Automatic Parameter Tuning for Image Denoising with Learned Sparsifying Transforms

Luke Pfister & Yoram Bresler
University of Illinois at Urbana-Champaign

ICASSP 2017

The Problem

$$y = Ax + e$$

- A is underdetermined or ill-conditioned
  - Must incorporate prior information!
  - Assume sparse representation

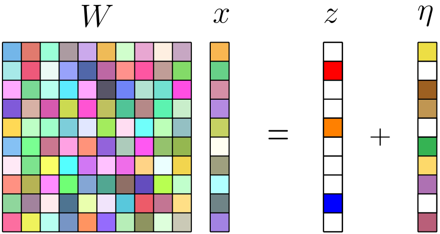
Analytic Sparse Representations

- Designed for optimal properties on a mathematical class of signals
- Fast implementation
- Hard to design for high dimensional data

Adaptive Sparse Representations

- Learn a good representation directly from data
- Synthesis sparsity: Dictionary Learning
- Analysis sparsity: Analysis Operator Learning
- Transform sparsity: Transform Learning

Transform Sparsity



- $W \in \mathbb{R}^{M \times K}$: Sparsifying transform
- $Wx \approx$ sparse
- W is left-invertible: $x \approx W^{\dagger} z$

Transform Sparsity

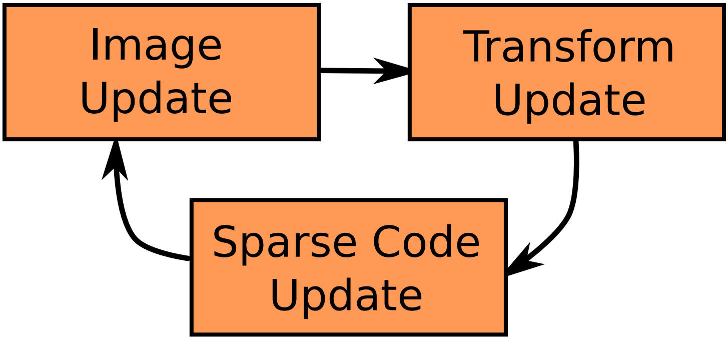
$$z^* = \min_z \frac{1}{2}\|Wx - z\|_2^2 + \nu\|z\|_0$$

$$z_i^* = \begin{cases} [Wx]_i & |[Wx]_i| \geq \sqrt{\nu} \\ 0 & \text{else} \end{cases}$$

$$\triangleq \psi(Wx)$$

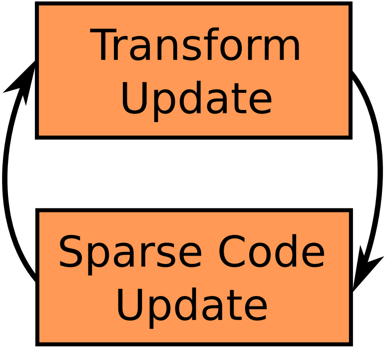
Data-Adaptive Models in Practice

Instance Approach

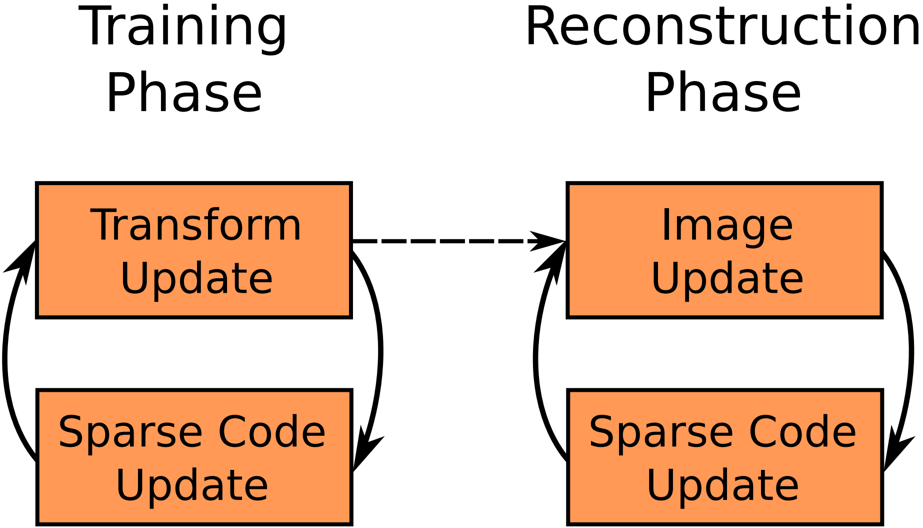


Universal Approach

Training
Phase



Universal Approach



Applications

- State of the art results in...
  - Image denoising
  - Video denoising
  - Inpainting
  - Magnetic Resonance Imaging
  - Computed tomography

Welcome to the Parameter Jungle

- Sparsity level
- Regularization
  - Conditioning, coherence, norm...
- Number of iterations
  - Learning iterations, reconstruction iterations...
- Should parameters change with iterations?

Towards A Partial Solution

- Universal paradigm: pre-learned $W : M \times K$
- Restrict attention to image denoising
- $y = x^* + e$

Sparsifying Transforms for Images

Patch-based Models

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

Patch-based Models

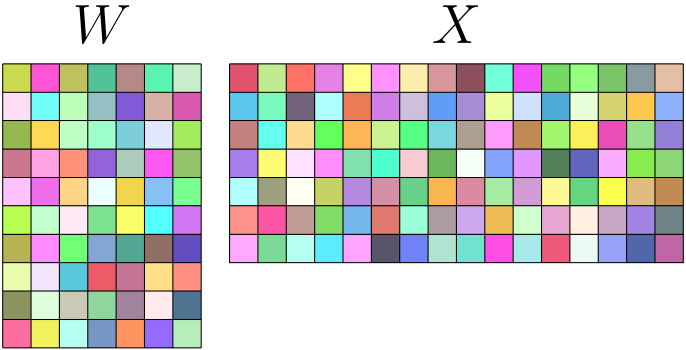
$$R_1 x$$

Patch-based Models

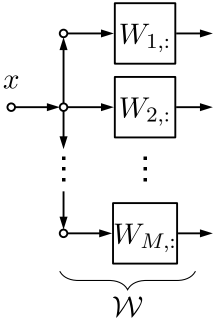
1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

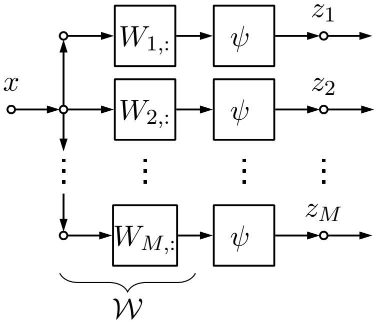
1	2
2	3
7	8
8	9

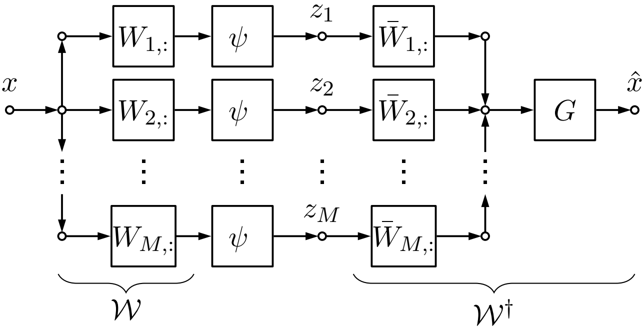
$$R_2 x$$

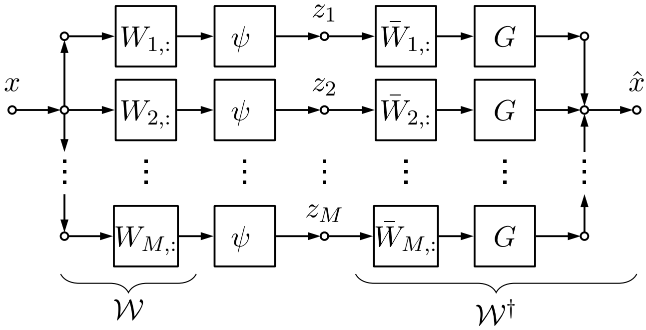

W
X

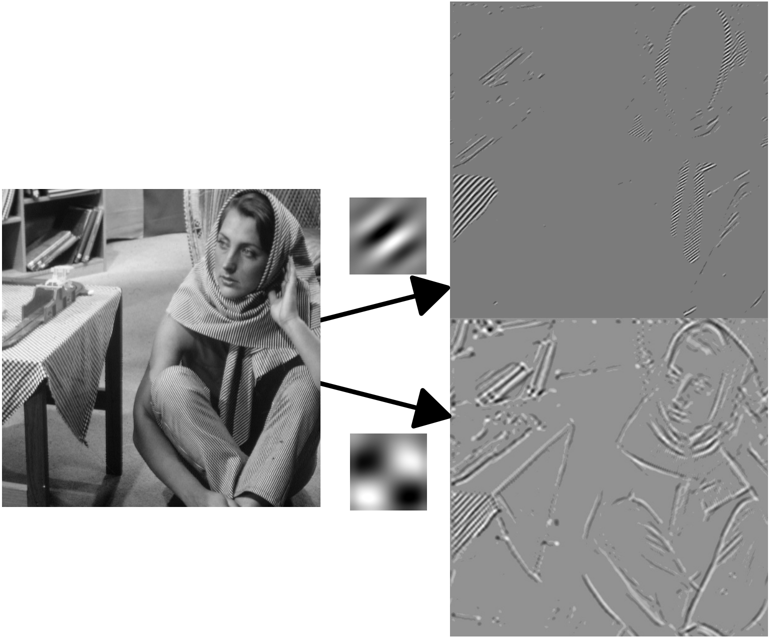
Filter bank Sparsifying Transforms


x
$W_{1,:}$
$W_{2,:}$
$W_{M,:}$
$\mathcal{W}$


x
W_{1,:}
W_{2,:}
W_{M,:}
ψ
ψ
ψ
z_1
z_2
z_M
𝒲


x
$W_{1,:}$
$W_{2,:}$
$W_{M,:}$
ψ
ψ
ψ
z_1
z_2
z_M
$\bar{W}_{1,:}$
$\bar{W}_{2,:}$
$\bar{W}_{M,:}$
G
$\hat{x}$
$\mathcal{W}$
$\mathcal{W}^\dagger$


x
$W_{1,:}$
$W_{2,:}$
$W_{M,:}$
ψ
ψ
ψ
z_1
z_2
z_M
$\bar{W}_{1,:}$
$\bar{W}_{2,:}$
$\bar{W}_{M,:}$
G
G
G
$\hat{x}$
$\mathcal{W}$
$\mathcal{W}^\dagger$



Automatic Parameter Tuning

The Setup

- Noisy signal $y = x^* + e$
- $e \sim \mathcal{N}(0, \sigma^2 I_n)$
- Denoising function: $\hat{x} = F_\theta(y)$
- Mean Squared Error: MSE $= n^{-1}\|x^* - F_\theta(y)\|_2^2$

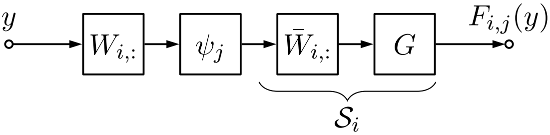
In a perfect world...

$$\hat{x} = \arg\min_{\theta} \|x^* - F_\theta(y)\|_2^2 / n$$

- Next best thing: **Stein's Unbiased Risk Estimator (SURE)**
- Unbiased estimate of MSE given only y, F_θ, and σ

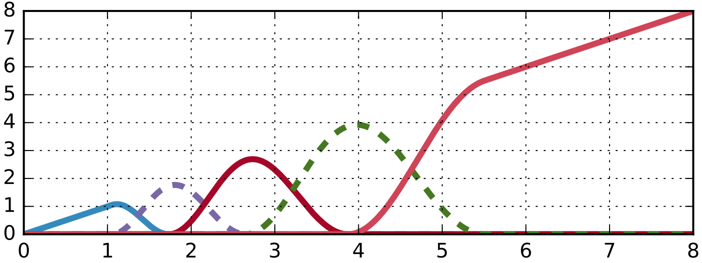
SURELET (Blu, 2007)

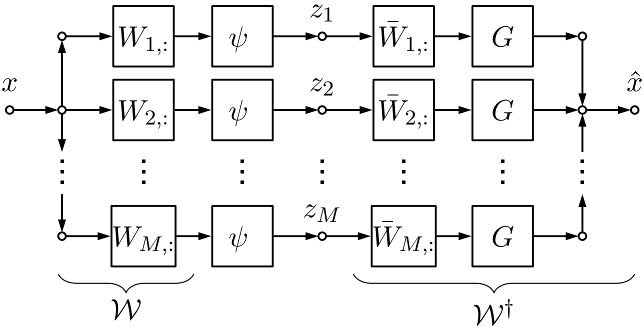
- Use a linear combination of elementary denoising functions
- $F_\theta(y) = \sum_{i,j} c_{i,j} F_{i,j}(y) = \mathbf{Fc}$
- $F_{i,j}(y) = \mathcal{S}_i \psi_j(\mathcal{W}_i y) \qquad i = 1, \dots M \quad j = 1, \dots 5$



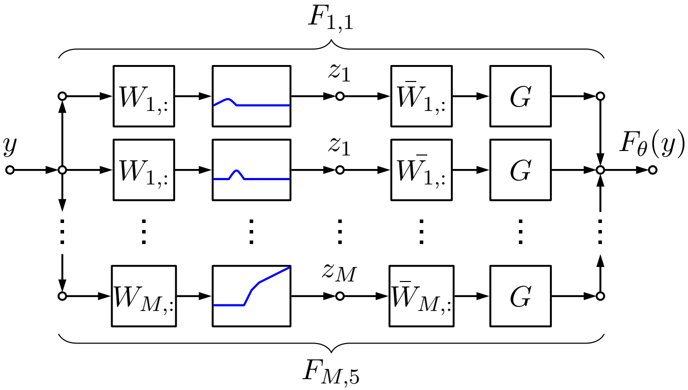
- ψ must be differentiable

SURE-BUMP Functions (Simoncelli, 2008)





x
$W_{1,:}$
$W_{2,:}$
$W_{M,:}$
ψ
ψ
ψ
z_1
z_2
z_M
$\bar{W}_{1,:}$
$\bar{W}_{2,:}$
$\bar{W}_{M,:}$
G
G
G
$\hat{x}$
$\mathcal{W}$
$\mathcal{W}^\dagger$


$F_{1,1}$
z_1
$W_{1,:}$
$\bar{W}_{1,:}$
G
y
z_1
$W_{1,:}$
$\bar{W}_{1,:}$
G
$F_\theta(y)$
z_M
$W_{M,:}$
$\bar{W}_{M,:}$
G
$F_{M,5}$

SURELET

$$\min_{c_{i,j}} \frac{1}{n} \|F_\theta(y) - y\|_2^2 + \frac{2\sigma^2}{n} \sum_{i,j} c_{i,j} \underbrace{\alpha_i^T \psi_j'(\mathcal{W}_i y)}_{v_{ij}} + \sigma^2$$

- $\alpha_i = \operatorname{diag}\{\mathcal{W}_i \mathcal{S}_i\}$

SURELET

$$\min_{\mathbf{c}} \frac{1}{n}\|\mathbf{Fc} - y\|_2^2 + \frac{2\sigma^2}{n}\mathbf{c}^T\mathbf{v} + \sigma^2$$

- $\alpha_i = \mathrm{diag}\left\{\mathcal{W}_i\mathcal{S}_i\right\}$

SURELET Denoising

1: Calculate α_i

2: $F_{i,j}(y) = \mathcal{S}_i \psi_j(\mathcal{W}_i y)$

3: $v_{i,j} = \alpha_i^T \psi_j'(\mathcal{W}_i y)$

4: $\mathbf{c} = (\mathbf{F}^T\mathbf{F})^{-1}(\mathbf{F}^T y + \sigma^2 \mathbf{v})$

5: $\hat{x} = \mathbf{F}\mathbf{c}$

- Main computation: $\mathcal{S}_i \phi_j(\mathcal{W}_i y)$
- but no parameter tuning!

Other Denoising Approaches

Iterative Patch Denoising (IPD)

- Independent collection of patches $\bar{x}_j = R_j y$

$$\min_{\bar{x}_j, z_j} \frac{1}{2}\|W\bar{x}_j - z_j\|_2^2 + \frac{\lambda}{2}\|R_j y - \bar{x}_j\|_2^2 + \nu\phi(z_j),$$

1: **repeat**
2: $\quad z_j^k \leftarrow \psi(W\bar{x}_j^k)$
3: $\quad \bar{x}_j^k \leftarrow (W^T W + \lambda I)^{-1}(W^T z_j^k + \lambda R_j y)$
4: **until** Halting condition
5: $x = \frac{1}{K}\sum_j R_j^T \bar{x}_j^k$

Global Patch Denoisng (GPD)

- Link patches directly to image

$$\min_{x,z_j} \frac{1}{2} \sum_{j=1} \|WR_jx - z_j\|_2^2 + \frac{\lambda}{2}\|y - x\|_2^2 + \nu\phi(z_j),$$

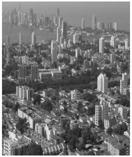
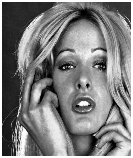
1: **repeat**
2: $\quad z_j^k \leftarrow \psi\left(WR_jx^k\right)$
3: $\quad x^k \leftarrow (\sum_j R_j^T W^T W R_j + \lambda I)^{-1}(\sum_j R_j^T W^T z_j^k + \lambda y)$
4: **until** Halting condition

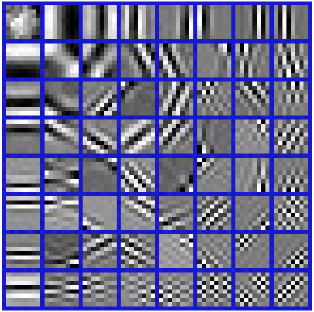
Experiments

Experiments

- Pre-learned transform
- 8×8 patches, 64 channels
- Compare against
  - SURELET with fixed transform
  - IPD/IGD with oracle information
  - IPD + instance paradigm
- Report mean PSNR (dB) over test set





Experiments

Experiments: Mean PSNR

σ	10	20	30
Input PSNR	28.13	22.11	18.59
BM3D	34.0	30.8	29.0
SURELET-SWT	33.0	29.5	27.4
SURELET-DCT	33.7	30.1	28.1
IGD-DCT	33.7	30.3	28.4
IPD-AST	33.8	30.2	28.2
IGD-LST	33.8	30.3	28.4
SURELET-LST	33.8	30.2	28.2

Conclusions

- SURELET denoising with learned sparsifying transform
  - Better than SURELET with analytic transform
  - Easier than iterative denoising with learned transform
- Next steps
  - Incorporate into the learning process
  - Extend to non-Gaussian noise and imaging problems

Thanks!

- Blu, T., & Luisier, F., The SURE-LET approach to image denoising. IEEE Transactions on Image Processing 2007
- Raphan, M., & Simoncelli, E., Optimal denoising in redundant representations, IEEE Transactions on Image Processing, 2008
- `http://transformlearning.csl.illinois.edu`

Patch Thresholding

1: $z_j \leftarrow \psi\left(W R_j y\right)$
2: $x \leftarrow \frac{1}{K} \sum_j R_j^T W^\dagger z_j$

- Why not use SURELET on patches?
  - Too many parameters!

Full Table

Method	baboon			peppers512			barbara		
BM3D	30.47	26.45	24.40	34.75	32.51	31.01	34.95	31.74	29.78
SURELET-SWT	30.03	25.84	23.80	34.14	31.49	29.72	32.69	28.48	26.17
SURELET-DCT	30.44	26.31	24.22	34.55	31.70	29.91	34.22	30.37	28.18
IGD-DCT	30.40	26.31	24.26	34.57	32.07	30.50	33.98	30.16	27.97
IPD-AST	30.43	26.35	24.22	34.58	31.90	30.13	34.34	30.52	28.27
IGD-LST	30.49	26.40	24.31	34.67	32.25	30.68	33.98	30.00	27.71
SURELET-LST	30.52	26.43	24.31	34.68	31.89	30.07	34.30	30.51	28.24
Method	**man**			**boat**			**sailboat**		
BM3D	33.94	30.56	28.81	33.90	30.84	29.04	32.51	29.52	27.84
SURELET-SWT	33.16	29.75	28.02	33.26	29.89	28.07	32.11	28.89	27.02
SURELET-DCT	33.47	29.92	28.05	33.64	30.24	28.28	32.56	29.26	27.41
IGD-DCT	33.53	30.10	28.33	33.68	30.45	28.61	32.41	29.36	27.66
IPD-AST	33.64	30.00	28.09	33.63	30.24	28.28	32.51	29.26	27.40
IGD-LST	33.77	30.24	28.39	33.76	30.51	28.65	32.48	29.42	27.67
SURELET-LST	33.74	30.16	28.20	33.79	30.38	28.39	32.68	29.36	27.48

Thresholding Functions

$$\psi_j(x) = x \cdot f(\alpha \log(|z|\,\sigma^{-1} + 1) + \beta - j)$$
$$f(x) = \cos^2\left(\frac{\pi x}{2}\right)$$

α, β set center of ψ_1, ψ_4 to $\sqrt{3}\|W_{i,:}\|_2$ and $\sqrt{15}\|W_{i,:}\|_2$

Calculating α_i for cyclic convolution

Q: 2D-DFT matrix

$$\alpha_i = \left\| \frac{|Qw_i|^2}{\sum_{j=1}^{M} |Qw_j|^2} \right\|_1 \cdot \mathbf{1}_n$$