Astronomy & Astrophysics manuscript no. MulPsfVar November 17, 2003
(DOI: will be inserted by hand later)

Multiple-image composition and deblurring with spatially-variant PSFs

R. Vio[1], J. Nagy[2], and W. Wamsteker[3]

[1] Chip Computers Consulting s.r.l., Viale Don L. Sturzo 82, S.Liberale di Marcon, 30020 Venice, Italy
ESA-VILSPA, Apartado 50727, 28080 Madrid, Spain
e-mail: robertovio@tin.it
[2] Department of Mathematics and Computer Science, Emory University, Atlanta, GA 30322, USA.
e-mail: nagy@mathcs.emory.edu
[3] ESA-VILSPA, Apartado 50727, 28080 Madrid, Spain
e-mail: willem.wamsteker@esa.int

Received .....; accepted .....

Abstract. In this paper we generalize a reliable and efficient algorithm, developed in the context of least-square (LS) methods, to estimate the image corresponding to a given object when a set of observed images are available with different and spatially invariant PSFs, to deal with the case of spatially variant PSFs. Noise is assumed additive and Gaussian. The proposed algorithm allows the use of the classical single-image deblurring techniques for the simultaneous deblurring of the observed images, with obvious advantages both in computational cost and ease of implementation. Its performance and limitations are analyzed through numerical simulations. In an appendix we also present a novel, computationally efficient, deblurring algorithm that is based on a Singular Value Decomposition (SVD) approximation of the variant PSF, and which is usable with any standard space-invariant direct deblurring algorithm.

Key words. Methods: data analysis – Methods: statistical – Techniques: Image processing

1. Introduction

In a recent paper Vio et al. (2003c) have presented an efficient solution for the problems of composition and simultaneous deblurring of a set of observed images (of a fixed object), where each observed image is degraded by a different spatially invariant point spread function (PSF). An important field where this methodology can be exploited are the restoration of images obtained by the Large Binocular Telescope (LBT). This instrument consists of two 8.4m mirrors on a common mount with a spacing of 14.4m between the centers (Angel et al. 1998). For a given orientation of the telescope, the diffraction-limited resolution along the center-to-center baseline is equivalent to that of a 22.8m mirror, while the resolution along the perpendicular direction is that of a single 8m mirror. A possible way to obtain an image with an improved and uniform spatial resolution is to simultaneously deconvolve the images taken with different orientations of the telescope. Another example is the multi-frequency observations of the Cosmic Microwave Background (CMB) obtained from satellites such as *PLANCK*. In fact, although some other physical components are present at the microwave frequencies, the images taken at high galactic latitudes are expected to be almost entirely dominated by the CMB contribution in a large range of observing frequencies (e.g., see Stolyarov et al. 2002). Since the PSFs corresponding to the images obtained at the various frequencies can be quite dissimilar, simultaneous deblurring can be useful for improving the extraction of CMB information from the data.

However, especially for LBT, because of the different corrections of the adaptive optics across the image domain, the PSF is expected to be space-variant (Bertero & Boccacci 2000). In this situation, the computational advantages provided by the methodology of Vio et al. (2003c) cannot be exploited. Here we extend such a methodology for dealing also with this more general case.

In Sect. 2 we formalize the problem and propose an efficient solution in Sect. 3. In Sect. 4 its performance is studied through numerical experiments. Finally, in Sect. 6 we close with final comments and conclusions.

Send offprint requests to: R. Vio

2. Formalization of the problem

Suppose one has p *observed* images, $g_j(x,y)$, $j=1,2,\ldots,p$ of a fixed two-dimensional object $f_0(x,y)$, each of which is degraded by a different blurring operator. In case of a generic blurring operator, $g_j(x,y)$ be modeled as

$$g_j(x,y) = \iint K_j(x,y,x'y')f_0(x',y')dx'dy', \tag{1}$$

where, for each j, $K_j(x,y)$ is a (possibly) space variant PSF.

In practice, the experimental images are discrete $N \times N$ arrays of pixels (for simplicity, we assume square images) contaminated by noise. Therefore model (1) has to be recast in the form

$$g_j(m,n) = \sum_{m',n'=0}^{N-1} K_j(m,n,m',n')f_0(m',n') + w_j(m,n), \tag{2}$$

where $w_j(m,n)$ is additive white noise. For the moment, we assume constant standard deviations across images: $\sigma_{w_j} = \sigma_w$.

The image restoration problem of interest is to find an estimate $f(m,n)$ of $f_0(m,n)$ given the observed images $\{g_j(m,n)\}$ and the known PSFs $\{K_j(m,n,m',n')\}$.

In case of a spatially invariant PSF, i.e., model (2) can be simplified in the form

$$g_j(m,n) = \sum_{m',n'=0}^{N-1} K_j(m-m',n-n')f_0(m',n') + w_j(m,n), \tag{3}$$

or equivalently

$$\hat{g}_j(m,n) = \hat{K}_j(m,n)\hat{f}_0(m,n) + \hat{w}_j(m,n), \tag{4}$$

where symbol "$\hat{}$" denotes the Discrete Fourier Transform (DFT). In the context of a LS approach, Vio et al. (2003c) have shown that a reliable and computationally efficient solution to this problem can be obtained on the basis of a weighted composition, say $\hat{\zeta}(m,n)$ (called the *mean image*), of the images $g_j(m,n)$

$$\hat{\zeta}(m,n) = \frac{\hat{g}_{j_0}(m,n) + \sum_{j\neq j_0}[\hat{K}_j^*(m,n)/\hat{K}_{j_0}^*(m,n)]\hat{g}_j(m,n)}{1+\sum_{j\neq j_0}[|\hat{K}_j(m,n)|^2/|\hat{K}_{j_0}(m,n)|^2]}, \tag{5}$$

where $\hat{K}_{j_0}(m,n)$, called the *mean PSF*, is given by

$$\hat{K}_{j_0}(m,n) = \max[\hat{K}_1(m,n), \hat{K}_2(m,n), \ldots, \hat{K}_p(m,n)]. \tag{6}$$

Here, the operator max[ ] extracts the element in the array with the largest modulus, and j_0 is the value of the corresponding index j. This composition is optimal in LS sense since $\hat{\zeta}(m,n)$ represents the DFT of "observed" image with the PSF whose energy distribution is maximally concentrated in the spatial domain compatibly with the original $\{K_j(m,n)\}$. In certain applications $\zeta(m,n)$ could be of interest by itself. It allows us to transform the multi-image deblurring problem into a classical single-image deblurring problem with a single PSF given by $\hat{K}_{j_0}(m,n)$. The computational benefits are obvious.

Unfortunately, in case of spatially variant PSFs, this approach cannot be used because model (2) cannot be represented in the Fourier domain.

3. A simple but efficient solution

Since the DFT is a linear operator, it is not difficult to see that if

$$\hat{s}(m,n) = \mathrm{DFT}\left[s(m,n)\right], \tag{7}$$

then

$$\hat{s}(m,n) = \mathrm{DFT}\left[\sum_{i=1}^{N_D} D_i(m,n)s(m,n)\right] = \sum_{i=1}^{N_D} \mathrm{DFT}\left[D_i(m,n)s(m,n)\right] = \sum_{i=1}^{N_D} \hat{s}_i(m,n). \tag{8}$$

Here, $D_i(m,n)$ are $N \times N$ matrices with entries in the range $[0,1]$ such that $\sum_i D_i(m,n) = 1$ for each value of (m,n), and $\hat{s}_i(m,n) = \mathrm{DFT}[D_i(m,n)s(m,n)]$. With this equation, it is possible to rewrite Eq. (5) in the form

$$\hat{\zeta}(m,n) = \sum_{i=1}^{N_D} \frac{\hat{g}_{ij_0}(m,n) + \sum_{j\neq j_0}[\hat{K}_j^*(m,n)/\hat{K}_{j_0}^*(m,n)]\hat{g}_{ij}(m,n)}{1+\sum_{j\neq j_0}[|\hat{K}_j(m,n)|^2/|\hat{K}_{j_0}(m,n)|^2]}. \tag{9}$$

Of course, this equation is of no practical interest. However, if the matrices $D_i(m,n)$ have non-zero entries only in correspondence to square or rectangular regions of the image domain, and each of such regions has its own PSF, then Eq. (9) can be written in the form

$$\hat{\zeta}(m,n) = \sum_{i=1}^{N_D} \frac{\hat{g}_{ij_0}(m,n) + \sum_{j\neq j_0}[\hat{K}^*_{ij}(m,n)/\hat{K}^*_{ij_0}(m,n)]\hat{g}_{ij}(m,n)}{1+\sum_{j\neq j_0}[|\hat{K}_{ij}(m,n)|^2/|\hat{K}_{ij_0}(m,n)|^2]} = \sum_{i=1}^{N_D} \hat{\zeta}_i(m,n). \tag{10}$$

In practice, this equation may be assumed to provide $\hat{\zeta}(m,n)$ in the case of a spatially variant PSF when the blur is approximately spatially invariant in the image domain fixed by the non-zero entries of $D_i(m,n)$.

The simplest form for $D_i(m,n)$ is obtained through *piecewise constant interpolation*, when the permitted values are either 0 or 1. In this case, the non-zero entries constitute a partitioning of the image domain. This means that for a given coordinate, say (m_0,n_0), $\hat{\zeta}_i(m_0,n_0)$ contains only the contribution of the component $\zeta_i(m_0,n_0)$ with non-zero entries in that subregion. Although in many situations such a choice provides satisfactory results, sometimes the resulting $\hat{\zeta}(m,n)$ suffers blocking artifacts that are problematic for the deblurring operation. A more sophisticated alternative, able to avoid this kind of problem, is based on a linear interpolation approach (see Nagy & O'Leary 1998). The basic idea is simpler to understand in the one-dimensional case.

If a one-dimensional domain is partitioned in a number of N_D subregions with dimension N_s, a useful set of overlapping masks, with dimension $2 \times N_s$, is shown in Fig. 1. In fact, since they go smoothly to zero, the boundary artifacts that affect each of the $\hat{s}_i(m,n)$ are strongly weakened. Moreover, the overlap works in such a way to further reduce these effects. The only exception is represented by the regions on the border of the domain.

In the two-dimensional case the argument is similar although made a bit more difficult by the larger number of masks that overlap at a give point of the image domain, and by the fact that there are two different situations for the subregions on the border of the frame. However, it is not difficult to see that, independently from the partitioning of the image domain, the resulting masks must have fixed values at the points marked in Fig. 1. Computationally, they can be obtained by interpolating these points through a *bilinear* interpolation (Press et al. 1992).

4. Some numerical experiments

We present results of some numerical experiments to show the reliability and the good performance of the methodology presented above. For each experiment we use a set of eight images, sized 400×400 pixels. Fig. 2 shows the spatial pattern of the variation of the PSF across the first image. The PSFs of the remaining images are obtained by rotating this figure by $22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5°$ degrees, respectively. The PSFs are bidimensional Gaussian with dispersion along the major axis set to twelve pixels, and to four pixels along the minor axis. The non-stationary pattern of the PSF has been deliberately chosen quite peculiar in order to test the capabilities of the algorithm in extreme situations.

Figs. 3-4 show the deblurring results for a set of 169 identical point-like objects uniformly distributed across the image. Noise is additive and Gaussian with a standard deviation of 2% of the maximum value of the blurred images. We have chosen these particular examples since, being the sources of non-smooth functions, deblurring is a difficult problem. The image $\hat{\zeta}_i(m,n)$ has been calculated by partitioning the *observed* images in 25 square subregions (i.e., $N_D = 25$), and by using piecewise constant interpolation for $D_i(m,n)$. Two deblurring methods have been used. The first is a sectioning iterative method based on a modified residual norm steepest descent method (MRNSD) that can be used to enforce non-negativity of the solution (Nagy & O'Leary 1997, 1998), whereas the second is a novel direct method based on a Singular Value Decomposition (SVD) approximation of the variant PSF (appendix A provides a short description) coupled with a Tikhonov technique that is computationally very fast. For comparison, the deblurring of the first image of the set (that with the PSF shown in Fig. 2) is also presented. In every case the improvement due to the use of the eight images is evident. The superiority of the results provided by the MRNSD approach is due to the fact that the Tikhonov method does not enforce a non-negativity constraint on the solution.

Although the use of piecewise constant interpolation for $D_i(m,n)$ can provide reasonable results, very often the use of the linearly interpolated version is able to remarkably improve them. Figs. 5-7 show the results of an experiment, carried out under the same condition of the previous one, that confirms the superiority of the interpolated method (here $N_s = 80$).

In various situations the choice of the linearly interpolation mask constitutes a necessity. In fact, when the constant interpolation mask is applied to the object shown in Fig. 8 (NB. here noise is additive and Gaussian with standard deviation of 2% the standard deviation of the dispersion of the blurred images), the results are not so good since $\hat{\zeta}_i(m,n)$ contains blocking artifacts that later are magnified by the deblurring operation. Consequently, the use of the linearly interpolated mask becomes necessary. In any case, although this approach produces better results, some problems still remain in form of fringes that contaminate $\zeta_i(m,n)$. This is linked to edge effects because the original image contains significant features close to its boundaries. The situation is made more troublesome by the fact that

the masks $D_i(m,n)$ overlap, thus causing a propagation of artifacts to the central regions of $\hat{\zeta}_i(m,n)$. For this reason, as explained in Vio et al. (2003c), before the calculation of $\hat{\zeta}_i(m,n)$, all the images $g_j(m,n)$ should be windowed through the matrix

$$h(m,n) = \begin{cases} 0.25 \times \alpha \times \beta, & 0 \leq m,n < N_w; \\ 0.25 \times \alpha \times \beta, & N - N_w \leq m,n < N; \\ 1 & \text{otherwise.} \end{cases} \tag{11}$$

Fig. 9 shows the results obtained after this operation with $N_s = 80$ and $N_w = 40$. A border of 40 pixels was removed by the windowing operation (see Vio et al. 2003c). Again, the improvement due to the use of the eight images is evident. In this specific case, since the nonnegativity of the solution is not as critical as in the experiments presented above, Tikhonov regularization is able to provide results as good as those provided by MRNSD, but at a much lower computational cost (and thus, much more quickly).

5. Extension to images with different noise levels

So far we have assumed that the noise level is the same in all the images ($\sigma_{w_j} = \sigma_w$), a condition that allows the combination of different images to provide the most interesting results. Moreover, this requirement is often satisfied in practice. For example, with LBT the images taken at different orientations of the telescope are expected to be characterized by the same noise level.

The extension of the method to images with different noise levels is straightforward (Vio et al. 2003c). If the different σ_{w_i} are known, one may account for the different variabilities by substituting Eq. (10) with

$$\hat{\zeta}(m,n) = \sum_{i=1}^{N_D} \frac{\hat{\varrho}_{ij_0}(m,n) + \sum_{j \neq j_0} [\hat{K}^*_{ij}(m,n)/\hat{K}^*_{ij_0}(m,n)]\hat{\varrho}_{ij}(m,n)}{1 + \sum_{j \neq j_0} [|\hat{K}_{ij}(m,n)|^2/|\hat{K}_{ij_0}(m,n)|^2]} = \sum_{i=1}^{N_D} \hat{\zeta}_i(m,n). \tag{12}$$

Here, $\hat{K}_{ij}(m,n) = \hat{K}_{ij}(m,n)/\sigma_{w_j}$ and $\hat{\varrho}_{ij}(m,n) = \hat{g}_{ij}(m,n)/\sigma_{w_j}$. The rest of the procedure remains as described in Sect. 3.

6. Conclusions

We have considered the problem of composition and simultaneously deblurring a set of images of a fixed object when the corresponding PSFs are space-variant and different from each other. We have developed a method, based on a LS approach, to efficiently transform a multi-image deblurring problem into a single-image one. This approach provides substantial savings in computational requirements, and can be implemented using numerical algorithms for single-image problems. These conclusions are confirmed by our numerical experiments. We have also presented a novel, computationally efficient, direct deblurring algorithm, which is based on a Singular Value Decomposition (SVD) approximation of the space variant PSF coupled with Tikhonov regularization.

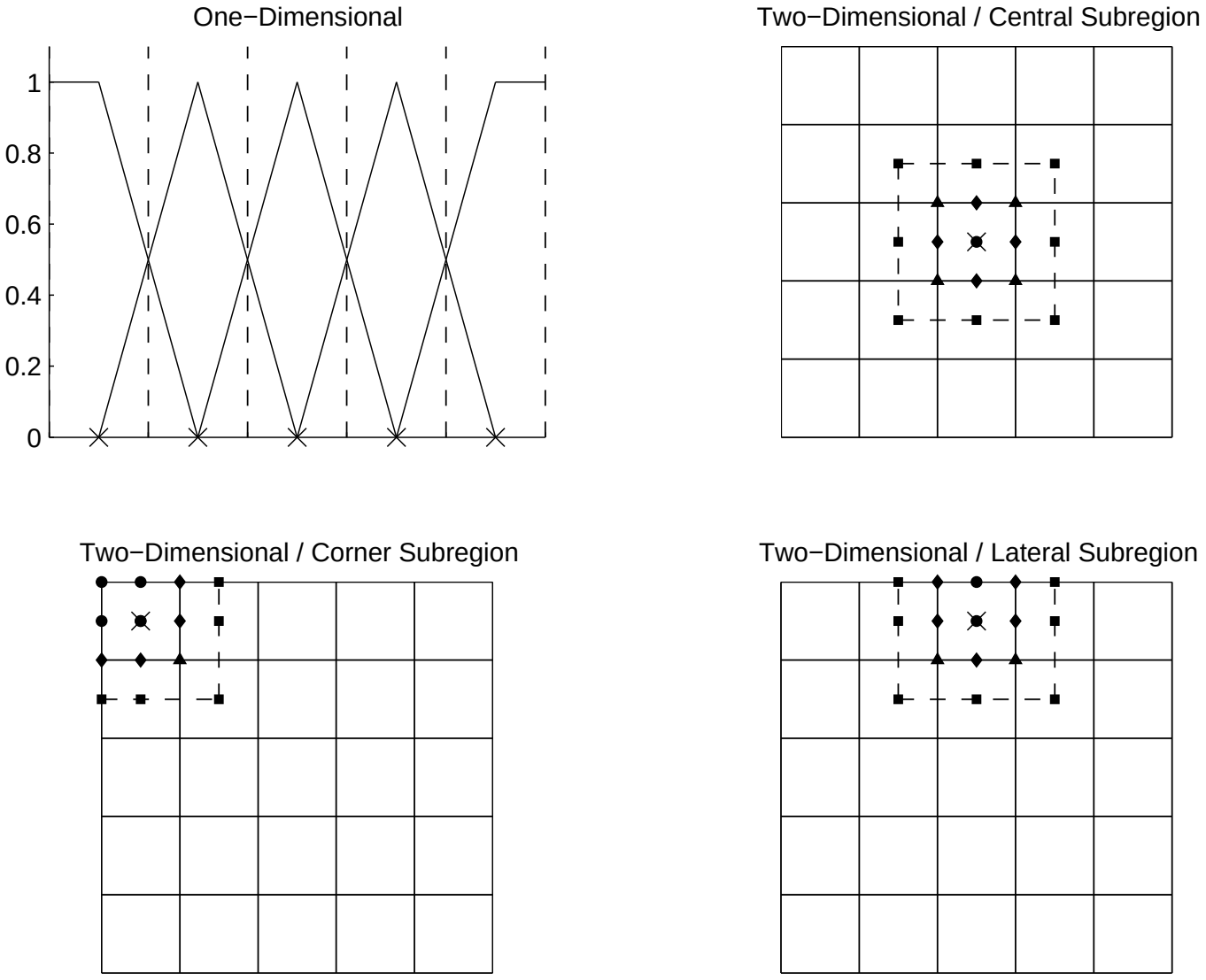

Fig. 1. One- and two-dimensional bases for the computation of the masks using the linear interpolation approach. In the one-dimensional case the dashed line indicates the limits of each subregion, whereas in the two-dimensional case they indicate the limits of the masks. Symbols $\times$ denote the center of a subregion, whereas the values of the masks at the marked points are given by $\bullet = 1$, $\blacklozenge = 0.5$, $\blacktriangle = 0.25$, $\blacksquare = 0$.

Appendix A: SVD approximation for spatially variant blurs

In matrix notation, model (2) can be written in the form

$$\mathbf{g} = \mathcal{K}\mathbf{f} + \mathbf{w}\,, \tag{A.1}$$

where $\mathcal{K}$ is a large $N^2 \times N^2$ ill-conditioned matrix which models the blurring phenomena, $\mathbf{f}$ is a vector of length N^2 representing the true image, and $\mathbf{g}$ is a vector representing the *observed* image. Because the problem is ill-posed, regularization methods are needed in order to compute an approximation of $\mathbf{f}$. A useful tool in constructing and analyzing regularization methods is the singular value decomposition (SVD). The problem is that $\mathcal{K}$ is a very large matrix, and it may not be computationally feasible to explicitly compute its SVD.

In Vio et al. (2003a,b) it has been shown that for spatially invariant blurs, spectral decompositions using fast transforms (such as DFT and the Discrete Cosine Transform) can sometimes be used in place of the SVD, but this depends on the PSF and the boundary condition. If it is not appropriate to use fast transforms, then another approach we considered was to compute an approximation of the SVD from a Kronecker product factorization of $\mathcal{K}$.

For spatially variant blurs, the situation is a bit more difficult. In fact, fast transform based decomposition methods are no longer a viable approach. However, here we show that the SVD approximation based on a Kronecker product factorization of $\mathcal{K}$ can be extended to the case of space variant blurs. This work builds on ideas initially proposed by Kamm & Nagy (1998).

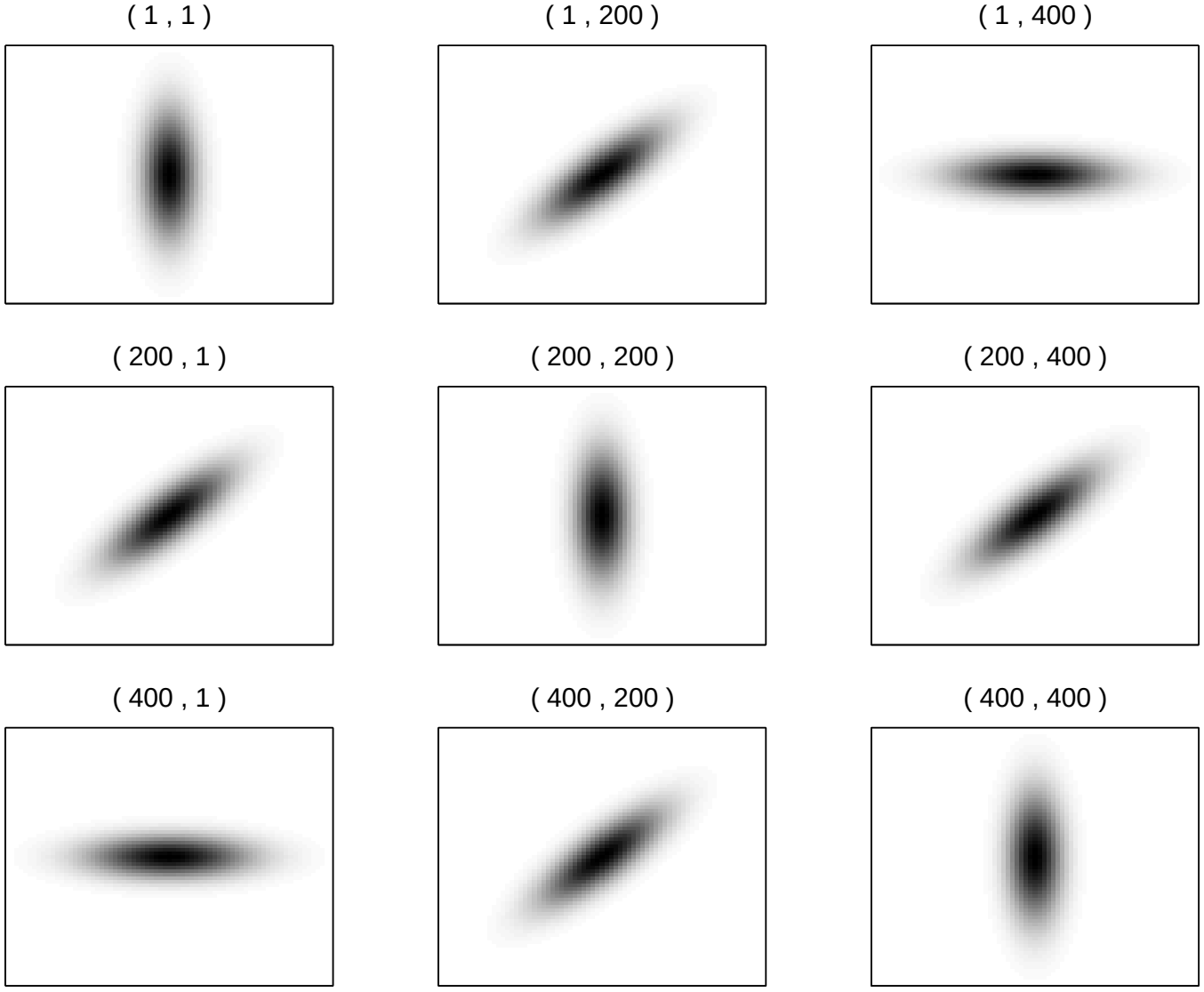

Fig. 2. Spatial variation of the PSF across the first image of the set used in the numerical experiments (see text). The pair of numbers in the header of each panel indicates the coordinates of the pixels to which the displayed PSF corresponds. In the intermediate pixels the PSFs are obtained through a linear interpolation of the displayed ones. Each PSF is given by a two-dimensional Gaussian function with a dispersion of 12 pixels along the major axis and 4 pixels along the minor axis.

A.1. Space Invariant Blurs

Suppose the blur is spatially invariant, and let $\boldsymbol{P}$ be an image representing the PSF. Then it can be shown that if $\boldsymbol{P}$ is decomposed as:

$$\boldsymbol{P} = \sum_{k=1}^{r} \mathbf{a}_k \mathbf{b}_k^T \,, \tag{A.2}$$

then the blurring matrix, $\boldsymbol{\mathcal{K}}$, can be decomposed as

$$\boldsymbol{\mathcal{K}} = \sum_{k=1}^{r} \boldsymbol{A}_k \otimes \boldsymbol{B}_k \,. \tag{A.3}$$

Here the notation $\otimes$ denotes Kronecker product, which is defined as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & a_{12}\boldsymbol{B} & \cdots & a_{1n}\boldsymbol{B} \\ a_{21}\boldsymbol{B} & a_{22}\boldsymbol{B} & \cdots & a_{2n}\boldsymbol{B} \\ \vdots & \vdots & & \vdots \\ a_{n1}\boldsymbol{B} & a_{n2}\boldsymbol{B} & \cdots & a_{nn}\boldsymbol{B} \end{bmatrix} . \tag{A.4}$$

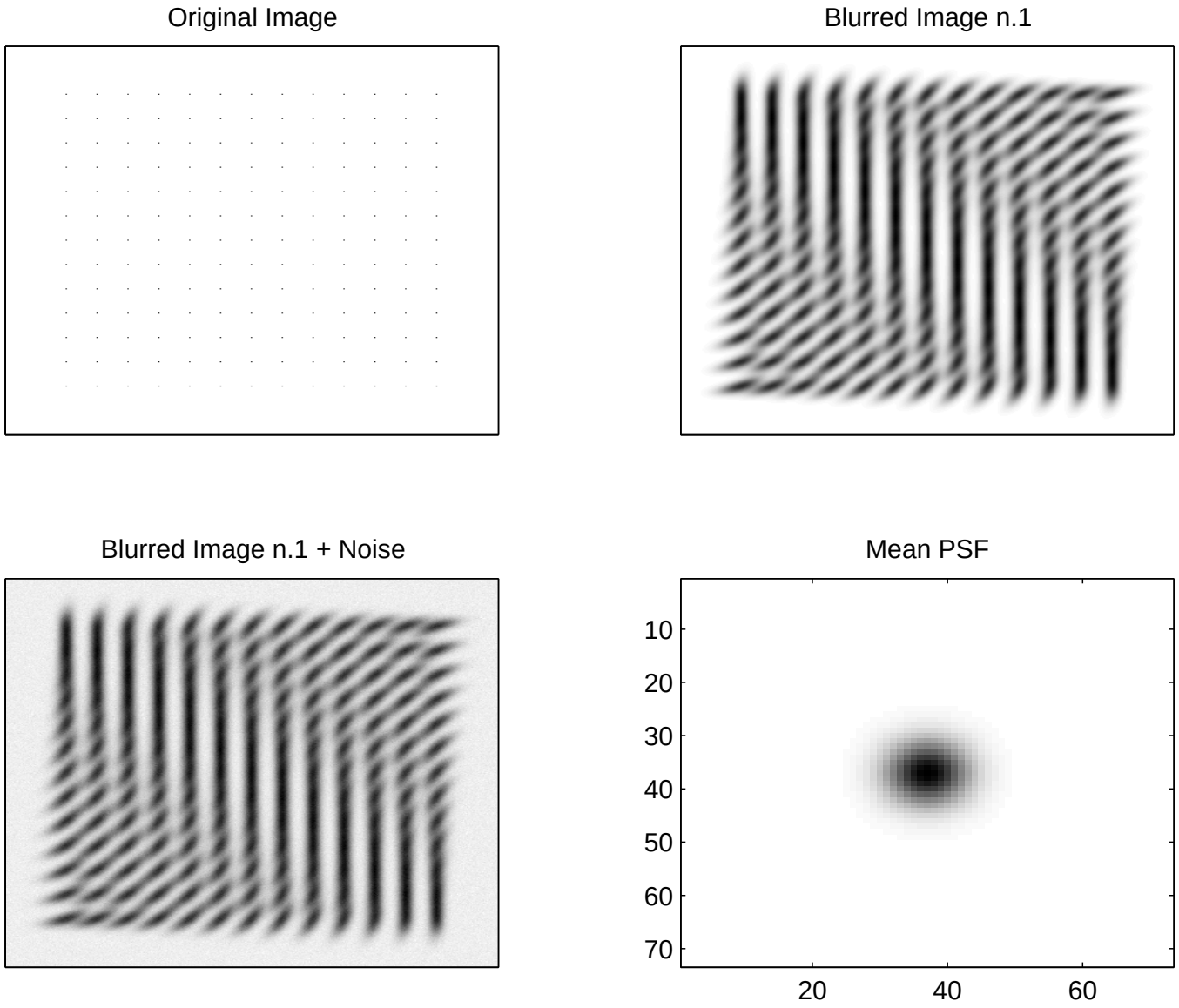

Fig. 3. Original image, blurred version and blurred + noise version of the first image of the set (see text), and corresponding mean PSF. The images are 400×400 pixels.

The matrices $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are defined by the vectors $\mathbf{a}_k$ and $\mathbf{b}_k$, respectively. In particular, if the center of the PSF is located at pixel (i, j), then

– Zero boundary conditions imply $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are Toeplitz matrices:

$$\boldsymbol{A}_k = \begin{pmatrix} a_i^{(k)} & \cdots & a_1^{(k)} & & \\ \vdots & \ddots & & \ddots & \\ a_n^{(k)} & & \ddots & & a_1^{(k)} \\ & \ddots & & \ddots & \vdots \\ & & a_n^{(k)} & \cdots & a_j^{(k)} \end{pmatrix} \quad \text{and} \quad \boldsymbol{B}_k = \begin{pmatrix} b_j^{(k)} & \cdots & b_1^{(k)} & & \\ \vdots & \ddots & & \ddots & \\ b_n^{(k)} & & \ddots & & b_1^{(k)} \\ & \ddots & & \ddots & \vdots \\ & & b_n^{(k)} & \cdots & b_j^{(k)} \end{pmatrix}. \tag{A.5}$$

– Similarly, for periodic boundary conditions, $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are circulant matrices.

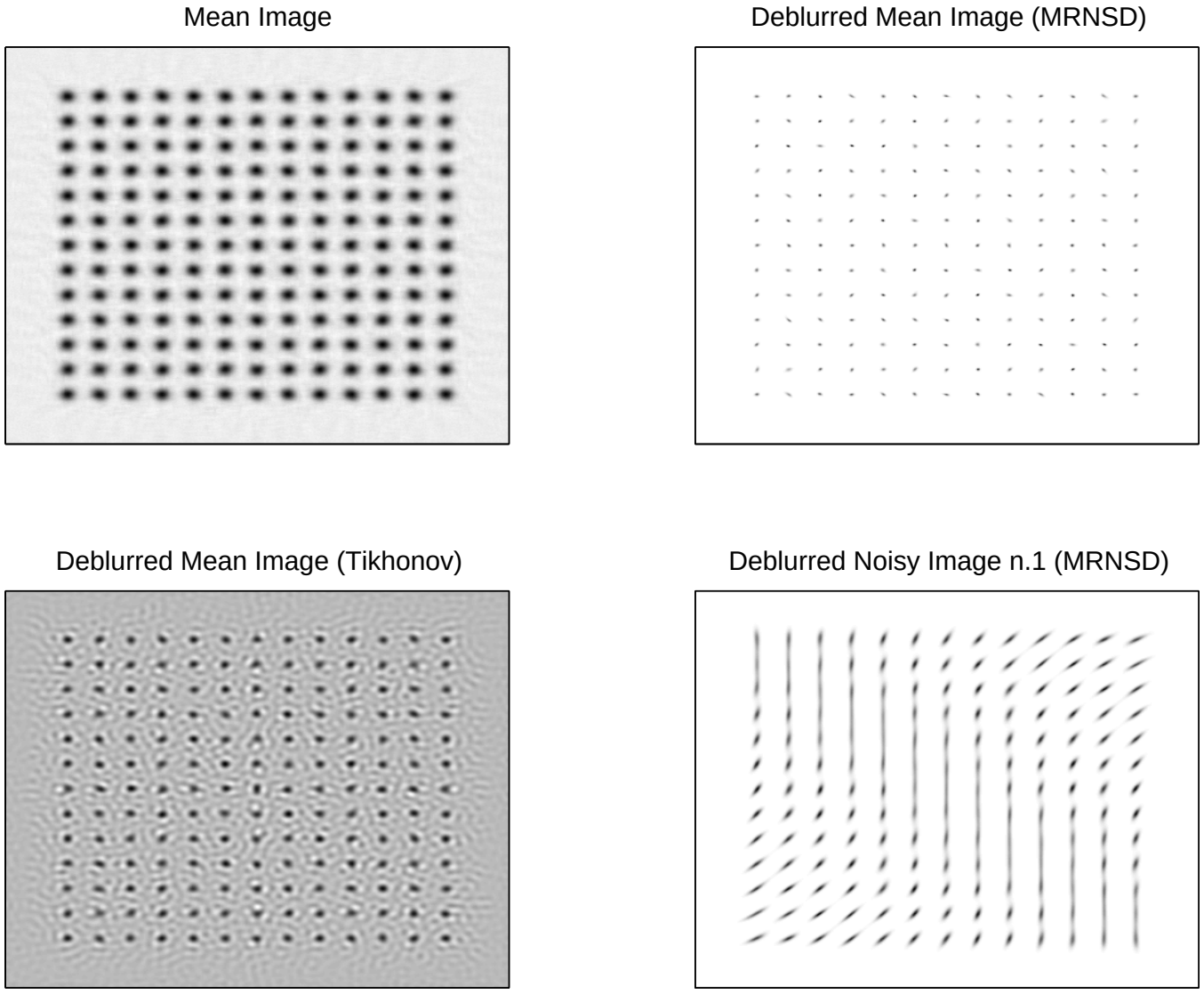

Fig. 4. Mean image $\zeta(m,n)$ (with $D_i(m,n)$ obtained using piecewise constant interpolation), MRNSD and Tikhonov deblurred versions of $\zeta(m,n)$, and MRNSD deblurred version of the first image of the set (see text) for the image shown in Fig. 3. For the MRNSD method the images shown are those obtained after 300 iterations of the algorithm, whereas for Tikhonov it is the one with the smallest standard deviation of the true residuals; for the mean image and first image of the set the standard deviation of the true residuals are, respectively, 0.028 (MRNSD) - 0.032 (Tikhonov), and 0.032.

– In the case of reflexive boundary conditions, $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are Toeplitz-plus-Hankel matrices:

$$\boldsymbol{A}_k = \begin{pmatrix} a_i^{(k)} & \cdots & a_1^{(k)} & & \\ \vdots & \ddots & & \ddots & \\ a_n^{(k)} & & \ddots & & a_1^{(k)} \\ & \ddots & & \ddots & \vdots \\ & & a_n^{(k)} & \cdots & a_i^{(k)} \end{pmatrix} + \begin{pmatrix} a_{i+1}^{(k)} & \cdots & a_n^{(k)} & & \\ \vdots & \iddots & & & \\ a_n^{(k)} & & & & a_1^{(k)} \\ & & & \iddots & \vdots \\ & a_1^{(k)} & \cdots & & a_{i-1}^{(k)} \end{pmatrix}, \tag{A.6}$$

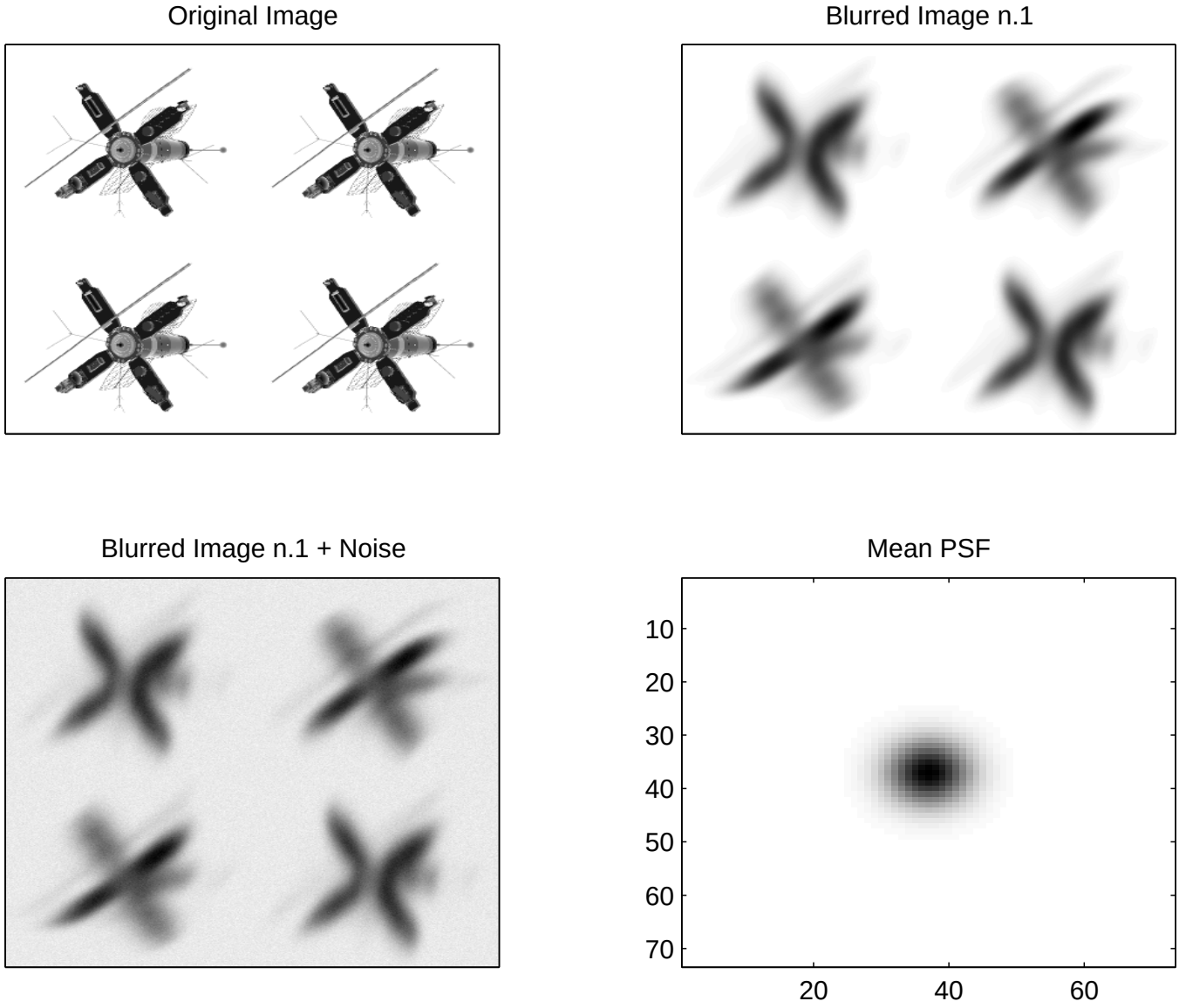

Fig. 5. Original image, blurred version and blurred + noise version of the first image of the set (see text), and corresponding mean PSF. The images are 400×400 pixels.

and

$$\boldsymbol{B}_k = \begin{pmatrix} b_j^{(k)} & \cdots & b_1^{(k)} & & \\ \vdots & \ddots & & \ddots & \\ b_m^{(k)} & & \ddots & & b_1^{(k)} \\ & \ddots & & \ddots & \vdots \\ & & b_m^{(k)} & \cdots & b_j^{(k)} \end{pmatrix} + \begin{pmatrix} b_{j+1}^{(k)} & \cdots & b_m^{(k)} & & \\ \vdots & \ddots & & & \\ b_m^{(k)} & & & & b_1^{(k)} \\ & & & \ddots & \vdots \\ & b_1^{(k)} & \cdots & & b_{j-1}^{(k)} \end{pmatrix} . \tag{A.7}$$

The factorization of $\boldsymbol{P}$ can be computed using the SVD. In particular, it is possible to compute

$$\boldsymbol{P} = \sum_{k=1}^{r} \sigma_k \mathbf{u}_k \mathbf{v}_k^T , \tag{A.8}$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, and thus $\mathbf{a}_k = \sqrt{\sigma_k}\mathbf{u}_k$ and $\mathbf{b}_k = \sqrt{\sigma_k}\mathbf{v}_k$. Using the SVD in this way, we obtain the Kronecker product factorization of $\boldsymbol{\mathcal{K}}$ given in equation (A.3) with the property that $\boldsymbol{A}_1$ has more information than $\boldsymbol{A}_2$, etc., and similarly for $\boldsymbol{B}_k$. In particular, in some measure, $\boldsymbol{A}_1 \otimes \boldsymbol{B}_1$ is the best Kronecker product factorization of

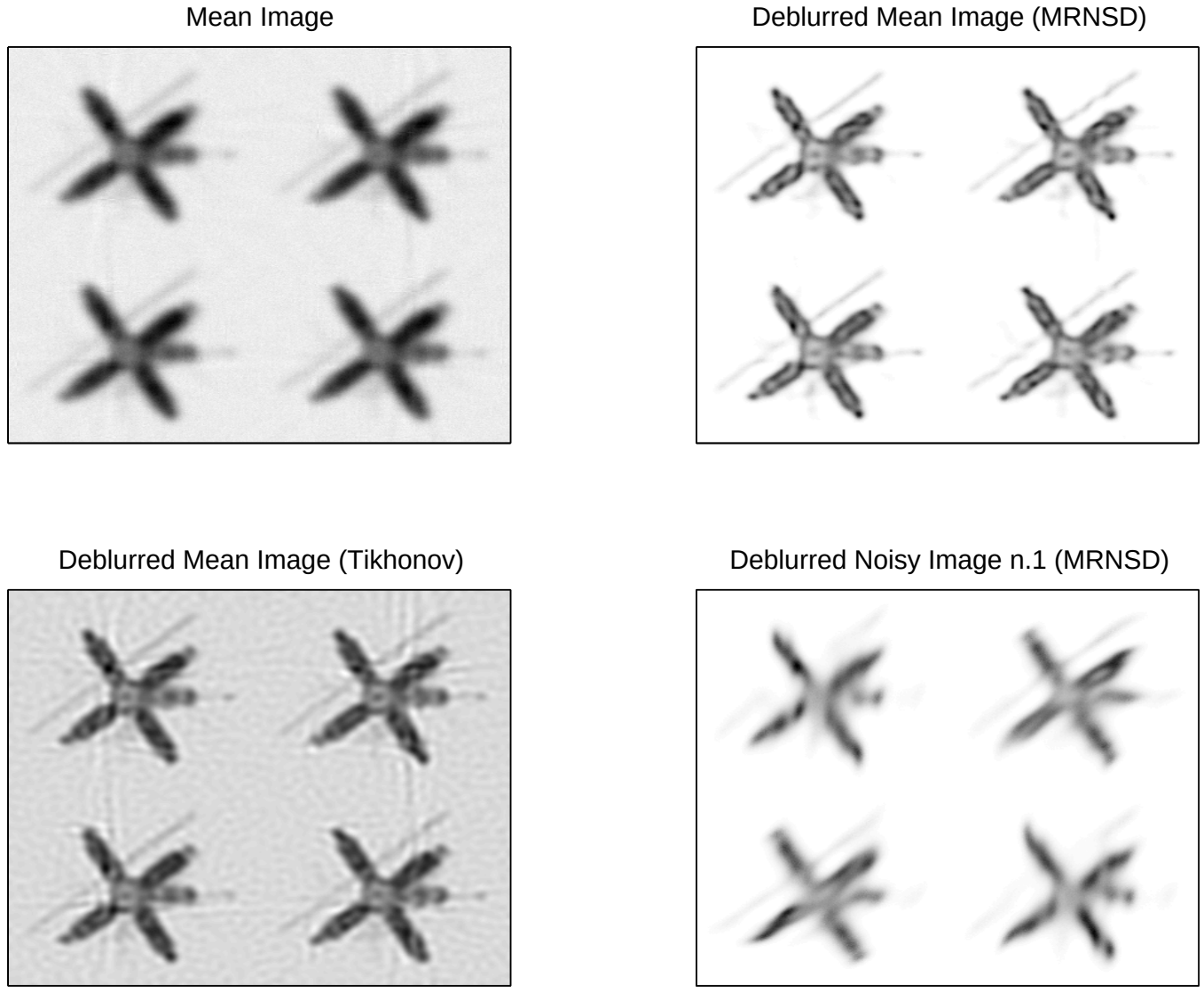

Fig. 6. Mean image $\zeta(m,n)$ (with $D_i(m,n)$ obtained using piecewise constant interpolation), MRNSD and Tikhonov deblurred versions of $\zeta(m,n)$, and MRNSD deblurred version of the first image of the set (see text) for the image shown in Fig. 5. The images shown are the ones with the smallest standard deviation of the true residuals; for the mean image and first image of the set these are, respectively, 0.082 (MRNSD) - 0.088 (Tikhonov), and 0.109.

$\mathcal{K}$. This observation is important because we can exploit Kronecker product structure of matrices in our computation. In particular, the SVD of a Kronecker product can be computed as:

$$\boldsymbol{A} \otimes \boldsymbol{B} = (\boldsymbol{U}_a \boldsymbol{\Sigma}_a \boldsymbol{V}_a^T) \otimes (\boldsymbol{U}_b \boldsymbol{\Sigma}_b \boldsymbol{V}_b^T) = (\boldsymbol{U}_a \otimes \boldsymbol{U}_b)(\boldsymbol{\Sigma}_a \otimes \boldsymbol{\Sigma}_b)(\boldsymbol{V}_a \otimes \boldsymbol{V}_b)^T . \tag{A.9}$$

We do not need to explicitly form the big matrices $(\boldsymbol{U}_a \otimes \boldsymbol{U}_b)$, etc., because properties of Kronecker products allow us to organize the computations so that we need only work with the smaller matrices $\boldsymbol{U}_a, \boldsymbol{U}_b$, etc.

The above remarks suggest that an SVD approximation can be obtained by computing the SVD of $\boldsymbol{A}_1 \otimes \boldsymbol{B}_1$. It is possible, though, to incorporate some information from $\boldsymbol{A}_2, \boldsymbol{A}_3, \ldots$ and $\boldsymbol{B}_2, \boldsymbol{B}_3, \ldots$. Following an approach outlined in Kamm & Nagy (2000); Nagy et al. (2003), this can be done as follows:

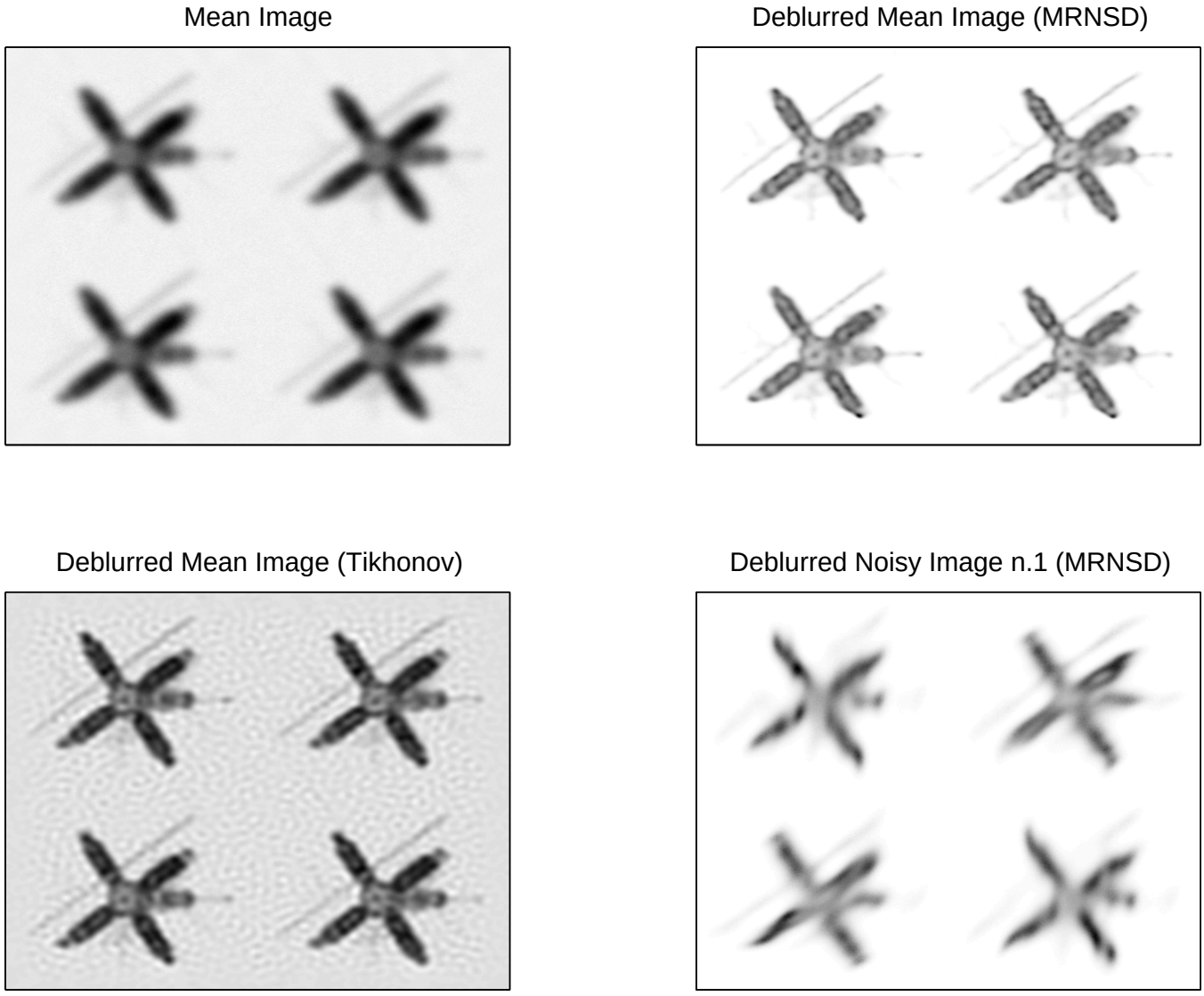

Fig. 7. Mean image $\zeta(m,n)$ (with $D_i(m,n)$ obtained using linear interpolation), MRNSD and Tikhonov deblurred versions of $\zeta(m,n)$, and MRNSD deblurred version of the first image of the set (see text) for the image shown in Fig. 5. The images shown are the ones with the smallest standard deviation of the true residuals; for the mean image and first image of the set these are, respectively, 0.076 (MRNSD) - 0.082 (Tikhonov) and 0.109.

– Let $\boldsymbol{A}_1 = \boldsymbol{U}_a \boldsymbol{\Sigma}_a \boldsymbol{V}_a^T$ and $\boldsymbol{B}_1 = \boldsymbol{U}_b \boldsymbol{\Sigma}_b \boldsymbol{V}_b^T$. We use $\boldsymbol{U}_a \otimes \boldsymbol{U}_b$ and $\boldsymbol{V}_a \otimes \boldsymbol{V}_b$ as the approximate singular vectors of $\boldsymbol{\mathcal{K}}$. The approximate singular values are obtained by computing the diagonal matrix:

$$\begin{aligned}
\boldsymbol{\Sigma} &= \mathrm{diag}\left[(\boldsymbol{U}_a \otimes \boldsymbol{U}_b)^T \boldsymbol{\mathcal{K}} (\boldsymbol{V}_a \otimes \boldsymbol{V}_b)\right] \\
&= \mathrm{diag}\left[(\boldsymbol{U}_a \otimes \boldsymbol{U}_b)^T \left(\sum_{k=1}^{r} (\boldsymbol{A}_k \otimes \boldsymbol{B}_k)\right) (\boldsymbol{V}_a \otimes \boldsymbol{V}_b)\right] \\
&= \mathrm{diag}\left[\sum_{k=1}^{r} (\boldsymbol{U}_a \otimes \boldsymbol{U}_b)^T (\boldsymbol{A}_k \otimes \boldsymbol{B}_k)(\boldsymbol{V}_a \otimes \boldsymbol{V}_b)\right] \\
&= \mathrm{diag}\left[\sum_{k=1}^{r} (\boldsymbol{U}_a^T \boldsymbol{A}_k \boldsymbol{V}_a) \otimes (\boldsymbol{U}_b^T \boldsymbol{B}_k \boldsymbol{V}_b)\right] \\
&= \sum_{k=1}^{r} \mathrm{diag}(\boldsymbol{U}_a^T \boldsymbol{A}_k \boldsymbol{V}_a) \otimes \mathrm{diag}(\boldsymbol{U}_b^T \boldsymbol{B}_k \boldsymbol{V}_b)
\end{aligned} \tag{A.10}$$

– That is, $\boldsymbol{\mathcal{K}} \approx (\boldsymbol{U}_a \otimes \boldsymbol{U}_b) \boldsymbol{\Sigma} (\boldsymbol{V}_a \otimes \boldsymbol{V}_b)^T$.

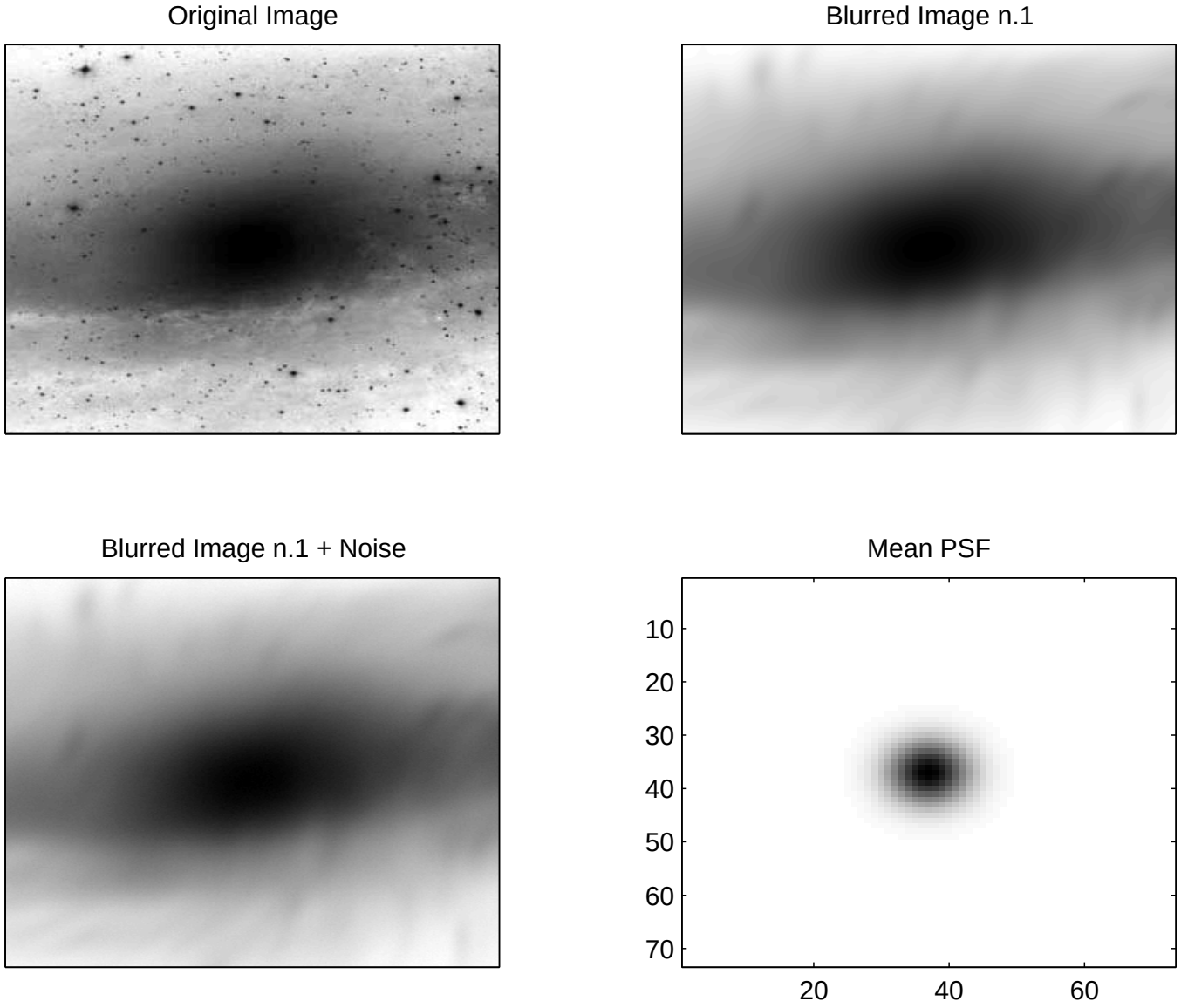

Fig. 8. Original image, blurred version and blurred + noise version of the first image of the set (see text), and corresponding mean PSF. The images are 400×400 pixels, but oly the central 360×360 pixels are shown (see text).

A.2. Space Variant Blurs

In the case of space variant blurs, a single PSF does not provide enough information to obtain a complete representation of the matrix, $\boldsymbol{\mathcal{K}}$. In fact, one PSF provides only information about one column (or row) of $\boldsymbol{\mathcal{K}}$. In order to get all N^2 columns, we would need N^2 PSFs, each one centered at a different pixel location. It would be extremely expensive to to find all of these PSFs. So instead, we consider a simplifying assumption that the blur is locally spatially invariant (this approach has been used in Boden et al. 1996; Faisal et al. 1995; Nagy & O'Leary 1998). Specifically, suppose that the image domain is partitioned into $p \times q$ rectangular regions, each with a PSF,

$$\begin{array}{|c|c|c|} \hline \boldsymbol{P}_{11} & \cdots & \boldsymbol{P}_{1q} \\ \hline \vdots & & \vdots \\ \hline \boldsymbol{P}_{p1} & \cdots & \boldsymbol{P}_{pq} \\ \hline \end{array}$$

Suppose the image has $N \times N$ pixels, and define the dimensions of the ijth region as $l_i \times m_i$, where $l_1 + \cdots + l_p = m_1 + \cdots + m_q = n$. Further, let $\boldsymbol{D}_{ij}$ be a $\{0,1\}$ diagonal matrix representing the pixels in the ijth region. That is, $\boldsymbol{D}_{ij} = \mathrm{diag}(d_1, \ldots, d_N)$, where $d_k = d_{(s-1)n+t} = 1$ if the point (s,t) is in the ijth region; otherwise, $d_k = 0$. With this

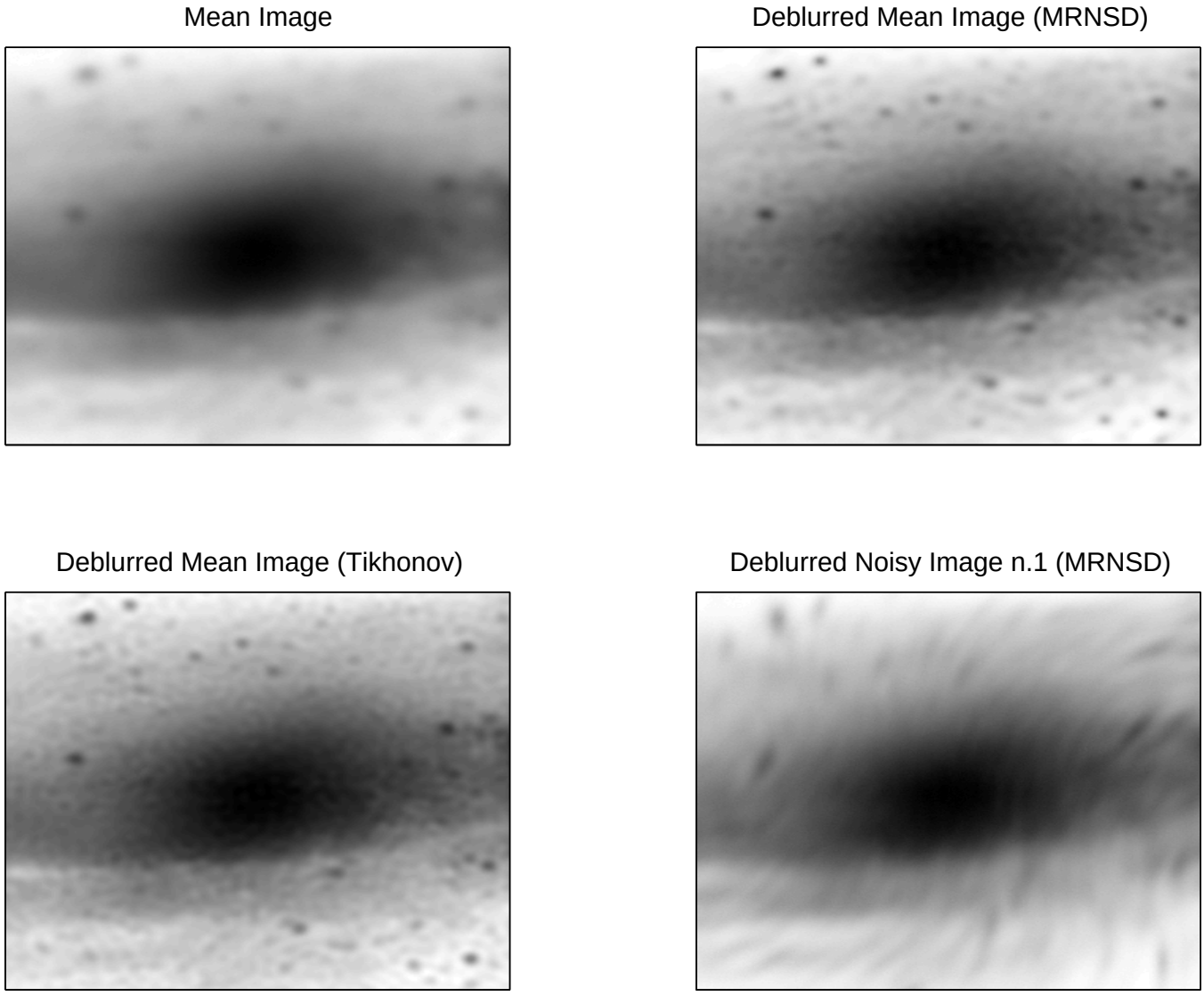

Fig. 9. Mean image $\zeta(m,n)$ (with $D_i(m,n)$ obtained using linear interpolation), MRNSD and Tikhonov deblurred versions of $\zeta(m,n)$, and MRNSD deblurred version of the first image of the set (see text) for the image shown in Fig. 8. The images shown are the ones with the smallest standard deviation of the true residuals; for the mean image and first image of the set these are, respectively, 0.036 (MRNSD) - 0.036 (Tikhonov), and 0.043.

notation, we represent the matrix $\boldsymbol{\mathcal{K}}$ as

$$\boldsymbol{\mathcal{K}} = \sum_{i=1}^{p} \sum_{j=1}^{q} \boldsymbol{D}_{ij} \boldsymbol{\mathcal{K}}_{ij} \,, \tag{A.11}$$

where $\boldsymbol{\mathcal{K}}_{ij}$ is a matrix defined be the spatially invariant PSF $\boldsymbol{P}_{ij}$.

Now using the Kronecker product factorization for spatially invariant blurs described in the previous section, we obtain (without loss of generality, we assume each of the PSFs has rank equal to r):

$$\boldsymbol{\mathcal{K}} = \sum_{i=1}^{p} \sum_{j=1}^{q} \boldsymbol{D}_{ij} \left(\sum_{k=1}^{r} \boldsymbol{A}_k^{(i)} \otimes \boldsymbol{B}_k^{(j)} \right) \,, \tag{A.12}$$

where the matrices $\boldsymbol{A}_k^{(i)}$ and $\boldsymbol{B}_k^{(j)}$ are Toeplitz or Toeplitz-plus-Hankel matrices (depending on the boundary conditions), as described in the previous section. We can simplify this expression by observing that each of the diagonal matrices, $\boldsymbol{D}_{ij}$ is separable, and can be written as:

$$\boldsymbol{D}_{ij} = \hat{\boldsymbol{D}}_i \otimes \tilde{\boldsymbol{D}}_j \,, \tag{A.13}$$

where

$$\begin{aligned}
\hat{\boldsymbol{D}}_1 &= \mathrm{diag}(\overbrace{1,\ldots,1}^{l_1},0,\ldots,0), & \tilde{\boldsymbol{D}}_1 &= \mathrm{diag}(\overbrace{1,\ldots,1}^{m_1},0,\ldots,0)\\
\hat{\boldsymbol{D}}_2 &= \mathrm{diag}(\overbrace{0,\ldots,0}^{l_1},\overbrace{1,\ldots,1}^{l_2},0,\ldots,0), & \tilde{\boldsymbol{D}}_2 &= \mathrm{diag}(\overbrace{0,\ldots,0}^{m_1},\overbrace{1,\ldots,1}^{m_2},0,\ldots,0)\\
&\vdots & &\vdots\\
\hat{\boldsymbol{D}}_p &= \mathrm{diag}(0,\ldots,0,\overbrace{1,\ldots,1}^{l_p}), & \tilde{\boldsymbol{D}}_q &= \mathrm{diag}(0,\ldots,0,\overbrace{1,\ldots,1}^{m_q}).
\end{aligned} \tag{A.14}$$

Thus,

$$\begin{aligned}
\boldsymbol{\mathcal{K}} &= \sum_{i=1}^{p}\sum_{j=1}^{q}\boldsymbol{D}_{ij}\left(\sum_{k=1}^{r}\boldsymbol{A}_k^{(i)}\otimes\boldsymbol{B}_k^{(j)}\right)\\
&= \sum_{i=1}^{p}\sum_{j=1}^{q}(\hat{\boldsymbol{D}}_i\otimes\tilde{\boldsymbol{D}}_j)\left(\sum_{k=1}^{r}\boldsymbol{A}_k^{(i)}\otimes\boldsymbol{B}_k^{(j)}\right)\\
&= \sum_{k=1}^{r}\sum_{i=1}^{p}\sum_{j=1}^{q}\left(\hat{\boldsymbol{D}}_i\boldsymbol{A}_k^{(i)}\otimes\tilde{\boldsymbol{D}}_k\boldsymbol{B}_k^{(j)}\right)\\
&= \sum_{k=1}^{r}\left[\left(\sum_{i=1}^{p}\hat{\boldsymbol{D}}_i\boldsymbol{A}_k^{(i)}\right)\otimes\left(\sum_{j=1}^{q}\tilde{\boldsymbol{D}}_j\boldsymbol{B}_k^{(j)}\right)\right]\\
&= \sum_{k=1}^{r}(\boldsymbol{A}_k\otimes\boldsymbol{B}_k)\ ,
\end{aligned} \tag{A.15}$$

where

$$\boldsymbol{A}_k = \sum_{i=1}^{p}\hat{\boldsymbol{D}}_i\boldsymbol{A}_k^{(i)} \quad \text{and} \quad \boldsymbol{B}_k = \sum_{j=1}^{q}\tilde{\boldsymbol{D}}_j\boldsymbol{B}_k^{(j)}\,. \tag{A.16}$$

Once $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are constructed, the approximate SVD of $\boldsymbol{\mathcal{K}}$ is computed just as it was in the spatially invariant case described in Section A.1.

References

Angel, J.R.P., Hill, J.M., Strittmatter, P.A., Salinari, P., & Weigelt, G., 1998, in: Astronomical Interferometry, eds., R.D. Reasenberg, Proc. SPIE, 3352, 881

Bertero, M., & Boccacci, P., 2000, A&A, 147, 323

Boden, A.F., Redding, D.C., Hanisch, R.J., & Mo, J. 1995, J. Opt. Soc. Am. A, 13, 1537

Faisal, M., Lanterman, A.D., Snyder, D.L., & White, R.L. 1995, J. Opt. Soc. Am. A, 12, 2593

Kamm, J., & Nagy, J.G. 1998, J. Kamm and J. G. Nagy., in Advanced Signal Processing Algorithms, Architectures, and Implementations VIII, ed. F.T. Luk, 3461, 358, SPIE

Kamm, J., & Nagy, J.G. 2000, SIAM J. Matrix Anal. Appl.,22, 155

Nagy, J.G., & O'Leary, in Advanced Signal Processing Algorithms, Architectures, and Implementations VIII, ed. F.T. Luk, 3462, 388, SPIE

Nagy, J.G., & O'Leary, D.P. 1998, SIAM J. Sci. Comput., 19, 1063

Nagy, J.G., Ng, M.K., & Perrone, L. 2003, *submitted*

Press, W.H., Teukolsky, S.A., Vetterling, W.T., & Flannery, B.P 1992, Numerical Recipes (Cambridge: Cambridge University Press)

Stolyarov, V., Hobson, M.P., Ashdown,, M.A.J., & Lasenby, A.N. 2002, MNRAS, 336, 97

Vio, R., Nagy, J., Tenorio, L., et al. 2003a, A&A, 389, 404

Vio, R., Nagy, J., Tenorio, L., et al. 2003b, A&A, 408, 835

Vio, R., Nagy, J., Tenorio, L., & Wamsteker, W. 2003c, A&A, *in press*

Vogel, C. R. 2002, Computational Methods for Inverse Problems, Ser. in Front. in Appl. Math., Vol.24, (Philadelphia: SIAM)