Blind Deconvolution of Images using Optimal Sparse Representations

Michael M. Bronstein, *Student Member, IEEE,* Alexander M. Bronstein*, *Student Member, IEEE,*
Michael Zibulevsky, *Member, IEEE,* and Yehoshua Y. Zeevi

***Abstract*— The relative Newton algorithm, previously proposed for quasi maximum likelihood blind source separation and blind deconvolution of one-dimensional signals is generalized for blind deconvolution of images. Smooth approximation of the absolute value is used in modelling the log probability density function, which is suitable for sparse sources. In addition, we propose a method of sparsification, which allows blind deconvolution of sources with arbitrary distribution, and show how to find optimal sparsifying transformations by training.**

***Index Terms*— blind deconvolution, quasi maximum likelihood, sparse representations, relative Newton optimization.**

I. Introduction

TWO-dimensional *blind deconvolution* (BD) is a special case of a more general problem of *image restoration*. The goal of BD is to reconstruct the original scene from an observation degraded by the action of a linear shift invariant (LSI) system, when no or very little *a priori* information about the scene and the degradation process is available, hence the term ”blind”. BD is critical in many fields, including astronomy [1], [2], remote sensing [3], biological and medical imaging [4], [5], microscopy [6], [7], etc. Typically, the image is degraded by imperfections of an optical system, and can be presented in terms of convolution of the source image with some *blurring kernel* or *point spread function* (PSF); in such applications, the term *deblurring* is synonymous to deconvolution.

A. Problem formulation

In the general setup of 2D BD, the observed sensor image x is created from the *source image* s passing through a convolutive system defined by its impulse response a,

$$x_{\boldsymbol{n}} \quad = \quad \sum_{\boldsymbol{m}} a_{\boldsymbol{m}} s_{\boldsymbol{n}-\boldsymbol{m}} + v_{\boldsymbol{n}},$$

and is possibly contaminated by additive *sensor noise* v. Sometimes, s can be contaminated by *source noise* u, usually multiplicative. For notation convenience, we use bold subscript indices to indicate multi-indices consisting of two variables: $\boldsymbol{n} = (n_1, n_2)$. Summation is performed over both of them. We assume that the action of w is invertible, at least approximately. The aim of BD is to find such *deconvolution* (*restoration*) kernel w that produces an estimate $\tilde{s}$ of s up to integer shift $\boldsymbol{\Delta} = (\Delta_1, \Delta_2)$ and scaling factor c:

$$\tilde{s}_{\boldsymbol{n}} \quad = \quad \sum_{\boldsymbol{m}} w_{\boldsymbol{m}} x_{\boldsymbol{n}-\boldsymbol{m}} \approx c s_{\boldsymbol{n}-\boldsymbol{\Delta}},$$

or equivalently, the *global system response* should be

$$g_{\boldsymbol{n}} \quad = \quad (a * w)_{\boldsymbol{n}} \approx c\delta_{\boldsymbol{n}-\boldsymbol{\Delta}},$$

where $\delta_{\boldsymbol{n}}$ denotes the Krönecker delta (discrete impulse signal).

B. Previous work

Various BD methods have been previously proposed. We will only briefly outline the basic approaches (for a comprehensive comparison see e.g. [8]). Most of the BD approaches can be divided into *parametric* and *non-parametric*. In applications where the form of the PSF can be assumed in advance (e.g. motion blur or defocus), it is possible to use a parametric model of the PSF and instead of finding the PSF itself, one can try to estimate the parameters of its model; the advantages are, obviously, in having a smaller number of variables. However, in real applications it is often difficult to derive a good model for the PSF

BD approaches can be divided into those that estimate the blurring kernel, those estimating the source image and the blurring kernel simulataneously, and those estimating the restoration kernel. The first class includes the so-called *a priory* blur identification methods, which first estimate the blurring kernel and then employ a non-blind deconvolution algorithm to find the source estimate [9]–[11].

The second class includes methods based on statistical or deterministic priors of the source image, the blurring kernel and the noise [12], [13]. Estimation of the source image is performed by maximizing some optimality criterion, which includes these priors. Since the variables in this problem are both the source image and the blurring kernel, the computational complexity is a major problem.

The third class of methods usually employs maximum likelihood (ML) estimators of the restoration kernel; such estimators usually incorporate priors on the image and the kernel and ignore the noise. Since in this case $\tilde{s} = w * x$, there is no need to estimate the source image, therefore these approaches demand the solution of more modest optimization problems and, consequently, are more computationally effi-

Manuscript received October 10, 2003. This research has been supported by the HASSIP Research Network Program HPRN-CT-2002-00285, sponsored by the European Commission and by the Ollendorff Minerva Center. *Asterisk indicates corresponding author.*

M. M. Bronstein is with the Department of Computer Science, Technion – Israel Institute of Technology, Haifa 32000, Israel. A. M. Bronstein, M. Zibulevsky and Y. Y. Zeevi are with the Department of Electrical Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel (e-mail: bronstein@ieee.org; alexbron@ieee.org; mzib@ee.technion.ac.il; zeevi@ee.technion.ac.il).

cient. However, the exact source distribution, required for the ML approach, is often unknown.

A possible remedy is to use an approximate probability density function; such a modified ML approach is usually referred to as *quasi maximum likelihood* (QML). Such estimation techniques were successfully used in blind source separation (BSS) [14]–[16]. The relative Newton quasi ML framework for BD of 1D signals was introduced in [17]. Here, we extend it to the 2D case. In addition, we present a novel approach of using optimal sparse representation, which can be used for BD of source images with arbitrary distributions.

II. QUASI ML BLIND DECONVOLUTION

The convolution operation $w * x$ can be thought of as application of an infinite Toeplitz block-Toeplitz operator $\mathcal{W}$, defined by the impulse response $w_{\boldsymbol{n}}$. Denoting the source estimate by $y = w * x$ and assuming that s is i.i.d., the likelihood of the observation x given the restoration kernel w in zero-noise conditions is

$$p(x|h) \;=\; p(s = w * x) = |\det \mathcal{W}| \cdot \prod_{\boldsymbol{n}} p_s(y_{\boldsymbol{n}}), \qquad (1)$$

where $p_s(s)$ stands for the source probability density function (PDF). For convenience, instead of maximizing the likelihood $p(x|w)$, maximum likelihood estimators usually minimize $-\log p(x|w)$. Neglecting edge effects, the following normalized minus-log likelihood function is obtained [18]:

$$\ell(x;w) \;=\; -\frac{M_X N_X}{4\pi^2}\int_{\Pi} \log|\hat{w}(\boldsymbol{\xi})| \; d\boldsymbol{\xi} + \sum_{\boldsymbol{n}} \varphi(y_{\boldsymbol{n}}), (2)$$

where $M_X N_X$ is the observation sample size, $\Pi = [-\pi,\pi] \times [-\pi,\pi]$, $\varphi(s) = -\log p_s(s)$, and

$$\hat{w}(\boldsymbol{\xi}) \;=\; \sum_{\boldsymbol{n}} w_{\boldsymbol{n}} \; e^{-\imath \boldsymbol{n}^T \boldsymbol{\xi}}$$

denotes the two-dimensional Fourier transform of $w_{\boldsymbol{n}}$. We will henceforth assume that $w_{\boldsymbol{n}}$ has finite impulse response (FIR), supported on $\boldsymbol{n} \in [-M,...,M] \times [-N,...,N]$. We will use $K_M = 2M{+}1$ and $K_N = 2N{+}1$ to denote the dimensions of the restoration kernel, and denote the first and the second terms of $\ell(x;w)$ as ℓ_1 and ℓ_2, respectively.

A. The choice of $\varphi(s)$

Natural images encountered in most applications are usually characterized by non-log-concave, multi-modal distributions, which are difficult to model and are not well-suited for optimization[1]. However, consistent estimator of s can be obtained by minimizing $\ell(x;w)$ even when $\varphi(s)$ is not exactly equal to $-\log p_s(s)$. Such quasi ML estimation has been shown to be practical in instantaneous BSS [14], [16], [19] and BD of 1D signals [17], [20]. For example, when the source is super-Gaussian[2] (e.g. it is sparse or sparsely representable), a smooth approximation of the absolute value function is a good choice for $\varphi(s)$ [17], [19], [21]. Although natural images are usually far from being sparse, they can be sparsely represented by a proper transformation [16], [22]. (In Section IV, we will show how to transform general classes of natural images into sparse ones.) We therefore focus our attention on modelling distributions of sparse signals using a family of convex smooth functions

$$\varphi_\lambda(s) \;=\; |s| - \lambda \log\left(1 + \frac{|s|}{\lambda}\right), \qquad (3)$$

where λ is a positive smoothing parameter [15]; $\varphi_\lambda(s) \to |s|$ as $\lambda \to 0^+$. For convenience, we henceforth omit λ from our notation whenever possible, and refer to $\varphi_\lambda(s)$ without using the subscript.

Yet another important advantage of working with sparsely-represented sources is the fact that the asymptotic restoration error variance is significantly smaller compared to the original ones, i.e., deconvolution is more accurate. This issue is addressed in [18].

B. Approximation of the log-likelihood function using FFT

In practice, the first term of $\ell(x;w)$, containing the integral, is difficult to evaluate. It can, however, be approximated with any desired accuracy by [17]

$$\frac{1}{4\pi^2}\int_{\Pi} \log|\hat{w}(\boldsymbol{\xi})| \; d\boldsymbol{\xi} \;\approx\; \frac{1}{M_F N_F}\sum_{\boldsymbol{k}} \log|\hat{w}_{\boldsymbol{k}}|,$$

where

$$\hat{w}_{\boldsymbol{k}} \;=\; \mathcal{F}\left\{w_{\boldsymbol{n}}\right\}_{\boldsymbol{k}} = \hat{w}\left(\frac{2\pi k_1}{M_F}, \frac{2\pi k_2}{N_F}\right) \qquad (4)$$

are the 2D DFT coefficients of $w_{\boldsymbol{n}}$, zero-padded to the support $\boldsymbol{k} \in [0,...,M_F-1] \times [0,...,N_F-1]$. $\mathcal{F}$ denotes the 2D FFT operator. The approximation error vanishes as M_F, N_F grow to infinity. Choosing M_F and N_F as integer powers of 2, allows to use 2D FFT.

C. Gradient and Hessian of $\ell(x;h)$

Optimization algorithms discussed in Section III-C require the knowledge of the gradient and the Hessian of $\ell(x;w)$. Since the optimization variable w is a $K_M \times K_N$ matrix, the gradient $\nabla\ell$ is also a $K_M \times K_N$ matrix, whereas the Hessian $\nabla^2\ell$ is a $K_M \times K_N \times K_M \times K_N$ fourth-order tensor. For convenience, we parse the variables column-wise into a $K_M K_N \times 1$ vector

$$\mathrm{vec}(w) \;=\; \left[\; w_{-M,-N}, ..., w_{M,-N}, ..., w_{M,N} \;\right]^T,$$

and define the gradient and the Hessian of $\ell(x;w)$ as a $K_M K_N \times 1$ vector and a $K_M K_N \times K_M K_N$ matrix, respectively.

The elements of the gradient and the Hessian of $\ell(x;w)$ are given by

$$\frac{\partial \ell}{\partial w_{\boldsymbol{i}}} \;=\; -\mathcal{F}^{-1}\left\{\hat{w}_{\boldsymbol{k}}^{-1}\right\}_{-\boldsymbol{i}} + \frac{1}{M_F N_F}\sum_{\boldsymbol{n}} \varphi'(y_{\boldsymbol{n}}) x_{\boldsymbol{n}-\boldsymbol{i}}$$

[1] Non-log-concave distributions lead to non-convex prior term in the minus log-likelihood function.

[2] Super-Gaussian sources possess positive kurtosis excess: $\kappa = \mathbb{E}s^4/\mathbb{E}^2 s^2 - 3 > 0$. Sub-Gaussian sources have $\kappa < 0$.

and

$$\begin{aligned}\frac{\partial^2 \ell}{\partial w_{\boldsymbol{i}} \partial w_{\boldsymbol{j}}} &= \mathcal{F}^{-1}\left\{\hat{w}_{\boldsymbol{k}}^{-2}\right\}_{-(\boldsymbol{i}+\boldsymbol{j})} \\ &\quad + \frac{1}{M_F N_F} \sum_{\boldsymbol{n}} \varphi''(y_{\boldsymbol{n}}) x_{\boldsymbol{n}-\boldsymbol{i}} x_{\boldsymbol{n}-\boldsymbol{j}}, \end{aligned} \tag{5}$$

respectively. For derivation see [18]. Computational complexity of the target function ℓ and its gradient is $\mathcal{O}(M_F N_F \log_2 M_F N_F + M_X N_X \log_2 M_X N_X)$; whereas evaluation of the Hessian requires $\mathcal{O}(M_F N_F \log_2 M_F N_F + MNM_X N_X \log_2 M_X N_X)$ operations, assuming efficient implementation involving the FFT.

III. Relative Newton algorithm

A fast relative optimization algorithm for BSS, based on the Newton method, was introduced in [15] as a modification of the approach presented in [14]. This method was extended in [17] to BD of time series. Here, we extend these results to BD of images.

A. Relative optimization algorithm

The main idea of relative optimization is to iteratively produce source estimate and use it as the current observation. This yields the following algorithm:

```
Relative optimization algorithm
1) Start with $w^{(0)}$, and with $x^{(0)} = x$.
2) For $k = 0, 1, 2, ...$, until convergence
    a) Start with: $w = \delta$.
    b) Using an unconstrained optimization method, find
       $w^{k+1}$ such that $\ell(x^{(k)}; w^{(k+1)}) < \ell(x^{(k)}; \delta)$.
    c) Update the source estimate $x^{(k+1)} = w^{(k+1)} * x^{(k)}$.
3) End loop
```

The restoration kernel estimate at k-th iteration is

$$w = w^{(0)} * w^{(1)} * ... * w^{(k)}, \tag{6}$$

and the source estimate is $\tilde{s} = x^{(k)}$. The method allows to construct large restoration kernels using a set of low-order factors. The algorithm assumes infinite memory and produces a restoration kernel of order growing at each iteration. In real applications it might be necessary to limit the support of the restoration kernel. This can be done by cropping w after each update.

Another remarkable property of the relative optimization algorithm is its equivariance: the relative optimization algorithm is *equivariant*, i.e. its step at iteration k depends only on $g^{(k-1)} = a * w^{(0)} * ... * w^{(k-1)}$. This property follows in a straightforward manner from the definition of the relative optimization algorithm. Equivariance implies that for any invertible kernel b, the estimator $\tilde{w}(x)$ of the restoration kernel w given the observation x, obtained by minimization of the target function $\ell(x; w)$ obeys [18]

$$\tilde{w}(b * x) = b^{-1} * \tilde{w}(x),$$

where b^{-1} denotes the inverse of b. The latter means that the parameters to be estimated (in our case, coefficients $w_{\boldsymbol{n}}$ of the restoration kernel) form a group. This is indeed the case for invertible kernels with the convolution operation. It must be noted, however, that when the restoration filter support is limited by cropping, equivariance holds only approximately.

B. Newton method

Newton method is often used for unconstrained optimization, since it provides a very fast (quadratic) rate of convergence. In this approach, the direction d at each iteration is given by solution of the linear system [23]

$$\nabla^2 \ell \cdot d = -\nabla \ell. \tag{7}$$

Since the objective function is non-convex, in order to guarantee descent direction, positive definiteness of the Hessian is forced by using modified Cholesky factorization [23]. Having the direction d, the new iterate $w^{(k+1)}$ is given by

$$w^{(k+1)} = w^{(k)} + \mu^{(k)} d^{(k)},$$

where $\mu^{(k)}$ is the step size determined, e.g., by backtracking line search restricted to the subspace where the inverse of the restoration kernel is stable, i.e. $\log|\det \mathcal{W}| > -\infty$ [18].

Newton method is used in Step 4 of the relative optimization algorithm [15], [17]; such an optimization algorithm will be referred to as the *relative Newton* method. Apart from gradient and Hessian evaluation, required at each relative Newton iteration, additional $\mathcal{O}(\frac{1}{6}K_M^3 K_N^3 + K_M^2 K_N^2)$ operations are required for solution of the Newton system (7) using modified Cholesky factorization [23].

C. Fast relative Newton step

Practical use of the relative Newton step is limited to cases of small M, N, M_X, N_X, due to the complexity of Hessian construction and solution of the Newton system. This complexity can be significantly reduced if special Hessian structure at the minimum is exploited.

Substituting $w = \delta$ and $x = cs$ to (5) yields $\hat{w} \equiv 1$ and $y = cs$, from where

$$\frac{\partial^2 \ell}{\partial w_{\boldsymbol{i}} \partial w_{\boldsymbol{j}}} = \delta_{\boldsymbol{i}+\boldsymbol{j}} + \frac{1}{M_F N_F} \sum_{\boldsymbol{n}} \varphi''(cs_{\boldsymbol{n}}) cs_{\boldsymbol{n}-\boldsymbol{i}} cs_{\boldsymbol{n}-\boldsymbol{j}}.$$

It can be easily seen that the first term contributes a unit anti-diagonal to the Hessian.

For sufficiently large sample size $M_X \times N_X$, the second term becomes approximately

$$\frac{\partial^2 \ell_2}{\partial w_{\boldsymbol{i}} \partial w_{\boldsymbol{j}}} \approx \mathbb{E}\varphi''(cs_{\boldsymbol{n}}) cs_{\boldsymbol{n}-\boldsymbol{i}} cs_{\boldsymbol{n}-\boldsymbol{j}}.$$

Without loss of generality, let us assume that $s_{\boldsymbol{n}}$ is zero-mean. Since s is i.i.d.,

$$\frac{\partial^2 \ell_2}{\partial w_{\boldsymbol{i}} \partial w_{\boldsymbol{j}}} \approx \begin{cases} \alpha c^2 & : \ \boldsymbol{i} = \boldsymbol{j} = 0 \\ \gamma \sigma'^2 & : \ \boldsymbol{i} = \boldsymbol{j} \neq 0 \\ 0 & : \ \text{otherwise}, \end{cases}$$

where $\alpha = \mathbb{E}\varphi''(cs)(cs)^2$, $\gamma = \mathbb{E}\varphi''(cs)$, $\sigma^2 = \mathbb{E}s^2$, and $\sigma' = c\sigma$. We conclude that $\nabla^2 \ell(cx; \delta)$ has an approximate diagonal-anti-diagonal form (see Figure 1). When $\gamma \sigma'^2 \gg 1$, $\nabla^2 \ell(cx; \delta)$ is approximately diagonal. When $\gamma^2 \sigma'^4 < 1$, the Hessian at

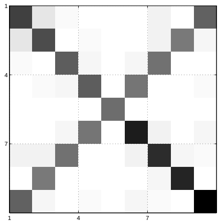

Fig. 1. Hessian structure for $w = \delta$ with $M = N = 1$ (3×3 kernel). White stands for near-zero elements of the matrix.

the solution point is not positive-definite, which means that the QML estimator is *asymptotically unstable*. This issue is addressed in depth in [18].

Using the diagonal approximation, which is valid for $\gamma\sigma'^2 \gg 1$, the Newton system (7) can be solved as a set of $K_M K_N$ independent linear equations

$$\left(\nabla^2 \ell\right)_{kk} \cdot d_k \quad = \quad -\left(\nabla \ell\right)_k ,$$

for $k = 1, ..., K_M K_N$. In order to guarantee decent direction and avoid saddle points, we force positive definiteness of the Hessian by forcing small diagonal elements to be above some positive threshold. For $\gamma\sigma'^2 \sim 1$, the diagonal-anti-diagonal approximation of the Hessian should be used, which allows to reduce Newton system solution to regularized solution of a set of 2×2 systems of the form

$$\begin{aligned} \left(\begin{array}{cc} \left(\nabla^2 \ell\right)_{kk} & 1 \\ 1 & \left(\nabla^2 \ell\right)_{K-k,K-k} \end{array} \right) \left(\begin{array}{c} d_k \\ d_{K-k} \end{array} \right) \\ = \left(\begin{array}{c} -\left(\nabla \ell\right)_k \\ -\left(\nabla \ell\right)_{K-k} \end{array} \right), \end{aligned} \tag{8}$$

and an additional 1×1 system

$$\left(\nabla^2 \ell\right)_{\frac{K}{2}} \cdot d_{\frac{K}{2}} \quad = \quad -\left(\nabla \ell\right)_{\frac{K}{2}} ,$$

where $K = K_M + K_N + 1$. Regularization is performed by forcing positive definiteness of each of the 2×2 submatrices in (8) by inverting the sign of negative eigenvalues and forcing small eigenvalues to be larger than some positive threshold.

When the diagonal or the diagonal-anti-diagonal approximations are used, fast relative Newton algorithm requires $\mathcal{O}\left(M_X N_X + 4 M_X N_X \log_2 M_X N_X\right)$ operations for approximate Hessian construction, which is of the same order as gradient computation. Additional $\mathcal{O}\left(K_M K_N\right)$ operations are required for approximate Hessian inversion in case of diagonal approximation, and slightly more in case of the diagonal-anti-diagonal approximation. This is compared to $\mathcal{O}\left(M_X N_X + K_M K_N \left[4 M_X N_X \log_2 M_X N_X + M_X N_X\right]\right)$ operations for exact Hessian evaluation and additional $\mathcal{O}\left(\frac{1}{6}(K_M K_N)^3 + (K_M K_N)^2\right)$ computations for exact Newton system solution required for the full relative Newton step.

IV. Optimal sparse representations of images

The QML framework presented in Section II is valid for sparse sources; this type of a prior of source distribution is especially convenient for the underlying optimization problem due to its convexity, and results in very accurate deconvolution. However, natural images arising in the majority of BD applications can by no means be considered to be sparse in their native space of representation (usually, they are sub-Gaussian), and thus such a prior is not valid for ”real-life” sources. On the other hand, it is very difficult to model actual distributions of natural images, which are often multi-modal and non-log-concave. This apparent gap between a simple model and the real world calls for an alternative approach. In this section, we show how to overcome this problem using sparse representation.

A. Sparsification

While it is difficult to derive a prior suitable for natural images, it is much easier to transform an image is such a way that it fits some universal prior. In this study, we limit our attention to the sparsity prior, and thus discuss sparsifying transformations, though the idea is general and is suitable for other priors as well. The idea of *sparsification* was successfully exploited in BSS [16], [22], [24], [25]. It was shown in [22] that even such simple transformation as a discrete derivative can make the image sparse. However, most of these transformations were derived from empirical considerations. Here we present a criterion for finding optimal sparsifying transformations.

Let assume that there exists a *sparsifying transformation* $\mathcal{T}_{[s]}$, which makes the source s sparse (wherever possible, the subscript s in $\mathcal{T}_{[s]}$ will be omitted for brevity). In this case, our algorithm is likely to produce a good estimate of the restoration kernel w since the source properties are in accord with the sparsity prior. The problem is, however, that in the BD setting, s is not available, and $\mathcal{T}$ can be applied only to the observation x. Hence, it is necessary that the sparsifying transformation commute with the convolution operation, i.e.

$$(\mathcal{T}s) * a = \mathcal{T}(s * a) = \mathcal{T}x, \tag{9}$$

such that applying $\mathcal{T}$ to x is equivalent to applying it to s. Obviously, $\mathcal{T}$ must be a shift-invariant (SI) transformation.[3]

In the most general case, the sparsifying kernel can be *complex*: $t = t_R + \imath t_I$. After sparsification with such t, the prior term ℓ_2 of the likelihood function, in which the real absolute value is replaced by the complex one, becomes

$$\sum_{\boldsymbol{n}} |(\mathcal{T}y)_{\boldsymbol{n}}| = \sum_{\boldsymbol{n}} \sqrt{(t_R * y)^2_{\boldsymbol{n}} + (t_I * y)^2_{\boldsymbol{n}}}. \tag{10}$$

The latter is a generalization of the 2D *total-variation* (TV) norm. The TV norm, which has been found to be a successful prior in numerous studies related to signal restoration and denoising [26]–[28], and was also used by Chan and Wong as a regularization in BD [29], is obtained when t_R, t_I are chosen to be discrete x- and y-directional derivatives.

For simplicity, we limit our attention in this study to linear shift-invariant (LSI) transformations, i.e. $\mathcal{T}$ that can be represented by convolution with a *sparsifying kernel*

$$\mathcal{T}s = t * s. \tag{11}$$

[3] In BSS problems, the sparsifying transformation needs to be linear and not necessarily shift-invariant, e.g. wavelet packets were used for sparsification in [16], [24].

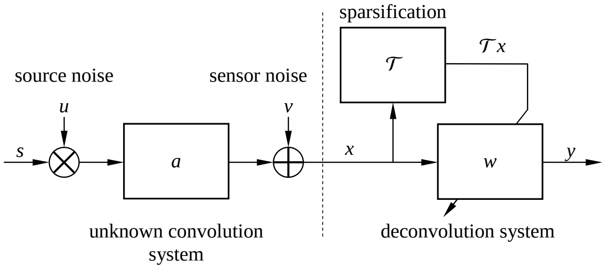

Fig. 2. Scheme of blind deconvolution using sparsification.

Thus, we obtain a general BD algorithm, which is not limited to sparse sources. We first sparsify the observation data x by convolving it with t (which has to be found in a way described in Section IV-C), and then apply the sparse BD algorithm on the result $x * t$. The obtained restoration kernel w is then applied to x to produce the source estimate.

B. The sparsifying kernel

An important practical issue is how to find the kernel t. By definition t must produce a sparse representation of the source; it is obvious that t would usually depend on s, and also, t does not necessarily have to be stable, since we use it as a pre-processing of the data and hence never need its inverse.

Let assume that the source s is given (this is, of course, impossible in reality; the issue of what to use instead of s will be addressed in Section IV-C). It is desired that the unity restoration kernel δ be a local minimizer of the QML function (2), given the transformed source $s * t$ as an observation, i.e.:

$$\nabla \ell(s * t; \delta) = 0. \tag{12}$$

Informally, this means that $s*t$ optimally fits the sparsity prior (at least in local sense). Due to the equivariance property, (12) is equivalent to $\nabla \ell(s; t) = 0$. In other words, we can define the following optimization problem:

$$\min_t \ell(s; t), \tag{13}$$

whose solution is the optimal sparsifying kernel for s. This problem is equivalent to the problem solved for deconvolution itself, with the exception of the stability condition, which is not needed here since t is not necessarily invertible. The term $\ell_1(t)$ in $\ell(s; t)$ eliminates the trivial solution $t = 0$.

Figures 3 and 4 (first row) show examples of optimal sparsifying transformations of 1D and 2D signals. In the 1D case, a row from a natrual image was taken; the optimal sparsifying kernel is a discrete derivative. In the 2D case of a block signal, as expected intuitively, the optimal sparsifying kernel is a corner detector.

C. Finding the sparsifying kernel by training

Since the source image s is not available, computation of the sparsifying kernel by the procedure described in Section IV-B is possible only theoretically. However, empirical results indicate that for images belonging to the same class, the proper sparsifying kernels are sufficiently similar.

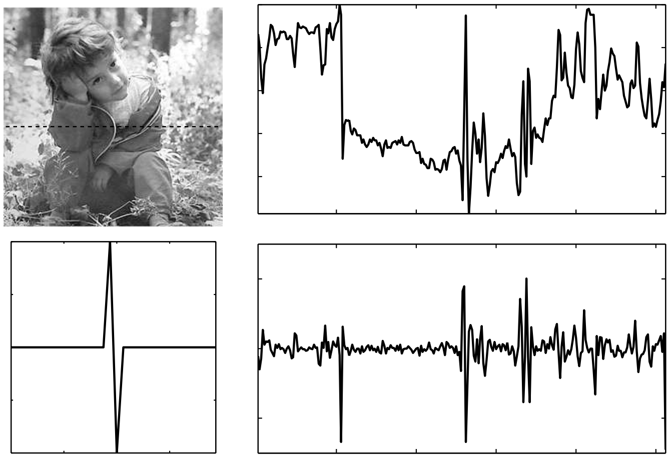

Fig. 3. A 1D example of optimal sparsification. First row, left: image; right: a 1D signal (line 140 from the image). Second row, left: optimal sparsifying kernel; right: sparsified signal.

Let $\mathcal{C}$ denote a class of images, e.g. human faces, and assume that the unknown source s belongs to $\mathcal{C}$. We can find images $s^{(1)}, s^{(2)}, ..., s^{(N_T)} \in \mathcal{C}$ and use them to find the optimal sparsifying kernel of s. Optimization problem (13) becomes in this case

$$\min_t \left\{ \ell_1(t) + \frac{1}{N_T} \sum_{n=1}^{N_T} \ell_2(s^{(i)} * t) \right\}, \tag{14}$$

i.e. T is required to be the optimal sparsifying kernel for all $s^{(1)}, s^{(2)}, ..., s^{(N_T)}$ simultaneously. The images $s^{(1)}, s^{(2)}, ..., s^{(N_T)}$ constitute a *training set*, and the process of finding such t as *training*. Given that the images in the training set are ”sufficiently similar” to t, the optimal sparsifying kernel obtained from (14) is similar enough to $t_{[s]}$.

Being identical to the deconvolution problem, the training problem (14) can be solved using the methods presented beforehand. A convex $\varphi(s)$ such as the smoothed absolute value, results in a convex prior term ℓ_2. However, the log-spectrum term ℓ_1 is non-convex and, consequently, the QML function is non-convex (e.g., if w^* is a minimizer of $\ell(x; w)$, then $-w^*$ is also a minimizer). Yet, our observations show that the QML function is well-behaving and when the relative update is used, the minimization algorithm does not converge to ”wrong” local minima, especially when the restoration kernel stability is enforced. Similar behavior of QML estimators is known in the context of BSS as well [14], [15]. This amazing fact deserves additional research.

D. Sparsification in the presence of noise

Although we have limited our attention to noiseless BD, it is important to emphasize that the sparsification framework is applicable in the presence of noise. In case of *source noise*, when s is noisy, for example, due to a low number of emitted photons, the training can be performed on training images, degraded by the expected level of noise. When the source noise is additive, the function $\varphi(s)$ can be modified in order to account for the noise distribution. For a detailed

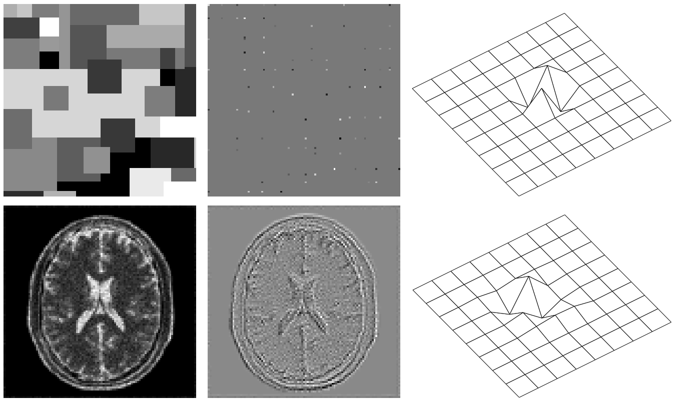

Fig. 4. Optimal sparsification of images. First column: original image; second column: sparsified image; third column: optimal sparsifying kernel. First row: synthetic block image; second row: brain image with Poisson noise.

discussion, see [18]. Figure 4 (second row) depicts the optimal sparsification of a medical image contaminated by Poisson noise. The optimal sparsifying kernel (constructed using the relative Newton algorithm) in this case is a smoothed corner detector.

The problem of *sensor noise* is more complicated. Although the relative Newton algorithm was found to be robust to noise levels of up to about 10 dB [17], the use of sparsification kernels (which are typically high-pass filters, since by their very nature sparse signals have high-frequency components) is liable to lower the SNR. To cope with this problem, the signal should be smoothed with a low-pass filter f and afterwards the sparsifying kernel t should be applied. Due to commutativity of the convolution, it is equivalent to carrying out the sparsification with a smoothed kernel $t * f$.

V. Simulation results

The QML-based deconvolution approach was tested in three experiments under zero-noise conditions. In the first experiment, the goal was to compare between the performance of fast relative Newton and full relative Newton algorithms. The purpose of the second experiment was to demonstrate the utility of the training approach for finding optimal sparse representations. In the third experiment, we used the sparsification approach to perform deconvolution of natural images. The fourth experiment demonstrates the performance of QML blind deconvolution in the presence of Poisson source noise. As a criterion for evaluation of the reconstruction quality, we used the signal-to-interference-ratio (SIR) in sense of the L_2, L_∞ norms, and the peak SIR (PSIR) in dB units [18].

A. Deconvolution of sparse images

An 101×101 Gauss-Bernoully i.i.d. image with $\rho = 0.2$ [18] was used as the source in the first experiment. The image was convolved with a 3×3 FIR kernel with a slowly-decaying inverse (see Figure 6). Full Newton and fast relative Newton (with a diagonal Hessian approximation) were used to estimate the inverse kernel. 3×3, 5×5, 7×7, and 9×9 restoration kernels were used. The smoothing parameter was set to $\lambda = 10^{-2}$. Optimization was terminated when $\|\nabla \ell\|$ reached 10^{-10}. Gradient norms, SIR and SIR$_\infty$ were measured as a function of CPU time[4] and iteration number.

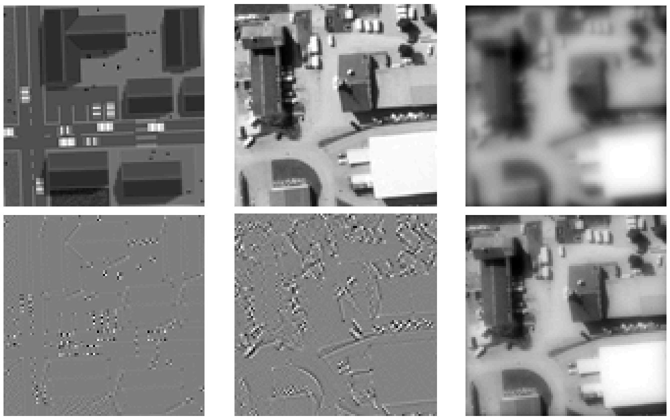

Fig. 5. First row, left to right: training synthetic image; source aerial image s; blurred image $s * w$. Second row, left to right: sparsified training image; sparsified source; restored image.

The experiments indicate convergence of both algorithms (Figure 6). The fast relative Newton converged about 10 times faster in terms of SIR, compared with the full Newton step. For the same values of M, N, the obtained restoration quality of the fast relative Newton algorithm, compared to the full Newton step, was better by about 2–5 dB (in terms of SIR and SIR$_\infty$), since the effective restoration kernel was of higher order.

B. Training

In the second experiment, a real aerial photo of a factory was used as the source image, and a synthetic one (drawn using PhotoShop) as the training image (Figure 5). A 3×3 sparsifying kernel is found by training on a single image, then the same kernel is used as a pre-processing for BD applied to a different blurred source image from the same class of images. The source image was convolved with a symmetric FIR 31×31 Lorenzian-shaped blurring kernel. Deconvolution kernel was of size 3×3. The sparsifying kernel obtained by training was very close to a corner detector. The signal-to-interference ratio in the deconvolution result was $\mathrm{SIR} = 20.16$ dB, $\mathrm{SIR}_\infty = 25.72$ dB.

C. Deconvolution of natural images

In the third experiment, four natural source images were used: s_1 (Susy), s_2 (Aerial), s_3 (Gabby) and s_4 (Hubble). Nearly-stable Lorenzian-shaped kernels were applied to the corresponding sources. This type of kernels characterizes scattering media, such as biological fluids and aerosols found in the atmosphere [30]. Quality of the degraded images in terms of SIR, SIR$_\infty$ and PSIR is presented in Table I.

Fast relative Newton step with kernel size set to 3×3 was used in this experiment. The smoothing parameter was set to

[4] All algorithms were implemented in MATLAB and executed on an ASUS portable computer with Intel Pentium IV Mobile processor and 640MB RAM.

TABLE I

SIR, SIR_∞ AND PSIR OF THE OBSERVED IMAGES.

Source		SIR [dB]	SIR_∞ [dB]	PSIR [dB]
s_1	Susy	-1.46	7.84	22.83
s_2	Aerial	-1.46	7.84	19.89
s_3	Gabby	4.90	11.55	30.41
s_4	Hubble	3.40	10.65	27.18

TABLE II

SIR, SIR_∞ AND PSIR OF THE RESTORED IMAGES.

Source		SIR [dB]	SIR_∞ [dB]	PSIR [dB]
s_1	Susy	17.80	22.21	27.23
s_2	Aerial	17.04	23.55	25.16
s_3	Gabby	19.32	23.81	40.64
s_4	Hubble	14.51	17.16	33.67

$\lambda = 10^{-2}$. Corner detector was used as the sparsifying kernel. Optimization was terminated when the gradient norm reached 10^{-10}. Convergence was achieved in $10-20$ iterations (about 10 sec). The images are depicted in Figure 7. Restoration quality results in terms of SIR, SIR_∞ and PSIR are presented in Table II. The use of the total variation prior in the place of ℓ_2 with the optimal sparsifying kernel yielded relatively poor results.

D. Deconvolution in presence of source noise

In the fourth experiment, we performed deconvolution of a source contaminated by multiplicative Poisson noise with maximum SNR of 10 dB. Such type of noise arises in emission processes (e.g. emission tomography and optical imaging with low photon yield). A brain image was used as the source. The noisy source was convolved with a Lorentzian-shaped blurring kernel, resulting in $\text{SIR} = 3.99$ dB, $\text{SIR}_\infty = 11.50$ dB and $\text{PSIR} = 23.43$ dB in the observed image.

The sparisification kernel was obtained by training on a different brain image (see Figure 4, second row) and had the form of a smoothed corner detector. The restoration resulted in $\text{SIR} = 12.44$ dB, $\text{SIR}_\infty = 16.50$ dB and $\text{PSIR} = 33.20$ dB (Figure 8).

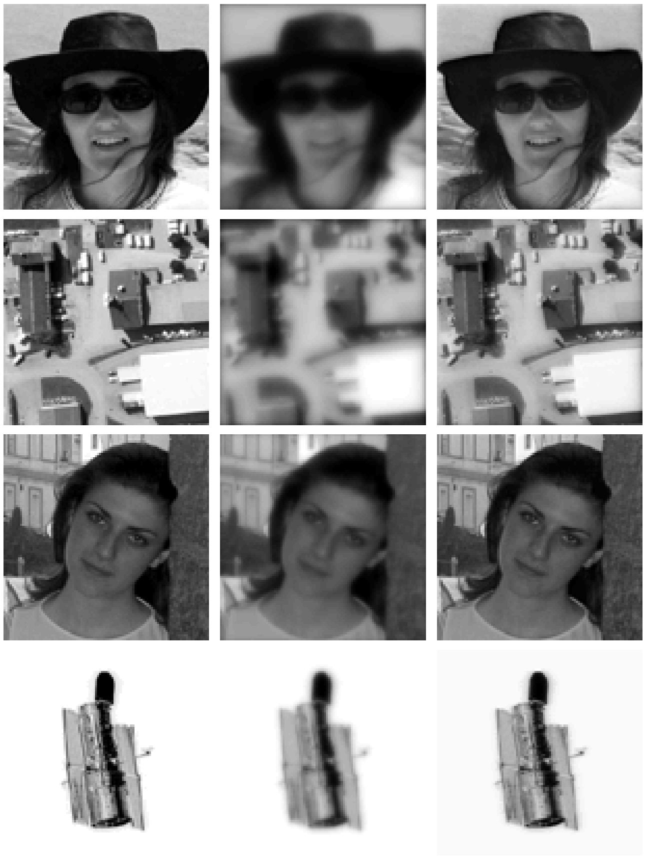

Fig. 7. First column: Source images, second column: blurred images, third column: Restoration results using the QML deconvolution approach.

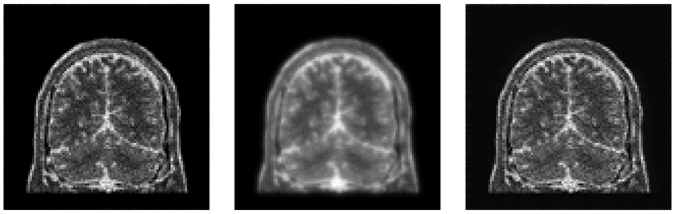

Fig. 8. Left to right: source image contaminated by Poisson noise; blurred noisy source image; deconvolution results.

VI. Conclusion

The QML framework, recently presented in the context of 1D deconvolution [17] is also attractive for BD of images. We presented an extension of the relative optimization approach to QML BD in the 2D case and studied the relative Newton method as its special case.

Similarly to previous works addressing deconvolution in other spaces (e.g. [31]) and our studies of using sparse representation in the context of BBS, in BD the sparse prior appears very efficient as well. We showed a training approach for finding optimal sparse representations, in order to yield a general-purpose BD method. A particular class of LSI sparsifying transformations generalizes some previous results such as the total variation prior [26]–[28]. We also showed how optimal sparsifying transformations can be found by training.

Simulation results demonstrated the efficiency of the proposed methods. Potential applications of our approach are in optics, remote sensing, microscopy and biomedical imaging, especially where the SNR is moderate. This approach is especially accurate and efficient in problems involving slowly-decaying (e.g. Lorenzian-shaped) kernels, which can be approximately inverted using a kernel with small support. Such kernels are typical of imaging through scattering media.

Acknowledgement

The authors thank Susy Pitzanti and Gabriella Sbordone for permission to use their photographs as test images in simulations. We acknowledge the anonymous referees for constructive comments, which helped us improve the paper.

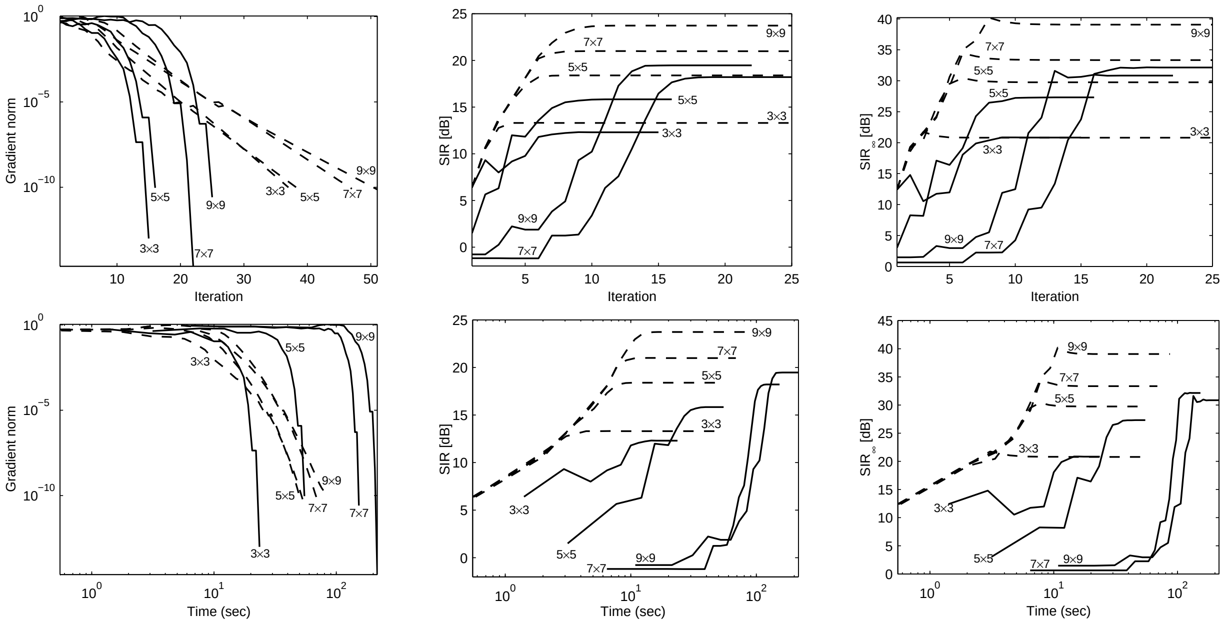

Fig. 6. Convergence of the Newton (solid) and of the fast relative Newton (dashed) methods, for various sizes of the restoration kernel vs. the number of iterations (first row) and the CPU time (second row).

REFERENCES

[1] T. J. Schulz, “Multiframe blind deconvolution of astronomical images,” *J. Opt. Soc. Am. A*, vol. 10, no. 5, pp. 1064–1073, 1993.

[2] M. Bertero and P. Boccacci, “Image restoration methods for the large binocular telescope,” *Astron. Astrophys. Suppl. Ser*, vol. 147, pp. 323–333, 2000.

[3] J. P. Muller, Ed., *Digital Image Processing in Remote Sensing*. Taylor & Francis, Philadelphia, 1988.

[4] M. Mignotte and J. Meunier, “Three-dimensional blind deconvolution of SPECT images,” *IEEE Trans. on Biomedical Eng.*, vol. 47, no. 2, pp. 274–280, 2000.

[5] D. Adam and O. Michailovich, “Blind deconvolution of ultrasound sequences using non-parametric local polynomial estimates of the pulse,” *IEEE Trans. on Biomedical Eng.*, vol. 42, no. 2, pp. 118–131, 2002.

[6] T. Wilson and S. J. Hewlett, “Imaging strategies in threedimensional confocal microscopy,” in *Proc. SPIE 1245*, 1991, pp. 35–45.

[7] T. J. Holmes, S. Bhattacharyya, J. A. Cooper, D. Hanzel, V. Krishnamurthi, W. Lin, B. Roysam, D. H. Szarowski, and J. N. Turner, “Light microscopic images reconstructed by maximum likelihood deconvolution,” in *Handbook of Biological and Confocal Microscopy, 2nd ed.*, J. B. Pawley, Ed. Plenum Press, New York, 1995.

[8] D. Kundur and D. Hatzinakos, “Blind image deconvolution,” *IEEE Sig. Proc. Magazine*, pp. 43–64, May 1996.

[9] M. Cannon, “Blind deconvolution of spatially invariant image blurs with phase,” *IEEE Trans. Acoust. , Speech, Signal Process.*, vol. 24, no. 1, pp. 58–63, February 1976.

[10] M. M. Chang, A. M. Tekalp, and A. T. Erdem, “Blur identification using the bispectrum,” *IEEE Trans. Signal Processing*, vol. 39, no. 10, pp. 2323–2325, October 1991.

[11] R. Fabian and D. Malah, “Robust identification of motion and outoffocus blur parameters from blurred and noisy images,” *CVGIP: Graphical Models and Image Processing*, vol. 53, no. 5, pp. 403–412, July 1991.

[12] E. Thiébaut and J.-M. Conan, “Strict a priori constraints for maximum likelihood blind deconvolution,” *J. Opt. Soc. Am. A*, vol. 12, no. 3, pp. 485–492, 1995.

[13] T. F. Chan and C. K. Wong, “Total variation blind deconvolution,” Tech. Rep., 1996.

[14] D. Pham and P. Garrat, “Blind separation of a mixture of independent sources through a quasi-maximum likelihood approach,” *IEEE Trans. Sig. Proc.*, vol. 45, pp. 1712–1725, 1997.

[15] M. Zibulevsky, “Sparse source separation with relative Newton method,” in *Proc. ICA2003*, April 2003, pp. 897–902.

[16] P. Kisilev, M. Zibulevsky, and Y. Zeevi, “Multiscale framework for blind source separation,” *JMLR*, 2003, in press.

[17] A. M. Bronstein, M. M. Bronstein, and M. Zibulevsky, “Blind deconvolution with relative Newton method,” Technion, Israel, Tech. Rep. 444, October 2003. [Online]. Available: http://visl.technion.ac.il/bron/michael

[18] ——, “Quasi maximum likelihood blind deconvolution of images using optimal sparse representations,” Technion, Israel, Tech. Rep. 455, December 2003. [Online]. Available: http://visl.technion.ac.il/bron/michael

[19] M. Zibulevsky, B. A. Pearlmutter, P. Bofill, and P. Kisilev, “Blind source separation by sparse decomposition,” in *Independent Components Analysis: Principles and Practice*, S. J. Roberts and R. M. Everson, Eds. Cambridge University Press, 2001.

[20] S.-I. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, “Multichannel blind deconvolution and equalization using the natural gradient,” in *Proc. SPAWC*, April 1997, pp. 101–104.

[21] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[22] A. M. Bronstein, M. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi, “Separation of reflections via sparse ICA,” in *Proc. IEEE ICIP*, 2003.

[23] D. P. Bertsekas, *Nonlinear Programming (2nd edition)*. Athena Scientific, 1999.

[24] M. Zibulevsky and B. A. Pearlmutter, “Blind source separation by sparse decomposition,” *Neural Computation*, vol. 13, no. 4, 2001.

[25] M. S. Lewicki and B. A. Olshausen, “A probabilistic framework for the adaptation and comparison of image codes,” *J. Opt. Soc. Am. A*, vol. 16, no. 7, pp. 1587–1601, 1999.

[26] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D*, vol. 60, pp. 259–268, 1992.

[27] P. Blomgren, T. F. Chan, P. Mulet, and C. Wong, “Total variation image restoration: numerical methods and extensons,” in *Proc. IEEE ICIP*, 1997.

[28] T. F. Chan and P. Mulet, “Iterative methods for total variation image restoration,” *SIAM J. Num. Anal*, vol. 36, 1999.

[29] T. F. Chan and C. K. Wong, “Total variation blind deconvolution,” in *Proc. ONR Workshop*, 1996.

[30] M. Moscoso, J. B. Keller, and G. Papanicolaou, “Depolarization and blurring of optical images by biological tissue,” *J. Opt. Soc. Am. A*, vol. 18, no. 4, pp. 948–960, 2001.

[31] M. R. Banham and A. K. Katsaggelos, “Spatially adaptive wavelet-based multiscale image restoration,” *IEEE Trans. Image Processing*, vol. 5, pp. 619–634, April 1996.