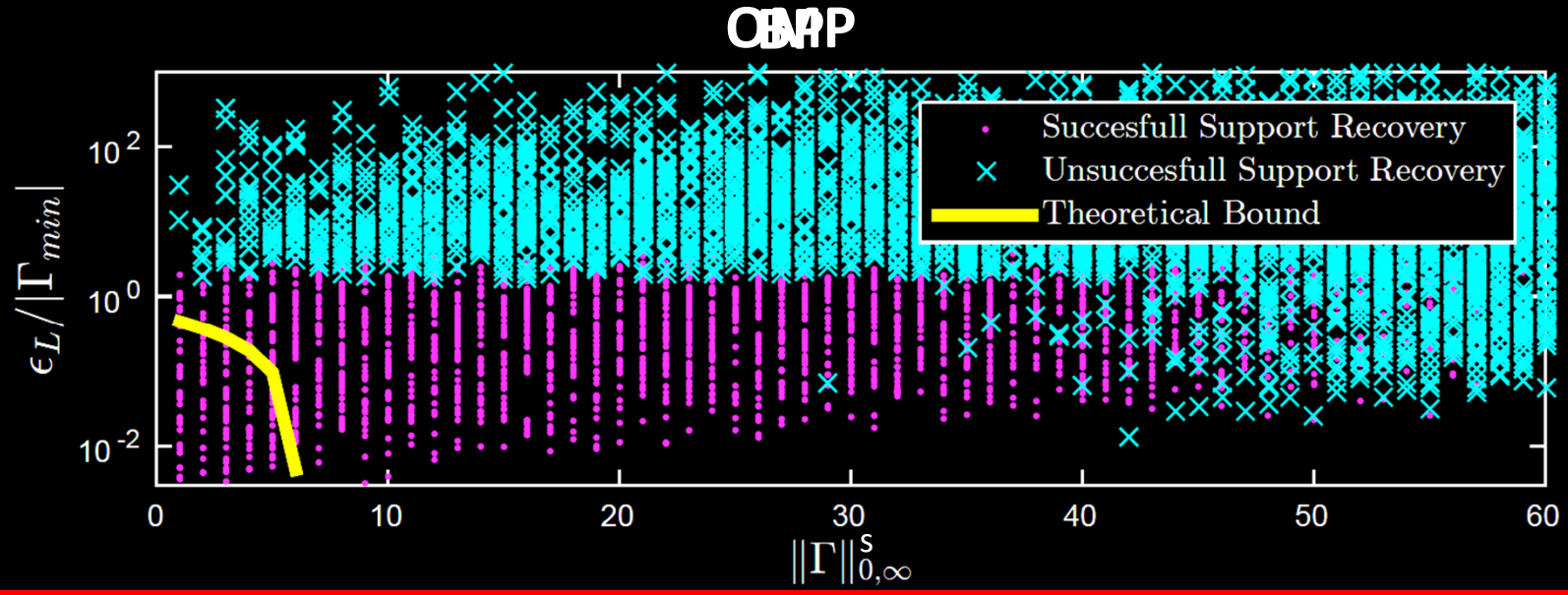
Theoretical foundation for recent works tackling the convolutional sparse coding problem via BP

[Bristow, Eriksson & Lucey '13]
[Wohlberg '14]
[Kong & Fowlkes '14]
[Bristow & Lucey '14]
[Heide, Heidrich & Wetzstein '15]
[Šorel & Šroubek '16]

Proof relies on the work of [Tropp '06]

Phase Transition - Noisy

- We use the same dictionary as in the noiseless case
- We generate random sparse vectors in which the non-zero entries are drawn randomly in the range $[-a, a]$ for different a values
- Given a sparse vector, we compute its global signal and attempt to recover it using the global OMP and BP

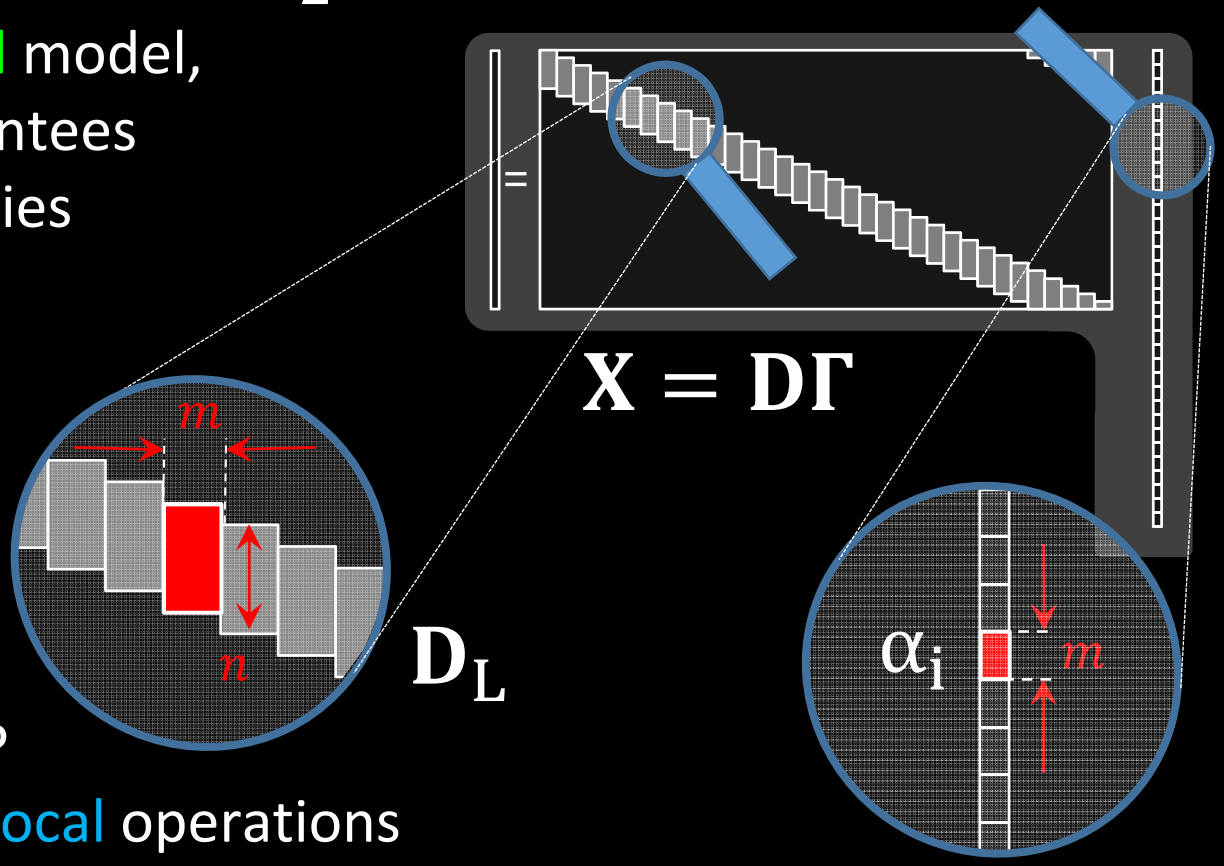


Global Pursuit via Local Processing

$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{\mathrm{BP}} = \min_{\boldsymbol{\Gamma}} \ \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \xi\|\boldsymbol{\Gamma}\|_1$$

- While CSC is a global model, its theoretical guarantees rely on local properties

- We aim to show that this global-local relation can also be exploited for solving the global BP problem using only local operations



Global Pursuit via Local Processing (1)

$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{\mathrm{BP}} = \min_{\boldsymbol{\Gamma}} \ \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \xi\|\boldsymbol{\Gamma}\|_1$$

- Recall: Iterative Soft Thresholding is an appealing method for handling the above minimization task

$$\boldsymbol{\Gamma}^t = \mathcal{S}_{\xi/c}\left(\underbrace{\boldsymbol{\Gamma}^{t-1} + \frac{1}{c}\mathbf{D}^T(\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}^{t-1})}_{\text{Gradient step}}\right)$$

Projection onto $\mathbf{L}_1$ ball

- This algorithm is guaranteed to solve the above problem
  [Daubechies, Defrise, De-Mol, 2004] [Blumensath & Davies '08]
- Proposal: We shall manipulate this algorithm to an equivalent form that operates locally

* $c > 0.5\,\lambda_{\max}(\mathbf{D}^T\mathbf{D})$

Global Pursuit via Local Processing (1)

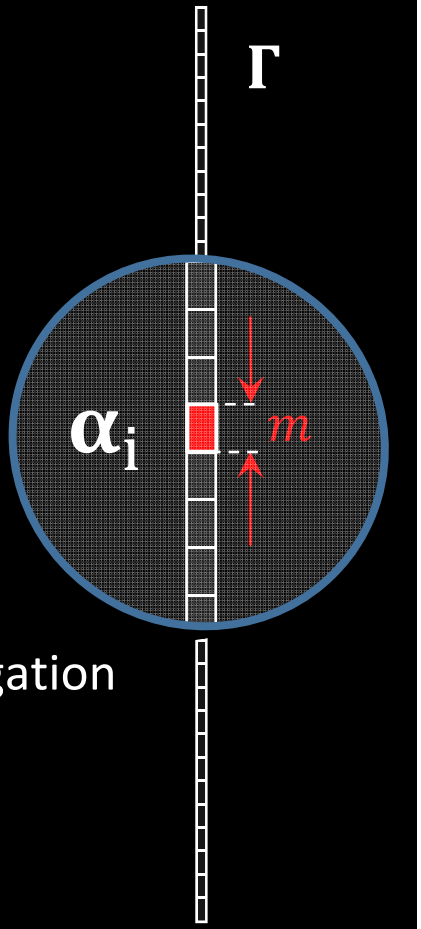
$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{\mathrm{BP}} = \min_{\boldsymbol{\Gamma}} \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \xi\|\boldsymbol{\Gamma}\|_1$$

$$\boldsymbol{\Gamma}^{\mathrm{t}} = \mathcal{S}_{\xi/c}\left(\boldsymbol{\Gamma}^{\mathrm{t-1}} + \frac{1}{\mathrm{c}}\mathbf{D}^{\mathrm{T}}(\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}^{\mathrm{t-1}})\right)$$

This can be equally written as

$\forall i \quad \boldsymbol{\alpha}_i^{\mathrm{t}}$ = local sp… global aggregation … $^{\mathrm{t-1}}))$ residual

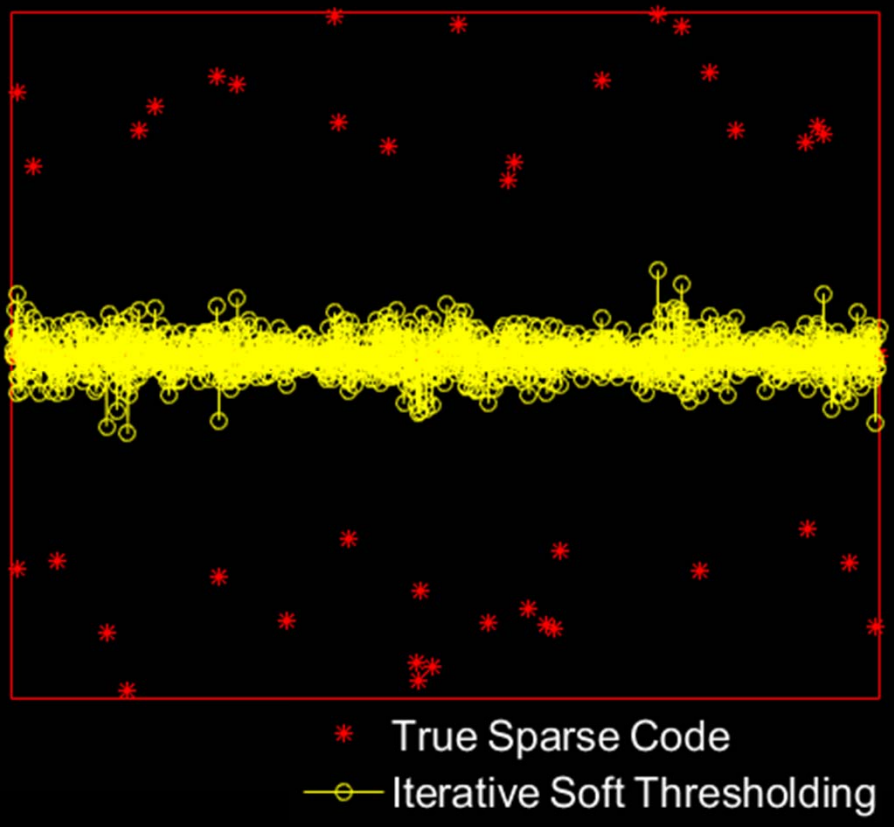
This algorithm operates locally while guaranteeing to solve the global problem



Simulation

Details:

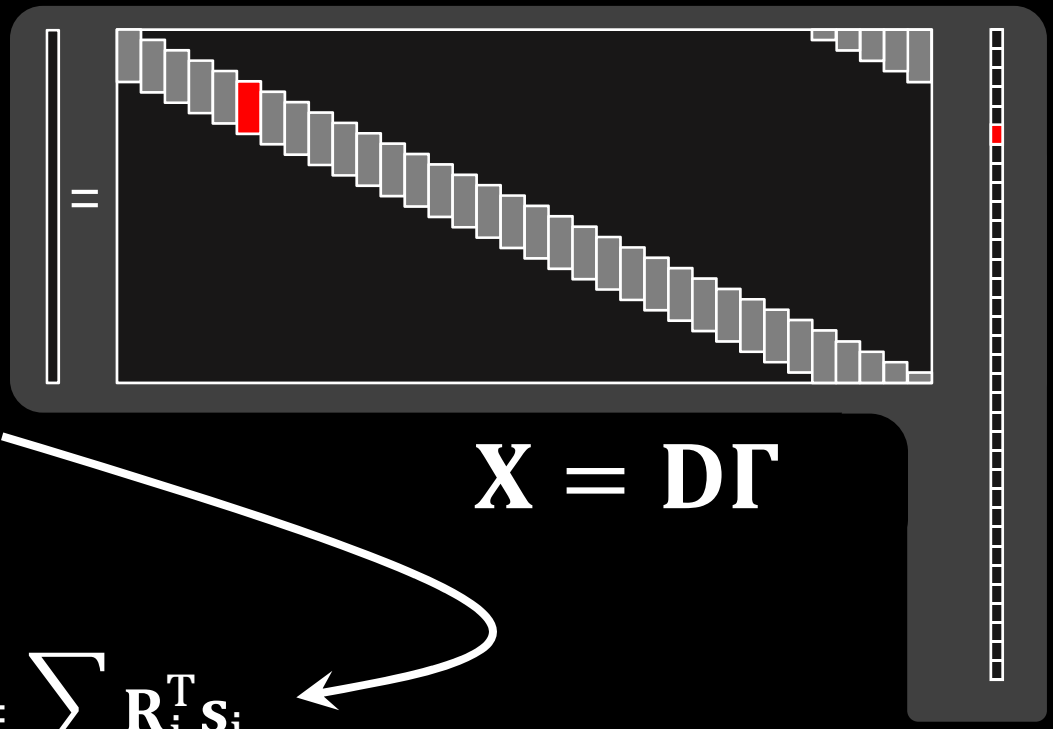
- Signal length: $N = 300$
- Patch size: $n = 25$
- Unique atoms: $p = 5$
- Local sparsity (k) is 11
- Global sparsity: $k = 40$
- Number of iterations: 400
- Lagrangian: $\xi = 4\|\mathbf{E}\|_{2,\infty}^{\mathrm{p}}$
- Noise level: PSNR= 0.03



Global Pursuit via Local Processing (2)

$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{BP} = \min_{\boldsymbol{\Gamma}} \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \xi\|\boldsymbol{\Gamma}\|_1$$

- Here is an alternative approach, based on a different interpretation of this linear system
- $\mathbf{s}_i$ are slices – local patches that overlap to form the full image



$$\mathbf{X} = \mathbf{D}\boldsymbol{\Gamma} = \sum_i \mathbf{R}_i^T \mathbf{D}_L \boldsymbol{\alpha}_i = \sum_i \mathbf{R}_i^T \mathbf{s}_i$$

Global Pursuit via Local Processing (2)

$$(\mathbf{P}_1^{\epsilon}): \quad \boldsymbol{\Gamma}_{\mathrm{BP}} = \min_{\boldsymbol{\Gamma}} \ \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\boldsymbol{\Gamma}\|_2^2 + \lambda\|\boldsymbol{\Gamma}\|_1$$

Turning to the local form and using the Augmented Lagrangian

$$\min_{\boldsymbol{\alpha}_i,\mathbf{s}_i} \ \frac{1}{2}\left\|\mathbf{Y}-\sum_i \mathbf{R}_i^T\mathbf{s}_i\right\|_2^2 + \sum_i\left(\lambda\|\boldsymbol{\alpha}_i\|_1 + \frac{\rho}{2}\|\mathbf{s}_i-\mathbf{D}_L\boldsymbol{\alpha}_i+u_i\|_2^2\right)$$

- These two problems are equivalent, and convex w.r.t their variables
- The new formulation targets the local slices, and their sparse representations
- The vectors u_i are the Lagrange multipliers for the constraints $\mathbf{s}_i=\mathbf{D}_L\boldsymbol{\alpha}_i$

Global Pursuit via Local Processing (2)

$$\min_{\boldsymbol{\alpha}_i,\mathbf{s}_i} \frac{1}{2}\left\|\mathbf{Y}-\sum_i \mathbf{R}_i^T\mathbf{s}_i\right\|_2^2 + \sum_i\left(\lambda\|\boldsymbol{\alpha}_i\|_1 + \frac{\rho}{2}\|\mathbf{s}_i-\mathbf{D}_L\boldsymbol{\alpha}_i+\mathrm{u}_i\|_2^2\right)$$

ADMM

- Slice-update: $\min_{\mathbf{s}_i} \frac{1}{2}\left\|\mathbf{Y}-\sum_i \mathbf{R}_i^T\mathbf{s}_i\right\|^2 + \sum \rho \|\mathbf{s}_i-\mathbf{D}_L\boldsymbol{\alpha}_i+\mathrm{u}_i\|_2^2$

  nd averaging

- Sparse-Upda $\frac{1}{2}\|\mathbf{s}_i-\mathbf{D}_L\boldsymbol{\alpha}_i+\mathrm{u}_i\|_2^2\Big)$

  Separable and local LARS problems

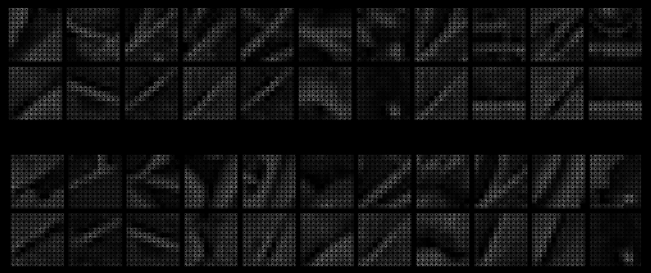
This algorithm operates locally while guaranteeing to solve the global problem

Comment: One iteration of this procedure amounts to … the very same patch-averaging algorithm we started with

Two Comments About this Scheme

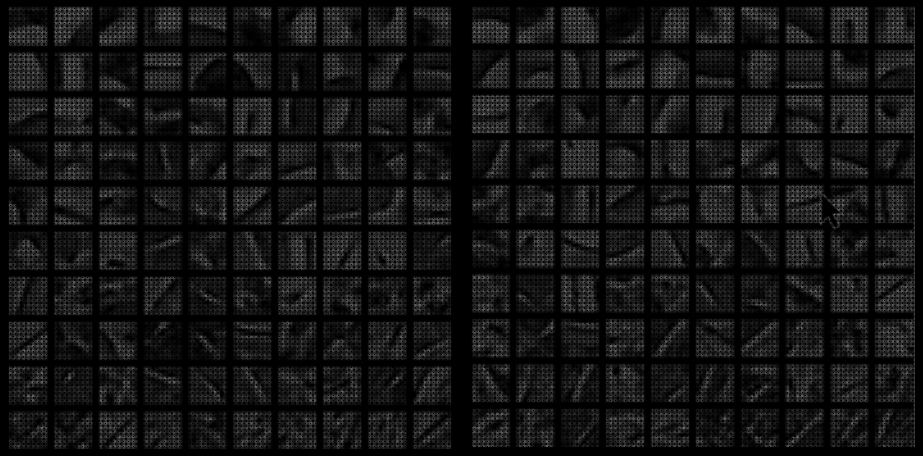
We work with Slices and not Patches

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

Partial Summary of CSC

- What we have seen so far is a new way to analyze the global CSC model using local sparsity constraints. We proved:
  -  **Uniqueness** of the solution for the noiseless problem
  -  **Stability** of the solution for the noisy problem
  -  **Guarantee of success and stability** of both OMP and BP
  -  We obtained guarantees and algorithms that operate locally while claiming global optimality
  -  We mentioned briefly the mater of learning the model (i.e. dictionary learning for CSC), and presented our competitive approach which is based on simple local steps

Part III
Going Deeper

Convolutional Neural Networks

Analyzed via

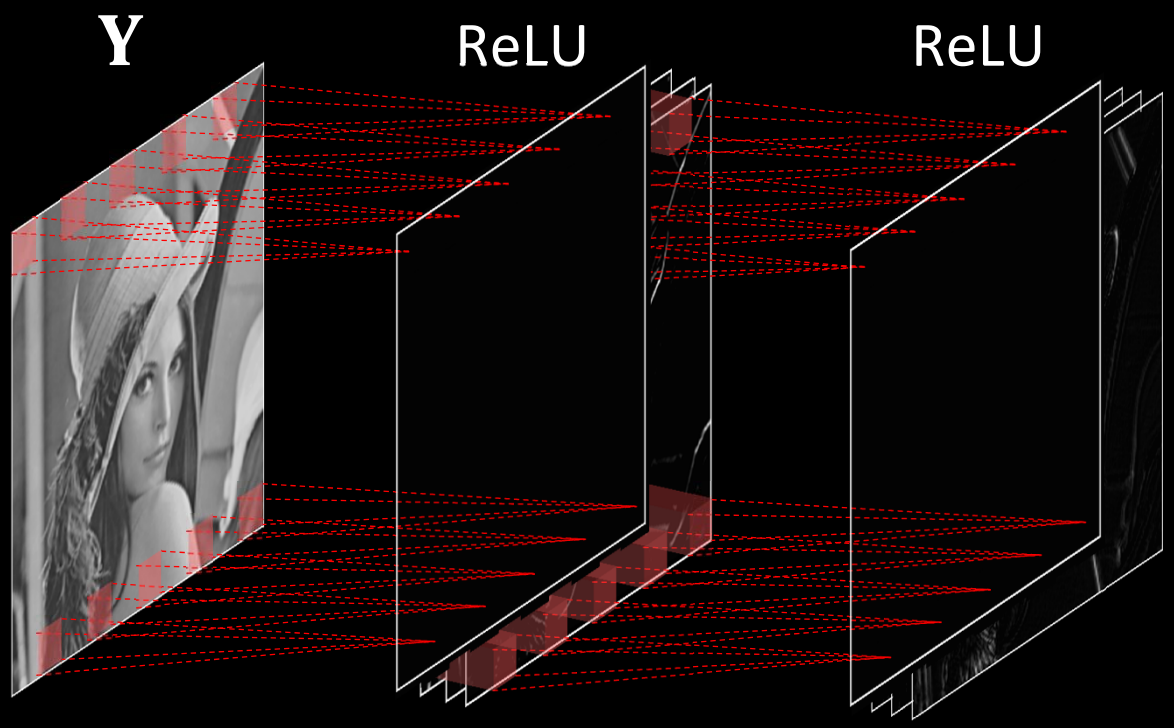
Convolutional Sparse Coding

Vardan Papyan, Yaniv Romano and Michael Elad

CSC and CNN

- There is an analogy between CSC and CNN:
  - Convolutional structure
  - Data driven models
  - ReLU is a sparsifying operator
- We propose a principled way to analyze CNN
- But first, a short review of CNN…

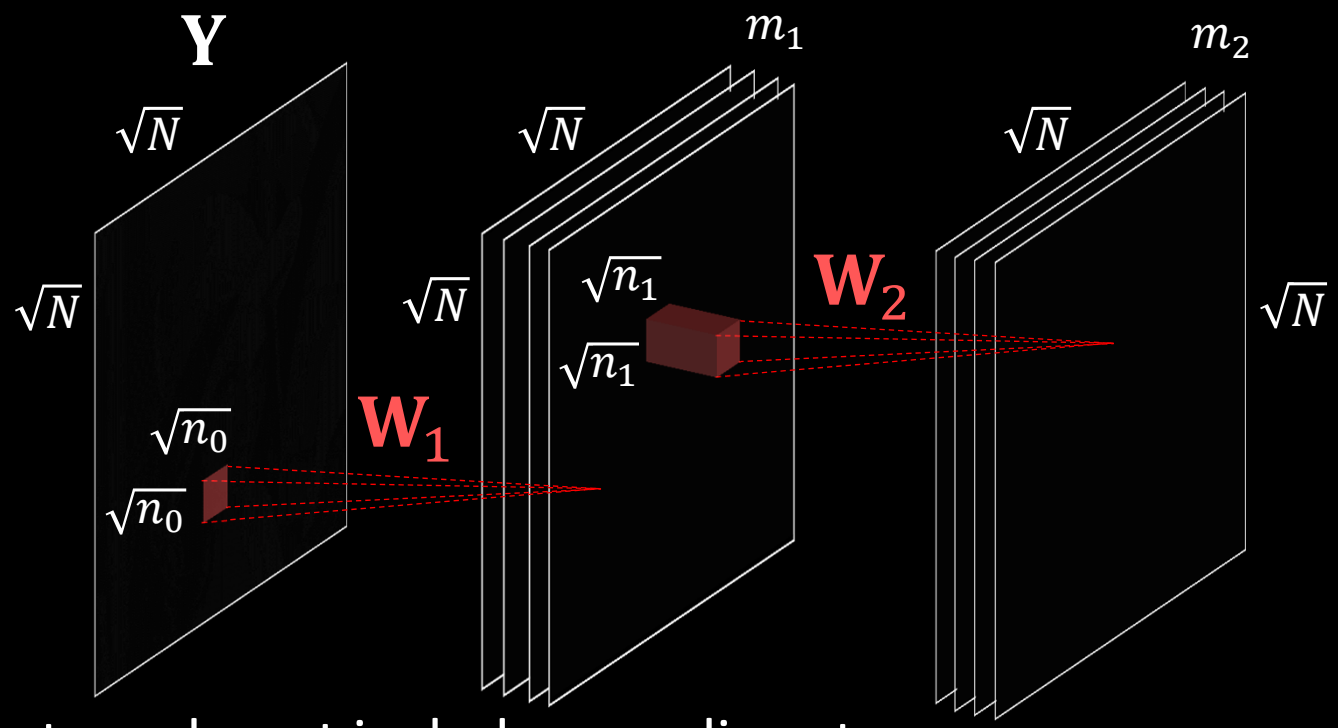
CNN



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

$$\mathrm{ReLU}(z) = \max(\mathrm{Thr}, z)$$

CNN



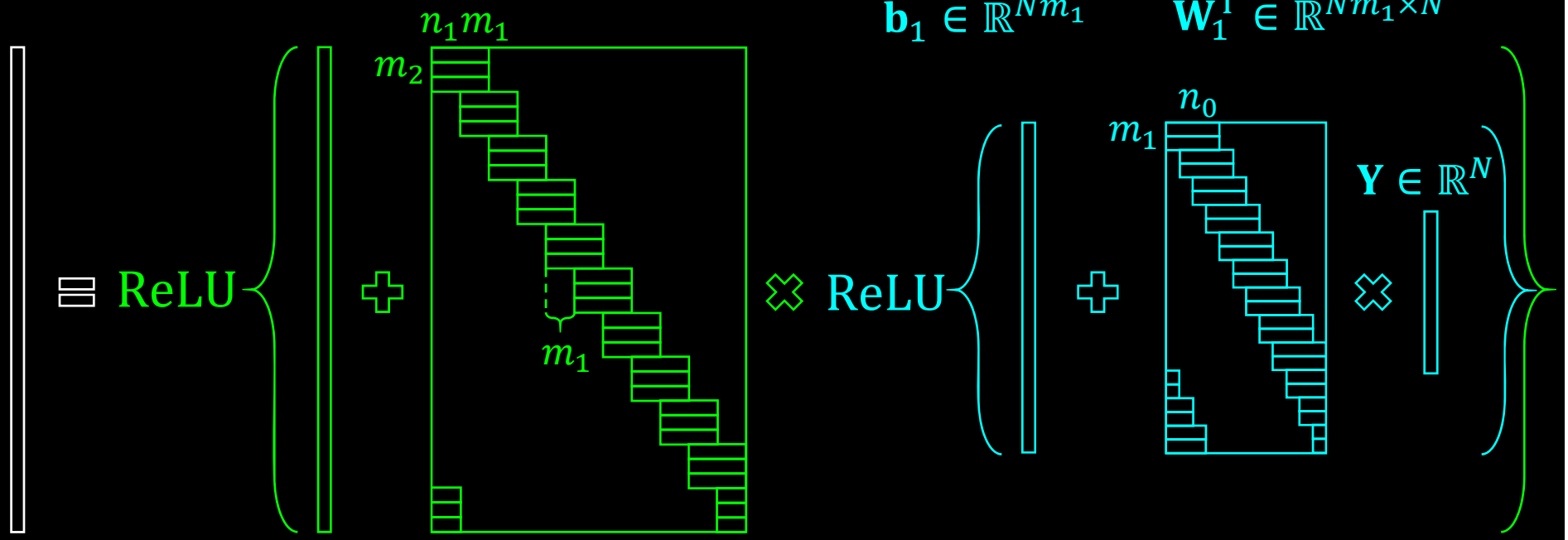
Notice that we do not include a pooling stage:

- Can be replaced by a convolutional layer with increased stride without loss in performance [Springenberg, Dosovitskiy, Brox & Riedmiller '14]
- The current state-of-the-art in image recognition does not use it [He, Zhang, Ren & Sun '15]

Mathematically…

$$f(\mathbf{Y},\{\mathbf{W}_i\},\{\mathbf{b}_i\}) = \mathrm{ReLU}\left(\mathbf{b}_2 + \mathbf{W}_2^{\mathrm{T}}\,\mathrm{ReLU}\left(\mathbf{b}_1 + \mathbf{W}_1^{\mathrm{T}}\mathbf{X}\right)\right)$$

$\mathbf{Z}_2 \in \mathbb{R}^{Nm_2}$ $\mathbf{b}_2 \in \mathbb{R}^{Nm_2}$ $\mathbf{W}_2^{\mathrm{T}} \in \mathbb{R}^{Nm_2 \times Nm_1}$ $\mathbf{b}_1 \in \mathbb{R}^{Nm_1}$ $\mathbf{W}_1^{\mathrm{T}} \in \mathbb{R}^{Nm_1 \times N}$ $\mathbf{Y} \in \mathbb{R}^{N}$



Training Stage of CNN

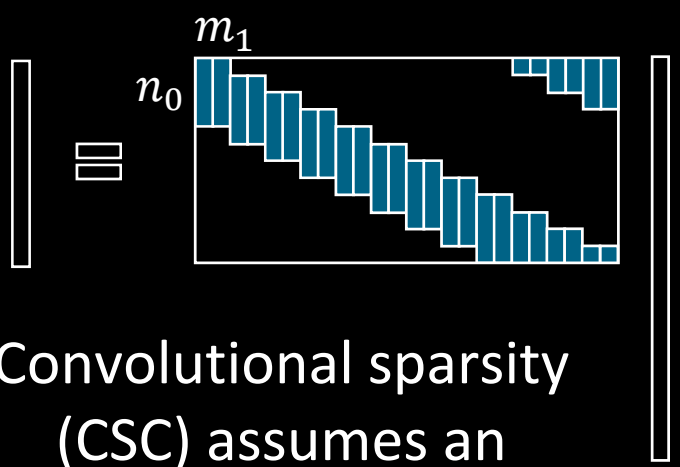
- Consider the task of classification
- Given a set of signals $\{\mathbf{Y}_j\}_j$ and their corresponding labels $\{h(\mathbf{Y}_j)\}_j$, the CNN learns an end-to-end mapping

$$\min_{\{\mathbf{W}_i\},\{\mathbf{b}_i\},U} \sum_j \ell\left(h(\mathbf{Y}_j), U, f\left(\mathbf{Y}_j, \{\mathbf{W}_i\}, \{\mathbf{b}_i\}\right)\right)$$

True label Classifier Output of last layer

Back to CSC

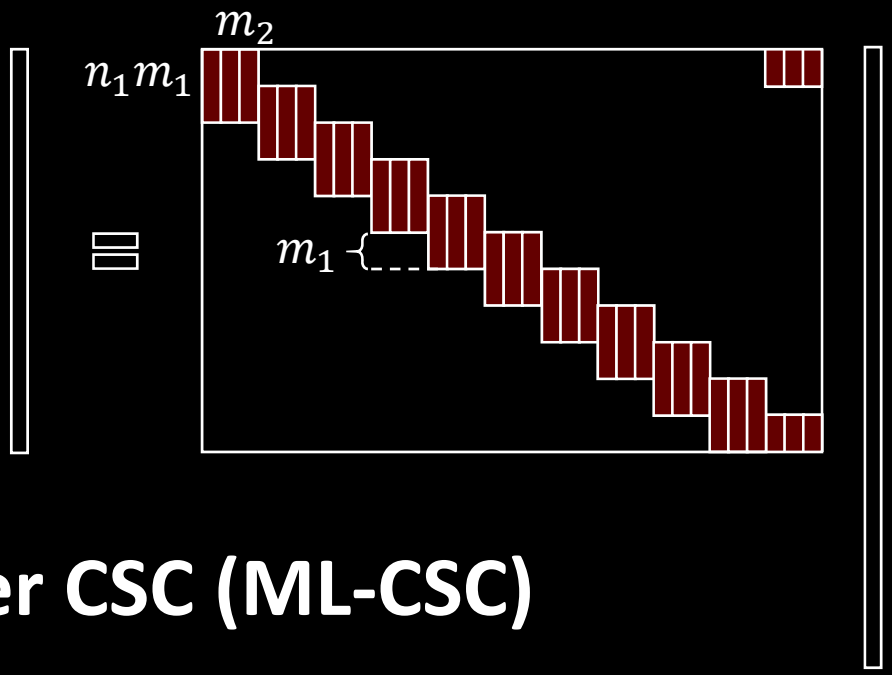
$\mathbf{X} \in \mathbb{R}^{N}$ $\mathbf{D}_1 \in \mathbb{R}^{N \times Nm_1}$ $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{Nm_1}$



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

We propose to impose the same structure on the representations themselves

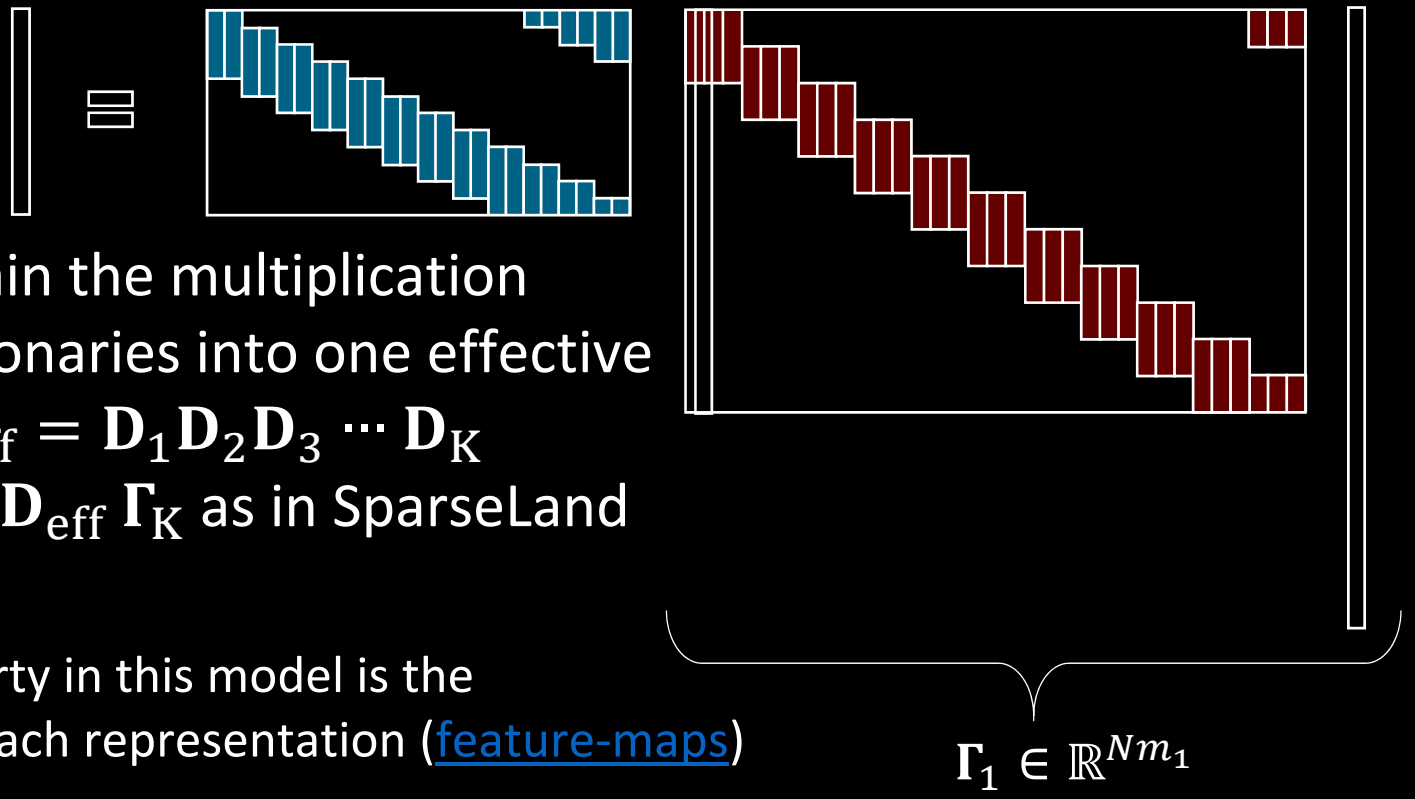
$\boldsymbol{\Gamma}_1 \in \mathbb{R}^{Nm_1}$ $\mathbf{D}_2 \in \mathbb{R}^{Nm_1 \times Nm_2}$ $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{Nm_2}$



Multi-Layer CSC (ML-CSC)

Intuition: From Atoms to Molecules

$\mathbf{X} \in \mathbb{R}^N$ $\mathbf{D}_1 \in \mathbb{R}^{N \times Nm_1}$ $\mathbf{\Gamma}_1 \in \mathbb{R}^{Nm_1}$ $\mathbf{D}_2 \in \mathbb{R}^{Nm_1 \times Nm_2}$ $\mathbf{\Gamma}_2 \in \mathbb{R}^{Nm_2}$



$\mathbf{\Gamma}_1 \in \mathbb{R}^{Nm_1}$

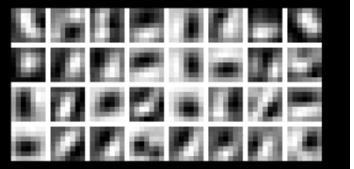
- One could chain the multiplication of all the dictionaries into one effective dictionary $\mathbf{D}_{\text{eff}} = \mathbf{D}_1\mathbf{D}_2\mathbf{D}_3 \cdots \mathbf{D}_{\text{K}}$ and then $\mathbf{x} = \mathbf{D}_{\text{eff}}\,\mathbf{\Gamma}_{\text{K}}$ as in SparseLand
- However:
  - A key property in this model is the sparsity of each representation (feature-maps)
  - The effective atoms are combinations of the original atoms - molecules

A Small Taste: Model Training (MNIST)

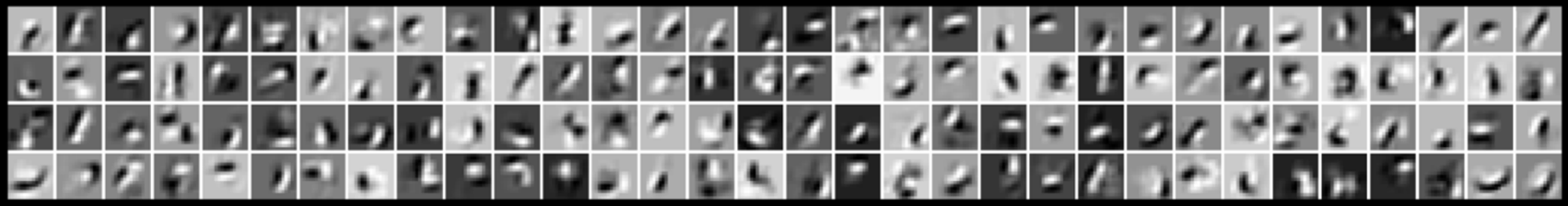
MNIST Dictionary:

- D_1: 32 filters of size 7×7, with stride of 2 (dense)
- D_2: 128 filters of size 5×5×32 with stride of 1 - 99.09 % sparse
- D3: 1024 filters of size 7×7×128 – 99.89 % sparse

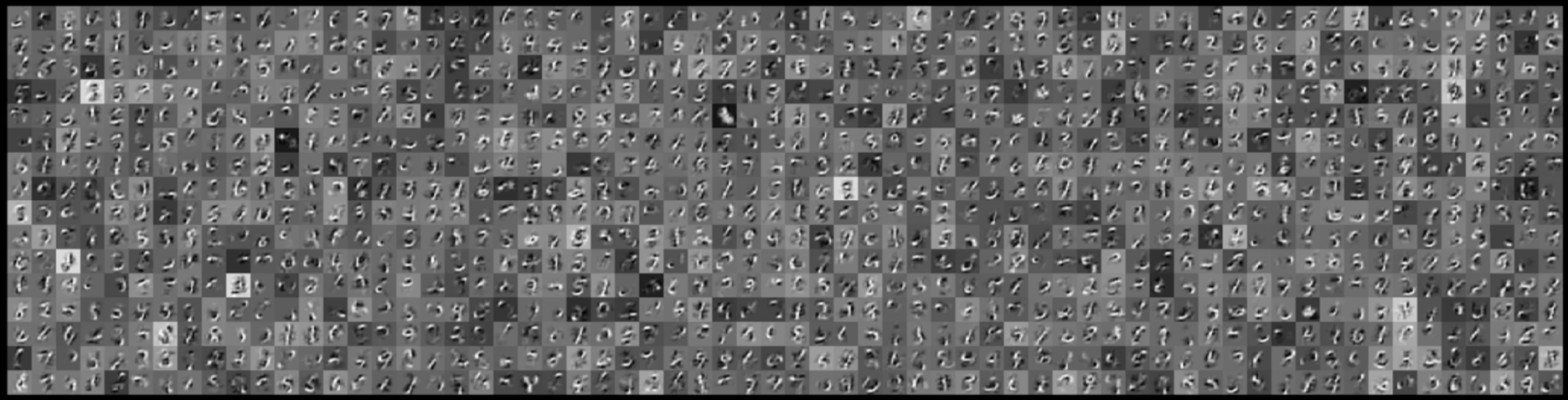
$\mathbf{D}_1$ (7×7)



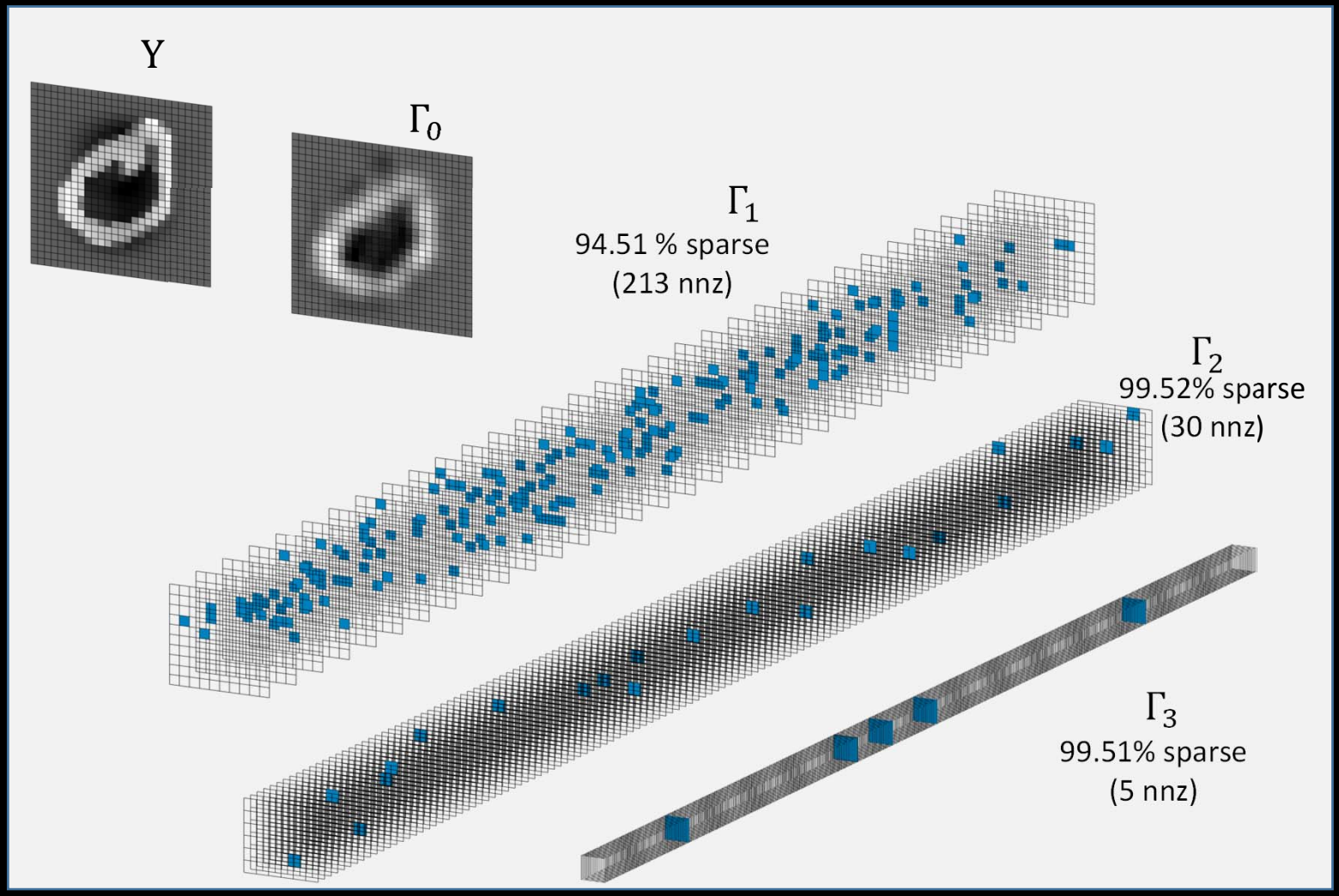
$\mathbf{D}_1\mathbf{D}_2$ (15×15)



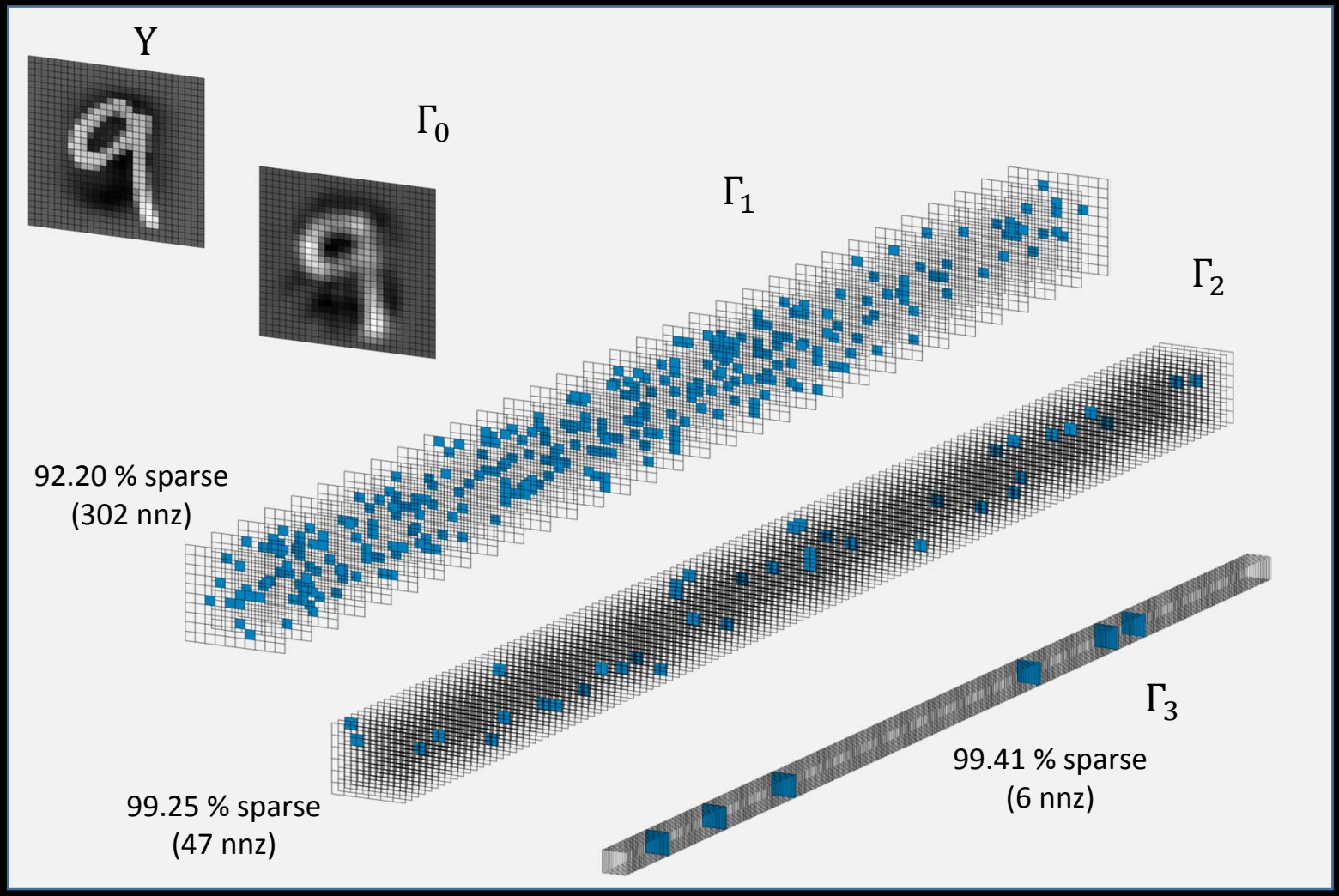
$\mathbf{D}_1\mathbf{D}_2\mathbf{D}_3$ (28×28)



A Small Taste: Pursuit



A Small Taste: Pursuit



A Small Taste: Model Training (CFAR)

$\mathbf{D}_1$ (5×5×3)

$\mathbf{D}_1\mathbf{D}_2$ (13×13)

$\mathbf{D}_1\mathbf{D}_2\mathbf{D}_3$ (32×32)

CIFAR Dictionary:

- D_1: 64 filters of size 5x5x3, stride of 2 dense
- D_2: 256 filters of size 5x5x64, stride of 2 82.99 % sparse
- D_3: 1024 filters of size 5x5x256 90.66 % sparse

ML-CSC: Pursuit

- Deep–Coding Problem ($\mathbf{DCP}_\lambda$) (dictionaries are known):

$$\text{Find} \quad \{\boldsymbol{\Gamma}_j\}_{j=1}^{K} \quad s.t. \quad \begin{cases} \mathbf{X} = \mathbf{D}_1\boldsymbol{\Gamma}_1 & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} \le \lambda_1 \\ \boldsymbol{\Gamma}_1 = \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} \le \lambda_2 \\ \vdots & \vdots \\ \boldsymbol{\Gamma}_{K-1} = \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} \le \lambda_K \end{cases}$$

- Or, more realistically for noisy signals,

$$\text{Find} \quad \{\boldsymbol{\Gamma}_j\}_{j=1}^{K} \quad s.t. \quad \begin{cases} \|\mathbf{Y} - \mathbf{D}_1\boldsymbol{\Gamma}_1\|_2 \le \mathcal{E} & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} \le \lambda_1 \\ \boldsymbol{\Gamma}_1 = \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} \le \lambda_2 \\ \vdots & \vdots \\ \boldsymbol{\Gamma}_{K-1} = \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} \le \lambda_K \end{cases}$$

ML-CSC: Dictionary Learning

- Deep-Learning Problem ($\mathbf{DLP}_\lambda$):

$$\text{Find} \quad \{\mathbf{D}_i\}_{i=1}^{K} \quad s.t. \quad \left\{ \begin{array}{ll} \left\|\mathbf{Y}_j - \mathbf{D}_1\boldsymbol{\Gamma}_1^j\right\|_2 \le \varepsilon & \left\|\boldsymbol{\Gamma}_1^j\right\|_{0,\infty}^s \le \lambda_1 \\ \boldsymbol{\Gamma}_2^j = \mathbf{D}_2\boldsymbol{\Gamma}_1^2 & \left\|\boldsymbol{\Gamma}_2^j\right\|_{0,\infty}^s \le \lambda_2 \\ \vdots & \vdots \\ \boldsymbol{\Gamma}_K^j = \mathbf{D}_K\boldsymbol{\Gamma}_K^j & \left\|\boldsymbol{\Gamma}_K^j\right\|_{0,\infty}^s \le \lambda_K \end{array} \right\}_{j=1}^{J}$$

- While the above is an unsupervised DL, a supervised version can be envisioned

  [Mairal, Bach & Ponce '12]

$$\min_{\{\mathbf{D}_i\}_{i=1}^{K},\mathbf{U}} \sum_j \ell\left(\mathrm{h}\left(\mathbf{Y}_j\right), \mathbf{U}, \mathbf{DCP}^\star\left(\mathbf{Y}_j, \{\mathbf{D}_i\}\right)\right)$$

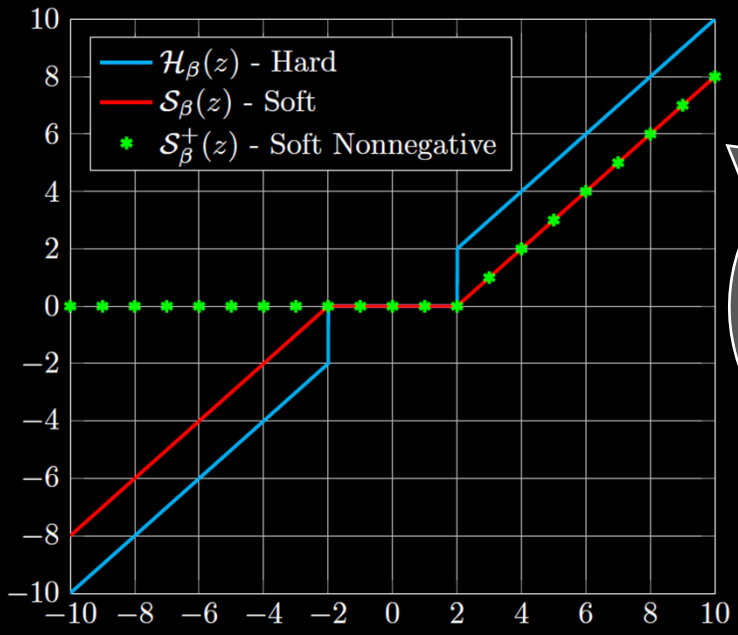
The deepest representation $\boldsymbol{\Gamma}_K$ obtained by solving the DCP

ML-CSC: The Simplest Pursuit

- The simplest pursuit algorithm (single-layer case) is the THR algorithm, which operates on a given input signal $\mathbf{Y}$ by:

$$\mathbf{Y} = \mathbf{D}\boldsymbol{\Gamma} + \boldsymbol{E} \text{ and } \boldsymbol{\Gamma} \text{ is sparse} \quad \Rightarrow \quad \hat{\boldsymbol{\Gamma}} = \mathcal{P}_{\beta}\left(\mathbf{D}^{\mathrm{T}}\mathbf{Y}\right)$$

- Restricting the coefficients to be nonnegative does not restrict the expressiveness of the model



ReLU = Soft Nonnegative Thresholding

Consider this for Solving the DCP

- Layered thresholding (LT):

Estimate $\boldsymbol{\Gamma}_1$ via the THR algorithm

$$\widehat{\boldsymbol{\Gamma}}_2 = \mathcal{P}_{\beta_2}\left(\mathbf{D}_2^{\mathrm{T}}\,\mathcal{P}_{\beta_1}\left(\mathbf{D}_1^{\mathrm{T}}\mathbf{Y}\right)\right)$$

Estimate $\boldsymbol{\Gamma}_2$ via the THR algorithm

$$\left(\mathbf{DCP}_\lambda^{\mathcal{E}}\right)\text{: Find } \left\{\boldsymbol{\Gamma}_j\right\}_{j=1}^{K} \quad s.t.$$

$$\begin{cases} \|\mathbf{Y}-\mathbf{D}_1\boldsymbol{\Gamma}_1\|_2 \le \mathcal{E} & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} \le \lambda_1 \\ \boldsymbol{\Gamma}_1 = \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} \le \lambda_2 \\ \vdots & \vdots \\ \boldsymbol{\Gamma}_{K-1} = \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} \le \lambda_K \end{cases}$$

- Forward pass of CNN:

$$f(\mathbf{X}) = \mathrm{ReLU}\left(\mathbf{b}_2 + \mathbf{W}_2^{\mathrm{T}}\,\mathrm{ReLU}\left(\mathbf{b}_1 + \mathbf{W}_1^{\mathrm{T}}\mathbf{Y}\right)\right)$$

The layered (soft nonnegative) thresholding and the forward pass algorithm are the very same things !!!

Consider this for Solving the DLP

- DLP (supervised*):

$$\min_{\{\mathbf{D}_i\}_{i=1}^{K},\mathbf{U}} \sum_j \ell\left(\mathbf{h}\left(\mathbf{Y}_j\right), \mathbf{U}, \underbrace{\mathbf{DCP}^{\star}\left(\mathbf{Y}_j, \{\mathbf{D}_i\}\right)}\right)$$

Estimate via the layered THR algorithm

The thresholds for the DCP should also learned

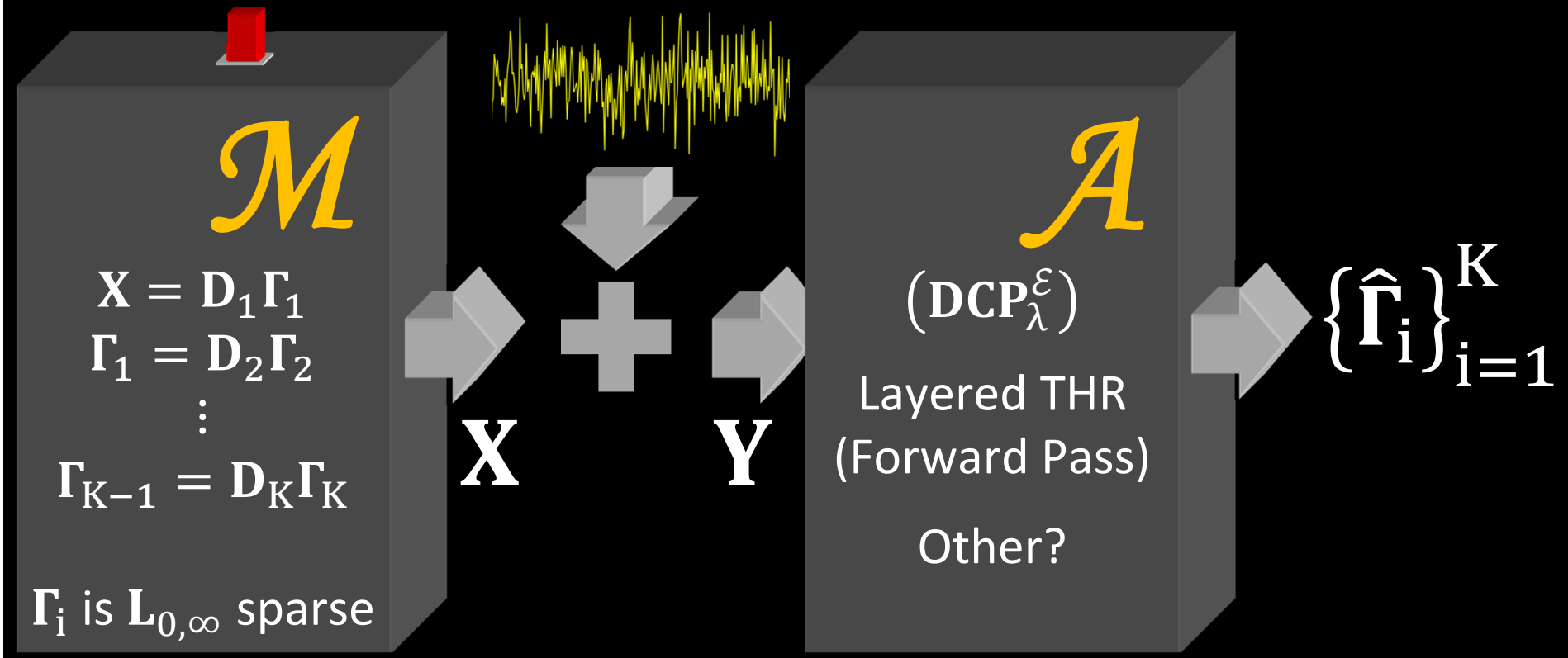
- CNN training:

$$\min_{\{\mathbf{W}_i\},\{\mathbf{b}_i\},\mathbf{U}} \sum_j \ell\left(\mathbf{h}\left(\mathbf{Y}_j\right), \mathbf{U}, f\left(\mathbf{Y}, \{\mathbf{W}_i\}, \{\mathbf{b}_i\}\right)\right)$$

The problem solved by the training stage of CNN and the DLP are equivalent as well, assuming that the DCP is approximated via the layered thresholding algorithm

* Recall that for the ML-CSC, there exists an unsupervised avenue for training the dictionaries that has no simple parallel in CNN

Theoretical Path



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

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. Uniqueness of the solution (set of representations) to the $(\mathbf{DCP}_\lambda)$?
II. Stability of the solution to the $\left(\mathbf{DCP}_\lambda^\mathcal{E}\right)$ problem ?

III. Stability of the solution obtained via the hard and soft layered THR algorithms (forward pass) ?
IV. Limitations of this (very simple) algorithm and alternative pursuit?

V. Algorithms for training the dictionaries $\{\mathbf{D}_i\}_{i=1}^K$ vs. CNN ?
VI. New insights on how to operate on signals via CNN ?

Uniqueness of $(\mathbf{DCP}_\lambda)$

$(\mathbf{DCP}_\lambda)$: Find a set of representations satisfying

$$\begin{aligned} \mathbf{X} &= \mathbf{D}_1\boldsymbol{\Gamma}_1 & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} &\le \lambda_1 \\ \boldsymbol{\Gamma}_1 &= \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} &\le \lambda_2 \\ &\vdots & &\vdots \\ \boldsymbol{\Gamma}_{K-1} &= \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} &\le \lambda_K \end{aligned}$$

Is this set unique?

Theorem: If a set of solutions $\{\boldsymbol{\Gamma}_i\}_{i=1}^{K}$ is found for $(\mathbf{DCP}_\lambda)$ such that:

$$\|\boldsymbol{\Gamma}_i\|_{0,\infty}^{s} \le \lambda_i < \frac{1}{2}\left(1+\frac{1}{\mu(\mathbf{D}_i)}\right)$$

then these are necessarily the unique solution to the DCP problem

The feature maps CNN aims to recover are unique

Stability of $\left(\mathbf{DCP}_{\lambda}^{\mathcal{E}}\right)$

Is this set stable?

- The problem we aim to solve is this

$\left(\mathbf{DCP}_{\lambda}^{\mathcal{E}}\right)$: Find a set of representations satisfying

$$\begin{aligned} \|\mathbf{Y}-\mathbf{D}_1\boldsymbol{\Gamma}_1\|_2 \le \mathcal{E} \quad & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} \le \lambda_1 \\ \boldsymbol{\Gamma}_1 = \mathbf{D}_2\boldsymbol{\Gamma}_2 \quad & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} \le \lambda_2 \\ \vdots \quad & \vdots \\ \boldsymbol{\Gamma}_{K-1} = \mathbf{D}_K\boldsymbol{\Gamma}_K \quad & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} \le \lambda_K \end{aligned}$$

- Suppose that we manage to solve the $\left(\mathbf{DCP}_{\lambda}^{\mathcal{E}}\right)$ and find a feasible set of representations satisfying all the conditions

$$\left\{\hat{\boldsymbol{\Gamma}}_i\right\}_{i=1}^{K}$$

- The question we pose is How close is $\hat{\boldsymbol{\Gamma}}_i$ to $\boldsymbol{\Gamma}_i$?

Stability of $\left(\mathbf{DCP}_{\lambda}^{\mathcal{E}}\right)$

Theorem: If the true representations $\{\boldsymbol{\Gamma}_i\}_{i=1}^K$ satisfy

$$\|\boldsymbol{\Gamma}_i\|_{0,\infty}^s \leq \lambda_i < \frac{1}{2}\left(1+\frac{1}{\mu(\mathbf{D}_i)}\right)$$

then the set of solutions $\{\widehat{\boldsymbol{\Gamma}}_i\}_{i=1}^K$ obtained by solving this problem (somehow) must obey

$$\left\|\widehat{\boldsymbol{\Gamma}}_i - \boldsymbol{\Gamma}_i\right\|_2^2 \leq \mathcal{E}_i^2 \quad \text{for}$$

$$\mathcal{E}_0^2 = 4\mathcal{E}^2, \qquad \mathcal{E}_i^2 = \frac{\mathcal{E}_{i-1}^2}{1-(2\lambda_i-1)\mu(\mathbf{D}_i)}$$

The problem CNN aims to solve is stable under certain conditions

Observe this annoying effect of error magnification as we dive into the model

Stability of Layered-THR

Theorem: If $\|\boldsymbol{\Gamma}_i\|_{0,\infty}^{s} < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D}_i)} \cdot \frac{|\boldsymbol{\Gamma}_i^{min}|}{|\boldsymbol{\Gamma}_i^{max}|}\right) - \frac{1}{\mu(\mathbf{D}_i)} \cdot \frac{\varepsilon_L^{i-1}}{|\boldsymbol{\Gamma}_i^{max}|}$

then the layered hard THR (with the proper thresholds) will find the correct supports* and

$$\left\|\boldsymbol{\Gamma}_i^{LT} - \boldsymbol{\Gamma}_i\right\|_{2,\infty}^{p} \leq \varepsilon_L^i$$

where we have defined $\varepsilon_L^0 = \|\mathbf{E}\|_{2,\infty}^{p}$ and

$$\varepsilon_L^i = \sqrt{\|\boldsymbol{\Gamma}_i\|_{0,\infty}^{p}} \cdot \left(\varepsilon_L^{i-1} + \mu(\mathbf{D}_i)\left(\|\boldsymbol{\Gamma}_i\|_{0,\infty}^{s} - 1\right)|\boldsymbol{\Gamma}_i^{max}|\right)$$

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

* Least-Squares update of the non-zeros?

Limitations of the Forward Pass

- The stability analysis reveals several inherent limitations of the forward pass (a.k.a. Layered THR) algorithm:
  - Even in the noiseless case, the forward pass is incapable of recovering the perfect solution of the DCP problem
  - Its success depends on the ratio $\left|\boldsymbol{\Gamma}_i^{\min}\right|/|\boldsymbol{\Gamma}_i^{\max}|$. This is a direct consequence of relying on a simple thresholding operator
  - The distance between the true sparse vector and the estimated one increases exponentially as a function of the layer depth
- We now turn to propose a new algorithm that attempts to solve some of these problems

Special Case – Sparse Dictionaries

- Throughout the theoretical study we assumed that the representations in the different layers are $\mathbf{L}_{0,\infty}$-sparse
- Do we know of a simple example of a set of dictionaries $\{\mathbf{D}_i\}_{i=1}^{K}$ and their corresponding signals $\mathbf{X}$ that will obey this property?

Yes, we do...

- Assuming the dictionaries are sparse:

$$\|\boldsymbol{\Gamma}_j\|_{0,\infty}^{s} \leq \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} \prod_{i=j+1}^{K} \|\mathbf{D}_i\|_0$$

Maximal number of non-zeros in an atom in $\mathbf{D}_i$

- In the context of CNN, the above happens if a sparsity promoting regularization, such as the $\mathbf{L}_1$, is employed on the filters

Better Pursuit ?

- $(\mathbf{DCP}_\lambda)$ Noiseless: Find a set of representations satisfying

$$\begin{aligned} \mathbf{X} &= \mathbf{D}_1\boldsymbol{\Gamma}_1 & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} &\le \lambda_1 \\ \boldsymbol{\Gamma}_1 &= \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} &\le \lambda_2 \\ &\vdots & &\vdots \\ \boldsymbol{\Gamma}_{K-1} &= \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} &\le \lambda_K \end{aligned}$$

- So far we proposed the Layered THR:

$$\widehat{\boldsymbol{\Gamma}}_K = \mathcal{P}_{\beta_K}\left(\mathbf{D}_K^T \ldots \mathcal{P}_{\beta_2}\left(\mathbf{D}_2^T\,\mathcal{P}_{\beta_1}\left(\mathbf{D}_1^T\mathbf{X}\right)\right)\right)$$

- The motivation is clear – getting close to what CNN use
- However, this is the simplest and weakest pursuit known in the field of sparsity – Can we offer something better?

Layered Basis Pursuit (Noiseless)

- Our Goal: $(\mathbf{DCP}_\lambda)$: Find a set of representations satisfying

$$\begin{aligned}
\mathbf{X} &= \mathbf{D}_1\boldsymbol{\Gamma}_1 & \|\boldsymbol{\Gamma}_1\|_{0,\infty}^{s} &\le \lambda_1 \\
\boldsymbol{\Gamma}_1 &= \mathbf{D}_2\boldsymbol{\Gamma}_2 & \|\boldsymbol{\Gamma}_2\|_{0,\infty}^{s} &\le \lambda_2 \\
&\vdots & &\vdots \\
\boldsymbol{\Gamma}_{K-1} &= \mathbf{D}_K\boldsymbol{\Gamma}_K & \|\boldsymbol{\Gamma}_K\|_{0,\infty}^{s} &\le \lambda_K
\end{aligned}$$

- We can propose a Layered Basis Pursuit Algorithm:

$$\boldsymbol{\Gamma}_1^{\mathrm{LBP}} = \min_{\boldsymbol{\Gamma}_1} \|\boldsymbol{\Gamma}_1\|_1 \quad \text{s.t.} \quad \mathbf{X}=\mathbf{D}_1\boldsymbol{\Gamma}_1$$

$$\boldsymbol{\Gamma}_2^{\mathrm{LBP}} = \min_{\boldsymbol{\Gamma}_2} \|\boldsymbol{\Gamma}_2\|_1 \quad \text{s.t.} \quad \boldsymbol{\Gamma}_1^{\mathrm{LBP}} = \mathbf{D}_2\boldsymbol{\Gamma}_2$$

$\bullet\bullet\bullet$

Deconvolutional networks
[Zeiler, Krishnan, Taylor & Fergus '10]

Guarantee for Success of Layered BP

- As opposed to prior work in CNN, we can do far more than just proposing an algorithm – we can analyze its terms for success:

Theorem: If a set of representations $\{\mathbf{\Gamma}_i\}_{i=1}^K$ of the Multi-Layered CSC model satisfy

$$\|\mathbf{\Gamma}_i\|_{0,\infty}^s \leq \lambda_i < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{D}_i)}\right)$$

then the Layered BP is guaranteed to find them

- Consequences:
  - The layered BP can retrieve the underlying representations in the noiseless case, a task in which the forward pass fails to provide
  - The Layered-BP's success does not depend on the ratio $\left|\mathbf{\Gamma}_i^{min}\right|/|\mathbf{\Gamma}_i^{max}|$

Layered Basis Pursuit (Noisy)

$$\boldsymbol{\Gamma}_1^{\mathrm{LBP}} = \min_{\boldsymbol{\Gamma}_1} \frac{1}{2}\|\mathbf{Y} - \mathbf{D}_1\boldsymbol{\Gamma}_1\|_2^2 + \lambda_1\|\boldsymbol{\Gamma}_1\|_1$$

$$\boldsymbol{\Gamma}_2^{\mathrm{LBP}} = \min_{\boldsymbol{\Gamma}_2} \frac{1}{2}\left\|\boldsymbol{\Gamma}_1^{\mathrm{LBP}} - \mathbf{D}_2\boldsymbol{\Gamma}_2\right\|_2^2 + \lambda_2\|\boldsymbol{\Gamma}_2\|_1$$

We can invoke a result we have seen already, referring to the BP for the CSC model:

RECALL

For $\mathbf{Y} = \mathbf{D}\boldsymbol{\Gamma} + \mathbf{E}$, if

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

then we are guaranteed that

$$\|\boldsymbol{\Delta}\|_{2,\infty}^{\mathrm{p}} \leq 7.5\,\varepsilon_{\mathrm{L}}^{0}\sqrt{\|\boldsymbol{\Gamma}\|_{0,\infty}^{\mathrm{p}}}$$

Stability of Layered BP

Theorem: Assuming that $\|\mathbf{\Gamma}_i\|_{0,\infty}^{s} < \frac{1}{3}\left(1+\frac{1}{\mu(\mathbf{D}_i)}\right)$ then For correctly chosen $\{\lambda_i\}_{i=1}^{K}$ we are guaranteed that

1. The support of $\mathbf{\Gamma}_i^{LBP}$ is contained in that of $\mathbf{\Gamma}_i$
2. The error is bounded: $\left\|\mathbf{\Gamma}_i^{LBP}-\mathbf{\Gamma}_i\right\|_{2,\infty}^{p} \le \varepsilon_L^i$, where

$$\varepsilon_L^i = 7.5^i\|\mathbf{E}\|_{2,\infty}^{p}\prod_{j=1}^{i}\sqrt{\|\mathbf{\Gamma}_j\|_{0,\infty}^{p}}$$

3. Every entry in $\mathbf{\Gamma}_i$ greater than $\varepsilon_L^i/\sqrt{\|\mathbf{\Gamma}_i\|_{0,\infty}^{p}}$ will be found

Layered Iterative Thresholding

Layered BP: $\boldsymbol{\Gamma}_j^{LBP} = \min_{\boldsymbol{\Gamma}_j} \frac{1}{2}\left\|\boldsymbol{\Gamma}_{j-1}^{LBP} - \mathbf{D}_j\boldsymbol{\Gamma}_j\right\|_2^2 + \xi_j\left\|\boldsymbol{\Gamma}_j\right\|_1$ j

Layered Iterative Soft-Thresholding:

t $\boldsymbol{\Gamma}_j^t = \mathcal{S}_{\xi_j/c_j}\left(\boldsymbol{\Gamma}_j^{t-1} + \frac{1}{c_j}\mathbf{D}_j^T\left(\widehat{\boldsymbol{\Gamma}}_{j-1} - \mathbf{D}_j\boldsymbol{\Gamma}_j^{t-1}\right)\right)$ j

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

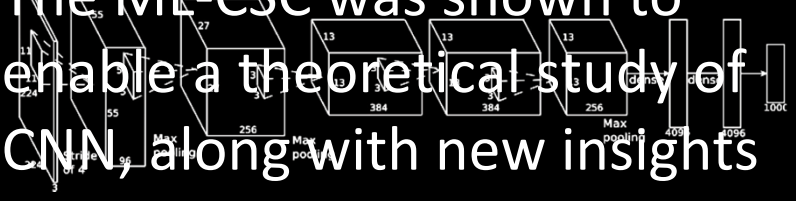
Can be seen as a recurrent neural network
[Gregor & LeCun '10]

* $c_i > 0.5\,\lambda_{max}(\mathbf{D}_i^T\mathbf{D}_i)$

Time to Conclude

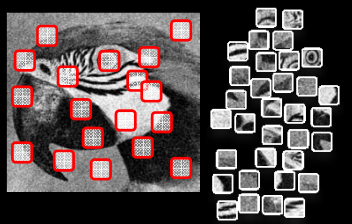
This Talk

The ML-CSC was shown to enable a theoretical study of CNN, along with new insights



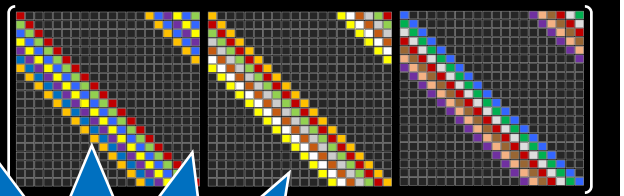
Independent patch-processing

Local Sparsity



Convolutional Neural Networks

Novel View of Convolutional Sparse Coding



Multi-Layer Convolutional Sparse Coding

The underlying idea:
Modeling the data source in order to be able to theoretically analyze algorithms' performance

Current/Future Work

In general, we aim to leverage our theoretical insights in order to get to practical implications

More specifically, we work on:

- Developing alternative (local) pursuit methods for the CSC and ML-CSC
- Could we propose an MMSE-driven pursuit
- Training the dictionaries – So far our efforts are focused on the unsupervised mode and the results are encouraging
- Explaining theoretically "known" tricks in CNN (local normalization, batch-normalization, the effect of stride, residual networks, dropout, …
- Better understanding this model by projecting true signals on to it to see what kind of sparsities and dictionaries are obtained
- Improving the corresponding performance bounds, and
- Tying all the above to applications
- …

These slides will be shared in my webpage in few days

Questions?