



HAL
open science

Distributed Deblurring of Large Images of Wide Field-of-View

Rahul Mourya, André Ferrari, Rémi Flamary, Pascal Bianchi, Cédric Richard

► **To cite this version:**

Rahul Mourya, André Ferrari, Rémi Flamary, Pascal Bianchi, Cédric Richard. Distributed Deblurring of Large Images of Wide Field-of-View. 2017. hal-01520266

HAL Id: hal-01520266

<https://hal.science/hal-01520266>

Preprint submitted on 15 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed Deblurring of Large Images of Wide Field-Of-View

Rahul Mourya, André Ferrari, Rémi Flamary, Pascal Bianchi, and Cédric Richard

Abstract—Image deblurring is an economic way to reduce certain degradations (blur and noise) in acquired images. Thus, it has become essential tool in high resolution imaging in many applications, e.g., astronomy, microscopy or computational photography. In applications such as astronomy and satellite imaging, the size of acquired images can be extremely large (up to gigapixels) covering wide field-of-view suffering from shift-variant blur. Most of the existing image deblurring techniques are designed and implemented to work efficiently on centralized computing system having multiple processors and a shared memory. Thus, the largest image that can be handle is limited by the size of the physical memory available on the system. In this paper, we propose a distributed nonblind image deblurring algorithm in which several connected processing nodes (with reasonable computational resources) process simultaneously different portions of a large image while maintaining certain coherency among them to finally obtain a single crisp image. Unlike the existing centralized techniques, image deblurring in distributed fashion raises several issues. To tackle these issues, we consider certain approximations that trade-offs between the quality of deblurred image and the computational resources required to achieve it. The experimental results show that our algorithm produces the similar quality of images as the existing centralized techniques while allowing distribution, and thus being cost effective for extremely large images.

Index Terms—Distributed optimization, Proximal projection, Consensus, Message-Passing-Interface, Shift-variant blur, Inverse problems, Image restoration.

I. INTRODUCTION

IN many applications, it is essential to have high resolution images for precise analysis and inferences from them. However, a certain amount of degradations (blur and noise) are unavoidable in many of the imaging systems due to several factors, such as the limited aperture size, the involved medium (both atmosphere and optics), and the imaging sensors. With advances in the computational technologies, the digital image restoration techniques, such as image deblurring, have been proven to be an economic way to enhance resolution, signal-to-noise ratio, and contrast of the acquired images. In this paper we can consider the situation where we assume that

The work was supported in part by Agence Nationale pour la Recherche (ODISSEE project, ANR-13-ASTR-0030 and MAGELLAN project, ANR-14-CE23-0004-01).

Rahul Mourya and Pascal Bianchi are with LTCI, Télécom ParisTech, Université Paris-Saclay, F-75013, Paris, France (e-mail:firstname.lastname@telecom-paristech.fr).

Rémi Flamary, André Ferrari and Cédric Richard are with Lagrange Laboratory, Université de Nice Sophia Antipolis, F-06108 Nice, France (email:firstname.lastname@unice.fr, and ferrari@unice.fr).

A preliminary shorter version of this work has been submitted to the conference EUSIPCO 2017.

the information about blur and the statistics of noise is known a priori, and then the restoration is referred to as nonblind deblurring. In general, image deblurring is an ill-posed inverse problem [1], [2]. In a Bayesian setting it can be expressed as *maximum-a-posteriori* estimation problem [3], which boils down to the following numerical optimization problem:

$$\mathbf{x}^* := \arg \min_{\mathbf{x}} \{ \Psi_{\text{data}}(\mathbf{y}, \mathbf{x}) + \lambda \Psi_{\text{prior}}(\mathbf{x}) \} \quad (1)$$

where the vectors $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{x} \in \mathbb{R}^n$ represent the observed (acquired blurry and noisy) image, and the unknown crisp image to be estimated, respectively. Two dimensional (2D) images are represented as column vectors by lexicographically ordering their pixels values. The first term Ψ_{data} in (1) is called *likelihood* or data-fidelity term that depends upon the noise and the image formation model. The second term Ψ_{prior} is called *a priori* or regularizer that imposes any prior knowledge on the unknown image \mathbf{x} . The scalar parameter λ keeps trade-off between the *likelihood* and *a priori* terms, which generally depends upon the noise level in the observed image.

Blur in an acquired image is characterized by the impulse response of the imaging system, commonly known as point-spread-function (PSF). As far as a narrow field-of-view is concerned, PSF can be considered constant throughout the field-of-view leading to a shift-invariant blur. The blurring operation in this case is modeled as simple a convolution between the sharp image and the PSF, and can be performed efficiently in Fourier domain. In many cases, however, blur varies throughout the field-of-view due to several causes: relative motion between the camera and the scene; moving objects with respect to the background; variable defocussing of non-planar scenes with some objects located in front or behind the in-focus plane; optical aberrations such as space-variant distortions, vignetting or phase aberrations. In here we consider shift-variant blur across the field-of-view of a planar scene, i.e. all the objects in the scene lie in-focus plane; see [4] for discussion on shift-variant blur in the case of non-planar scene. Two subtly different situations can be considered for shift-variant blur in planar scene: i) a large field-of-view is captured into several pieces by multiple imaging systems each capturing a small portion of the whole scene, and ii) the same scene is captured by a single imaging system having wide field-of-view. In the former case, each imaging system can have slightly different blur from each other operating under different settings. Thus, the captured images can have slightly different blurs from each other resulting into piecewise constant blur in the whole scene. We will refer to this

situation as piece-wise constant shift-variant blur. In the latter case, the blur in the whole field-of-view can vary smoothly, e.g. blur due to optical aberrations. We will refer to this situation as smooth shift-variant blur. The shift-invariant and piece-wise constant shift-variant blurs can be considered as special cases of the smooth shift-variant blur. The blurring operation in the smooth shift-variant case cannot be modeled by a simple convolution, and there does not exist any efficient and straight-forward way to perform the operation. However, there are some fast approximations of smooth shift-variant blur operators proposed in [5]–[7]; see [8] for detailed comparison between different approximations.

If we approximate the noise in observed image by a non-stationary white Gaussian noise as considered in [9] (see [10]–[12] for more refined noise model), then a discrete image formation model can be written as:

$$\mathbf{y} = \mathbf{H} \mathbf{x} + \varepsilon \quad (2)$$

where the matrix $\mathbf{H} \in \mathbb{R}^{m \times n}$, $m < n$, denotes the blur operator¹, and ε is the zero-mean non-stationary white Gaussian noise, i.e. $\varepsilon(\ell) \sim \mathcal{N}(0, \sigma^2(\ell))$ with $\sigma^2(\ell)$ denoting the noise variance at ℓ th component of the vector \mathbf{y} . We assume that the raw captured image is preprocessed to yield an image that closely follows the above image formation model (2). The preprocessing may include the correction of the background and of the flat field, the correction of defective pixels and possibly of its correlated noise, and the scaling of the image in photons.

Before we proceed further, let us introduce some more notations. Hereafter, we will use upper-case bold and lower-case bold letters for denoting matrices (and linear operators), and column vectors, respectively. For every set \mathcal{T} , we denote by $|\mathcal{T}|$ the cardinality of the set, and by $\mathbb{R}^{\mathcal{T}}$ the set of functions on $\mathcal{T} \rightarrow \mathbb{R}$. Let \mathcal{X} represent Euclidean space, then we denote by $\langle \cdot, \cdot \rangle$ the standard inner product on \mathcal{X} , and by $\|\cdot\|$ the Euclidean norm. Letting \mathbf{V} denote a positive definite linear operator of \mathcal{X} onto itself, we use the notation $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{V}} = \langle \mathbf{x}, \mathbf{V} \mathbf{y} \rangle$, and we denote by $\|\cdot\|_{\mathbf{V}}$ the corresponding norm. When \mathbf{V} is a diagonal operator of the form $(\mathbf{V} \mathbf{x}) : \ell \mapsto \alpha(\ell) \mathbf{x}(\ell)$ we equivalently denote $\|\mathbf{x}\|_{\mathbf{V}}^2$ by $\|\mathbf{x}\|_{\alpha}^2 = \sum_{\ell} \alpha(\ell) \mathbf{x}(\ell)^2$, where $\alpha = (\alpha_1, \alpha_2, \dots)$. We also denote $\|\mathbf{x}\|_{1, \mathbf{V}} = \sum_{\ell} \alpha(\ell) |\mathbf{x}(\ell)|$ the weighted L_1 -norm, simply noted $\|\mathbf{x}\|_1$ when \mathbf{V} is the identity. Moreover, if \mathbf{L} is a linear operator, we denote by \mathbf{L}^{\top} its adjoint operator.

Considering the above image formation model (2) with a known PSF, the (nonblind) image deblurring problem (1) can be explicitly expressed as:

$$\mathbf{x}^* := \arg \min_{\mathbf{x} \geq 0} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{H} \mathbf{x}\|_{\mathbf{W}}^2 + \lambda \phi(\mathbf{D} \mathbf{x}) \right\} \quad (3)$$

where the matrix \mathbf{W} is diagonal with its components given by $\mathbf{W}(\ell, \ell) = 1/\sigma^2(\ell)$ for observed pixels and $\mathbf{W}(\ell, \ell) = 0$ for unmeasured pixels. The function ϕ represents some regularizer (e.g., Tikhonov's L_2 -norm, sparsity promoting L_1 -norm, or

Huber's mixed L_1 - L_2 -norm), and \mathbf{D} is some linear operator, e.g., finite forward difference or some discrete wavelet transform. The formalism (3) of image deblurring problem is referred to as analysis-based approach where the unknown variable is expressed in the image domain itself. An alternative formalism, referred to as synthesis-based approach, is also considered in literature where the unknown variable is in some transformed domain, e.g., coefficients of some discrete wavelet; see [13] for detailed discussion on the two approaches and their comparisons. Without loss of generality, in this paper we consider only the analysis-based approach for expressing the image deblurring problem.

Depending upon the structures of the two terms in the optimization problem (3), the solution can be obtained in a single step i.e., a closed-form solution (e.g., Wiener deconvolution), or one has to rely on iterative solvers, which may require, at each iteration, the same computational expense or even more than the closed-form solution². The latter one is the frequently occurring case in many applications, e.g., the regularization term is not differentiable (nonsmooth). Though, there exists a vast literature on numerical optimization, but we refer the readers to [14]–[17] and the references therein for some of the recent and fast optimization algorithms specially developed for inverse problems in imaging.

A. Motivation: Deblurring Extremely Large Image

With the advances in imaging technologies, the applications in astronomy, satellite imagery and others are able to capture extremely large image of wide field-of-view. The size of such images can vary from few tens megapixels up to gigapixels. As discussed previously, we consider two imaging situations where such large images are acquired: piece-wise constant and smooth shift-variant blur. In the former situation, the images from multiple narrow field-of-view imaging systems are supposed to be mosaicked together into a single image provided that their relative positions within the whole field-of-view are known. For further simplification, we assume that all the narrow field-of-view imaging systems have the same noise performance. In the both imaging situations, it is highly desirable to obtain a single crisp image from the acquired image(s) for any further applications.

Image deblurring is well studied topic and a vast literature exists [1]–[3], [11], [16], [18]–[20] for moderate size (few megapixels) images with shift-invariant blur. These methods differ from each other in the way they formulate the data-fidelity term or the regularization term, and the optimization algorithms they propose to solve the resulting problem. The image deblurring problem becomes more complicated when one considers the images suffering from shift-variant blur. Some recent works [5]–[8] proposed efficient methods for deblurring images with shift-variant blur. However, all the methods listed above are designed and implemented to work efficiently on a centralized computing system, possibly with multiple processor cores sharing a single memory space.

¹Blur operator is a rectangular matrix since size of observed image is restricted by the physical size of the image sensor, i.e., sensor captures little less than what optics can see.

²Applying blur operator \mathbf{H} or its adjoint \mathbf{H}^{\top} is an expensive operation, and one may require to apply them several times per iterations of the solver.

Such a computing system is commonly referred to as multi-threaded shared memory system based on single instruction multiple data (SIMD) architecture. Hereafter, we will refer to all such existing methods by *centralized* deblurring techniques. In contrast to deblurring a moderate size image, deblurring an extremely large image is a challenging problem since the largest size of the image that can be handled is then limited by the capacity of shared memory available on the centralized system. It is not cost effective to build a centralized system with several processor cores and a huge shared memory when the modern distributed computing systems (consist of several connected processing nodes each having reasonable amount of computational resources) are proving to be far more economical way for solving huge-scale problems than the centralized system. Many domains such as machine learning, data mining [21], and others are already benefiting from distributed computing approach for discovering patterns in huge datasets distributed at different nodes. As per our knowledge, there are very few works, e.g., [22]–[25], considering distributed computing approach for image restoration/reconstruction problems. Within this context, we propose a distributed image deblurring algorithm for large images acquired from either of the imaging situations mentioned above.

B. Our contributions

In this paper, we propose a distributed algorithm for deblurring large images, which needs less frequent inter-node communication. Our algorithm can handle the acquired image(s) from both the imaging situations: smooth shift-variant and piece-wise constant shift-variant blur. For the first imaging situation, we consider splitting the large image into sufficiently small overlapping blocks and then deblur them simultaneously on the several processing nodes while maintaining certain coherencies among them so as to obtain a single crisp image without any transition artifacts among the deblurred blocks. For the second imaging situation, we assume that the relative positions of narrow images in the whole field-of-view are known, and there are certain overlaps among them. If a narrow image is not sufficiently small to fit in the memory of a node, then we can split it further into smaller overlapping blocks. Similar to the first situation, we deblur the small images simultaneously to obtain a single crisp image. To do so, we reformulate the image deblurring problem (1) into a distributed optimization problem with consensus, and then present an efficient optimization method to solve it. Our distributed deblurring algorithm is rather generic in the sense that it can handle different situations such as shift-invariant or shift-variant blur with any possible combination of the data-fidelity and regularization terms. Depending on the structures of the data-fidelity and regularizer, we can select any fast optimization algorithm for solving the local deblurring problems at processing nodes. By several numerical experiments, we show that our algorithm is cost effective in term of computational resources for large images, and is able obtain the similar quality of deblurred images as the existing *centralized* deblurring techniques that are applicable only to images of moderate size.

The remaining parts of the paper is organized as follows. In Section II, we discuss the difficulties associated with distributed formulations of image deblurring problem, and the possible approaches to overcome it. In Section III, we present the distributed formulation and an efficient distributed algorithm to solve it. We discuss the criteria for selecting the different parameters associated with our approach. In Section IV, we discuss the implementation details and present several numerical experiments where we compare the results obtained from different approaches. Finally, in Section V, we conclude our work with possible future enhancements.

II. DISTRIBUTED COMPUTING APPROACH FOR IMAGE DEBLURRING

A generic approach for dealing with large-scale problem is the “divide and conquer” strategy, in which a large-scale problem is decomposed into smaller subproblems in such way that solving them and assembling their results would produce the expected final result or, at least, reasonably close to it. Distributed computing approach has emerged as a framework of such a strategy. Distributed computing systems are consist of several processing nodes, each having a reasonable amount of computational resources (in term of memory and processor cores), connected together via some communication network so that nodes can exchange messages to achieve a certain common goal. They are built based on the multiple instructions multiple data (MIMD) architecture. Nodes in a distributed system may not be necessarily located at the same physical location, so a high speed communication links may not be always feasible among them. Thus, an efficient distributed algorithm is the one which is computation intensive rather than communication demanding. In many applications such as machine learning, data mining, distributed approaches have become de-facto standard for efficiently estimating extremely large number of parameters using huge dataset distributed on the different nodes. Taking such an inspiration, one can devise a distributed strategy for deblurring extremely large images. A possible approach is to use the distributed array abstraction available on modern distributed systems, and reimplement the standard *centralized* deblurring techniques using distributed arrays instead of a shared memory array. However, the bottleneck of such an approach would be extensive data communication among the nodes at each iteration of the optimization algorithm. To overcome this, a straightforward approach would be to split the given observed image \mathbf{y} into N smaller contiguous blocks $\mathbf{y}_i \in \mathbb{R}^{m_i}, i = 1, 2, \dots, N$, and deblur them independently to obtain the deblurred blocks $\hat{\mathbf{x}}_i \in \mathbb{R}^{n_i}, i = 1, \dots, N$ that can be merged to obtain the single deblurred image $\hat{\mathbf{x}} \in \mathbb{R}^n$. However, there may arise several issues from both the theoretical and practical point-of-views as discussed below.

A. Issues with Distributed Approach for Image Deblurring

For sake of simplicity, let us first consider the shift-invariant image deblurring problem, and point out some of the major issues:

- 1) Practically, it is infeasible to explicitly create the blur operator matrix \mathbf{H} . For this reason blur operation is efficiently performed in Fourier domain using Fast Fourier Transforms (FFT) algorithm. Thus, there is no straight forward way to split the matrix \mathbf{H} into block matrices $\mathbf{H}_i \in \mathbb{R}^{m_i \times n_i}$ so that they could be used to perform independent deblurring of the observed blocks \mathbf{y}_i such that the final result would be equivalent to using the original matrix \mathbf{H} on whole image \mathbf{y} . Any approximation of block matrices \mathbf{H}_i chosen not appropriately may create artifacts such as nonsmooth transition among the deblurred blocks.
- 2) Blur operation performed in Fourier domain assumes circular boundary condition. The circulant blur assumption in image deblurring process can lead to severe artifacts due to discontinuity at boundaries caused by the periodic extension of image [16]. Thus, deblurring the individual blocks \mathbf{y}_i can introduce boundary artifacts into them that will eventually worsen the quality of final deblurred image.
- 3) Splitting the image into blocks can also raise issues with the regularization term. Many frequently used regularizers are not separable, i.e., its value on the whole image is not equivalent to the sum its value on the blocks. Such an approximation can result into structural incoherencies among the deblurred blocks.

The issues mentioned above complicate further the shift-variant image deblurring problem, since there is no any efficient and straightforward way to formulate shift-variant blur operator. Considering the above issues associated with the naïve “split, deblur independently, and merge” approach, it is obvious that one needs to approximate the problem (3) in a way so that it can be solved efficiently in a distributed manner, yet obtain the solution that remains reasonably close to the solution of the original problem (3). In the next section, we describe how to tackle the above issues, and then we will present our distributed image deblurring algorithm.

B. Tackling the Issues in Distributed Image Deblurring

As discussed above, it is nontrivial problem to explicitly create a blur operator and then decompose it into block matrices to operate them independently for distributed image deblurring. However, we can formulate an approximation of the block matrices \mathbf{H}_i such that when they are used in deblurring process, certain homogeneities among the deblurred blocks are imposed. Let us consider the smooth shift-variant blur case. Provided that we are able to sample N local PSFs $\mathbf{h}_i, i = 1, \dots, N$ at regular grid points within the field-of-view, the approximation of shift-variant blur operator proposed in [6], [7] suggests an interesting idea to formulate the block matrices \mathbf{H}_i . Their approximations of the shift-variant blur operator is based upon the idea that a PSF at any point within field-of-view can be well approximated by linear combination (interpolation) of the PSFs sampled at the neighboring grid points. Said so, the shift-variant blurring operation can be written as: $\mathbf{y} = \mathbf{R} \sum_{i=1}^N \mathbf{Z}_i \mathbf{H}_i \mathbf{W}_i \mathbf{C}_i \mathbf{x}$, where \mathbf{C}_i are chopping operators that select overlapping

blocks from the image \mathbf{x} , $\mathbf{W}_i = \text{diag}(\boldsymbol{\omega}_i)$ are interpolation weights corresponding to each blocks, \mathbf{H}_i are blur operators corresponding to the sampled local PSFs \mathbf{h}_i , \mathbf{Z}_i are operators that zero-pads the corresponding blurred blocks keeping it in the same relative position with respect to the whole image, and \mathbf{R} is a chopping operator that restrict the final blurred image to sensor size. Interpolation weights $\boldsymbol{\omega}_i$ are non-zero only within certain locality depending upon the order of interpolation. This approximation allows to perform shift-variant blurring combining several local shift-invariant blurring, which can be performed efficiently in Fourier domain. This approximation renders smooth variation of blur throughout the whole image. Higher accuracy in blurring operation can be achieved by denser sampling of PSFs with the field-of-view. The image deblurring results presented in [6], [8] suggest that the first-order interpolation (e.g., $\boldsymbol{\omega}_i$ is a ramp within the range $[0, 1]$ between the two adjacent grid points for 1D signal as illustrated in Fig.1) is sufficient to achieve reasonably good quality of deblurred images. Using first-order interpolation, the shift-variant blur operator is only four times expensive than the shift-invariant blur operation on the same size of image. However, this fast approximation of shift-variant blur operator is efficient only for a centralized multi-threaded shared memory implementation. It is not directly applicable for distributed setting as it will lead to intensive data communication among the nodes each time the blur operator or its adjoint is used by an iterative solver. Nevertheless, this approximation suggests the following idea for distributed image deblurring: i) split the observed image \mathbf{y} into N overlapping blocks \mathbf{y}_i , ii) generate 2D first-order interpolation weights $\boldsymbol{\omega}_i$ corresponding to each block, iii) provided with N PSFs \mathbf{h}_i sampled locally within each block, distribute the observed blocks, the PSFs, and the interpolation weights among the N nodes, and iv) then on each node perform local image deblurring while maintaining certain consensus among the overlapping pixels of the adjacent blocks. The consensus operation can be a weighted averaging among the overlapping pixels. Using the interpolation weights $\boldsymbol{\omega}_i$ for averaging operation will indirectly impose the smooth variation of blur across whole observed image. Without loss of generality, the aforementioned strategy is also applicable to the cases when an image suffers from shift-invariant or piece-wise constant shift-variant blur since these two operations can be well approximated by the above fast shift-variant blur operator for the smooth blur variation.

As pointed out above, performing local deblurring in Fourier domain may lead to boundary artifacts, thus to avoid such artifacts in the deblurred blocks, we borrow the idea from [16]. We express the local blur operator as: $\mathbf{H}_i = \mathbf{C}_i \mathcal{H}_i, i = 1, \dots, N$, where $\mathcal{H}_i \in \mathbb{R}^{n_i \times n_i}$ are circular convolution matrices formed from PSFs \mathbf{h}_i , and $\mathbf{C}_i \in \mathbb{R}^{m_i \times n_i}$ are chopping operators, which restrict the convolution results to the valid regions. Expressing the local blur operators in this form allows an efficient non-circulant blur operation in Fourier domain while suppressing boundary artifacts in the deblurred patches.

Concerning the last issue about approximating regularizer on the whole image by sum of it on blocks, it depends upon the structure of regularizer. If the regularizer is separable in image domain, e.g., $\phi(\mathbf{D}\mathbf{x}) = \|\mathbf{x}\|_1$ or $\|\mathbf{x}\|_2^2$, then the aforesaid

approximation holds. However, for many of the frequently used regularizers, e.g., $\|\mathbf{D}\mathbf{x}\|_2^2$ or $\|\mathbf{D}\mathbf{x}\|_1$ or $\|\mathbf{D}\mathbf{x}\|_2$ where \mathbf{D} represents finite-forward difference operator or some discrete wavelet transform, the aforementioned approximation does not hold. This can lead to incoherencies among the deblurred blocks, which can render nonsmooth transitions among them. However, as shown hereafter a sufficient amount of overlaps among the adjacent blocks with aforementioned consensus imposed on the overlapping pixels during deblurring process will limit the incoherencies, and suppress any nonsmooth transition in the final deblurred image.

III. THE PROPOSED APPROACH

Considering all the ideas developed in Section II, we propose a generic framework for distributed image deblurring applicable to images suffering from both the shift-invariant and shift-variant blur. We make some reasonable approximations, and reformulate the original problem (3) into the distributed optimization problem presented below. The resulting optimization problem is solved in a distributed manner by Douglas-Rachford (D-R) splitting algorithm that first appeared in [26].

A. General Setting

Consider a distributed computing system with a set of N nodes having peer-to-peer bidirectional connections among them, or at least connection topology shown in Fig. 2. Given an observed image $\mathbf{y} \in \mathbb{R}^m$, we split it into N blocks $\mathbf{y}_i \in \mathbb{R}^{m_i}, \forall i = 1, 2, \dots, N$, with certain overlaps among them. We generate 2D first-order interpolation weights ω_i corresponding to the observed blocks as shown in Fig. 3(e). Provided that we are able to sample N PFSs \mathbf{h}_i within the blocks, and $\mathbf{H}_i \in \mathbb{R}^{m_i \times n_i}$ be the corresponding blur operators, we distribute the observed blocks, the corresponding PSFs, and the interpolation weights among the nodes. We, then, seek to distributively estimate the whole unknown crisp image $\mathbf{x} \in \mathbb{R}^n$. Let us denote by $\mathcal{P}_1, \dots, \mathcal{P}_N$ a collection of N subsets of $\{1, \dots, n\}$. For every $i = 1, \dots, N$, we assume that i th compute node is in charge of estimating the components of \mathbf{x} corresponding to the indices \mathcal{P}_i . The subsets $\mathcal{P}_1, \dots, \mathcal{P}_N$ are overlapping. Hence, different nodes handling a common component of \mathbf{x} must eventually agree on the value of the latter. Formally, we introduce the product space $\mathcal{X} := \mathbb{R}^{\mathcal{P}_1} \times \dots \times \mathbb{R}^{\mathcal{P}_N}$, and we denote by \mathcal{C} the set of vectors $(\mathbf{x}_1, \dots, \mathbf{x}_N) \in \mathcal{X}$ satisfying the restricted consensus condition

$$\forall (i, j) \in \{1, \dots, N\}^2, \forall \ell \in \mathcal{P}_i \cap \mathcal{P}_j, \mathbf{x}_i(\ell) = \mathbf{x}_j(\ell).$$

Moreover, we assume that every i th node is provided with a local convex, proper and lower semicontinuous function $f_i : \mathbb{R}^{\mathcal{P}_i} \rightarrow (-\infty, +\infty]$. We consider the following constrained minimization problem on $\mathbb{R}^{\mathcal{P}_1} \times \dots \times \mathbb{R}^{\mathcal{P}_N}$:

$$\arg \min_{\mathbf{x}_1, \dots, \mathbf{x}_N} \sum_{i=1}^N f_i(\mathbf{x}_i) \text{ s.t. } (\mathbf{x}_1, \dots, \mathbf{x}_N) \in \mathcal{C}. \quad (4)$$

For our image deblurring problem, the local cost function f_i is composed of the local data-fidelity term

$$\mathbf{x}_i \mapsto \frac{1}{2} \|\mathbf{y}_i - \mathbf{H}_i \mathbf{x}_i\|_{\mathbf{W}_i}^2,$$

for some positive definite $\mathbf{W}_i(\ell, \ell) = 1/\sigma_i^2(\ell)$ as in (3), and a regularizer $\phi_i(\mathbf{D}_i \mathbf{x}_i)$ with positivity constraint on \mathbf{x}_i , i.e., $\mathbf{x}_i \in \mathbb{R}_+^{\mathcal{P}_i}$. If \mathcal{A} is a set, the notation $\iota_{\mathcal{A}}$ stands for the indicator function of the set \mathcal{A} , equal to zero on that set and to $+\infty$ elsewhere. Thus, the local cost function needed to be minimized at each of the nodes have the form:

$$f_i(\mathbf{x}_i) = \frac{1}{2} \|\mathbf{y}_i - \mathbf{H}_i \mathbf{x}_i\|_{\mathbf{W}_i}^2 + \lambda_i \phi_i(\mathbf{D}_i \mathbf{x}_i) + \iota_{\mathbb{R}_+^{\mathcal{P}_i}}(\mathbf{x}_i)$$

where ϕ_i are convex, proper and lower semicontinuous functions and \mathbf{D}_i are linear operators on $\mathbb{R}^{\mathcal{P}_i}$.

B. Optimization Algorithm

Before we present our distributed optimization algorithm for solving problem (4), let us introduce one more notation. For any convex, proper and lower semicontinuous function $h : \mathcal{X} \rightarrow (-\infty, +\infty]$, we introduce the *proximity operator*

$$\text{prox}_{\mathbf{V}^{-1}, h}(\mathbf{v}) = \arg \min_{\mathbf{w} \in \mathcal{X}} h(\mathbf{w}) + \frac{\|\mathbf{w} - \mathbf{v}\|_{\mathbf{V}}^2}{2}$$

for every $\mathbf{v} \in \mathcal{X}$.

For solving (4), we consider D-R Splitting algorithm, thus we reformulate (4) as

$$\arg \min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) + g(\mathbf{x})$$

where $g = \iota_{\mathcal{C}}$ is the indicator function of \mathcal{C} and $f(\mathbf{x}) = \sum_i f_i(\mathbf{x}_i)$ for every $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$ in \mathcal{X} . Let us equip the Euclidean space \mathcal{X} with the inner product $\langle \cdot, \cdot \rangle_{\mathbf{V}}$ for some positive definite linear operator $\mathbf{V} : \mathcal{X} \rightarrow \mathcal{X}$. Let $\rho^{(k)}$ be a sequence in $]0, 2[$, and $\epsilon_1^{(k)}$ and $\epsilon_2^{(k)}$ be sequences in \mathcal{X} , then the D-R splitting algorithm writes as:

$$\begin{aligned} \mathbf{x}^{(k+1)} &= \text{prox}_{\mathbf{V}^{-1}, f}(\mathbf{u}^{(k)}) + \epsilon_1^{(k)} \\ \mathbf{z}^{(k+1)} &= \text{prox}_{\mathbf{V}^{-1}, g}(2\mathbf{x}^{(k+1)} - \mathbf{u}^{(k)}) + \epsilon_2^{(k)} \\ \mathbf{u}^{(k+1)} &= \mathbf{u}^{(k)} + \rho^{(k)} (\mathbf{z}^{(k+1)} - \mathbf{x}^{(k+1)}) \end{aligned}$$

If the following holds:

- 1) the set of minimizers of (4) is non-empty,
- 2) $\mathbf{0} \in \text{ri}(\text{dom}(f) - \mathcal{C})$, where ri represents relative interior,
- 3) $\sum_k \rho^{(k)}(2 - \rho^{(k)}) = +\infty$,
- 4) and $\sum_k \|\epsilon_1^{(k)}\|_2 + \|\epsilon_2^{(k)}\|_2 < +\infty$,

then the iterates $\mathbf{u}^{(k)}$ converges weakly to some point in \mathcal{X} and $\mathbf{x}^{(k)}$ converge to a minimizer of (4) as $k \rightarrow \infty$; see [27, Corollary 5.2] for the proof. The parameter $\rho^{(k)}$ is referred to as a relaxation factor, which can be tuned to improve the convergence. The sequences $\epsilon_1^{(k)}$ and $\epsilon_2^{(k)}$ allow some perturbations in the two prox operations, which is very useful in the cases when the prox operations do not have closed-form solutions and have to rely on some iterative solvers.

From now onward, we assume that \mathbf{V} is a diagonal operator of the form $\mathbf{V}\mathbf{x} = (\mathbf{V}_1\mathbf{x}_1, \dots, \mathbf{V}_N\mathbf{x}_N)$, where for every i ,

$$\begin{aligned} \mathbf{V}_i \mathbf{x}_i : \mathcal{P}_i &\rightarrow \mathbb{R} \\ \ell &\mapsto \alpha_i(\ell) \mathbf{x}_i(\ell), \end{aligned}$$

where $\alpha_i(\ell)$ is a positive coefficient to be specified later. For every $\ell \in \{1, \dots, n\}$, we introduce the set $\mathcal{P}_\ell^- = \{i : \ell \in \mathcal{P}_i\}$.

Lemma III.1. For every $\mathbf{x} \in \mathcal{X}$, the quantity $\mathbf{z} = \text{prox}_{\mathbf{V}^{-1}, \iota_{\mathcal{C}}}(\mathbf{x})$ is such that for every $i \in \{1, \dots, N\}$ and every $\ell \in \mathcal{P}_i$,

$$\mathbf{z}_i(\ell) = \frac{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell) \mathbf{x}_i(\ell)}{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell)}.$$

Proof: For every $\mathbf{x} \in \mathcal{X}$ of the form $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$, and every $\ell \in \{1, \dots, n\}$, we use the notation $\mathbf{x}_{\mathcal{P}_\ell^-}(\ell) = (\mathbf{x}_i(\ell) : i \in \mathcal{P}_\ell^-)$. We denote by \mathcal{C}_ℓ the linear span of the vector $(1, \dots, 1)$ in $\mathbb{R}^{|\mathcal{P}_\ell^-|}$, where $|\mathcal{P}_\ell^-|$ is the cardinality of \mathcal{P}_ℓ^- . The function $g = \iota_{\mathcal{C}}$ writes

$$g(\mathbf{x}) = \sum_{\ell=1}^n \iota_{\mathcal{C}_\ell}(\mathbf{x}_{\mathcal{P}_\ell^-}(\ell)).$$

For an arbitrary $\mathbf{x} \in \mathcal{X}$, the quantity $\mathbf{z} = \text{prox}_{\mathbf{V}^{-1}, g}(\mathbf{x})$ is given by

$$\begin{aligned} \mathbf{z} = \arg \min_{\mathbf{w} \in \mathcal{X}} \sum_{\ell=1}^n \left(\iota_{\mathcal{C}_\ell}(\mathbf{w}_{\mathcal{P}_\ell^-}(\ell)) \right. \\ \left. + \frac{1}{2} \sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell) (\mathbf{w}_i(\ell) - \mathbf{x}_i(\ell))^2 \right) \end{aligned}$$

Clearly, for every $\ell \in \{1, \dots, n\}$, the components $\mathbf{z}_i(\ell)$ are equal for all $i \in \mathcal{P}_\ell^-$, to some constant w.r.t. i , say $\bar{\mathbf{z}}(\ell)$. The later quantity is given by

$$\begin{aligned} \bar{\mathbf{z}}(\ell) &= \arg \min_{\mathbf{w} \in \mathbb{R}} \sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell) (\mathbf{w} - \mathbf{x}_i(\ell))^2 \\ &= \frac{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell) \mathbf{x}_i(\ell)}{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell)}. \end{aligned}$$

By the above lemma, the D-R splitting algorithm can be explicitly written as follows. For every $i \in \{1, \dots, N\}$ and every $\ell \in \{1, \dots, n\}$,

$$\begin{aligned} \mathbf{x}_i^{(k+1)} &= \text{prox}_{\mathbf{V}_i^{-1}, f_i}(\mathbf{u}_i^{(k)}) \\ \bar{\mathbf{z}}^{(k+1)}(\ell) &= \frac{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell) (2\mathbf{x}_i^{(k+1)}(\ell) - \mathbf{u}_i^{(k)}(\ell))}{\sum_{i \in \mathcal{P}_\ell^-} \alpha_i(\ell)} \\ \mathbf{u}_i^{(k+1)}(\ell) &= \mathbf{u}_i^{(k)}(\ell) + \rho^{(k)} \left(\bar{\mathbf{z}}^{(k+1)}(\ell) - \mathbf{x}_i^{(k+1)}(\ell) \right). \end{aligned}$$

The resulting algorithm is a synchronous distributed algorithm without any explicit master node. It is depicted in Algorithm 1. Hereafter, we will refer to it by *proposed* deblurring method. Given some initial guess of the local solutions at each node, the first step of Algorithm 1 execute the LOCAL-SOLVER in parallel on all the nodes to obtain local deblurred blocks. Then all the nodes synchronize at the start of the second step, and then they exchange the overlapping pixels with its adjacent nodes to distributively perform the weighted averaging of those pixels. Then, the last step of the algorithm is executed in parallel on all the nodes.

The convergence speed of *proposed* deblurring method is dependent upon the parameters α_i , and like in other optimization algorithms e.g., ADMM [28], selecting the optimal values of α_i for fast convergence is a tedious task. We select

Algorithm 1 Distributed Image Deblurring

procedure DISTRIBUTED-SOLVER

Initialize: $\mathbf{u}_i \leftarrow \mathbf{u}_i^{(0)}, \forall i = 1, 2, \dots, N$

while not converged **do**

for $i = 1 \dots N$ **do**

$\mathbf{x}_i \leftarrow \text{LOCAL-SOLVER}(\mathbf{u}_i; \alpha_i, f_i)$

end for

for $\ell = 1 \dots n$ **do**

 Compute distributively at nodes $i \in \mathcal{P}_\ell^-$:

$\bar{\mathbf{z}}_i(\ell) \leftarrow \sum_{i \in \mathcal{P}_\ell^-} \omega_i(\ell) (2\mathbf{x}_i(\ell) - \mathbf{u}_i(\ell)), \forall \ell \in \mathcal{P}_i$

end for

for $i = 1 \dots N$ **do**

$\mathbf{u}_i(\ell) \leftarrow \mathbf{u}_i(\ell) + \rho(\bar{\mathbf{z}}_i(\ell) - \mathbf{x}_i(\ell)), \forall \ell \in \mathcal{P}_i$

end for

end while

return $\mathbf{x}_1, \dots, \mathbf{x}_N$

end procedure

procedure LOCAL-SOLVER($\mathbf{u}; \alpha, f$)

$\mathbf{w} \leftarrow \text{prox}_{\alpha^{-1}, f}(\mathbf{u}) = \arg \min_{\mathbf{w}} \{f(\mathbf{w}) + \frac{1}{2} \|\mathbf{w} - \mathbf{u}\|_\alpha^2\}$

return \mathbf{w}

end procedure

$\alpha_i = \gamma \omega_i$ for $\gamma > 0$, so that we can tune γ for fast convergence, and as well impose the smooth variation of the blur among adjacent blocks.

C. Size of Image Blocks and the Extent of the Overlaps

The size of observed image blocks and the extent of overlaps among them have impact upon the computational cost and the quality of final deblurred image. Thus, they must be chosen appropriately depending upon the situations: shift-invariant, piece-wise constant shift-variant, and smooth shift-variant blur. Without loss of generality, we assume that all local PSFs have the same size.

In the case of smooth shift-variant blur, the size of image blocks and the extent of overlaps depend upon how densely the grid points, thus the local PSFs, can be sampled within the field-of-view. Denser the grid points, better is the approximation of shift-variant blur in the image. If we consider N equidistant grid points within the field-of-view, then the support of image blocks extend (in both horizontal and vertical directions) over three consecutive grid points, except for the image blocks at the boundaries of image. Thus, each block overlaps half of its size from top, bottom, left and right with its adjacent blocks, i.e., a pixel in the observed image appears into four adjacent blocks. As discussed previously, this extent of overlaps are necessary to impose smooth variation of blur among the image blocks. See the illustration in Fig. 1, which demonstrate for 1D signal the scheme of splitting the observed image into blocks, and the extent of overlaps among them. Similarly, Fig. 5(a) shows 5×5 grid points overlaid upon the observed image where the local PSFs are sampled, and Fig. 5(c) shows resulting overlapping image blocks.

In the case of shift-invariant blur, again we can consider N equidistant grid points within the field-of-view, and split the observed image into N overlapping blocks. However, in

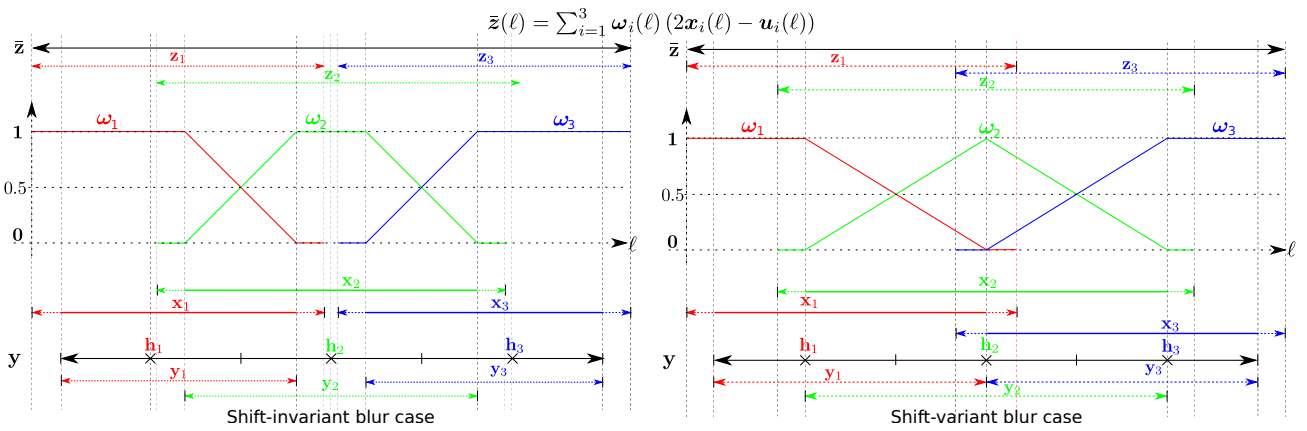


Fig. 1. An 1D illustration showing the splitting of the observed image, the extent of the overlap, and shape of the interpolation weights for the shift-invariant and smooth shift-variant blur cases. Given an observed image \mathbf{y} , the crossbar are equidistant reference grid points where PSFs \mathbf{h}_i are sampled. Depending upon the case: shift-invariant or shift-variant blur, the extent of the overlap is selected. The observed image is split into 3 overlapping patches \mathbf{y}_i . \mathbf{x}_i are the locally deblurred patches at i th node from the corresponding observed patches \mathbf{y}_i . The dashed parts at the ends of each \mathbf{x}_i are the extra pixels estimated at boundaries assuming no measurements were available for them. ω_i are the interpolation weights corresponding to the support of \mathbf{x}_i , and its values are within range $[0, 1]$ such that $\sum_{i=1}^3 \omega_i(\ell) = 1, \forall \ell = 1, \dots, n$.

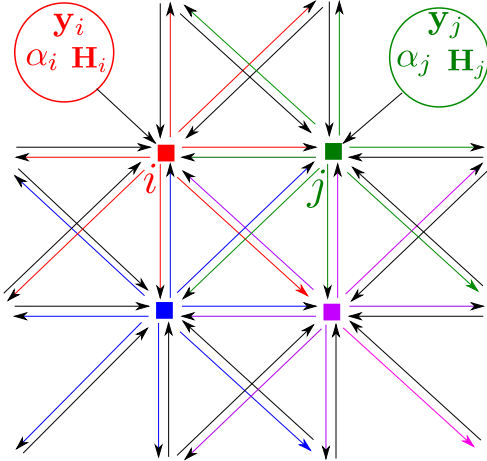


Fig. 2. The communication network topology for our distributed image deblurring algorithm. Each square represents a node connected to its neighbors by bi-directional communication channel. Provided with blurred patches, and the corresponding local PSFs and the interpolation weights, the nodes run simultaneously the LOCAL-SOLVERS, and exchange their local estimate of overlapping pixels to perform the consensus operation.

this case, we do not need the same extent of overlaps among the blocks as in the case of smooth shift-variant blur, but a smaller extent of overlaps. In fact, it is sufficient to have extent of overlaps slightly larger than half the size of the local PSF. In this case, overlaps among the blocks are required not to impose smooth variation of blur (as the PSF same over whole field-of-view), but to minimize any discrepancies in the deblurred image due to approximation introduced in the regularization term. Similarly, in the case of piece-wise constant shift-variant blur, we can consider the same splitting scheme as in the case of shift-invariant blur for the same reason.

In the case of *proposed* deblurring method, the maximum number blocks we can split the image is dependent upon the number of available nodes to process in parallel. In such a

situation, the larger the number of available nodes, the larger the number of blocks we can chose, thus, smaller the size of blocks. From a computational point of view, the smaller the size of the blocks, the smaller the memory and computation time required by nodes to execute the LOCAL-SOLVER³. Moreover, the smaller the size of blocks, the smaller is the extent of overlaps among them, thus, the smaller the amount of data to be exchanged among the nodes. With all these advantages from a computational point-of-view, we may intend to split the observed image into as many blocks as the number of available nodes, given that we are provided with as many local PSFs within the field of view. However, the size of the local PSF suggests a lower limit on the size of blocks. We should not select the size of blocks smaller than the size of the PSF, otherwise the local deblurring problem at each node will be highly underdetermined (more number of unknown pixels at boundaries of the blocks need to be estimated), which would eventually deteriorate the quality of the final deblurred image. Thus, to achieve better quality deblurred image, we should select the size of blocks at least twice the size of PSF.

IV. NUMERICAL EXPERIMENTS AND RESULTS

Our algorithm depicted in Algorithm 1 can be seen as general framework for distributed image deblurring. Depending upon the situation, we can derive a particular instance from it, e.g., shift-invariant or shift-variant deblurring, and depending upon the structures of data-fidelity and regularizer, we can select any efficient optimization method as the LOCAL-SOLVER. In order to validate the *proposed* deblurring method, we performed two numerical experiments, first, considering the simpler case of shift-invariant deblurring, and then more difficult case of the shift-variant deblurring.

³The size of vectors affects the efficiency of Fast Fourier Transform algorithm and other operations. Smaller vector fit well into the different levels of data-caches in processor so that it consumes lesser clock cycles to execute the operations compared to the situation when the vectors are larger to fit into the data-caches leading to many cache misses and eventually requiring several cycles to fetch them from main memory.

Moreover, to evaluate the performance of the *proposed* deblurring method in terms of quality of deblurred image, we compared it with two other possible approaches. First obvious choice was *centralized* deblurring as a reference method that solves the original problem (3). Since *centralized* deblurring method would require a huge memory for a large image, thus we selected images of reasonable sizes for our experiments. We chose the naïve “split, deblur independently, and merge” approach as the second method. In order to minimize incoherencies among the deblurred blocks obtained by the second method, we split the observed image into overlapping blocks, and blended the locally deblurred blocks into single image by weighted averaging of the overlapping regions by using the same 2D first-order interpolation weights as in *proposed* method. Hereafter, we will refer to this approach as *independent* deblurring method. The *independent* deblurring method is not intended to solve correctly either of the optimization problems (3) or (4), but it is a straight forward way to deblur large images with minimal computational resources. Similarly, in the case of shift-variant image deblurring, we consider comparing our algorithm with the *centralized* and *independent* deblurring methods. For the *centralized* deblurring method, we used the fast shift-variant blur operator based on first-order PSF interpolation as described in [8].

Let us first present the general settings of our experiments, and later more specific details. For the case of shift-invariant deblurring, we considered “Lena” image, which we resized to 1024×1024 pixels, and extended its dynamic range linearly to have maximum intensity up to 6000 photons/pixels. We refer to this image as the reference image. The reference image was blurred with an Airy disk PSF (of size 201×201 pixels) formed due to a circular aperture of radius 6 pixels, and was corrupted with white Gaussian noise of variance $\sigma^2 = 400$ photons/pixels to obtain the observed image. To study the impact of factors discussed in Section III-C, we considered different cases by varying number of blocks and the extent of overlaps among them. Figure 3 shows an instance when observed image is split into 3×3 blocks with overlaps of 100×100 pixels among the adjacent blocks.

For the shift-variant deblurring case, we considered “Barbara” image, which was resized it to 1151×1407 pixels, and its dynamic range was extended linearly to have maximum intensity up to 6000 photons/pixels. As above, we call this image as reference image. We generated 9×9 normalized Gaussian PSFs each of size 201×201 pixels with the central PSF having full-width-half-maximum (FWHM) = 3.5×3.5 pixels, and linearly increasing FWHM in the radial direction up to 16.5×10.5 pixels for the PSF at extreme corner of the reference image as depicted in Fig. 4. These PSFs, somehow, mimic the shift-variant blur due to optical aberrations (coma) [29]. We blurred the reference image with this shift-variant PSFs using shift-variant blur operator based on PSF interpolation described in [8]. We obtained the final observed image by adding white Gaussian noise of variance $\sigma^2 = 400$ photons/pixels to the blurred image. As in shift-invariant deblurring case, we considered different scenarios by varying the number of blocks. Figure 5 shows an experimental setup when using only 5×5 grid of PSFs, i.e., when observed

image split is into 5×5 overlapping blocks.

In both the cases, we deliberately selected low level noise in the observed image so that any incoherency or artifact arising in the deblurred image due to the approximations we made was not superseded by the strong regularization level required for low signal-to-noise ratio in observed image. Also, we selected sufficiently large size of PSFs so that they are not band-limited, which is the case in many real imaging systems.

A. Choice of Regularization Term

For our experiments, we selected the regularization function ϕ to be Huber loss and \mathbf{D} to be circular forward finite difference operator, so the regularizer is written as

$$\phi(\mathbf{D}_i \mathbf{x}_i) = \begin{cases} \frac{1}{2} \|\mathbf{D}_i \mathbf{x}_i\|_2^2 & \|\mathbf{D}_i \mathbf{x}_i\|_2 \leq \delta \\ \delta (\|\mathbf{D}_i \mathbf{x}_i\|_2 - \frac{\delta}{2}) & \text{otherwise} \end{cases}$$

We chose this regularizer for two reasons: i) it is smooth (differentiable), which makes the functions f_i in (4) smooth so that we can choose any fast optimization algorithm such as quasi-Newton methods (e.g., BFGS class methods [30]) as the LOCAL-SOLVER, and ii) it behaves in between Tikhonov and the total-variation regularization, depending upon the value of δ . Thus, it is able to preserve the sharp structures in the images while avoiding stair-case artifacts in smoothly varying regions usually rendered by total-variation regularizer [9]. However, our *proposed* deblurring method is not restricted to any particular choice of regularizer; depending upon the application one can choose any regularizer and any fast optimization algorithm for local deblurring problem.

B. Implementation Details

Since the optimization problems arising in all deblurring methods were smooth, thus we chose a quasi-Newton method called limited-memory variable metric with bound-constraint (VMLM-B)⁴ [31]. VMLM-B is a variant of the standard limited-memory BFGS with bound-constraint (LBFGS-B) [32], but with some subtle differences. VMLM-B does not require any manual tuning of parameters to achieve fast convergence (the only parameter step-length at each iteration of VMLM-B is estimated by line-search methods satisfying Wolfe’s conditions). Thus, this left us with only a single parameter γ to be tuned to achieve faster convergence of our algorithm. Moreover, using the same algorithm (VMLM-B) for solving the optimization problems in all three deblurring methods ensured a fair comparison among them in terms of quality of deblurred images and the computational expenses.

In all our experiments, we set $\rho^{(k)} = 1$. After few trials, we found that $\gamma = 0.001$ results in fast convergence of our algorithm. Since D-R splitting algorithm converges even when the prox operations are carried out inexactly, thus to speed-up our algorithm, we allowed the LOCAL-SOLVER to be less accurate during the initial iterations, and then more accurate as the iterations progress [28, Section 3.4.4]. To do so, we allowed VMLM-B to perform 10 inner iterations at

⁴An open source implementation of VMLM-B in C programming is available at <https://github.com/emmt/OptimPack>.

the beginning, and increased it by 10 at every next iteration of main loop. To speed-up further, we did warm-start of the VMLM-B at every iterations of our algorithm by supplying $\mathbf{x}^{(k)}$ as an initial guess for next $\mathbf{x}^{(k+1)}$ estimation. All the results presented below were obtained after 25 iterations of the proposed algorithm. It was observed that 25 iterations were generally sufficient, and more than 25 iterations did not bring any noticeable difference in the results. For the *centralized* and *independent* deblurring methods, we allowed VMLM-B to perform a maximum of 1000 iterations or until it satisfied its own stopping criterion based on gradient convergence or progress in the line-search method. It was noticed that, generally, 500 to 600 iterations were sufficient and further iterations did not bring any noticeable difference.

All the three deblurring methods including the *centralized*, *independent* and the *proposed* were implemented in high-level dynamic programming language “Julia”⁵ [33]. The *proposed* distributed deblurring is implemented using Message Passing Interface (MPI) based on Open MPI library⁶. The source codes for all the demonstrations shown in this paper will be freely available at <https://github.com/mouryarahul/ImageReconstruction>.

C. Results and Discussion

As mentioned above, we conducted two different experiments, one for shift-invariant, and another for smooth shift-variant blurred image. To compare the quality of the deblurred images obtained from all the three methods we considered two image quality metrics: signal-to-noise ratio (SNR), and Structural Similarity Index (SSIM) [34]. In all the cases, we heuristically fixed the regularization parameter $\delta = 100$, and performed deblurring for different values of λ in a sufficiently large range to see if one of the methods produces better quality image for certain range of λ than others. The plots (SNR vs λ and SSIM vs λ) in the Fig. 3(i–j) show the results obtained from shift-invariant deblurring when observed image was split into 3×3 blocks. To study the influence of size of blocks and extent of overlaps on the quality of deblurred image, we conducted several trails with different settings. The plots in the Fig. 3(k–l) show the influence of extent of overlaps on the image quality. The plots shows that our algorithm performed slightly better than the *centralized* deblurring in terms of both the SNR and SSIM. This could be due to the approximation introduced in the regularization term in the *proposed* deblurring method, e.g., sum of the regularizer on the blocks may be favorable than the regularizer on whole for some images depending upon the contents in the image. Moreover, the *proposed* deblurring may also be benefited by the explicit overlaps among the adjacent blocks; each node can result into slightly different values of the overlapping pixels, which eventually leads to better estimate of those pixels after averaging the estimates from different nodes. This is indicated by the fact that there is slight increase in both the SNR and SSIM with increase in the extent of overlaps as seen in the plots Fig. 3(k–l). It also indicates that an overlap

equal to half the size of PSF is sufficient, and a larger overlap did not produce much improvement, but, of course, it did increase the computational and communication cost. We also noticed that the SNR and SSIM of deblurred image obtained from *proposed* deblurring is slightly less dependent upon the extent of overlaps compared to that obtained by *independent* deblurring.

The plots in Fig. 6(a–b) compare the results obtained by the three deblurring methods for the case of smooth shift-variant blurred image. First, we noticed that the image quality obtained by all the three methods improves drastically with the increase in density of grid of PSFs sampled within field-of-view. The SNR and SSIM of deblurred image obtained using only 3×3 PSFs is significantly lower than the cases when 5×5 , 6×6 and 8×8 grids of PSFs are used. This is due to the fact that the observed image was simulated using a finer 9×9 grids of PSFs, and the coarser grids of PSFs are less accurate in capturing smooth variation of blur than a finer grid of PSFs. We also noticed that unlike the shift-invariant deblurring case, the values of SNR and SSIM obtained by the *proposed* deblurring is slightly lower than that obtained by the *centralize* deblurring. This could be due to the fact that both the *centralized* and *proposed* deblurring benefits from the explicit overlaps among the block, but some information is always lost at the boundaries of the deblurred blocks in the case of *proposed* deblurring, which is not the case for *centralized* deblurring. Also, there must be some influence of the approximation introduced into the regularization term for the case of *proposed* deblurring. We observed from the plots in Fig. 3(i–j) and Fig. 6(a–b) that three methods do not attained the highest SNR or SSIM at the same value of the regularization parameter λ .

As expected, we observed that the naïve *independent* deblurring method performed significantly lower than the other two methods. As pointed out above, this is due to the fact that the method is not intended to solve correctly either the original problem (3) or the distributed formulation (4), but it is the crudest and computationally cheapest way to perform image deblurring by splitting image into pieces.

In our experiments, we observed that the first step, i.e., the LOCAL-SOLVER, of the *proposed* deblurring algorithm is the one which consumed the significant computation time among the three steps; LOCAL-SOLVER took 600 to 800 times more computation time than the consensus step (including communication time among the nodes). This should be true, in general, for many other local deblurring algorithms devised using different combination of data-fidelity and regularization terms depending upon the applications. Thus, the *proposed* deblurring algorithm is efficient in the sense that it is computation intensive rather than being communication intensive.

For the small or moderate size of images as we considered in our experiments, the *proposed* deblurring algorithm is computationally more expensive than the *centralized* and *independent* deblurring methods; it takes at least 10 to 15 times more computation time than the latter ones. However, in the case of extremely large images for which *centralized* deblurring is practically not feasible, computational expenses of *proposed* deblurring is justified by the better quality of deblurred image,

⁵<http://julialang.org/>

⁶<https://www.open-mpi.org/>

which cannot be achieved by the computationally cheaper *independent* deblurring. Some more simulations performed with different set of images and PSFs are presented in Appendix A, and the results suggest similar conclusions as above.

V. CONCLUSION

In this paper we have proposed a distributed image deblurring algorithm for large images for which the existing *centralized* deblurring methods are practically inapplicable due to requirement of huge physical memory on a single system. The proposed algorithm is rather a generic framework for distributed image deblurring for it can handle different imaging situations such as deblurring a single large image suffering from shift-invariant blur, a wide field-of-view captured into multiple narrow images by different imaging systems with slightly different PSFs, and a large image of a wide field-of-view suffering from smoothly varying blur captured by a single imaging system. Depending upon the application, one can easily adapt it to include different data-fidelity and regularization terms, and then select any fast optimization algorithm to solve the local deblurring problem at the different nodes of a distributed computing system. Our algorithm is efficient in the sense that it is computation intensive rather than being communication intensive. We showed by experimental results on simulated observed images that the *proposed* deblurring algorithm produce almost similar or little lower quality (measured in term of SNR and SSIM) of deblurred images than that obtained by *centralized* deblurring. But, this small compromise in the quality of deblurred image is trade-off by the cost effectiveness of our distributed approach for large images, which is practically not feasible for the *centralized* deblurring methods. Moreover, we compared the *proposed* deblurring to a naïve and computationally cheaper *independent* deblurring, and showed the latter always performed significantly lower than the former. Thus, when high accuracy is desirable, e.g., in astronomical application, the *proposed* deblurring should be preferred over *independent* deblurring method, of course at expense of extra computational cost.

In this paper, we considered nonblind image deblurring, i.e., the PSFs were known a priori, however, in real imaging scenario calibrating PSFs accurately is a tedious and challenging task. A more practical way would be to follow a blind image deblurring approach which is able estimate the PSFs from the observed image(s), and recover the single crisp image. Thus, the next perspective step would be to extend the *proposed* algorithm toward distributed blind image deblurring method which should be able to estimate distributively the PSFs imposing certain regularity among them, and eventually estimate the unknown crisp image.

REFERENCES

- [1] D. Titterton, "General structure of regularization procedures in image reconstruction," *Astronomy and Astrophysics*, vol. 144, p. 381, 1985.
- [2] G. Demoment, "Image reconstruction and restoration: Overview of common estimation structures and problems," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 2024–2036, 1989.
- [3] W. H. Richardson, "Bayesian-based iterative method of image restoration," *JOSA*, vol. 62, no. 1, pp. 55–59, 1972.
- [4] T. Porter and T. Duff, "Compositing digital images," in *ACM Siggraph Computer Graphics*, vol. 18, no. 3. ACM, 1984, pp. 253–259.
- [5] J. G. Nagy and D. P. O'Leary, "Restoring images degraded by spatially variant blur," *SIAM Journal on Scientific Computing*, vol. 19, no. 4, pp. 1063–1082, 1998.
- [6] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling, "Efficient filter flow for space-variant multiframe blind deconvolution," in *Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 607–614.
- [7] L. Denis, E. Thiébaud, and F. Soulez, "Fast model of space-variant blurring and its application to deconvolution in astronomy," in *IEEE ICIP*, Brussels, 2011, pp. 2817 – 2820.
- [8] L. Denis, E. Thiébaud, F. Soulez, J.-M. Becker, and R. Mourya, "Fast Approximations of Shift-Variant Blur," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 253–278, 2015.
- [9] L. M. Mugnier, T. Fusco, and J.-M. Conan, "Mistral: a myopic edge-preserving image restoration method, with application to astronomical adaptive-optics-corrected long-exposure images," *JOSA A*, vol. 21, no. 10, pp. 1841–1854, 2004.
- [10] H. Lanteri and C. Theys, "Restoration of Astrophysical Images The Case of Poisson Data with Additive Gaussian noise," *EURASIP Journal on Applied Signal Processing*, vol. 15, pp. 2500–2513, 2005.
- [11] F. Benvenuto, A. L. Camera, C. Theys, a. Ferrari, H. Lanteri, and M. Bertero, "The study of an iterative method for the reconstruction of images corrupted by Poisson and Gaussian noise," *Inverse Problems*, vol. 24, no. 3, p. 035016, jun 2008.
- [12] E. Chouzenoux, A. Jezierska, J.-c. Pesquet, and H. Talbot, "A Convex Approach for Image Restoration with Exact Poisson-Gaussian Likelihood," *SIAM J. Imaging Science*, vol. 8, no. 4, pp. 2662–2682, 2015.
- [13] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947–968, 2007.
- [14] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Trans. on image processing*, vol. 18, no. 11, pp. 2419–34, 2009.
- [15] M. V. Afonso, J. M. Bioucas-Dias, and M. a. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. on image processing*, vol. 19, no. 9, pp. 2345–2356, 2010.
- [16] A. Matakos, S. Ramani, and J. A. Fessler, "Accelerated edge-preserving image restoration without boundary artifacts," *IEEE Trans. on Image Processing*, vol. 22, no. 5, pp. 2019–2029, 2013.
- [17] R. Mourya, L. Denis, J.-M. Becker, and E. Thiébaud, "Augmented lagrangian without alternating directions: Practical algorithms for inverse problems in imaging," in *IEEE ICIP*, Quebec, 2015, pp. 1205–1209.
- [18] E. Thiébaud, "Introduction to image reconstruction and inverse problems," *Optics in Astrophysics*, vol. 198, pp. 1–26, 2005.
- [19] P. C. Hansen, J. G. Nagy, and D. P. O'leary, *Deblurring images: matrices, spectra, and filtering*. SIAM, 2006, vol. 3.
- [20] M. A. Figueiredo and J. M. Bioucas-Dias, "Deconvolution of poissonian images using variable splitting and augmented lagrangian optimization," in *SSP, IEEE/SP 15th Workshop on*. IEEE, 2009, pp. 733–736.
- [21] R. Bekkerman, M. Bilenko, and J. Langford, *Scaling up machine learning: Parallel and distributed approaches*. Cambridge University Press, 2011.
- [22] J. Cui, G. Pratz, B. Meng, and C. S. Levin, "Distributed mlem: An iterative tomographic image reconstruction algorithm for distributed memory architectures," *IEEE Trans. on medical imaging*, vol. 32, no. 5, pp. 957–967, 2013.
- [23] L. Wang, A. Cai, H. Zhang, B. Yan, L. Li, and G. Hu, "Distributed reconstruction via alternating direction method," *Computational and mathematical methods in medicine*, vol. 2013, 2013.
- [24] A. Ferrari, D. Mary, R. Flamary, and C. Richard, "Distributed image reconstruction for very large arrays in radio astronomy," in *2014 IEEE 8th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 2014, pp. 389–392.
- [25] C. Meillier, P. Bianchi, and W. Hachem, "Two distributed algorithms for the deconvolution of large radio-interferometric multispectral images," in *EUSIPCO*, Budapest, Hungary, 2016, pp. 728–732.
- [26] P. L. Lions and B. Mercier, "Splitting Algorithms for the Sum of Two Nonlinear Operators," *SIAM Journal of Numerical Analysis*, vol. 16, no. 6, pp. 964–979, 1979.
- [27] P. L. Combettes, "Solving monotone inclusions via compositions of nonexpansive averaged operators," *Optimization*, vol. 53, no. 5-6, pp. 475–504, 2004.
- [28] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Journal of Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 001–122, 2011.

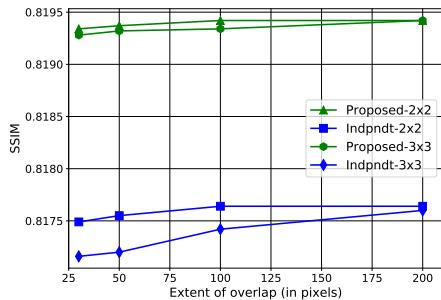
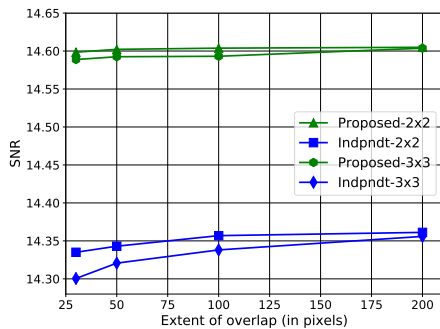
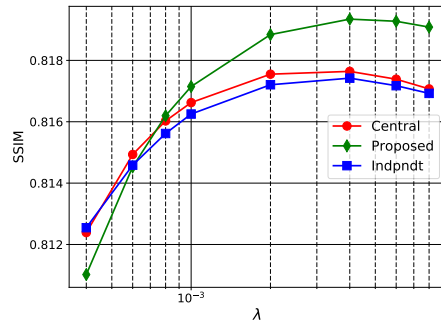
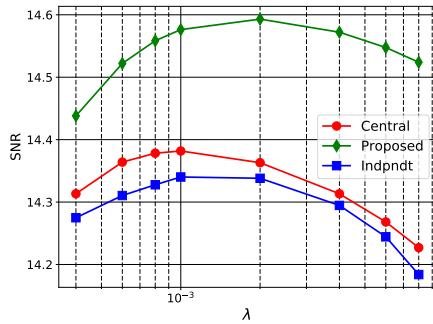
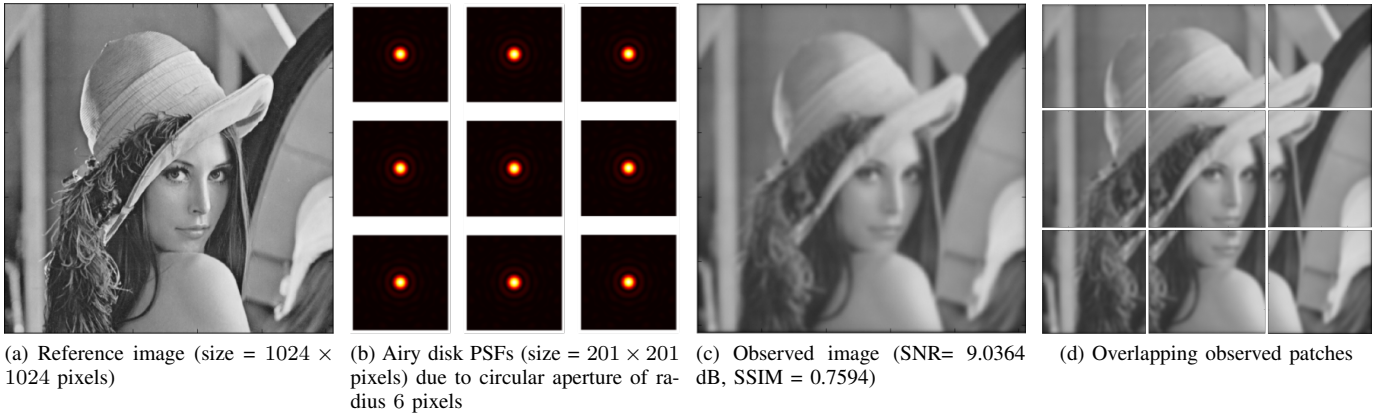
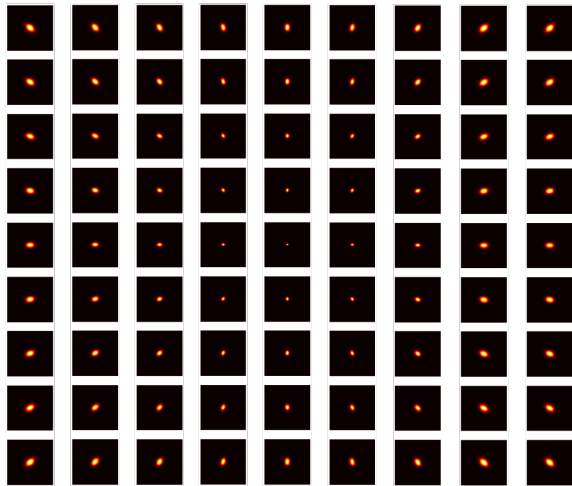


Fig. 3. Experiment 1: experimental setup and results from shift-invariant deblurring of “Lena” image. The observed image (c) is obtained by blurring the reference image (a) with shift-invariant PSF (b), and then corrupting with white Gaussian noise of variance $\sigma^2 = 400$. Image blocks (d) are obtained by splitting observed image into 3 × 3 blocks with overlaps of 100 × 100 pixels among them. The 2D first-order interpolation weights (e) are of the same size as observed blocks. Plots (i–j) show the image quality of deblurred images obtained for different strengths of regularization. The legends “Central”, “Proposed”, and “Indpndt” represent results from the *centralized*, the *proposed* and the *independent* deblurring methods, respectively. Plots (k–l) show impact of extent of overlap on the image quality of deblurred images. The legends “Proposed-2x2” and “Indpndt-2x2” represent the *proposed* and *independent* deblurring methods for the case when the image is split into 2 × 2 blocks. Similarly, the other two legends represent the case when the image is split into 3 × 3 blocks.



(a) Reference image (1151×1407 pixels) with 9×9 grid points (overlaid in green) where PSFs are sampled.



(b) Shift-variant PSFs (each of size 201×201 pixels) generated at the 9×9 grid points in (a).

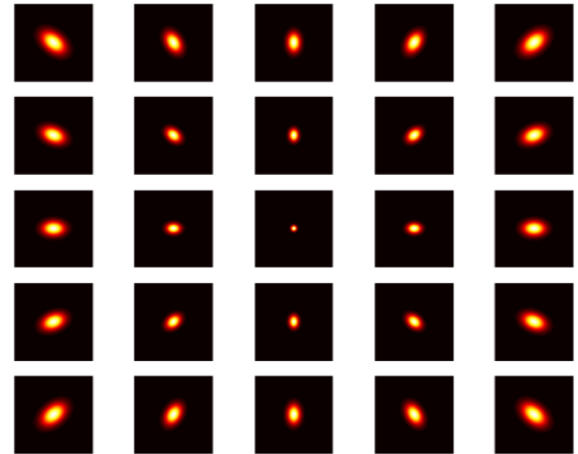


(c) Blurred and Noisy (observed) image

Fig. 4. Experiment 2: experimental setup for shift-variant deblurring of “Barbara” image. The grid of PSFs (b) contains normalized Gaussian PSFs with central PSF having $\text{FWHM} = 3.5 \times 3.5$ pixels, and linearly increased up to $\text{FWHM} = 16.5 \times 10.5$ pixels for the extreme corner PSF. The observed image (c) is obtained by blurring the reference image (a) with the shift-variant PSFs, and then corrupting white Gaussian noise of variance $\sigma^2 = 400$ photons/pixels.



(a) 5×5 grid points overlaid upon the observed image Fig. 4(c) where the PSFs are sampled.

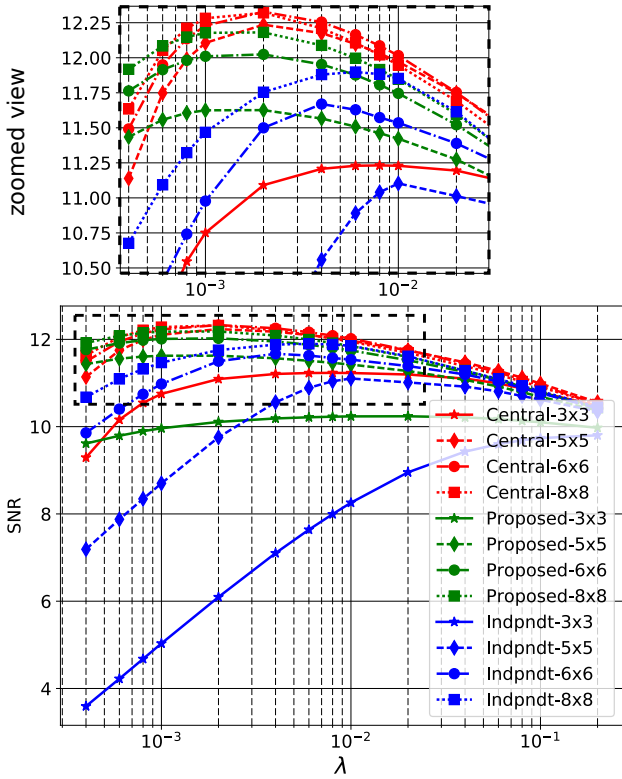


(b) Shift-variant PSFs (each of size 201×201 pixels) sampled at 5×5 grid points shown in (a).

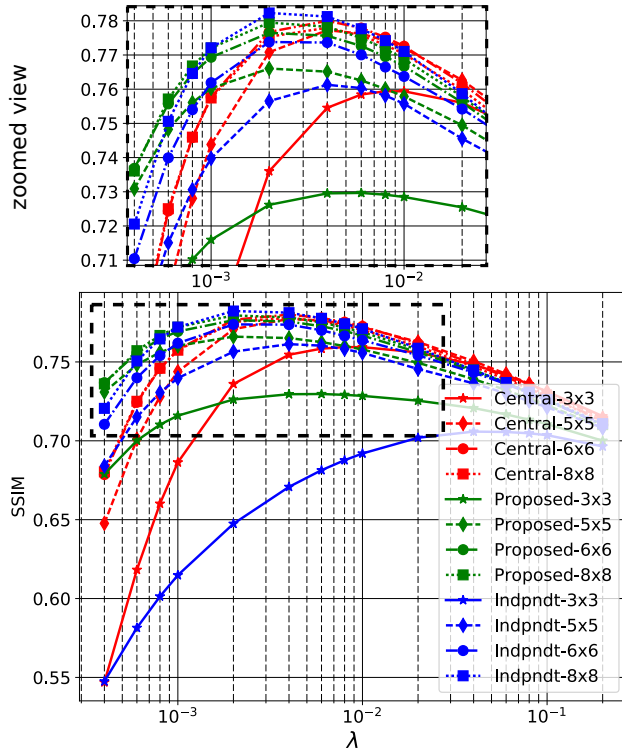


(c) 5×5 overlapping observed blocks obtained after splitting image in (a).

Fig. 5. Experiment 2: experimental setup for shift-variant deblurring when using 5×5 grid of PSFs, i.e., observed image is split into 5×5 overlapping blocks.



(a) SNR vs λ



(b) SSIM vs λ

Fig. 6. Experiment 2: results from shift-variant image deblurring comparing the image quality (in terms of SNR and SSIM) obtained by the three different deblurring methods for different strength of regularization. The legends “Central3x3”, “Proposed-3x3”, “Indpndt-3x3” denotes the results from the *centralized*, the *proposed*, and *independent* deblurring methods, respectively, when using only 3×3 grid of PSFs. Similarly, other legends denotes for the results obtained when using 5×5 , 6×6 and 8×8 grid of PSFs sampled in the field-of-view.



(a) Estimated by *centralized* deblurring (SNR = 12.3278 dB, SSIM = 0.7767 at $\lambda = 0.002$).



(b) Estimated by *independent* deblurring (SNR = 11.6696 dB, SSIM = 0.7736 at $\lambda = 0.004$).



(c) Estimated by *proposed* deblurring (SNR = 12.0239 dB, SSIM = 0.7764 at $\lambda = 0.002$).

Fig. 7. Experiment 2: deblurred images obtained by the three methods when using 6×6 grid of PSFs.

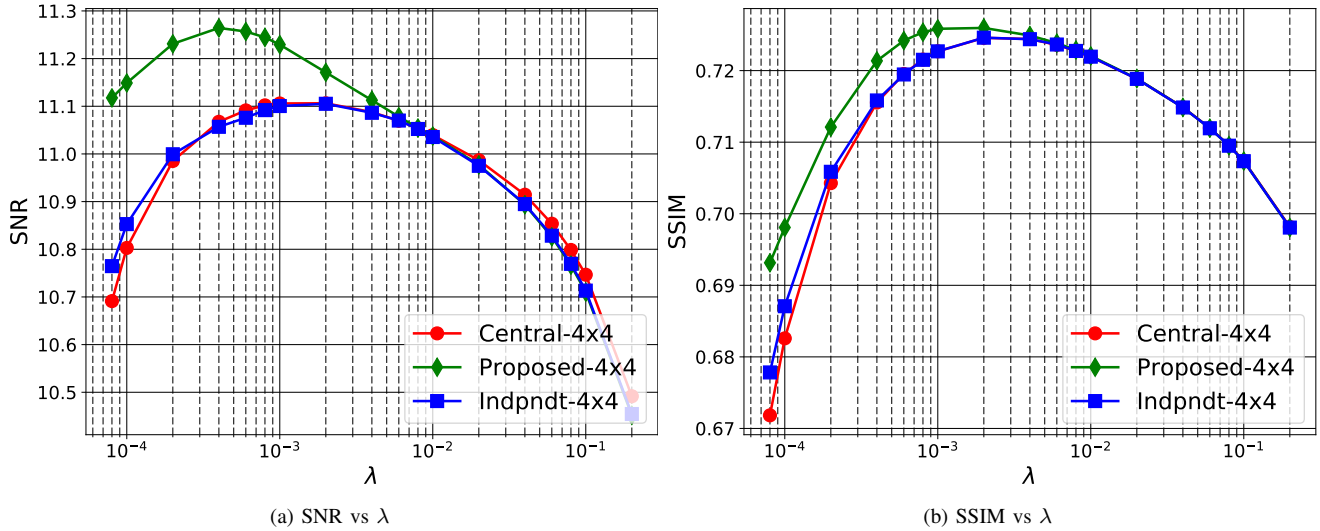


Fig. 8. Experiment 3: results from shift-invariant deblurring of “Barbara” image. The *independent* and *proposed* deblurring methods used 4×4 blocks with overlap of 100×100 pixels among them. The observed image had SNR = 8.7269 dB and SSIM = 0.6399.

[29] V. N. Mahajan, *Optical Imaging And Aberrations-Part-II Wave Diffraction Optics*, 2nd ed. Bellingham, Washington USA: SPIE Press, 2011.

[30] J. Nocedal and S. J. Wright, *Numerical Optimization*, 1st ed. Springer-Verlag New York, 2006.

[31] E. Thiébaud, “Optimization issues in blind deconvolution algorithms,” in *Astronomical Telescopes and Instrumentation*. International Society for Optics and Photonics, 2002, pp. 174–183.

[32] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, “A Limited Memory Algorithm for Bound Constrained Optimization,” *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.

[33] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, “Julia: A fast dynamic language for technical computing,” *arXiv preprint arXiv:1209.5145*, 2012.

[34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

APPENDIX
SOME MORE SIMULATION RESULTS

We repeated the above experiments for different set of images and PSFs, what we referred to as Experiment 3 and 4. This time we chose PSFs that caused less blurriness in the observed images than the previous experiments. For shift-invariant image deblurring, we chose “Barbara” image, and a Airy disk PSF of size 201×201 pixels formed due to a circular aperture of radius 9.5 pixels. For smooth shift-variant image deblurring, we chose “Pentagon” image of size 2048×2048 pixels, and a grid of shift-variant normalized Gaussian PSFs of size 201×201 pixels with central PSF having FWHM = 3.5×3.5 pixels and linearly increasing FWHM in the radial direction up to 8.5×6.5 pixels for the PSF at extreme corner of the reference image. In both the cases, they dynamic range of the reference image was extended linearly up to 6000 photons/pixels and the noise in the observed images was white Gaussian with variance $\sigma^2 = 400$ photons/pixels. The results from both new experiments are shown in Fig. 8 and Fig. 10. Again, we have very similar conclusions from this repeated experiments as from the previous experiments.

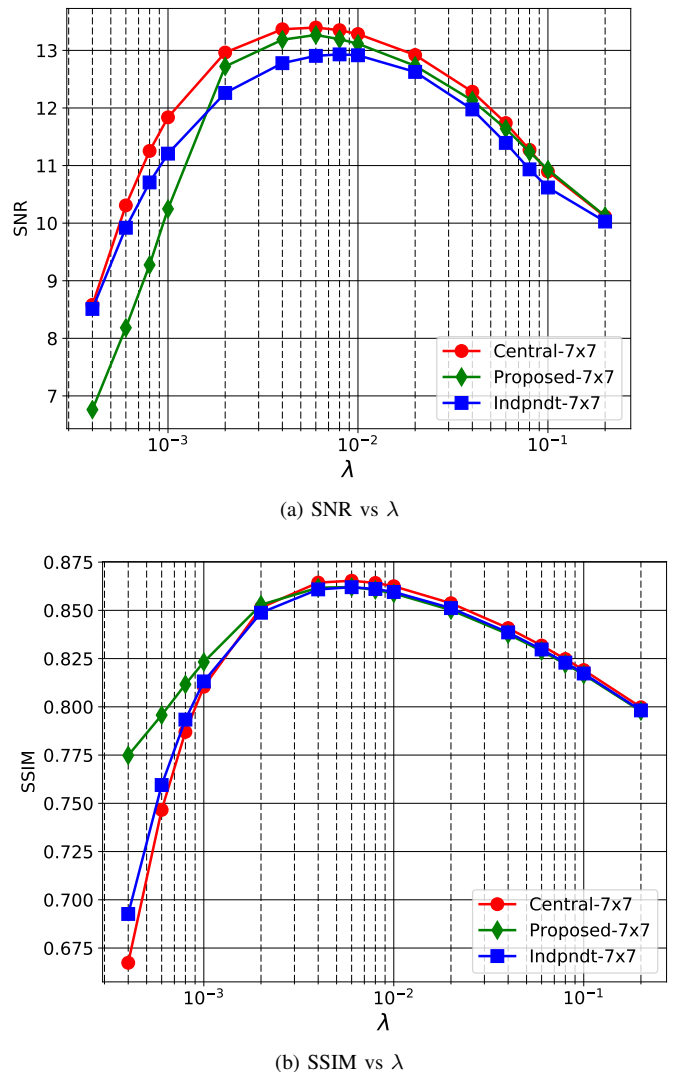


Fig. 9. Experiment 4: results from smooth shift-variant deblurring of “Pentagon” image shown below in Fig. 10.

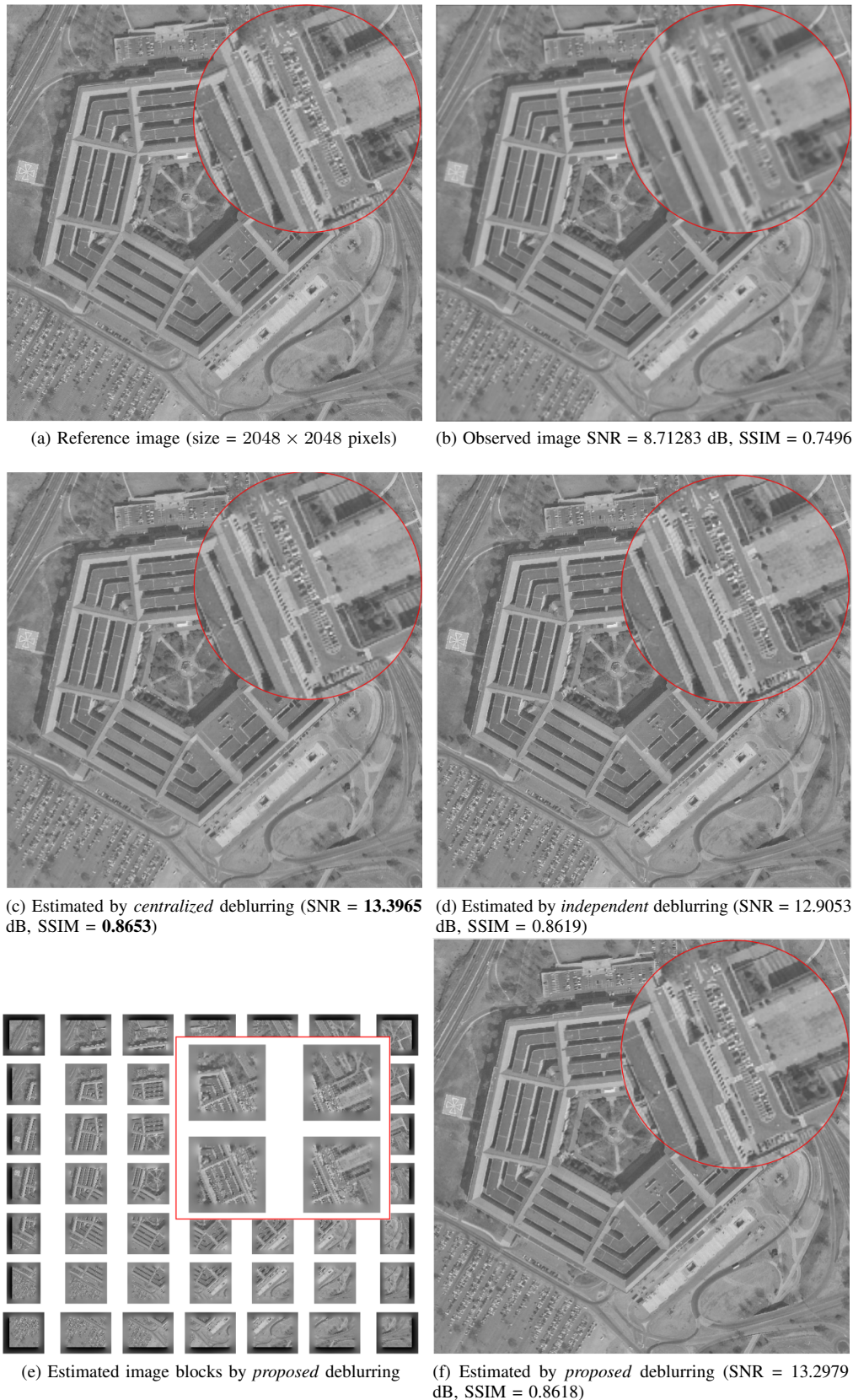


Fig. 10. Experiment 4: experimental setup and results from smooth shift-variant deblurring of “Pentagon” image. The observed image is generate by blurring with a 9×9 grid of shift-variant normalized Gaussian PSFs having FWHM= 3.5×3.5 pixels in the center and linearly increasing FWHM in the radial direction up to 8.56.5 pixels for the PSF at extreme corner. The deblurred image are estimated using only 7×7 grid of PSFs sampled within the field-of-view.



Rahul Mourya Biography text here.



André Ferrari Biography text here.



Rémi Flamary Biography text here.



Pascal Bianchi Biography text here.



Cédric Richard Biography text here.