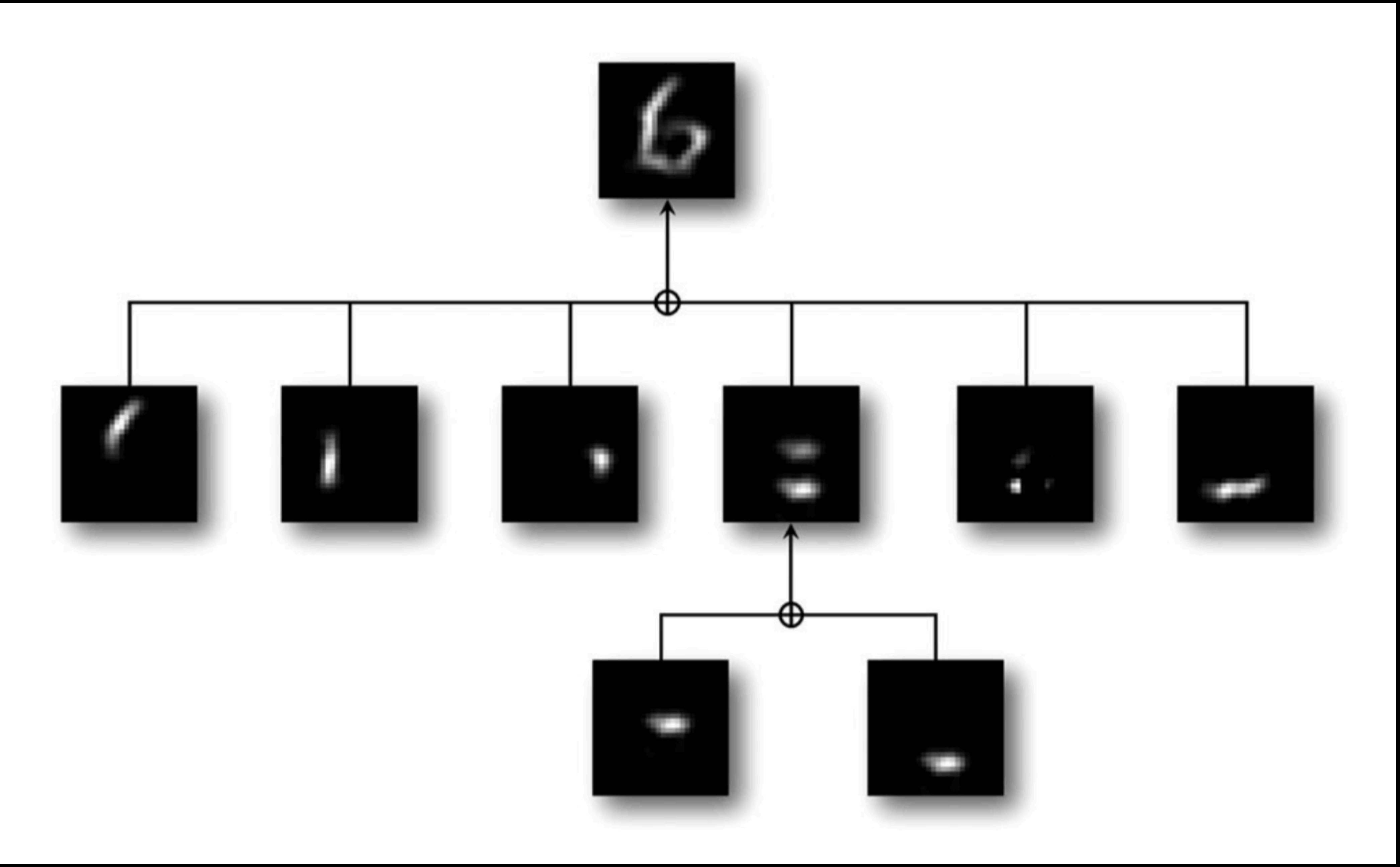
Tutorial 10: Sparse Representations

Digital Image Processing (236860)

Agenda

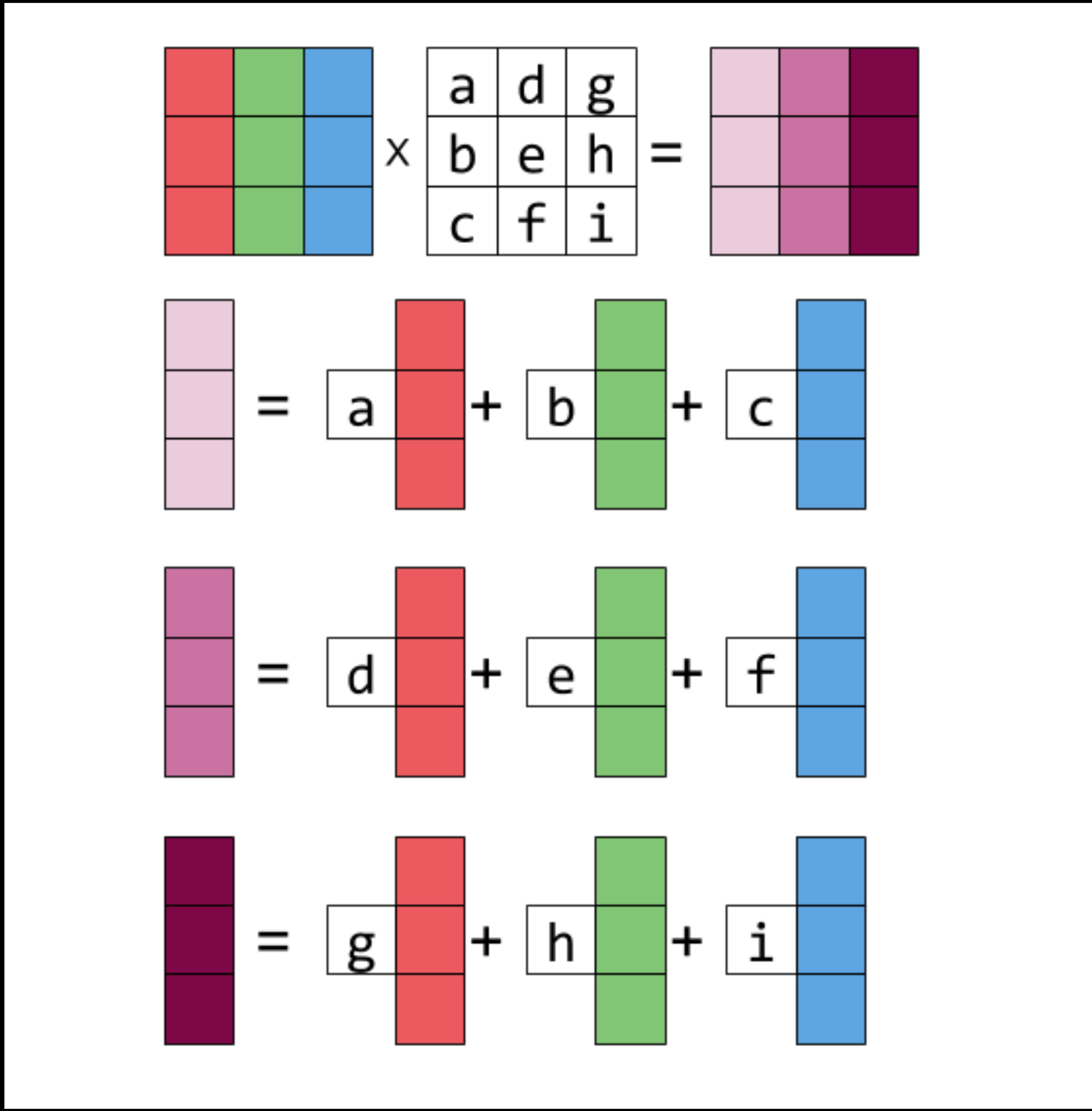
- Introduction
- Sparse coding: OMP, ISTA, IHT
- Dictionary learning
  - MOD
  - Double-sparsity
  - Application: Face inpainting
- Extensions: CSC, ML-CSC

Atoms of an image

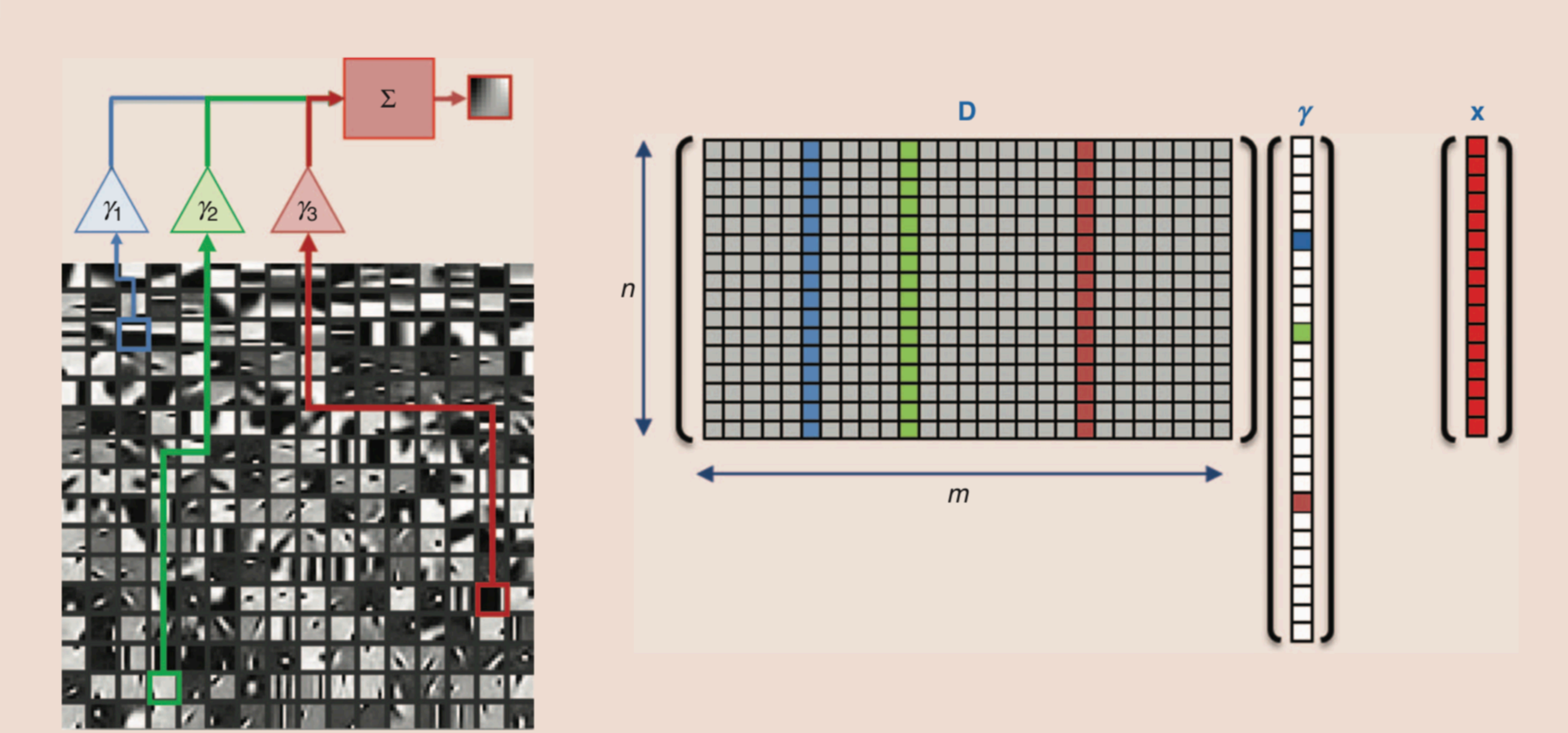


Sulam, Jeremias, et al. "Multilayer convolutional sparse modeling: Pursuit and dictionary learning." *IEEE Transactions on Signal Processing* 66.15 (2018): 4090-4104

Matrix-vector multiplication



Putting it all together



(a) (b)

Papyan, Vardan, et al. "Theoretical Foundations of Deep Learning via Sparse Representations."

Model

- $\boldsymbol{y} = \boldsymbol{D}\boldsymbol{x} + \boldsymbol{\eta}$
- Signal/observation: $\boldsymbol{y} \in \mathbb{R}^n$.
- Dictionary: $\boldsymbol{D} \in \mathbb{R}^{n \times m}$.
- Representation: $\boldsymbol{x} \in \mathbb{R}^m$.
- Noise: $\boldsymbol{\eta} \sim \mathcal{N}\left(0, \sigma_N^2\right)$.

The columns will act as our “atoms”

Should be sparse

Formal definition

- $\| \boldsymbol{x} \|_0 =$ number of non-zero elements in $\boldsymbol{x}$
- $\left(P_0^{\epsilon}\right):\quad \min_x \| \boldsymbol{x} \|_0$ s.t. $\left\| \boldsymbol{y} - \boldsymbol{D}\boldsymbol{x} \right\|_2 < \epsilon$
- $\left(P_0^{\lambda}\right):\quad \min_x \frac{1}{2} \left\| \boldsymbol{y} - \boldsymbol{D}\boldsymbol{x} \right\|_2^2 + \lambda \| \boldsymbol{x} \|_0$
- NP-hard!

Trivial cases

- Imagine we know $\| \boldsymbol{x} \|_0 = 1$. Can we determine $\boldsymbol{x}$?
- Use the atom $\boldsymbol{a}_i$ that maximises $\boldsymbol{a}_i^T \boldsymbol{y}$.
- Imagine we know the support is S. Can we determine $\boldsymbol{x}$?
- Solve $\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \|_2^2$ s.t. $\sup\{\boldsymbol{x}\} = S$ using LS.

Greedy scheme

OMP

Guaranteed to recover the support perfectly if x is sparse enough.

Our goal: Approximating the solution of

$$\min_{\underline{x}} \|\underline{x}\|_0^0 \quad \text{s.t.} \quad \mathbf{A}\underline{x} = \underline{b}$$

Initialization

$k = 0,\ \underline{x}_0 = \underline{0}$

$\underline{r}_0 = \underline{b} - \mathbf{A}\underline{x}_0 = \underline{b}$

and $S_0 = \{\ \}$

$k = k + 1$

Main Iteration

1. Compute $p(i) = \left|\underline{a}_i^T \underline{r}_{k-1}\right|$ for $1 \le i \le m$
2. Choose i_0 s.t. $\forall 1 \le i \le m,\ p(i_0) \ge p(i)$
3. Update S_k : $S_k = S_{k-1} \cup \{i_0\}$
4. LS : $\underline{x}_k = \min_{\underline{x}} \|\mathbf{A}\underline{x} - \underline{b}\|_2^2$ s.t. $\sup\{\underline{x}\} = S_k$
5. Update Residual: $\underline{r}_k = \underline{b} - \mathbf{A}\underline{x}_k$

$\|\underline{r}_k\|_2 \le \varepsilon$

No

Yes

Stop

ℓ_1 relaxation

- $\left(P_1^\lambda\right): \quad \min_x \frac{1}{2} \| \boldsymbol{y} - \boldsymbol{D}\boldsymbol{x} \|_2^2 + \lambda \| \boldsymbol{x} \|_1$
- ℓ_1 is convex (and differentiable except when $x_i = 0$).
- Can be solved using convex optimisation tools!

Proximal gradient

convex and differentiable

$$\min_{\mathbf{x}} g(\mathbf{x}) + h(\mathbf{x})$$

convex

Proximal gradient

- Initialize $\mathbf{x}_0$
- For $k = 0, 1, \ldots,$ until convergence

project to minimise h

$$\mathbf{x}_{k+1} = \mathcal{P}_h\left(\mathbf{x}_k - \eta_k \nabla g(\mathbf{x}_k)\right)$$

gradient descent to minimize g

Proximal operator = generalized projection

ISTA

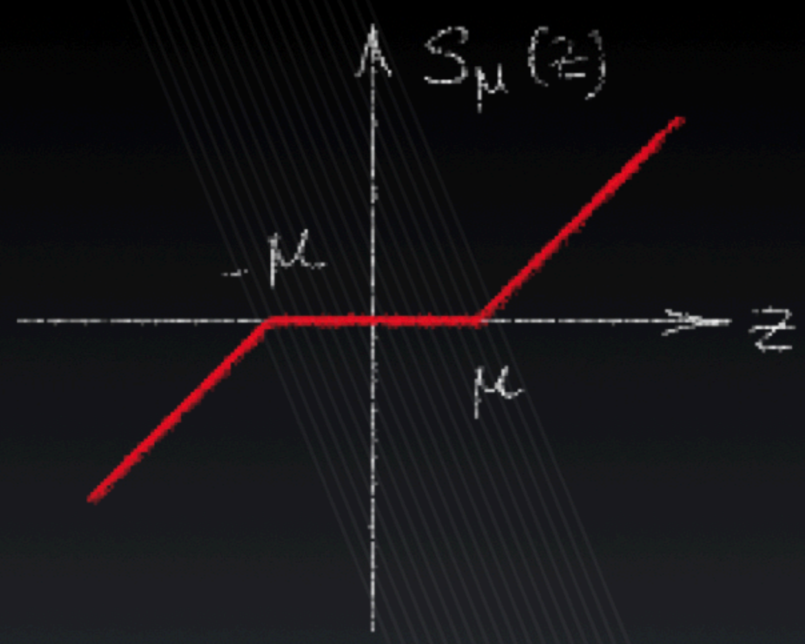
$$\min_{\mathbf{c}} \frac{1}{2}\|\boldsymbol{\Phi}\mathbf{c}-\mathbf{y}\|_2^2 + \lambda\|\mathbf{c}\|_1$$

Iterated soft thresholding (ISTA)

- Start with initial $\mathbf{c}^0$
- For $k = 0,1,\ldots,$ until convergence

$$\mathbf{c}^{k+1} = \mathcal{S}_{\eta\lambda}\left(\mathbf{c} - \eta\boldsymbol{\Phi}^*\left(\boldsymbol{\Phi}\mathbf{c}^k - \mathbf{y}\right)\right)$$

Both priors are widely used



IHT

$$\min_{\mathbf{c}} \frac{1}{2} \|\mathbf{\Phi c} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{c}\|_0$$

Iterated hard thresholding (IHT)

- Start with initial $\mathbf{c}^0$
- For $k = 0,1, \ldots,$ until convergence

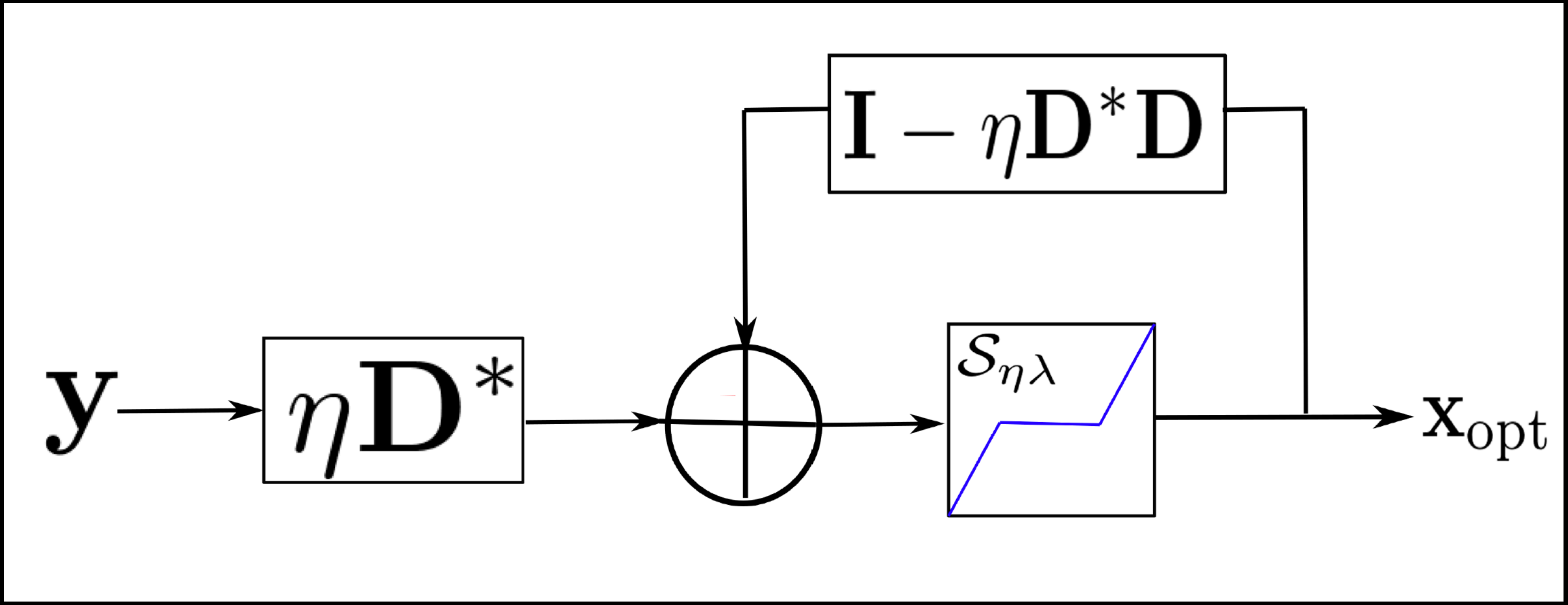
$$\mathbf{c}^{k+1} = \mathcal{H}_{\eta\sqrt{2\lambda}}\left(\mathbf{c} - \eta\mathbf{\Phi}^*\left(\mathbf{\Phi c}^k - \mathbf{y}\right)\right)$$

$$\mathcal{P}_{\lambda h}(c) = \begin{cases} c : |c| > \sqrt{2\lambda} \\ 0 : |c| < \sqrt{2\lambda} \end{cases}$$

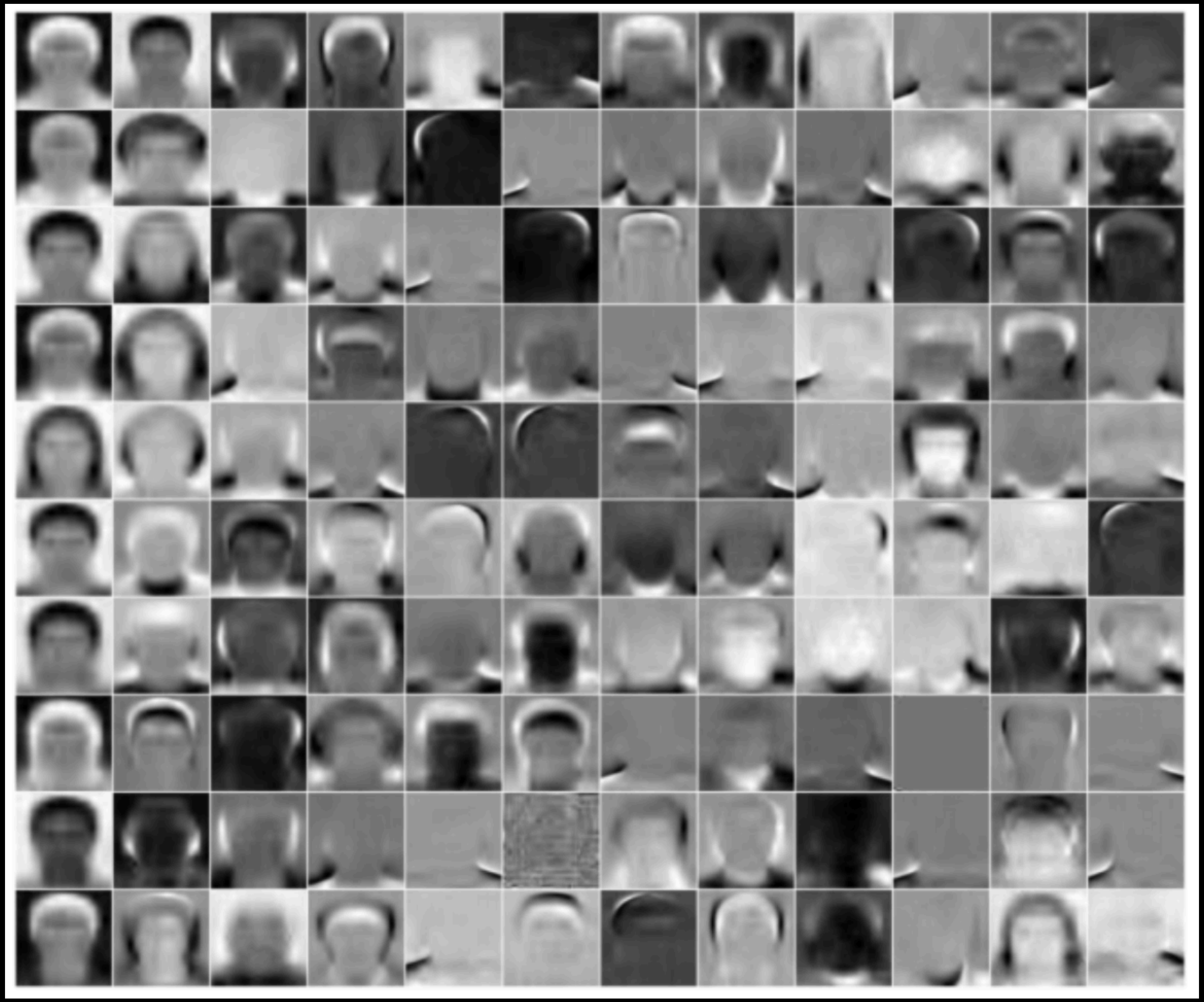
-

ISTA

- $\boldsymbol{x}^{k+1} = \mathcal{S}_{\eta\lambda}\left(\boldsymbol{x}^{k} - \eta \boldsymbol{D}^{*}\left(\boldsymbol{D}\boldsymbol{x}^{k} - \boldsymbol{y}\right)\right)$



Dictionary learning



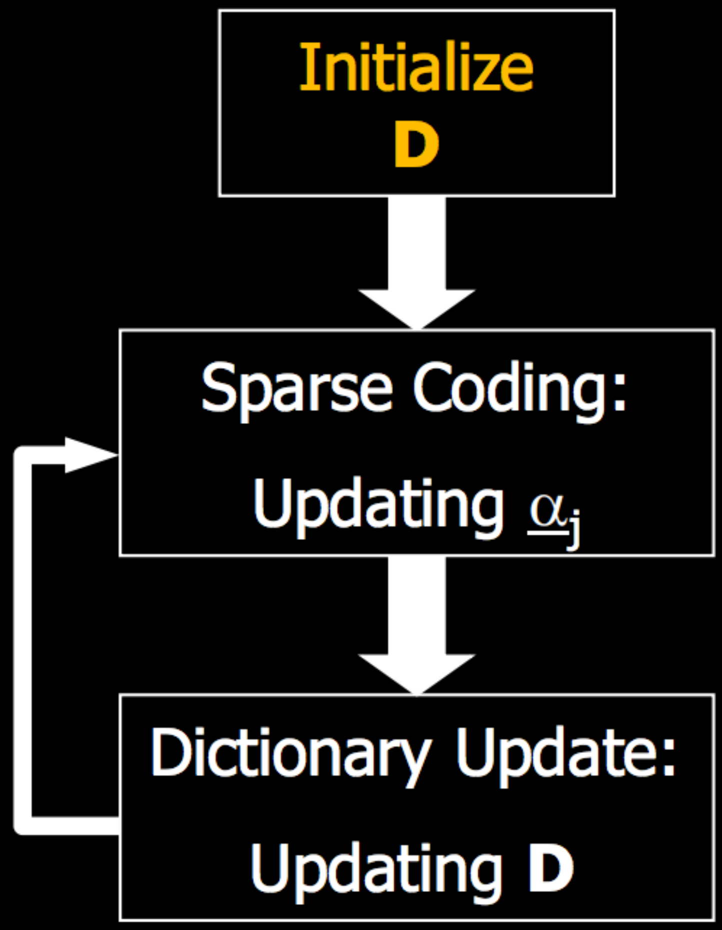
Sulam, Jeremias, et al. "Trainlets: Dictionary learning in high dimensions." *IEEE Transactions on Signal Processing* 64.12 (2016): 3180-3193

Dictionary learning scheme

- Input:
  - A matrix of column signals $\boldsymbol{Y}$
  - An initial dictionary $\boldsymbol{D}$
- Iterate between:
  - Fix $\boldsymbol{D}$. Find sparse representations $\boldsymbol{X}$ given $\boldsymbol{D}$ and $\boldsymbol{Y}$.
  - Fix $\boldsymbol{X}$. Optimize $\boldsymbol{D}$ such that $\| \boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X} \|_F^2$ is minimal.

MOD

$$\min_{\mathbf{D},\{\underline{\alpha}_j\}} \sum_{j=1}^{N} \left\| \underline{y}_j - \mathbf{D}\underline{\alpha}_j \right\|_2^2 \quad \text{s.t.} \quad \forall j, \quad \left\| \underline{\alpha}_j \right\|_0 \le k_0$$

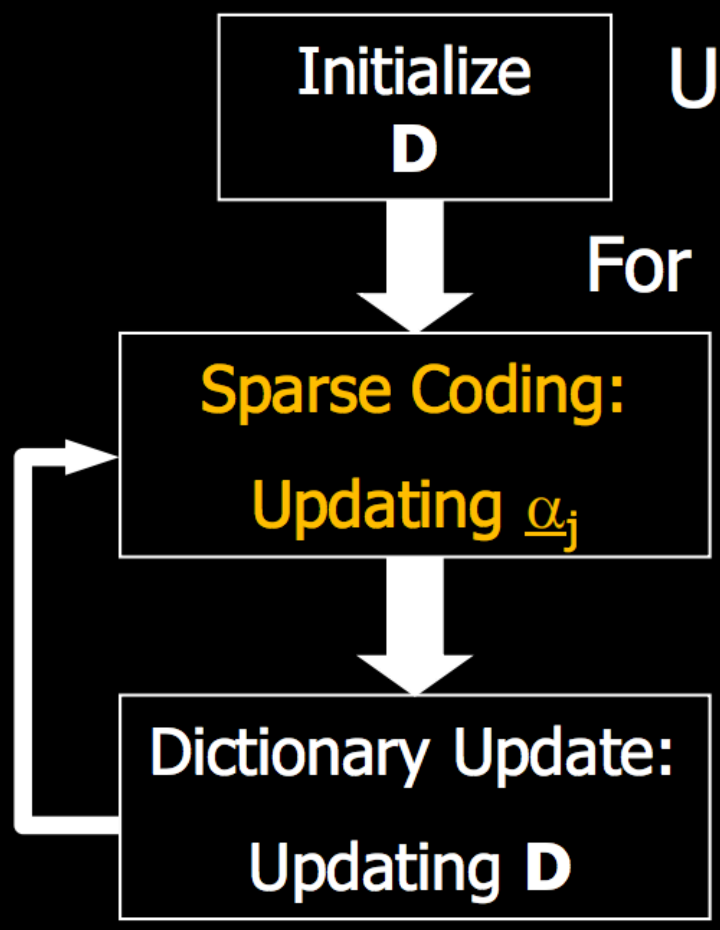


Initialize **D**:

- By choosing a predefined dictionary
- Choosing m random elements of the training set

MOD

$$\min_{\mathbf{D},\{\underline{\alpha}_j\}} \sum_{j=1}^{N} \left\| \underline{y}_j - \mathbf{D}\underline{\alpha}_j \right\|_2^2 \quad \text{s.t.} \quad \forall j, \quad \left\| \underline{\alpha}_j \right\|_0 \le k_0$$



Update $\underline{\alpha}_j$ (**D** is fixed):

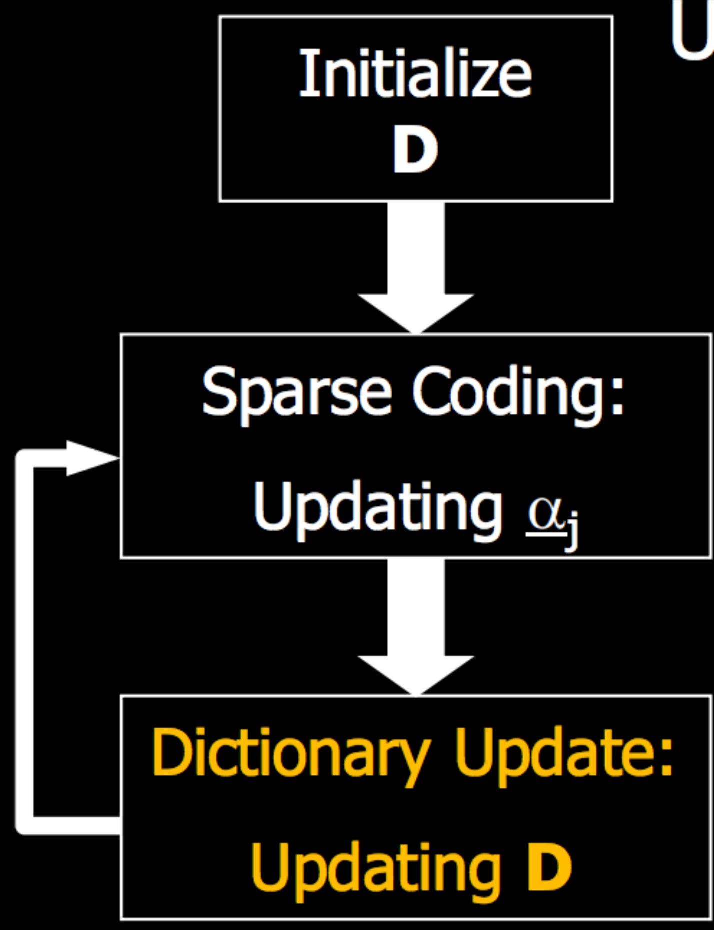
$$\text{For } 1 \le j \le N \quad \min_{\underline{\alpha}_j} \left\| \underline{y}_j - \mathbf{D}\underline{\alpha}_j \right\|_2^2$$

$$\text{s.t.} \quad \left\| \underline{\alpha}_j \right\|_0 \le k_0$$

Apply your favorite pursuit (OMP, BP, …)

MOD

$$\min_{\mathbf{D},\{\underline{\alpha}_j\}} \sum_{j=1}^{N} \left\| \underline{y}_j - \mathbf{D}\underline{\alpha}_j \right\|_2^2 \quad \text{s.t.} \quad \forall j, \ \left\| \underline{\alpha}_j \right\|_0 \le k_0$$



Update **D** ($\underline{\alpha}_j$ are fixed):

$$\min_{\mathbf{D}} \left\| \underline{Y} - \mathbf{DA} \right\|_F^2$$

$$\nabla_{\mathbf{D}} \left\| \underline{Y} - \mathbf{DA} \right\|_F^2 = \left(\underline{Y} - \mathbf{DA} \right) \mathbf{A}^T = 0$$

$$\Rightarrow \quad \mathbf{D} = \underline{Y}\mathbf{A}^T \left(\mathbf{AA}^T \right)^{-1}$$

Do not forget to normalize the atoms once finished

MOD

$$\min_{\mathbf{D},\{\underline{\alpha}_j\}} \sum_{j=1}^{N} \left\| \underline{y}_j - \mathbf{D}\underline{\alpha}_j \right\|_2^2 \quad \text{s.t.} \quad \forall j, \quad \left\| \underline{\alpha}_j \right\|_0 \le k_0$$

Initialize **D**

↓

Sparse Coding: Updating $\underline{\alpha}_j$

↓

Dictionary Update: Updating **D**

Stopping Condition:

$$\left\| \underline{Y} - \mathbf{DA} \right\|_2^2 \le N\varepsilon^2$$

Because then the set of representations and dictionary match the model assumptions

Implementation tricks

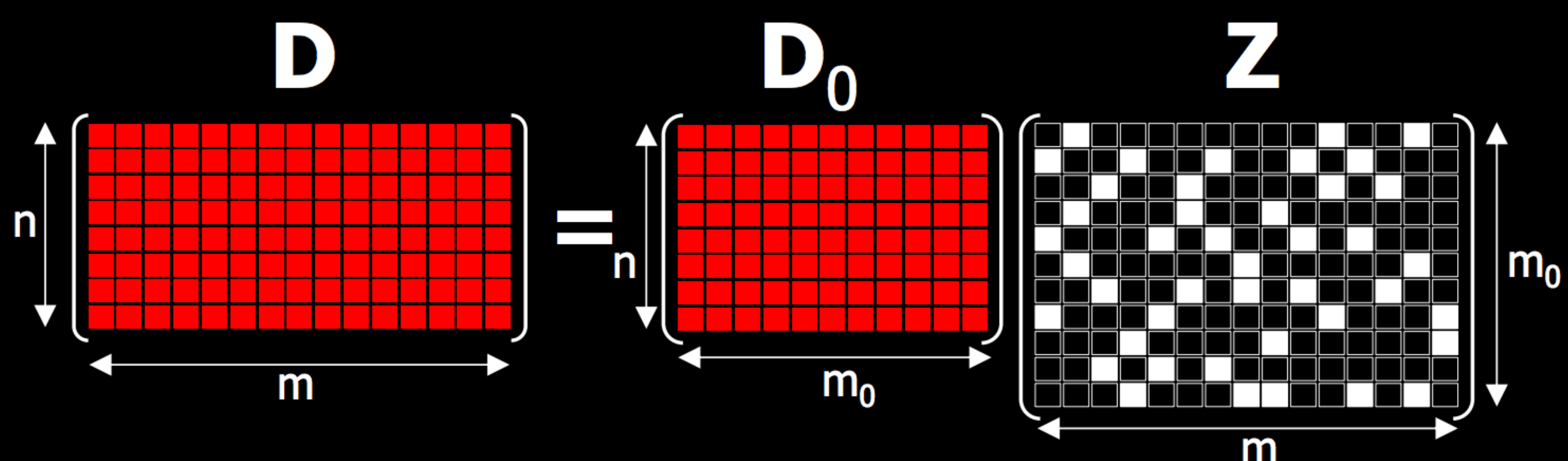
- After each dictionary update stage:
  - If two atoms are too similar, discard one of them.
  - If an atom in the dictionary is rarely used, discard it.
  - Replace discarded atoms by the most ill-represented signal examples.

Difficulties of dictionary learning

- Slow (to train and to use)
- Cannot be used in high-dimensions
- Why?
  - Too many degrees of freedom
  - Multiplying by $\boldsymbol{D}$ is costly
- Solution?
  - Force $\boldsymbol{D}$ to have a certain structure.

Double-sparsity

- Assume that the dictionary to be found **D** can be written as $\mathbf{D}=\mathbf{D}_0\mathbf{Z}$, where
  - $\mathbf{D}_0$ is a fixed base-dictionary with a fast deployment, and
  - **Z** is a very sparse matrix, having $k_1 \ll n$ non-zeros in each of its columns



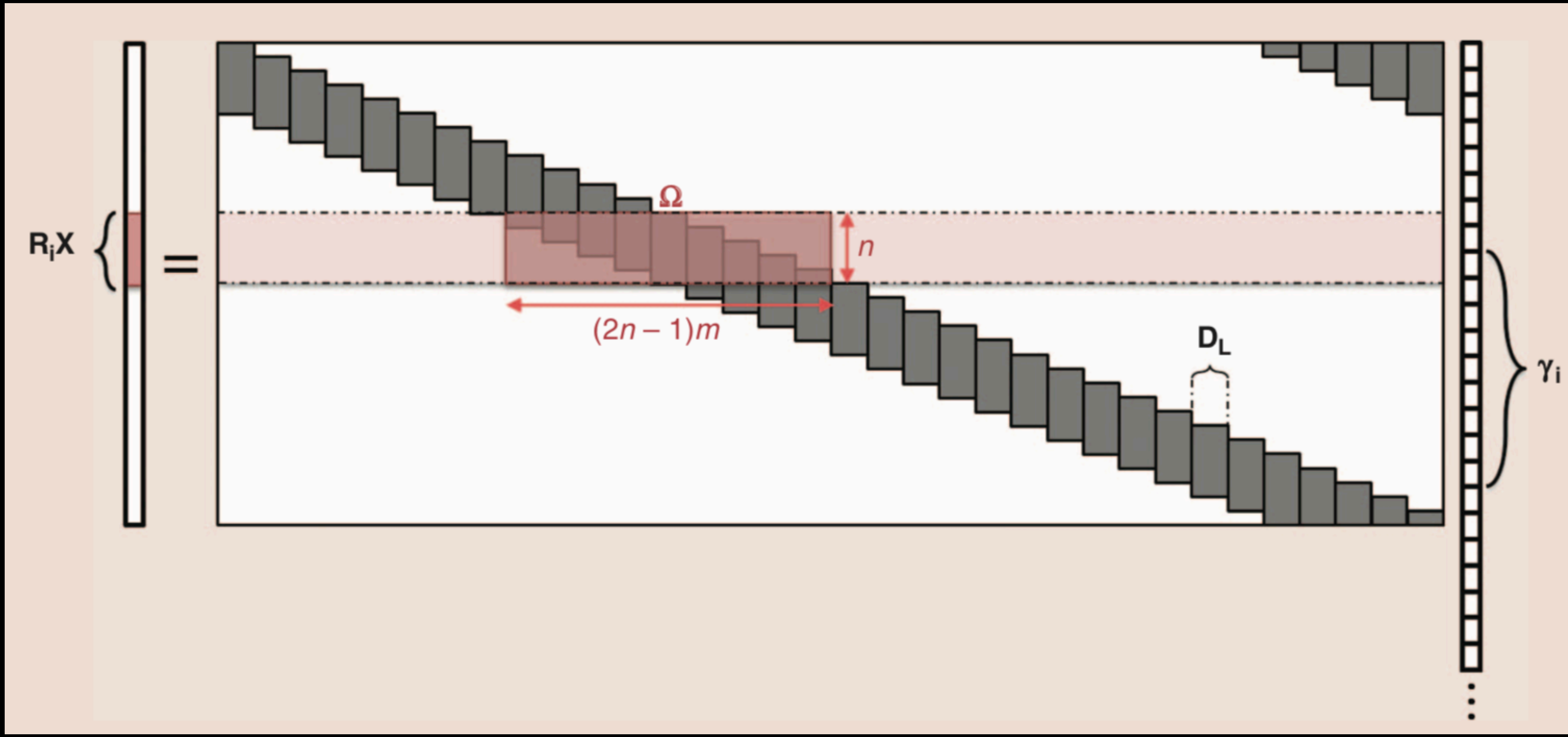
Example: Face inpainting

- Train a dictionary using exemplar face images.
- Solve $\min_{x} \| \boldsymbol{y} - \boldsymbol{MDx} \|_2 + \lambda \| \boldsymbol{x} \|_1$ where $\boldsymbol{M}$ is a mask that zeros all missing pixels.
- Obtain the full image via $\boldsymbol{Dx}$.

Sulam, Jeremias, and Michael Elad. "Large inpainting of face images with trainlets." *IEEE signal processing letters* 23.12 (2016): 1839-1843

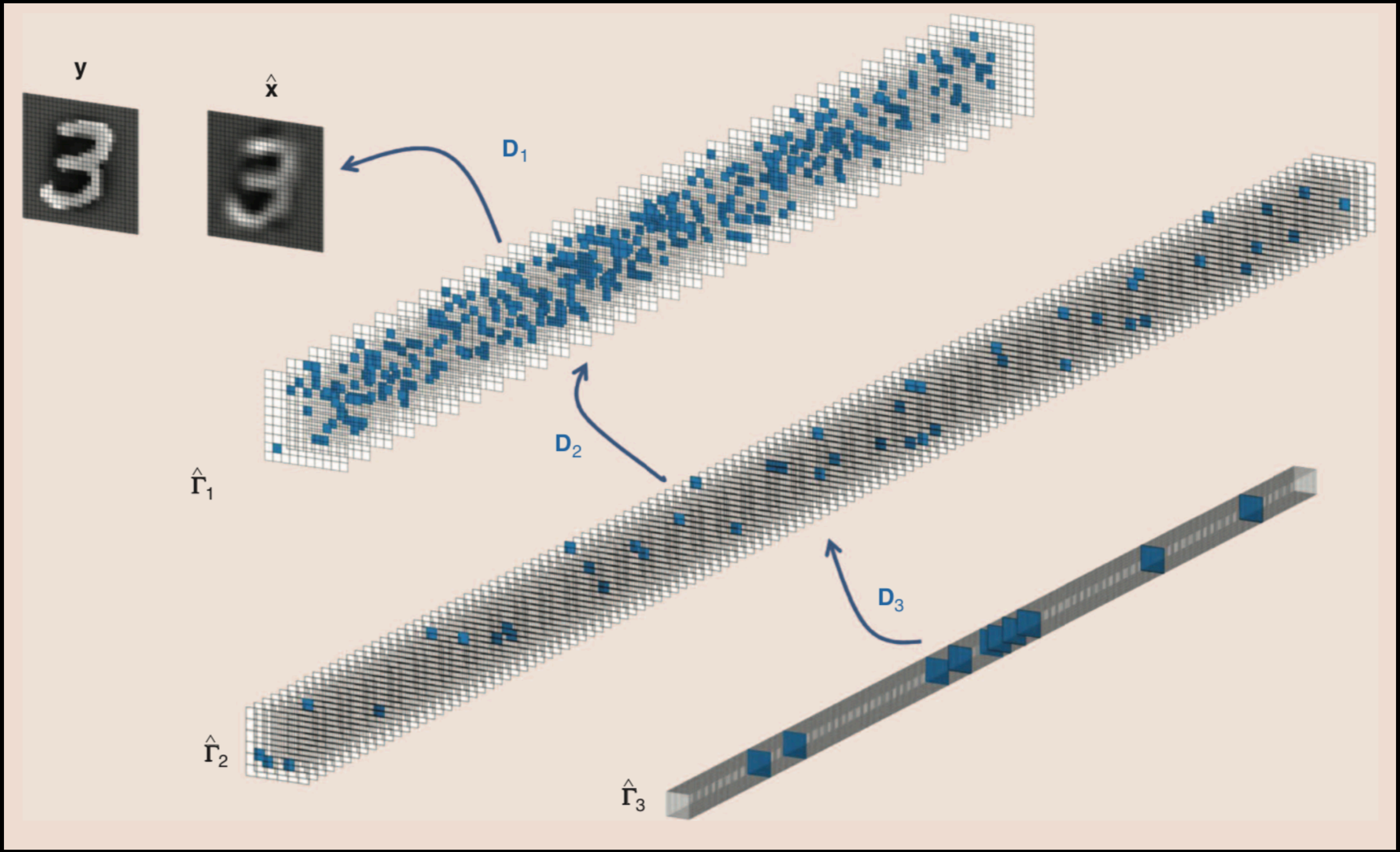
Convolutional sparse coding (CSC)

$$\mathbf{X} = \sum_{i=1}^{m} \mathbf{C}_i \mathbf{\Gamma}_i = [\mathbf{C}_1 \ \mathbf{C}_2 \ \cdots \ \mathbf{C}_m] \begin{bmatrix} \mathbf{\Gamma}_1 \\ \mathbf{\Gamma}_2 \\ \vdots \\ \mathbf{\Gamma}_m \end{bmatrix} = \mathbf{D}\mathbf{\Gamma}.$$



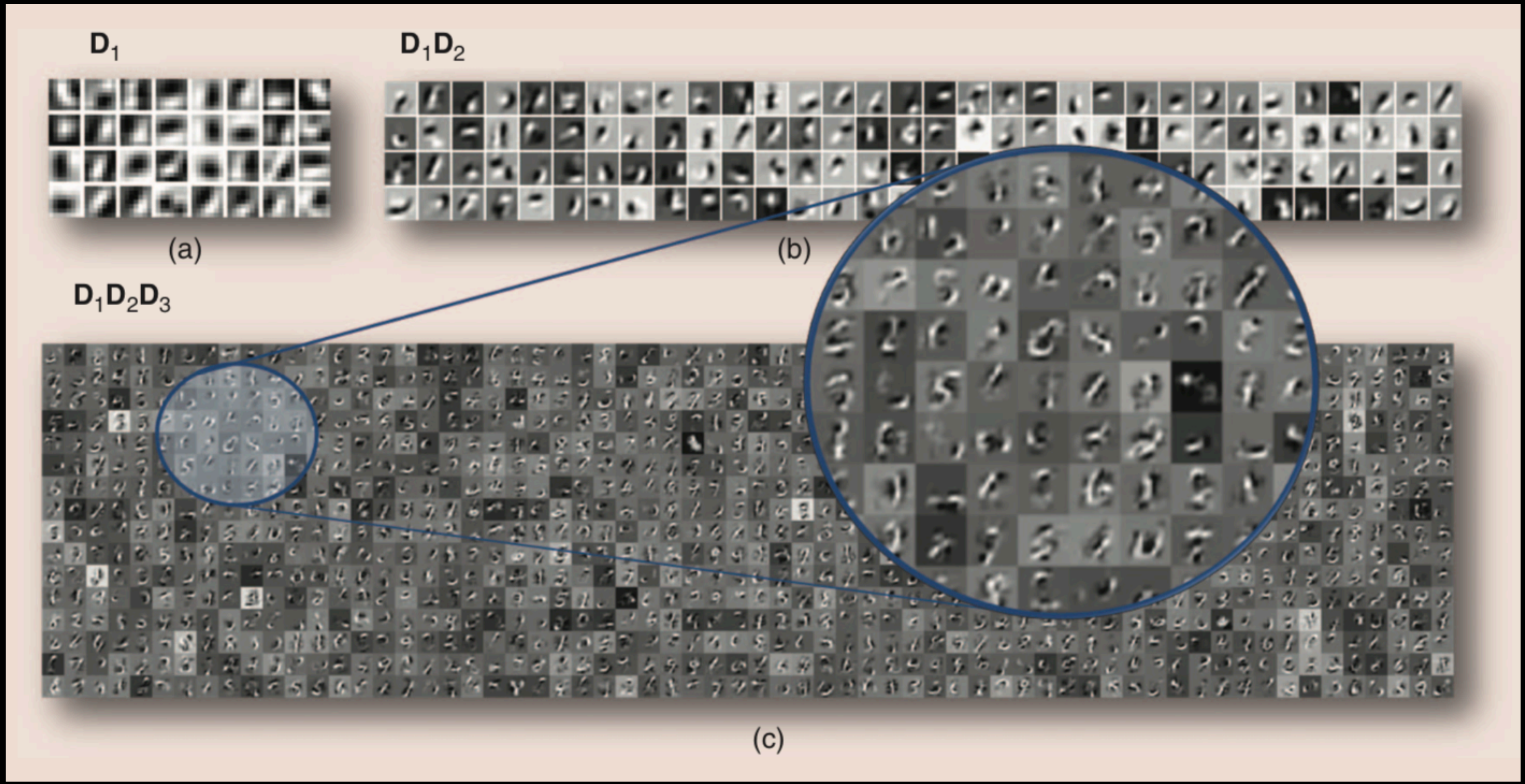
Papyan, Vardan, et al. "Theoretical Foundations of Deep Learning via Sparse Representations."

Multi-layered CSC (ML-CSC)



Papyan, Vardan, et al. "Theoretical Foundations of Deep Learning via Sparse Representations."

Multi-layered CSC (ML-CSC)



Papyan, Vardan, et al. "Theoretical Foundations of Deep Learning via Sparse Representations."