RESEARCH   **Open Access**

Efficient shift-variant image restoration using deformable filtering (Part I)

David Miraut[1] and Javier Portilla[2*]

Abstract

In this study, we propose using the least squares optimal deformable filtering approximation as an efficient tool for linear shift variant (SV) filtering, in the context of restoring SV-degraded images. Based on this technique we propose a new formalism for linear SV operators, from which an efficient way to implement the transposed SV-filtering is derived. We also provide a method for implementing an approximation of the regularized inversion of a SV-matrix, under the assumption of having smoothly spatially varying kernels, and enough regularization. Finally, we applied these techniques to implement a SV-version of a recent successful sparsity-based image deconvolution method. A high performance (high speed, high visual quality and low mean squared error, MSE) is demonstrated through several simulation experiments (one of them based on the Hubble telescope PSFs), by comparison to two state-of-the-art methods.

Keywords: shift variant filtering, deformable kernel, deformable filtering, Singular Value Decomposition, integration kernel, point spread function, restoration, sparsity, wavelet frames

1 Introduction

For many real imaging devices, especially those designed to be wide-angle, small, cheap and/or extra robust (for instance, because of having very simple optics), it may still be reasonable to consider their degradation model as linear, but they may significantly depart from shift-invariance (SI). That is, different areas of the image support may present substantially different blurring functions (shift-variant (SV), behavior). In those cases a local point spread function (PSF) for each spatial location of the image support must be considered, constituting globally a *PSF field*. Then, from a digital restoration perspective, this new scenario becomes much harder, compared to the SI case.

Whereas in the (linear) SI case matrix inversion can be efficiently done in the Fourier domain (eigenvalues/ eigenvectors of the convolution matrix), obtaining the eigenvectors and eigenvalues of an arbitrary square SV matrix of a large size is computationally unaffordable in most situations. Nevertheless, some authors have approached the large SV-matrix inversion problem, by

using the Singular Value Decomposition (SVD [1]) directly on the blur matrix. This requires some approximations to make the SVD tractable, like hierarchically extracting singular vectors until their associated singular values become irrelevant (see, e.g., [2,9]). Other researchers have addressed the problem of inverting the linear transformation in a stable, iterative way, mostly by using the so-called Krylov sub-spaces and related techniques [5,10]. A serious problem of all these methods, besides their high computational cost, is their lack of modeling upon which to establish a criterion for fixing the amount of required regularization. For instance, trying to do a "practical pseudo-inverse" of the linear transform, by setting to zero the small singular values and inverting the rest is quite arbitrary for two reasons: (1) the choice of the considered threshold, and (2) the fact of using an *all-or-nothing* weighting function for the singular values. In this sense, even a Wiener estimate, which applies a smooth mask on the singular values, is both more correct and should, if properly used, provide better results in general. Another important problem is that, despite their conceptual and computational complexity, the referred methods' performance is seriously limited, because they are linear. Whenever the amount of noise in the observation is

* Correspondence: javier.portilla@csic.es
[2]Instituto de Optica, Consejo Superior de Investigaciones Científicas, Madrid, Spain
Full list of author information is available at the end of the article

significant, the reduced ability of linear methods to simultaneously compensate for the blur and suppress noise will manifest in terms of a poor trade-off between these two kind of degradations in the restored image. Only non-linear methods (which typically alternate linear and non-linear steps) are able to effectively regenerate lost spatial frequencies (like blurred edges), and, at the same time, to suppress a large amount of noise. Not surprisingly, the extra computation of making a costly linear inversion iterative within a non-linear method is rarely held [3,11].

An alternative to attack the SV blurring inversion problem globally consists of making a spacial partition of the image, i.e. to divide it into non-overlapping regions, and then modeling the pixels within each region as convolved by a single (semi-local) kernel. This strategy is used by the so-called *sectional methods* (e.g., [7,12]). We can improve significantly the results and avoid annoying boundary artifacts by allowing for a certain amount of overlapping of adjacent regions. In such a way, the signal in overlapping areas can be modeled as a linear combination of the convolutions of the image with the kernels corresponding to the overlapping regions, using spatially varying weights [4,8,13]. However, the best possible approximation of the local kernels as linear interpolations of a set of reference kernels (for a certain error measurement) is obtained when no spatial constraints are imposed to the linear coefficients. That is, nothing prevents us from using all the reference kernels, not only the adjacent, to improve the approximation of the local kernel as local linear combinations of them. Furthermore, we can take an even larger step farther from arbitrary decisions, by no longer fixing a priori a set of reference kernels. Instead, we may try to optimize them as well, by minimizing a measurement of the average interpolation error (typically, the mean squared error, MSE) for the whole image support. If we look for both a given number of unknown reference kernels and their corresponding unknown weights to jointly minimize the quadratic error of the local integration kernels (IKs) we want to approximate, we are facing the so-called total least squares regression problem [14]. Its solution can be obtained through the SVD of the matrix obtained by stacking all the local IKs (vectorized) we want to interpolate. In 1995 this technique was proposed by Perona [15] for having a convenient and efficient way to obtain linearly *deformable kernels*, that is, kernels whose shape changes in a certain desired fashion by means of linearly interpolating a set of reference kernels specifically designed for that task. Perona's main motivation was to obtain a set of *tunable* kernels for mimicking early vision. However, another direct application of this approach is SV filtering (see, e.g., [16]) in image processing: the local integration of a signal with a locally varying linear combination of some reference kernels can be expressed as a locally varying linear combination of the convolutions of the original signal with those kernels (as explained in detail in Section 3). From a different, analysis-driven point of view, PCA-based optimal basis functions have been recently used to describe with increased accuracy the response of astronomical instruments [17-20] in a more compact and numerically-stable way than other recent techniques. In this study, we propose for the first time (to the best of our knowledge), to use this LS-optimal deformable filtering approximation as an efficient tool for restoring linearly SV-degraded images.

However, one must note that being able to efficiently perform SV filtering does not solve, by itself, the problem of SV-image restoration. Most restoration methods, including non-linear ones, require to perform regularized inversions of the SV-blurring matrix $\mathbf{H}$. Typically, the regularized inverse is computed as $(\mathbf{H}^T\mathbf{H} + \mathbf{R})^{-1}\mathbf{H}^T$, where $\mathbf{R}$ is a positive definite matrix (often chosen to be the identity matrix multiplied by a positive constant). As we can see, this computation requires both transposition and inverse operations of very big non-circulant (in the SV case) matrices. Following the deformable filtering approach, we propose in this study a new formalism for linear SV operators which it is applied to derive an efficient and arbitrarily accurate implementation of the transposed SV-filtering. We propose, too, an approximation for the SV-matrix inversion, under the assumption of having a smoothly varying PSF field and enough regularization as to ensure its local regularized inverse also changes smoothly in the space.

In this study, we have also devoted some attention to distinguish between two concepts that are sometimes wrongly taken as synonyms, or their meanings incorrectly exchanged, in the context of modeling linear SV degradations: the IK versus the PSF. Our endeavor here is not just for rigor sake: rather, it is practical and methodological. First, note that using the PSFs (one for each spatial location of the image support) to characterize the linear response of an imaging device is more natural than using the IKs. The reason is that the IKs, unlike the PSFs, are not directly observable. In spite of this fact, many SV-simulations start from considering a given set of IKs. Furthermore, very often those IKs are forced to be normalized to one in their integral, although there is no general physical reason why the IKs should integrate to a constant. On the other hand, under the simplifying assumption of neglecting losses in the imaging process, energy conservation may be argued to force the PSFs to integrate to one. Thus, contrarily to what is usual (e.g., [10,21]), we propose here to use the set of PSFs describing the linear response of the imaging

device (the PSF field), and not using explicitly the IKs, neither in the direct nor in the transposed SV filtering.

The set of techniques presented here are aimed at obtaining short restoration times (ideally real time, or close), whereas preserving the most relevant features of realistic degradations (specifically, their SV nature, under noise). Therefore, we chose a state-of-the-art, non-linear, image restoration generic method with a high speed potential, named L0-AbS [22] for testing them, which is applicable to any kind of linear blur (SI or not), and which assumes presence of additive white Gaussian noise. This method had previously been used only for deconvolution (SI case). Note that the emphasis in this article is not in the estimation procedure and its associated statistical model (the interested reader is referred to the original study [22], which has been improved and extended in [23,24]), but rather in a set of efficient solutions for adapting a (generic) restoration method assuming a linear degradation plus noise, to a SV-scenario.[a] We demonstrate a high performance (high speed, high visual quality, low MSE) through several simulation experiments (one of them using the Hubble telescope's PSFs), comparing to two reference methods.

This study is organized as follows. Section 2 poses the linear SV observation model, and introduce some general concepts relevant for the linear SV filtering and its (linear) compensation. Then, Section 3 describes what a deformable kernel is, and how the concept can be applied for SV-filtering. Section 3.2 introduces a new formalism to express a SV-filtering matrix as a sum of (pre- or post-) locally-masked convolutions, which makes the associated transposed SV-filtering easy to implement. It also provides a method to approximately invert smoothly varying PSFs, by combining the Fourier transform and the deformable filtering formulation. Finally, it briefly describes how to apply these techniques to the linear filtering stage of the method described in [22]. Section 5.3 shows and discuss some simulation results with test images and several SV degradations under additive white Gaussian noise. Results are compared to two state-of-the-art methods. Section 6 concludes the article.

2 Shift variant image filtering

We can express a SV 2-D signal filtering in a discrete domain as:

$$y_0(\mathbf{p}^0) = \sum_{\mathbf{p} \in D} x(\mathbf{p}) h(\mathbf{p}; \mathbf{p}^0), \tag{1}$$

where $y_0(\mathbf{p}^0)$ is the SV-filtered image at location $\mathbf{p}^0$, D is the discrete image support, x is the image being filtered, and $h(\mathbf{p}; \mathbf{p}^0)$, considered as a function of $\mathbf{p}$, is the IK at location $\mathbf{p}^0 : IK_{\mathbf{p}^0}(\mathbf{p}) = h(\mathbf{p}; \mathbf{p}^0)$. In this formalism,

the output image is obtained through inner products of the input image with spatially varying local IKs. This is the most usual approach to understand SV filtering, but not the only one. We can also express it in terms of the PSFs of the system. To express the PSFs in terms of h we substitute the input image by a Kronecker delta placed at location $\mathbf{q}$:

$$PSF_{\mathbf{q}}(\mathbf{p}^0) = \sum_{\mathbf{p} \in D} \delta(\mathbf{p} - \mathbf{q}) h(\mathbf{p}; \mathbf{p}^0) \tag{2}$$

$$= h(\mathbf{q}; \mathbf{p}^0). \tag{3}$$

We can interpret Equation 1 as a *linear superposition* of all the PSFs, each weighted by the value of the underlying image at its location [25]:

$$y_0(\mathbf{p}^0) = \sum_{\mathbf{p} \in D} x(\mathbf{p}) PSF_{\mathbf{p}}(\mathbf{p}^0). \tag{4}$$

Let us now re-write Equation 1 in terms of two indices, i and j, expressing a 2-D location in the image support, given a certain lexico-graphical order to map the support D into a finite 1-D sequence:

$$y_0(\mathbf{p}_j) = \sum_{i=1}^{N} x(\mathbf{p}_i) h(\mathbf{p}_i; \mathbf{p}_j).$$

If we call $h_{ij} = h(\mathbf{p}_i; \mathbf{p}_j)$, $y_{0j} = y_0(\mathbf{p}_j)$, and $x_i = x(\mathbf{p}_i)$, then, using matrix algebra, we may write the previous equation as:

$$\mathbf{y}_0 = \mathbf{H}\mathbf{x}. \tag{5}$$

It is immediate to check that, under this formalism, the vectorized IKs correspond to the rows of **H**, and the vectorized PSFs to its columns, as illustrated in Equations 6 and 7 in following section.

2.1 Integration kernel versus point spread function

Usually, having the set of PSFs (one for each location) to characterize a linear SV device is more natural than having the set of IKs. The main reason is that a PSF is the output of the system when an impulse (a point light source) is fed at its input. Therefore, a PSF is *observable* (though in practice there will always be some noise and other imperfections), unlike the IKs. Another reason to use the PSFs instead of the IKs to characterize the SV-filtering is that it seems more natural to impose a gain constraint (basically, a fixed value for their integral, usually chosen to be one) to the PSFs than to the IKs. For instance, in an optical system without significant losses it makes sense to assume unity gain, with respect to an ideal optical system with perfect (no blur, no optical distortion) behavior. Note that PSF and IK normalization are, in general, incompatible one of each other.

$$\begin{bmatrix} \boxed{\text{IK}_1} \\ \boxed{\text{IK}_1} \\ \vdots \\ \vdots \\ \boxed{\text{IK}_M} \end{bmatrix} \begin{vmatrix} \\ x \\ \end{vmatrix} = \begin{bmatrix} \langle \text{IK}_1, x \rangle \\ \langle \text{IK}_2, x \rangle \\ \vdots \\ \vdots \\ \langle \text{IK}_M, x \rangle \end{bmatrix} = \begin{vmatrix} \\ y \\ \end{vmatrix} \tag{6}$$

$$\begin{vmatrix} \boxed{\text{PSF}_1}\boxed{\text{PSF}_2}\boxed{\text{PSF}_N} \\ \cdots \end{vmatrix} \begin{vmatrix} \\ x \\ \end{vmatrix} = \boxed{\text{PSF}_1} \cdot x_1 + \boxed{\text{PSF}_2} \cdot x_2 + \cdots + \boxed{\text{PSF}_N} \cdot x_N = \begin{vmatrix} \\ y \\ \end{vmatrix} \tag{7}$$

On the other hand, interpreting the SV filtered image as a superposition of the local weighted PSFs, as in Equations 4 and 7, it is usually less convenient, in a computational and operational sense, than the more usual local inner product interpretation, using the IKs, as in Equation 6. Specifically, for the set of tools proposed in this study, it is essential to use the inner product interpretation (which will lead us to use locally weighted convolutions, as explained in following section). In this study, we propose a solution to have "the best of both worlds" of using PSF-based and IK-based formulations. First, we consider $\mathbf{H}^T$, instead of $\mathbf{H}$, and thus, we have inner products with PSFs, instead of with IKs. Then, we propose a very simple and efficient way to implement the original filtering from its transposed version. Furthermore, in many real restoration methods it is also necessary to apply $\mathbf{H}^T$, so it is also convenient to have an efficient approximation for it. We have assumed that $\mathbf{H}$ is square, unless stated otherwise. Finally, in this article we have not applied any rigorous boundary handling technique. Robust and stable boundary handling for image deblurring is a very important and difficult matter on its own (even in the SI case), and it has been left as future study. Details on how boundaries have been handled in the presented simulations are explained in Section 5.1.1.

3 Deformable kernels applied to SV filtering
3.1 Deformable kernels
The deformable kernel idea consists of approximating a discrete kernel whose shape we want to control in a certain way by doing linear combinations of some (unknown a priori, in a general case) reference kernels [15]. If we express our kernel as $k(\mathbf{p}; \mathbf{d})$, where $\mathbf{d}$ is a vector indicating a deformation parameter, we can, same as we did in previous section, express it as a matrix $\mathbf{K}$ whose elements are $k_{ij} = k(\mathbf{p}_i; \mathbf{d}_j)$, for $i = 1... N$ and $j = 1...M$. Thus, its column vectors contain the desired kernel, each with a different deformation parameter. Now,

by means of the SVD we solve the Total Least Squares problem of finding both the reference kernels and the interpolation functions that minimize, for a given number J of reference kernels, the mean squared error (MSE) of the linear combination [15]:

$$\mathbf{K} = \mathbf{U}_K \mathbf{S}_K \mathbf{V}_K^T = \sum_{r=1}^{R} \mathbf{u}_r s_r \mathbf{v}_r^T \simeq \sum_{r=1}^{J} \mathbf{u}_r s_r \mathbf{v}_r^T, \tag{8}$$

where $R = \text{rank}(\mathbf{K})$, holding the approximation whenever $\sum_{r=1}^{J} s_r^2 \simeq \text{tr}(\mathbf{SS}^T)$, and assuming $|s_n| \geq |s_m|$, for $n < m$. It is easier to interpret the previous equation by making the indices explicit:

$$k(\mathbf{p}_i; \mathbf{d}_j) = \sum_{r=1}^{R} u_r(\mathbf{P}_i) s_r v_r(\mathbf{d}_j) = \sum_{r=1}^{R} \alpha_r(\mathbf{d}_j) b_r(\mathbf{p}_i) \simeq \sum_{r=1}^{J} \alpha_r(\mathbf{d}_j) b_r(\mathbf{p}_i). \tag{9}$$

Here $\alpha_r(\mathbf{d}) = s_r v_r(\mathbf{p})$, for $r = 1...R$, are the LS-optimal interpolation functions, and $b_r(\mathbf{p}) = u_r(\mathbf{p})$ are the LS-optimal reference kernels [15]. In many real cases, there is a substantial amount of overlapping among the kernels with different deformations, so one can use a $J \ll R$.

3.2 From deformable kernels to SV deformable filtering
Now we try to use the technique of deformable filtering to efficiently implement an approximation of the transposed SV-filtering, which is made by using the local PSFs as local IKs (see Section 2.1). Let us assume that there is no magnification between input and output coordinates, and that the amount of geometrical (e.g., optical) distortion is negligible. Then each PSF is located around the position of the input impulse ($\mathbf{q}$, in Equation 2). If we shift each local PSF to compensate for this displacement, all local PSFs will be centered at zero. We can write $\hat{\mathbf{y}}_0 = \mathbf{H}^T \mathbf{x}$, and:

$$\hat{y}_0 = \sum_{i=1}^{N} x(\mathbf{p}_i) h(\mathbf{p}_0; \mathbf{p}_i) = \sum_{i=1}^{N} x(\mathbf{p}_i) h^0(\mathbf{p}_0 - \mathbf{p}_i; \mathbf{p}_0),$$

where now, using Equation 9, we can consider h^0 a deformable kernel, and thus, closely approximate it as a linear interpolation (depending on its location $\mathbf{p}_0$) of a reduced set of reference kernels, by applying a SVD to the matrix $\mathbf{H}^0$ made of all these centered PSFs, stacked by vectorized columns (see Section 3.1):

$$\begin{aligned} \hat{y}_0(\mathbf{p}_0) &\approx \sum_{i=1}^{N} x(\mathbf{p}_i) \sum_{r=1}^{J} \alpha_r(\mathbf{p}_0) b_r(\mathbf{p}_i - \mathbf{p}_0) \\ &= \sum_{r=1}^{J} \alpha_r(\mathbf{p}_0) \sum_{i=1}^{N} x(\mathbf{p}_i) b_r(\mathbf{p}_i - \mathbf{p}_0) \\ &= \sum_{r=1}^{J} \alpha_r(\mathbf{p}_0) [x * \tilde{b}_r](\mathbf{p}_0), \end{aligned} \tag{10}$$

where the symbol '*' denotes convolution, and $\tilde{b}_r$ are the reference kernels rotated 180°. Therefore, we have been able to approximate the transposed filtering operation as a sum of a (reduced, in favorable cases) number of convolutions post-masked (with a different mask each) in the spatial domain. Each of these convolutions can be done very efficiently, either in the Fourier domain, or directly, for small reference kernels. The idea of using deformable kernels for filtering was already in Perona's original study. However, it was not applied to perform SV filtering, but rather to obtain continuous *tunable* filters from a (reduced) set of discrete filters. Later on, several authors (including a coauthor of this study, see [16]) used it for SV filtering, although no one, to the best of our knowledge, for the purpose of image restoration.

We want to emphasize here that, whereas other SV-image restoration techniques use the SVD for directly inverting a linear SV blur, these previous methods–unlike ours–applied the SVD to the matrix made of the IKs/PSFs kernels *at their original spatial locations*. Because of the kernels' spatial shift, there is much less overlapping among kernels than when using the matrix made of the *centered* IKs/PSFs, as we do. This is a very relevant difference, in practice, as it translates, in our case, into a much higher concentration of the singular values, and, as a consequence, into a much better approximation of the linear transform for a given number of reference kernels.

4 Application to image restoration: transposition and inversion

Many restoration methods apply (iteratively or not) a regularized inversion involving $\mathbf{H}^T$, or $(\mathbf{R} + \mathbf{H}^T\mathbf{H})^{-1}$ (where $\mathbf{R}$ is a regularization matrix), or a combination of both, such as their multiplication (which is a regularized version of $\mathbf{H}^{-1}$). Specifically, the restoration method implemented in this work, requires both the transposition and the inversion.

4.1 A new formalism yielding a fast SV transposed filtering algorithm

In our case, because we have used the PSFs instead of the IKs to build a deformable kernel–with which local inner products are done with the image–we have already obtained an approximation to the associated transposed operator of the original SV filtering. We can easily rewrite Equation 10 in matrix shape, for instance by expressing the convolution as the sequence of a direct Fourier Transform, a masking in the frequency domain (using a diagonal matrix multiplication), and an inverse Fourier transform. The spatial domain masking is also easily expressible by means of a diagonal matrix:

$$\mathbf{H}^T = \sum_{r=1}^{R} \mathbf{D}_{\alpha_r} \mathbf{F} \mathbf{D}_{B_r^*} \mathbf{F}^*. \tag{11}$$

Here $\mathbf{F}^*$ is the Hermitian matrix performing the Fourier transform (thus $\mathbf{F}$ performs the inverse Fourier transform), and B_r^* represents the Fourier transform of a (180° rotated) reference kernel. If we transpose in the previous expression we come back to the original SV filtering operator:

$$\mathbf{H} = \sum_{r=1}^{R} \mathbf{F} \mathbf{D}_{B_r} \mathbf{F}^* \mathbf{D}_{\alpha_r}. \tag{12}$$

We believe that this result is both beautiful and useful: when using a deformable kernel formulation to implement a SV filtering, we (i) convolve the original image with the set of R (J for a practical approximation, with, typically $J \ll R$) reference kernels (180 degrees rotated); (ii) apply a different spatial mask to the output of each convolution; and (iii) add all of them. Whereas for implementing the transpose linear SV operator we do, instead: (i) apply the same set of R (J for a practical approximation) masks as before, directly to the original image, obtaining R (J) masked images; (ii) convolve each with the corresponding reference kernel; and (iii) add all of them. Using this result simplifies a lot the task of transpose-filtering in SV-restoration problems. Note, again, that our starting point for computing the deformable kernel has not been the rows (IKs) but the columns (PSFs) of $\mathbf{H}$. If we were building the deformable kernel from the IKs (as it is usually done in the literature), instead, then the roles of $\mathbf{H}$ and $\mathbf{H}^T$ in Equations 11 and 12 would be exchanged.

4.2 Regularized inversion

In image restoration very often we search for a linear operator aiming to compensate for the observed blur, either as part of an iterative non-linear approach, or as a single-step linear restoration (e.g., Wiener filter). As mentioned in the introduction, due to noise and possible singularity of $\mathbf{H}$, instead of trying to invert it directly, a regularized inversion it is usually preferred, i.e., $\hat{\mathbf{H}}^i = (\mathbf{R} + \mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$, where $\mathbf{R}$ typically being a circulant symmetric definite positive matrix, sometimes expressing the noise power relative to the signal power, in the system. Thus, this regularized inversion can be done in two steps, by first applying the transposed SV filtering associated to $\mathbf{H}^T$, and then applying to the previous output the inverted matrix.

When the IK associated with the matrix to be inverted changes very little within its support, and the amount of regularization in the inversion is enough to avoid high

absolute values in its Jacobian, then it is a good approximation to use the Fourier-reciprocal kernel (the kernel such that, if convolved with the kernel of interest, yields a unity gain Kronecker delta) of each IK to build, stacking by columns, an approximation of the inverted matrix. Being, by assumption, the neighbor IKs shifted versions of each other, in a local approximation, their Fourier-reciprocal kernels will also be approximately shifted versions of each other. This implies that, when multiplying the original matrix times this approximation of the inverse matrix, the corresponding inner products will almost vanish (when taking the local IK and a close approximation to its reciprocal kernel, spatially shifted), except for the reference location (diagonal terms of the resulting matrix), where an exact value of one will be obtained (inner product of the local IK with its reciprocal kernel), as desired. Provided that each IK associated with the matrix to be inverted is invertible in the Fourier domain (i.e., that has no zeros in that domain, because of the regularization term), the reciprocal kernel can be computed in that domain. In practice, we can start from the Fourier transform of the PSFs of **H**, and compute, for each location, the inverse Fourier transform of $(R(\mathbf{f}) + |H_\mathbf{p}(\mathbf{f})|^2)^{-1}$, for every location $\mathbf{p} \in D$, where **f** represents a 2-D frequency component in Fourier. Using these kernels we can obtain, through the SVD technique, an approximation to the LS-optimal deformable inversion kernel with whom to reduce the complexity of the associated SV filtering without compromising the accuracy, in typical cases. It is important to note that, although the calculations of these reciprocal kernels, and their LS-optimal deformable filtering representation may be relatively costly, all that can be done offline, for each known degradation. The resulting SV approximate inverse filtering, though, is done on-line, and that is very fast, requiring only J' convolutions and J' image maskings, J' being the number of reference kernels necessary for an accurate enough approximation of the reciprocal kernels.

4.3 Using the L0-AbS deblurring algorithm

L0-AbS deblurring stands for L0-pseudonorm Analysis-based Sparsity image restoration [22]. It is a recent, efficient, highly non-linear method. L0-AbS technique is based on iteratively imposing to the solution being approximately sparse in a given redundant frame (or a finite set of them), and being, simultaneously, likely in terms of the observation. The first requirement is accomplished by hard-thresholding the image representation in the *x-let* (redundant pyramidal wavelet-like representation) domain, and the second by linear filtering the reconstructed image from the sparse coefficients, and combining the result it with a linear filtered version of the observation. These two operations are alternated

iteratively until convergence (or until a given number of iterations are run).

This deblurring method is applicable whenever the observation suffers from linear blur and contaminating noise is Gaussian, no matter the blur is SI or SV. The set of tools presented here, thus, are applicable to do efficiently the linear SV filtering stages of the referred non-linear method.

The steps of the algorithm are summarized below:

$$\text{Step 0 } \mathbf{x}^{(0)} = \mathbf{y}$$
$$\text{Step 1 } \mathbf{b}^{(n)} = \Phi^T \mathbf{x}^{(n)}$$
$$\text{Step 2 } \mathbf{a}^{(n)} = \Theta_{\text{hard}}(\mathbf{b}^{(n)}, \lambda^{-1/2})$$
$$\text{Step 3 } \mathbf{z}^{(n)} = \Phi \mathbf{a}^{(n)}$$
$$\text{Step 4 } \mathbf{x}^{(n+1)} = (\Phi\Phi^T + \nu \mathbf{H}^*\mathbf{H})^{-1}(\mathbf{z}^{(n)} + \nu \mathbf{H}^*\mathbf{y})$$
$$\text{\textit{back to} Step 1}$$

where **z** is the image reconstructed from a sparse x-let coefficient vector, ν is a positive constant depending on the statistical parameters of the signal model and on the noise variance (see Equation 14 below), and Φ^* is the x-let frame used to impose sparsity. When Φ^* is a tight frame (as in our case, which we used a combined frame made of a Haar redundant wavelet and the Dual-Tree Complex Wavelet Transform [26], as in [23]), then $\Phi\Phi^*$ becomes the identity matrix. Note that for this algorithm we only need to apply $\mathbf{H}^T$ once (to the observation **y**), whereas the inverted matrix is applied once per iteration. Clearly, given the iterative nature of this method it is critical to have a fast SV filtering implementation.

The L0-AbS method is based on a simple image model, which has two statistical parameters: α, controlling the amount of sparsity of the sparse approximation of the image in the representation domain, and σ_r, which is the standard deviation of the sparse approximation from the actual vector of the image in the representation domain. These two parameters set both the threshold T (static, in the implementation used here, as in [22]), used to iteratively *sparsify* the solution candidate in the *x-let* domain, and the ν parameter of Step 4. The threshold applied in Step 2 is defined by the expression:

$$T = \lambda^{-\frac{1}{2}} = 2\alpha^{-\frac{1}{2}}\sigma_r. \tag{13}$$

And the ν parameter is given by:

$$\nu = \sigma_r^2/\sigma^2, \tag{14}$$

being σ^2 the image domain noise variance corrupting the observation [22].

The values of α and σ_r have been obtained by training to optimize the quality of restored images on a set of deconvolution tests (see details in 5.1.2). Because of the dependency of ν on σ^2, calculations of the inverse deformable filters must be done separately for each noise level. Also, note that inverse deformable filtering will require, in general, a larger spatial support for its reference kernels than the direct one (see, e.g., Figure 1, BLUR1 case), due to the generally less smooth Fourier shape of the Fourier-reciprocal kernels.

5 Simulations: results and discussion

We have simulated degradations on standard test images for obtaining some objective measurements of the performance of our deformable SV-filtering techniques, combined with the L0-AbS deblurring approach published in [22]. For that purpose, we have prepared two synthetic SV blurs that are SV versions of previously used degradations in deconvolution tests, with the particularity of that the "average PSF" corresponds, approximately, to the convolution kernel in those tests [22]. This allows the reader to compare, also, for the same noise levels, with the ISNR results for SI degradations (results should be similar using the same restoration method). In addition, we have used a set of central PSFs modelling the Wide Field Camera of the Hubble Telescope in the year 1991, before being fixed, as a realistic (and, at a certain moment, useful) SV restoration problem. Simulated blurred images have been also contaminated by additive white Gaussian noise, in 3 different

levels (low, medium and high). We have compared to two reference methods: The Hybrid Bidiagonalization Regularization (HyBR [6]) and the original sectional method for spatially variant problems [12].

HyBR [6] is a recent method that has proven to be effective for solving large-scale ill-posed inverse problems, such as the restoration of noisy and SV blurry images. As other Lanczos methods, it restricts the solution to lie in a Krylov subspace, so the dimension of the linear systems to solve is much smaller than the initial problem (Equation 5). It applies a hybrid approach based both on a standard regularization technique (like Tikhonov) and on Krylov filtering. Usually, the appropriate choice of the regularization parameter at each iteration in the standard regularization algorithm is a difficult issue in this kind of methods. HyBR is quite robust in this sense, thanks to the weighted generalized cross validation mechanism devised by its authors.

On the other hand, sectional methods apply a partition in the image domain, thus obtaining non-overlapping regions that are processed independently as spatially invariant subimages. These methods have been widely used because of its simplicity, low memory requirements and computational cost, trivial parallelization and good trade-off between speed and quality. Nowadays, more than 40 years after the release of the first sectional method [12], new variants are still being published [8] and applied [27]. The method implemented here for comparison purposes is a FFT-based local Wiener filtering [12].

Figure 1 Examples of estimated PSFs in detector chip 1 of Wide Field Camera (Hubble Spatial Telescope). Left image corresponds to pixel q (0,0) and right image to pixel q (256, 256). Gray-level scale has been jointly modified to highlight the secondary lobes.
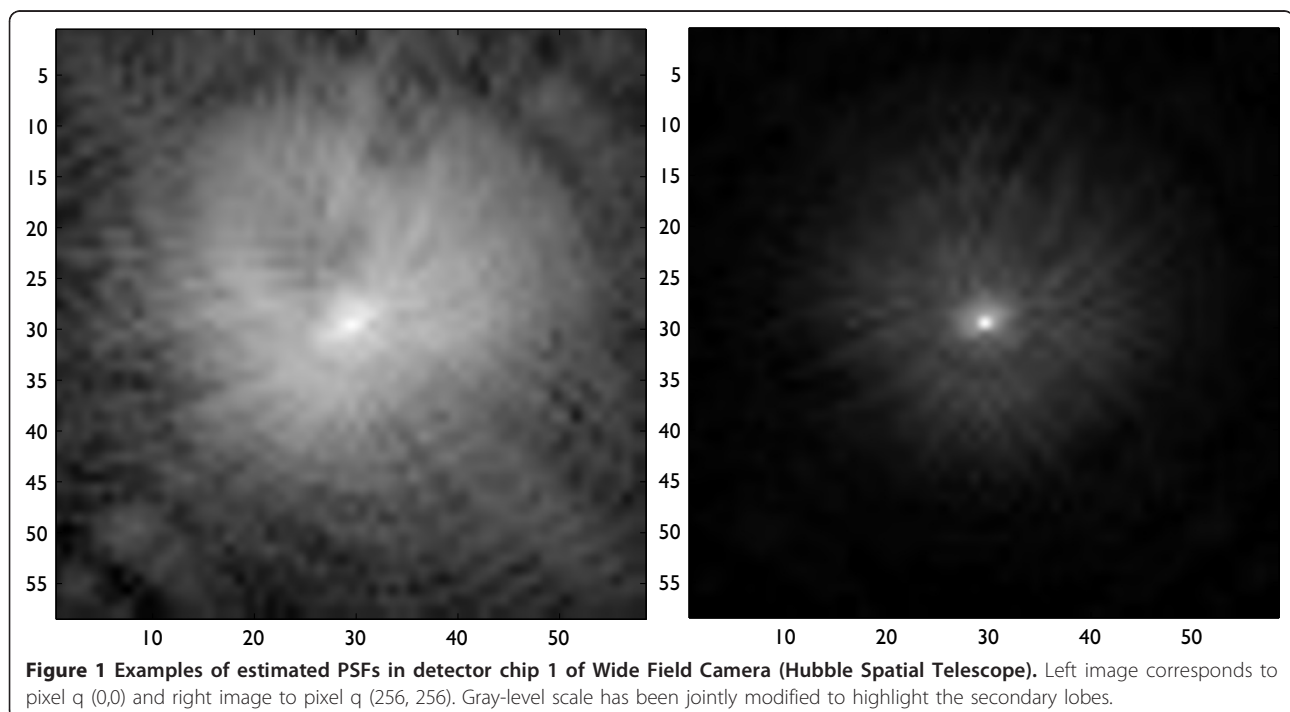
Note that the two methods we compare to here are strictly linear, in contrast with the SV extension we propose here of the L0-AbS method (highly non-linear).

5.1 Experiments

5.1.1 Simulation of degraded images

We have used three SV blurs (BLUR1, BLUR2, and BLUR3), three images, and three noise (AWGN) levels. The design of these experiments aims to give a broad view of the characteristics of the new algorithm, in a way that it can be easily compared with the articles in the literature dealing with variant and invariant systems.

The first synthetic SV filter (BLUR1) is a SV extension of the radial PSF $h(n, m) \propto (1 + n^2 + m^2)^{-1}$. [b] We have re-scaled this function using a factor k_q that changes with the distance of the central pixel q to the center of the image (r_{img}). The new PSF is defined as:

$$\text{PSF}_q(n, m) \propto \frac{1}{1 + (n/k_q)^2 + (m/k_q)^2} \quad k_q = \frac{1}{\sqrt{2}}\sqrt{1 + \frac{r^2_{img}}{(2/3)L^2}}, \quad (15)$$

where L is the image support side (assumed squared), and n, m = -7...7 indices represent the discrete positions of the PSF support. Using these PSFs we obtain $k_{max} = \sqrt{2}$ at the corners of the image, and $k_{min} = \frac{\sqrt{2}}{2}$ located at the center, therefore having a spatial scale factor of 2 between these two extreme cases, in a "fovea-like" manner.

The second synthetic SV filter (BLUR2) is a Gaussian variant filter. Its height grows as a factor of two from the left to right of the image, while its width keeps constant for all image locations. At the image center we have an isotropic blur with $\sigma_n = \sigma_m = 1.6$. The change of the height along the abscissa is exponential (with base 2), ranging the exponent from -1/2 to 1/2, spanning a full octave, as in the previous case. It also uses a PSF support size of 15 × 15 pixels.

The third SV filter (BLUR3) tries to model a more realistic case: the aberrated PSFs of Hubble Space Telescope. Shortly after its deployment on April 24th 1990, a fabrication error in the 2.4 m primary mirror (PM) was discovered. The wide field and planetary camera (WFPC) is the principal instrument of the HST, occupying the central portion of the telescope's focal plane. The HST suffered mainly from spherical aberration in the primary mirror. So, the aberration caused a strongly space-variant PSF. Restoration is further complicated by incomplete knowledge of the PSF and the time variance resulting from changes in focus and spacecraft motion. While the stars are almost ideal point sources, obtaining experimental PSFs is not a simple task. Image cannot be saturated so calibration experiments must be carefully designed, and the dependence on wavelength and the large sensor size requires a huge amount of measures. So, a complete practical study of aberrations in WFPC was dismissed by the Spatial Agency, because the required observation time was not affordable. Nevertheless, PSF modeling programs, such as TinyTIM [28,29], have been very helpful for characterization and restoration of the images of all the instruments of the Hubble, in its different configurations for filters, wavelengths and even variations over time. Despite their limitations and simplifications assumed, these programs give us a good starting point. TinyTIM C source code is freely distributed, so it can be compiled and executed on any machine.

The first WFPC consisted of two separate cameras, each comprising four 800 × 800 pixel Texas Instruments CCDs arranged to cover a contiguous field of view. We have modified TinyTIM C source code to fix some minor bugs and model a section of 256 × 256 pixels (the size of standard test images) and its associated PSFs in detector chip 1 of Wide Field Camera. The configuration was set as the one in 24th April 1991 (one year after the deployment) for *f555w* filter, object spectrum K4V and a null secondary despace (0.0 microns). The non-subsampled PSF size was 6.0 arc seconds, which leads to a discrete support of 58 × 58 pixels for each PSF. This simulation took one week on three computers working in parallel and generated 1.8 GB of data. Because of this heavy computational task, it could be considered to save all the possible PSFs to use them in a faster way, but the complete dataset for WFCP is estimated to require storage around 8 TB [30], and it should be recomputed each time the telescope is refocused. Hubble WFPC imaging seems to be a classic ill posed restoration problem, but it is a daunting challenge because the extension, variability in time and space, and lack of accuracy in PSF definitions. Here we only intend to use estimated PSFs as a means to compare different algorithms. The restoration of real images involves many more factors (e.g., Poissonian instead of Gaussian noise) that go beyond the scope of this article. Figure 2 shows the estimated PSFs for two different image support locations.

We have used three classic 256×256 test images: Cameraman, House and Satellite. Cameraman and House are 256 × 256 8-bit gray-level standard test images. Satellite image was synthesized at the US Air Force Phillips Laboratory, Lasers and Imaging Directorate, Kirtland Air Force Base, New Mexico. It belongs to the Restore Tools toolbox data set [31].

In addition, three different amounts of white Gaussian noise have been added to the blurred images: low level ($\sigma^2 = \sigma_L^2 = 0.308$), medium ($\sigma_M^2 = 2$), and high ($\sigma_H^2 = 8$).

Figure 2 Dependence of σr2 image model parameter with noise level.
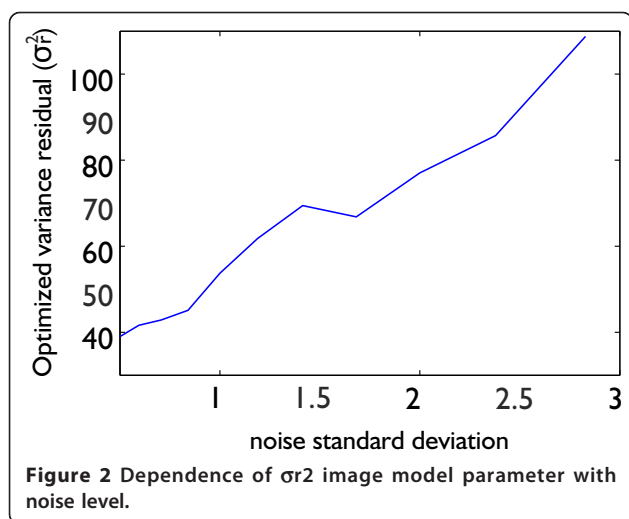
As a first naïve approximation to boundary handling, all original images have been extended by mirroring its outer edges, prior to their SV-filtering. Of course, this is not a legitimate procedure to apply to real observations, because in these simulations filtering and noise addition have been applied also to the extended original image. At the end of the restoration, the previously added extra pixels are removed from the restored image, in order to obtain the same resolution as in the original. In addition, at each iteration of L0-AbS, each convolution is done using mirror extension boundary conditions (note that, contrary to the extension of the original image, this intermediate mirror extension is legitimate). Rigorous treatment of boundary conditions is left as future study.

5.1.2 Image representation and training of statistical model parameters

In this study, we have used two frames for the non linear method: a translation invariant version of the Haar pyramid (TIHP, quasi-Parseval), as described in [32], and the Dual-Tree Complex Wavelet Transform [26] which is a Parseval Frame, both with four scales.

As introduced in Section 4.3, the L0-AbS method leans on an image model using two statistical parameters: α and σ_r. In this article, these parameters are estimated by optimizing the algorithm's performance for a set of training images and degradations. We have chosen eight gray-scale images from USC-SIPI Image Database that represent natural scenes: Couple, Man, Bridge, Elaine, MoonSurface, ChemicalPlant, Pentagon and Aerial5.2.09; and two spatially variant blurring kernels: BLUR1 and BLUR2 (PSF support size is smaller than BLUR3 and faster to compute in spatial domain). Some of the reference images have a higher resolution than our test images, so in the training set we only have used the central portion of 256 × 256 pixels, without any

scaling. As figure of merit to be maximized we have used the average ISNR (increment in Signal-to-Noise Ratio), in decibels, of all the considered deblurring tests (two blurs, eight images), independently for each of the three noise levels considered. Whereas it may be argued that this is not an ideal quality measurement (for several reasons), it is nevertheless quite simple, and it seems to behave robustly enough for this purpose. We believe that in this case the overlapping of BLUR1 and BLUR2 in the training and test stages, though not being ideal, it is a justified decision in the context of achieving a reasonable tradeoff among computational cost, quality and generality. Note that the test images are different from the training images, and that the test blurs include a case not used in the training set (BLUR3).

The optimization process has been computed independently for a range of noise levels from $\sigma^2 = 0.25$ to $\sigma^2 = 8$ in half octave intervals, as it is shown in Figure 3 for σ_r^2 parameter. We have obtained the optimized values $\alpha = 4.52$ and $\sigma_r = 40.56$ for low noise level, $\alpha = 8.31$ and $\sigma_r = 69.45$ for medium noise level, and $\alpha = 15.07$ and $\sigma_r = 108.79$ for high noise level.

The optimized image model parameter σ_r^2 exhibits an unexpected behavior, which has been consistently observed in further tests: it is approximately proportional to σ (see Figure 3). Dependency of the optimized α on the noise variance, on the contrary, was much less pronounced. Making the image model parameters to depend on the degradation parameters does not fit a pure Bayesian estimation framework, for which the prior knowledge and the observation conditions are intrinsically independent. Within the empirical Bayes approach, however, it is normal to adjust the prior features (in our case, the image model parameters) depending on the amount and type of degradation in the observation.[c]

5.1.3 Algorithm parameters spatially variant L0-AbS

There are two main parameters in the algorithm: the number of iterations N_{iter} in the spatially variant L0-AbS algorithm and the number J of reference kernels for the deformable filtering, both for the original PSFs and for the regularize inverse filtering (we have used the same value $J' = J$). In addition, one must choose a certain spatial support for the deformable kernels and their regularized inverse quadratic version ("the inverse kernels", in short). In the first case the obvious choice is picking the maximal effective spatial support among all the PSFs. However, for the inverse kernel the effective support will depend on the amount of regularization: the effective size will decrease as the regularization increases, and vice versa, because regularization increases the smoothness of the Fourier spectrum. As shown in Figure 1, we have chosen a square support of

Figure 3 SNR between the array of staked PSFs centered at zero and its approximation. Direct (top) and inverse regularized (bottom) cases are shown.
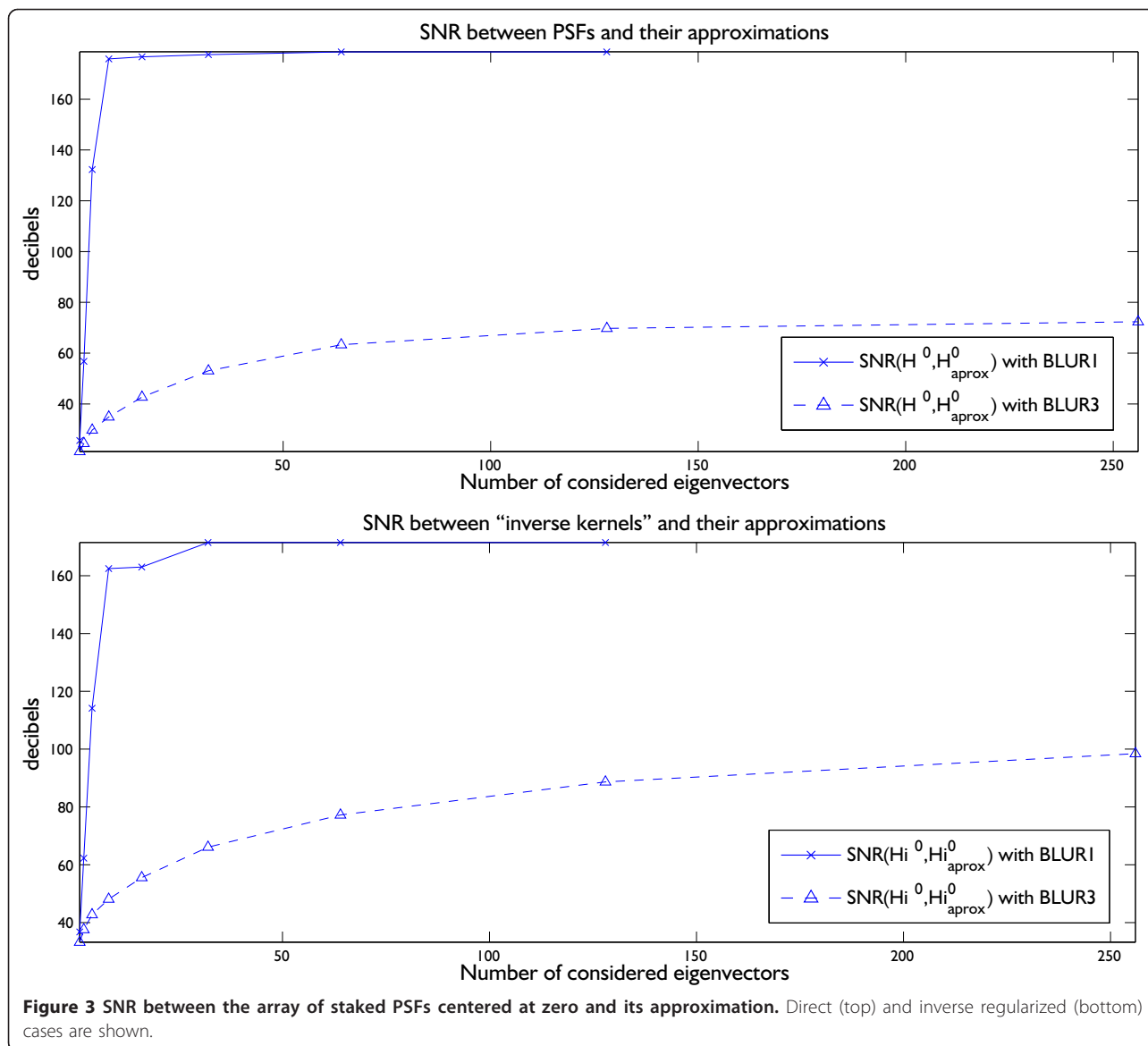
15×15 for the PSFs and of 29×29 for the inverse kernels.

We have observed that, similarly to what happened in the SI case, the L0-AbS method converges to a point near the MSE-optimum. It gets very close after a few steps, and then it moves slightly away but remains in the immediate vicinity. We have found a good compromise between speed and quality in the studied cases by taking $N_{iter} = 20$.

As mentioned above, in many real cases there is a large amount of overlapping among (centered) PSFs, both in the direct and inverse regularized SV filtering. Therefore, the number J of regularized and inverted deformable kernels can be set quite small to reduce the number of convolutions without losing much accuracy.

The number of deformable kernels considered in the study cases is only $J = 4$, although it could be particularly tuned for each blur. This aspect is further studied in the Section 5.2.

As far as we have observed, the number of iterations has stronger influence than the number of deformable kernels on the final quality, provided that the number of deformable kernels is not too low. It is immediate to check that the complexity of the spatially variant L0-AbS method grows linearly with each of these parameters.

5.1.4 Algorithm parameters in HyBR and sectional methods

The examples of RestoreTools Toolbox [31] are limited to a small subset of PSFs, so that degradation is modeled as a set of locally spatially invariant regions. It

resembles to sectional methods although HyBR algorithm is more flexible. When the number of PSFs increases memory needs grow dramatically, and the original code cannot be run. Therefore, we have slightly modified the HyBR code in order to support real spatially variant PSFs in an efficient way, even with an arbitrary number of PSFs. In our case, a different PSF is associated to each pixel in the images. No pre-conditioning has been applied.

For the sectional method [12] we have defined 8×8 = 64 uniform regions, so each local invariant region is only 32 × 32 pixels. For each region it applies a classical FFT-based Wiener filtering. Obviously, this method should give better results than the application of a global Wiener filtering on the complete image, because it is approximately adapted to the filtering at each region. Only the central PSF of each region is considered in the restoration. Its computational performance grows superlinearly with the number of processed regions in multicore implementations thanks to the effect of memory hierarchy in the small test images we used.

5.2 Two study cases

In this section, we study in some depth how the described deformable filtering techniques have been applied in two SV filtering cases: BLUR1 (synthetic scaled radial PSFs) and BLUR3 (Hubble PSFs).

The PSF field in BLUR1 is quite smooth, according to Equation 15. So, as expected, it shows a significant amount of overlapping among centered PSFs, and, therefore, a small effective dimensionality of its matrix of stacked, centered at zero, PSFs. From Figures 1 and 4a, we can see that $J = 4$ is more than enough to obtain a high accuracy in all the SV-filtering stages.

The eigenvalues of H_0 are represented in the top left of Figure 1. They fall steeply more than 12 orders of magnitude. Most of the energy is concentrated in a few singular vectors. Figure 4 shows the SNR between the matrix of stacked PSFs centered at zero and its approximation with a number of eigenvectors. Such behavior is shared by the inverse kernel case (top right).

The first column in the lowest part of Figure 1 shows the reference kernels, and the second column their interpolation functions along the image. These interpolation functions shows the local weight of each reference kernel in the image (in the figure without taking into account the eigenvalue weight, for visualization sake).

It is easy to appreciate that the main reference kernel has a shape similar to the original radial PSF (like an average of all of them), whose influence is more important in the center of the image. The second one reminds a first derivative of this shape, it may be positive or negative weighted to approximate the extension of each PSF. The remaining ones help to define the subtle details to capture the variability of the original PSFs [33]. Their relative influences -as seen in the interpolation functions- show a more complex radial character.

Right columns, in Figure 1, have the four first inverse regularized deformable kernels and their interpolation functions. Although the nature of these PSFs is different, a similar pattern is observed.

A similar analysis is presented in Figure 5 for BLUR3. The PSFs correspond to a region of 256 × 256 pixels in chip0 of the WFPC of Hubble Space Telescope in the configuration of 24th April 1991. The degradation is quite variant. Most of the energy is located in only a few elements again (top left), but it does not fall so sharply as in the previous case. Because of this, we need more deformable kernels to approximate the original PSFs accurately (Figure 4). Four main direct and inverse deformable kernels are represented in columns 1 and 3 with linear contrast, so secondary lobes are not highlighted. They are nonsymmetric and rather different from the original ones (Figure 2). The influence of these kernels take different directions as it is illustrated by the interpolation functions.

In the following section, the number of deformable kernels has been cut to $J = 4$ to show the robustness of the algorithm in a set of representative scenarios, even in cases where very few filters are applied.

5.3 Results

Figures 6, 7, 8, and 9 show some comparative visual examples of our method's performance, for different blurs and noise levels, for the three considered images. The improvement of visual quality with respect to the observation and also with respect to the competitors is outstanding. Probably, the most relevant difference with the compared methods is the much more powerful noise removal, whereas edges and other salient features in the images are, in general, substantially better re-generated. This is clearly reflecting the much higher performance of non-linear methods compared to classical linear approaches.

Tables 1, 2, and 3 show average (and standard deviation, for 16 different noise realizations) Increment in Signal-to-Noise Ratio, in decibels, for the three different implemented SV blurs (respectively), the three images and the three noise levels. Results are compared to the two methods used as reference here. Overall, the improvement is very strong. As simulated observations where extended beyond the original images boundaries, we have applied the same post-cutting of the valid area after the estimation for all three compared methods.

Finally, we have measured the computation times for our method (its MATLAB implementation) as compared to the competitors (see Table 4) in a server with 2 Intel
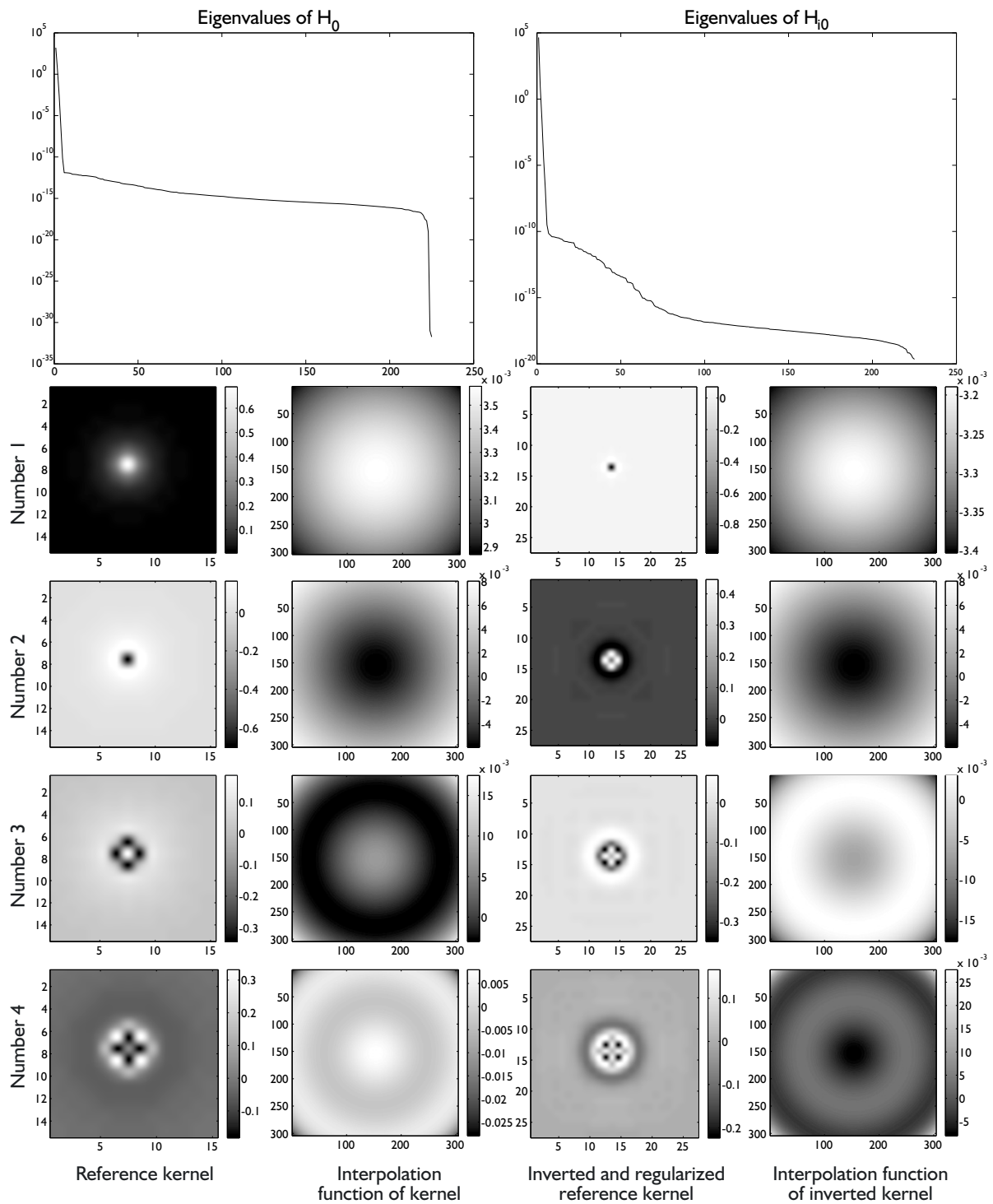
Figure 4 BLUR1: Eigenvalues of the array of stacked PSFs centered at zero are shown in the top, for both direct (left) and inverse regularized (right) cases. Main (direct) deformable kernels are represented in the first column, their interpolation functions are in the second column, inverse regularized deformable kernels are in the third column and their interpolation functions are in the last column.
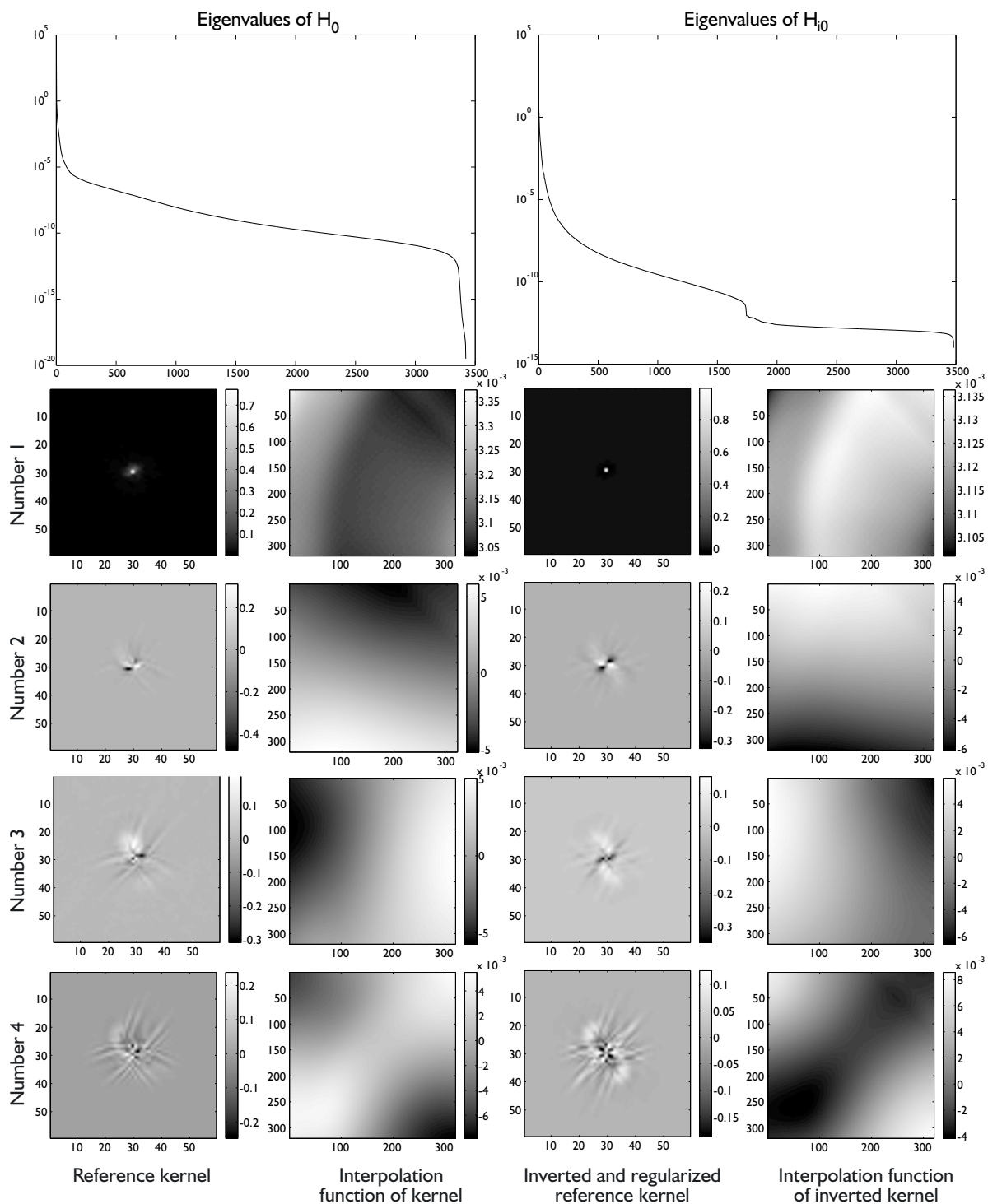
Figure 5 BLUR3: Eigenvalues of the array of stacked PSFs centered at zero are shown in the top, for both direct (left) and inverse regularized (right) cases. Main (direct) deformable kernels are represented in the first column, their interpolation functions are in the second column, inverse regularized deformable kernels are in the third column and their interpolation functions are in the last column.

Figure 6 Top left: Original Cameraman test image. Top right: Degraded image with BLUR2 and medium amount of additive gaussian noise $\left(\sigma_M^2 = 2\right)$. Middle left: restored image with sectional method. Middle right: restored image with HyBR algorithm. Bottom: restored image with spatially variant L0-AbS.
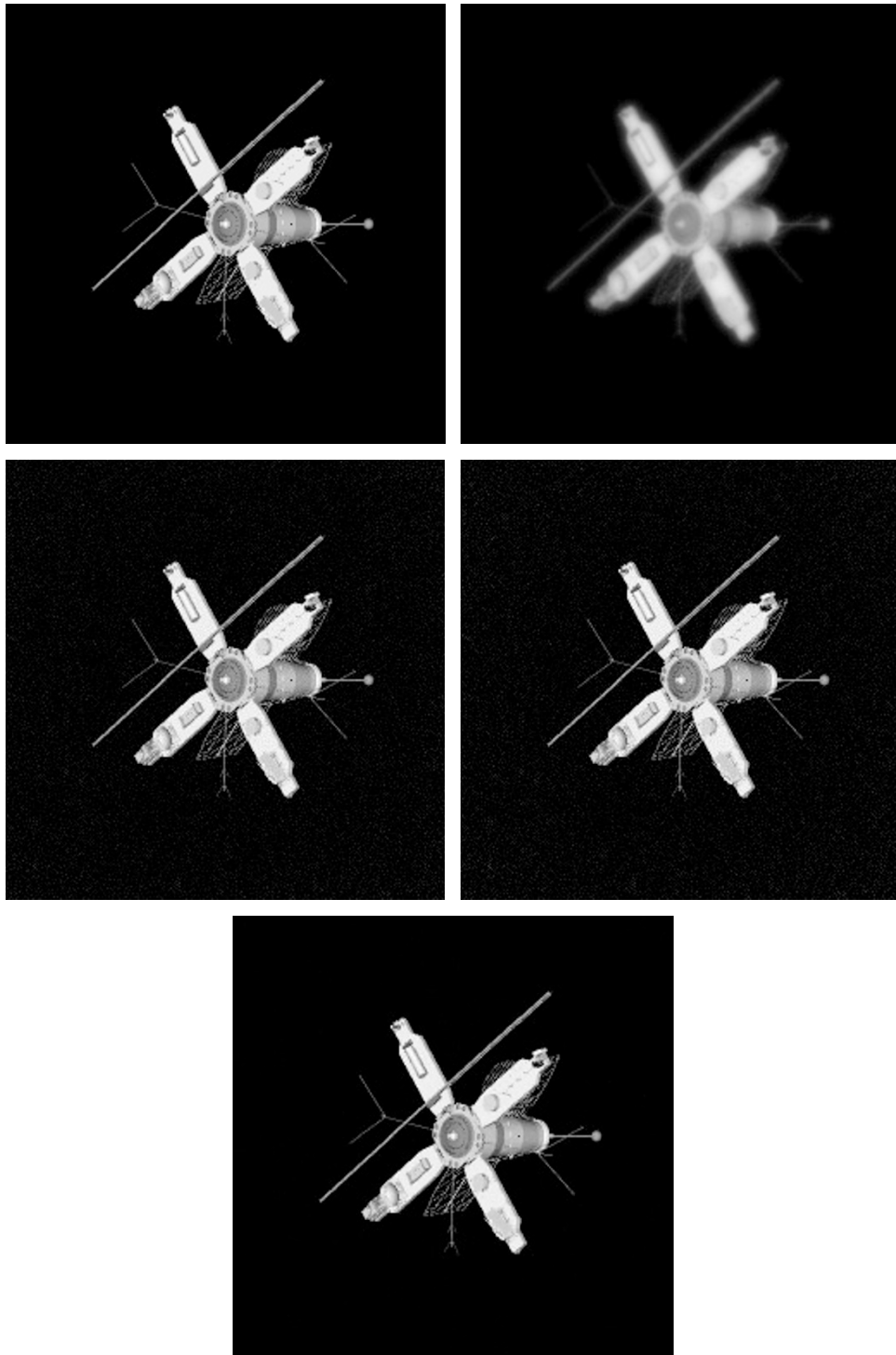
Figure 7 Top left: Original Satellite test image. Top right: Degraded image with BLUR1 and small amount of additive gaussian noise $\left(\sigma_M^2 = 0.308 \right)$. Middle left: restored image with sectional method. Middle right: restored image with HyBR algorithm. Bottom: restored image with spatially variant L0-AbS.
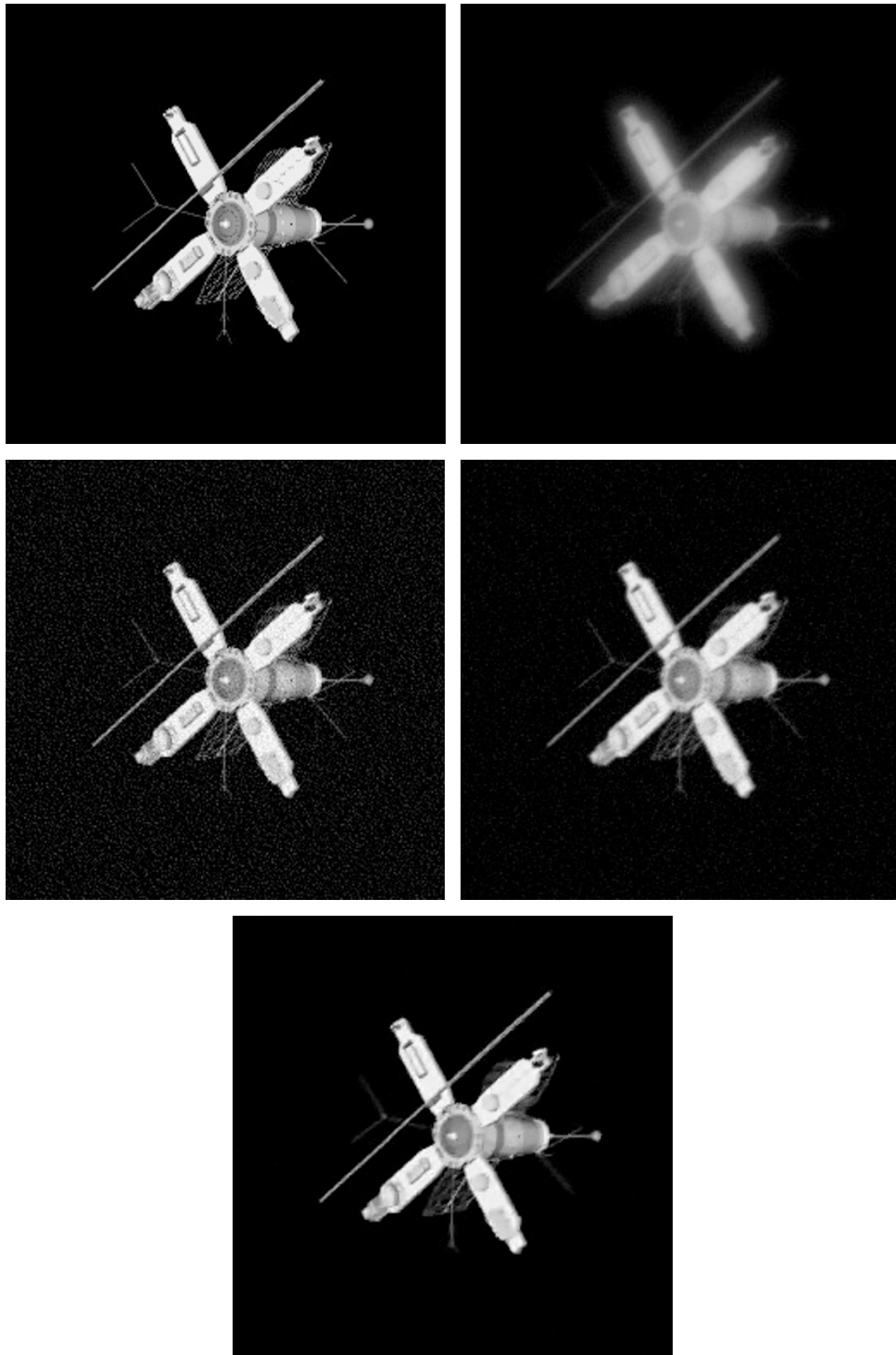
Figure 8 Top left: Original Satellite test image. Top right: Degraded image with BLUR3 and a medium amount of additive gaussian noise $\left(\sigma_M^2 = 2\right)$. Middle left: restored image with sectional method. Middle right: restored image with HyBR algorithm. Bottom: restored image with spatially variant L0-AbS.
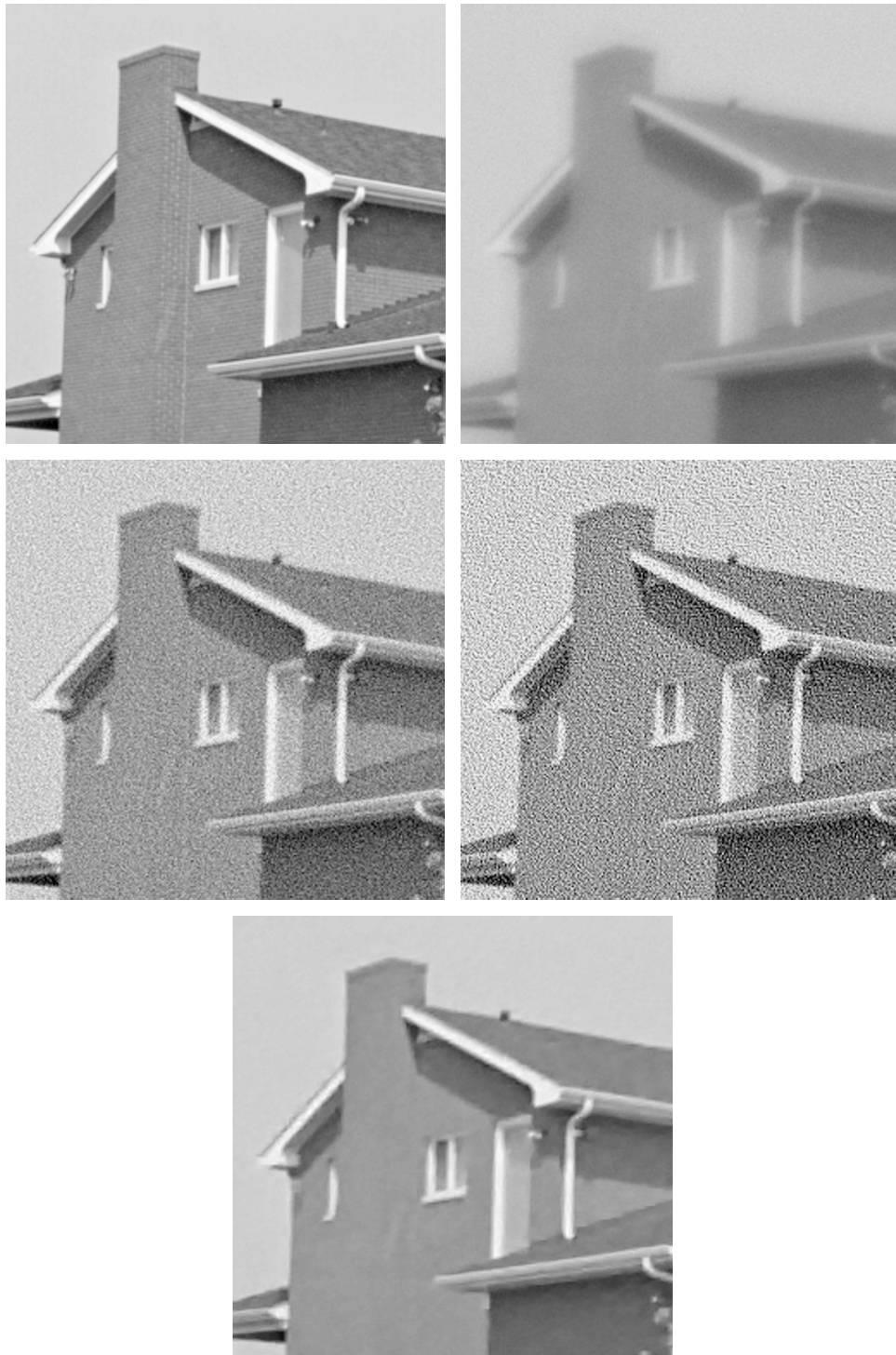
Figure 9 Top left: Original House test image. Top right: Degraded image with BLUR3 and a large amount of additive gaussian noise $\left(\sigma_M^2 = 8\right)$. Middle left: Restored image with sectional method. Middle right: Restored image with HyBR algorithm. Bottom: Restored image with spatially variant L0-AbS.

Table 1 Averaged ISNR of restored images with several noise levels with BLUR1

Image	Method	BLUR1		
		$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$
Cameraman	HyBR	6.39 ± 0.03	-1.51 ± 0.04	0.49 ± 0.06
	Trussell	6.73 ± 0.03	0.71 ± 0.02	-1.45 ± 0.03
	var. L0AbS	12.76 ± 0.04	8.90 ± 0.02	6.49 ± 0.05
House	HyBR	2.33 ± 0.03	-5.65 ± 0.04	-3.76 ± 0.08
	Trussell	2.82 ± 0.03	-2.84 ± 0.03	-4.30 ± 0.03
	var. L0AbS	11.13 ± 0.03	8.43 ± 0.03	6.69 ± 0.05
Satellite	HyBR	6.99 ± 0.03	5.81 ± 0.01	3.74 ± 0.02
	Trussell	7.61 ± 0.04	2.17 ± 0.02	0.65 ± 0.03
	var. L0AbS	15.79 ± 0.06	9.73 ± 0.09	6.73 ± 0.05

Table 3 Averaged ISNR of restored images with several noise levels with BLUR3

Image	Method	BLUR3		
		$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$
Cameraman	HyBR	8.66 ± 0.03	0.77 ± 0.02	-4.91 ± 0.02
	Trussell	8.59 ± 0.03	2.83 ± 0.03	0.86 ± 0.02
	var. L0AbS	13.95 ± 0.04	9.13 ± 0.04	6.30 ± 0.03
House	HyBR	5.44 ± 0.04	-2.61 ± 0.03	-8.10 ± 0.01
	Trussell	5.68 ± 0.03	0.05 ± 0.03	-1.22 ± 0.02
	var. L0AbS	13.02 ± 0.03	10.62 ± 0.05	8.68 ± 0.04
Satellite	HyBR	9.73 ± 0.03	7.06 ± 0.04	4.83 ± 0.02
	Trussell	10.27 ± 0.02	5.26 ± 0.03	4.05 ± 0.02
	var. L0AbS	17.96 ± 0.06	10.43 ± 0.06	7.73 ± 0.04

Quad core Xeon E5504 CPUs and 16 GB of main memory. Although the sectional method is a bit faster, overall times of the three methods are in the same range. We consider this a notable achievement, as our method is non-linear and iterative, so conceptually much more complex (which clearly translates into a superior MSE-performance, as we have seen) than the compared linear ones. Execution times and main memory requirements are notably higher in BLUR3 than in the other degradation cases in our method. This is due to the fact that the expression $\mathbf{H}^T\mathbf{y}$ in step 4 must be explicitly calculated as a sparse matrix-vector multiplication (not like a deformable filtering), because Hubble PSFs are not so spectrally concentrated as BLUR1 and BLUR2 (as it is shown in Figures 4b and 5).

6 Conclusions and future study

Many real imaging processes involve SV linear degradation. Thus, a single PSF is not enough to characterize the blur in the observation model, and a local PSF for each spatial location must be considered. In this study, a least squares optimal deformable filtering approximation is introduced as an efficient tool for linear SV filtering, in the context of restoring SV-degraded images. Based on this technique a new formalism for linear SV operators has been proposed, that highlights difference between IKs and PSFs. This formalism helped us to formulate an efficient way to implement the transposed SV-filtering based uniquely on the PSFs. We also have proposed a method for implementing an approximation of the filtering of a SV-matrix regularized inverted, under the assumption of having smoothly varying kernels, and enough regularization. We applied these techniques to implement a SV-version of a recent successful sparsity-based image deconvolution method. A high performance (high speed, high visual quality and low mean squared error, MSE) is demonstrated through several simulation experiments (one of them based on the Hubble telescope PSFs), by comparison to two state-of-the-art methods. As a initial test, synthetic star cluster images have been processed with the same set of obtained parameters for natural images, in order to check optimized parameters' robustness in a case with a very different image statistics. Results were comparable to the ones obtained with other methods. We have also adapted the underlying statistical model by proper training with typical real astronomical images (from Kitt Peak observatory) with different blurs and noise levels; as we expected in this case, results are clearly better than the previous ones (in a range between 1 and 3 dBs). So, it is recommended to choose properly statistical parameter values to achieve a very high-performance in each application (astronomical data, natural images in photography, micrography, medical images, etc.).

Whereas we have used the SVD as a key tool for the proposed SV-filtering techniques, the fact of using it over the set of centered PSFs, instead of on the PSFs at their original locations, as previous studies have done, makes a huge difference in efficiency and accuracy terms. By means of this extra degree of efficiency, for a given accuracy, we have been able to include these SV-filtering techniques as part of an iterative non-linear deblurring

Table 2 Averaged ISNR of restored images with several noise levels with BLUR2

Image	Method	BLUR2		
		$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$
Cameraman	HyBR	3.07 ± 0.00	2.30 ± 0.00	1.72 ± 0.01
	Trussell	0.05 ± 0.03	0.69 ± 0.03	0.83 ± 0.02
	var. L0AbS	4.21 ± 0.01	3.27 ± 0.02	2.63 ± 0.02
House	HyBR	4.33 ± 0.01	3.46 ± 0.01	2.79 ± 0.01
	Trussell	-0.48 ± 0.04	-0.26 ± 0.05	-0.09 ± 0.05
	var. L0AbS	5.14 ± 0.03	4.18 ± 0.02	3.68 ± 0.02
Satellite	HyBR	2.77 ± 0.00	1.97 ± 0.00	1.43 ± 0.01
	Trussell	2.67 ± 0.02	2.30 ± 0.02	2.02 ± 0.03
	var. L0AbS	4.51 ± 0.02	3.83 ± 0.02	3.35 ± 0.02

Table 4 Averaged execution times (in seconds) for the applied algorithms in different images and noise levels.

Image	Method	BLUR1			BLUR2			BLUR3		
		$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$	$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$	$\sigma_L^2 = 0.308$	$\sigma_M^2 = 2$	$\sigma_H^2 = 8$
	HyBR	19.48	24.96	18.90	4.59	3.03	2.27	127.35	143.13	159.90
Cameraman	Trussell	1.00	0.80	0.68	0.88	0.89	0.88	1.02	1.13	0.89
	var. L0AbS	3.54	3.53	3.64	3.50	3.49	3.51	10.64	10.47	10.50
	HyBR	16.99	21.47	19.27	3.48	2.47	1.79	120.34	139.24	158.00
House	Trussell	1.02	0.87	0.83	0.75	0.89	0.96	0.99	0.97	1.12
	var. L0AbS	3.91	3.90	3.91	3.51	3.52	3.50	10.70	10.62	10.65
	HyBR	17.82	11.51	3.76	3.38	2.38	1.86	130.84	148.19	51.04
Satellite	Trussell	0.89	0.71	0.91	0.81	0.88	0.95	1.12	0.97	1.06
	var. L0AbS	3.91	3.90	3.92	3.51	3.51	3.53	10.65	10.58	10.60

All the implementations have been coded and executed in MATLAB

method which is much more powerful than classical linear approaches, in a similar computation time.

As future study, it is still pending to explore efficient ways to deal with image boundaries. Finally, in this first part of the article we only deal with simulations, because of the complexity to process real images when the PSF field is unknown. The second part of this article provides an in-depth study with real astronomical data.

Endnotes

[a]Other SV-restoration techniques could benefit as well of the proposed techniques, e.g., adapting a classical Wiener restoration to a SV-blurred noisy degradation. [b]We have forced that all PSFs integrate to one. This is the reason of using "∞" instead of " = " in their definitions. [c]An in-depth analysis of this issue lies beyond the scope of this article. However, a partial explanation of the observed behavior is that the image model parameters are also estimated from the observation, and, as such, they are subject to error. The bias-variance trade-off in these estimates, empirically optimized in terms of the final image estimation performance, may shift a lot, depending on the observation conditions, even when the uncorrupted original remains the same.

Author details

[1]E.T.S. Ingenieria Informatica, Universidad Rey Juan Carlos, Madrid, Spain [2]Instituto de Optica, Consejo Superior de Investigaciones Científicas, Madrid, Spain

References

1. G Golub, W Kahan, Calculating the singular values and pseudo-inverse of a matrix. Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis. **2**, 205–224 (1965). doi:10.1137/0702016
2. H Andrews, C Patterson, Singular value decompositions and digital image processing. IEEE Transactions on Acoustics, Speech and Signal Processing. **24**(1), 26–53 (Feb 1976). doi:10.1109/TASSP.1976.1162766
3. S Ben Hadj, L Blanc-Fraud, Restoration mehod for spatially variant blurred images. Research Report 7654, INRIA (June 2011)
4. AF Boden, DC Redding, RJ Hanisch, J Mo, Massively parallel spatially variant maximum-likelihood restoration of hubble space telescope imagery. J Opt Soc Am A. **13**(7), 1537–1545 (1996). doi:10.1364/JOSAA.13.001537
5. D Calvetti, B Lewis, L Reichel, Restoration of images with spatially variant blur by the gmres method, in *Advanced Signal Processing Algorithms, Architectures, and Implementations X, ed. FT Luk, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE)*, San Diego, CA, USA, **4116**, 364–374 (November 2000)
6. J Chung, JG Nagy, DP OLeary, A weighted-gcv method for lanczos-hybrid regularization. Electronic Transactions on Numerical Analysis. **28**, 149–167 (2008)
7. TP Costello, WB Mikhael, Efficient restoration of space-variant blurs from physical optics by sectioning with modified wiener filtering. Digital Signal Processing. **13**(1), 1–22 (January 2003). doi:10.1016/S1051-2004(02)00004-0
8. L Denis, E Thiebaut, F Soulez, Fast model of space-variant blurring and its application to deconvolution in astronomy. in *2011 18th IEEE International Conference on Image Processing (ICIP)*. Brussels 2817–2820 (Sep 2011)
9. DA Fish, J Grochmalicki, ER Pike, Scanning singular-value-decomposition method for restoration of images with space-variant blur. J Opt Soc Am A. **13**(3), 464–469 (1996). doi:10.1364/JOSAA.13.000464
10. G Nagy James, P O'Leary Dianne, Restoring images degraded by spatially-variant blur. SIAM J Sci Comput. **19**, 1063–1082 (1998). doi:10.1137/S106482759528507X
11. N Hajlaoui, C Chaux, G Perrin, F Falzon, A Benazza-Benyahia, Satellite image restoration in the context of a spatially varying point spread function. J Opt Soc Am A. **27**(6), 1473–1481 (2010). doi:10.1364/JOSAA.27.001473
12. H Trussell, B Hunt, Image restoration of space-variant blurs by sectioned methods. IEEE Transactions on Image Processing. **26**(6), 608–609 (December 1978)

13. G Nagy James, P O'Leary Dianne, Fast iterative image restoration with a space-varying psf. The International Society for Optical Engineering, editor, Advanced Signal Processing Algorithms, Architectures, and Implementations IV. Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE). Bellingham. **3162**, 388–399 (1997)

14. Gene Golub, Charles Van Loan, Total least squares, in *Smoothing Techniques for Curve Estimation, of Lecture Notes in Mathematics*, vol. 757, ed. by Th Gasser, M Rosenblatt (Springer Berlin/Heidelberg, 1979), pp. 69–76. doi:10.1007/BFb0098490

15. P Perona, Deformable kernels for early vision. IEEE Transactions on Pattern Analysis and Machine Intelligence. **17**(5), 488–499 (May 1995). doi:10.1109/34.391394

16. A Tabernero, J Portilla, R Navarro, Duality of log-polar image representations in the space and spatial-frequency domains. IEEE Transactions on Signal Processing. **47**(9), 2469–2479 (1999). doi:10.1109/78.782190

17. R Lupton, JE Gunn, Z Ivezic, GR Knapp, S Kent, The sdss imaging pipelines, in *Astronomical data analysis software and systems X: proceedings of a meeting held at Boston, Massachusetts, USA, 12-15 November 2000*, Astronomical Society of the pacific, 2001, **238**, 269–279 (12–15 November 2000)

18. M Jarvis, B Jain, Principal component analysis of psf variation in weak lensing surveys. Arxiv preprint astroph/0412234 (2004)

19. Tim Schrabback, Jan Hartlap, Benjamin Joachimi, Martin Kilbinger, Patrick Simon, Karim Benabed, Marusa Bradac, Tim Eifler, Thomas Erben, D Fassnacht Christopher, F William High, Stefan Hilbert, Hendrik Hildebrandt, Henk Hoekstra, Konrad Kuijken, Phil Marshall, Yannick Mellier, Eric Morganson, Peter Schneider, Elisabetta Semboloni, Ludovic Van Waerbeke, Malin Velander, Evidence of the accelerated expansion of the universe from weak lensing tomography with COSMOS. Astronomy & Astrophysics. **516**, 1–26 (2010)

20. MJ Jee, JP Blakeslee, M Sirianni, AR Martel, RL White, HC Ford, Principal component analysis of the time-and position-dependent point-spread function of the advanced camera for surveys. Publications of the Astronomical Society of the Pacific. **119**(862), 1403–1419 (2007). doi:10.1086/524849

21. Per Christian Hansen, G Nagy James, P O'Leary Dianne, Deblurring Images: Matrices, Spectra, and Filtering (Fundamentals of Algorithms 3) (Fundamentals of Algorithms), (Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2006)

22. J Portilla, Image restoration through L0 analysis-based sparse optimization in tight frames. in *16th IEEE International Conference on Image Processing (ICIP)* 3909–3912 (Nov 2009)

23. J Portilla, E Gil-Rodrigo, D Miraut, R Suarez-Mesa, CONDY: Ultra-fast high performance restoration using multi-frame L2-relaxed-L0 sparsity and constrained dynamic heuristics. in *18th IEEE International Conference on Image Processing (ICIP)*. Brussels 1837–1840 (Sep 2011)

24. E Gil-Rodrigo, J Portilla, D Miraut, R Suarez-Mesa, Efficient joint poisson-gauss restoration using multi-frame L2-relaxed-L0 analysis-based sparsity. in *18th IEEE International Conference on Image Processing (ICIP)*. Brussels 1385–1388 (Sep 2011)

25. JW Goodman, *Introduction to Fourier optics*, (Roberts & Company Publishers, 2005)

26. N Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals. Applied and Computational Harmonic Analysis. **10**(3), 234–253 (2001). doi:10.1006/acha.2000.0343

27. M Rerabek, P Páta, Advanced processing of astronomical images obtained from optical systems with ultra wide field of view. in *Proceedings of SPIE*. **7443**, 74431Z (2009)

28. J Krist, Tiny Tim: an HST PSF simulator, in *Astronomical Data Analysis Software and Systems II*, vol. 52, ed. by RJ Hanisch, RJV Brissenden, Barnes Jeannette (ADASS, 1993), pp. 536–539

29. J Krist, H Hasan, Deconvolution of HST WFPC images using simulated PSFs. in *Astronomical Data Analysis Software and Systems II*. **52**, 530–532 (1993)

30. RJ Hanisch, RL White, Hst image restoration: current capabilities and future prospects [invited]. in *Very High Angular Resolution Imaging*. **158**, 61 (1994)

31. JG Nagy, K Palmer, L Perrone, Iterative methods for image deblurring: a matlab object-oriented approach. Numerical Algorithms. **36**(1), 73–93 (2004)

32. JA Guerrero-Colón, L Mancera, J Portilla, Image restoration using space-variant gaussian scale mixtures in over-complete pyramids. Image Processing, IEEE Transactions on. **17**(1), 27–41 (2008)

33. R Lauer Tod, Deconvolution with a spatially-variant PSF. Arxiv preprint astro-ph/0208247 (2002)