Image Noise: Detection, Measurement and Removal Techniques

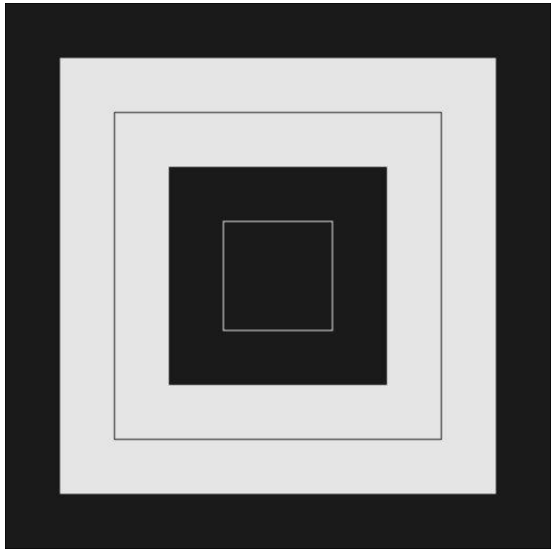
Zhifei Zhang

Outline

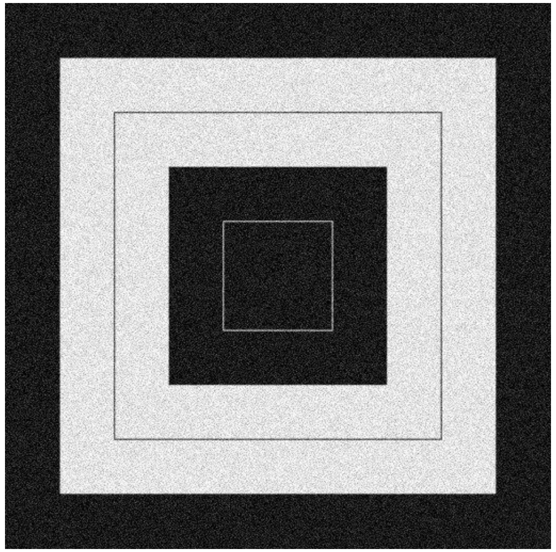
- ❖ Noise measurement
  - Filter-based
  - Block-based
  - Wavelet-based
- ❖ Noise removal
  - Spatial domain
  - Transform domain
  - Non-local methods
  - TV, SC, DL

Noise Measurement --- Noise Model

Additive White Gaussian Noise (AWGN)



Without noise

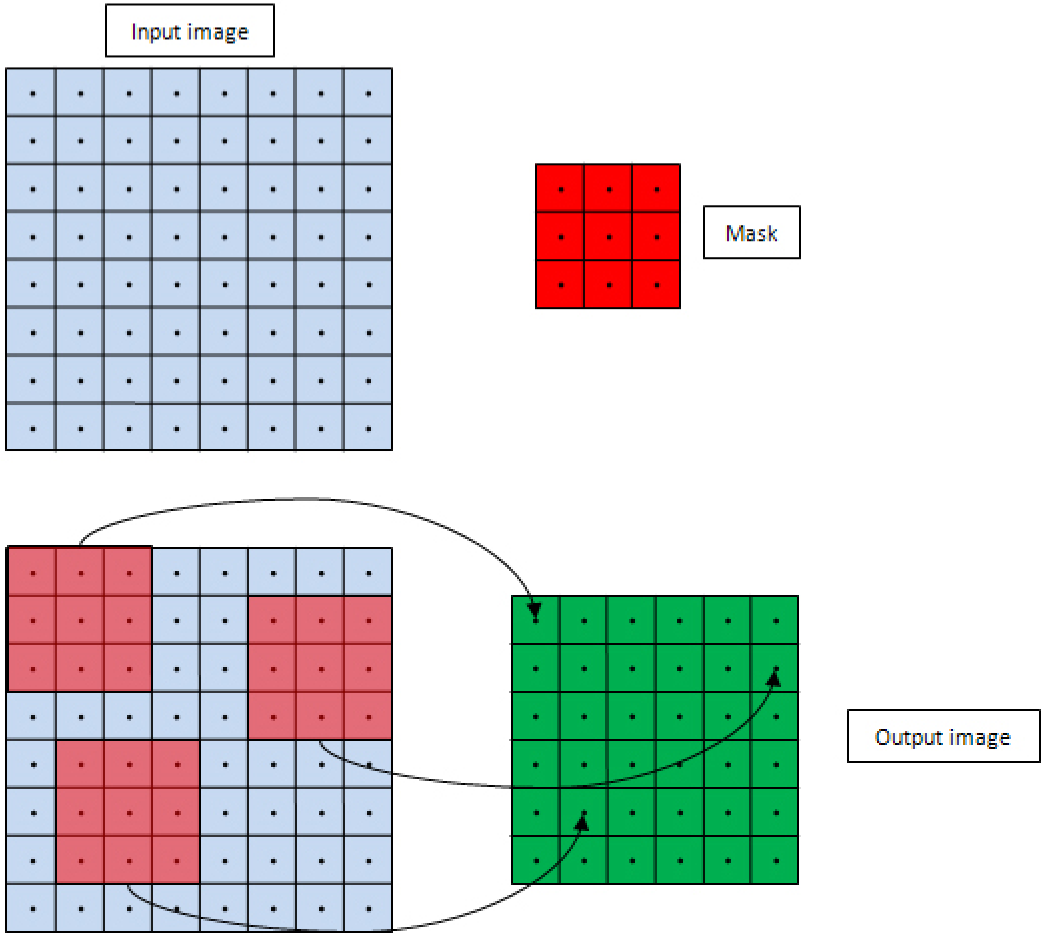


With Gaussian noise

$$\underbrace{v_{xy}}_{\text{Observation}} = \underbrace{u_{xy}}_{\text{True value}} + \underbrace{n_{xy}}_{\text{White noise}}$$

$$n_{xy} \sim \mathcal{N}(0, \sigma^2)$$

Noise Measurement --- Filter-based



❖ **Low-pass filter:**

$$n_{xy} = v_{xy} - k_l * v_{xy}$$

k_l **can be average, median, Gaussian, etc.**

❖ **High-pass filter:**

$$n_{xy} = k_h * v_{xy}$$ **(Faster)**

k_h **can be Laplacian, gradient, etc.**

Noise Measurement --- Filter-based

Limitation: See no difference between details and noise.

Remedy: Edge detector + Laplacian filter [Tai, et al. 2008]

Edge detection → Laplacian Convolution → Averaging

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\sigma = \sqrt{\frac{\pi}{2}} \frac{1}{6(W-2)(H-2)} \sum_{image\ I} |I(x,y) * N|$$

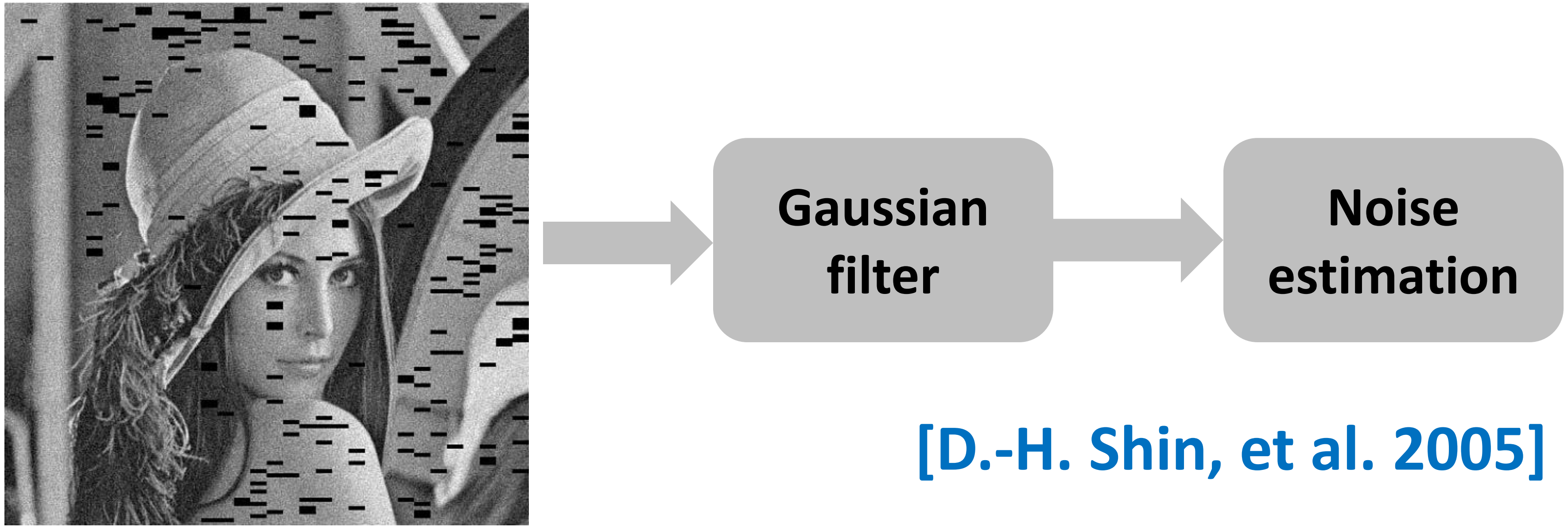
$$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$N = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

Noise Measurement --- Filter-based

Another limitation: The filtered result is assumed to be the noise, which is not always true especially for images with complex structure or fine details.

Remedy: Measure the noise only on those smooth blocks.



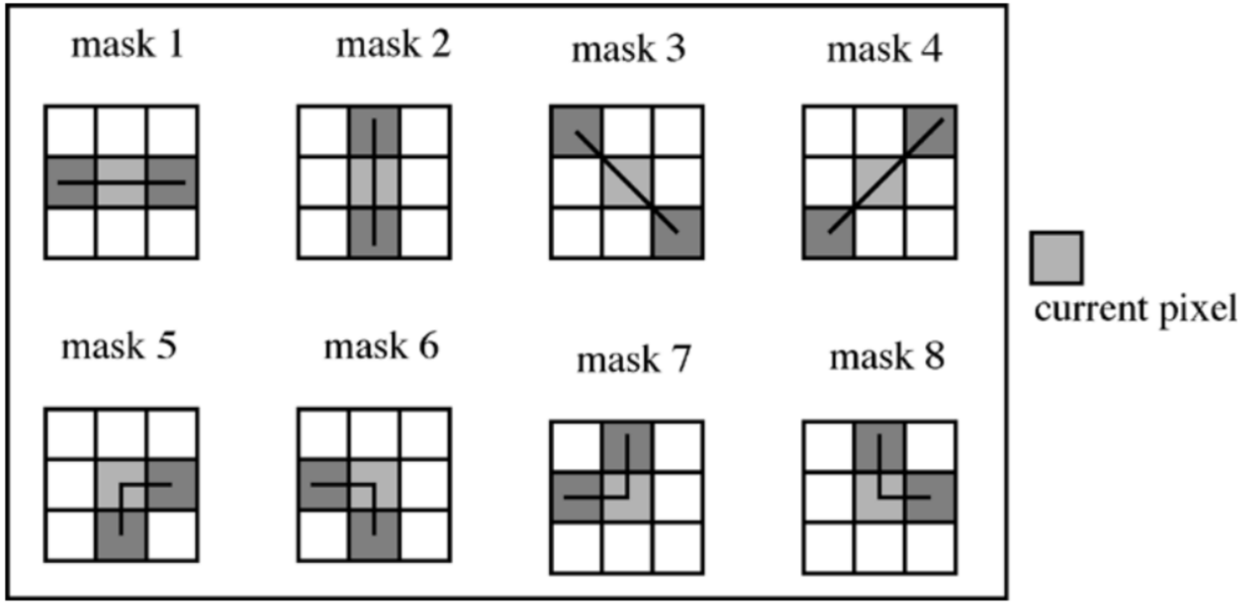
[D.-H. Shin, et al. 2005]

Noise Measurement --- Block-based

How to determine the homogeneous blocks ?
A homogeneous block is a block with uniform intensities.

- ❖ **Compute local variation**

- ❖ **Analyze local structure** [Amer, et al. 2005]

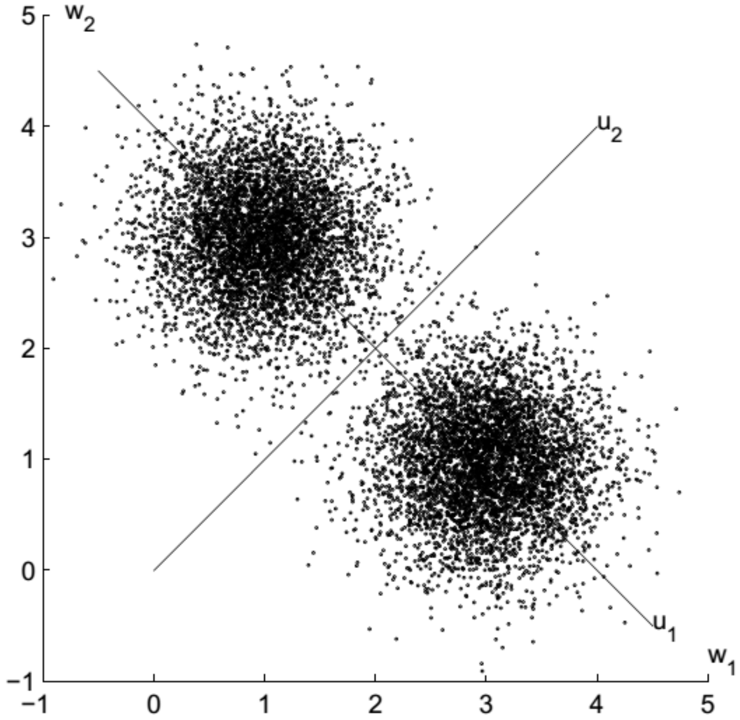


Noise Measurement --- Block-based

Limitation: Assume the existence of homogeneous blocks.

Remedy: Principle component analysis [Pyatykh, et al. 2013]

The noise variance can be estimated as the smallest eigenvalue of the image block covariance matrix.

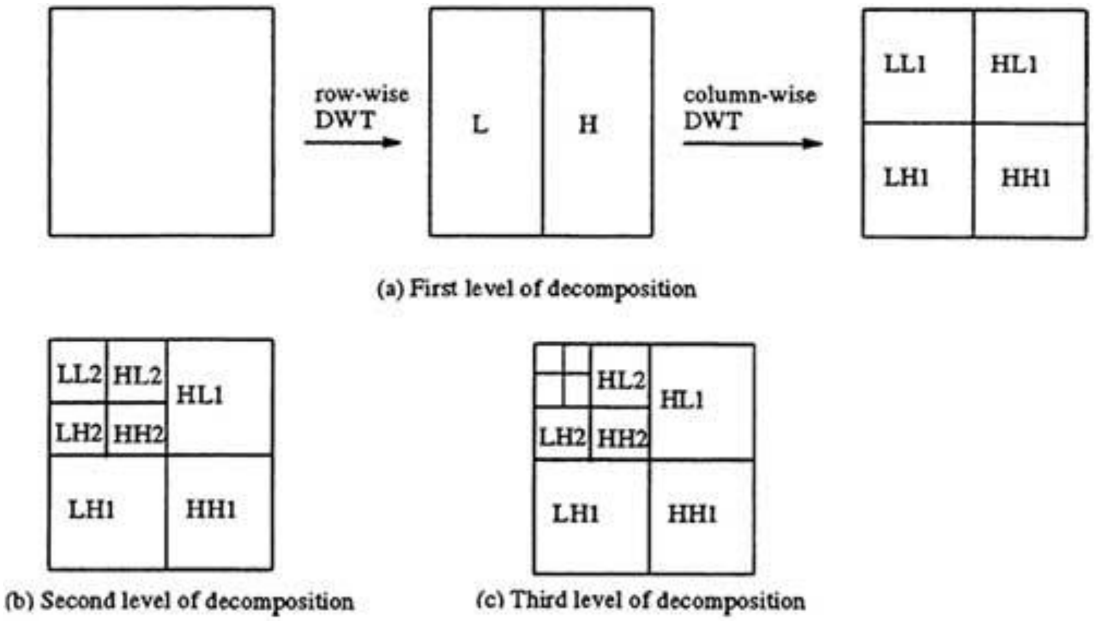


Toy example:

1D signal $\{x_k\} = \{1, 3, 1, 3, \ldots\}$

Noise $n_k \sim N(0, 0.5^2)$

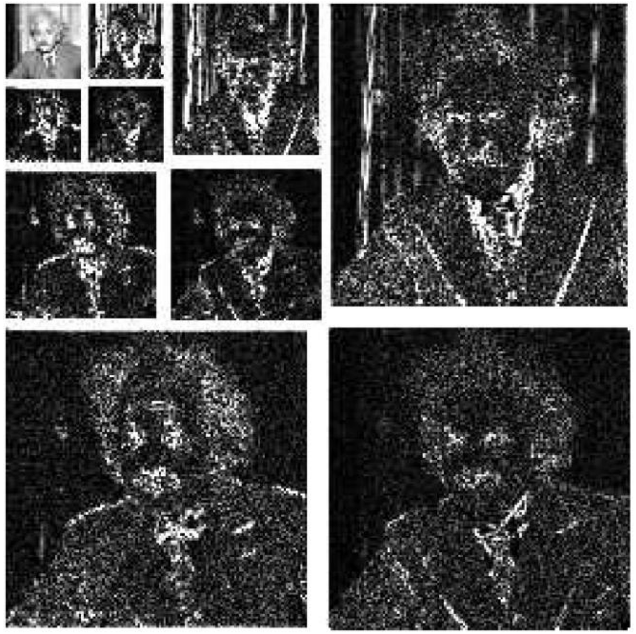
Noise Measurement --- Wavelet-based

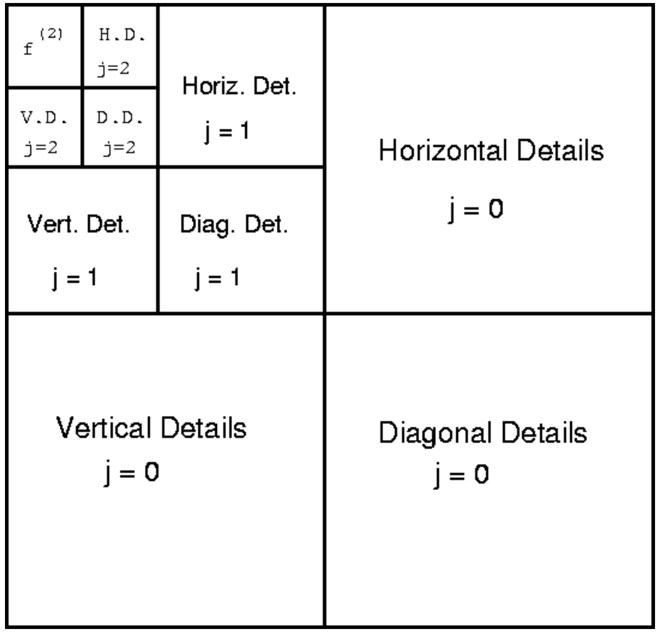


(a) First level of decomposition

(b) Second level of decomposition

(c) Third level of decomposition





Median Absolute Deviation (MAD) $= \dfrac{median\{|c_i|\}}{constant}$

[D. L. Donoho, 1995]

Noise Measurement --- Wavelet-based

Limitation: MAD assumes HH1 associates only to the noise, and it tends to overestimate the noise standard deviation under high SNR.

Remedy:

Adaptive thresholding

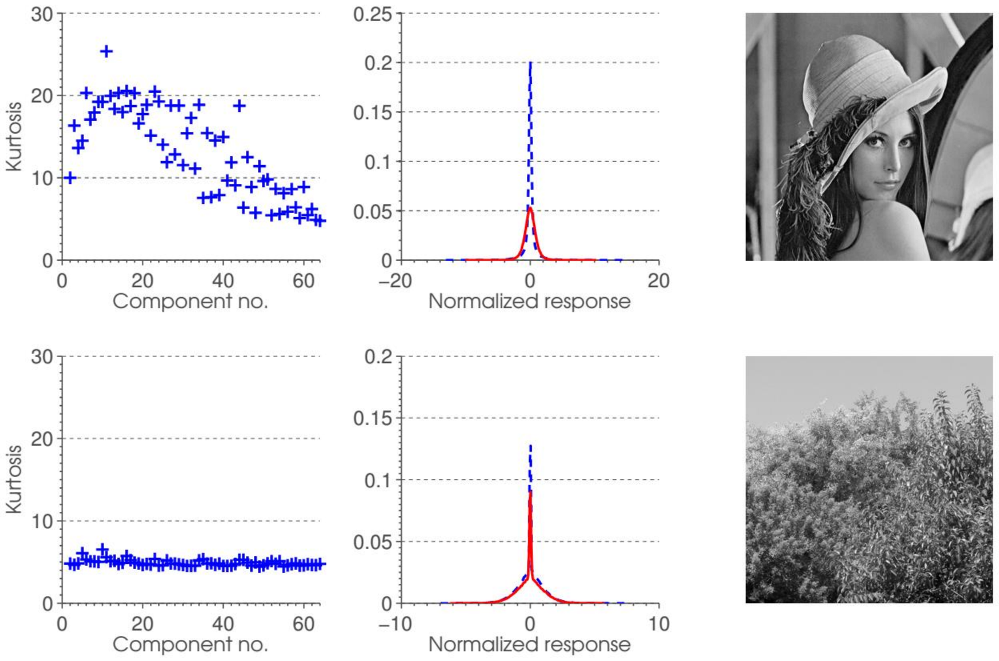
- SureShrink [Donoho, et al. 1995]
- BayesShrink [Simoncelli, et al. 1996]

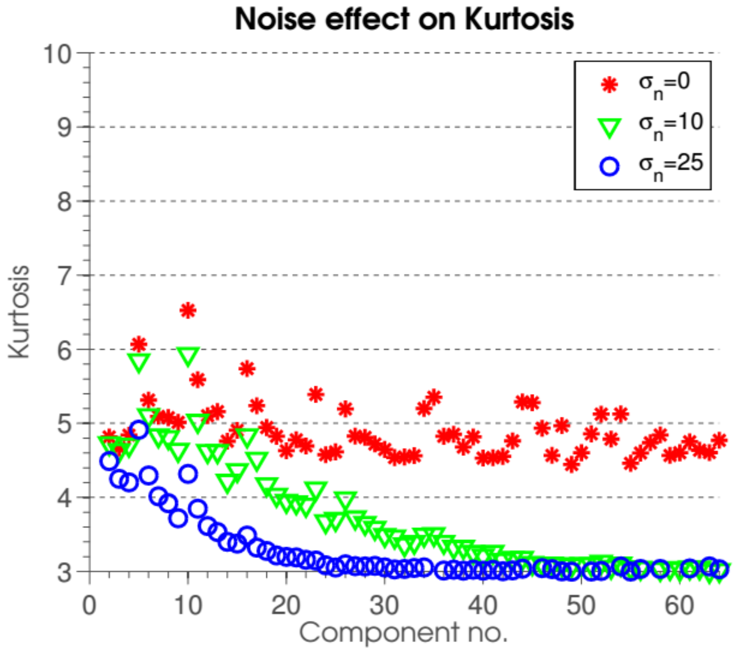
Non-local (BM3D) [Danielyan, et al. 2009]

Noise Measurement --- Wavelet-based

❖ **Statistical model** [Zoran, et al. 2009]

kurtosis of marginal distributions in clean natural images is constant throughout scales

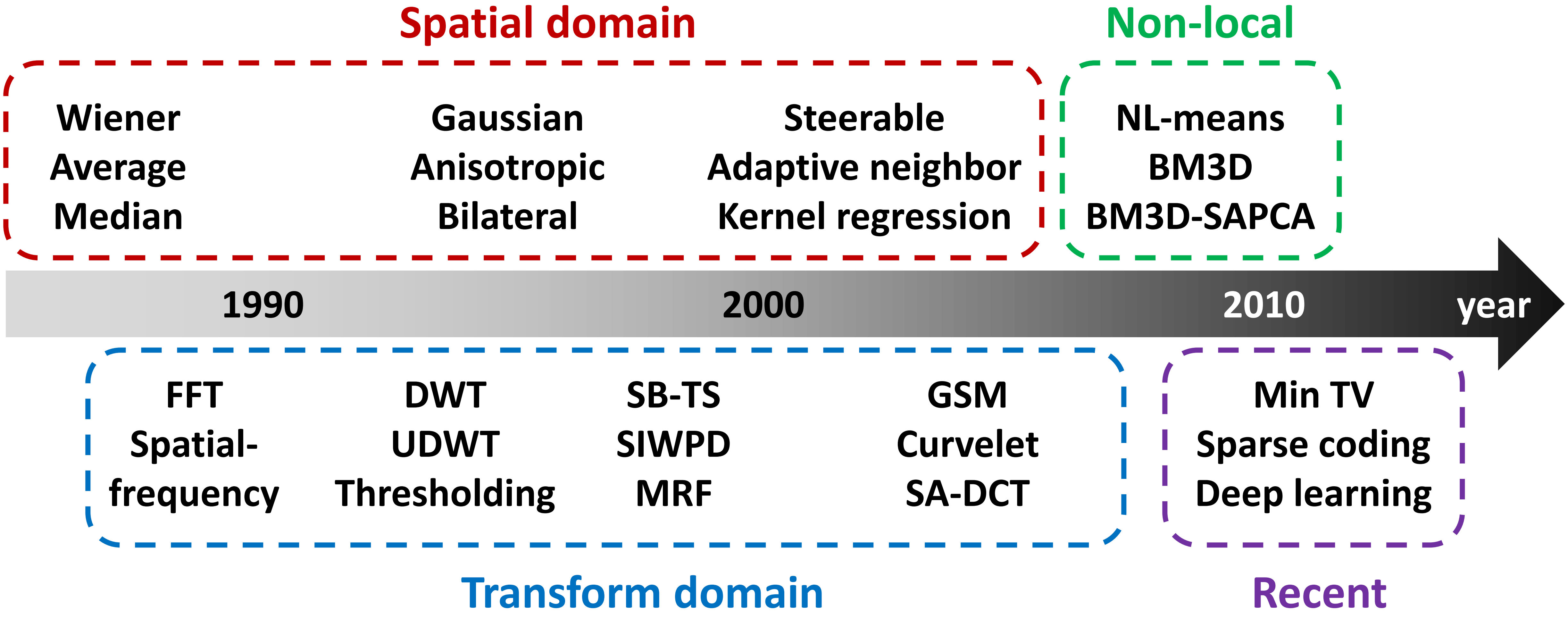




Outline

- Noise measurement
  - Filter-based
  - Block-based
  - Wavelet-based
- Noise removal
  - Spatial domain
  - Transform domain
  - Non-local methods
  - TV, SC, DL

Noise Removal --- Evolution



Noise Removal --- Spatial Domain

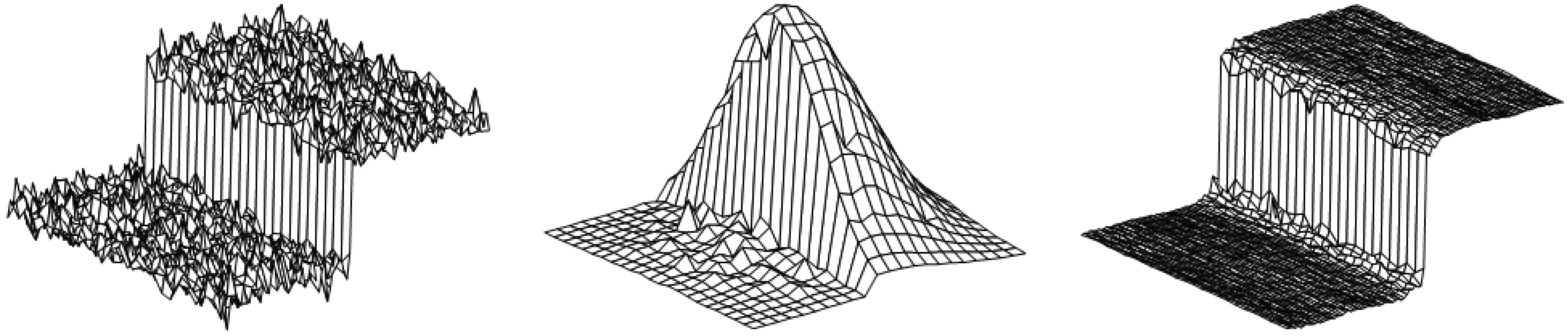
Design of the kernel (sliding window)

Non-linear

- **Isotropic (Blur edges)**
  - Median
  - Average
  - Gaussian
- **Anisotropic (Preserve edges)**
  - Bilateral
  - Steerable
  - Kernel regression

Noise Removal --- Spatial Domain

Bilateral filter [Tomasi, et al. 1998]



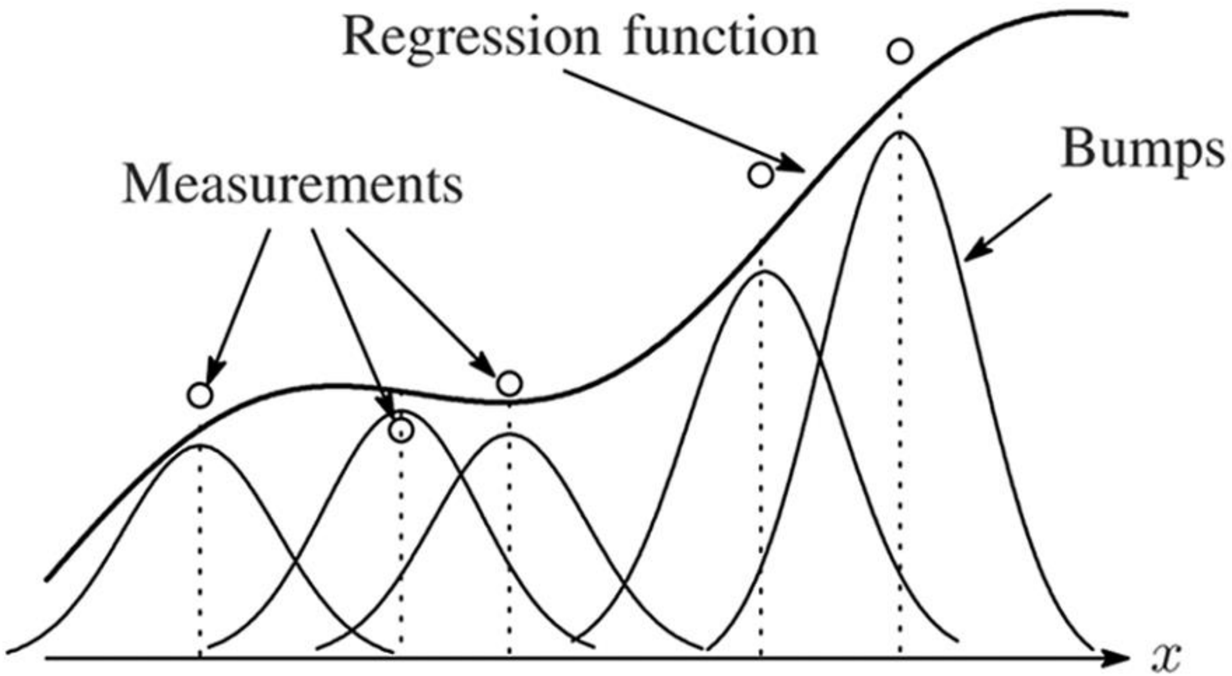
(a) A 100-gray-level step perturbed by Gaussian noise

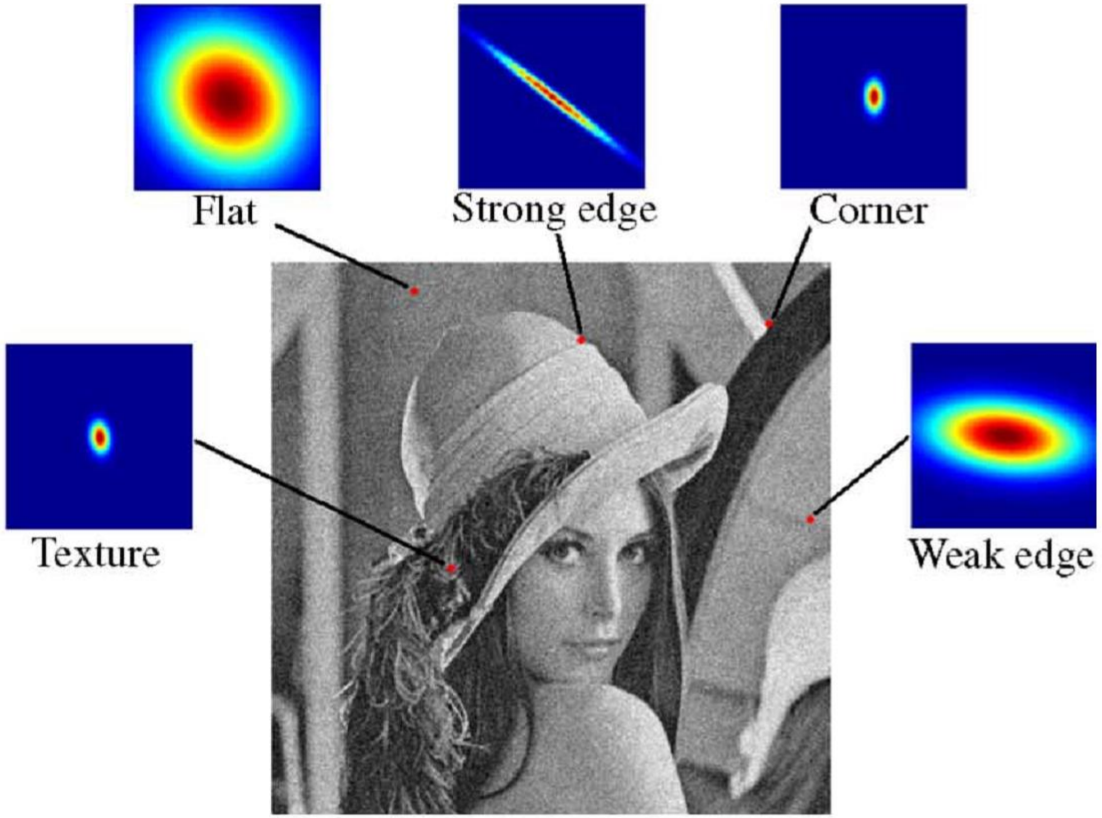
(b) Similarity weights

(c) After bilateral filter

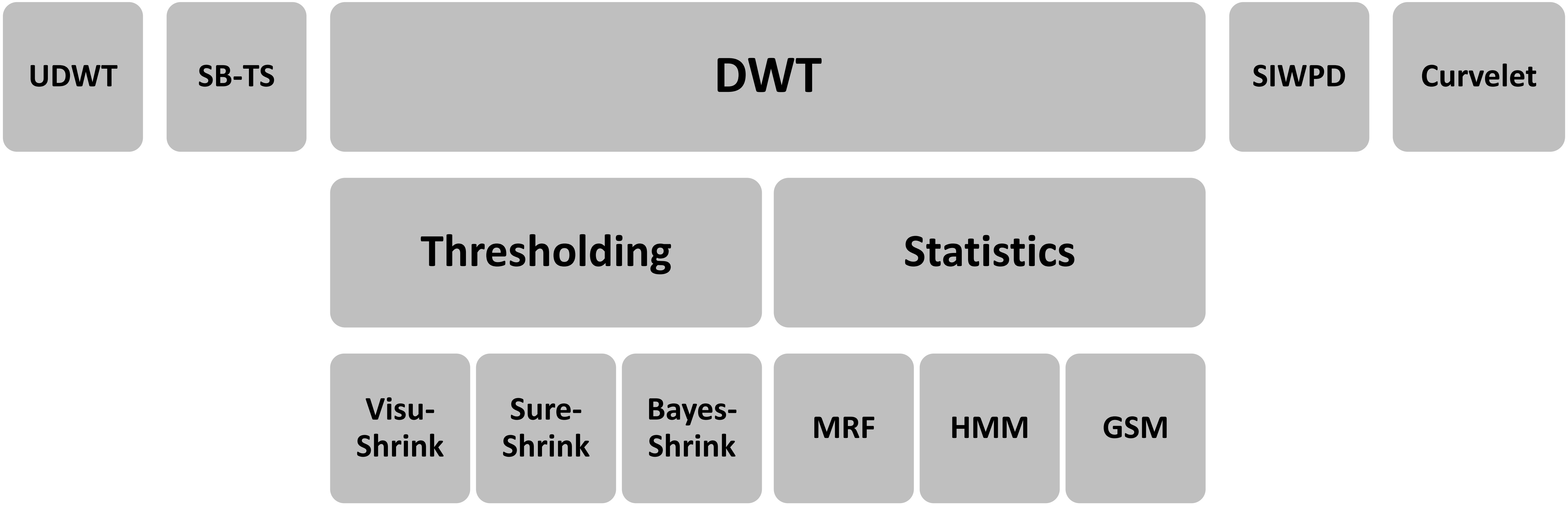
Noise Removal --- Spatial Domain

Kernel regression / Steerable filter [Takeda, et al. 2007]



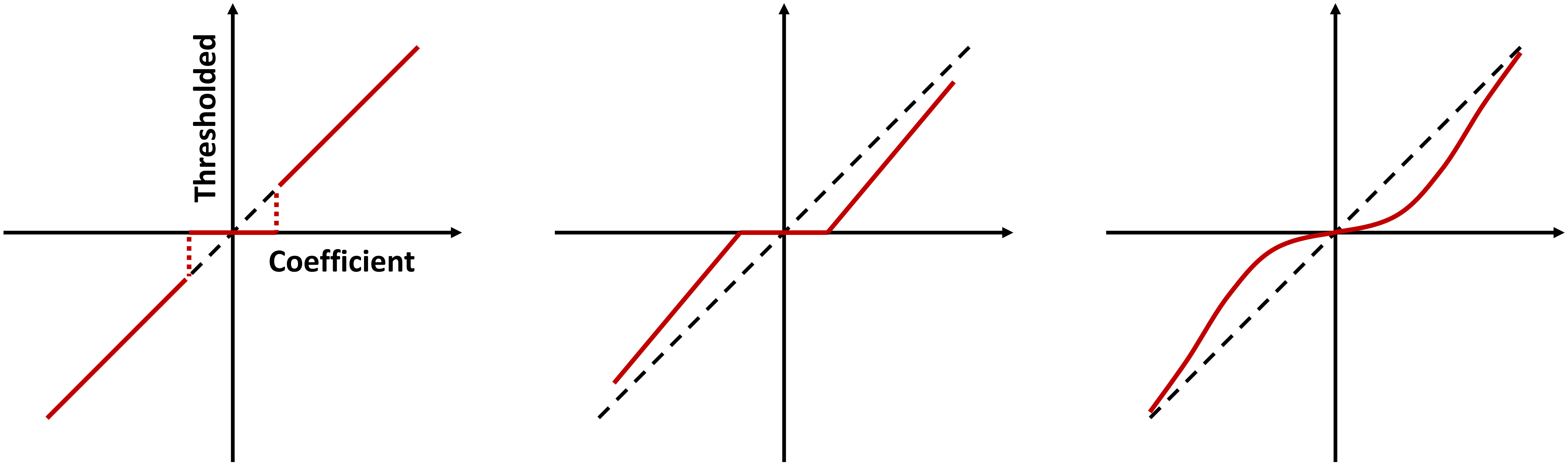


Noise Removal --- Transform Domain



Noise Removal --- Transform Domain

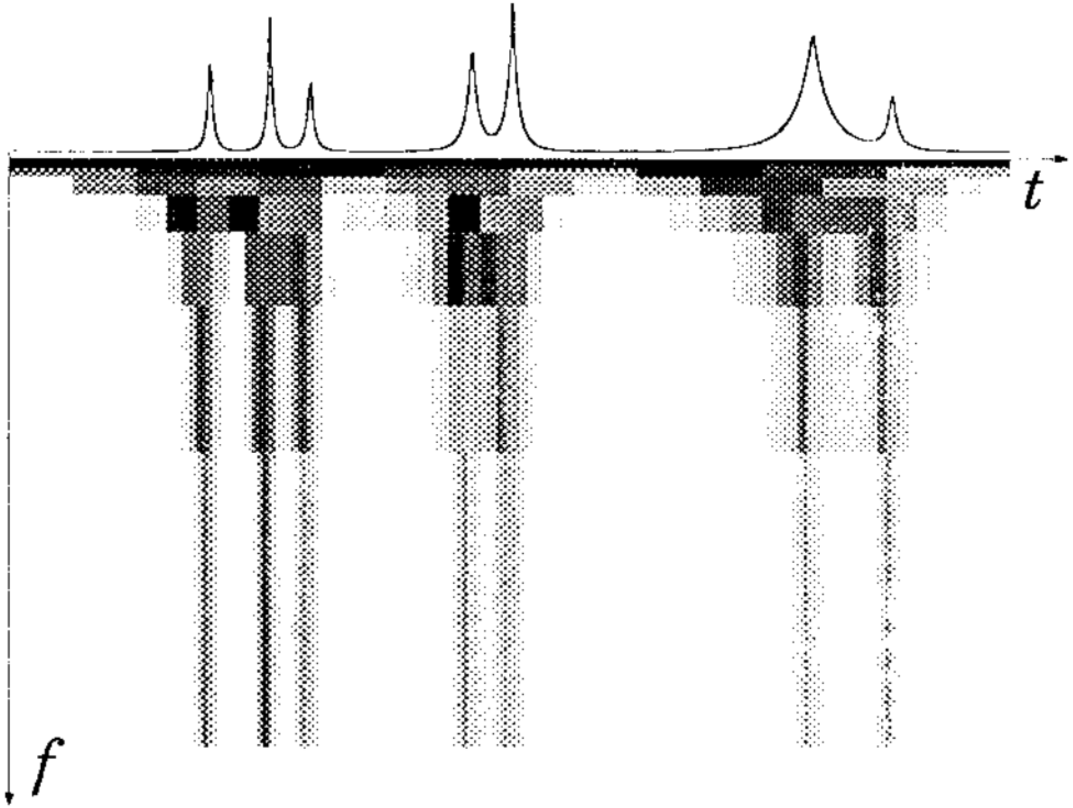
❖ DWT thresholding

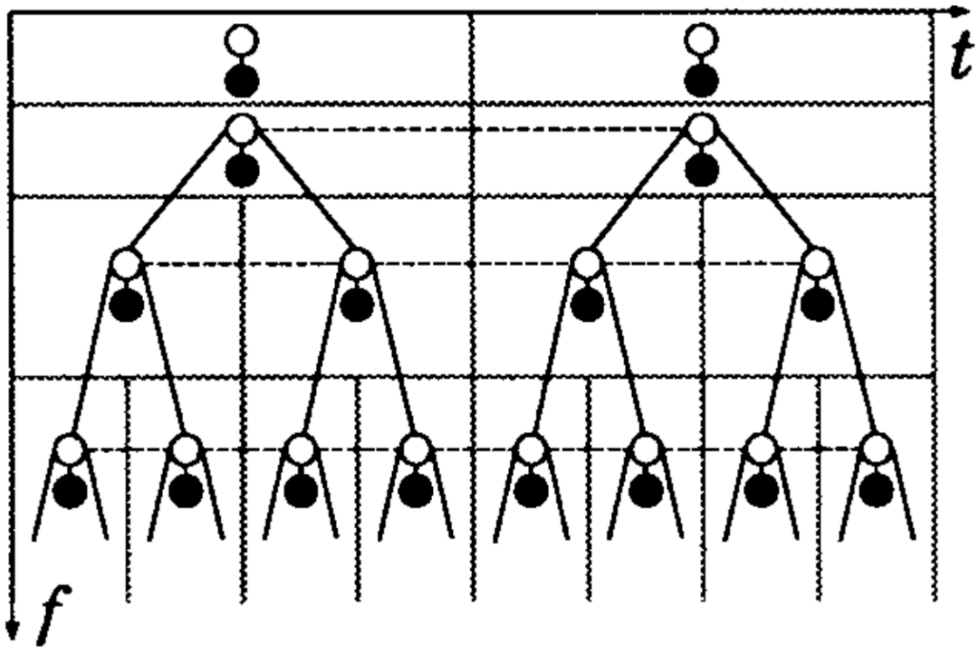


Hard thresholding | Soft thresholding | Bayes thresholding

Noise Removal --- Transform Domain

- DWT statistical modeling --- HMMs [Crouse, et al. 1998]

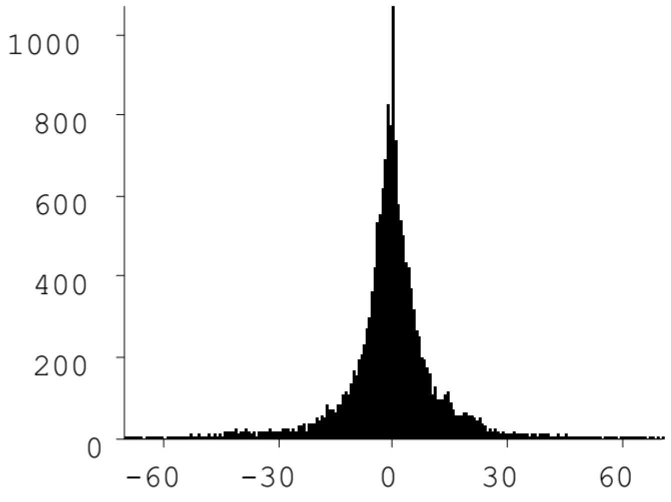


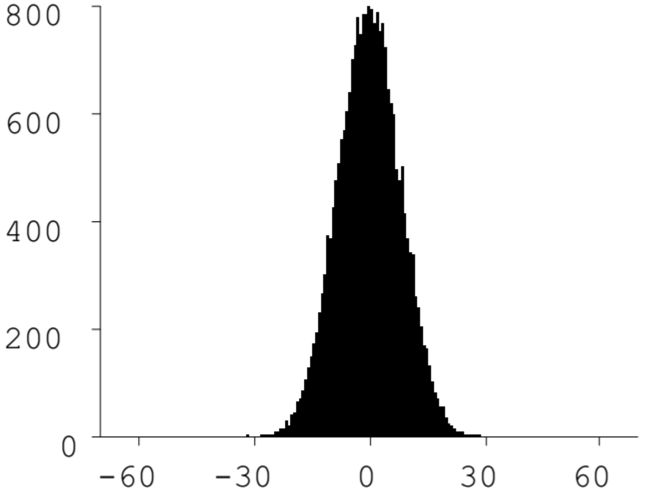


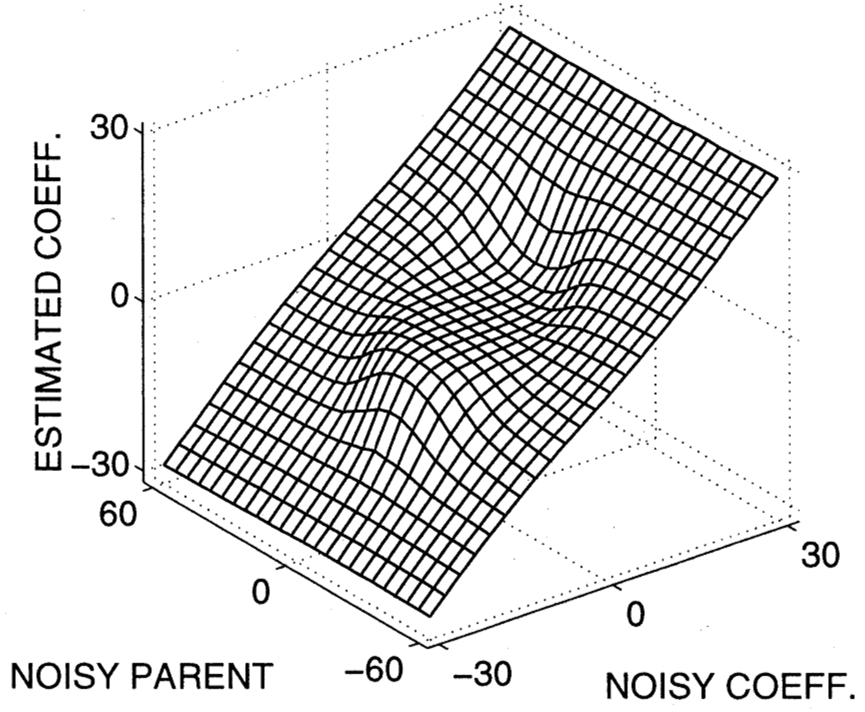
Noise Removal --- Transform Domain

❖ **DWT statistical modeling --- GSM [Portilla, et al. 2003]**

$$y = x + \varepsilon = \sqrt{z}u + \varepsilon$$



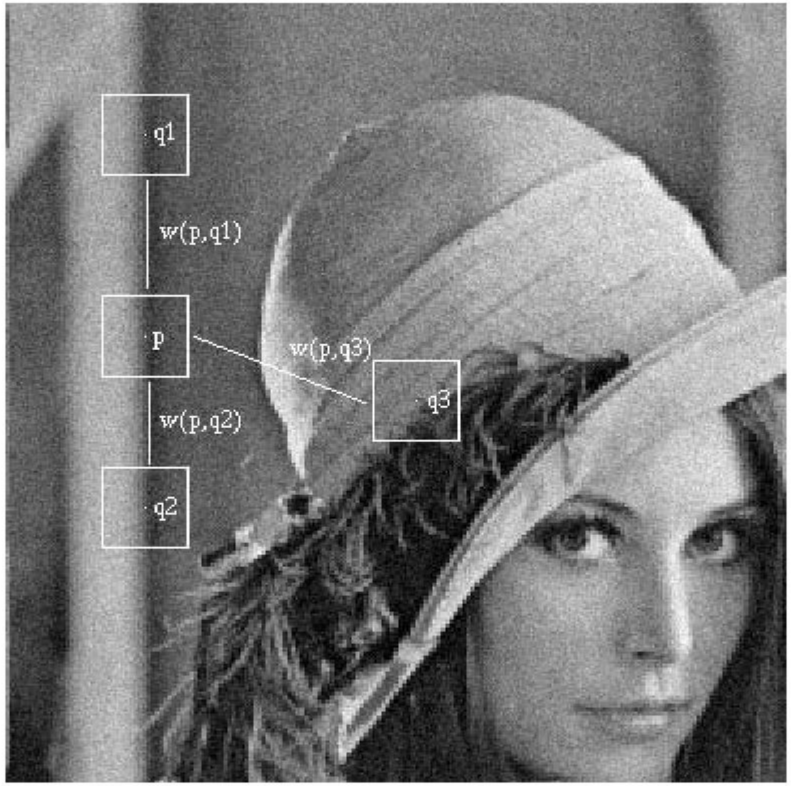




DWT coefficient histogram of noise-free and Gaussian-noise image

Noise Removal --- Non-local Method

❖ **Non local means (NL-means)** [Buades, et al. 2005]

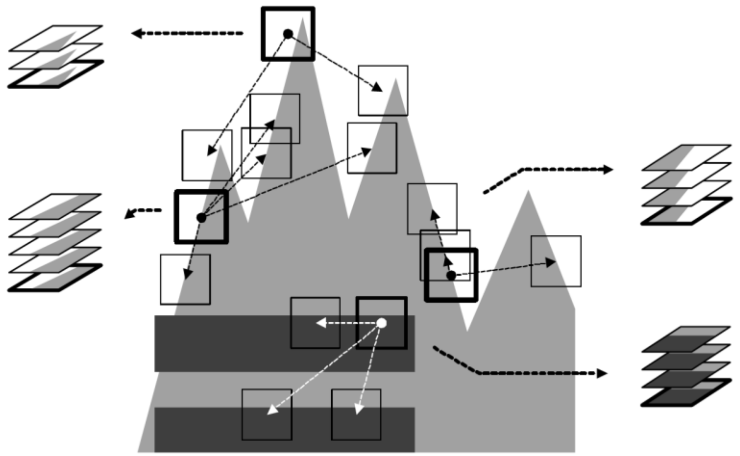


Given a discrete noisy image $\boldsymbol{v} = \{\boldsymbol{v}(\boldsymbol{i}) | \boldsymbol{i} \in \boldsymbol{I}\}$

$$\boldsymbol{NL}[\boldsymbol{v}](\boldsymbol{i}) = \sum_{\boldsymbol{j} \in \boldsymbol{I}} \boldsymbol{w}(\boldsymbol{i}, \boldsymbol{j}) \boldsymbol{v}(\boldsymbol{j})$$

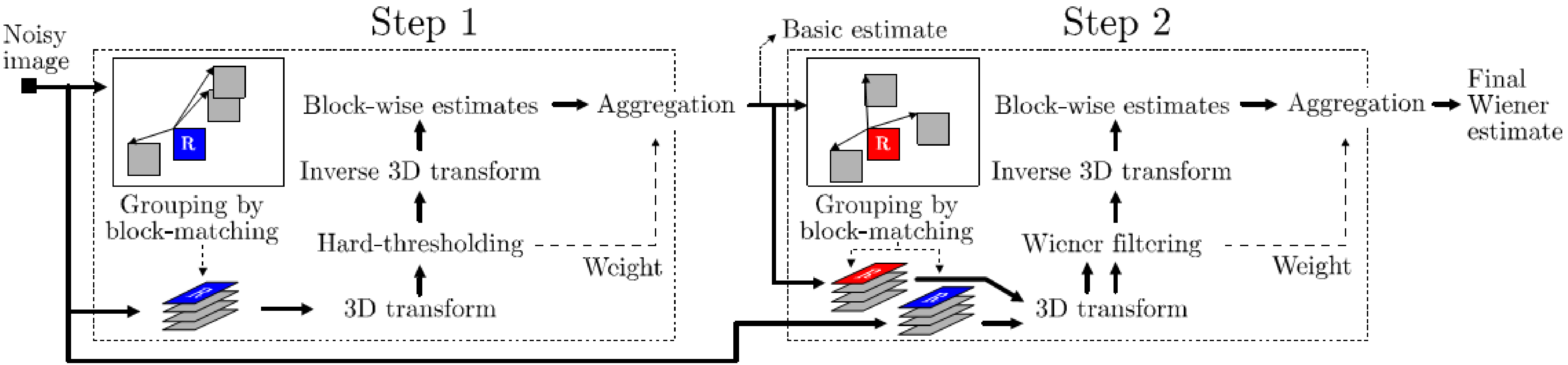
$$\boldsymbol{w}(\boldsymbol{i}, \boldsymbol{j}) = \frac{\mathbf{1}}{\boldsymbol{Z}(\boldsymbol{i})} \boldsymbol{e}^{-\frac{\left\| \boldsymbol{v}(\mathcal{N}_{\boldsymbol{i}}) - \boldsymbol{v}(\mathcal{N}_{\boldsymbol{j}}) \right\|_{\mathbf{2},\boldsymbol{a}}^{\mathbf{2}}}{\boldsymbol{h}^{\mathbf{2}}}}$$

Noise Removal --- Non-local Method



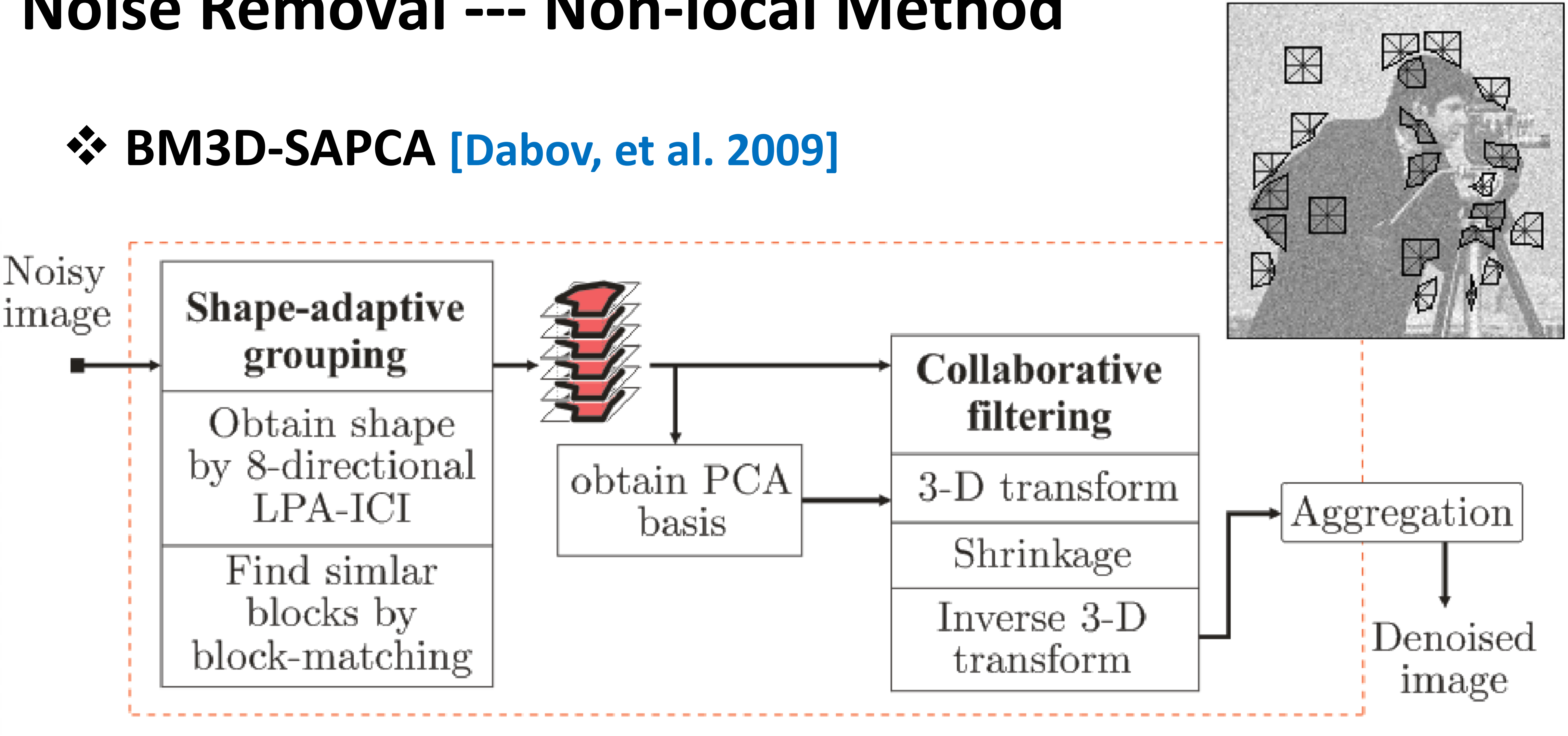
- ❖ **BM3D** [Dabov, et al. 2007]

Block matching + 3D transform + Thresholding



Noise Removal --- Non-local Method

❖ **BM3D-SAPCA** [Dabov, et al. 2009]



Noise Removal --- TV Minimization

$$TV = \sum_{i,j=1}^{n} \left|I_{ij} - I_{i,j+1}\right| + \left|I_{ij} - I_{i+1,j}\right|$$

TV

$$I = \begin{pmatrix} 124 & 100 & 30 \\ 69 & 80 & 200 \\ 66 & 92 & 211 \end{pmatrix}$$

$$\begin{pmatrix} 124 & 100 & 30 \\ 69 & 80 & 200 \\ 66 & 92 & 211 \end{pmatrix}$$

Lower TV

Noise Removal --- TV Minimization

❖ ROF [Rudin, Osher and Fatemi, 1992]

$$\min_{\mathbf{u}} \underbrace{\|\mathbf{Ku} - \mathbf{b}\|_2^2}_{\text{Fidelity term (Gaussian noise)}} + \lambda \underbrace{\sum_{i=1}^{n} \left(\nabla_x^2 u_i + \nabla_y^2 u_i\right)^{\frac{1}{2}}}_{\text{Regularization term (TV)}}$$

K --- linear operator (identity in denoising).

b --- the observation .

u --- denoised image.

Noise Removal --- TV Minimization

$$\min_{\mathbf{u}} \|\mathbf{u}-\mathbf{b}\|_2^2 + \lambda \sum_{i=1}^{n} \left(\nabla_x^2 u_i + \nabla_y^2 u_i\right)^{\frac{1}{2}}$$

Impulse noise $\|\mathbf{u}-\mathbf{b}\|_0$

Laplace noise $\|\mathbf{u}-\mathbf{b}\|_1$

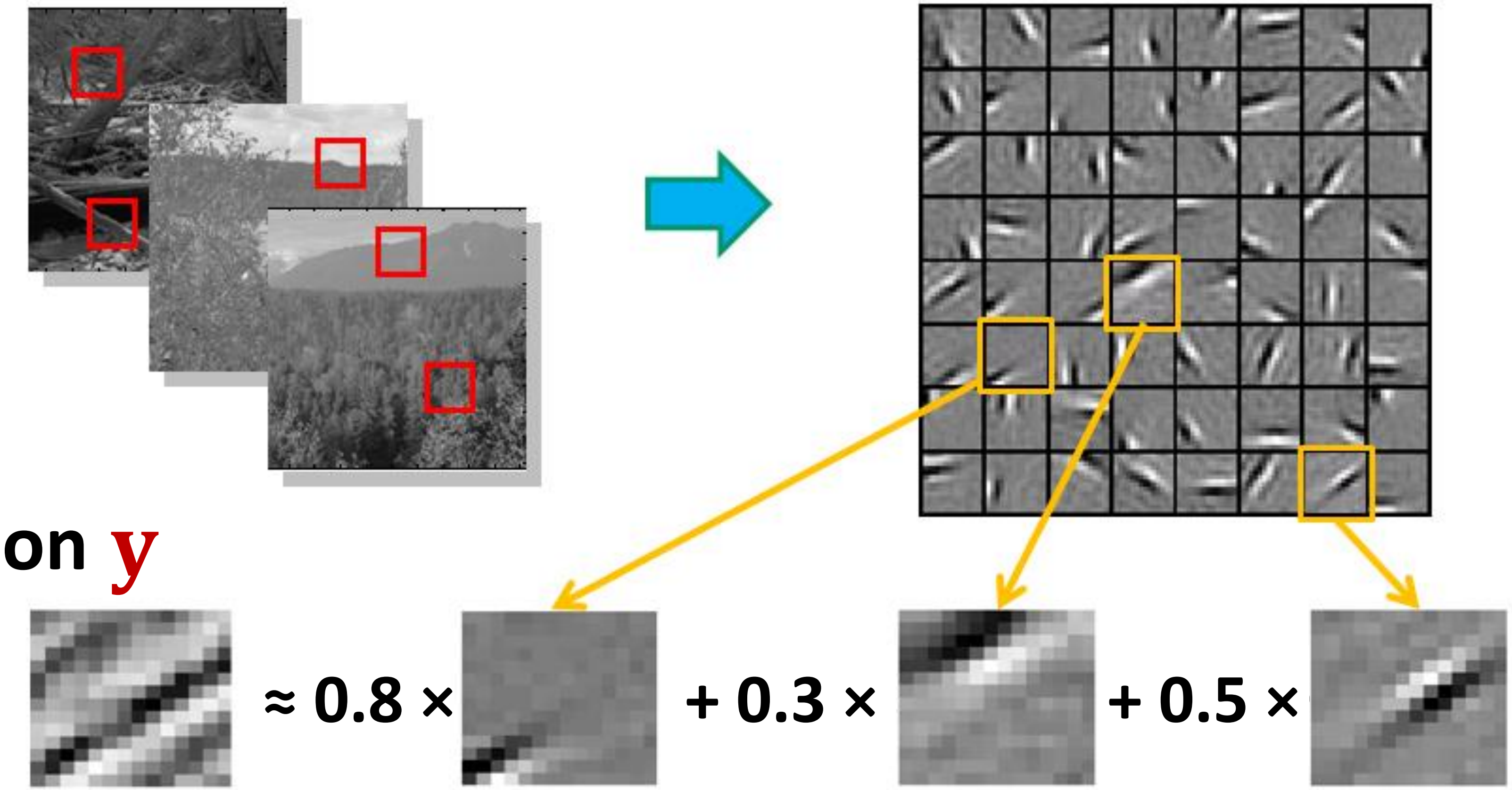
Uniform noise $\|\mathbf{u}-\mathbf{b}\|_\infty$

$$\sum_{i=1}^{n} |\nabla_x u_i| + |\nabla_y u_i|$$

$$\boxed{\sum_{i=1}^{n} \|\nabla_x u_i\|_0 + \|\nabla_y u_i\|_0}$$

Achieve best performance in [Xu, et al. 2011]

Noise Removal --- Sparse Coding



Dictionary $\mathbf{D}$

Observation $\mathbf{y}$

$[a_1, ..., a_{64}] = [0, 0, ..., 0, \mathbf{0.8}, 0, ..., 0, \mathbf{0.3}, 0, ..., 0, \mathbf{0.5}, 0]$ (feature representation)

Sparse coefficients $\boldsymbol{\alpha}$

Noise Removal --- Sparse Coding

- Sparse coding [Elad and Aharon, et al. 2006]

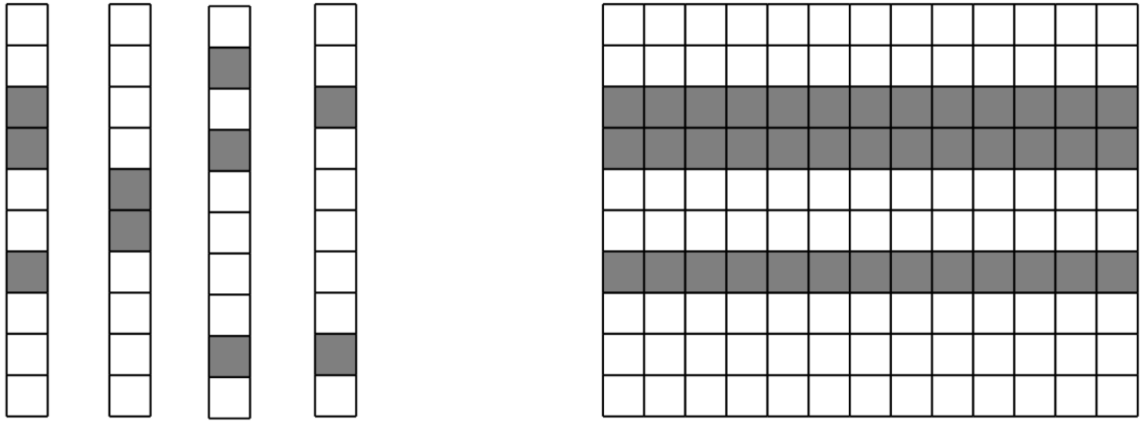
$$\hat{\alpha} = \arg\min_{\alpha} \|\mathbf{D}\alpha - \mathbf{y}\|_2^2 + \lambda \|\alpha\|_0$$

- Dictionary learning
  - K-SVD [Aharon and Elad, et al. 2006]
  - K-LLD (learned local dictionary) [Chatterjee, et al. 2009]

Noise Removal --- Sparse Coding

❖ **Learned simultaneous sparse coding (LSSC)** [Mairal, et al. 2009]

BM3D + grouped sparsity



Sparsity vs. simultaneous sparsity

$$\min_{(\mathbf{A}_i)_{i=1}^{n}, \mathbf{D} \in \mathcal{C}} \sum_{i=1}^{n} \frac{\|\mathbf{A}_i\|_{p,q}}{|S_i|^p}$$

$$\text{s.t. } \forall i \ \sum_{j \in S_i} \left\| \mathbf{y}_i - \mathbf{D}\alpha_{ij} \right\|_2^2 \leq \varepsilon_i$$

$$S_i \triangleq \left\{ j = 1, \cdots, n \ \text{ s.t. } \left\| \mathbf{y}_i - \mathbf{y}_j \right\|_2^2 \leq \xi \right\}$$

$$\mathbf{A}_i = \left[\alpha_{ij} \right]_{j \in S_i}$$

Noise Removal --- Sparse Coding

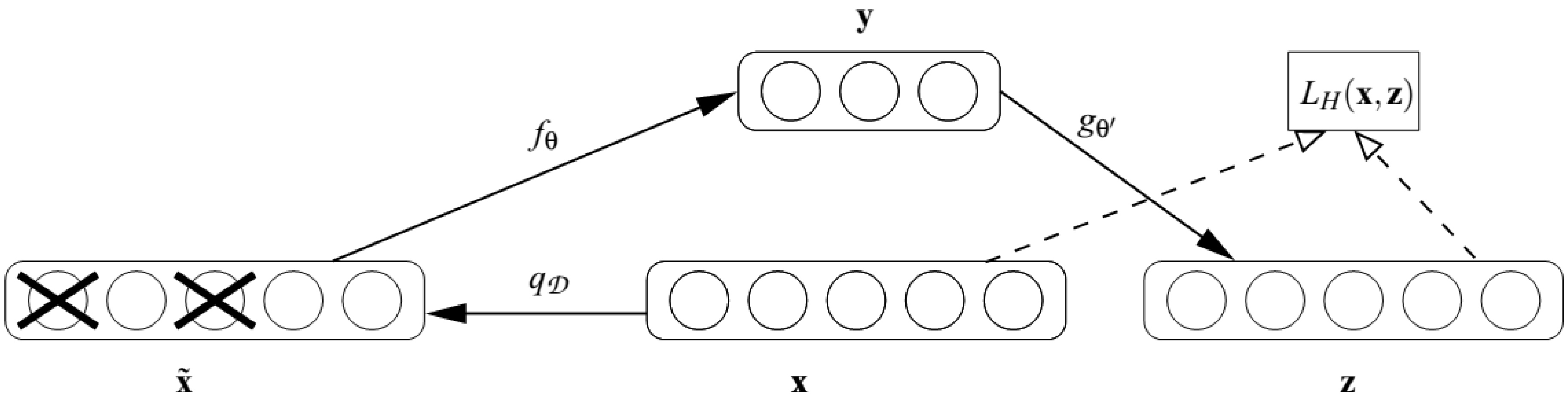
❖ **Clustering-based sparse representation (CSR)** [Dong, et al. 2011]

$$\hat{\alpha} = \arg\min_{\alpha} \frac{1}{2} \|\mathbf{D}\alpha - \mathbf{y}\|_2^2 + \lambda_1 \|\alpha\|_1 + \lambda_2 \underbrace{\sum_{k=1}^{K} \sum_{i \in C_k} \|\boldsymbol{\alpha}_i - \boldsymbol{\beta}_k\|_1}_{\text{Cluster-based regularization}}$$

- **Combine global thinking with local fitting**
- **Combine clustering and sparsity under a uniform framework**

Noise Removal --- Deep Learning

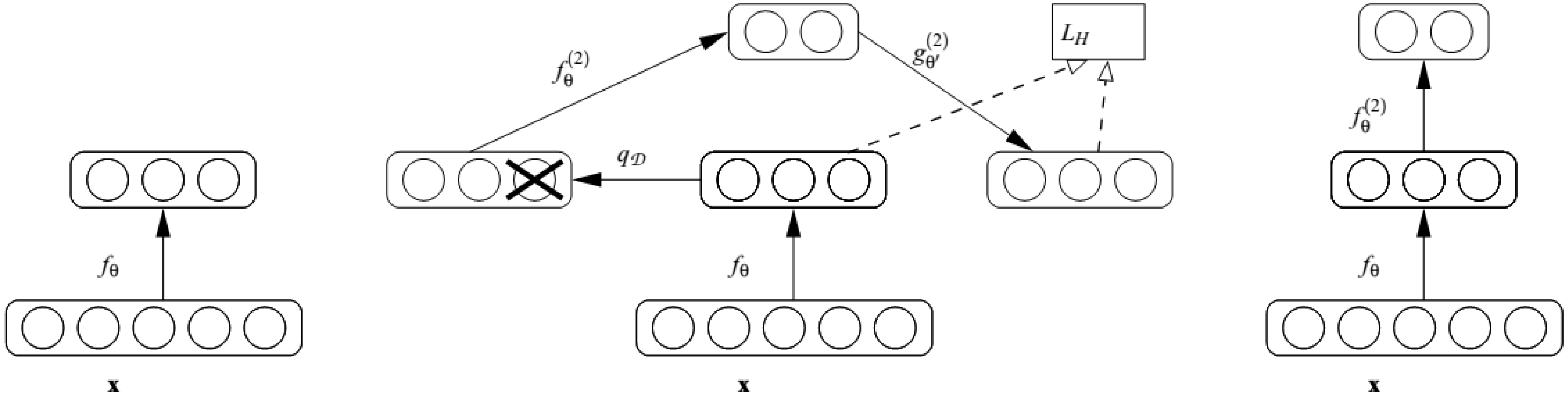
❖ **Stacked denoising auto-encoder (SDAE)** [Vincent, et al. 2010]



The denoising autoencoder architecture. An example $\mathbf{x}$ is stochastically corrupted (via $q_{\mathcal{D}}$) to $\tilde{\mathbf{x}}$. The autoencoder then maps it to $\mathbf{y}$ (via encoder f_θ) and attempts to reconstruct $\mathbf{x}$ via decoder $g_{\theta'}$, producing reconstruction $\mathbf{z}$. Reconstruction error is measured by loss $L_H(\mathbf{x},\mathbf{z})$.

Noise Removal --- Deep Learning

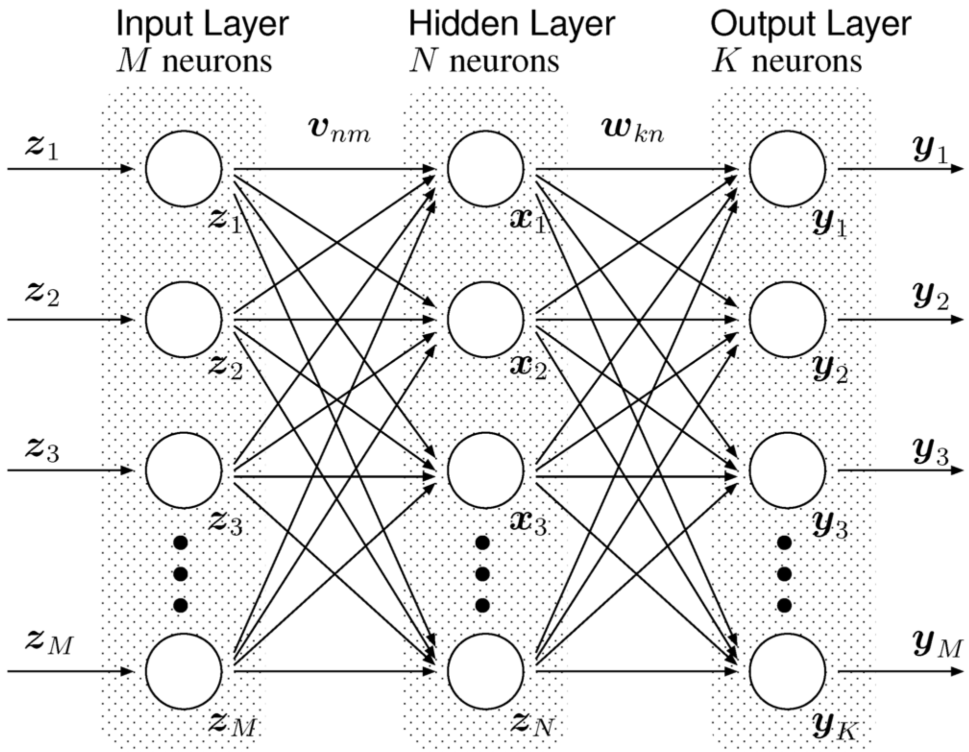
❖ **Stacked denoising auto-encoder (SDAE)** [Vincent, et al. 2010]



Stacking denoising autoencoders. After training a first level denoising autoencoder (see Figure 1) its learnt encoding function f_θ is used on clean input (left). The resulting representation is used to train a second level denoising autoencoder (middle) to learn a second level encoding function $f_\theta^{(2)}$. From there, the procedure can be repeated (right).

Noise Removal --- Deep Learning

Can neural networks compete with BM3D? [Burger, et al. 2012]



- **Clean image x**
- **Noisy image z by corrupting x with noise**
- **Denoised image y**
- **Minimize $\|x - y\|_2$**

Conclusion --- Development Tendency

Spatial domain → Transform domain

Local statistics → Non-local statistics

Thresholding → Statistical modeling

Direct estimation → Regularized optimization

Conclusion --- State-of-the-Art

- ❖ Local in spatial domain
  Kernel regression [Takeda, et al. 2007]
- ❖ Neighborhood in transform domain
  GSM [Portilla, et al. 2003]
- ❖ Non-local in transform domain
  BM3D [Dabov, et al. 2007] → BM3D-SAPCA [Dabov, et al. 2009]
- ❖ Sparse coding in transform domain
  K-SVD [Aharon and Elad, 2006] → CSR [Dong, et al. 2011]

Conclusion --- The Future

Non-local + Transform-domain + Sparsity

Intensity similarity,
Geometrical similarity
[Chatterjee, et al. 2012]

DFT, DWT, DCT
PCA, SVD
[Rajwade, et al. 2013]

Thank You!