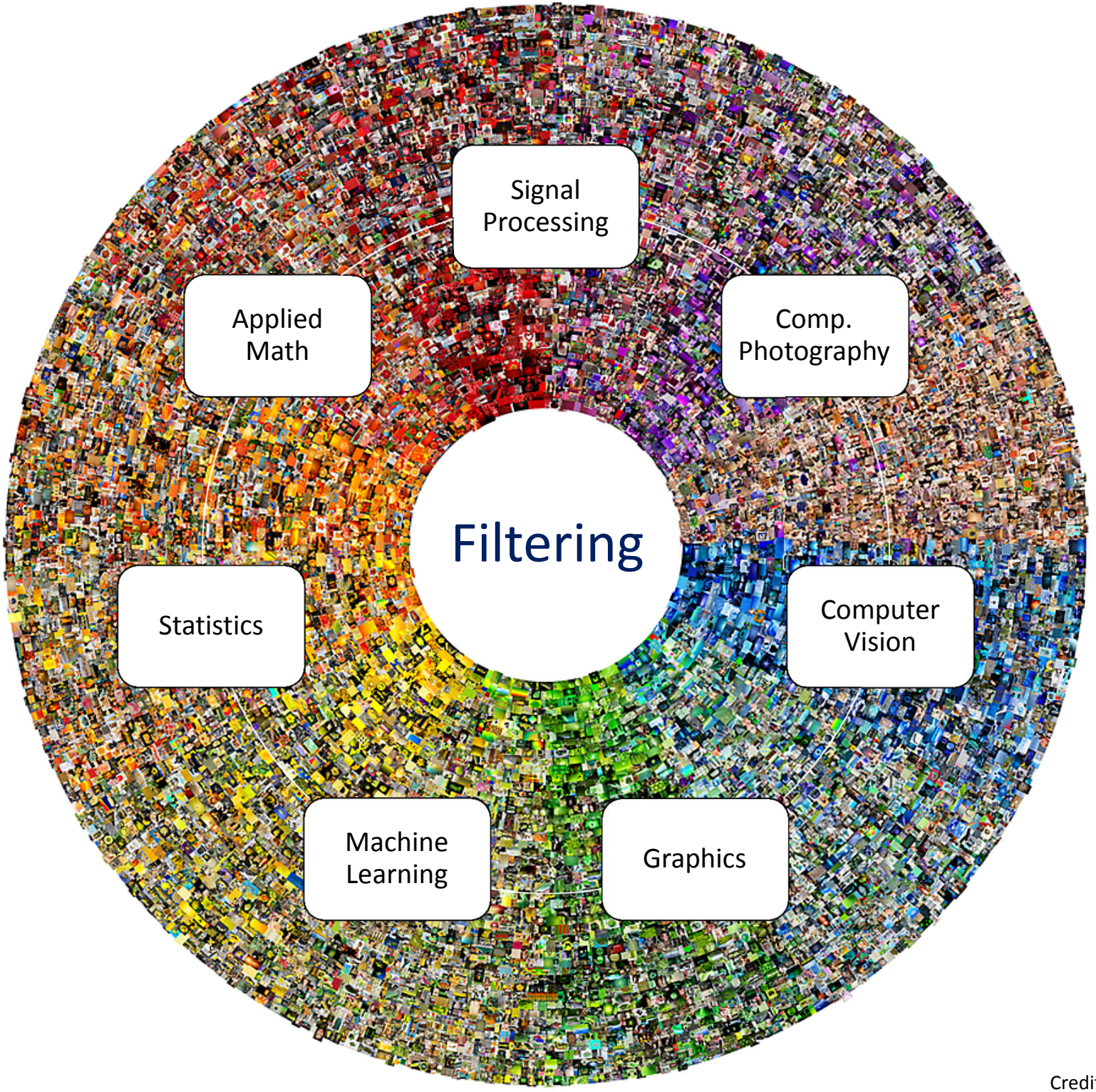
A Wide-Angle View of Image Filtering

Peyman Milanfar

EE Department
University of California, Santa Cruz

SIAM Imaging Science, May 2012



Credit: Jason Salavon

The Smoothing Problem

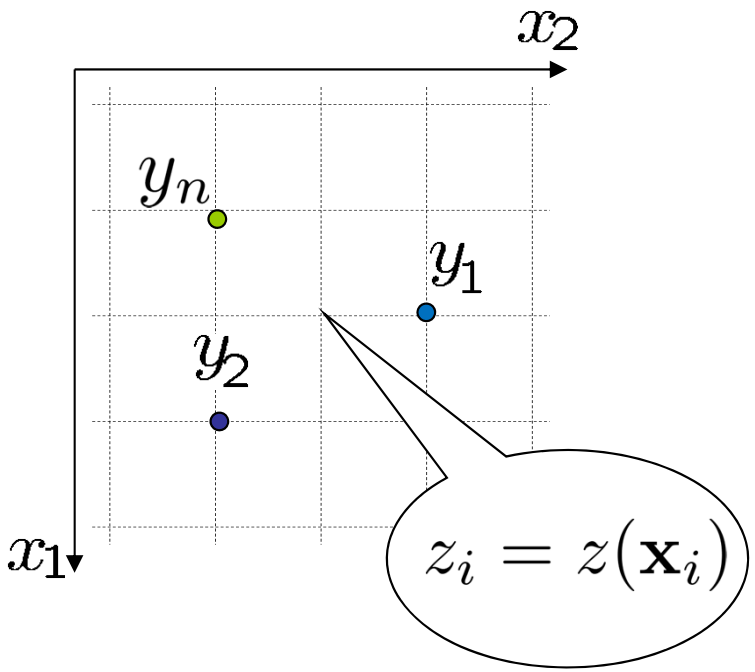
Zero-mean, i.i.d noise (No other assump.)

$$y_i = z_i + e_i, \qquad \text{for} \quad i = 1, \cdots, n$$

Noisy samples

Clean samples

The number of samples



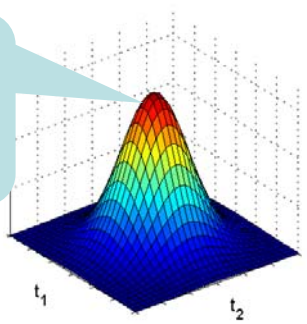
Non-parametric Regression

$$y_i = z_i + e_i, \qquad \text{for} \quad i = 1, \cdots, n$$

- The point estimate (one pixel from many):

$$\widehat{z}_j = \arg\min_{z_j} \sum_{i=1}^{n} (y_i - z_j)^2 \; K(x_i, x_j, y_i, y_j)$$

This **Kernel** measure similarity between two data points i and j.



(Point) Estimate: Matrix Formulation

- Weighted Least Squares problem:

$$\widehat{z}_j = \arg\min_{z_j} \left[\mathbf{y} - z_j \mathbf{1}\right]^T \mathbf{K}_j \left[\mathbf{y} - z_j \mathbf{1}\right]$$

where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \qquad \mathbf{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}, \qquad \mathbf{K}_j = \text{diag} \begin{bmatrix} K(x_1, x_j, y_1, y_j) \\ K(x_2, x_j, y_2, y_j) \\ \vdots \\ K(x_n, x_j, y_n, y_j) \end{bmatrix}$$

Solution: Locally Adaptive Filters

$$\widehat{z}_j = \arg\min_{z_j} \left[\mathbf{y} - z_j \mathbf{1}\right]^T \mathbf{K}_j \left[\mathbf{y} - z_j \mathbf{1}\right]$$

$$\Downarrow$$

$$\begin{aligned}
\widehat{z}_j &= \left(\mathbf{1}^T \mathbf{K}_j \mathbf{1}\right)^{-1} \mathbf{1}^T \mathbf{K}_j \mathbf{y} \\
&= \sum_i \frac{K(x_i, x_j, y_i, y_j)}{\sum_i K(x_i, x_j, y_i, y_j)} y_i \\
&= \boxed{\sum_i W_{i,j}\, y_i} \\
&= \mathbf{w}_j^T \mathbf{y}.
\end{aligned}$$

Convex combination of **all** the data.

Some Familiar Special Cases

- Bilateral Filter (Tomasi, Manduchi, '98)

$$K_{ij} = \exp\left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-(y_i - y_j)^2}{h_y^2} \right\}$$

- Non-local Means (Buades et al. '05)

Patches

$$K_{ij} = \exp\left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h_y^2} \right\}$$

∞

- LARK (Takeda et al. '07)

"Learned" Metric

$$K_{ij} = \exp\left\{ -(x_i - x_j)^T \widehat{\mathbf{C}}_{ij}(y)(x_i - x_j) \right\}$$

Generalizations I

- General Gaussian Kernel with $\mathbf{t} = \begin{bmatrix} x \\ \mathbf{y} \end{bmatrix}$

Position

Gray-level

$$K(\mathbf{t}_i, \mathbf{t}_j) = \exp\left\{-(\mathbf{t}_i - \mathbf{t}_j)^T \mathbf{Q}_{i,j} (\mathbf{t}_i - \mathbf{t}_j)\right\}$$

$$\mathbf{Q}_{i,j} = \begin{bmatrix} \mathbf{Q}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_y \end{bmatrix}$$

← Symmetric, positive-definite

- Classical: $\mathbf{Q}_x = \frac{1}{h_x^2}\mathbf{I}$ and $\mathbf{Q_y} = \mathbf{0}$
- Bilateral: $\mathbf{Q}_x = \frac{1}{h_x^2}\mathbf{I}$ and $\mathbf{Q_y} = \frac{1}{h_y^2}\mathrm{diag}[0, 0, \cdots, 1, \cdots, 0, 0]$
- Non-local Means: $\mathbf{Q_x} = \mathbf{0}$ and $\mathbf{Q_y} = \frac{1}{h_y^2}\mathbf{G}$
- LARK: $\mathbf{Q}_x = \mathbf{C}_{ij}$ and $\mathbf{Q_y} = \mathbf{0}$.

Generalizations

$$K(\mathbf{t}_i, \mathbf{t}_j) = \exp\left\{-(\mathbf{t}_i - \mathbf{t}_j)^T \mathbf{Q}_{i,j} (\mathbf{t}_i - \mathbf{t}_j)\right\}$$

$$\mathbf{Q}_{i,j} = \begin{bmatrix} \mathbf{Q}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_y \end{bmatrix}$$ ← Symmetric, positive-definite

- Introduce off-diagonal blocks for Q
- Define the "feature" vector **t** more generally
- Use General class of Reproducing Kernels

Generalizations III

- Admissible Kernels
  - $K(\mathbf{t}, \mathbf{s}) = K(\mathbf{s}, \mathbf{t}) \geq 0$
  - Positive definiteness:

    For $\{\mathbf{t}_i\}_{i=1}^n$, the *Gram* matrix $\mathbf{K}_{i,j} = K(\mathbf{t}_i, \mathbf{t}_j)$ is symmetric positive definite.

- Given $K_1(\mathbf{t}, \mathbf{s})$, and $K_2(\mathbf{t}, \mathbf{s})$
  - Endless new constructions are possible:
    - $K(\mathbf{t}, \mathbf{s}) = \alpha K_1(\mathbf{t}, \mathbf{s}) + \beta K_2(\mathbf{t}, \mathbf{s}) \qquad \alpha,\, \beta \geq 0$
    - $K(\mathbf{t}, \mathbf{s}) = K_1(\mathbf{t}, \mathbf{s})\, K_2(\mathbf{t}, \mathbf{s})$
    - $\cdots$

The Matrix Formulation

- Collect all the point-wise estimates:

$$\widehat{z}_j = \sum_i W_{i,j}\, y_i = \mathbf{w}_j^T \mathbf{y}$$

rows

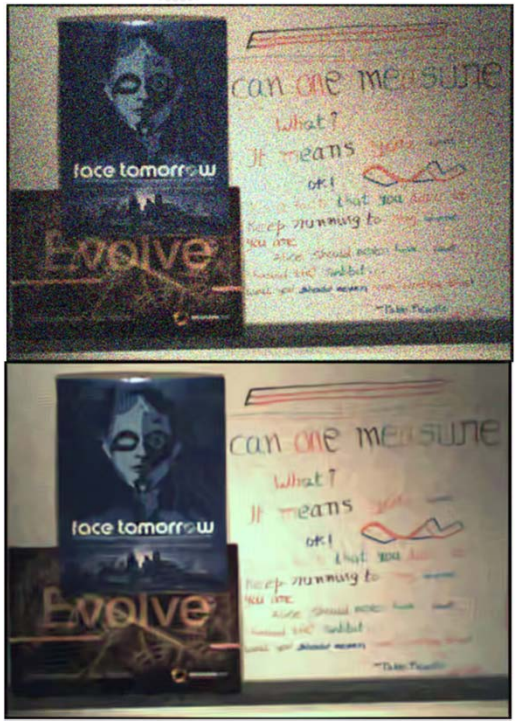
$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_n^T \end{bmatrix}$$

$$\widehat{\mathbf{z}} = \mathbf{W}\, \mathbf{y}$$

Generally data-dependent nxn matrix

Note: “Patch-free” formulation

The Ubiquity of $\hat{\mathbf{z}} = \mathbf{W}\,\mathbf{y}$



BM3D

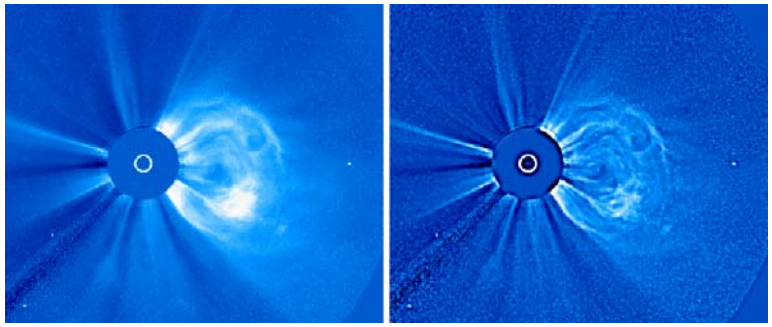
Beltrami Kernel

Shock Filters, Diffusion

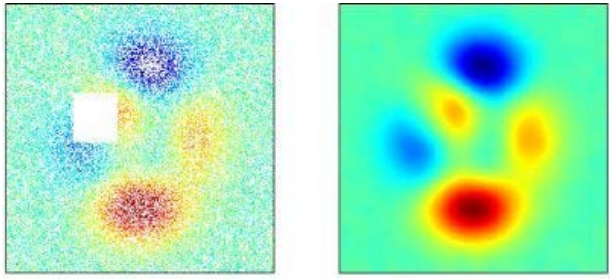
Moving Least-Squares

Gaussian Filtering

Bilateral Filter



Wavelet Filtering



Spline Smoother

Non-local Means

The Matrix W

- ….. Is very special:

$$\mathbf{w}_j^T \mathbf{y} = \sum_i W_{i,j}\, y_i = \sum_i \frac{K_{i,j}}{\sum_i K_{i,j}}\, y_i$$

$$\Longrightarrow \qquad \mathbf{W} = \mathbf{D}^{-1}\,\mathbf{K}, \qquad \text{where } \mathbf{D}_{jj} = \text{diag}\{\textstyle\sum_i K_{i,j}\}$$

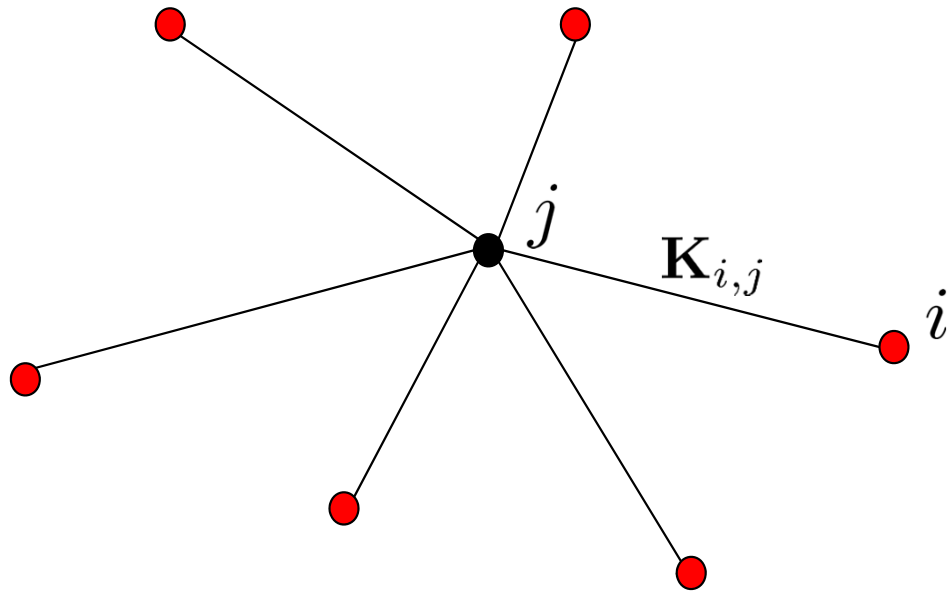
- W has real, positive, spectrum but is <u>not</u> symmetric
  - though it is almost ….. (more on this later)

Properties of W

- Key properties (Perron-Frobenius Thm.):
  - $\mathbf{W}$ is row-stochastic $(\mathbf{w}_j^T \mathbf{1} = 1)$
  - $\mathbf{W}$ has spectral radius $1 = \lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n \geq 0$
  - Dominant left, right eigen-vector: $\mathbf{v} = \frac{1}{\sqrt{n}}\mathbf{1}$ and $\mathbf{u} > 0$
  - Ergodicity: $\lim_{k \to \infty} \mathbf{W}^k = \mathbf{v}\mathbf{u}^T > 0$
- Repeatedly applying W gives a constant vector $c\mathbf{1}$.

Other Interpretations of W

- Transition Matrix for a Markov Chain
- Graphical Models
- Spectral Methods



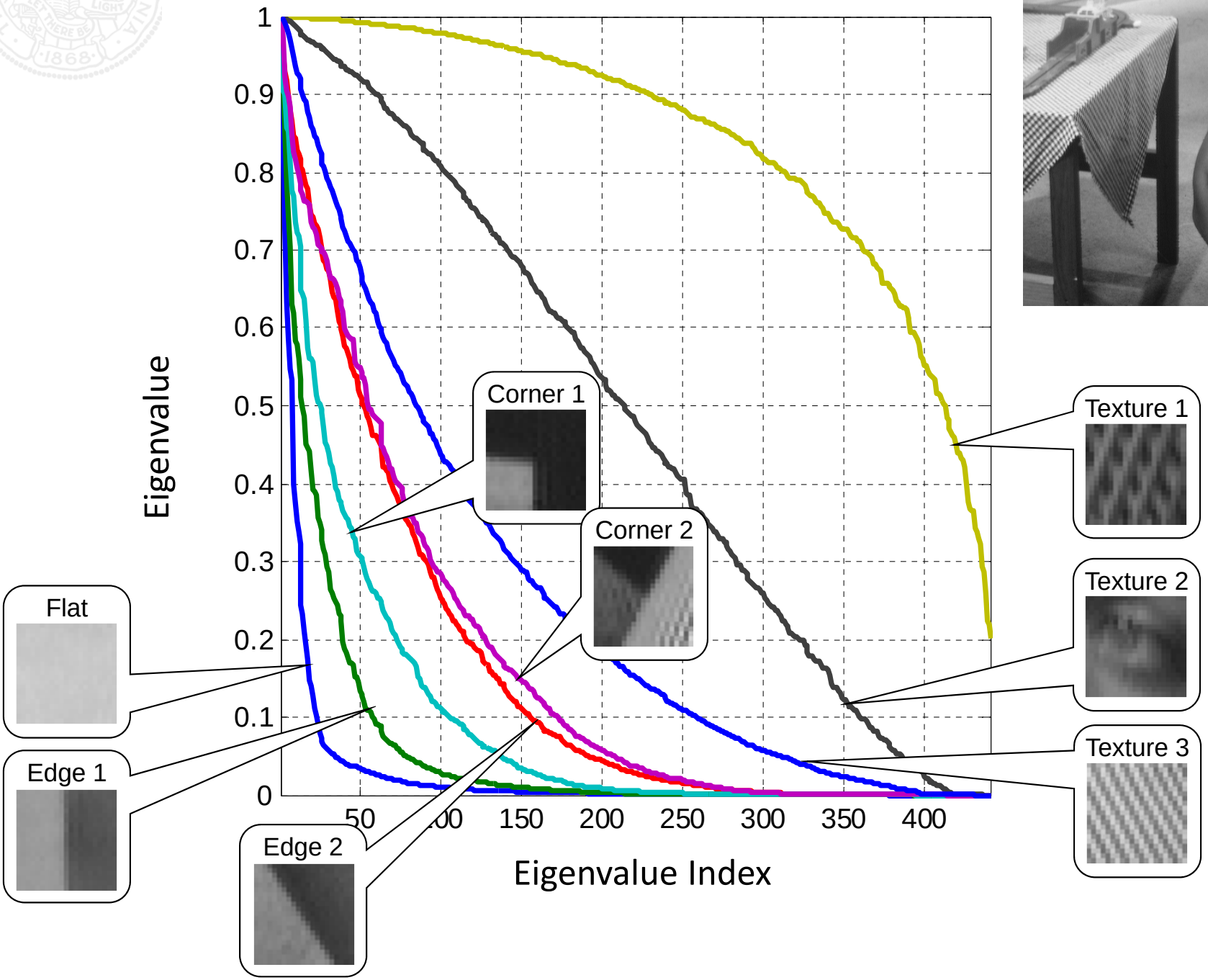
$$\mathbf{K}$$

$$\mathbf{W} = \mathbf{D}^{-1}\,\mathbf{K}$$

$$\mathcal{L} = \mathbf{D}^{1/2}\,(\mathbf{W} - \mathbf{I})\,\mathbf{D}^{-1/2}$$

"Graph Laplacian"

Spectra of the LARK Filter



Some "challenges" with $\hat{\mathbf{z}} = \mathbf{W}\,\mathbf{y}$

- Ad-hoc design
  - pick a kernel, generate W, filter
  - Adjust parameters, iterate, …..
- Asymmetry of W
  - Filters are "inadmissible" (Cohen '66)
  - No Bayesian interpretation (Hastie & Tibshirani '00)
  - No orthogonal decomposition
- We want to fix these …….

Remedy 1: Symmetrizing W

Algorithm 1 Diagonal scaling of $\mathbf{W}$

for $k = 1 : iter$;
  Normalize *Columns*
  Normalize *Rows*
end
$\mathbf{C} = \text{diag}(\mathbf{c})$; $\mathbf{R} = \text{diag}(\mathbf{r})$;
$\widehat{\mathbf{W}} = \mathbf{R}\,\mathbf{W}\,\mathbf{C}$

- Matrix Balancing
  - Sinkhorn and Knopp ('67)
- Iterative Proportional Scaling
  - Darroch and Ratcliff ('72)

- Convergence is guaranteed for non-negative $\mathbf{W}$
- $\widehat{\mathbf{W}}$ is symmetric, positive-definite, and doubly-stochastic

Is the Approximation Any Good?

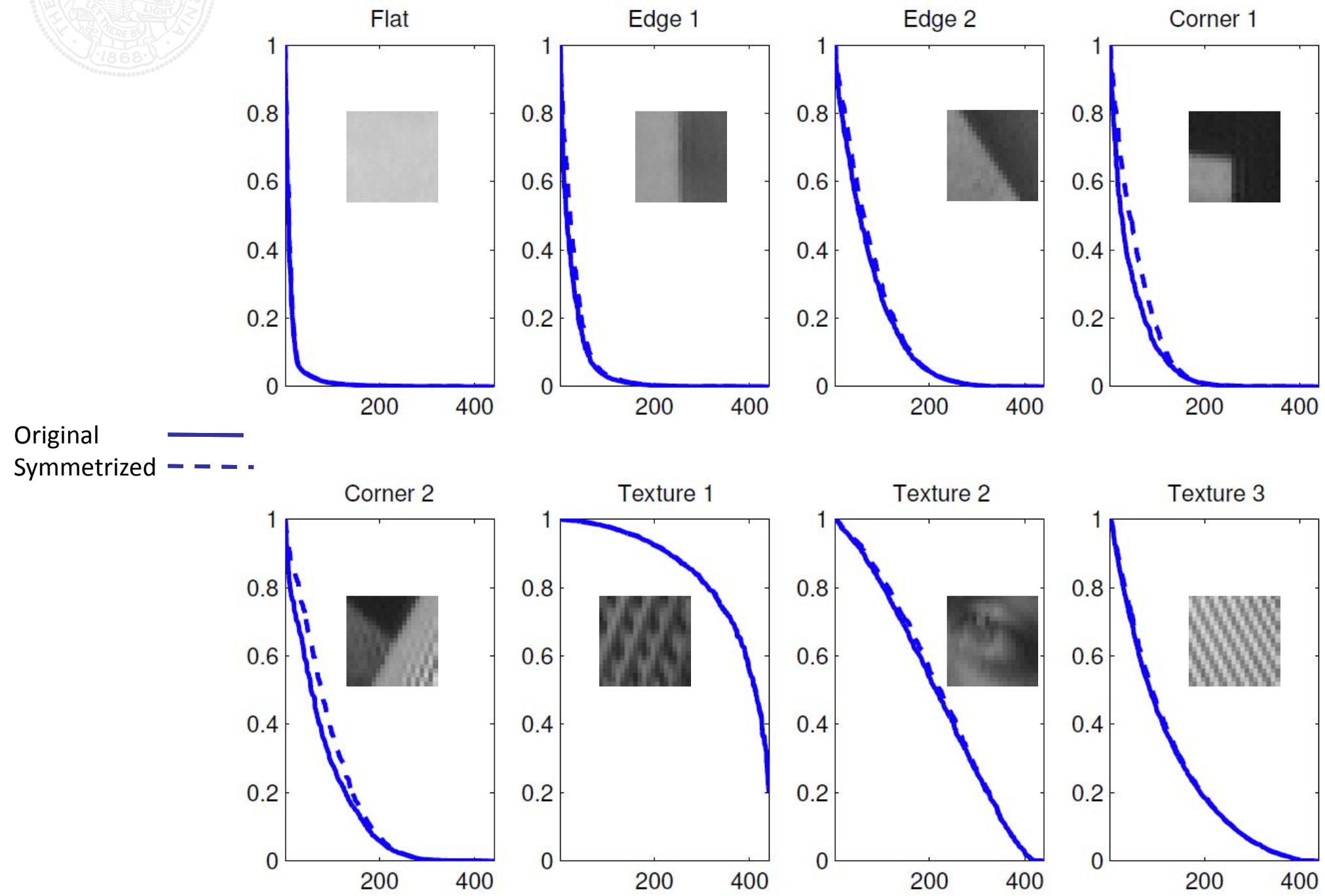
- Yes! Minimizes the cross-entropy:

$$\sum_{i,j} \widehat{\mathbf{W}}_{ij} \log \frac{\widehat{\mathbf{W}}_{ij}}{\mathbf{W}_{ij}}$$

- Smaller error with increasing dimension……

$$\underbrace{\frac{1}{n}\|\mathbf{W} - \widehat{\mathbf{W}}\|_F}_{\text{RMS difference in the elements}} = \mathrm{O}(n^{-1/2})$$

The Spectra


Flat
Edge 1
Edge 2
Corner 1
Corner 2
Texture 1
Texture 2
Texture 3
1
0.8
0.6
0.4
0.2
0
200
400
Original
Symmetrized

Some Benefits of Symmetry

$\mathbf{W}$

- Pixel Domain

$$\widehat{\mathbf{z}} = \mathbf{W}\,\mathbf{y}$$

(Nonlinear)
Spatially Adaptive Filter

$\widehat{\mathbf{W}}$

- Transform Domain

$$\widehat{\mathbf{z}}_s = \mathbf{V}\,\mathbf{S}\,\mathbf{V}^T\,\mathbf{y}$$

Shrinkage

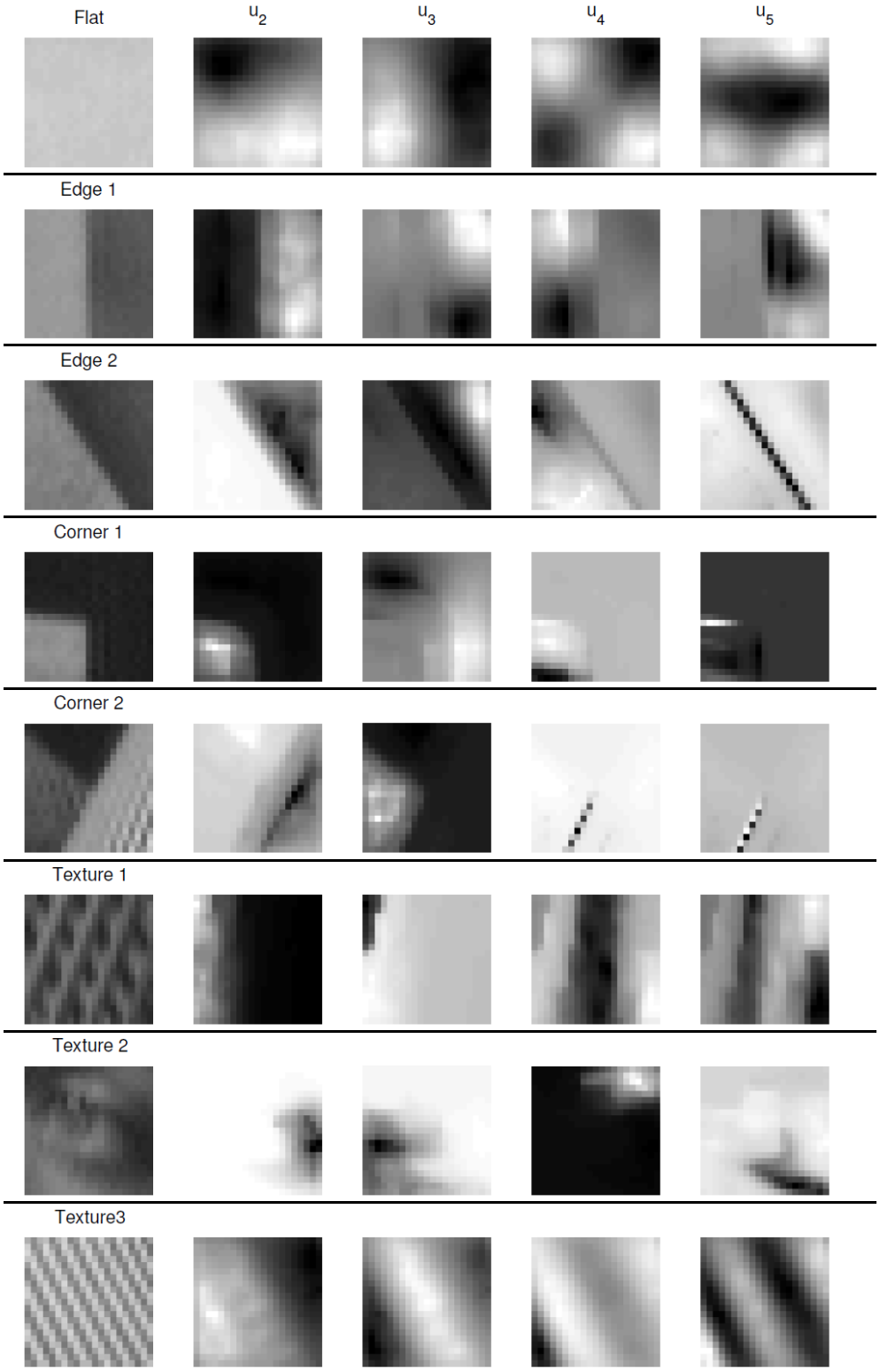
Orthonormal Transform

Other Advantages

- Performance Improvement
- Stability (iterative filtering)

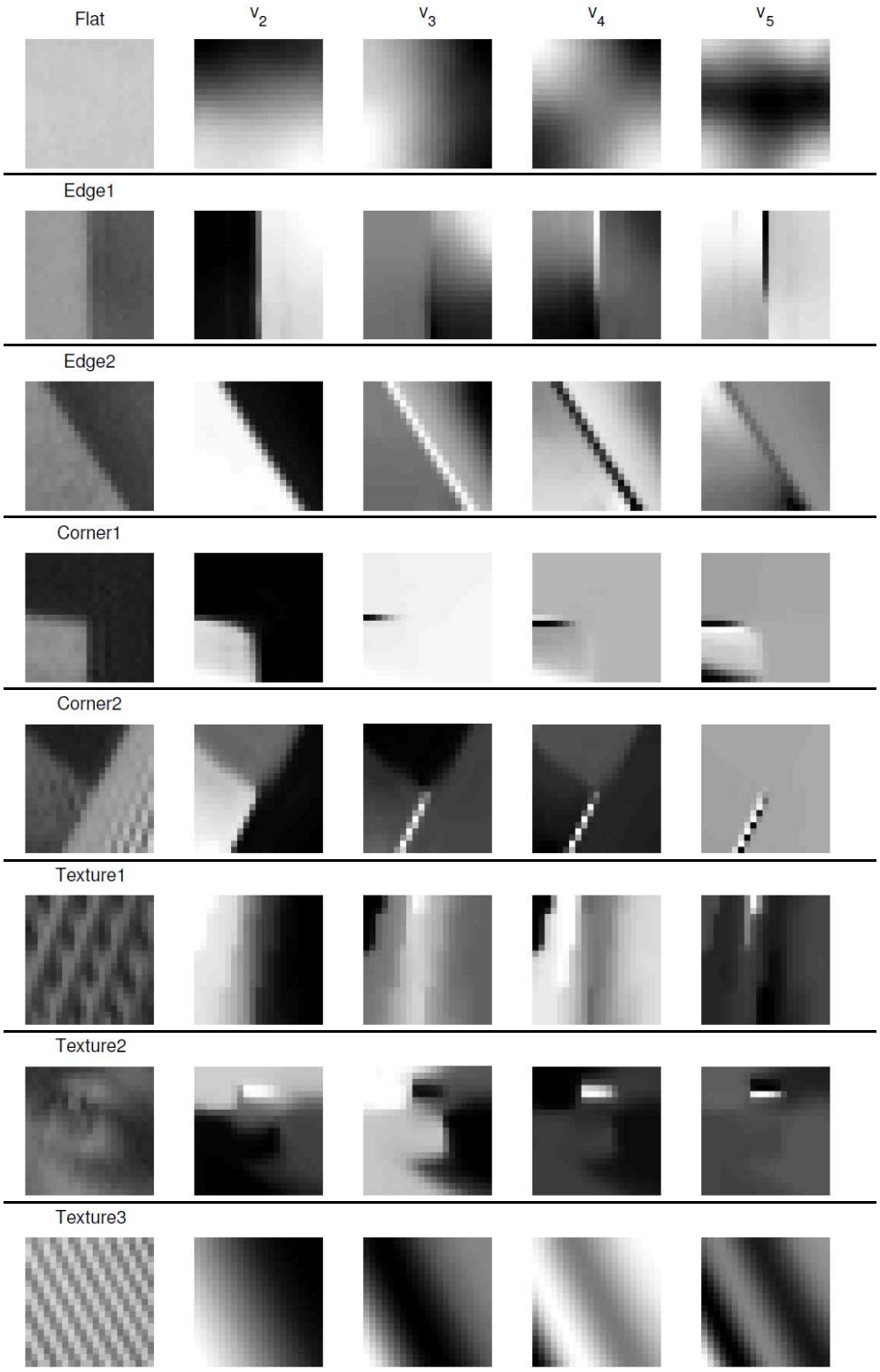
Dominant Eigenvectors of Original W

(Non Orthogonal Basis)



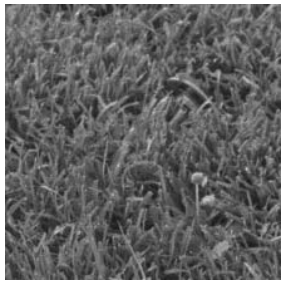
Dominant Eigenvectors of Symmetrized W

(Orthogonal Basis)



What's the Best We Can Do?

- For complex images, no room for improvement.
- For simpler images, room for improvement.



complex

simple

"Is Denoising Dead?" [Chatterjee, Milanfar, TIP 2010]

"Natural Image Denoising: Optimality and Inherent Bounds" [Levin, Nadler, CVPR 2011]

Performance Analysis (W Symm.)

- Spectral Decomposition: $\mathbf{W} = \mathbf{V}\mathbf{S}\mathbf{V}^T$

$$\text{where } \mathbf{S} = \text{diag}\left[\lambda_1, \cdots, \lambda_n\right]$$

$$0 \leq \lambda_n \leq \cdots \leq \lambda_1 = 1.$$

$$\mathbf{z} = \mathbf{V}\mathbf{b}$$

Signal coefficients in the basis given by eigenvectors of W

$$\mathbf{MSE} = \sum_{i=1}^{n} \underbrace{(\lambda_i - 1)^2 b_i^2}_{\text{Bias}^2} + \underbrace{\sigma^2 \lambda_i^2}_{\text{Variance}}$$

An Observation

- What is the “ideal” (oracle) spectrum for W?

- Minimize the Mean-Squared Error w.r.t. λ_i

$$\mathbf{MSE}(\lambda_i) = \sum_{i=1}^{n} (\lambda_i - 1)^2 b_i^2 + \sigma^2 \lambda_i^2$$

- Optimal Spectrum:

$$\lambda_i^* = \frac{b_i^2}{b_i^2 + \sigma^2} = \frac{1}{1 + \mathbf{snr}_i^{-1}}$$

← “Ideal” Wiener Filter

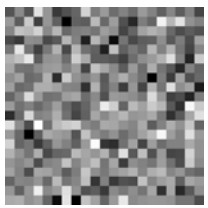
- Explains performance of state of the art denoising

Worst Case Performance

- The oracle MSE: $$\mathbf{MSE}_{min} = \sigma^2 \sum_{i=1}^{n} \frac{b_i^2}{b_i^2 + \sigma^2}$$
- Maximize this over all signals:

$$\max_{\mathbf{b}} \ \mathbf{MSE}_{min} \quad \text{subject to} \quad \mathbf{b}^T\mathbf{b} = 1$$

- Hardest Patches to Denoise: $$b_i^2 = \frac{1}{n}$$
  - flat + “white noise”



Remedy 2: Iterations

- Diffusion (Perona-Malik '90, Coifman et al. '06, ….)

$$\widehat{\mathbf{z}}_k = \mathbf{W}\,\widehat{\mathbf{z}}_{k-1}$$

$$\widehat{\mathbf{z}}_k - \widehat{\mathbf{z}}_{k-1} = (\mathbf{W} - \mathbf{I})\,\widehat{\mathbf{z}}_{k-1}$$

$$\frac{\partial \mathbf{z}}{\partial t}$$

$$\widehat{\mathbf{z}}_k - \widehat{\mathbf{z}}_{k-1} = \left[\mathbf{D}^{-1/2}\,\mathcal{L}\,\mathbf{D}^{1/2}\right]\,\widehat{\mathbf{z}}_{k-1}$$

Laplacian

$$\nabla^2 \mathbf{z}$$

- Performance:

$$\mathbf{MSE}_k = \sum_{i=1}^{n} \underbrace{(\lambda_i^k - 1)^2 b_i^2}_{\text{Bias}} + \underbrace{\sigma^2 \lambda_i^{2k}}_{\text{Variance}}$$

Remedy 2: Iterations again

- Twicing (Tukey '77), (L_2-) Boosting (Buhlmann, Yu '03), Bregman Iteration (Osher et al. '05), Reaction-Diffusion (Nordstrom '90)

$$\widehat{\mathbf{z}}_k = \widehat{\mathbf{z}}_{k-1} + \mathbf{W}\underbrace{(\mathbf{y} - \widehat{\mathbf{z}}_{k-1})}$$

Residuals

- Example:

$$\begin{aligned} \widehat{\mathbf{z}}_1 &= \widehat{\mathbf{z}}_0 + \mathbf{W}(\mathbf{y} - \widehat{\mathbf{z}}_0) \\ &= \mathbf{W}\mathbf{y} + \mathbf{W}(\mathbf{y} - \mathbf{W}\mathbf{y}) \\ &= \underbrace{(2\mathbf{I} - \mathbf{W})}\ \underbrace{\mathbf{W}\,\mathbf{y}} \end{aligned}$$

Sharpening (inv. diffusion) step

Blurring (diffusion) step

$$\mathbf{MSE}_k = \sum_{i=1}^{n} \underbrace{(1-\lambda_i)^{2k+2} b_i^2}_{\text{Bias}} + \underbrace{\sigma^2 \left(1-(1-\lambda_i)^{k+1}\right)^2}_{\text{Variance}}$$

Bias ⇩ Variance ⇧

Statistical Performance Analysis

- Diffusion

$$\mathbf{MSE}_k = \sum_{i=1}^{n} \underbrace{(\lambda_i^k - 1)^2 b_i^2}_{\text{Bias} \uparrow} + \underbrace{\sigma^2 \lambda_i^{2k}}_{\text{Variance} \downarrow}$$

- Residual

$$\mathbf{MSE}_k = \sum_{i=1}^{n} \underbrace{(1-\lambda_i)^{2k+2} b_i^2}_{\text{Bias} \downarrow} + \underbrace{\sigma^2 \left(1 - (1-\lambda_i)^{k+1}\right)^2}_{\text{Variance} \uparrow}$$

Which to Use?

- Depends on (1) the filter, (2) the latent image, and (3) the noise level.

- Diffusion if W "under-smooths"
  - Doesn't do much denoising

- Residual if W "over-smooths"
  - Signal is left in residuals
  - *Weak learner* in boosting

Bayesian Interpretation

- Regularization

$$\widehat{\mathbf{z}} \quad = \quad \arg\min_{\mathbf{z}} \; \frac{1}{2}\|\mathbf{y}-\mathbf{z}\|^2 + \frac{\lambda}{2}\mathcal{R}(\mathbf{y},\mathbf{z})$$

Empirical log-Prior

- Steepest Descent Iteration:

$$\widehat{\mathbf{z}}_k \quad = \quad \widehat{\mathbf{z}}_{k-1} - \mu \underbrace{\left[(\widehat{\mathbf{z}}_{k-1}-\mathbf{y}) + \lambda \nabla \mathcal{R}(\mathbf{y},\mathbf{z}_{k-1})\right]}_{\text{Gradient}}$$

Step size

Given W, what is R?

1. MAP SD: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k - \mu\left[(\widehat{\mathbf{z}}_k - \mathbf{y}) + \lambda \nabla \mathcal{R}(\widehat{\mathbf{z}}_k)\right]$

2. Residuals: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k + \mathbf{W}(\mathbf{y} - \widehat{\mathbf{z}}_k)$

3. Diffusion: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k + (\mathbf{W} - \mathbf{I})\,\widehat{\mathbf{z}}_k$

1 2 $\nabla \mathcal{R}(\mathbf{z}_k) = \dfrac{-1}{\mu\lambda}(\mathbf{W} - \mu\mathbf{I})(\mathbf{y} - \widehat{\mathbf{z}}_k)$

1 3 $\nabla \mathcal{R}(\mathbf{z}_k) = \dfrac{1}{\mu\lambda}(\mathbf{W} - (1-\mu)\mathbf{I})(\mathbf{y} - \widehat{\mathbf{z}}_k) - \dfrac{1}{\mu\lambda}(\mathbf{I} - \mathbf{W})\,\widehat{\mathbf{y}}$

Given W, what is R?

1. MAP SD: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k - \mu\left[(\widehat{\mathbf{z}}_k - \mathbf{y}) + \lambda \nabla \mathcal{R}(\widehat{\mathbf{z}}_k)\right]$

2. Residuals: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k + \mathbf{W}(\mathbf{y} - \widehat{\mathbf{z}}_k)$

3. Diffusion: $\widehat{\mathbf{z}}_{k+1} = \widehat{\mathbf{z}}_k + (\mathbf{W} - \mathbf{I})\,\widehat{\mathbf{z}}_k$

○○ $\mathcal{R}(\mathbf{z}_k) = \frac{1}{2\mu\lambda}(\mathbf{y} - \widehat{\mathbf{z}}_k)^T (\mathbf{W} - \mu\mathbf{I}) \underbrace{(\mathbf{y} - \widehat{\mathbf{z}}_k)}_{\text{residuals}}$ With symmetric W !!

○○ $\mathcal{R}(\mathbf{z}_k) = \frac{1}{2\mu\lambda}(\mathbf{y} - \widehat{\mathbf{z}}_k)^T ((1-\mu)\mathbf{I} - \mathbf{W})(\mathbf{y} - \widehat{\mathbf{z}}_k) + \frac{1}{\mu\lambda}\mathbf{y}^T (\mathbf{I} - \mathbf{W})\,\widehat{\mathbf{z}}_k$

Data-dependent "Priors"

Simplify…… $\widehat{\mathbf{z}}_k = \mathbf{z} \quad \text{and} \quad \mu = 1$

$$\widehat{p}(\mathbf{z}) = c \exp\left[-\widehat{\mathcal{R}}(\mathbf{z})\right]$$

- Residuals:

$$\widehat{p}(\mathbf{z}) = c \exp\left[-\frac{1}{2}\mathbf{z}^T(\mathbf{W}-\mathbf{I})\mathbf{z} + \mathbf{y}^T(\mathbf{W}-\mathbf{I})\mathbf{z}\right]$$

- Diffusion:

$$\widehat{p}(\mathbf{z}) = c \exp\left[-\frac{1}{2}\mathbf{z}^T\mathbf{W}\mathbf{z} + \mathbf{y}^T\mathbf{z}\right]$$

Data-dependent

Conclusions

- The $\widehat{\mathbf{z}} = \mathbf{W}\ \mathbf{y}$ framework is very general.
- Kernel filters can almost always be improved.
- Many open problems remain
  - Local models for signal
  - W negative;
  - Kernelizing Bayesians
  - Are patches the best way forward?