TAMPERE UNIVERSITY OF TECHNOLOGY

Department of Signal Processing – Computational Imaging/Transforms and Spectral Methods

Sparse image representation in nonlocal transform domain: the BM3D filter and its applications

Karen Egiazarian   Alessandro Foi

http://www.cs.tut.fi/~karen
http://www.cs.tut.fi/~foi

http://sp.cs.tut.fi/groups/trans

Queen Mary University of London   *March 25, 2010*

Outline

1. Block-Matching and 3D filtering (BM3D) algorithm

Grouping and collaborative filtering, block-based algorithm and shape-adaptive PCA implementation.

2. Extension and applications

V-BM3D for video and multiframe data, image sharpening, image deblurring, iterative reconstruction algorithms for inverse imaging (compressive sensing, upsampling, and super-resolution)

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

Generalizes NL-means (Buades, Coll, Morel) and overcomplete transform methods.

A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one", *Multisc. Model. Simulat.*, vol. 4, no. 2, pp. 490-530, 2005.

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, January 2006.

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

V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From local kernel to nonlocal multiple-model image denoising", *Int. J. Computer Vision*, vol. 86, no. 1, pp. 1-32, January 2010.

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, $\eta(\cdot) \sim \mathcal{N}(0, \sigma^2)$

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:

$f_{\|U} : U \to \mathbb{R}$	the *restriction* of f on U, $f_{\|U}(x) = f(x)\ \forall x \in U$;
$g^{\|X} : X \to \mathbb{R}$	the *zero-extension* of g to X, $\left(g^{\|X}\right)_{\|U} = g$ and $g^{\|X}(x) = 0\ \forall x \in X \setminus U$;
$\chi_U = 1_{\|U}{}^{\|X}$	the *characteristic function (indicator)* of U;
$\|U\|$	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 $\tau_{\mathrm{match}} \geq 0$.

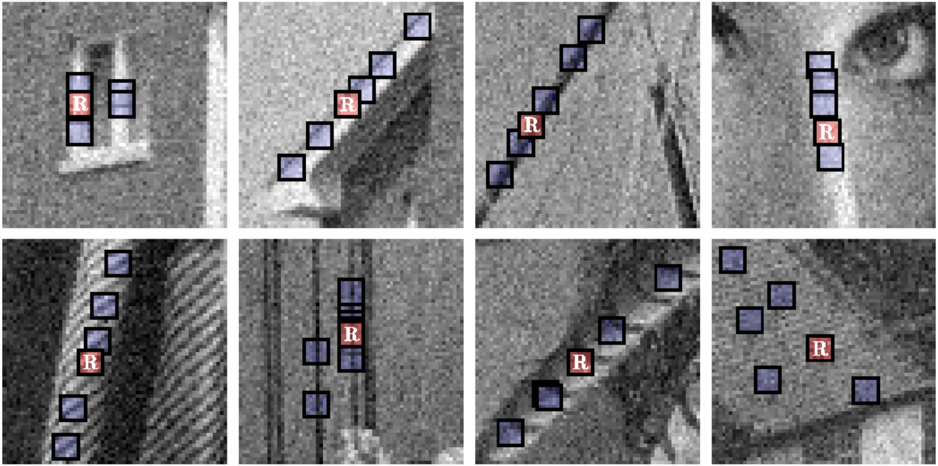
Thus, we construct the set S_x that contains the central points of the found blocks:

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

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:

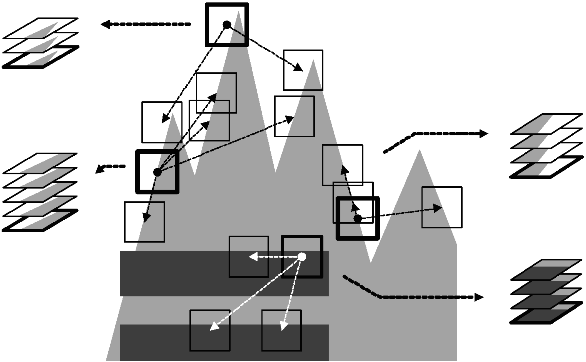
$$\begin{aligned} \widetilde{\mathbb{B}}_{x_R} &= \coprod_{x \in S_{x_R}} \tilde{B}_x = \\ &= \left\{ \left(\tilde{B}_x, x \right) : x \in S_{x_R} \right\} \subset X \times S_{x_R} \subset X \times X. \end{aligned}$$

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

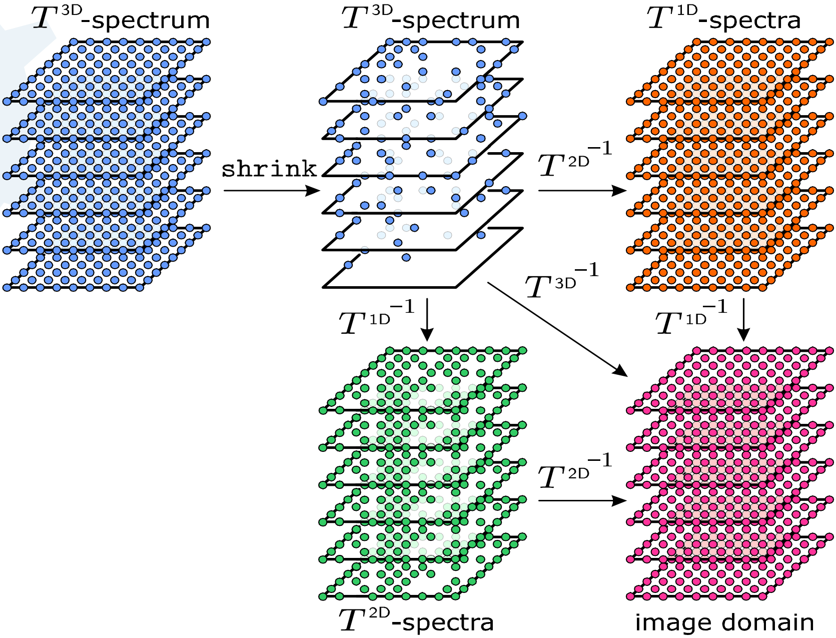
E.g.: 2D-DCT $\circ$ DCT $=$ 3D-DCT
or, restricting h and $|S_{x_R}|$ to powers of two,
biorth. 2D-DWT $\circ$ Haar 1D-DWT
shrinkage: hard-thresholding

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

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 $\operatorname{tsvar}\left\{\widehat{\mathbf{Y}}_{x_R}\right\} \approx \sigma^2 N_{x_R}^{\text{har}}$,
$N_{x_R}^{\text{har}}$ 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:

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

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

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

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

Collaborative Wiener filtering

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

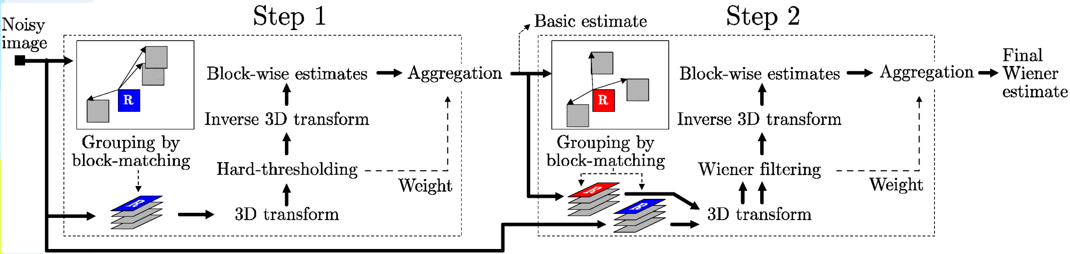
Wiener attenuation factors $$\boxed{\mathbf{W}_{x_R} = \frac{\left(\mathcal{T}^{\mathsf{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2}{\left(\mathcal{T}^{\mathsf{3D}}\left(\widehat{\mathbf{Y}}^{\mathrm{ht}}_{x_R}\right)\right)^2+\sigma^2}}$$

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

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 3-D transform on the formed group.
  - (c) Attenuate the noise by shrinkage the 3-D transform spectrum.
  - (d) invert the 3-D 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).

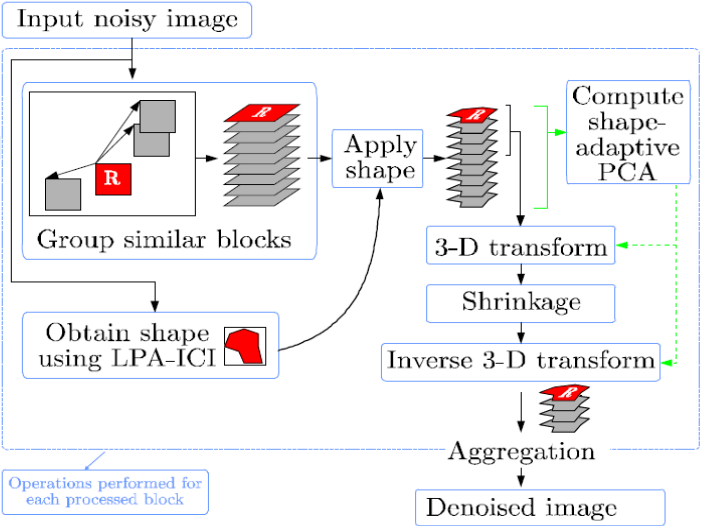
BM3D with Shape-Adaptive PCA (BM3D-SAPCA)

Main ingredients:

- **Local Polynomial Approximation - Intersection of Confidence Intervals (LPA-ICI)** to adaptively select support for 2-D transform;
- **Block-Matching** to enable non-locality;
- **Shape-Adaptive PCA (SA-PCA);**
- **Shape-Adaptive DCT** low-complexity 2-D transform on arbitrarily-shaped domains (when SA-PCA is not feasible).

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D Image Denoising with Shape-Adaptive Principal Component Analysis", *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'09)*, Saint-Malo, France, April 2009.

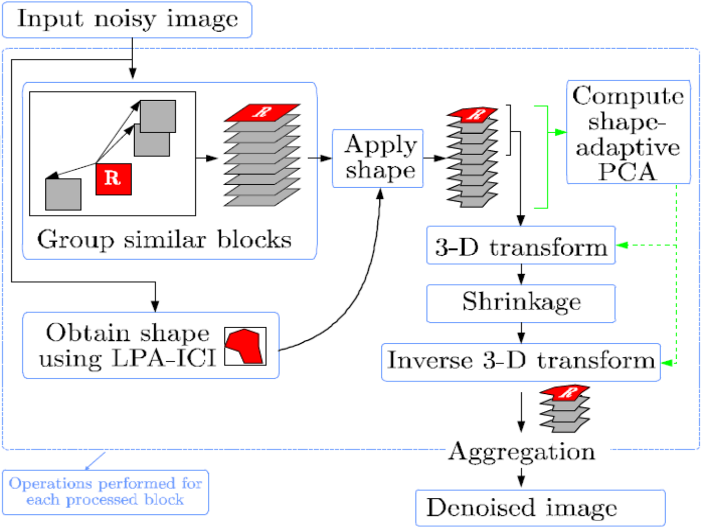
BM3D-SAPCA



At each pixel:

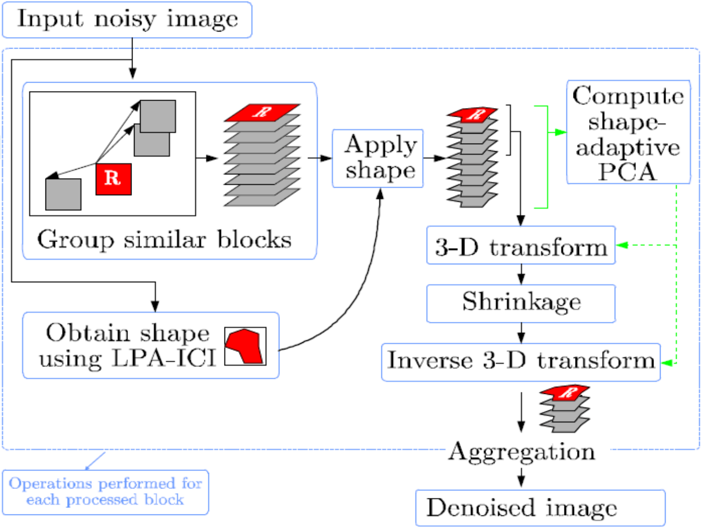
1. Group together square image blocks that are similar to the block centered at the current pixel.

BM3D-SAPCA



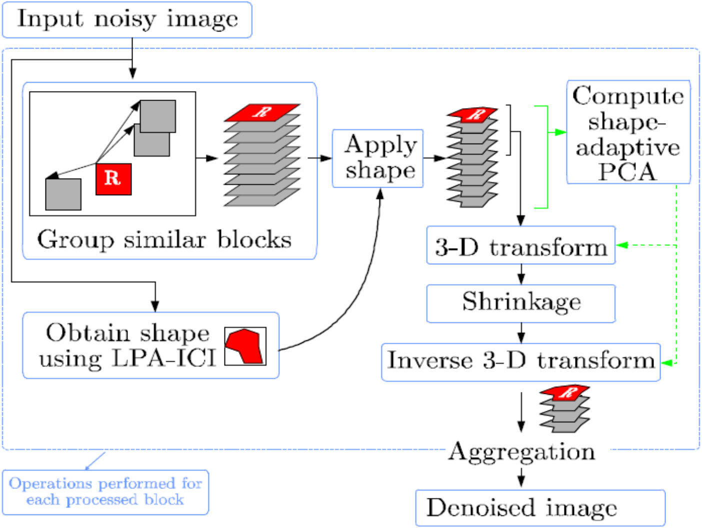
2. Obtain the anisotropic neighborhood at the current pixel using 8-directional LPA-ICI. Apply its shape on each of the grouped blocks, producing a group of adaptive-shape neighborhoods.

BM3D-SAPCA



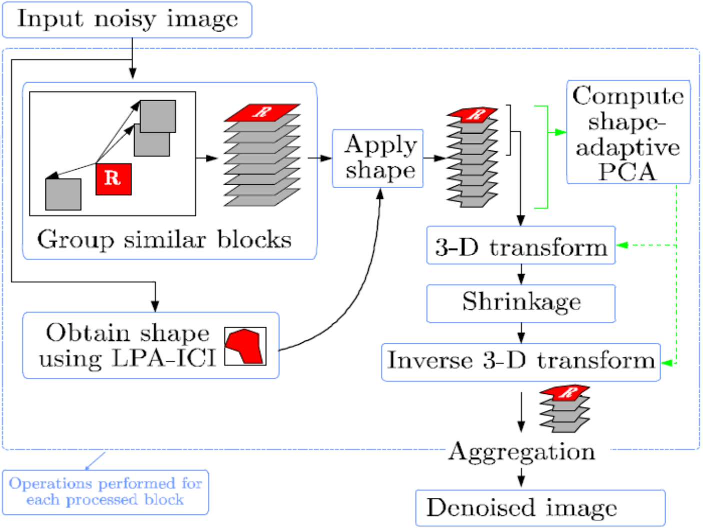
3. Use this group as training data for computing Shape-Adaptive PCA (SVD of the empirical second-moment matrix estimated from the group of similar adaptive-shape neighborhoods).

BM3D-SAPCA



3b. Keep only the eigenvectors (PC) whose corresponding eigenvalues are greater than a threshold proportional to the noise variance (*trimmed PCA*).
The overall 3-D transform is a separable composition of the PCA (applied on each image patch) and a fixed orthogonal 1-D transform in the third dimension.

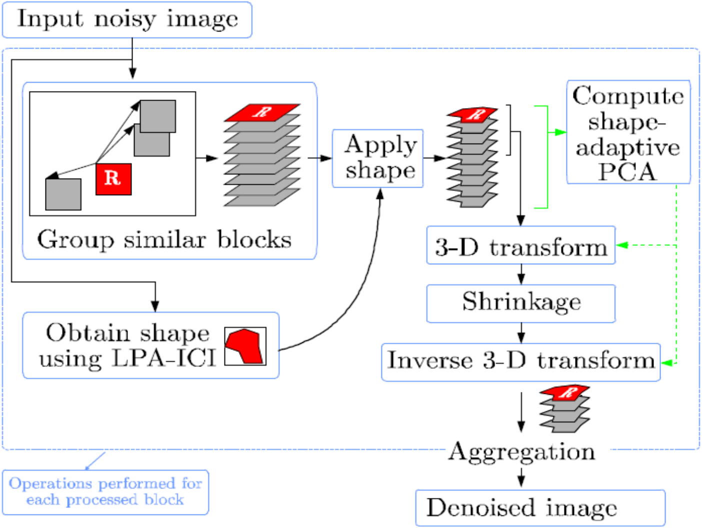
BM3D-SAPCA



4. Apply the 3-D transform on a group of adaptive-shape neighborhoods.

5. Attenuate noise by shrinakage (hard-thresholding or empirical Wiener filtering).

BM3D-SAPCA



6. Apply the inverse 3-D transform to obtain filtered neighborhoods,
7. Return the filtered neighborhoods to their original locations and aggregate in case of overlapping.

BM3D-SAPCA

The scheme is implemented in three iterations:

- **I:** hard-thresholding, BM and PCA on noisy data
- **II:** hard-thresholding, BM and PCA on estimate from I.
- **III:** empirical Wiener filtering, BM and PCA on estimate from II.

Directional varying-scale LPA estimates

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

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

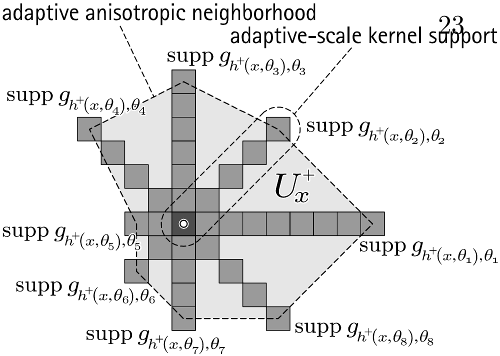
directions: $\theta_k = \frac{(k-1)}{4}\pi, \; k = 1, \dots, 8$

ICI directional adaptive scales

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

Adaptive neighborhood of the origin

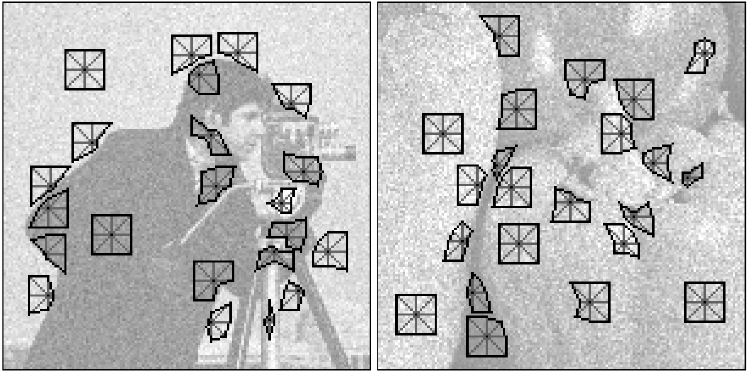
$$U_x^+ = \texttt{polygonal\_hull}\{\operatorname{supp} g_{h^+(x,\theta_k),\theta_k}\}_{k=1}^{8}$$



Adaptive neighborhood of estimation point x

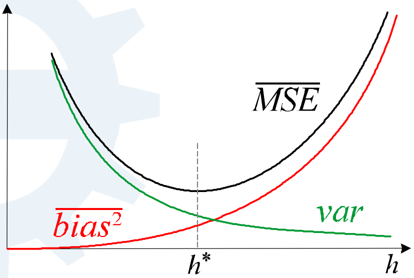
(mirror-translates)

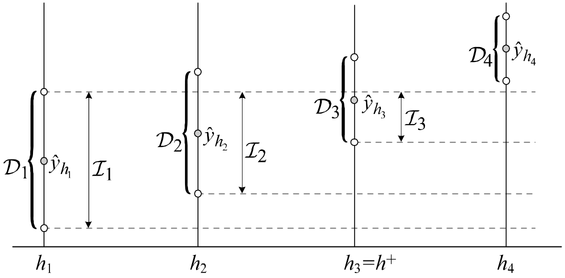
$$\tilde{U}_x^+ = \{v \in X : (x - v) \in U_x^+\}$$



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 $\tilde{U}^+_{x_R}$

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

$\tilde{U}^+_{x,x_R}$ 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 PCA

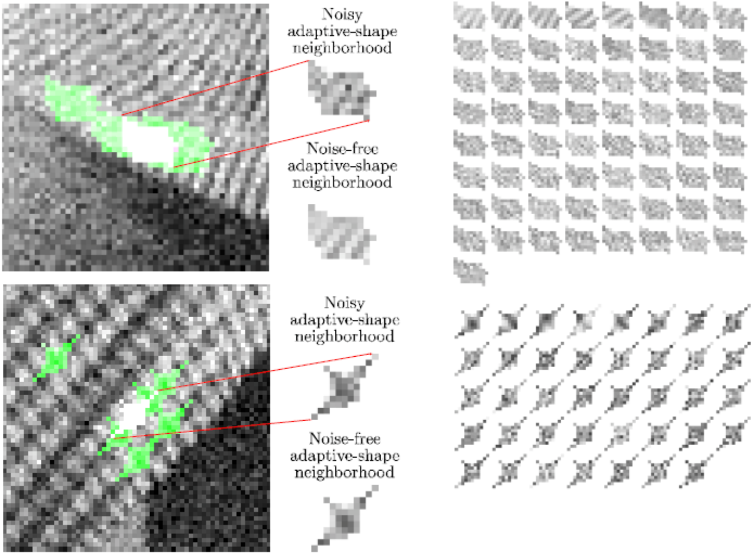
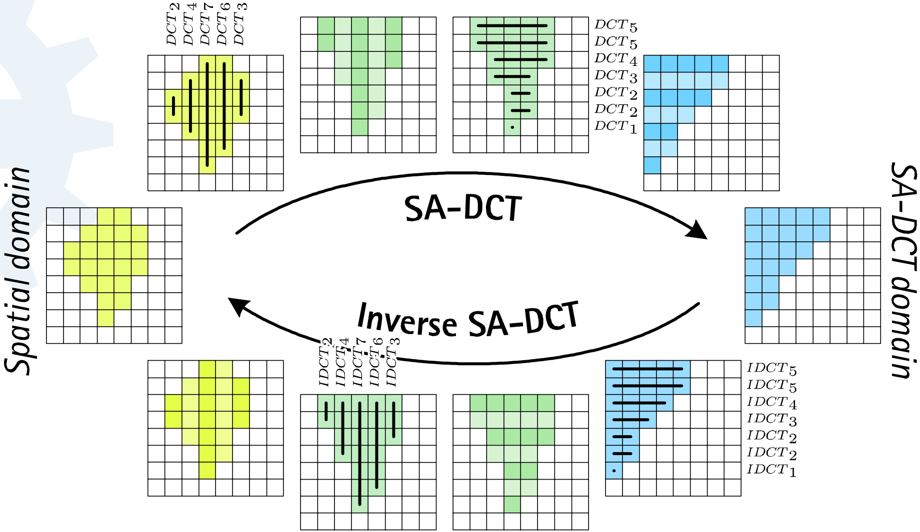


Fig. Illustration of the PCs (listed by decreasing eigenvalue magnitude) for two adaptive-shape neighborhoods. The green overlay shows the grouped similar neighborhoods.

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)

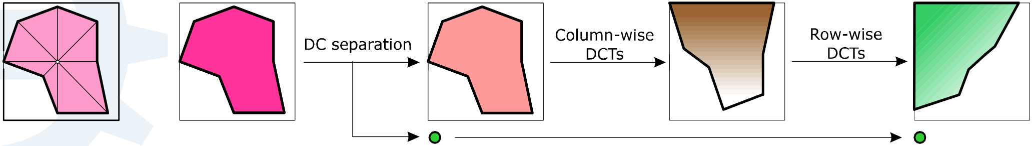
- direct generalization of the classical block-DCT (B-DCT);
- on rectangular domains (e.g., squares) the SA-DCT and B-DCT coincide;
- the same computational complexity as the B-DCT (separable);
- SA-DCT is part of the MPEG-4 standard;
- efficient (low-power) hardware implementations available;

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

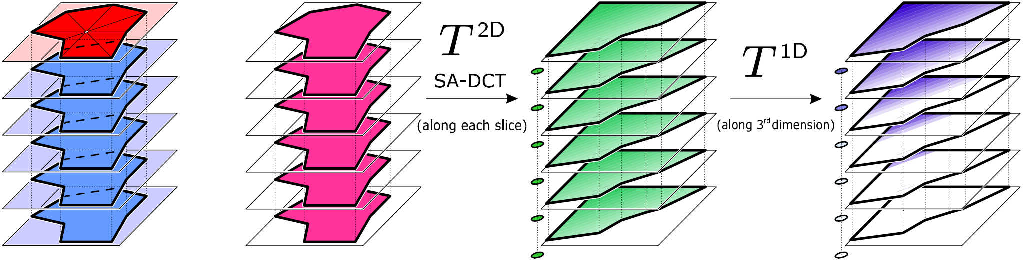
Orthonormal SA-DCT does not have a DC term and works best if applied on zero-mean data: "Orthonormal SA-DCT with DC separation and ΔDC compensation", Kauff et al. 1997.

SA-DCT (forward transform)

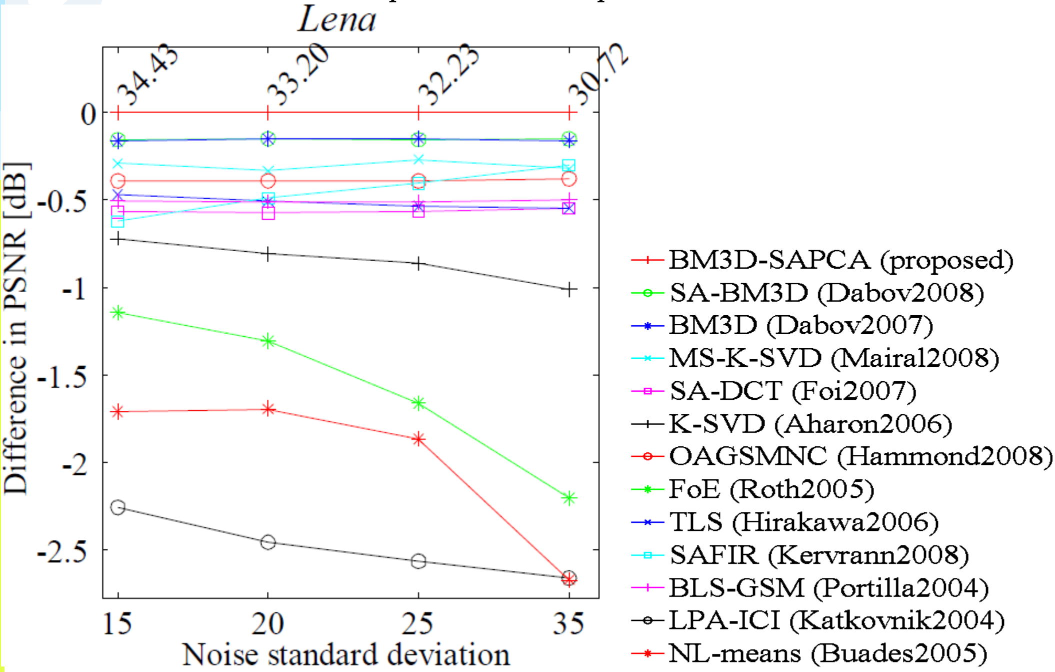
[as used in Pointwise SA-DCT denoising algorithm (Foi et al., IEEE TIP 2007)]



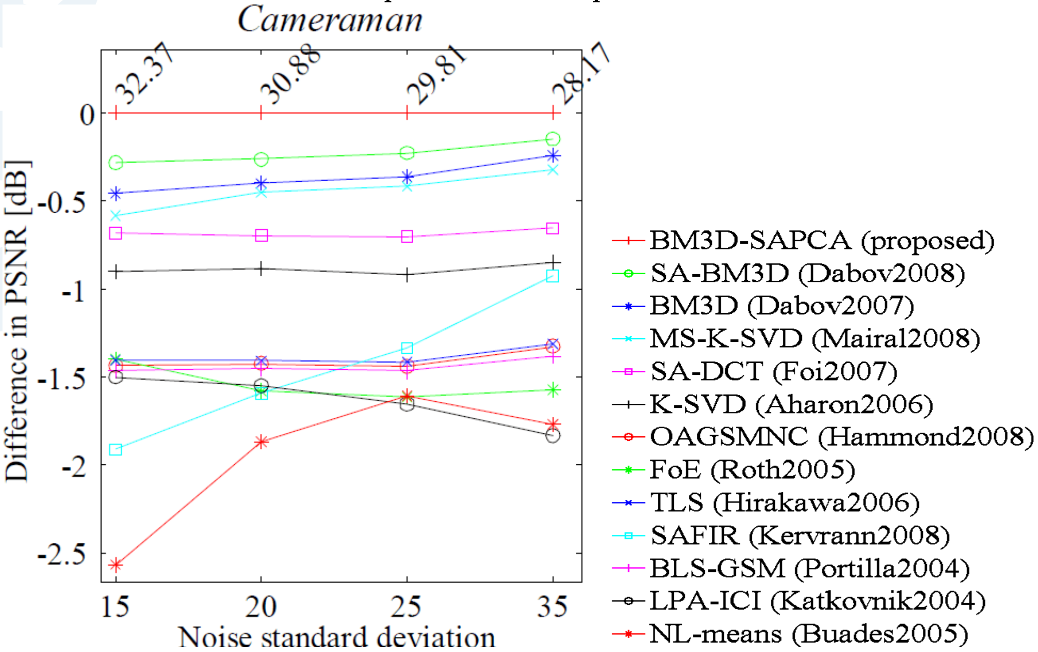
Shape-adaptive collaborative filtering (forward transform)

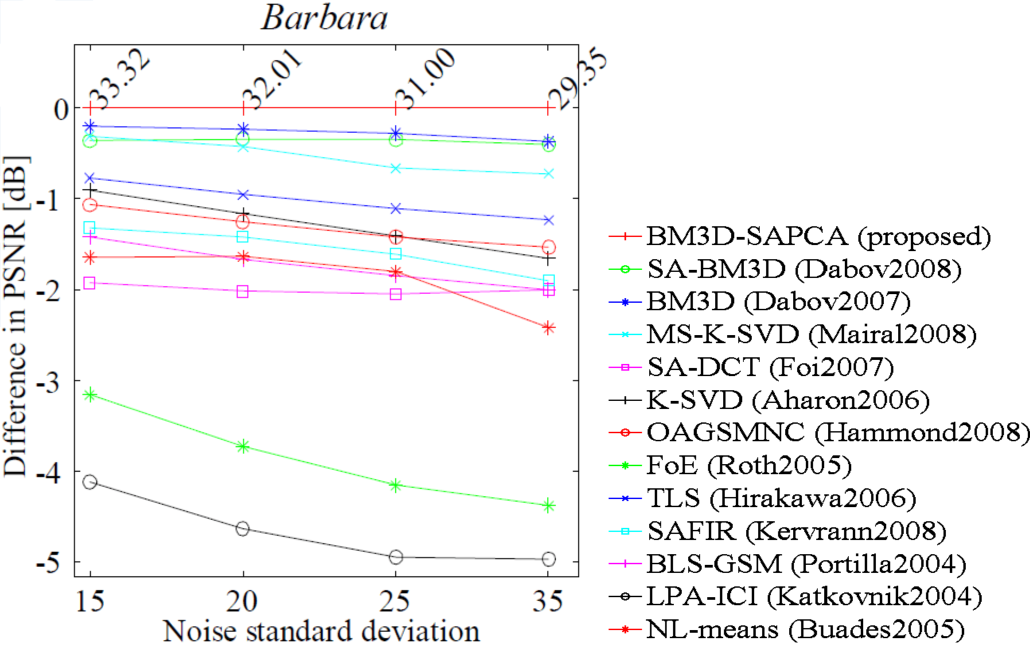


Experimental comparison

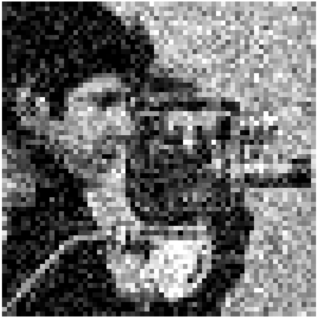


Experimental comparison




Barbara
33.32
32.01
31.00
29.35
0
-1
-2
-3
-4
-5
Difference in PSNR [dB]
15
20
25
35
Noise standard deviation
BM3D-SAPCA (proposed)
SA-BM3D (Dabov2008)
BM3D (Dabov2007)
MS-K-SVD (Mairal2008)
SA-DCT (Foi2007)
K-SVD (Aharon2006)
OAGSMNC (Hammond2008)
FoE (Roth2005)
TLS (Hirakawa2006)
SAFIR (Kervrann2008)
BLS-GSM (Portilla2004)
LPA-ICI (Katkovnik2004)
NL-means (Buades2005)

Original



Noisy, $\sigma = 35$

BM3D (27.82, 0.8207)

P.SADCT (27.51, 0.8143)

SA-BM3D (28.02, 0.8228)

BM3D-SAPCA (28.16, 0.8269)

Independent benchmarking of BM3D denoising performance

Perceptual/Subjective

- Van der Weken, D., E. Kerre, E. Vansteenkiste, and W. Philips, "Evaluation of fuzzy image quality measures using a multidimensional scaling framework", *Proc. 2nd Int. Workshop Video Process. Quality Metrics Consum. Electron., VPQM2006*, Scottsdale, AZ, Jan. 2006.
- Vansteenkiste, E., D. Van der Weken, W. Philips, and E. Kerre, "Perceived image Quality Measurement of state-of-the-art Noise Reduction Schemes", *Lecture Notes in Computer Science 4179 - ACIVS 2006*, pp. 114-124, Springer, Sep. 2006.

PSNR & SSIM

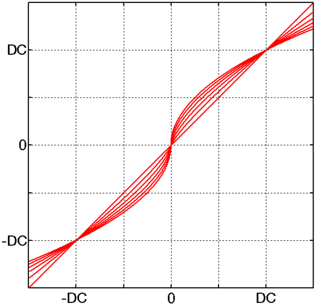
- Lansel, S., D. Donoho, and T. Weissman, "DenoiseLab: A Standard Test Set and Evaluation Method to Compare Denoising Algorithms", `http://www.stanford.edu/~slansel/DenoiseLab/`.
- F. Estrada, D. Fleet, and A. Jepson, `http://www.cs.utoronto.ca/~strider/Denoise/Benchmark/`

2. Extensions and applications

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

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

Aggregation weights for sharpening

Variance of sharpened coefficients (using first order approximations)

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

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

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

Hence, aggregation weights are

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

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

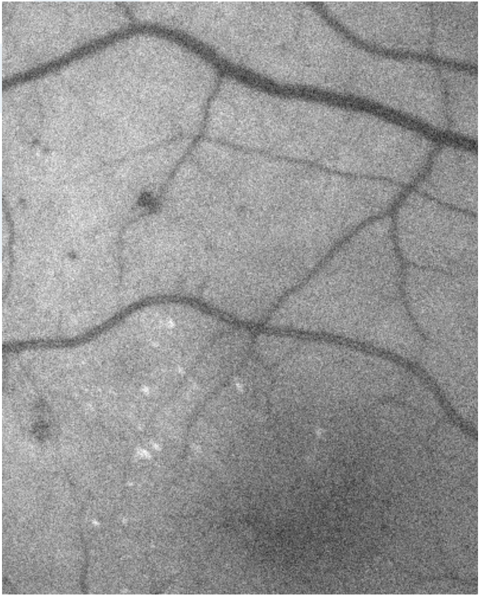
BM3D-SH3D, $\alpha = 1.8$

Collaborative sharpening: experiments

BM3D-SH3D, $\alpha = 2.0$

Collaborative sharpening: experiments

Noisy *Fundus* $\sigma = 20$



BM3D-SH3D

BM3D Deconvolution (non blind)

Approach:

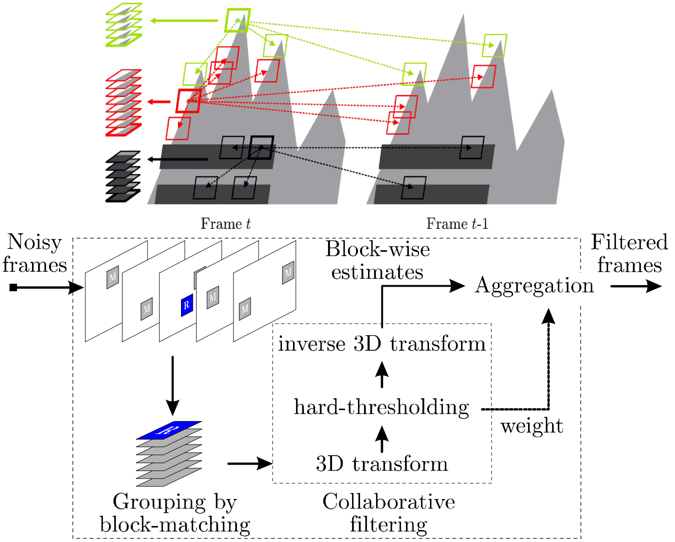
standard Tikhonov regularized deconvolution coupled with BM3D regularization
(in practice the filtering is equivalent to colored noise removal)

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.

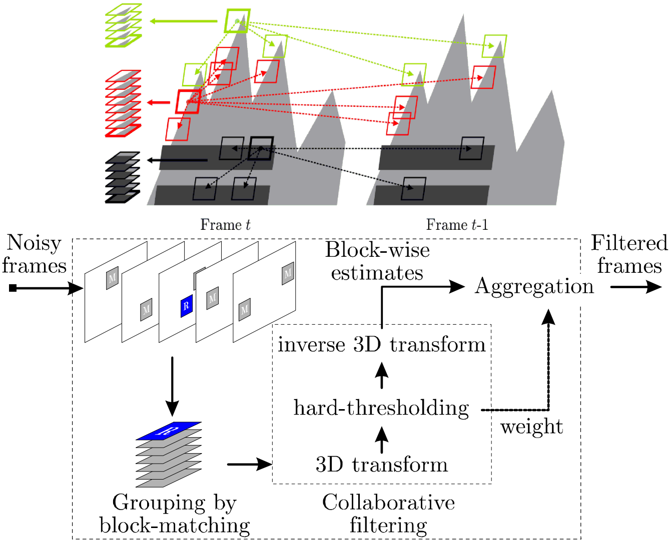
A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-adaptive DCT for denoising and image reconstruction", *Proc. SPIE El. Imaging 2006, Image Process.: Algorithms and Systems V*, 6064A-18, San Jose (CA), USA, January 2006.

V-BM3D for video and multiframe data



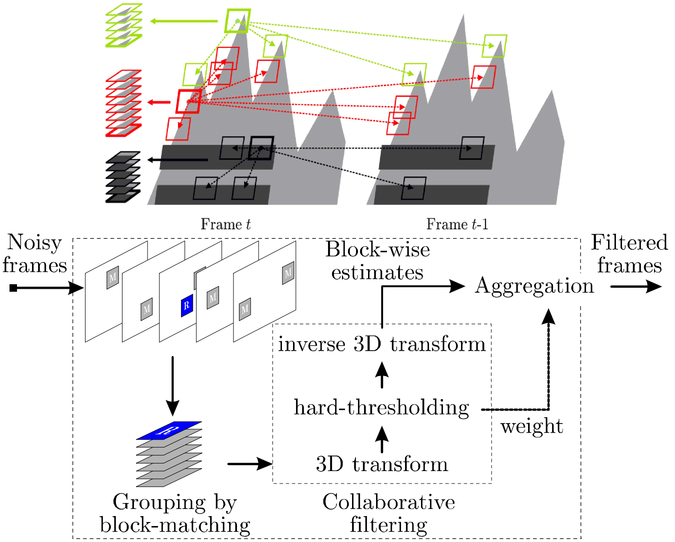
1a. *Grouping.* Searching within all images in the sequence, find blocks that are similar to the currently processed one, and then stack them together in a 3-D array (group).

V-BM3D for video and multiframe data



1b. *Collaborative hard-thresholding.* Apply a 3-D transform to the group, attenuate noise by hard-thresholding of the spectrum, invert the 3-D transform to produce estimates of all grouped blocks, and return the estimates of the blocks to their original place.

V-BM3D for video and multiframe data



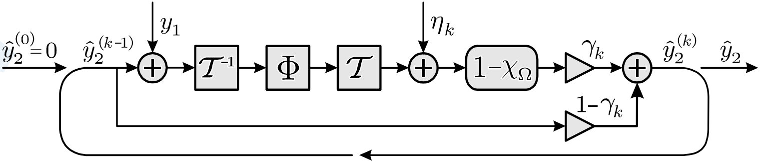
2. *Aggregation.* Compute the estimates of the output images by weighted averaging all of the obtained block-wise estimates that are overlapping.

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



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

$\mathcal{T}$	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

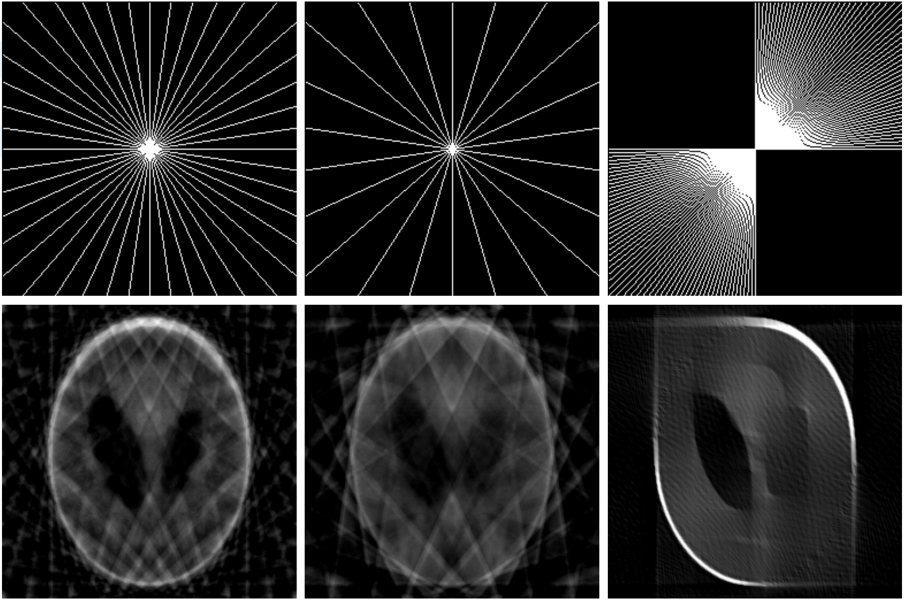
$\mathcal{T} = \mathcal{F}$ Fourier transform

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

11 radial lines | 11 radial line | limited angle (61 lines)

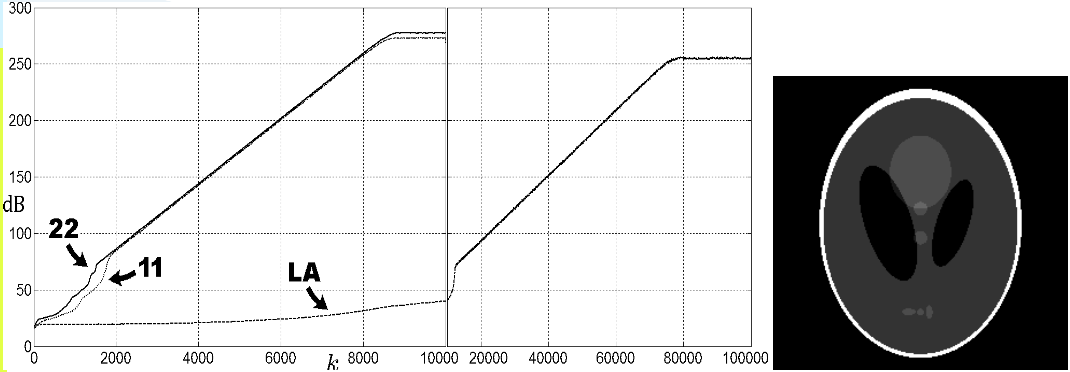
χ_Ω

image (b.p.)



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

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



Towards image upsampling

χ_Ω

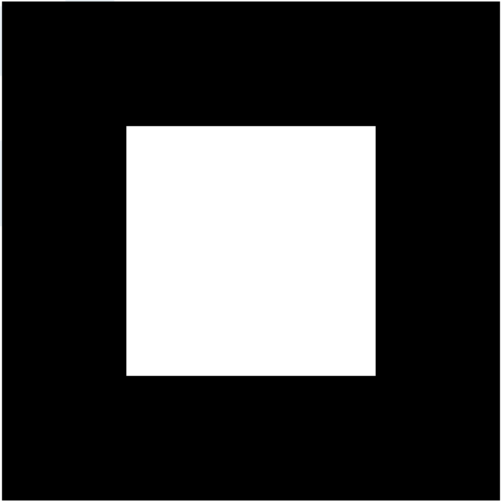


image (b.p.)

Towards image upsampling

extrapolating missing high-frequencies

Upsampling and super-resolution

Image ***upsampling*** or ***zooming***, can be defined as the process of resampling a *single* low-resolution (LR) image on a high-resolution grid. Different resampling methods can be used to obtain zoomed images with specific desired properties, such as edge preservation, degree of smoothness, etc.

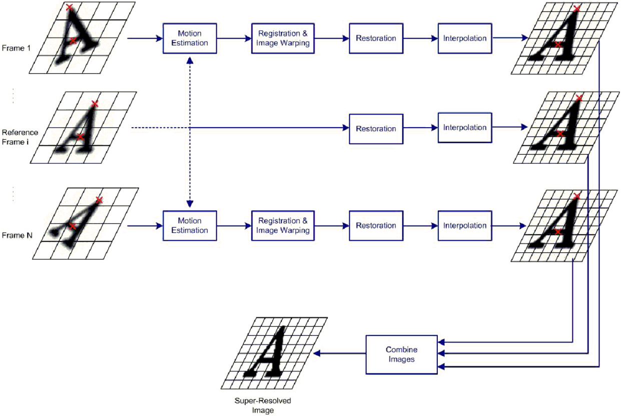
However, fine details missing or distorted in the low-resolution image cannot be reconstructed in the upsampled one. There is no sufficient information in the LR image to do this.

When a *number* of LR images portraying slightly different views of the same scene are available, the reconstruction algorithm can try to improve the spatial resolution by incorporating into the final HR result the additional new details revealed in each LR image.

The process of combining a sequence of undersampled and degraded low-resolution images in order to produce a single high-resolution image is commonly referred to as a

Super-resolution (SR) reconstruction.

Classical approach and problem of registration



From NLM to V-BM3D

Modern SR methods (e.g., Protter et al. 2008, Ebrahimi and Vrscay 2008) are based on the nonlocal means (NLM) filtering paradigm (Buades-Coll-Morel, 2005).

No explicit registration: **one-to-one** pixel mapping between frames is replaced by a **one-to-many** mapping.

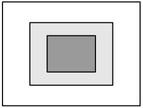
Multiple pixels can be assigned to a given one, with weights typically defined by the similarity of the patches/blocks surrounding the pixels. The HR image is estimated through a weighted average of these multiple pixels (or of their surrounding patches) with their corresponding weights.

The BM3D and V-BM3D (Dabov et al. 2007) algorithms share with the NLM the idea of exploiting nonlocal similarity between blocks. However, in (V-)BM3D a more powerful transform-domain modeling is used. These turn out to be a much more effective filter than the NLM not only for denoising, but also for super-resolution.

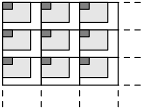
M. Ebrahimi, E. R. Vrscay, "Multi-frame super-resolution with no explicit motion estimation", *Proc. Int. Conf. on Image Process., Computer Vision, and Pattern Recognition, IPCV 2008*, Las Vegas, Nevada, USA, July. 2008.

M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the Non-Local-Means to Super-Resolution Reconstruction", *IEEE Trans. Image Process.*, vol. 18, no. 8, pp. 1899-1904, Jan. 2009.

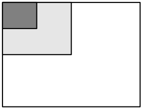
Preliminaries: scaling family of orthonormal transforms



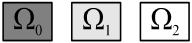
(a) DFT



(b) Block DCT



(c) DWT, DCT



$\{\mathcal{T}_m\}_{m=0}^{M}$ family of orthonormal transforms of increasing sizes

$$x_m^{\mathrm{h}} \times x_m^{\mathrm{v}}, \qquad x_m^{\mathrm{h}} < x_{m'}^{\mathrm{h}}, \quad x_m^{\mathrm{v}} < x_{m'}^{\mathrm{v}} \qquad \forall m, m', \quad m < m'$$

the whole $\mathcal{T}_m$-spectrum can be considered as a smaller portion of the $\mathcal{T}_{m'}$-spectrum.

Supports Ω_m of the $\mathcal{T}_m$-transform coefficients form a *nested sequence* of subsets of Ω_M:

$$\Omega_0 \subset \cdots \subset \Omega_M .$$

Examples: DCT, DFT,
DWT associated to one common scaling function,
block DCT, DFT and DWT.

Notation

For $m < m'$ we define three operators:

- the ***restriction operator*** $|_{\Omega_{m,m'}}$ - given $\mathcal{T}_{m'}$-spectrum, extracts smaller portion defined on Ω_m, which can be considered as the $\mathcal{T}_m$-spectrum of a smaller image;

- the ***zero-padding operator*** $\mathcal{U}_{m,m'}$ - expands a $\mathcal{T}_m$-spectrum defined on Ω_m to the $\mathcal{T}_{m'}$-spectrum defined on the superset $\Omega_{m'} \supset \Omega_m$ by introducing zeros in the complementary $\Omega_{m'} \setminus \Omega_m$;

  $\mathcal{U}_{m,m'}$ is "dual" operator of $|_{\Omega_{m,m'}}$

- the ***projection operator*** $\mathcal{P}^{\perp}_{m,m'}$ - zeroes out all coefficients of $\mathcal{T}_{m'}$-spectrum defined on Ω_m.

$\mathcal{U}_{m,m'}(A)|_{\Omega_m} = A$ for any $\mathcal{T}_m$-spectrum A

$B = \mathcal{P}^{\perp}_{m,m'}(B) + \mathcal{U}_{m,m'}\left(B|_{\Omega_m}\right)$ for any $\mathcal{T}_{m'}$-spectrum B.

Observation model

Given:

sequence of R low-resolution images $\{y_{\text{low } r}\}_{r=1}^{R}$ of size $x_0^{\text{h}} \times x_0^{\text{v}}$

$$y_{\text{low } r} = \mathcal{T}_0^{-1}\left(\beta_{0,M}^{-1}\ \mathcal{T}_M\left(y_{\text{hi } r}\right)\big|_{\Omega_{0,M}}\right), \tag{1}$$

SR reconstruction problem:

to reconstruct $\{y_{\text{hi } r}\}_{r=1}^{R}$ from $\{y_{\text{low } r}\}_{r=1}^{R}$.

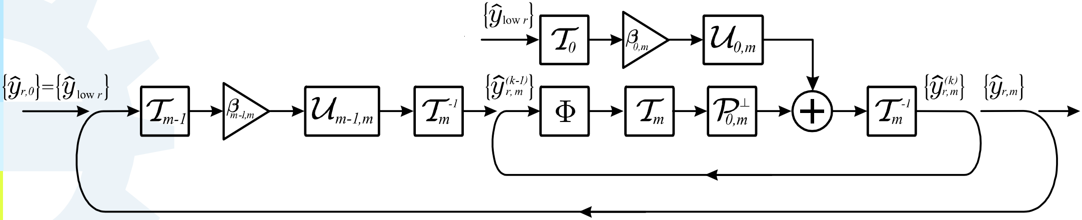
Constraint:

for a fixed r, an estimate $\hat{y}_r$ of $y_{\text{hi } r}$ must have its Ω_0 subband equal to $\beta_{0,M}\mathcal{T}_0\left(y_{\text{low } r}\right) = \mathcal{T}_M\left(y_{\text{hi } r}\right)\big|_{\Omega_{0,M}}$.

Remark:

$R = 1$ gives image *upsampling* problem

Multistage iterative reconstruction



$$\begin{cases} \hat{y}_{r,0} = y_{\mathrm{low}\ r} & \texttt{(algorithm input)} \\ \hat{y}_{r,m} = \hat{y}_{r,m}^{(k_{\mathrm{final}\ m})} & \texttt{(stage input)} \\ \hat{y}_{r,m}^{(0)} = \mathcal{T}_m^{-1}\left(\mathcal{U}_{m-1,m}\left(\beta_{m-1,m}\mathcal{T}_{m-1}\left(\hat{y}_{r,m-1}\right)\right)\right) \\ \hat{y}_{r,m}^{(k)} = \mathcal{T}_m^{-1}\left(\mathcal{U}_{0,m}\left(\beta_{0,m}\mathcal{T}_0\left(y_{\mathrm{low}\ r}\right)\right) + \mathcal{P}_{0,m}^{\perp}\left(\mathcal{T}_m\left(\Phi\left(r, \left\{\hat{y}_{r,m}^{(k-1)}\right\}_{r=1}^{R}, \sigma_{k,m}\right)\right)\right)\right) \end{cases} \quad (2)$$

m	stage number	$m = 1, \ldots, M$
k	iteration number	$k = 0, \ldots, k_{\mathrm{final}\ m}$
$\hat{y}_{r,m}^{(k)}$	estimate for $\hat{y}_r$ on iter. k of stage m	
$\mathcal{T}_m$	transform	
Φ	spatially adaptive filter (V-BM3D)	
$\sigma_{k,m}$	parameter controlling the strength of the filter	$\sigma_{k,m} = \sigma_{k,m-1} - \Delta_m$

Image upsampling $4\times$ in wavelet domain (Danielyan et al. EUSIPCO 2008)

Video super-resolution

Results for the 23rd frame from the *Foreman* sequence. From left to right and from top to bottom: pixel-replicated low-resolution image; original image (ground truth); super-resolved by the algorithm by Protter et al.; super-resolved by the proposed algorithm.

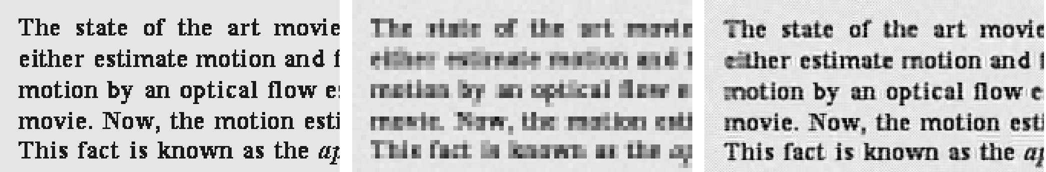
Video super-resolution

Results for the 23rd frame from the *Suzie* sequence. From left to right and from top to bottom: pixel-replicated low-resolution image; original image (ground truth); super-resolved by the algorithm by Protter et al.; super-resolved by the proposed algorithm.

Video super-resolution

Results for the 23rd frame from the *Miss America* sequence. From left to right and from top to bottom: pixel-replicated low-resolution image; original image (ground truth); super-resolved by the algorithm by Protter et al.; super-resolved by the proposed algorithm.

Image super-resolution



Super-resolution result for the *Text* image. From left to right: original high-resolution image (ground truth); pixel-replicated low-resolution image; image super-resolved by the proposed algorithm.

References:

`http://sp.cs.tut.fi/groups/trans`

- K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering", *IEEE Trans. Image Process.*, vol. 16, no. 8, Aug. 2007.
- K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Joint image sharpening and denoising by 3D transform-domain collaborative filtering," *Proc. 2007 Int. TICSP Workshop Spectral Meth. Multirate Signal Process., SMMSP 2007*, Moscow, Russia, September 2007.
- K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image restoration by sparse 3D transform-domain collaborative filtering, " *Proc. SPIE Electronic Imaging '08*, vol. 6812, no. 6812-1D, San Jose (CA), USA, January 2008.
- K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D Image Denoising with Shape-Adaptive Principal Component Analysis", *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'09)*, Saint-Malo, France, April 2009.
- V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From local kernel to nonlocal multiple-model image denoising", *Int. J. Computer Vision*, vol. 86, no. 1, pp. 1-32, January 2010.

- K. Egiazarian, 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, Sep. 2007.
- A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Image upsampling via spatially adaptive block-matching filtering", *Proc. of 16th European Signal Processing Conference, EUSIPCO2008,* Lausanne, Switzerland, Aug. 2008.
- A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Image and video super-resolution via spatially adaptive block-matching filtering", *Proc. Int. Workshop on Local and Non-Local Approx. in Image Process., LNLA 2008*, Lausanne, Switzerland, August 2008.
- A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Spatially adaptive filtering as regularization in inverse imaging: compressive sensing, upsampling, and super-resolution", in *Super-Resolution Imaging* (P. Milanfar, ed.), CRC Press / Taylor & Francis, 2010. ISBN: 978-1-4398-1930-2. (in press)

Thank you!