NONLOCAL IMAGE DEBLURRING: VARIATIONAL FORMULATION WITH NONLOCAL COLLABORATIVE L_0-NORM PRIOR

Vladimir Katkovnik and Karen Egiazarian

Department of Signal Processing, Tampere University of Technology,
Tampere, Finland, e-mail: katkov@cs.tut.fi, karen@cs.tut.fi.

ABSTRACT

Spatially adaptive nonlocal patch-wise estimation is one of the most promising recent directions in image processing. Within this framework a set of the state-of-the-art Block Matching 3-D (BM3D) algorithms has been developed for different imaging problems [1]-[5]. Recently, a special prior has been proposed allowing to reformulate the multi-stage hard-thresholding BM3D denoising as global minimization of an energy criterion [6]. The outstanding performance of BM3D works as a strong argument in favor of this prior giving an efficient multilayer redundant image model. The variational formulation is used in [6] in order to design a novel recursive denoising algorithm. In this paper the nonlocal collaborative l_0-norm prior is a tool to design deblurring algorithms, where the global penalty function works as an adaptive regularizator. The main contribution concerns the development and testing of algebraic and frequency domain recursive algorithms minimizing the global criterion. Simulation demonstrate a very good performance of the novel algorithms.

1. INTRODUCTION

Suppose we have independent random observation pairs $\{z_i, x_i\}$ given in the form

$$z_i = (g * y)(x_i) + \varepsilon_i, \tag{1}$$

where $z_i = z(x_i)$ and $y_i = y(x_i)$ are noisy observations and signal of interest, respectively; $x_i \in \mathbb{R}^2$ denotes a vector of "features" or explanatory variables, $*$ stands for the $2D$ discrete convolution of the signal y with a blur point-spread function (PSF) g , and $\varepsilon_i = \varepsilon(x_i)$ is an additive noise, $\varepsilon_i \sim N(0, \sigma^2)$. The deblurring problem is to reconstruct $y(x_i)$ from noisy observations $\{z_i\}$.

Most recent imaging algorithms work on image blocks (patches). Let us introduce this concept following [4]. Let the signals from (1) be given in the matrix $(n \times m)$ form as Y and Z and defined on the corresponding $n \times m$ regular 2-D grid X.

Consider a windowing $\mathcal{C} = \{X_r, r = 1, \ldots, N_s\}$ of X with N_s blocks (uniform windows) $X_r \subset X$ of size $n_r \times n_r$ such that $\cup_{r=1}^{N_s} X_r = X$, i..e., this windowing is a *covering of* X. Thus, each $x_i \in X$ belongs to at least one subset X_r. The noise-free data Y and the noisy data Z windowed on X_r are arranged in $n_r \times n_r$ blocks denoted as Y_r and Z_r, respectively. Typically, the blocks are overlapping and therefore some of the elements may belong to more than one block.

We use transforms (orthonormal series) of pixels in the blocks in conjunction with the concept of the redundancy of natural signals. Mainly these are orthogonal polynomials, discrete Fourier, cosine and wavelet transforms. The transform, denoted as $\mathcal{T}_r^{2D}$, is applied on each window X_r independently as

$$\theta_r = \mathcal{T}_r^{2D}(Y_r), \qquad \left[= D_r Y_r D_r^T \right] \quad r = 1, \ldots, N_s, \tag{2}$$

where θ_r is the spectrum of Y_r. The equality enclosed in square brackets holds when the transform $\mathcal{T}_r^{2D}$ is realized as a separable composition of 1-D transforms, each computed by matrix multiplication against an $n_r \times n_r$ orthogonal matrix D_r. The inverse $\mathcal{T}_r^{2D\,-1}$ of $\mathcal{T}_r^{2D}$ defines the signal from the spectrum as

$$Y_r = \mathcal{T}_r^{2D\,-1}(\theta_r), \qquad \left[= D_r^T \theta_r D_r \right] \quad r = 1, \ldots, N_s.$$

The noisy spectrum of the noisy signal is defined as

$$\tilde{\theta}_r = \mathcal{T}_r^{2D}(Z_r), \qquad \left[= D_r Z_r D_r^T \right] \quad r = 1, \ldots, N_s. \tag{3}$$

The signal y is *sparse* if it can be well approximated by a small number of non-zero elements of the spectrum θ_r. The number of non-zero elements of θ_r, denoted using the standard notation of the l_0-norm as $||\theta_r||_{l_0}$, is interpreted as the complexity of the model in the block.

If the blocks are overlapping the total number of the spectrum elements θ_r, $r = 1, \ldots, N_s$, is larger (much larger) than the image size and we arrive to the *overcomplete* or *redundant* data approximation. This redundancy is an important element of the efficiency of this modeling overall.

The blockwise estimates are simpler for calculation than the estimates produced for the whole image because the blocks are much smaller than the whole image. This is a computational motivation for the blocking. Another even more important point is that the blocking imposes a localization of the image on small pieces where simpler models may fit the observations.

978-1-4244-5167-8/09/$25.00 ©2009 IEEE

2. NONLOCAL COLLABORATIVE L_0-NORM PRIOR

2.1. Group-wise penalty

We consider the signal blocks Y_j corresponding to a given windowing X_j. The transforms are defined and calculated for these blocks. Furthermore, it is assumed that there is a similarity between some of the blocks and the similar blocks are clustered in *"groups"*. As a measure of this similarity between the reference block r and the block j we use the Eucledian norm for matrix calculated as $||Y_r - Y_j||_2^2 \triangleq \sum_{k,l} |Y_r(k,l) - Y_j(k,l)|^2$ [1]:

$$w_h(r,j) = 1(||Y_r - Y_j||_2^2 \leq h). \tag{4}$$

This binary weight $w_h(r,j)$ takes value 1 if the Eucledian distance is smaller or equal to h, then the block j belongs to the group r. Otherwise if the Eucledian distance is larger than h then, $w_h(r,j) = 0$ and the block j is not included in the group r. Let K_r^h be a set of the blocks included in the rth group.

We introduce the penalty first for *groups* and further *globally* for the whole image [6].

The penalty for the rth group is defined as

$$pen_r(\{\vartheta_{r,j}\}_j) = \tag{5}$$
$$= \left(\sum_j w_h(r,j) ||\theta_j - \vartheta_{r,j}||_2^2 \right) + \lambda_r ||\{\vartheta_{r,j}\}_j||_{l_0}.$$

Here $\{\vartheta_{r,j}\}_j$ is a set of the models for all jth blocks included in the rth group, and $\{\theta_j\}_j$ is a set of the spectrums of these true signal blocks in this group.

The group-wise penalty $pen(\{\vartheta_{r,j}\}_{r,j})$ is a criterion characterizing the quality of the rth group by the sum of two very different terms. The first summand is the accuracy of the signal approximation by the spectrums $\vartheta_{r,j}$, and the second one $||\{\vartheta_{r,j}\}_j||_{l_0}$ is the complexity of this spectrum model defined as a total number of nonzero (active) elements in the set $\{\vartheta_{r,j}\}_j$. Thus, this group-wise penalty takes into consideration both the accuracy and the complexity of the group-wise modeling.

It has been demonstrated in [1] that a much higher sparsity of the signal representation and as a result a much lower complexity (better quality) of the model can be achieved using a joint 3D group-wise transform instead of 2D block-wise transforms (with spectrums $\vartheta_{r,j}$) as it is in (5). This joint 3D transform dramatically improves the efficiency of image spectrum approximation and introduced in [1] as a special tool for noise removal from images.

Following [6] we use this 3D collaborative transform for modification of the introduced group-wise penalty.

Let $\Theta_r^Y = \{\theta_{r,j}\}_{j \in K_r^h}$ be a collection of the $2D$ block-wise spectrums treated as 3-D array, where j is the index used for the third dimension. We will denote the elements of the $3D$ array Θ_r^Y as $\Theta_{r,j}^Y(k,l)$, where the indices (k,l) concern $2D$ array of the jth block in the rth group. Apply a $1D$ orthonormal transform $\mathcal{T}^{1D}$ with respect to j. In this way we arrive to a group-wise $3D$ spectrum of the signal Y in the rth group as

$$\Omega_r^Y = \mathcal{T}^{1D}(\Theta_r^Y). \tag{6}$$

Following [1], [4] we replace the set K_r^h of the $2D$ spectrum-approximations $\{\vartheta_{r,j}\}_{j \in K_r^h}$ with the corresponding joint $3D$ spectrum $\Omega = \mathcal{T}^{1D}(\{\vartheta_j\}_{j \in K_r^h})$, presumably obtained by applying $\mathcal{T}^{1D}$ on a collection of $2D$ spectra $\{\vartheta_j\}_{j \in K_r^h}$ considered as a function of j. Then, the l_0-norm $||\{\vartheta_{r,j}\}_{j \in K_r^h}||_{l_0}$ in (8) is replaced with the equivalent norm in this $3D$ spectrum space defined as $||\Omega||_{l_0} = \sum_{k,l,j \in K_r^h} 1(\Omega_j(k,l) \neq 0)$.

This $3D$ spectrum representation is used as a joint collaborative model of the signal clustered in the rth group. For this group the group-wise $3D$ spectrum penalty (5) takes the form

$$pen_r(\Omega) = ||\Omega_r^Y - \Omega||_2^2 + \lambda_r ||\Omega||_{l_0}. \tag{7}$$

Recall again, that here Ω is the $3D$ array of the spectrum approximations (estimates) we are looking for, and Ω_r^Y (with index Y) is the spectrum of the blocks of the true signal values Y_j collected into the rth group $\{Y_j\}_{j \in K_r^h}$ accordingly to the rule (4).

2.2. Global penalty

Let us go further and introduce the *global* penalty as the weighted mean of the *group-wise* penalties (5):

$$PEN(\{\vartheta_{r,j}\}_{r,j}) = \sum_r g_r pen_r(\{\vartheta_{r,j}\}_j) = \tag{8}$$
$$\sum_r g_r \left(\sum_j w_h(r,j) ||\theta_j - \vartheta_{r,j}||_2^2 + \lambda_r ||\{\vartheta_{r,j}\}_j||_{l_0} \right),$$

with the group-weights g_r calculated as

$$g_r = \frac{1/||\{\vartheta_{r,j}\}_j||_{l_0}}{\sum_r 1/||\{\vartheta_{r,j}\}_j||_{l_0}}. \tag{9}$$

These weights are inversely proportional to the complexity of the group-wise models. This rule perfectly corresponds to the idea of the sparse image modeling with a low complexity model as a main goal. According to this idea the low complexity groups are preferable and taken in (8) with larger weights.

Using the $3D$ collaborative spectrum representation of the group-wise penalty in the form (7) the global penalty (8) takes the form

$$PEN(\{\Omega_r\}_r) = \sum_r g_r \cdot pen_r(\Omega) = \tag{10}$$
$$\sum_r g_r \left(||\Omega_r^Y - \Omega_r||_2^2 + \lambda_r ||\Omega_r||_{l_0} \right),$$
$$g_r = \frac{1/||\Omega_r||_{l_0}}{\sum_r 1/||\Omega_r||_{l_0}},$$

where the spectrum Ω_r is an approximation for the spectrums Ω_r^Y in the rth group, and $||\Omega_r||_{l_0}$ is the l_0-norm penalty for this approximation.

In the signal domain the global penalty $PEN(\{\Omega_r\}_r)$ can be represented using the block-wise true signals Y_j and the signal approximations $Y_{r,j}$ in the following form

$$PEN(\{Y_j\}_j) = \sum_r g_r \left(\sum_j w_h(r,j) ||Y_j - Y_{r,j}||_2^2 + \lambda_r ||\Omega_r||_{l_0} \right), \quad (11)$$

where

$$Y_{r,j} = \mathcal{T}^{2D\,-1}(\theta_{r,j}), \Theta_r = \{\theta_{r,j}\}_{j \in K_r^h} = \mathcal{T}^{1D\,-1}(\Omega_r). \quad (12)$$

The global penalty has been introduced in the forms (10) and (11) in [6] and used for variational formulation of the image denoising problem.

2.3. Meaning of the global penalty

1. The global penalty (10) is unusual in a number of aspects and very different from the standard formulations of the penalty functions (e.g., [7]). One of the most important is that this penalty function is multilayer. The blocks in the reference group rth are selected as the ones close (similar) to the reference block rth and taken from different part of the image. These blocks form a 3D multilayer group used for group-wise collaborative $3D$ spectrum calculation. Each block can be selected for various reference blocks and in this way it can serve as layers in many blocks. The sets K_r^h of the multilayer constructions can be tracked in the formula (10) explicitly using the window function (4). These multilayer constructions are signal dependent and can be quite complex.

2. Another form of multilayer penalty constructions appears when we go from the spectrum to signal domain (11). The windows are ovelapping and for each image pixel there are multiple window-wise approximations. These constructions depend on image location of the windows collected in the groups, which are the same in the spectrum and the signal domains. This grouping is explicitly revealed by the projection matrices P_j introduced further in (16).

3. The weights g_r defined by the complexity of the group-wise models gives the aggregation weights fusing the multiple group-wise models in the global one. These weights define the weights of the group-wise estimates when they are fused into the final one as it will be clear from the formula (17).

4. The proposed penalties, both group-wise and global, are inspired by the similar constructions developed in the BM3D algorithm for the group-wise multi-model collaborative filtering [1], [4]. The global penalty and variation formulation of estimation result in novel recursive algorithms sharing with BM3D the distinctive features of the latter algorithm: grouping (block-matching), 3D collaborative filtering and the complexity depending weights used for fusing of the group-wise estimates into the final ones.

3. DEBLURRING BY GLOBAL ENERGY MINIMIZATION

Let us use for the signals given by the matrices Y, Z, Y_j, $\hat{Y}_{r,j}$ the lexicographical vector representations with the corresponding bold letter notations $\mathbf{Y}$, $\mathbf{Z}$, $\mathbf{Y}_j$, $\hat{\mathbf{Y}}_{r,j}$. The vectors $\mathbf{Y}_j$ are projections of the vector $\mathbf{Y}$, which can be defined through the projection matrices P_j, $\mathbf{Y}_j = P_j\mathbf{Y}$. Here P_j are binary matrices with items (0,1).

Using the vector-matrix representation of the observation model (1) and the global penalty (10)-(11) the deblurring problem can be formulated as the variational problem [6]:

$$\hat{\mathbf{Y}} = \arg \min_{\mathbf{Y}, \{\Omega_r\}_r} J, \quad (13)$$
$$J = ||\mathbf{Z} - A\mathbf{Y}||_2^2/\sigma^2 + \mu \cdot PEN(\{\Omega_r\}_r),$$

where $||\cdot||_2^2$ stands for the Eucledian norm of vectors.

For solution of (13) we exploit a recursive alternative minimization of J on $\{\Omega_r\}_r$ and $\mathbf{Y}$. In minimization on $\{\Omega_r\}_r$ the global penalty is used in the form (10) while in minimization on Y the global penalty is of the form (11).

If $\mathbf{Y}$ is given the minimization on $\{\Omega_r\}_r$ concerns the penalty term $PEN(\{\Omega_r\}_r)$ only. With fixed g_r the minimization is reduced to scalar calculations independent for each element of Ω_r:

$$\hat{\Omega}_r(k,l) = \arg \min_{x \in R^1} \left(\Omega_r^Y(k,l) - x)^2 + \lambda_r \cdot 1(x \neq 0) \right).$$

This solution is the hard-thresholding of $\Omega_r^Y(k,l)$ calculated as

$$\hat{\Omega}_r(k,l) = \Omega_r^Y(k,l) \cdot 1\left(|\Omega_r^Y(k,l)| \geq \sqrt{\lambda_r} \right). \quad (14)$$

When $\hat{\Omega}_r(k,l)$ are found the signal estimates are calculated as

$$\hat{\Theta}_r = \{\hat{\theta}_{r,j}\}_{j \in K_r^h} = \mathcal{T}^{1D\,-1}\left(\hat{\Omega}_r\right), \quad (15)$$
$$\hat{Y}_{r,j} = \mathcal{T}^{2D\,-1}\left(\hat{\theta}_{r,j}\right).$$

The consecutive $\mathcal{T}^{1D\,-1}$ and $\mathcal{T}^{2D\,-1}$ inverse transforms return first the estimates $\hat{\Theta}_r = \{\hat{\theta}_{r,j}\}_{j \in K_r^h}$ of $\mathcal{T}^{2D}$-spectra of the blocks in the group, and hence the estimates $\hat{Y}_{r,j}$ of these blocks. Because these estimates can be different in different groups, we use the double indexes for the signal estimates $\hat{Y}_{r,j}$, where j stays for the index of the block and r for the group where these estimates are obtained.

Consider minimization of J on $\mathbf{Y}$ provided $\{\Omega_r\}_r$ are given as $\left\{\hat{\Omega}_r\right\}_r$. The spectrums Ω_r^Y depend on $\mathbf{Y}$ and this dependence should be taken into considerations. In order to do it we use the global penalty in the signal domain form (11) where the spectrums in the quadratic norms are replaced by the corresponding signals.

Using the vector-matrix notation and the penalty in the form (11) the criterion J in (13) can be represented as

$$J = ||\mathbf{Z} - A\mathbf{Y}||_2^2/\sigma^2 + \mu \cdot \sum_r g_r \times \quad (16)$$
$$\left(\sum_j w_h(r,j)||P_j\mathbf{Y} - \hat{\mathbf{Y}}_{r,j}||_2^2 + \lambda_r||\Omega_r||_{l_0} \right).$$

Differentiation on $\mathbf{Y}$ gives after some manipulations the estimate of $\mathbf{Y}$:

$$\hat{\mathbf{Y}} = \Phi^{-1} \times \quad (17)$$
$$\left(A^T\mathbf{Z}/\sigma^2 + \mu \cdot \sum_r g_r \sum_j w_h(r,j) P_j^T \hat{\mathbf{Y}}_{r,j} \right),$$
$$\Phi = A^TA/\sigma^2 + \mu \cdot \sum_r g_r \sum_j w_h(r,j) P_j^T P_j.$$

Note, that the matrix $P_j^T P_j$ is diagonal, and then the matrix

$$W = \sum_r g_r \sum_j w_h(r,j) P_j^T P_j \quad (18)$$

is also diagonal.

In the form (16)-(17) the solution of the deblurring problem has been proposed in [6]. A promising feature of this solution is that the regularization of the matrix A^TA is produced by the data-depending weght-matrix W with the weights defined by the complexity of the group-wise models.

In this paper we develop recursive algorithms minimizing the criterion J and test the performance of these algorithms. We call these algorithms the deblurring nonlocal energy minimization (DEB-NEM) algorithms and present them in two different versions: algebraic (matrix) and frequency domains.

3.1. Matrix DEB-NEM algorithm

Recursive calculations based on the formulas (15) and (17) result in the following algorithm:
1: Initialization: $\hat{Y}^{(0)}$ and $g_r^{(0)} = 1$;
2: For every $t = 0, 1, ...$

- Calculate the windowed signals $\hat{Y}_r^{(t)}$, the weights

$$w_h^{(t)}(r,j) = 1(||\hat{Y}_r^{(t)} - \hat{Y}_j^{(t)}||_2^2 \le h) \quad (19)$$

and the windowed spectrums $\tilde{\theta}_{r,j}^{(t)} = \mathcal{T}_r^{2D}\left(\hat{Y}_j^{(t)}\right)$, $j \in K_r^h$, for all groups r;

- Calculate the group-wise *"noisy"* spectrums $\tilde{\Omega}_r^{\hat{Y}^{(t)}}$, the 3D spectrum estimates $\hat{\Omega}_r^{(t)}$ using (14), updated windowed $2D$ spectrum estimates $\hat{\theta}_{r,j}^{(t)}$ and the corresponding updated windowed signal estimates $\hat{Y}_{r,j}^{(t)}$ using the inverse transforms (15);

- Calculate the complexity $||\hat{\Omega}_r^{(t)}||_{l_0}$ of the group models and the weights

$$g_r^{(t)} = \frac{1/||\hat{\Omega}_r^{(t)}||_{l_0}}{\sum_r 1/||\hat{\Omega}_r^{(t)}||_{l_0}}; \quad (20)$$

- Update the signal estimate $\hat{\mathbf{Y}}^{(t+1)}$ using (17)

$$\hat{\mathbf{Y}}^{(t+1)} = \mathbf{\Phi}^{-1} \times \quad (21)$$
$$\left(A^T\mathbf{Z}/\sigma^2 + \mu \cdot \sum_r g_r^{(t)} \sum_j w_h^{(t)}(r,j) P_j^T \hat{\mathbf{Y}}_{r,j}^{(t)} \right),$$
$$\mathbf{\Phi} = A^TA/\sigma^2 + \mu \cdot \sum_r g_r^{(t)} \sum_j w_h^{(t)}(r,j) P_j^T P_j;$$

- Continue until convergence.

Each iteration of the presented algorithm can be treated as composed from the three successive stages. These stages are as follows: (1) grouping defined by the window (19), (2) 3D collaborative approximation (filtering) of the image distribution returning $\tilde{\Omega}_r^{\hat{Y}^{(t)}}$, $\hat{\theta}_{r,j}^{(t)}$, $\hat{\mathbf{Y}}_{r,j}^{(t)}$, and (3) aggregation (21) of the group-wise estimates $\hat{\mathbf{Y}}_{r,j}^{(t)}$ into $\hat{\mathbf{Y}}^{(t+1)}$ estimating the image distribution y.

The first two stages concern calculations of the spectrums $\tilde{\Omega}_r^{\hat{Y}^{(t)}}$, $\hat{\Omega}_r^{(t)}$, $\hat{\theta}_{r,j}^{(t)}$ and the signals $\hat{\mathbf{Y}}_{r,j}^{(t)}$. These two stages are identical to the corresponding stages in the hard-thresholding basic BM3D algorithm. The third aggregation stages are essentially different in the BM3D and in the proposed algorithms. The aggregation in the hard-thresholding BM3D is performed by element-wise division of the vector

$$\tilde{\mathbf{Y}}^{(t)} = \sum_r g_r^{(t)} \sum_j w_h^{(t)}(r,j) P_j^T \hat{\mathbf{Y}}_{r,j}^{(t)} \quad (22)$$

by the elements of the weight-vector

$$diag\left\{ \sum_r g_r^{(t)} \sum_j w_h^{(t)}(r,j) P_j^T P \right\}. \quad (23)$$

It is very different from the aggregation (21) where actually we have a combination of the aggregation of the block-wise estimates $\hat{\mathbf{Y}}_{r,j}^{(t)}$ as well as the regularized inverse of the original blurred observation $\mathbf{Z}$.

The complexity of the presented algebraic algorithm is dominated by the complexity of the recursive procedure (21) where the size of the matrix $\mathbf{A}$ is a main limiting factor. This algorithm can be used only for small size images: in our experiments limited to 64×64.

3.2. Frequency domain DEB-NEM algorithm

First, let us rewrite (17) as a set of the linear equations

$$\left(A^TA/\sigma^2 + \mu \cdot W\right)\hat{\mathbf{Y}} = \quad (24)$$
$$A^T\mathbf{Z}/\sigma^2 + \mu \cdot \sum_r g_r \sum_j w_h(r,j) P_j^T \hat{\mathbf{Y}}_{r,j},$$

where W is given in (18), and solve this equations with respect to $\mathbf{Y}$ using the recursive procedure

$$\hat{\mathbf{Y}}_{(k+1)} = \hat{\mathbf{Y}}_{(k)} - \alpha_k[(A^TA/\sigma^2+\mu\cdot W)\hat{\mathbf{Y}}_{(k)} - A^T\mathbf{Z}/\sigma^2-\mu\cdot\tilde{\mathbf{Y}}], \quad \tilde{\mathbf{Y}} = \sum_r g_r \sum_j w_h(r,j)P_j^T\hat{\mathbf{Y}}_{r,j},\ k=1,...,L. \tag{25}$$

Second, assume that the blur kernel g in (1) is shift-invariant separable with respect to the arguments $x = (x^1, x^2)$, i.e. $g(x) = \tilde{g}(x^1)\tilde{g}(x^2)$. Then, the convolution $u = (g * y)$ can be presented in the matrix form

$$U = G\cdot Y\cdot G^T, \tag{26}$$

where U and Y are $n \times m$ matrices of the original and blurred images and G is a Toeplitz matrix of the values of $\tilde{g}$ on the corresponding grid. The vectorization of (26) gives the convolution in the matrix form used in (13), where A is a structured Toeplitz matrix calculated using the Kronecker matrix product $A = G \otimes G$. Then, the discrete Fourier transform (DFT) can be used for calculation of the products with the matrix $\mathbf{A}$ with the following replacements:

$$A^T\cdot\mathbf{Z} \Longrightarrow \mathrm{col}(\mathcal{F}^{-1}\{\mathcal{F}\{G\}^*\cdot\mathcal{F}\{Z\}\}), \quad A^TA\cdot\hat{\mathbf{Y}}_{(k)} \Longrightarrow \mathrm{col}(\mathcal{F}^{-1}\{|\mathcal{F}\{G\}|^2\cdot\mathcal{F}\{Y_{(k)}\}\}), \tag{27}$$

where $\mathcal{F}\{\cdot\}$ and $\mathcal{F}^{-1}\{\cdot\}$ are DFT and inverse DFT of the corresponding variables, respectively.

In order to make this arguments leading to calculations using DFT more natural and straightforward, let us assume that the convolution in (1) is circular. In this case, the all matrix operations can be replaced by the equivalent calculations in the frequency domain using the replacement shown in (27).

Then, the algorithm (25) can be implemented without variable vectorization for variables organized as image-size matrices

$$Y_{(k+1)} = Y_{(k)} - \alpha_k[\mathcal{F}^{-1}\{|\mathcal{F}\{G\}|^2\cdot\mathcal{F}\{Y_{(k)}\}\}/\sigma^2+\mu\cdot(\tilde{W}\circ Y_{(k)})- \mathcal{F}^{-1}\{\mathcal{F}\{G\}^*\cdot\mathcal{F}\{Z\}\}/\sigma^2-\mu\cdot\tilde{Y}],\ k=1,...,L, \tag{28}$$

where $\tilde{W} = reshape_{n\times m}[diag\{W\}]$, $\tilde{Y} = reshape_{n\times m}[\tilde{\mathbf{Y}}]$, and $\tilde{W}\circ Y_{(k)}$ means the element-wise product of two matrices.

The DEB-NEM algorithm is organized as it is presented in the previous section with the only difference that the estimate $Y^{(t+1)} = reshape_{n\times m}[\hat{\mathbf{Y}}^{(t+1)}]$ is calculated according to the imbedded recursive procedure (28): for given g_r, $w_h(r,j)$, P_j, $\tilde{Y}$ calculate $Y^{(t+1)}$ according to (28), where the initialization is given by $Y_{(1)} = y^{(t)}$ and the final estimate is $Y^{(t+1)} = Y_{(L+1)}$.

This algorithm is applicable for images of usual sizes, in our experiments upto 512×512 pixels.

The convergence of the algorithm (25) depends on the step-size parameter α_k. According to the well known theory (e.g. [8]), the best geometrical convergence rate with the parameter $q = (a_{\max}-a_{\min)/}(a_{\max}+a_{\min})$ is achieved for $\alpha = 2/(a_{\max}+a_{\min})$, where $a_{\max}$ and $a_{\min}$ are the maximum and minimum eigenvalues of the matrix $A^TA/\sigma^2+\mu\cdot W$.

For the circular convolution eigenvalues of the matrix A^TA can be calculated using DFT of the corresponding functions as $a_{\max}(A^TA) = \max_{f_1,f_2}|G(f_1,f_2)|^2$ and $a_{\min}(A^TA) = \min|G(f_1,f_2)|^2$. Then, the inequalities for $a_{\max}$ and $a_{\min}$ are of the form

$$a_{\max} \le \hat{a}_{\max} = a_{\max}(A^TA)/\boldsymbol{\sigma}^2 + \mu\max(diag\{W\}), \quad a_{\min} \ge \hat{a}_{\min} = a_{\min}(A^TA)/\boldsymbol{\sigma}^2 + \mu\min(diag\{W\}),$$

with the step-size parameter α calculated as $\hat{\alpha} = 2/(\hat{a}_{\max}+\hat{a}_{\min})$.

In our implementation of the algebraic and frequency domain algorithms for calculation of the estimate $\tilde{\mathbf{Y}}$ and the weight $\tilde{W}$ we use the hard-thresholding part of BM3D algorithm, where these variables are used for denoising.

4. SIMULATION EXPERIMENTS

In our experiments, we use 9×9 uniform kernel (box-car) blur PSF. The noise is white zero-mean Gaussian with blurred-signal-to-nose-ratio $BSNR = 40$ dB. The image restoration is characterized by the values of $PSNR$ and $ISNR$.

4.1. Matrix DEB-NEM algorithm

For these experiments we exploit 64×64 fragments of the test-images *cameraman, lena, barbara*. For initialization (as initial guess) we use the estimates given by $BM3D$ deblurring (DEBBM3D) algorithm [3]. Table 1 shows $PSNR$ and $ISNR$ with the index 0 for the initial guess and the $PSNR$ and $ISNR$ values obtained after 10 iterations of the DEB-NEM algorithm. These iterations give a very valuable improvement varying from 1dB for *barbara* and upto more than 3.5 dB for *cameraman* and *lena* .

The parameters of the algorithm are fixed as $\lambda/\sigma = 10$ and $\mu\sigma = 1$. The main goal of these experiments is to check a potential of the proposed penalty function in the deblurring problem. In these experiments the matrix A is calculated assuming the circular convolution for blurred images.

Table 1. Initial (DEBBM3D) $PSNR$ and $ISNR$ values given with the index 0 and final (DEB-NEM) after 10 iterations (in dB).

	$PSNR_0$ $[ISNR_0]$	$PSNR_{10}$ $[ISNR_{10}]$
cameraman	23.02 [7.18]	**26.65 [10.81]**
lena	30.81 [6.41]	**34.35 [9.95]**
barbara	25.27 [7.93]	**26.27 [8.93]**

Visual improvement in imaging is illustrated in Fig. 1 and Fig. 2 for *lena* and *cameraman* images after 10 iterations. The original image fragments and blurred noisy

images are shown in Fig. 3 and Fig. 4 for *lena* and *cameraman* images respectively.

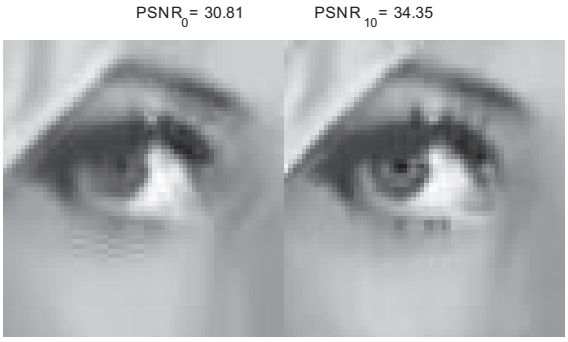

Figure 1. *Lena* fragment: initialization and 10-th iterations of DEB-NEM reconstruction.

Figure 2. *Cameraman* fragment: initialization and 10-th iterations of DEB-NEM reconstruction.

4.2. Frequency domain DEB-NEM algorithm

The frequency domain DEB-NEM algorithm works with the circular convolution of the observed image and can be used for large size images. Of course, using the imbedded recursive algorithm (28) instead of the accurate inverse of the matrix Φ does not allow to exploit fully the ability of the global penalty function. Nevertheless, simulation experiments show that the algorithm is able to achieve essential improvements in image restoration.

Some of the results are shown in Table 2. The indexes of $PSNR$ and $ISNR$ are the numbers of iterations used for image reconstruction. The second column corresponds to the DEBBM3D algorithm used for initialization of the DEB-NEM algorithm. The criterion values are denoted as $PSNR_0$ $[ISNR_0]$. Numbers in square brackets correspond to $ISNR$. The results shown in third column are given for 2 and 20 iterations of the DEB-NEM algorithm.

On default, the parameters of the algorithm are $\mu\sigma = 1$, $\lambda/\sigma = 10$ and the number of the embedded iterations $L = 100$. For images *cameraman*, *lena* and *boats* the advantage of the proposed algorithm with respect to

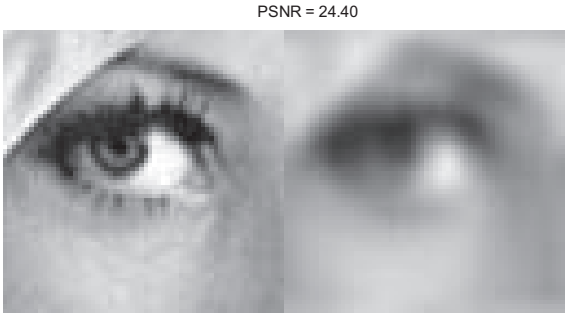

Figure 3. *Lena* fragment: true and blurred noisy images.

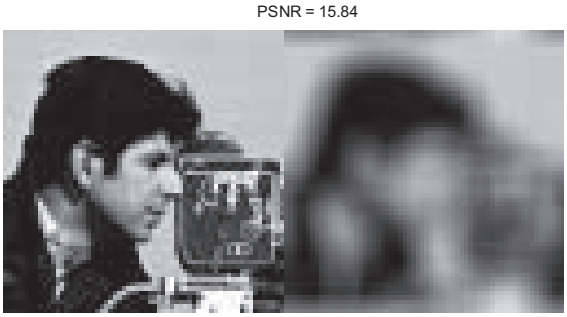

Figure 4. *Cameraman* fragment: true and blurred noisy images.

DEBBM3D is obvious with improvement for *cameraman* about 0.6 dB after 2 iterations and with improvement about 1.0 dB after 20 iterations. The corresponding figures for *lena* are about 0.5 dB after 2 iterations and with about $.75$ dB after 20 iterations. For boats the improvement is about 0.6 dB after 2 iterations and a bit more than 1 dB for 20 iterations.

The visual improvement is illustrated in Fig. 5 and Fig. 6 where the DEBBM3D reconstructions and the reconstruction by the proposed algorithm after 20 iterations are shown. It is seen in the set of images for *cameraman*, that the DEB-NEM algorithm eliminates the artifacts induced by DEBBM3D near the shoulder of *cameraman* and tripod elements. Note that the grass and sky are not so oversmoothed by DEB-NEM as in the DEBBM3D reconstruction.

A difference between the images obtained for *lena* is also in favor of the DEB-NEM algorithm. In particular, the *lena's* shoulder is smooth while the DEBBM3D reconstruction shows quite visible artifacts/bands. The eye's and face of *lena* are reconstructed much close to the true image by the DEB-NEM algorithm than that is in the DEBBM3D reconstruction.

The *barbara* test image appeared to be the most difficult for the proposed algorithm. The improvement is reduced to 0.12 dB after 2 iterations and to 0.19 dB after 20 iterations. It is obtained for the parameters of the algo-

rithm $\mu\sigma = 10$, $\lambda/\sigma = 5$.

A very good performance is demonstrated for the binary *chessboard* test image with improvement about 4 dB and 5 dB after 2 and 20 iterations, respectively. It is interesting to note that in all these experiments $PSNR$ (and $ISNR$) are monotonically growing functions approaching some limit values as a number of iterations becomes larger.

Table 2. Initial (DEBBM3D) $PSNR$ and $ISNR$ values given with the index 0 and DEB-NEM after 2 and 20 iterations (in dB).

	$PSNR_0$ $[ISNR_0]$	$PSNR_{2,20}$ $[ISNR_{2,20}]$
cameraman, 256^2	29.2 [8.4]	**30.0, 30.7 [9.2, 9.9]**
lena, 512^2	33.83 [7.99]	**34.26, 34.65 [8.42, 8.81]**
barbara, 512^2	28.40 [5.91]	**28.52, 28.59 [6.03, 6.10]**
house, 256^2	35.05 [10.95]	**36.61, 37.03 [12.50, 12.93]**
boats, 512^2	31.84 [8.48]	**32.46, 32.89 [9.01, 9.53]**
chessboard, 128^2	42.02 [30.21]	**46.28, 46.80 [34.47, 35.00]**

In this comparative study we refer to the results given by the DEBBM3D only, because as it is shown in [3] these results are mainly the best in the field of image deblurring algorithms.

5. CONCLUSION AND FURTHER WORK

The main results of this paper is a development of the deblurring algorithms based the nonlocal collaborative penalty. The essentially nonlocal algorithms using the adaptive patch matching are obtained by minimization of the energy criterion. While these algorithms are from the class of nonlocal means [9], [10] both of them are close relatives of the BM3D denoising algorithm [1].

The experiments demonstrate a very good performance of the proposed deblurring algorithms. It is a principal point that these algorithms are able to improve the results obtained by the DEBBM3D algorithm [3], which is currently one of the best algorithms in the field.

The following topics are further steps in the development of the proposed variational approach to deblurring:

(1) Development of matrix inverse algorithms more efficient than the frequency domain procedure (28);

(2) Selection of the threshold λ and the weight μ depending on the PSF and observations.

6. ACKNOWLEDGEMENT

This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program 2006 – 2011).

7. REFERENCES

[1] Dabov, K., A. Foi, V. Katkovnik, and Egiazarian, K., "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080 - 2095, 2007.

[2] Dabov, K., A. Foi, V. Katkovnik, and Egiazarian, K., "A nonlocal and shape-adaptive transform-domain collaborative filtering," in *Proceedings 2008 Int. Workshop on Local and Non-Local Approximation in Image Processing, LNLA 2008*, Lausanne, Switzerland, August 2008.

[3] Dabov, A. Foi, and K. Egiazarian, "Image restoration by sparse 3D transform-domain collaborative filtering," *Proc. SPIE Electronic Imaging '08*, no. 6812-07, San Jose, California, USA, January 2008.

[4] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "Nonparametric regression in imaging: from local kernel to multiple-model nonlocal collaborative filtering," in *Proc. 2008 Int. Workshop on Local and Non-Local Approximation in Image Processing, LNLA 2008*, Lausanne, Switzerland, August 2008.

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

[6] V. Katkovnik, "Nonlocal collaborative l_0-norm prior for image denoising," In *Festschrift in Honor of Jaakko Astola on the Occasion of his 60th Birthday,* TICSP, Series 47, pp. 305-319, 2009.

[7] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D,* 60 2, pp. 259-268, 1993.

[8] B. T. Polyak. *Introduction to optimization*. Optimization Software Inc, 1987.

[9] Buades, A., B. Coll, and J.M. Morel, "A review of image denoising algorithms, with a new one," *SIAM Multiscale Modeling and Simulation*, vol. 4, pp. 490–530, 2005.

[10] S. Kindermann, S. Osher, and P.W. Jones, "Deblurring and denoising of images by nonlocal functionals," *SIAM Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1091-1115, 2005.

Figure 5. Camereman test image: true (a), blurred noisy (b), DEB-NEM reconstruction after 20 iterations (c), DEBBM3D reconstruction (d).

Figure 6. A fragment of lena test image: true (a), blurred noisy (b), DEB-NEM reconstruction after 20 iterations (c), DEBBM3D reconstruction (d).