Image denoising with block-matching and 3D filtering

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, Karen Egiazarian

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, Karen Egiazarian, "Image denoising with block-matching and 3D filtering," Proc. SPIE 6064, Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning, 606414 (17 February 2006); doi: 10.1117/12.643267

SPIE.

Event: Electronic Imaging 2006, 2006, San Jose, California, United States

Image denoising with block-matching and 3D filtering

Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian
Institute of Signal Processing, Tampere University of Technology, Finland
PO BOX 553, 33101 Tampere, Finland
firstname.lastname@tut.fi

ABSTRACT

We present a novel approach to still image denoising based on effective filtering in 3D transform domain by combining sliding-window transform processing with block-matching. We process blocks within the image in a sliding manner and utilize the block-matching concept by searching for blocks which are similar to the currently processed one. The matched blocks are stacked together to form a 3D array and due to the similarity between them, the data in the array exhibit high level of correlation. We exploit this correlation by applying a 3D decorrelating unitary transform and effectively attenuate the noise by shrinkage of the transform coefficients. The subsequent inverse 3D transform yields estimates of all matched blocks. After repeating this procedure for all image blocks in sliding manner, the final estimate is computed as weighed average of all overlapping block-estimates. A fast and efficient algorithm implementing the proposed approach is developed. The experimental results show that the proposed method delivers state-of-art denoising performance, both in terms of objective criteria and visual quality.

Keywords: image denoising, block-matching, 3D transforms

1. INTRODUCTION

Much of the recent research on image denoising has been focused on methods that reduce noise in transform domain. Starting with the milestone work of Donoho,[1,2] many of the later techniques[3–7] performed denoising in wavelet transform domain. Of these methods, the most successful proved to be the ones[4,5,7] based on rather sophisticated modeling of the noise impact on the transform coefficients of overcomplete multiscale decompositions. Not limited to multiscale techniques, the overcomplete representations have traditionally played a significant role in improving the restoration abilities of even the most basic transform-based methods. This is manifested by the sliding-window transform denoising,[8,9] where the basic idea is to successively denoise overlapping blocks by coefficient shrinkage in local 2D transform domain (e.g. DCT, DFT, etc.). Although the transform-based approaches deliver very good overall performance in terms of objective criteria, they fail to preserve details which are not suitably represented by the used transform and often introduce artifacts that are characteristic of this transform.

A different denoising strategy based on non-local estimation appeared recently,[10,11] where a pixel of the true image is estimated from regions which are found similar to the region centered at the estimated pixel. These methods, unlike the transform-based ones, introduce very few artifacts in the estimates but often oversmooth image details. Based on an elaborate adaptive weighting scheme, the exemplar-based denoising[10] appears to be the best of them and achieves results competitive to the ones produced by the best transform-based techniques.

The concept of employing similar data patches from different locations is popular in the video processing field under the term of "block-matching", where it is used to improve the coding efficiency by exploiting similarity among blocks which follow the motion of objects in consecutive frames. Traditionally, block-matching has found successful application in conjunction with transform-based techniques. Such applications include video compression (MPEG standards) and also video denoising,[12] where noise is attenuated in 3D DCT domain.

We propose an original image denoising method based on effective filtering in 3D transform domain by combining sliding-window transform processing with block-matching. We undertake the block-matching concept for a single noisy image; as we process image blocks in a sliding manner, we search for blocks that exhibit similarity to the currently-processed one. The matched blocks are stacked together to form a 3D array. In this manner,

Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning
edited by E.R. Dougherty, J.T. Astola, K.O. Egiazarian, N.M. Nasrabadi, S.A. Rizvi
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6064, 606414, © 2006 SPIE-IS&T · 0277-786X/06/$15

we induce high correlation along the dimension of the array in which the blocks are stacked. We exploit this correlation by applying a 3D decorrelating unitary transform which produces a sparse representation of the true signal in 3D transform domain. Efficient noise attenuation is done by applying a shrinkage operator (e.g. hard-thresholding or Wiener filtering) on the transform coefficients. This results in improved denoising performance and effective detail preservation in the local estimates of the matched blocks, which are reconstructed by an inverse 3D transform of the filtered coefficients. After processing all blocks, the final estimate is the weighted average of all overlapping local block-estimates. Because of overcompleteness which is due to the overlap, we avoid blocking artifacts and further improve the estimation ability.

Although the proposed approach is general with respect to the type of noise, for simplicity of exposition, we restrict our attention to the problem of attenuating additive white Gaussian noise (AWGN).

The basic approach and its extension to Wiener filtering are presented in Sections 2 and 3, respectively. An efficient algorithm which implements the proposed approach is developed in Section 4. Finally, Section 5 is devoted to demonstration and discussion of experimental results.

2. DENOISING BY SHRINKAGE IN 3D TRANSFORM DOMAIN WITH BLOCK-MATCHING

Let us introduce the observation model and notation used throughout the paper. We consider noisy observations $z : X \to \mathbb{R}$ of the form $z(x) = y(x) + \eta(x)$, where $x \in X$ is a 2D spatial coordinate that belongs to the image domain $X \subset \mathbb{Z}^2$, y is the true image, and $\eta(x) \sim \mathcal{N}\left(0, \sigma^2\right)$ is white Gaussian noise of variance σ^2. By Z_x we denote a block of fixed size $N_1 \times N_1$ extracted from z, which has $z(x)$ as its upper-left element; alternatively, we say that Z_x is located at x. With $\widehat{y}$ we designate the final estimate of the true image.

Let us state the used assumptions. We assume that some of the blocks (of fixed size $N_1 \times N_1$) of the true image exhibit mutual correlation. We also assume that the selected unitary transform is able to represent sparsely these blocks. However, the diversity of such blocks in natural images often makes the latter assumption unsatisfied in 2D transform domain and fulfilled only in 3D transform domain due to the correlation introduced by block-matching. The standard deviation σ of the AWGN can be accurately estimated (e.g.[1]), therefore we assume its a-priori knowledge.

2.1. Local Estimates

We successively process all overlapping blocks of fixed size in a sliding manner, where "process" stands for the consecutive application of block-matching and denoising in local 3D transform domain. For the sub-subsections to follow, we fix the currently processed block as Z_{x_R}, where $x_R \in X$, and denominate it as "reference block".

2.1.1. Block-matching

Block-matching is employed to find blocks that exhibit high correlation to Z_{x_R}. Because its accuracy is significantly impaired by the presence of noise, we utilize a block-similarity measure which performs a coarse initial denoising in local 2D transform domain. Hence, we define a block-distance measure (inversely proportional to similarity) as

$$d(Z_{x_1}, Z_{x_2}) = N_1^{-1} \left\| \Upsilon\left(\mathcal{T}_{2D}(Z_{x_1}), \lambda_{thr2D}\sigma\sqrt{2\log\left(N_1^2\right)}\right) - \Upsilon\left(\mathcal{T}_{2D}(Z_{x_2}), \lambda_{thr2D}\sigma\sqrt{2\log\left(N_1^2\right)}\right) \right\|_2, \quad (1)$$

where $x_1, x_2 \in X$, $\mathcal{T}_{2D}$ is a 2D linear unitary transform operator (e.g. DCT, DFT, etc.), Υ is a hard-threshold operator, λ_{thr2D} is fixed threshold parameter, and $\|\cdot\|_2$ denotes the L^2-norm. Naturally, Υ is defined as

$$\Upsilon(\lambda, \lambda_{thr}) = \begin{cases} \lambda, & \text{if } |\lambda| > \lambda_{thr} \\ 0, & \text{otherwise.} \end{cases}$$

The result of the block-matching is a set $\mathcal{S}_{x_R} \subseteq X$ of the coordinates of the blocks that are similar to Z_{x_R} according to our d-distance (1); thus, $\mathcal{S}_{x_R}$ is defined as

$$\mathcal{S}_{x_R} = \{x \in X \mid d(Z_{x_R}, Z_x) < \tau_{match}\}, \quad (2)$$

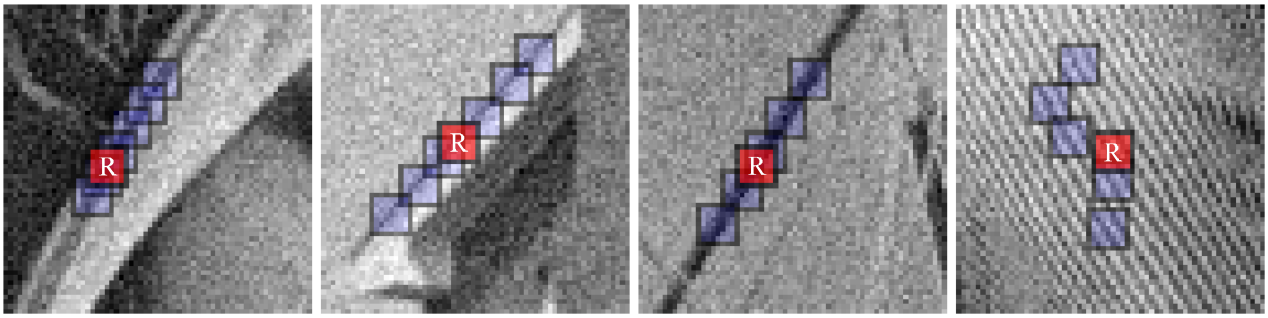


Figure 1. Fragments of *Lena, House, Boats* and *Barbara* corrupted by AWGN of $\sigma = 15$. For each fragment block-matching is illustrated by showing a reference block marked with 'R' and a few of its matched ones.

where τ_{match} is the maximum d-distance for which two blocks are considered similar. Obviously $d\left(Z_{x_R}, Z_{x_R}\right) = 0$, which implies that $|\mathcal{S}_{x_R}| \geq 1$, where $|\mathcal{S}_{x_R}|$ denotes the cardinality of $\mathcal{S}_{x_R}$.

The matching procedure in presence of noise is demonstrated on Figure 1, where we show a few reference blocks and the ones matched as similar to them.

2.1.2. Denoising in 3D transform domain

We stack the matched noisy blocks $Z_{x \in \mathcal{S}_{x_R}}$ (ordering them by increasing d-distance to Z_{x_R}) to form a 3D array of size $N_1 \times N_1 \times |\mathcal{S}_{x_R}|$, which is denoted by $\mathbf{Z}_{\mathcal{S}_{x_R}}$. We apply a unitary 3D transform $\mathcal{T}_{3D}$ on $\mathbf{Z}_{\mathcal{S}_{x_R}}$ in order to attain sparse representation of the true signal. The noise is attenuated by hard-thresholding the transform coefficients. Subsequently, the inverse transform operator $\mathcal{T}_{3D}^{-1}$ yields a 3D array of reconstructed estimates

$$\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}} = \mathcal{T}_{3D}^{-1}\left(\Upsilon\left(\mathcal{T}_{3D}\left(\mathbf{Z}_{\mathcal{S}_{x_R}}\right), \lambda_{thr3D}\sigma\sqrt{2\log\left(N_1^2\right)}\right)\right), \tag{3}$$

where λ_{thr3D} is a fixed threshold parameter. The array $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ comprises of $|\mathcal{S}_{x_R}|$ stacked local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ of the true image blocks located at $x \in \mathcal{S}_{x_R}$. We define a weight for these local estimates as

$$\omega_{x_R} = \begin{cases} \frac{1}{N_{har}}, & \text{if } N_{har} \geq 1 \\ 1, & \text{otherwise,} \end{cases} \tag{4}$$

where N_{har} is the number of non-zero transform coefficients after hard-thresholding. Observe that $\sigma^2 N_{har}$ is equal* to the total variance of $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$. Thus, sparser decompositions of $\mathbf{Z}_{\mathcal{S}_{x_R}}$ result in less noisy estimates which are awarded greater weights by (4).

2.2. Estimate Aggregation

After processing all reference blocks, we have a set of local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$, $\forall x_R \in X$ (and their corresponding weights ω_{x_R}, $\forall x_R \in X$), which constitute an overcomplete representation of the estimated image due to the overlap between the blocks. It is worth mentioning that a few local block estimates might be located at the same coordinate (e.g. $\widehat{Y}^{x_a}_{x_b}$ and $\widehat{Y}^{x_b}_{x_b}$ are both located at x_b but obtained while processing the reference blocks at x_a and x_b, respectively). Let $\widehat{Y}^{x_R}_{x_m}(x)$ be an estimate of $y(x)$, where $x, x_R \in X$, and $x_m \in \mathcal{S}_{x_R}$. We zero-extend $\widehat{Y}^{x_R}_{x_m}(x)$ outside its square support in order to simplify the formulation. The final estimate $\widehat{y}$ is computed as a weighted average of all local ones as given by

$$\widehat{y}(x) = \frac{\sum_{x_R \in X}\sum_{x_m \in \mathcal{S}_{x_R}} \omega_{x_R}\widehat{Y}^{x_R}_{x_m}(x)}{\sum_{x_R \in X}\sum_{x_m \in \mathcal{S}_{x_R}} \omega_{x_R}\chi_{x_m}(x)}, \ \forall x \in X, \tag{5}$$

*Equality holds only if the matched blocks that build $\mathbf{Z}_{\mathcal{S}_{x_R}}$ are non-overlapping; otherwise, a certain amount of correlation is introduced in the noise.

where $\chi_{x_m} : X \to \{0, 1\}$ is the characteristic function of the square support of a block located at $x_m \in X$.

One can expect substantially overcomplete representation of the signal in regions where a block is matched to many others. On the other hand, if a match is not found for a given reference block, the method reduces to denoising in 2D transform domain. Thus, the overcomplete nature of the method is highly dependent on the block-matching and therefore also on the particular noisy image.

3. WIENER FILTER EXTENSION

Provided that an estimate of the true image is available (e.g. it can be obtained from the method given in the previous section), we can construct an empirical Wiener filter as a natural extension of the above thresholding technique. Because it follows the same approach, we only give the few fundamental modifications that are required for its development and thus omitting repetition of the concept. Let us denote the initial image estimate by $e : X \to \mathbb{R}$. In accordance with our established notation, E_x designates a square block of fixed size $N_1 \times N_1$, extracted from e and located at $x \in X$.

3.1. Modification to Block-Matching

In order to improve the accuracy of block-matching, it is performed within the initial estimate e rather than the noisy image. Accordingly, we replace the thresholding-based d-distance measure from (1) with the normalized L^2-norm of the difference of two blocks with subtracted means. Hence, the definition (2) of $\mathcal{S}_{x_R}$ becomes

$$\mathcal{S}_{x_R} = \left\{ x \in X \mid N_1^{-1} \left\| \left(E_{x_R} - \overline{E_{x_R}}\right) - \left(E_x - \overline{E_x}\right) \right\|_2 < \tau_{match} \right\}, \tag{6}$$

where $\overline{E_{x_R}}$ and $\overline{E_x}$ are the mean values of the blocks E_{x_R} and E_x, respectively. The mean subtraction allows for improved matching of blocks with similar structures but different mean values.

3.2. Modification to Denoising in 3D Transform Domain

The linear Wiener filter replaces the nonlinear hard-thresholding operator. The attenuating coefficients for the Wiener filter are computed in 3D transform domain as

$$\mathbf{W}_{\mathcal{S}_{x_R}} = \frac{\left| \mathcal{T}_{3D}\left(\mathbf{E}_{\mathcal{S}_{x_R}}\right) \right|^2}{\left| \mathcal{T}_{3D}\left(\mathbf{E}_{\mathcal{S}_{x_R}}\right) \right|^2 + \sigma^2},$$

where $\mathbf{E}_{\mathcal{S}_{x_R}}$ is a 3D array built by stacking the matched blocks $E_{x \in \mathcal{S}_{x_R}}$ (in the same manner as $\mathbf{Z}_{\mathcal{S}_{x_R}}$ is built by stacking $Z_{x \in \mathcal{S}_{x_R}}$). We filter the 3D array of noisy observations $\mathbf{Z}_{\mathcal{S}_{x_R}}$ in $\mathcal{T}_{3D}$-transform domain by an elementwise multiplication with $\mathbf{W}_{\mathcal{S}_{x_R}}$. The subsequent inverse transform gives

$$\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}} = \mathcal{T}_{3D}^{-1}\left(\mathbf{W}_{\mathcal{S}_{x_R}} \mathcal{T}_{3D}\left(\mathbf{Z}_{\mathcal{S}_{x_R}}\right)\right), \tag{7}$$

where $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ comprises of stacked local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ of the true image blocks located at the matched locations $x \in \mathcal{S}_{x_R}$. As in (4), the weight assigned to the estimates is inversely proportional to the total variance of $\widehat{\mathbf{Y}}_{\mathcal{S}_{x_R}}$ and defined as

$$\omega_{x_R} = \left(\sum_{i=1}^{N_1} \sum_{j=1}^{N_1} \sum_{t=1}^{|\mathcal{S}_{x_R}|} \left| \mathbf{W}_{\mathcal{S}_{x_R}}(i, j, t) \right|^2 \right)^{-1}. \tag{8}$$

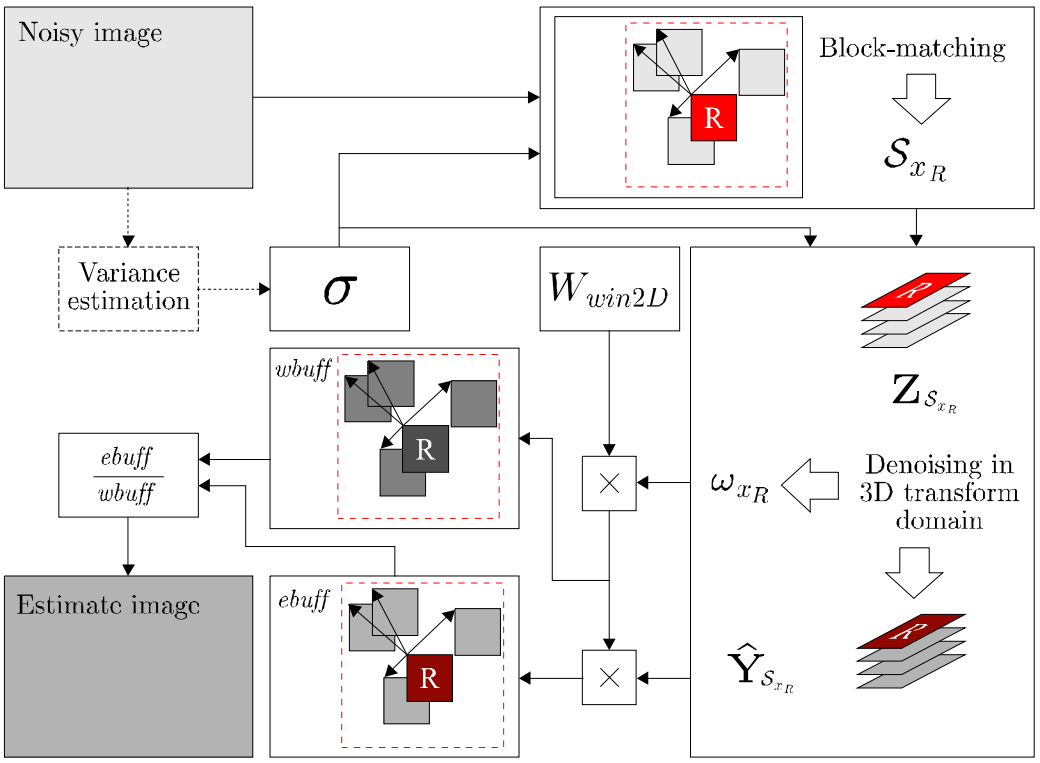


Figure 2. Flowchart for denoising by hard-thresholding in 3D transform domain with block-matching.

4. ALGORITHM

We present an algorithm which employs the hard-thresholding approach (from Section 2) to deliver an initial estimate for the Wiener filtering part (from Section 3) that produces the final estimate. A straightforward implementation of this general approach is computationally demanding. Thus, in order to realize a practical and efficient algorithm, we impose constraints and exploit certain expedients. In this section we introduce these aspects and develop an efficient implementation of the proposed approach.

The choice of the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ is governed by their energy compaction (sparsity) ability for noise-free image blocks (2D) and stacked blocks (3D), respectively. It is often assumed that neighboring pixels in small blocks extracted from natural images exhibit high correlation; thus, such blocks can be sparsely represented by well-established decorrelating transforms, such as the DCT, the DFT, wavelets, etc. From computational efficiency point of view, however, very important characteristics are the separability and the availability of fast algorithms. Hence, the most natural choice for $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ is a fast separable transform which allows for sparse representation of the true-image signal in each dimension of the input array.

4.1. Efficient Image Denoising Algorithm with Block-Matching and 3D Filtering

Let us introduce constraints for the complexity of the algorithm. First, we fix the maximum number of matched blocks by setting an integer N_2 to be the upper bound for the cardinality of the sets $\mathcal{S}_{x_R \in X}$. Second, we do block-matching within a local neighborhood of fixed size $N_S \times N_S$ centered about each reference block, instead of doing it in the whole image. Finally, we use N_{step} as a step by which we slide to every next reference block. Accordingly, we introduce $X_R \subseteq X$ as the set of the reference blocks' coordinates, where $|X_R| \approx \frac{|X|}{N_{step}^2}$(e.g., $N_{step} = 1$ implies $X_R = X$).

In order to reduce the impact of artifacts on the borders of blocks (border effects), we use a Kaiser window W_{win2D} (with a single parameter β) as part of the weights of the local estimates. These artifacts are inherent of

many transforms (e.g. DFT) in presence of sharp intensity differences across the borders of a block.

Let the input noisy image be of size $M \times N$, thus $|X| = MN$. We use two buffers of the same size—*ebuff* for estimates and *wbuff* for weights—to represent the summations in the numerator and denominator, respectively, in (5). For simplicity, we extend our notation so that $ebuff\,(x)$ denotes a single pixel at coordinate $x \in X$ and $ebuff_x$ designates a block located at x in *ebuff* (the same notation is to be used for *wbuff*).

A flowchart of the hard-thresholding part of the algorithm is given in Figure 2 (but we do not give such for the Wiener filtering part since it requires only the few changes given in Section 3). Following are the steps of the image denoising algorithm with block-matching and 3D filtering.

1. **Initialization.** Initialize $ebuff\,(x) = 0$ and $wbuff\,(x) = 0$, for all $x \in X$.
2. **Local hard-thresholding estimates.** For each $x_R \in X_R$, do the following sub-steps.
   (a) *Block-matching.* Compute $\mathcal{S}_{x_R}$ as given in Equation (2) but restrict the search to a local neighborhood of fixed size $N_S \times N_S$ centered about x_R. If $|\mathcal{S}_{x_R}| > N_2$, then let only the coordinates of the N_2 blocks with smallest d-distance to Z_{x_R} remain in $\mathcal{S}_{x_R}$ and exclude the others.
   (b) *Denoising by hard-thresholding in local 3D transform domain.* Compute the local estimate blocks $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ and their corresponding weight ω_{x_R} as given in (3) and (4), respectively.
   (c) *Aggregation.* Scale each reconstructed local block estimate $\widehat{Y}^{x_R}_x$, where $x \in \mathcal{S}_{x_R}$, by a block of weights $W\,(x_R) = \omega_{x_R} W_{win2D}$ and accumulate to the estimate buffer: $ebuff_x = ebuff_x + W\,(x_R)\,\widehat{Y}^{x_R}_x$, for all $x \in \mathcal{S}_{x_R}$. Accordingly, the weight block is accumulated to same locations as the estimates but in the weights buffer: $wbuff_x = wbuff_x + W\,(x_R)$, for all $x \in \mathcal{S}_{x_R}$.
3. **Intermediate estimate.** Produce the intermediate estimate $e\,(x) = \frac{ebuff\,(x)}{wbuff\,(x)}$ for all $x \in X$, which is to be used as initial estimate for the Wiener counterpart.
4. **Local Wiener filtering estimates.** Use e as initial estimate. The buffers are re-initialized: $ebuff\,(x) = 0$ and $wbuff\,(x) = 0$, for all $x \in X$. For each $x_R \in X_R$, do the following sub-steps.
   (a) *Block-matching.* Compute $\mathcal{S}_{x_R}$ as given in (6) but restrict the search to a local neighborhood of fixed size $N_S \times N_S$ centered about x_R. If $|\mathcal{S}_{x_R}| > N_2$, then let only the coordinates of the N_2 blocks with smallest distance (as defined in Subsection 3.1) to E_{x_R} remain in $\mathcal{S}_{x_R}$ and exclude the others.
   (b) *Denoising by Wiener filtering in local 3D transform domain.* The local block estimates $\widehat{Y}^{x_R}_{x \in \mathcal{S}_{x_R}}$ and their weight ω_{x_R} are computed as given in (7) and (8), respectively.
   (c) *Aggregation.* It is identical to step 2c.
5. **Final estimate.** The final estimate is given by $\widehat{y}(x) = \frac{ebuff\,(x)}{wbuff\,(x)}$, for all $x \in X$.

4.2. Complexity

The time complexity order of the algorithm as a function of its parameters is given by

$$\mathcal{O}\left(MN\mathcal{O}_{\mathcal{T}_{2D}}\left(N_1, N_1\right)\right) + \mathcal{O}\left(MN\frac{\left(N_1^2 + N_2\right)N_S^2}{N_{step}^2}\right) + \mathcal{O}\left(MN\frac{\mathcal{O}_{\mathcal{T}_{3D}}\left(N_1, N_1, N_2\right)}{N_{step}^2}\right),$$

where the first two addends are due to block-matching and the third is due to $\mathcal{T}_{3D}$ used for denoising and where $\mathcal{O}_{\mathcal{T}_{2D}}\left(N_1, N_1\right)$ and $\mathcal{O}_{\mathcal{T}_{3D}}\left(N_1, N_1, N_2\right)$ denote the complexity orders of the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$, respectively. Both $\mathcal{O}_{\mathcal{T}_{2D}}$ and $\mathcal{O}_{\mathcal{T}_{3D}}$ depend on properties of the adopted transforms such as separability and availability of fast algorithms. For example, the DFT has an efficient implementation by means of fast Fourier transform (FFT). The 2D FFT, in particular, has complexity $\mathcal{O}\left(N_1 N_2 \log\left(N_1 N_2\right)\right)$ as opposed to $\mathcal{O}\left(N_1^2 N_2^2\right)$ of a custom non-separable transform. Moreover, an effective trade-off between complexity and denoising performance can be achieved by varying N_{step}.

Table 1. Results in output PSNR (dB) of the denoising algorithm with block-matching and filtering in 3D DFT domain.

	Image						
σ/ PSNR	*Lena* 512×512	*Barbara* 512×512	*House* 256×256	*Peppers* 256×256	*Boats* 512×512	*Couple* 512×512	*Hill* 512×512
5/ 34.15	38.63	38.18	39.54	37.84	37.20	37.40	37.11
10/ 28.13	35.83	34.87	36.37	34.38	33.79	33.88	33.57
15/ 24.61	34.21	33.08	34.75	32.31	31.96	31.93	31.79
20/ 22.11	33.03	31.77	33.54	30.87	30.65	30.58	30.60
25/ 20.17	32.08	30.75	32.67	29.80	29.68	29.57	29.74
30/ 18.59	31.29	29.90	31.95	28.97	28.90	28.75	29.04
35/ 17.25	30.61	29.13	31.21	28.14	28.20	28.03	28.46
50/ 14.16	29.08	27.51	29.65	26.46	26.71	26.46	27.21
100/ 8.13	26.04	24.14	25.92	23.11	24.00	23.60	24.77

5. RESULTS AND DISCUSSION

We present experiments conducted with the algorithm introduced in Section 4, where the transforms $\mathcal{T}_{2D}$ and $\mathcal{T}_{3D}$ are the 2D DFT and the 3D DFT, respectively. All results are produced with the same fixed parameters—but different for the hard-thresholding and Wiener filtering parts. For the hard-thresholding, N_1 is automatically selected in the range $7 \leq N_1 \leq 13$ based on σ, $\tau_{match} = 0.233$, $N_2 = 28$, $N_{step} = 4$, $N_S = 73$, $\beta = 4$, $\lambda_{th2D} = 0.82$, and $\lambda_{th3D} = 0.75$. For the Wiener filtering, N_1 is automatically selected in the range $7 \leq N_1 \leq 11$ based on σ, $\tau_{match} = \frac{\sigma}{4000} + 0.0105$, $N_2 = 72$, $N_{step} = 3$, $N_S = 35$, and $\beta = 3$. In Table 1, we summarize the results of the proposed technique in terms of output peak signal-to-noise ratio (PSNR) in decibels (dB), which is defined as

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{255^2}{|X|^{-1} \sum_{x \in X} \left(y\left(x\right) - \widehat{y}\left(x\right) \right)^2} \right).$$

At `http://www.cs.tut.fi/~foi/3D-DFT`, we provide a collection of the original and denoised test images that were used in our experiments, together with the algorithm implementation (as C++ and MATLAB functions) which produced all reported results. With the mentioned parameters, the execution time of the whole algorithm is less than 9 seconds for an input image of size 256×256 on a 3 GHz Pentium machine.

In Figure 3, we compare the output PSNR of our method with the reported ones of three[6, 7, 10] state-of-art techniques known to the authors as best. However, for standard deviations 30 and 35 we could neither find nor reproduce the results of both the *FSP+TUP*[7] and the exemplar-based[10] techniques, thus they are omitted.

In Figure 4, we show noisy ($\sigma = 35$) *House* image and the corresponding denoised one. For this test image, similarity among neighboring blocks is easy to perceive in the uniform regions and in the regular-shaped structures. Hence, those details are well-preserved in our estimate. It is worth referring to Figure 1, where block-matching is illustrated for a fragment of *House*.

Pairs of noisy ($\sigma = 35$) and denoised *Lena* and *Hill* images are shown in Figures 5 and 6, respectively. The enlarged fragments in each figure help to demonstrate the good quality of the denoised images in terms of faithful detail preservation (stripes on the hat in *Lena* and the pattern on the roof in *Hill*).

We show fragments of noisy ($\sigma = 50$) and denoised *Lena, Barbara, Couple*, and *Boats* images in Figure 7. For this relatively high level of noise, there are very few disturbing artifacts and the proposed technique attains good preservation of: sharp details (the table legs in *Barbara* and the poles in *Boats*), smooth regions (the cheeks of Lena and the suit of the man in *Couple*), and oscillatory patterns (the table cover in *Barbara*). A fragment of *Couple* corrupted by noise of various standard deviations is presented in Figure 8.

In order to demonstrate the capability of the proposed method to preserve textures, we show fragments of heavily noisy ($\sigma = 100$) and denoised *Barbara* in Figure 9. Although the true signal is almost completely buried under noise, the stripes on the clothes are faithfully restored in the estimate.

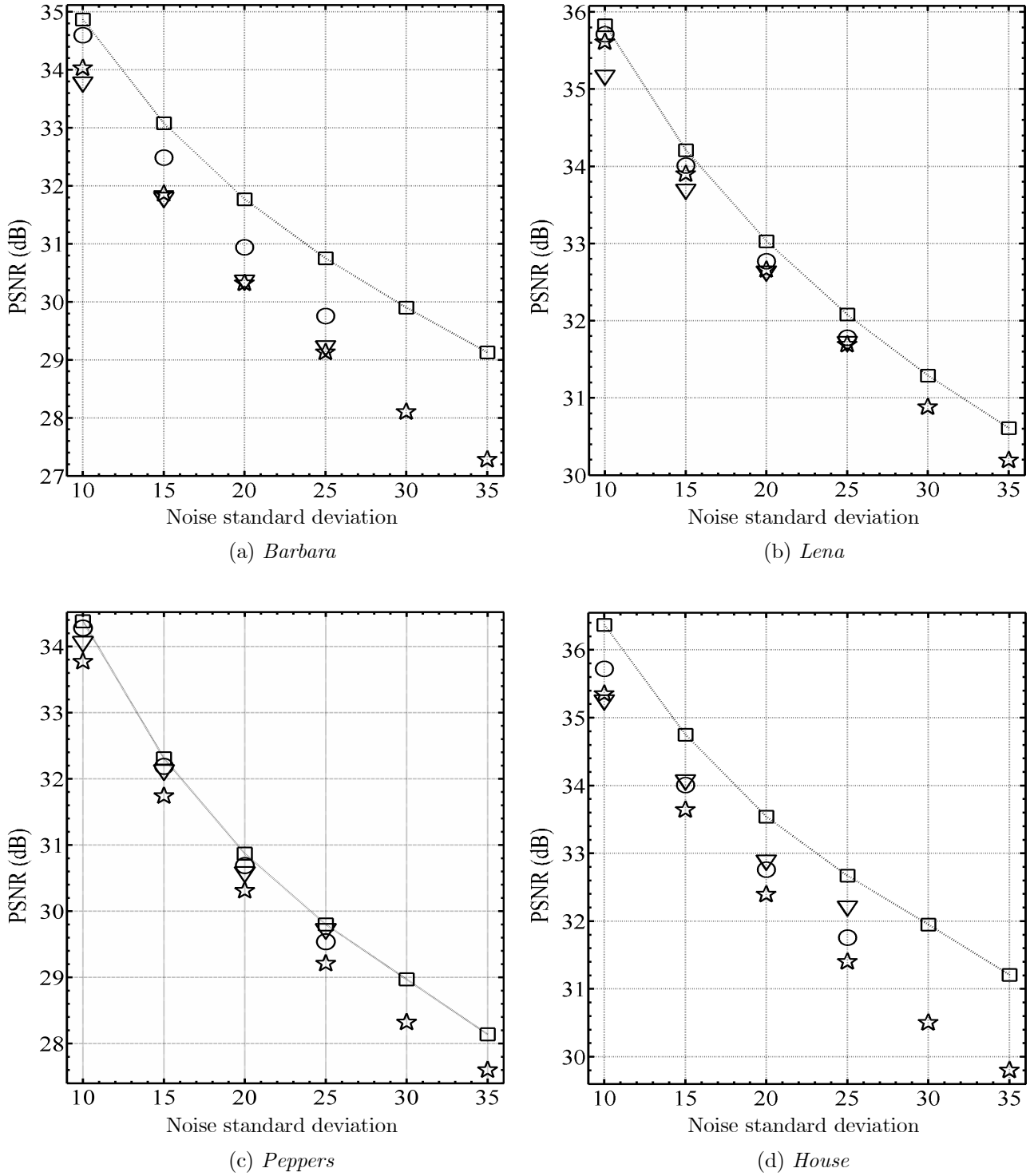


Figure 3. Output PSNR as a function of the standard deviation for *Barbara* (a), *Lena* (b), *Peppers* (c), and *House* (d). The notation is: proposed method (squares), *FSP+TUP*[7] (circles), *BLS-GSM*[6] (stars), and exemplar-based[10] (triangles).

Figure 4. On the left are a noisy ($\sigma = 35$) *House* and two enlarged fragments from it; on the right are the denoised image (PSNR 31.21 dB) and the corresponding fragments.

We conclude by remarking that the proposed method outperforms—in terms of objective criteria—all techniques known to us. Moreover, our estimates retain good visual quality even for relatively high levels of noise.

Our current research extends the presented approach by the adoption of variable-sized blocks and shape-adaptive transforms,[13] thus further improving the adaptivity to the structures of the underlying image. Also, application of the technique to more general restoration problems is being considered.

REFERENCES

1. D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Stat. Assoc.*, vol. **90**, pp. 1200–1224, 1995.
2. D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Theory*, vol. **41**, pp. 613–627, 1995.
3. S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Processing*, vol. **9**, pp. 1532–1546, 2000.
4. A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Processing*, vol. **11**, pp. 545–557, 2002.
5. L. Sendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. **9**, pp. 438–441, 2002.
6. J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Processing*, vol. **12**, pp. 1338–1351, 2003.
7. J. A. Guerrero-Colon and J. Portilla, "Two-level adaptive denoising using Gaussian scale mixtures in overcomplete oriented pyramids," in *Proc. of IEEE Int'l Conf on Image Processing*, Genoa, Italy, September 2005.

Figure 5. On the left are noisy ($\sigma = 35$) *Lena* and two enlarged fragments from it; on the right are the denoised image (PSNR 30.61 dB) and the corresponding fragments.

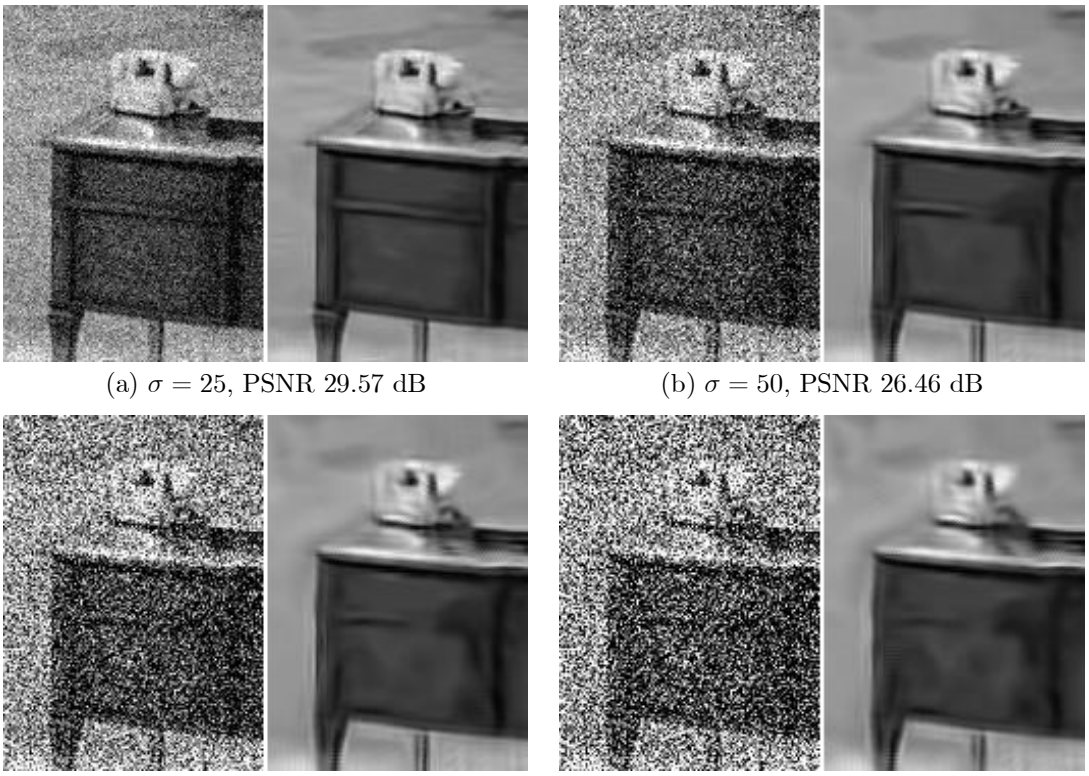
Figure 6. On the left are noisy ($\sigma = 35$) *Hill* and two fragments from it; on the right are the denoised image (PSNR 28.46 dB) and the corresponding fragments from it.

(a) *Lena* ($\sigma = 50$, PSNR 29.08 dB) (b) *Barbara* ($\sigma = 50$, PSNR 27.51 dB)

(c) *Couple* ($\sigma = 50$, PSNR 26.46 dB) (d) *Boats* ($\sigma = 50$, PSNR 26.71 dB)

Figure 7. Fragments of noisy ($\sigma = 50$) and denoised test images.

8. L. Yaroslavsky, K. Egiazarian, and J. Astola, "Transform domain image restoration methods: review, comparison and interpretation," in *Nonlinear Image Processing and Pattern Analysis XII*, *Proc. SPIE* **4304**, pp. 155–169, 2001.
9. R. Öktem, L. Yaroslavsky and K. Egiazarian, "Signal and image denoising in transform domain and wavelet shrinkage: a comparative study," in *Proc. of EUSIPCO'98*, Rhodes, Greece, September 1998.
10. C. Kervrann and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation," *Research Report INRIA*, RR-5624, July 2005.
11. A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. **4**, pp. 490–530, 2005.
12. D. Rusanovskyy and K. Egiazarian, "Video denoising algorithm in sliding 3D DCT domain," in *Proc. of ACIVS'05*, Antwerp, Belgium, September 2005.
13. A. Foi, K. Dabov, V. Katkovnik, and K. Egiazarian, "Shape-Adaptive DCT for Denoising and Image Reconstruction," in *Electronic Imaging'06, Proc. SPIE* **6064**, no. 6064A-18, San Jose, California USA, 2006.



(a) $\sigma = 25$, PSNR 29.57 dB (b) $\sigma = 50$, PSNR 26.46 dB

(c) $\sigma = 75$, PSNR 24.74 dB (d) $\sigma = 100$, PSNR 23.60 dB

Figure 8. Pairs of fragments of noisy and denoised *Couple* for standard deviations: 25 (a), 50 (b), 75 (c), and 100 (d).

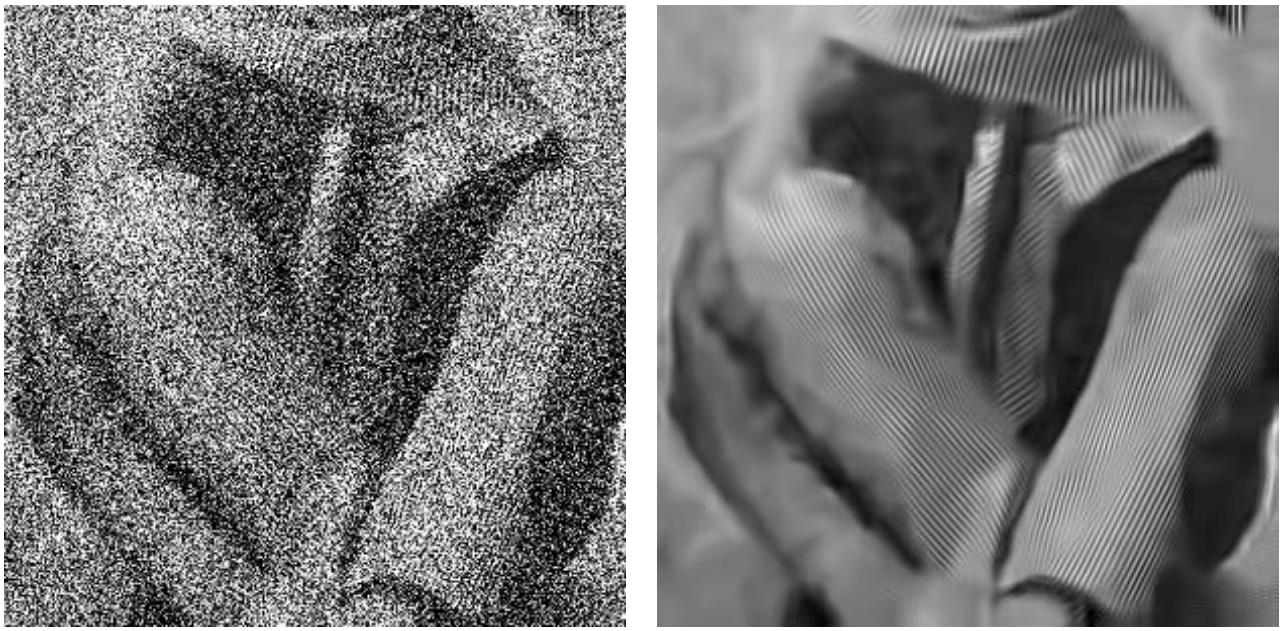


Figure 9. Fragments of noisy ($\sigma = 100$) and denoised (PSNR 24.14 dB) *Barbara*.