TAMPERE UNIVERSITY OF TECHNOLOGY
Department of Signal Processing – Transforms and Spectral Methods Group

Adaptive non-local transforms for image/video denoising, restoration, and enhancement, and for compressive sensing image reconstruction

Alessandro Foi

www.cs.tut.fi/~foi

Joint work with
Kostadin Dabov, Aram Danielyan, Karen Egiazarian, Vladimir Katkovnik.

SIAM IS08 - San Diego (CA), USA, July 9, 2008

Outline

Block-Matching and 3D filtering (BM3D) algorithm

Shape-adaptive BM3D (BM3D-SADCT) algorithm

[BM3D Deblurring/Deconvolution]

BM3D Sharpening/Enhancement

Iterative reconstruction
for compressive sensing, image upsampling, super-resolution

Block-Matching and 3D filtering (BM3D) denoising algorithm

Generalizes NL-means and overcomplete transform methods.

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising with block-matching and 3D filtering", *Proc. SPIE El. Imaging 2006, Image Process.: Algorithms and Systems V*, no. 6064A-30, San Jose (CA), USA, Jan. 2006.

— , "Image denoising by sparse 3D transform-domain collaborative filtering", *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080-2095, Aug. 2007.

Block-Matching and 3D filtering (BM3D) denoising algorithm

Represents the state-of-the-art in image denoising.

"Probably the most impressive results for a block matching based denoising have been just reported by Dabov *et al.*" – Buades-Coll-Morel, July 2007.

"[...] the current state-of-the-art denoising method, BM3D." – Lyu-Simoncelli, 2008.

A. Buades, B. Coll, and J. M. Morel, "Nonlocal image and movie denoising", *Int. J. Computer Vision*, July 2007.

S. Lyu, and E. Simoncelli, "Modeling multiscale subbands of photographic images with fields of Gaussian scale mixtures", *IEEE TPAMI*, to appear.

E. Vansteenkiste, D. Van der Weken, W. Philips, and E. Kerre, "Perceived image quality measurement of state-of-the-art noise reduction schemes", *LNCS 4179 - ACIVS 2006*, pp. 114-124, Springer, Sept. 2006.

S. Lansel, D. Donoho, and T. Weissman, "DenoiseLab: a standard test set and evaluation method to compare denoising algorithms", `http://www.stanford.edu/~slansel/DenoiseLab/`.

Observation model for the image denoising problem

$$z(x) = y(x) + \eta(x), \qquad x \in X \subset \mathbb{Z}^2,$$

$z : X \to \mathbb{R}$	observed noisy image
$y : X \to \mathbb{R}$	unknown original image (grayscale)
$\eta : X \to \mathbb{R}$	i.i.d. Gaussian white noise, $\eta(\cdot) \sim \mathcal{N}\left(0, \sigma^2\right)$

Notation

Given a function $f : X \to \mathbb{R}$, a subset $U \subset X$, and a function $g : U \to \mathbb{R}$, we denote by:

$f_{\vert U} : U \to \mathbb{R}$	the *restriction* of f on U, $f_{\vert U}(x) = f(x)\ \forall x \in U$;
$g^{\vert X} : X \to \mathbb{R}$	the *zero-extension* of g to X, $\left(g^{\vert X}\right)_{\vert U} = g$ and $g^{\vert X}(x) = 0\ \forall x \in X \setminus U$;
$\chi_U = 1_{\vert U}{}^{\vert X}$	the *characteristic function (indicator)* of U;
$\vert U \vert$	the *cardinality* of U (i.e. the number of its elements of U);
$\circledast$	the *convolution* operation.

Block-matching

Let $x \in X$ and denote by $\tilde{B}_x \subset \mathbb{Z}^2$ be the square block of size $l \times l$ "centered" at x. Let $\mathbb{B}$ be the collection of all such blocks which are entirely contained in X, $\mathbb{B} = \left\{ \tilde{B}_x : x \in X,\ \tilde{B}_x \subset X \right\}$. Equivalently, define $X_{\mathbb{B}} = \left\{ x \in X : \tilde{B}_x \in \mathbb{B} \right\} = \left\{ x \in X : \tilde{B}_x \subset X \right\} \subset X$.

For each block $\tilde{B}_x \in \mathbb{B}$, (i.e. for each point $x \in X_{\mathbb{B}}$), we look for "similar" blocks $\tilde{B}_{x'}$ whose range distance $d_z(x, x')$ with respect to $\tilde{B}_x$,

$$d_z(x, x') = \left\| z_{|\tilde{B}_x} - z_{|\tilde{B}_{x'}} \right\|_2 ,$$

is smaller than a fixed threshold $\tau_{\text{match}} \geq 0$.

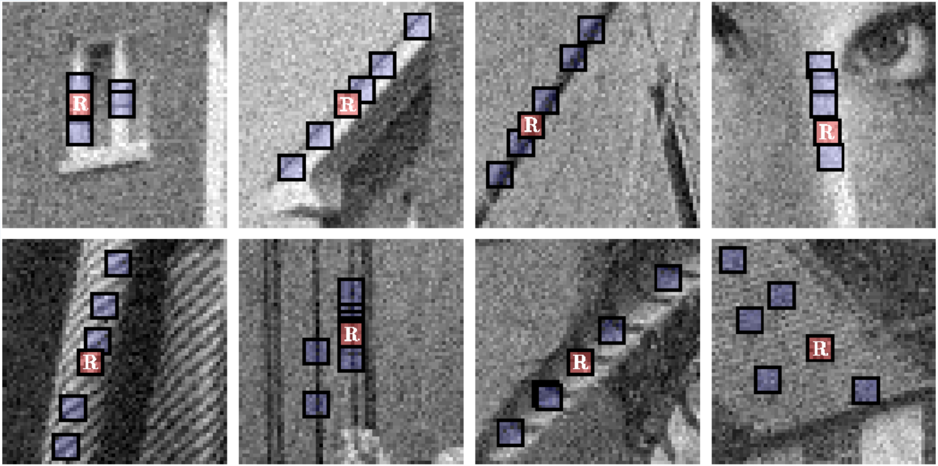
Thus, we construct the set S_x that contains the central points of the found blocks:

$$S_x = \left\{ x' \in X_{\mathbb{B}} : d_z(x, x') \leq \tau_{\text{match}} \right\}.$$

The threshold τ_{match} is the maximum d_z-distance for which two blocks are considered similar.

In case of heavy noise, we embed a coarse prefiltering within d_z (e.g., ℓ^2-distance of thresholded spectra). Otherwise, we need to increase l.

Block-matching



To a fixed "reference" block $\tilde{B}_{x_R} \in \mathbb{B}$ associate a collection (disjoint union) $\widetilde{\mathbb{B}}_{x_R}$ of neighborhoods:

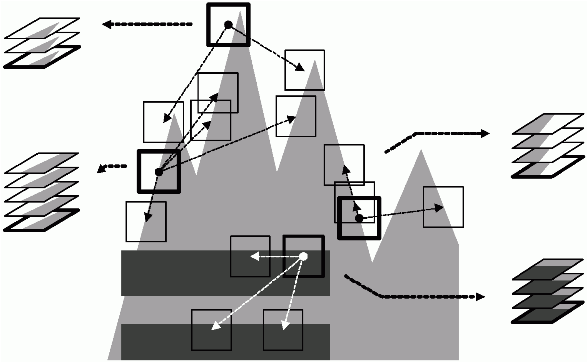
$$\begin{aligned} \widetilde{\mathbb{B}}_{x_R} &= \coprod_{x \in S_{x_R}} \tilde{B}_x = \\ &= \left\{ \left(\tilde{B}_x, x \right) : x \in S_{x_R} \right\} \subset X \times S_{x_R} \subset X \times X. \end{aligned}$$

Group

collection of the noisy patches $\quad z_{|\tilde{B}_x},\ \tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$

(Compact notation) $\quad \mathbf{Z}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}.$

The patches can be stacked together into a 3-D data array defined on the square prism $B \times \{1, \ldots, |S_{x_R}|\}$.



Why groups are good and why do we need to be careful

Groups are characterized by both:

- ⋄ *intra*-block correlation between the pixels of each grouped block (natural images);
- ∘ *inter*-block correlation between the corresponding pixels of different blocks (grouped block are similar);

Warnings:

- ⋄ blocks are not necessary flat or smooth but can be anything;
- ∘ “similar” does not mean “identical”.

Goals:

- ⋄ exploit intra-block correlation whenever possible, without smoothing away the unexpected;
- ∘ exploit similarity in the forms in which it exists, without forcing dissimilar blocks to become identical.

Collaborative filtering

- *each grouped block collaborates for the filtering of all others, and vice versa.*
- *provides individual estimates for all grouped blocks (not necessarily equal).*

Realized as shrinkage in a 3-D transform domain.

Typically separable transform: $T^{\text{3D}} = T^{\text{2D}} \circ T^{\text{1D}}$.

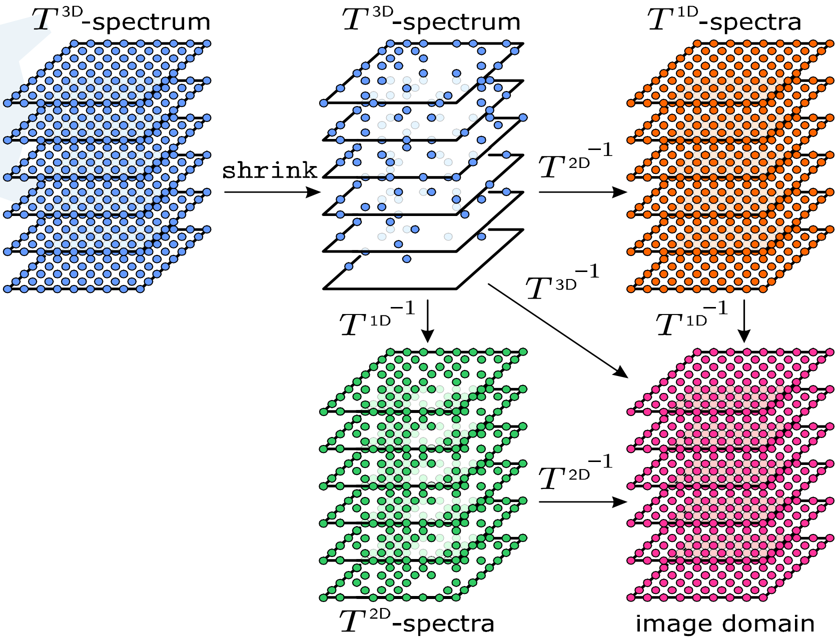
E.g.: 2D-DCT $\circ$ DCT $=$ 3D-DCT
or, restricting h and $|S_{x_R}|$ to powers of two,
biorth. 2D-DWT $\circ$ Haar 1D-DWT
shrinkage: hard-thresholding

$$\widehat{\mathbf{Y}}_{x_R} = T^{\text{3D}\,-1}\left(\mathtt{shrink}\left(T^{\text{3D}}\left(\mathbf{Z}_{x_R}\right)\right)\right)$$

The group estimate $\widehat{\mathbf{Y}}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}$ is composed of slices with local block estimates $\hat{y}_{x,x_R} : \tilde{B}_x \to \mathbb{R}$ for each $\tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$.

Total variance of $\widehat{\mathbf{Y}}_{x_R}$ can be estimated as $\text{tsvar}\left\{\widehat{\mathbf{Y}}_{x_R}\right\} \approx \sigma^2 N^{\text{har}}_{x_R}$,
$N^{\text{har}}_{x_R}$ is number of coefficients of $T^{\text{3D}}(\mathbf{Z}_{x_R})$ that survive thresholding (so-called "number of harmonics").

Collaborative filtering



Aggregation

For each reference point $x_R \in X$, grouping and collaborative filtering generate a group $\widehat{\mathbf{Y}}_{x_R}$ of $|S_{x_R}|$ distinct *local* estimates of y.

Overall, we have a highly redundant and rich representation of the original image y composed of the estimates

$$\coprod_{x_R \in X,\ x \in S_{x_R}} \hat{y}_{x,x_R}, \quad \text{where} \quad \hat{y}_{x,x_R} : \tilde{B}_x \to \mathbb{R}.$$

Note: different groups $\mathbf{Z}_{x_R}$ and $\mathbf{Z}_{x'_R}$ can lead to different estimates $\hat{y}_{x,x_R}$ and $\hat{y}_{x,x'_R}$ even when these estimates are defined on the same block $\tilde{B}_x$!

In order to obtain a single *global* estimate $\hat{y}^{\mathrm{ht}} : X \to \mathbb{R}$ defined on the whole image domain, all these local estimates are averaged together using adaptive weights $w_{x_R} > 0$ in the following convex combination:

$$\hat{y}^{\mathrm{ht}} = \frac{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \hat{y}_{x,x_R}{}^{|X}}{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \chi_{\tilde{B}_x}} \qquad w_{x_R} = \frac{1}{\sigma^2 N_{x_R}^{\mathrm{har}}}.$$

Wiener filtering stage

Denoising can be improved by performing matching within this estimate and replacing hard-thresholding by empirical Wiener filtering in the collaborative shrinkage.

Block-Matching

Noise in $\hat{y}^{\mathrm{ht}}$ is significantly attenuated: more accurate matching by replacing the distance d_z by a distance $d_{\hat{y}^{\mathrm{ht}}}$:

$$d_{\hat{y}^{\mathrm{ht}}}(x_R, x) = \left\| \hat{y}^{\mathrm{ht}}{}_{|\tilde{B}_{x_R}} - \hat{y}^{\mathrm{ht}}{}_{|\tilde{B}_x} \right\|_2,$$

The sets S_{x_R} are redefined as

$$S_{x_R} = \left\{ x \in X_{\mathbb{B}} : d_{\hat{y}^{\mathrm{ht}}}(x_R, x) \leq \tau_{\mathrm{match}} \right\}.$$

These new sets S_{x_R} lead to new collections (disjoint unions) of blocks $\widetilde{\mathbb{B}}_{x_R} = \coprod_{x \in S_{x_R}} \tilde{B}_x$.

Grouping: two groups

$$\mathbf{Z}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}, \quad \text{built by stacking together the noisy patches } \; z_{|\tilde{B}_x}, \tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$$

$$\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R} : \widetilde{\mathbb{B}}_{x_R} \to \mathbb{R}, \quad \text{built by stacking together the estimate patches } \; \widehat{y}^{\mathrm{ht}}_{|\tilde{B}_x}, \tilde{B}_x \in \widetilde{\mathbb{B}}_{x_R}$$

Collaborative Wiener filtering

Group Wiener estimate
$$\widehat{\mathbf{Y}}_{x_R} = T^{\mathsf{3D}\,-1}\left(\mathbf{W}_{x_R} T^{\mathsf{3D}}\left(\mathbf{Z}_{x_R}\right)\right)$$

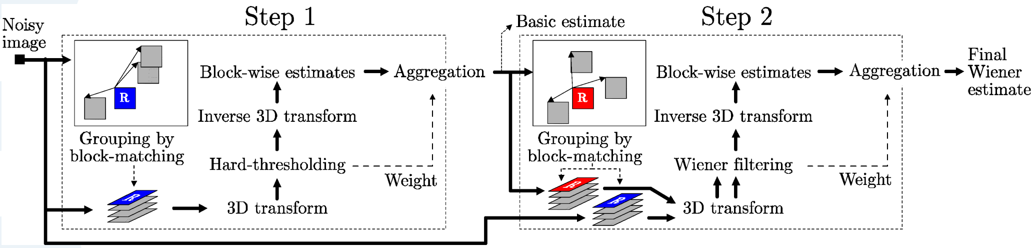
Wiener attenuation factors
$$\mathbf{W}_{x_R} = \frac{\left(T^{\mathsf{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2}{\left(T^{\mathsf{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2 + \sigma^2}$$

Estimate of total variance $\quad \mathrm{tsvar}\left\{\widehat{\mathbf{Y}}_{x_R}\right\} \approx \sigma^2 \left\|\mathbf{W}_{x_R}\right\|_2^2.$

Aggregation

Global estimate
$$\hat{y}^{\mathrm{wie}} = \frac{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \hat{y}_{x,x_R}{}^{|X}}{\sum_{x_R \in X} \sum_{x \in S_{x_R}} w_{x_R} \chi_{\tilde{B}_x}}, \qquad w_{x_R} = \frac{1}{\sigma^2 \left\|\mathbf{W}_{x_R}\right\|_2^2}.$$

BM3D flowchart



▷ Process overlapping blocks in a raster scan. For each such block, do the following:

(a) Use block-matching to find the locations of the blocks that are similar to the currently processed one. Form a 3D array (group) by stacking the blocks located at the obtained locations.

(b) Apply a 3D transform on the formed group.

(c) Attenuate the noise by shrinkage the 3D transform spectrum.

(d) invert the 3D transform to produce filtered grouped blocks.

▷ Return the filtered blocks to their original locations in the image domain and compute the resultant filtered image by a weighted average of these filtered blocks (aggregation).

Is BM3D slow?

Based on blockwise estimates, the algorithm allows for dramatic complexity reduction and thus acceleration with negligible loss of denoising performance.

Complexity reduction:

- use predictive-search for block-matching;
- use separable wavelets for the 3D transform (Haar/biorthogonal);
- scalable (controlling the level of overcompleteness and predictive-search).

Eventually, BM3D is faster than other algorithms of comparable denoising performance.

Shape-adaptive BM3D (BM3D-SADCT)

Associate to every $x \in X$ an adaptive neighborhood $\tilde{U}_x^+$ where a low-order polynomial model fits to the data.

By demanding the local fit of a polynomial model, we are able to avoid the presence of singularities, discontinuities, or sharp transitions within the transform support. In this way, we increase further the sparsity in the transform domain, improving the effectiveness of shrinkage.

Main ingredients:

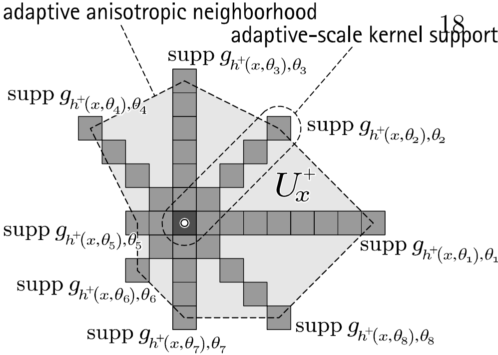
- **Local Polynomial Approximation - Intersection of Confidence Intervals (LPA-ICI)** to adapt with respect to unknown smoothness of the image;
- **Block-Matching** to enable non-locality;
- **Shape-Adaptive DCT** low-complexity 2D transform on arbitrarily-shaped domains enables efficient **shape-adaptive collaborative filtering**.

Directional varying-scale LPA estimates

$$\hat{y}_{h,\theta_k} = z \circledast g_{h,\theta_k}$$

scales: $h \in \{h_1, \ldots, h_J\} = H$

directions: $\theta_k = \frac{(k-1)}{4}\pi,\ k = 1, \ldots, 8$



ICI directional adaptive scales

$$\{h^+(x,\theta_k)\}_{k=1}^{8}$$

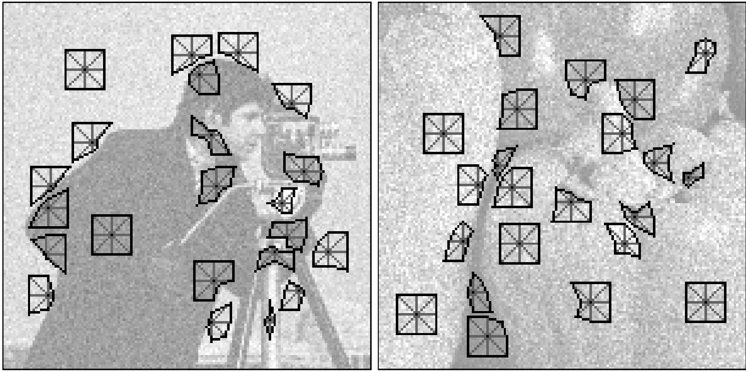
Adaptive neighborhood of the origin

$$U_x^+ = \texttt{polygonal\_hull}\left\{\mathrm{supp}\, g_{h^+(x,\theta_k),\theta_k}\right\}_{k=1}^{8}$$

Adaptive neighborhood of estimation point x

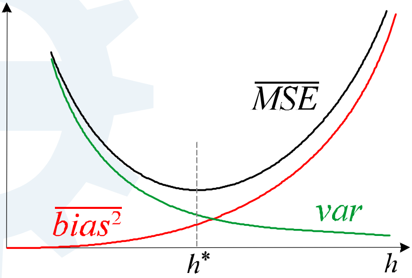
(mirror-translates)

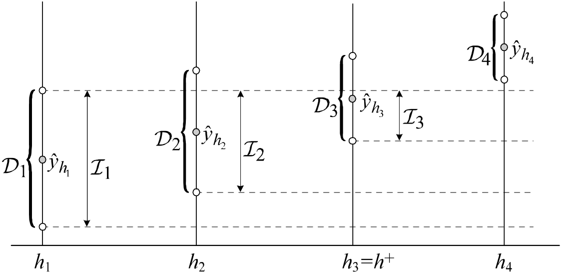
$$\tilde{U}_x^+ = \\ = \{v \in X : (x - v) \in U_x^+\}$$



Intersection of Confidence Intervals (*ICI*) (Goldenshluger&Nemirovski, 1997)

(for each fixed direction θ_k)





The estimates $\hat{y}_h(x)$ are calculated for a set $H = \{h_j\}_{j=1}^{J}$ of increasing scales. The *ICI* rule yields a pointwise adaptive estimate $\hat{y}_{h^+}(x)$, where for every x an adaptive scale $h^+(x) \in H$ is used such that $\hat{y}_{h^+}(x) \approx \hat{y}_{h^*(x)}(x)$.

ICI rule: *Consider the intersection of confidence intervals*

$$\mathcal{I}_j = \bigcap_{i=1}^{j} \mathcal{D}_i, \quad \text{where} \quad \mathcal{D}_i = \left[\hat{y}_{h_i}(x) - \Gamma\sigma_{\hat{y}_{h_i}}, \hat{y}_{h_i}(x) + \Gamma\sigma_{\hat{y}_{h_i}}\right]$$

and $\Gamma > 0$ *is a threshold parameter, and let* j^+ *be the largest of the indexes* j *for which* $\mathcal{I}_j$ *is non-empty,* $\mathcal{I}_{j^+} \neq \varnothing$ *and* $\mathcal{I}_{j^++1} = \varnothing$*. Then,* h^+ *is defined as* $h^+ = h_{j^+}$ *and the adaptive estimate is* $\hat{y}_{h^+}(x)$*.*

Block-matching

Adaptive neighborhoods can be *too small* for reliable matching!

Matching for $\tilde{U}_x^+$ needs to be carried out for a superset.

We use square blocks of size $(2h_{\max}-1)\times(2h_{\max}-1)$ centered at x, $h_{\max}=\max\{H\}$.

$$\text{Adaptive neighborhoods } \tilde{U}_x^+ \quad \forall x\in X$$
$$\text{Blocks } \tilde{B}_x \quad \forall x\in X_{\mathbb{B}}\subsetneq X$$

To every $x\in X$ we associate $x_{\mathbb{B}}\in X_{\mathbb{B}}$ such that $\|\delta_{\mathbb{B}}(x)\|_2$ of $\delta_{\mathbb{B}}(x)=x_{\mathbb{B}}-x$ is minimal.

The mapping $x\mapsto x_{\mathbb{B}}$ and $\delta_{\mathbb{B}}(x)$ are univocally defined (for convex X).

$\delta_{\mathbb{B}}(x)\neq 0$ only for x sufficiently close to the boundary ∂X of X.

Shape-adaptive grouping

For given points x, x_R define the translate of $\tilde{U}^+_{x_R}$

$$\tilde{U}^+_{x,x_R} = \left\{v \in X : (x - v) \in U^+_{x_R}\right\} = \left\{v \in X : (x_R - x + v) \in \tilde{U}^+_{x_R}\right\}.$$

$\tilde{U}^+_{x,x_R}$ is an adaptive neighborhood of x which uses the adaptive scales of the "reference point" x_R.

It can happen that $\tilde{U}^+_{x,x_R} \neq \tilde{U}^+_x$.

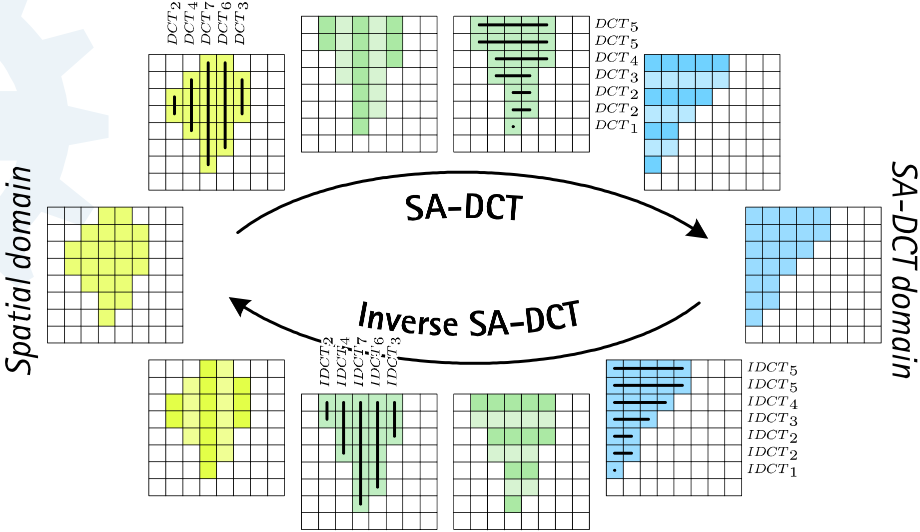
To a given "reference" point x_R we can now associate not only its own adaptive neighborhood $\tilde{U}^+_{x_R}$, but a collection (disjoint union) $\widetilde{\mathbb{U}}_{x_R}$ of neighborhoods defined as

$$\widetilde{\mathbb{U}}_{x_R} = \coprod_{x+\delta_{\mathbb{B}}(x_R)\in S_{x_R+\delta_{\mathbb{B}}(x_R)}} \tilde{U}^+_{x,x_R} = \left\{\tilde{U}^+_{x,x_R} : x + \delta_{\mathbb{B}}(x_R) \in S_{x_R+\delta_{\mathbb{B}}(x_R)}\right\},$$

where $S_{x_R+\delta_{\mathbb{B}}(x_R)}$ is the result of block-matching for $\tilde{B}_{x_R+\delta_{\mathbb{B}}(x_R)}$.

All neighborhoods in $\widetilde{\mathbb{U}}_{x_R}$ have the same shape, completely determined by adaptive scales $\{h^+(x_R, \theta_k)\}_{k=1}^{8}$ at x_R.

Shape-Adaptive Discrete Cosine Transform (SA-DCT) (Sikora et al., 1995)



Shape-Adaptive Discrete Cosine Transform (SA-DCT) and its inverse. Transformation is computed by cascaded application of one-dimensional varying-length DCT transforms, along the columns and along the rows.

Shape-Adaptive Discrete Cosine Transform (SA-DCT)

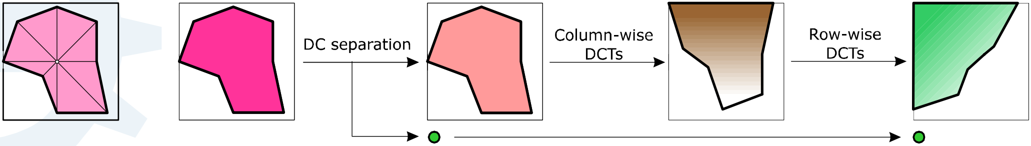
- direct generalization of the classical block-DCT (B-DCT);
- on rectangular domains (e.g., squares) the SA-DCT and B-DCT coincide;
- the same computational complexity as the B-DCT (separable);
- SA-DCT is part of the MPEG-4 standard;
- efficient (low-power) hardware implementations available;

- shape must be coded separately (constitutes some overhead).

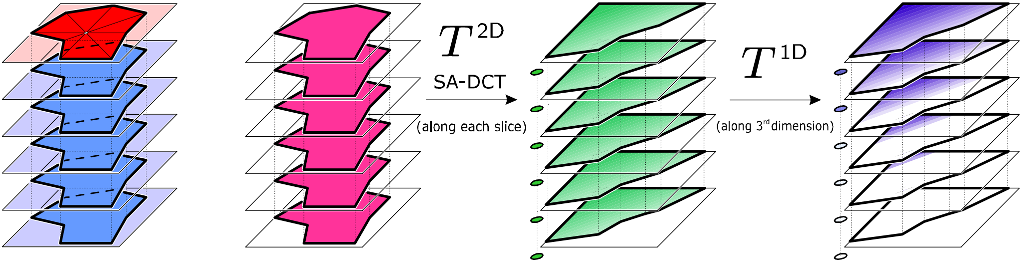
Orthonormal SA-DCT does not have a DC term and works best if applied on zero-mean data: "Orthonormal SA-DCT with DC separation and ΔDC compensation", Kauff et al. 1997.

SA-DCT (forward transform)

[as used in Pointwise SA-DCT denoising algorithm (Foi et al., IEEE TIP 2007)]



Shape-adaptive collaborative filtering (forward transform)



BM3D-SADCT

BM3D-SADCT algorithm follows the general scheme of BM3D:

LPA-ICI + Block-Matching/Grouping + Collaborative hard-thresholding + Aggregation

LPA-ICI + Block-Matching/Grouping + Collaborative Wiener filtering + Aggregation

!!! Delicate points with respect to standard BM3D are:
Wiener filtering (DC separation)
aggregation weights (supports of different sizes)

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "A non-local and shape-adaptive transform-domain collaborative filtering", *LNLA 2008*, August 2008.

Denoising results

Noisy *Cameraman*, $\sigma = 25$

BM3D PSNR=29.48dB

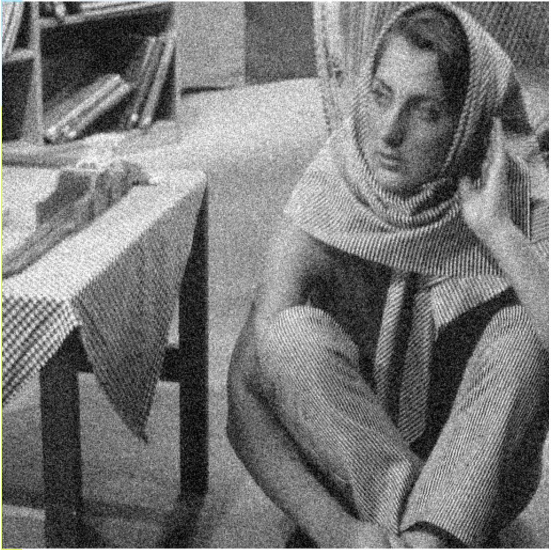
Denoising results

P.SADCT PSNR=29.10dB

BM3D-SADCT PSNR=29.56dB

Denoising results

Noisy *Barbara*, $\sigma = 25$



BM3D PSNR=30.72dB

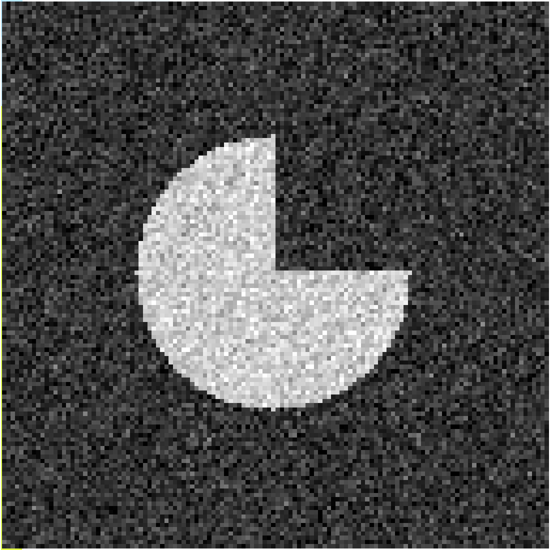
Denoising results

P.SADCT PSNR=28.94dB

BM3D-SADCT PSNR=30.59dB

Denoising results

Noisy *GreyCheese*, $\sigma = 25$



BM3D PSNR=36.71dB

Denoising results

P.SADCT PSNR=38.68dB

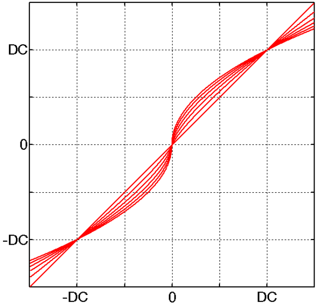
BM3D-SADCT PSNR=39.62dB

Video denoising (VBM3D)

Collaborative sharpening

Introduce **alpha-rooting** immediately after shrinkage, before inverting the T^{3D} transform. Modify aggregation weights (sharpening changes the total variance of the estimate group).

Alpha-rooting (Aghagolzadeh&Ersoy, 1992)



Transform spectrum t of a signal with DC coefficient $t(0)$, $\alpha \geq 1$ sharpening exponent

$$t_{\text{sharp}}(i) = \begin{cases} \operatorname{sign}[t(i)]\, |t(0)| \left| \dfrac{t(i)}{t(0)} \right|^{\frac{1}{\alpha}}, & \text{if } t(0) \neq 0 \\ t(i), & \text{otherwise.} \end{cases}$$

Aggregation weights for sharpening

Variance of sharpened coefficients (using first order approximations)

$$\begin{aligned} \operatorname{var}\{t_{\mathrm{sharp}}(i)\} &\simeq \left(1-\frac{1}{\alpha}\right)^2 |t(0)|^{-\frac{2}{\alpha}} |t(i)|^{\frac{2}{\alpha}} \sigma^2 + \frac{1}{\alpha^2} |t(i)|^{\frac{2}{\alpha}-2} |t(0)|^{2-\frac{2}{\alpha}} \sigma^2 = \\ &= \omega_i \sigma^2. \end{aligned}$$

Total variance of the thresholded and sharpened group $\widehat{\mathbf{Y}}_{x_R}^{\mathrm{sharp}}$ is approximated as

$$\operatorname{tsvar}\left\{\widehat{\mathbf{Y}}_{x_R}^{\mathrm{sharp}}\right\} = \sigma^2 + \sum_{t(i)\neq 0, i>0} \omega_i \sigma^2.$$

Hence, aggregation weights are

$$w_{x_R} = \frac{1}{\operatorname{tsvar}\left\{\widehat{\mathbf{Y}}_{x_R}^{\mathrm{sharp}}\right\}}.$$

Collaborative sharpening: experiments

Noisy *House*, $\sigma = 10$

Collaborative sharpening: experiments

BM3D-SH3D, $\alpha = 1.2$

Collaborative sharpening: experiments

BM3D-SH3D, $\alpha = 1.4$

Collaborative sharpening: experiments

BM3D-SH3D, $\alpha = 1.6$

Collaborative sharpening: experiments

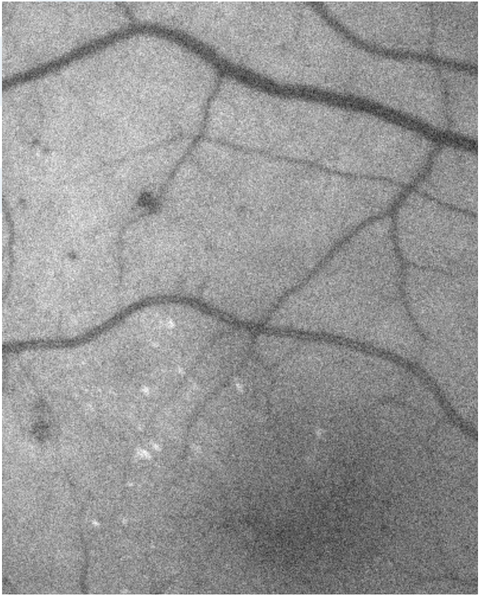
BM3D-SH3D, $\alpha = 1.8$

Collaborative sharpening: experiments

BM3D-SH3D, $\alpha = 2.0$

Collaborative sharpening: experiments

Noisy *Fundus* $\sigma = 20$



BM3D-SH3D

BM3D Deconvolution (non blind)

Approach:

standard Tikhonov regularized deconvolution coupled with BM3D regularization
(in practice the filtering is equivalent to colored noise removal)

References:

K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, "Image restoration by sparse 3D transform-domain collaborative filtering", *Proc. SPIE El. Imaging 2008, Image Process.: Algorithms and Systems VII*, 6812-06, San Jose (CA), USA, January 2008.

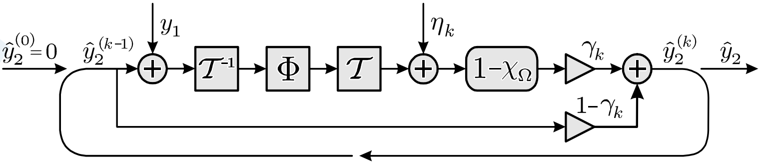
A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction", *Proc. SPIE El. Imaging 2006, Image Process.: Algorithms and Systems V*, 6064A-18, San Jose (CA), USA, January 2006.

Iterative image reconstruction

Ω is the support of the available portion of the spectrum y

$$y = y_1 + y_2 = \chi_\Omega y + (1 - \chi_\Omega)\, y$$

Recursive algorithm



$$\begin{cases} \hat{y}_2^{(0)} = 0, \quad \texttt{(initialization)} & k = 0, \\ \hat{y}_2^{(k)} = \hat{y}_2^{(k-1)} - \gamma_k \left[\hat{y}_2^{(k-1)} - (1 - \chi_\Omega)\, \mathcal{T}\left(\Phi\left(\mathcal{T}^{-1}\left(y_1 + \hat{y}_2^{(k-1)} \right)\right)\right) + (1 - \chi_\Omega)\, \eta_k \right], & k \geq 1. \end{cases}$$

$\mathcal{T}$	transform	$\mathcal{T} = \mathcal{F}$ Fourier
Φ	spatially adaptive filter	$\Phi =$ BM3D
η_k	excitation noise	$\eta_k = \mathcal{N}\left(0, \alpha^{-k-\beta}\right)$
γ_k	step size	$\gamma_k = 1$

Possible interpretations:

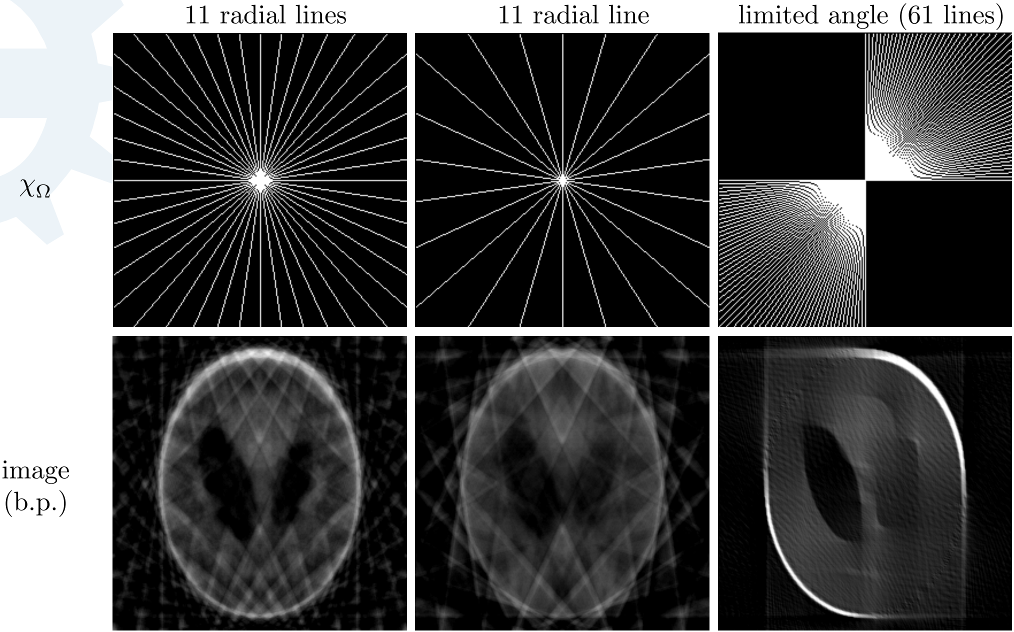
stochastic optimization (Robbins-Monro type),
random search,
simulated annealing,
randomized alternated projections / POCS,
etc.

Compressive sensing toy examples: Radon inversion from sparse projections and limited-angle tomography

Shepp-Logan phantom

$\mathcal{T} = \mathcal{F}$ Fourier transform

Compressive sensing toy examples: Radon inversion from sparse projections and limited-angle tomography



Compressive sensing toy examples: Radon inversion from sparse projections and limited-angle tomography

In all three cases we achieve *exact* reconstruction (PSNR≃260dB)

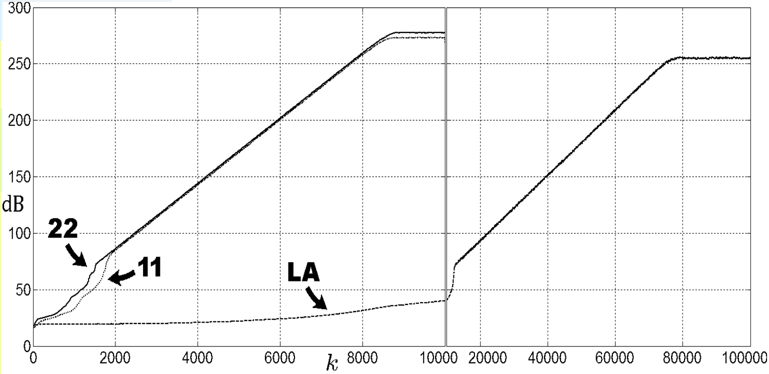


Image upsampling

χ_Ω

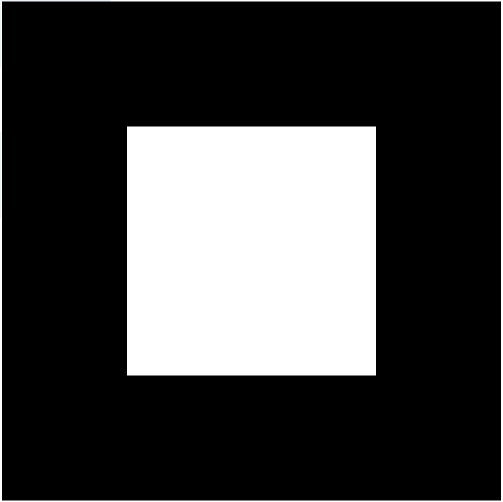


image (b.p.)

Image upsampling

extrapolating missing high-frequencies

Image upsampling (4×, blind case)

References:

Egiazarian, K., A. Foi, and V. Katkovnik, “Compressed Sensing Image Reconstruction via Recursive Spatially Adaptive Filtering”, *Proc. IEEE Int. Conf. Image Process., ICIP 2007*, San Antonio, TX, USA, pp. 549-552, September 2007.

Image upsampling across wavelet approximation subbands (nested recursion)

A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian: "Image Upsampling via Spatially Adaptive Block-Matching Filtering”, *EUSIPCO 2008*, August 2008.

Super-resolution of video

A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian: "Image and video super-resolution via spatially adaptive block-matching filtering", *LNLA 2008*, August 2008.

LNLA 2008

2008 International Workshop on Local and Non-Local Approximation in Image Processing

Lausanne, Switzerland - August 23-24, 2008

`http://sp.cs.tut.fi/ticsp/lnla08`

LNLA 2008 is a satellite event of EUSIPCO 2008 (August 25-29, 2008).

IEEE Finland Section SP / CAS Chapter

TICSP Tampere International Center for Signal Processing

EURASIP