Contents

- Motivation
- Processing quality criteria and MSE optimal linear filters
- Sliding Window Transform Domain (SWTD) Filters
- Selection of the transform
- Wavelet (WL) shrinkage
- Image denoising with SWTD and WL filtering: comparison
- A hybrid SWTD/WL processing
- SWTD and WL processing: a unified interpretation
- Conclusion and future plans
CRITERIA OF PROCESSING QUALITY:

$$AVLOSS(k,l) = AV\left\{\sum_{m,n} LOC(m,n;a_{k,l})LOSS(\hat{a}_{m,n},a_{m,n})\right\}$$

\(\{m,n\}\) - pixel coordinates in image plane; \(\{k,l\}\) - coordinates of the current pixel.
\(\{a_{m,n}\}\) - vector of pixels’ “true” values; \(\{\hat{a}_{m,n}\}\) - vector of pixels’ estimated values.

\(LOSS(\hat{a}_{m,n},a_{m,n})\) - loss function; measures “losses” due to the estimation error.

\(LOC(m,n;a_{k,l})\) - “locality” function that defines pixel’s neighborhood \(NBH\):

\[LOC(m,n;a_{k,l}) = \begin{cases} W(m,n;k,l) > 0, & \text{if } \{m,n\} \in NBH(k,l) \\ 0, & \text{otherwise} \end{cases}\]

\(AV\) - a statistical averaging operator

Let \(b = \{b_{m,n}\}\) - vector of observed image pixels and \(M(b \rightarrow \hat{a})\) is a processing algorithm.
Optimal processing algorithm is the algorithm that provides

\(\hat{a}_{k,l}^{opt} = \arg\min_{M(b \rightarrow \hat{a})} \left\{AV\left\{\sum_{m,n} LOC(m,n;a_{k,l})LOSS(\hat{a}_{m,n},a_{m,n})\right\}\right\}\)
MSE OPTIMAL LINEAR FILTERS

Loss function: 
\[ \text{LOSS}(\hat{a}_{m,n}, a_{m,n}) = |\hat{a}_{m,n} - a_{m,n}|^2 \]

Locality function: 
\[ \text{LOC}(m,n; a_{k,l}) = \begin{cases} d_{k-m,l-n} & |k-m| \leq M_1; |l-n| \leq M_2, \\ 0, & \text{otherwise} \end{cases} \]

For a general linear filter defined by a matrix \( H \): \( \hat{a} = Hb \)

\[ AV\text{LOSS}(k,l) = AV\{Hb - a\}^T D (Hb - a) = AV\|H_w D^{1/2} b - D^{1/2} a\|; \]
\[ D = \text{diag}\{ d_{k,l} \} \]

Optimal filter is the filter that provides

\[ \hat{a}_{opt} = \arg \min_H \{ AV\|H_w D^{1/2} b - D^{1/2} a\| \} \]
Transform Domain Linear Scalar Filters

Scalar filtering ($H = T^{-1}H_d T$): $$\hat{a}_w = T^{-1}H_d T \hat{b}_w,$$

where $T$ and $T^{-1}$ are matrices of the direct and inverse orthogonal transforms, and $H_d = \text{diag}\{\eta_r\}$,

By virtue of the Parceval’s relation: $AVLOSS(k,l) = AV\|\hat{T}_w - T \hat{a}_w\|$

For scalar linear filters and $\alpha_r = T \hat{a}_w$, $\hat{\alpha}_r = T \hat{\alpha}_w$, $\beta_r = T \hat{b}_w$

$$\hat{\alpha}_r = \eta_r \beta_r$$

$$AVLOSS(k,l) = AV\sum_{r=0}^{N-1}|\alpha_r - \eta_r \beta_r|^2$$

By minimizing $AVLOSS(k,l)$ with respect to $\{\eta_r\}$ one can find

$$\eta_r^{opt} = \frac{AV(\alpha_r \beta_r^*)}{AV(\beta_r^2)} \quad (\ast \text{ for complex conjugate})$$
Filters for image deblurring and denoising (signal independent noise):

\[
\eta_{r^{\text{opt}}} = \begin{cases} 
\max \left( AV \left( |\beta_r|^2 \right) - \text{thr} , 0 \right) / \lambda_r AV \left( |\beta_r|^2 \right) , & \lambda_r \neq 0 \\
0 , & \lambda_r = 0
\end{cases}
\]

Filters for image deblurring and denoising (signal dependent noise):

\[
\eta_{r^{\text{opt}}} = \begin{cases} 
\max \left( AV \left( |\beta_r|^2 \right) - \text{thr} \cdot (\beta_0)^p , 0 \right) / \lambda_r AV \left( |\beta_r|^2 \right) , & \lambda_r \neq 0 \\
0 , & \lambda_r = 0
\end{cases}
\]

Rejective filters:

\[
\eta_{r^{\text{opt}}} = \begin{cases} 
1 / \lambda_r , & \text{if } AV \left( |\beta_r|^2 \right) > \text{thr} \\
0 , & \text{otherwise}
\end{cases}
\]

\[\eta_0 = 1\]

“Fractional” spectrum filter:

\[
\eta_r = \begin{cases} 
|\beta_r|^{p-1} , & \text{if } AV \left( |\beta_r|^2 \right) > \text{thr} , r = 1, \ldots, N - 1 \\
0 , & \text{otherwise}
\end{cases}
\]
IMAGE PROCESSING: LOCAL VERSUS GLOBAL
Sliding window filters are designed in a transform domain in a running window and operate, in each position $(k,l)$ of the window, in the following 3 steps:

1. **Direct Transform**
   \[ \{ \beta_{r,s}^{(k,l)} \} = T b_{m,n} \]

2. **Filtering**
   \[ \hat{\alpha}_{r,s}^{(k,l)} = \eta_{r,s}^{(k,l)} \beta_{r,s}^{(k,l)} \]

3. **Inverse Transform**
   \[ \{ \hat{a}_{m,n}^{(k,l)} \} = T^{-1} \hat{\alpha}_{r,s}^{(k,l)} \]

**Sliding window transform domain (SWTD) filters**
Sliding window transform domain (SWTD) filters
The selection of the transform for the filter implementation is governed by

- A priori knowledge about image spectra in the transform domain
- Accuracy of empirical spectrum estimation
- Transform energy compaction capability
- Computational complexity of the filtering in the transform domain

Feasible transforms: DFT, DCT, DST, LOT, Haar, Walsh, wavelets.

2-D basis functions of (left to right) DCT, Walsh, Haar Transforms
Odds of DCT:

- Good energy compaction capability
- Good suitability for signal/image restoration tasks with signaling and imaging system specification in terms of their frequency responses
- Suitability for multi-component signal/image processing
- Low computational complexity: recursive computing SWDCT is possible with the complexity of $O(\text{number of coefficients required})$
Transform energy compaction capability: comparison

In most cases DCT outperforms other transforms in terms of energy compaction capability.
Sliding window DCT (odd window size)


Direct transform:
\[
\alpha_r^{(k)} = \frac{1}{\sqrt{2N_w}} \sum_{n=0}^{N_w-1} a_{k+n-\text{fix}(N/2)} \cos \left( \pi \frac{(n + 1/2)}{N} r \right)
\]

Inverse transform for the window central pixel:
\[
a_k = \frac{1}{\sqrt{2N}} \left\{ \alpha_0^k + 2 \sum_{r=1}^{N_w-1} \alpha_r^{(k)} \cos \left( \pi \frac{\text{fix}(N/2)+1/2}{N} r \right) \right\}
\]

For odd N:
\[
a_k = \frac{1}{\sqrt{2N_w}} \left\{ \alpha_0^k + 2 \sum_{r=1}^{(N_w-1)/2} \alpha_r^{(k)} (-1)^r \right\}
\]

Therefore, only DCT coefficients with even indices are involved.
For adjacent $k$-th and $(k+1)$-th window positions, DCT spectra $\{\alpha_r^{(k)}\}$ and $\{\alpha_r^{(k+1)}\}$ are

$$\alpha_r^{(k)} = \frac{1}{\sqrt{2N_w}} \sum_{n=0}^{N_w-1} a_{k+n-\text{fix}(N_w/2)} \cos\left(\pi \frac{(n+1/2)r}{N}\right)$$

and

$$\alpha_r^{(k+1)} = \sum_{n=0}^{N_w-1} a_{k+1+n-\text{fix}(N_w/2)} \cos\left(\pi \frac{(n+1/2)r}{N}\right).$$

Introduce auxiliary spectra

$$\tilde{\alpha}_r^{(k)} = \sum_{n=0}^{N_w-1} a_{k+n-\text{fix}(N_w/2)} \exp\left(i\pi \frac{(n+1/2)r}{N}\right)$$

Spectrum $\{\tilde{\alpha}_r^{(k+1)}\}$ can be represented through spectrum $\{\tilde{\alpha}_r^{(k)}\}$:

$$\tilde{\alpha}_r^{(k+1)} = \tilde{\alpha}_r^{(k)} \exp\left(-i\pi r / N_w\right) + \left[a_{k+N_w-\text{fix}(N_w/2)}(-1)^r - a_{k-\text{fix}(N_w/2)}\right] \exp\left(-i\pi r / 2N\right).$$

Therefore, auxiliary spectra $\{\tilde{\alpha}_r^{(k)}\}$ can be computed recursively and local DCT spectra $\{\alpha_r^{(k)}\}$ can then be found from the relationship:

$$\alpha_r^{(k)} = \text{real}(\tilde{\alpha}_r^{(k)}).$$
SWTD DCT filtering for image restoration

SWTD DCT filtering for image blind deblurring
SWTD DCT filtering speckle noise

Leonid P. Yaroslavsky, Ben-Zion Shaick
Transform Oriented Image Processing
Technology for Quantitative Analysis of
Fetal Movements in Ultrasound Image Sequences.
In: Signal Processing IX. Theories and Applications,
Sliding Window DCT 3x3

Involved transform basis functions:

\[
\begin{bmatrix}
DCT_{00} & DCT_{20} \\
DCT_{02} & DCT_{22}
\end{bmatrix} = \begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{bmatrix} \begin{bmatrix}
1 & -2 & 1 \\
1 & -2 & 1 \\
1 & -2 & 1
\end{bmatrix};
\]

Basis functions \(DCT_{20}, DCT_{02}, \) and \(DCT_{22}\) are kernels of 2-D Laplacians: horizontal, vertical and isotropic ones. One can further decompose basis function \(DCT_{22}\) into a sum of two diagonal Laplacians:

\[
DCT_{22} = \begin{bmatrix}
2 & -1 & -1 \\
-1 & 2 & -1 \\
-1 & -1 & 2
\end{bmatrix} + \begin{bmatrix}
-1 & -1 & 2 \\
-1 & 2 & -1 \\
2 & -1 & -1
\end{bmatrix}
\]
SWTD DCT3x3: Restoration of high resolution satellite images

## Potentials of SWTD DCT filtering

### Standard. deviation of residual noise for 3 test noisy images


<table>
<thead>
<tr>
<th>Test image</th>
<th>Initial noise level</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15</td>
</tr>
<tr>
<td>Aero1</td>
<td>Rej. Fltr</td>
</tr>
<tr>
<td></td>
<td>Emp. Wiener</td>
</tr>
<tr>
<td></td>
<td>Av.Rej.Fltr</td>
</tr>
<tr>
<td></td>
<td>Id.Wiener</td>
</tr>
<tr>
<td>Aero2</td>
<td>Rej. Fltr</td>
</tr>
<tr>
<td></td>
<td>Emp. Wiener</td>
</tr>
<tr>
<td></td>
<td>Av.Rej.Fltr</td>
</tr>
<tr>
<td></td>
<td>Id.Wiener</td>
</tr>
<tr>
<td>Lenna</td>
<td>Rej. Fltr</td>
</tr>
<tr>
<td></td>
<td>Emp. Wiener</td>
</tr>
<tr>
<td></td>
<td>Av.Rej.Fltr</td>
</tr>
<tr>
<td></td>
<td>Id.Wiener</td>
</tr>
</tbody>
</table>
Wavelet Shrinkage

Input

Low pass filtering and downsampling

Interpolation

Soft/hard thresholding

Interpolation

Low pass filtering and downsampling

Interpolation

Soft/hard thresholding

Interpolation

Output
Empirical Wiener Denoising Filter

\[ \eta_{r}^{\text{opt}} = \frac{\max(\beta_r^2 - |v_r|^2, 0)}{|\beta_r|^2} \]

Rejective filter:

\[ \eta_{r}^{\text{opt}} = \begin{cases} 1, & \text{if } |\beta_r|^2 > \text{thr} \\ 0, & \text{otherwise} \end{cases} \]

Wavelet Shrinkage Filter (soft threshold)

\[ \eta_{r}^{\text{opt}} = \frac{\max(|\beta_r| - \text{thr}, 0)}{|\beta_r|} \]

Wavelet Shrinkage Filter (hard threshold)

\[ \eta_{r}^{\text{opt}} = \begin{cases} 1, & \text{if } |\beta_r| > \text{thr} \\ 0, & \text{otherwise} \end{cases} \]
## Wavelet vs SWDCT denoising: comparison


<table>
<thead>
<tr>
<th>Filtering method</th>
<th>Signals</th>
<th>Images</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ECG</td>
<td>Piece-wise constant</td>
</tr>
<tr>
<td></td>
<td>RMSE</td>
<td>MAE</td>
</tr>
<tr>
<td>WL-Haar</td>
<td>0.06</td>
<td>0.042</td>
</tr>
<tr>
<td>WL-Db4</td>
<td>0.052</td>
<td>0.038</td>
</tr>
<tr>
<td>SWDCT</td>
<td>0.04</td>
<td>0.028</td>
</tr>
<tr>
<td>SWHaar</td>
<td>0.052</td>
<td>0.033</td>
</tr>
</tbody>
</table>
Hybrid WaveLet-SWTD DCT filtering

Low pass filtering and downsampling

Interpolation

SWDCT3x3 filtering

Interpolation

Low pass filtering and downsampling

Interpolation

SWDCT3x3 filtering

Interpolation

Low pass filtering and downsampling

Effective basis functions in multi resolution DCT
SWTD-DCT, Wavelet-Shrinkage and Hybrid filtering: Performance comparison
SWTD-DCT, Wavelet-Shrinkage and Hybrid filtering: Performance comparison
MWTD-DCT, Wavelet-Shrinkage and Hybrid filtering: Performance Comparison

<table>
<thead>
<tr>
<th>Filter</th>
<th>P-W const. image</th>
<th>Lenna image</th>
<th>MRI</th>
<th>Air photo</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard</td>
<td>8.1 ((3\times3))</td>
<td>9.4 ((3\times3))</td>
<td>6.7 ((5\times5))</td>
<td>8.2 ((3\times3))</td>
</tr>
<tr>
<td>Soft</td>
<td>7.5 ((3\times3))</td>
<td>8.6 ((5\times5))</td>
<td>6.3 ((7\times7))</td>
<td>7.7 ((3\times3))</td>
</tr>
<tr>
<td>WL-Shrinkage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard</td>
<td>8.6 ((\text{binom5}))</td>
<td>10.1 ((\text{binom5}))</td>
<td>8.5 ((\text{binom5}))</td>
<td>9.3 ((\text{binom5}))</td>
</tr>
<tr>
<td>Soft</td>
<td>8.4 ((\text{binom5}))</td>
<td>9.0 ((\text{qmf13}))</td>
<td>7.8 ((\text{binom5}))</td>
<td>8.1 ((\text{binom5}))</td>
</tr>
<tr>
<td>Hybrid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard</td>
<td>8.7 ((\text{binom5}))</td>
<td>9.4 ((\text{binom5}))</td>
<td>6.6 ((\text{binom5}))</td>
<td>8.2 ((\text{binom5}))</td>
</tr>
<tr>
<td>Soft</td>
<td>7.9 ((\text{binom5}))</td>
<td>8.6 ((\text{binom5}))</td>
<td>6.2 ((\text{binom5}))</td>
<td>7.5 ((\text{binom5}))</td>
</tr>
</tbody>
</table>

Standard deviation of residual noise for 4 test noisy images (initial St. Dev. 13; optimized filter parameters)

SWTD “time-frequency” signal representation

Direct transform of a signal \( \{a_k\} \) \((k = 0,1,\ldots,N-1)\) in a sliding window of width \( N_w \) over set of basis functions \( \{\tau_r(n)\} \)

\((n = 0,1,\ldots,N_w-1; r = 0,1,\ldots,N_w-1)\),

\[
\alpha_r^{(k)} = \sum_{n=0}^{N_w-1} a_{k+n-\text{fix}(N_w/2)} \tau_r(n)
\]

Inverse transform restores window samples:

\[
a_{k+n-\text{fix}(N/2)} = \sum_{r=0}^{N_w-1} \alpha_r^{(k)} \tau_n^r(r),
\]

where \( \{\tau_n^r(r)\} \) are basis functions orthogonal to \( \{\tau_r(n)\} \).

For the window central sample:

\[
a_k = \sum_{r=0}^{N_w-1} \alpha_r^{(k)} \tau_{\text{fix}(N/2)}^r(r).
\]

\( \alpha_r^{(k)} \) is signal “time” (index \( k \))-transform domain (index \( r \)) signal representation.
SWTD signal representation as subband decomposition

For fixed $r$, $\alpha_r^{(k)}$ is a vector of $N$ samples. Compute its DFT over index $k$:

$$AT_f^{(r)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \alpha_r^{(k)} \exp \left( i 2\pi \frac{kf}{N} \right) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N_w-1} \tau_r(n) \sum_{k=0}^{N-1} a_{k+n-\text{fix}(N_w/2)} \exp \left( i 2\pi \frac{kf}{N} \right)$$

Let $A_f = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \exp \left( i 2\pi \frac{kf}{N} \right)$ is DFT spectrum of signal $\{a_k\}$. By shift theorem for DFT,

$$\frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_{k+n-\text{fix}(N_w/2)} \exp \left( i 2\pi \frac{kf}{N} \right) = A_f \exp \left( -i 2\pi \frac{n-\text{fix}(N_w/2)}{N} f \right)$$

Then obtain:

$$AT_f^{(r)} = A_f \cdot \sum_{n=0}^{N_w-1} \tau_r(n) \exp \left( -i 2\pi \frac{n-\text{fix}(N_w/2)}{N} f \right) = A_f \cdot \sum_{n=0}^{N-1} \text{rect} \left( \frac{n}{N_w} \right) \tau_r(n) \exp \left( -i 2\pi \frac{n-\text{fix}(N_w/2)}{N} f \right) = \boxed{AT_f^{(r)} \propto A_f T_f^{(r)}}$$

where $T_f^{(r)} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \text{rect} \left( \frac{n}{N_w} \right) \tau_r(n) \exp \left( -i 2\pi \frac{n}{N} f \right)$
SWDCT7x7 subbands

Map of standard deviations of DCT7x7 subbands

SWDCT7x7 subbands (1-D slice)

Frequencies within the bandwidth

Map of standard deviations of DCT7x7 subbands
Subband decomposition for SWDCT, SWWalshT, SWHaarT: comparison

Subbands SWDCT4x4  Subbands SWWalsh4x4  Subbands SWHaar4x4
SWTD DCT, Walsh and Haar: subband decompositions
Subbands (1-D) of Binom5 wavelets (left) and hybrid Binom5 & SWDCT3x3 (right)
Subbands (2-D) of Binom5 wavelets and hybrid Binom5&MWDCT3x3

Binom5 wavelets subbands

Hybrid Binom5&MWDCT3x3 subbands
Hybrid SWTD/wavelet processing: coarse/fine signal subband representation

“I would say that music gives the best index of the large-scale patterns in the brain. But words give a better index of the fine-scale patterns”,

Fred Hoyle, The black cloud, Penguin books, 1960

Hybrid SWTD/Wavelet filtering combines signal subband decomposition in coarse/fine scales: logarithmic coarse scale due to wavelet decomposition and uniform fine scale within each wavelet scale due to SWTD decomposition.

It is remarkable that musical tones are distributed in exactly the way: octaves are arranged in logarithmic scale and 12 semitones are equally spaced within octaves in Bach’s equal tempered scale (J. Backus, The Acoustical Foundations of Music, Norton and Co., New York, 1969)

Floating point (order/mantissa) representation of numbers in computers follows the same idea
CONCLUSION

Sliding Window Transform Domain (SWTD) and Wavelet (WL) signal decompositions represent different implementations of signal subband decompositions.

In image denoising applications SWTD DCT and WL filtering have similar capabilities though SWTD DCT most frequently outperforms WL denoising.

In image restoration and multi component image processing SWTD DCT filtering has additional advantages: it is

- free of any restrictions on signal length
- better suited to multi component signals and to image blurring specification

Hybrid Wavelet/SWTD DCT filtering combines subband decomposition in uniform and logarithmic scales and demonstrates improved denoising capability.