Background
Improved results can be achieved by processing the residual/method-noise image:

Noisy image $y$

Denoised image $\hat{x} = f(y)$

Method Noise $y - \hat{x}$
Twicing [Tukey (’77), Charest et al. (’06)]

\[ \hat{x}^{k+1} = \hat{x}^k + f(y - \hat{x}^k) \]

Method noise whitening [Romano & Elad (‘13)]

Recovering the “stolen” content from the method-noise using the same basis elements that were chosen to represent the initially denoised patches.

TV denoising using Bregman distance [Bregman (‘67), Osher et al. (’05)]

\[ \hat{x}^{k+1} = f(\hat{x}^k + \sum_{i=1}^{k} (y - \hat{x}^i)) \]
Boosting Methods

- **Diffusion** [Perona-Malik (’90), Coifman et al. (’06), Milanfar (’12)]
  - Removes the noise leftovers that are found in the denoised image
  - $\hat{x}^{k+1} = f(\hat{x}^k)$

- **SAIF** [Talebi et al. (’12)]
  - Chooses automatically the local improvement mechanism:
    - Diffusion
    - Twicing
Reducing the Local/Global Gap

- **EPLL** [Zoran & Weiss (’09), Sulam & Elad (’14)]

  - Treats a major shortcoming of patch-based methods:
    - The gap between the **local patch processing** and the **global need** for a whole restored image
  - By encouraging the patches of the final image (i.e. after patch aggregation) to comply with the local prior
  - In practice – iterated denoising with a diminishing variance

  I. Denoising the **patches** of $\hat{X}^k$

  II. Obtain $\hat{X}^{k+1}$ by **averaging** the overlapping patches and the noisy image
Boosting of Image Denoising Algorithms
SIAM Journal on Imaging Sciences, 2015
Given any denoiser, how can we improve its performance?
Given any denoiser, how can we improve its performance?

I. **Strengthen** the signal
II. **Operate** the denoiser
Given any denoiser, how can we improve its performance?

I. **Strengthen** the signal

II. **Operate** the denoiser

III. **Subtract** the previous estimation from the outcome

**SOS formulation:** \( \hat{x}^{k+1} = f(y + \hat{x}^k) - \hat{x}^k \)
Strengthen - Operate - Subtract Boosting

- An improvement is obtained since \( \text{SNR}\{y + \hat{x}\} > \text{SNR}\{y\} \)
  - In the ideal case, where \( \hat{x} = x \), we get
    \[
    \text{SNR}\{y + x\} = 2 \cdot \text{SNR}\{y\}
    \]

- We suggest strengthening the underlying signal, rather than
  - Adding the residual back to the noisy image
    - Twicing converges to the noisy image
  - Filtering the previous estimate over and over again
    - Diffusion could lead to over-smoothing, converging to a piece-wise constant image
In order to study the convergence of the SOS function, we represent the denoiser in its matrix form:

$$\hat{x} = f(y) = W y$$

The properties of $W$:

- Kernel-based methods (e.g. Bilateral filter, NLM, Kernel Regression) can be approximated as row-stochastic positive definite matrices [Milanfar ('13)]
  - Has eigenvalues in the range $[0,...,1]$

- What about sparsity-based methods?
  - We have showed that it also has eigenvalues in the range $[0,...,1]$ (and other interesting properties)
Convergence Study

- The SOS recursive function converges if \[ \|I - W\|_2 < 1 \]
  - Holds both for kernel-based (Bilateral filter, NLM, Kernel Regression), and sparsity-based methods (K-SVD)

For most denoising algorithms the SOS boosting is “guaranteed” to converge to

\[ \hat{x}^* = \left( I + (I - W) \right)^{-1} Wy \]

- The condition for convergence is alleviated in the generalized SOS version (more can be found in our paper)

SOS formulation: \[ \hat{x}^k = W_k(y + \hat{x}^{k-1}) - \hat{x}^{k-1} \]
Reducing the “Local-Global” Gap

Patch-Disagreement as a Way to Improve K-SVD Denoising
ICASSP, 2015
Reaching a Consensus

- It turns out that the SOS boosting reduces the local/global gap, which is a shortcoming of many patch-based methods:
  - Local processing of patches VS. the global need in a whole denoised result

- We define the local disagreements by:
  - Naturally exist since each noisy patch is denoised independently
  - Are based on the intermediate results

Boosting of Image Denoising Algorithms
By Yaniv Romano and Michael Elad
“Sharing the Disagreement”

Inspired by the “Consensus and Sharing” problem from game-theory:

- There are several agents, each one of them aims to minimize its individual cost (i.e., representing the noisy patch sparsely)
- These agents affect a shared objective term, describing the overall goal (i.e., obtaining the globally denoised image)

Imitating this concept, we suggest sharing the disagreements.
Connection to SOS Boosting

- Interestingly, for a fixed filter matrix $W$, “sharing the disagreement” and the SOS boosting are equivalent:

$$\hat{x}^{k+1} = W(y + \hat{x}^k) - \hat{x}^k$$

- The connection to the SOS is far from trivial because:
  - The SOS is blind to the intermediate results (the independent denoised patches, before patch-averaging)
  - The intermediate results are crucial for “sharing the disagreement” approach

The SOS boosting reduces the Local/Global gap
Graph-Based Interpretation
Graph-Based Analysis

- Essentially, the filter-matrix $W$ is an adaptive filter, where the $i^{th}$ denoised pixel is obtained by

$$\hat{x}_i = \sum_j W_{i,j} y$$

- $W_{i,j}$ measures the similarity between the $i^{th}$ and $j^{th}$ pixels
  - A large value implies large similarity
  - A small value implies small similarity

- For kernel-based methods (NLM, Bilateral, LARK): $W_{i,j}$ is a function of the Euclidean distance between pixels

- For sparsity-based methods (K-SVD): $W_{i,j}$ is a function of the dictionary atoms that were chosen to represent the patch
  - Measures the affinity between pixels, through the dictionary
The normalized Graph Laplacian can be defined as

$$\mathcal{L} = I - W$$

- Encapsulates the structure of underlying signal
  - Most of the image content is represented by the eigenvectors that correspond to the small eigenvalues
  - Most of the noise is represented by eigenvectors that correspond to the large eigenvalues

What can we do with $\mathcal{L}$?

- Regularize the inverse problem by encouraging similar pixels to remain similar in the final estimate
Graph Laplacian Regularization

- The regularization can be defined as [Elmoataz et al. (’08), Bougleux et al. (’09)]

\[
\hat{x} = \min_x \|x - y\|^2_2 + \rho x^T \mathcal{L}x
\]

Seeks for an estimation that is close to the noisy version

While promoting similar pixels to remain similar

- Another option is to integrate the filter also in the data fidelity term [Kheradmand and Milanfar (’13)]

\[
\hat{x} = \min_x (x - y)^T \mathbf{W} (x - y) + \rho x^T \mathcal{L}x
\]

Using the adaptive filter as a weight-matrix
Graph Laplacian Regularization

- Another natural option is to minimize the following cost function
  \[
  \hat{x}^* = \min_x \|x - W y\|_2^2 + \rho x^T \mathcal{L} x
  \]

  Seeks for an estimation that is close to the denoised version

  - Its closed-form solution is the steady-state outcome of the SOS
    \[
    \hat{x}^* = \left( \mathbf{I} + \rho (\mathbf{I} - \mathbf{W}) \right)^{-1} W y = (\mathbf{I} + \rho \mathcal{L})^{-1} W y
    \]

  The SOS boosting acts as a graph Laplacian regularizer

- We have also expressed the previous cost functions in the "SOS language"... more can be found in our paper
Experiments
Results

- We successfully boost several state-of-the-art denoising algorithms:
  - K-SVD, NLM, BM3D, and EPLL
  - Without any modifications, simply by applying the original software as a “black-box”

- We manually tuned two parameters
  - $\rho$ – signal emphasis factor
  - $\sigma$ – noise level, which is an input to the denoiser
    - Since the noise level of $y + \rho x^k$ is higher than the one of $y$
Quantitative Comparison

- Average boosting in PSNR* over 5 images (higher is better):

<table>
<thead>
<tr>
<th>Noise std</th>
<th>Improved Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>K-SVD</td>
</tr>
<tr>
<td>$\sigma$</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.13</td>
</tr>
<tr>
<td>20</td>
<td>0.22</td>
</tr>
<tr>
<td>25</td>
<td>0.26</td>
</tr>
<tr>
<td>50</td>
<td>0.77</td>
</tr>
<tr>
<td>75</td>
<td>1.26</td>
</tr>
<tr>
<td>100</td>
<td>0.81</td>
</tr>
</tbody>
</table>

$\text{PSNR} = 20\log_{10}\left(\frac{255}{\sqrt{\text{MSE}}}\right)$
Visual Comparison: K-SVD

- Original K-SVD results, $\sigma = 25$

29.06dB
Visual Comparison: K-SVD

- **SOS** K-SVD results, $\sigma = 25$

29.41dB
Visual Comparison: EPLL

- Original EPLL results, $\sigma = 25$

Forman: 32.44dB
House: 32.07dB
Visual Comparison: EPLL

- **SOS EPLL results, \( \sigma = 25 \)**

**Forman**
- 32.78 dB

**House**
- 32.38 dB
The SOS boosting algorithm:

- Easy to use
  - In practice, we treat the denoiser $f(\cdot)$ as a “black-box”
- Applicable to a wide range of denoising algorithms
- Guaranteed to converge for most denoising algorithms
  - Thus, has a straightforward stopping criterion
- Reduces the local-global gap
- Acts as a graph Laplacian regularizer
- Improves the state-of-the-art methods
We are Done...

Thank you!

Questions?