Technical Report

TR-2005-018

Numerical Methods for Coupled Super-Resolution

by

Julianne Chung, Eldad Haber, James Nagy

Mathematics and Computer Science

EMORY UNIVERSITY

Numerical Methods for Coupled Super-Resolution

Julianne Chung* Eldad Haber† James Nagy‡

December 15, 2005

Abstract

The process of combining, via mathematical software tools, a set of low resolution images into a single high resolution image is often referred to as *super-resolution*. Algorithms for super-resolution involve two key steps: registration and reconstruction. Most approaches proposed in the literature decouple these steps, solving each independently. This can be effective if there are very simple, linear displacements between the low resolution images. However, for more complex, nonlinear, nonuniform transformations, estimating the displacements can be very difficult, leading to severe inaccuracies in the reconstructed high resolution image. This paper presents a mathematical framework and optimization algorithms that can be used to jointly estimate these quantities. Efficient implementation details are considered, and numerical experiments are provided to illustrate the effectiveness of our approach.

*Mathematics and Computer Science, Emory University, Atlanta, GA 30322. jmchung@emory.edu.

†Mathematics and Computer Science, Emory University, Atlanta, GA 30322. haber@mathcs.emory.edu. Research supported in part by the DOE under grant DE-FG02-05ER25696, and the NSF under grant CCF-0427094.

‡Mathematics and Computer Science, Emory University, Atlanta, GA 30322. nagy@mathcs.emory.edu. Research supported in part by the NSF under grant DMS-05-11454 and by an Emory University Research Committee grant.

1 Introduction

Using images for analytic, diagnostic and other purposes is an integral part of research in engineering, medicine, and the sciences. In most cases it is desirable to have images with high spatial resolution. One approach to obtain such images is to build sophisticated instrumentation having intrinsically high resolution capabilities. In addition to being costly, other limitations are difficult to overcome. For example, reducing the pixel sensor size decreases the signal to noise ratio and also results in a build up of shot noise [16]. An alternative, less expensive approach which has gained popularity in digital imaging and video applications, is to use mathematical software tools to combine the information given by a set of low resolution images into one high resolution image. This process is commonly referred to as *super-resolution* [11].

In order for super-resolution techniques to work, the multiple low resolution images must contain different information of the same scene. This is typically accomplished by capturing low resolution images of slightly shifted versions of the scene, with the shifts occurring at sub-pixel distances. For efficient implementation, algorithms often assume uniform, linear sub-pixel shifts. Another approach that has been proposed is to use low resolution images of a stationary scene, but where each image has different amounts of defocus [5]. The idea of achieving super-resolution dates back to the early 1970s [1], but most substantial work on algorithms has been done more recently; see for example [3, 4, 5, 13, 14, 17]. Recent overview papers on super-resolution include [6, 16].

There are two main ingredients in standard super-resolution algorithms[1]. First, one has to estimate the relative displacement or deformation of each point in each image from each point of a reference image. Second, after the displacements have been evaluated, a linear inverse problem (which is equivalent to an image restoration problem) must be solved to obtain the high resolution image. Most approaches proposed in the literature decouple these steps. Decoupling makes sense if relative displacements are known a-priori (possibly from a calibration process) or, if they can be estimated from the low resolution images [3, 5, 6, 13, 14, 16]. For simple displacements, such as linear uniform translation, this procedure can work well. However,

[1]By *standard* super-resolution we mean that low resolution images are captured of slightly shifted scenes. An approach such as that suggested in [5] where the images are of a stationary scene we refer to as *non* standard.

for more complex, nonlinear, nonuniform transformations, estimating the displacement on the coarse image can be very difficult. In particular, as we illustrate in Section 4, because the estimation of displacements is done on the coarse image, fine-scale details of the high resolution image are ignored, which can lead to severe inaccuracies in the reconstructed image.

Clearly the registration and reconstruction parts of the problem are not independent. It should be possible to obtain better results by considering a coupled approach that jointly estimates the displacements and reconstructed high resolution image. Although substantial work has been done in the development of algorithms for the uncoupled super-resolution problem, relatively little work has been done for the coupled problem, which requires solving a nonlinear optimization problem. This can be very expensive to implement unless one makes other simplifying assumptions; as pointed out in [6], several difficulties related to the joint estimation task still remain largely open. Tom and Katsaggelos [17] use a maximum likelihood formulation, and expectation maximization algorithm (EM) to solve the joint estimation problem. They implement the algorithm in the frequency domain, implying spatial invariance, and hence linear uniform displacements. Cheeseman et al. [4] use the maximum a-posteriori (MAP) framework, with very naïve numerical methods, such as the simplex algorithm to solve a least squares problem, and the Jacobi algorithm to solve a system of linear equations. Hardie, Barnard and Armstrong [9] also use a MAP framework. They consider only simple horizontal and vertical displacements, and a numerical optimization approach that alternates between the two sets of variables.

In this paper we present a new mathematical framework that enables us to couple the problem of estimating the displacements with the problem of estimating the high resolution image. To solve the problem we consider three approaches. The first is a completely coupled optimization algorithm that uses a Gauss-Newton type method to estimate the two sets of unknown parameters. In the second, partially coupled approach, we mathematically eliminate one set of parameters to obtain a reduced cost functional. We again use a Gauss-Newton type method for this optimization problem, which is a slight modification of the methods proposed by Golub and Pereyra [7] for separable nonlinear least squares problems, and is similar to the approach used by Vogel, Chan and Plemmons [18] for phase diversity blind deconvolution. The third, completely decoupled approach, uses the simple idea of alternating between minimization of the two sets of variables, similar to the approach used in [9]. We show how to implement these algorithms effi-

ciently for the super-resolution problem with non-trivial displacements, and that the coupled methods produce results superior to those of the decoupled approaches.

This paper is organized as follows. In Section 2 we set up notation, and provide a mathematical framework for super-resolution with arbitrary displacements. In Section 3 we present the algorithms that enable us to solve the resulting optimization problem. Finally, in Section 4 we provide numerical experiments that demonstrate the advantages of our approach.

2 Mathematical Framework

In this section we describe a mathematical model of the super-resolution problem. Assume that the true image scene can be represented as a piecewise smooth function, $f : \mathcal{R}^2 \rightarrow \mathcal{R}^1$. In most imaging applications it is not possible to know f precisely, but instead we have only a discrete image of pixel values. For high resolution, fine grid approximations, one might assume that the pixel values of the discrete image are simply samples of the function, $f(x)$, where $x = (x_1, x_2)$. In this case we use lexicographical ordering to define a vector, $\mathbf{f}$, containing these pixel values; that is,

$$\mathbf{f} = [\, \mathrm{f}_i \,]_{i=1}^{n}$$

where $\mathrm{f}_i = f(x_i)$.[2] For low resolution, coarse grid approximations, a more realistic model of the discrete image is given by the vector $\mathbf{d}$,

$$\mathbf{d} = [\, \mathrm{d}_i \,]_{i=1}^{m}$$

where $m < n$, and

$$\mathrm{d}_i = \int_{\Omega} \mathcal{K}(x_i, y) f(y) dy \,. \tag{1}$$

In this model d_i represents a discrete pixel value that is obtained by averaging over a set of $f(x)$ values. The averaging process is defined by the kernel, $\mathcal{K}(x, y)$. Note that with this model we could also assume that $\mathcal{K}(x, y)$ additionally models blurring, such as defocus, but for ease of presentation we consider only the averaging involved in the discretization process.

[2]Note that each $x_i \in \mathcal{R}^2$, so if we want to refer to specific components of x_i we use the notation $x_i = (x_{i,1}, x_{i,2})$.

A complete discrete model of the image formation process that relates a coarse grid approximation of the image scene to a fine grid approximation can be obtained by replacing the integration in (1) with a quadrature rule. Any quadrature scheme can be used; typically something simple, like the midpoint rule, is sufficient. However, in order that the quadrature approximation results in a fine grid approximation of $\mathbf{f}$, it is necessary to choose the number of quadrature points, n, sufficiently larger than m, such as $n = 2^\ell m$, where ℓ is an integer greater than or equal to one. The result is a linear system

$$\mathbf{d} = \boldsymbol{K}\mathbf{f}\,,$$

where $\mathbf{f}$ is a vector of length n representing the high resolution (fine grid approximation) image, $\mathbf{d}$ is a vector of length m representing a low resolution (coarse grid approximation) image, and $\boldsymbol{K}$ is an $m \times n$ sparse matrix approximating the averaging process.

In a super-resolution problem we must collect a set of discrete images, typically with a fixed (low) resolution. Each observation must provide different information about the true image scene, which can be accomplished, for example, by viewing the scene from different locations, or, equivalently, by moving the objects in the scene before collecting the image data. To describe this mathematically, let $f(x)$ represent a particular true image scene. Then we can represent movements, or deformations of $f(x)$ through a set of displacements, $u_j(x) = [u_{j,1}, u_{j,2}]^\top$; that is, a deformed image scene, $f_j(x)$ is given by

$$f_j(x) = f(x + u_j(x))\,, \quad j = 1, \ldots, k. \tag{2}$$

Assume that the goal is to compute $\mathbf{f}$, which is the vector that represents the fine grid discretization of the true scene, $f(x)$. We would like to write vectors $\mathbf{f}_j$, which are the discrete versions of $f_j(x)$, in terms of $\mathbf{f}$. The difficulty is that we cannot assume that $f_j(x_i) = f(x_i + u_j(x_i))$ falls precisely on one of the discrete pixel values given by the vectors $\mathbf{f}$, and so we must approximate the value. To do this we use bilinear interpolation, which connects any point, $f(x_i + u_j)$, on the displaced image to four pixel values in the reference image that surround it. Specifically, suppose f^{NE}, f^{NW}, f^{SE}, f^{SW} are four pixel values that surround $f(x_i + u_j)$, as illustrated in Figure 1. Then, assuming without loss of generality a grid size of 1, the interpolated point is written as

$$\begin{aligned} f(x_i + u_j) \;=\; & (1 - u_{j,1})(1 - u_{j,2})\mathrm{f}^{NW} + u_{j,1}(1 - u_{j,2})\mathrm{f}^{NE} + \\ & (1 - u_{j,1})u_{j,2}\mathrm{f}^{SW} + u_{j,1}u_{j,2}\mathrm{f}^{SE}. \end{aligned} \tag{3}$$

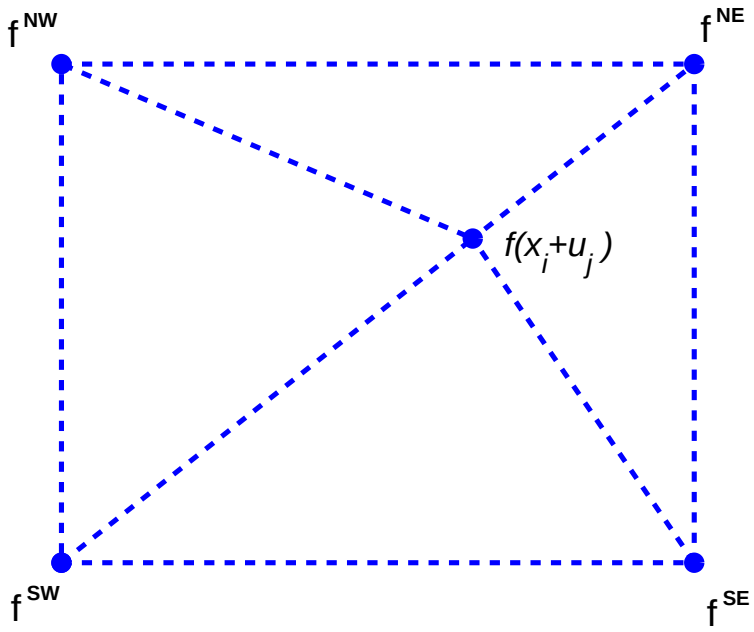

Figure 1: An illustration of bilinear interpolation. Here the corners f^{NW}, f^{NE}, f^{SW} and f^{SE} represent given discrete pixel values, and $f(x_i + u_j)$ is a value that must be approximated.

With this notation, we can write the interpolation operation in matrix notation

$$\mathbf{f}_j \approx \boldsymbol{S}(u_j)\mathbf{f} \tag{4}$$

where $\boldsymbol{S}(u_j)$ is a sparse $n \times n$ matrix (to simplify notation, we often write $\boldsymbol{S}_j \equiv \boldsymbol{S}(u_j)$). The nonzero elements of $\boldsymbol{S}_j$ are the interpolation weights; that is, the bilinear products of $u_{j,1}$ and $u_{j,2}$.

Combining the discretizations of the kernel and the discretization of the deformation process we obtain the following discrete set of nonlinear equations for the low resolution images

$$\mathbf{d}_j = \boldsymbol{K}\boldsymbol{S}_j\mathbf{f}\,, \quad j = 1, \ldots, k. \tag{5}$$

Thus, the general model we use to relate the desired high resolution image, $\mathbf{f}$, to the observed low resolution images, $\mathbf{d}_j$, is given by

$$\mathbf{d} = (\boldsymbol{I} \otimes \boldsymbol{K})\boldsymbol{S}(u)\mathbf{f} + \boldsymbol{\eta} \tag{6}$$

where

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_k \end{bmatrix}, \quad \boldsymbol{I} \otimes \boldsymbol{K} = \begin{bmatrix} \boldsymbol{K} & & \\ & \ddots & \\ & & \boldsymbol{K} \end{bmatrix}, \quad \boldsymbol{S}(u) = \begin{bmatrix} \boldsymbol{S}_1 \\ \vdots \\ \boldsymbol{S}_k \end{bmatrix},$$

and $\boldsymbol{\eta}$ is a vector representing unknown errors in the observed data, such as discretization errors and noise.

Our goal is to reconstruct the discrete high resolution image $\mathbf{f}$. It is obvious that if the displacement fields $u_j(x)$ are known then we obtain a linear problem. Nevertheless, in many cases the displacement is unknown and therefore one needs to evaluate the displacements and the high resolution image. It should be clear from the above equation that the two are tightly coupled. Indeed, note that even if $\boldsymbol{S}(u)$ is linear with respect to u the mixed derivative $\frac{\partial^2 \mathbf{d}}{\partial u \partial \mathbf{f}}$ does not vanish. Thus we see that two problems need to be solved in order to obtain $\mathbf{f}$ from $\mathbf{d}$. First, registration of the various low resolution images is needed in order to find the displacements, then the image restoration, or deblurring problem given in equation (5) needs to be solved. Each of these problems, considered independently, is known to be ill-posed [8, 10], and thus regularization is needed in order to obtain meaningful solutions.

To regularize the registration process, there are two main approaches. First, elastic potential or fluid registration are often used to regularize highly deformable image registration problems [12]. Second, it is possible to use a parametric approach and to assume that the displacements can be spanned by a small subspace of known vectors. While the former approach is more general, it is much more complex and requires special care. Therefore, in this paper, we have assumed that the displacement can be spanned by a small set of known vectors (we address extensions to more complex displacement fields in the summary). We set $u_j(x)$ to be a set of affine linear transformations, which allows for rotation, translation, scaling and shear. Thus, we assume that our deformation field at pixel j has the form

$$u_j(x) = \begin{pmatrix} u_{j,1} \\ u_{j,2} \end{pmatrix} = \begin{pmatrix} \gamma_{j,1} & \gamma_{j,2} \\ \gamma_{j,4} & \gamma_{j,5} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \gamma_{j,3} \\ \gamma_{j,6} \end{pmatrix}, \tag{7}$$

and thus

$$u_j(x_i) = \begin{pmatrix} u_{j,1} \\ u_{j,2} \end{pmatrix} = \begin{pmatrix} \gamma_{j,1} & \gamma_{j,2} \\ \gamma_{j,4} & \gamma_{j,5} \end{pmatrix} \begin{pmatrix} x_{i,1} \\ x_{i,2} \end{pmatrix} + \begin{pmatrix} \gamma_{j,3} \\ \gamma_{j,6} \end{pmatrix}. \tag{8}$$

The parameters $\gamma_{j,1}, \gamma_{j,2}, \ldots, \gamma_{j,6}$ are shared by all pixels in the jth deformed image. Therefore, if we define coordinate vectors

$$\mathbf{x}_1 = \begin{bmatrix} x_{1,1} \\ x_{2,1} \\ \vdots \\ x_{n,1} \end{bmatrix} \quad \text{and} \quad \mathbf{x}_2 = \begin{bmatrix} x_{1,2} \\ x_{2,2} \\ \vdots \\ x_{n,2} \end{bmatrix}$$

then we can represent corresponding deformation vectors as

$$\mathbf{u}_j = \begin{bmatrix} \mathbf{u}_{j,1} \\ \mathbf{u}_{j,2} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{e} & & & \\ & & & \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{e} \end{bmatrix} \begin{bmatrix} \gamma_{j,1} \\ \vdots \\ \gamma_{j,6} \end{bmatrix} = \boldsymbol{P}\boldsymbol{\gamma}_j \tag{9}$$

where $\mathbf{e}$ is the vector of all ones. Thus we see that the dependence of the interpolation matrices, $\boldsymbol{S}_j$, on $u_j(x)$ has been reduced to a dependence on only six parameters given by the vector $\boldsymbol{\gamma}_j$. We use this relationship in the next section as we explore numerical optimization techniques to solve the super-resolution problem.

3 Solution through Optimization

In order to solve the inverse problem we consider a Tikhonov-like framework where the solution is obtained by minimizing a functional of the form

$$\min_{\mathbf{f}, \boldsymbol{\gamma}_{1,\ldots,\ell}} \quad \frac{1}{2} \sum_{j=1}^{k} \|\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j\|^2 + \alpha R(\mathbf{f}) \tag{10}$$

where $\boldsymbol{S}_j \equiv \boldsymbol{S}(u_j) = \boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}_j)$ is the interpolation matrix that depends on $\boldsymbol{\gamma}_j$, $R(\mathbf{f})$ is a regularization operator, and α is a regularization parameter. For simplicity, in this work we consider only quadratic regularization operators of the form

$$R(\mathbf{f}) = \frac{1}{2}\|\boldsymbol{L}\,\mathbf{f}\|^2, \tag{11}$$

where $\boldsymbol{L}$ is a matrix representing a discrete differentiation operator. However, other regularization methods that allow for discontinuities, such as the Huber approach [2], can easily be accommodated within our framework.

To solve the optimization problem we need to compute derivatives. Consider first the term

$$\phi = \frac{1}{2}\|\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j\|^2.$$

Differentiating with respect to $\mathbf{f}$ is straightforward, and gives

$$\frac{\partial \phi}{\partial \mathbf{f}} = \boldsymbol{S}_j^\top \boldsymbol{K}^\top(\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j). \tag{12}$$

Differentiating with respect to $\boldsymbol{\gamma}_j$ is not trivial. First notice, using the interpolation formula (3), we have

$$\frac{\partial f(x_i + u_j)}{\partial u_{j,1}} = (1 - u_{j,2})(\mathrm{f}^{NE} - \mathrm{f}^{NW}) + u_{j,2}(\mathrm{f}^{SE} - \mathrm{f}^{SW}), \tag{13a}$$

and

$$\frac{\partial f(x_i + u_j)}{\partial u_{j,2}} = (1 - u_{j,1})(\mathrm{f}^{SW} - \mathrm{f}^{NW}) + u_{j,1}(\mathrm{f}^{SE} - \mathrm{f}^{NE}). \tag{13b}$$

Observe that the expressions given in (13) are equivalent to a simple discretization of the gradient of the image, assuming it is a piecewise linear function. We therefore define the Jacobian of the image with respect to the displacements $u_j(x)$ as

$$\boldsymbol{G}_j \equiv \boldsymbol{G}(u_j) \equiv \frac{\partial \mathbf{f}_j}{\partial u_j} = \frac{\partial[\boldsymbol{S}_j\mathbf{f}]}{\partial u_j}. \tag{14}$$

Using the chain rule, we obtain an expression for the partial derivatives of ϕ with respect to $\boldsymbol{\gamma}_j$:

$$\frac{\partial \phi}{\partial \boldsymbol{\gamma}_j} = \boldsymbol{P}^\top \boldsymbol{G}_j^\top \boldsymbol{K}^\top(\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j). \tag{15}$$

Using the partial derivatives (12) and (15), we obtain the Euler Lagrange equations which are the necessary conditions for a minimum of (10):

$$\mathbf{g}_1 \;=\; \sum_j \boldsymbol{S}_j^\top \boldsymbol{K}^\top(\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j) \;+\; \alpha R_{\mathbf{f}} = 0 \tag{16a}$$

$$\mathbf{g}_{2,j} \;=\; \boldsymbol{P}^\top \boldsymbol{G}_j^\top \boldsymbol{K}^\top(\boldsymbol{K}\boldsymbol{S}_j\mathbf{f} - \mathbf{d}_j) = 0, \;\; j = 1, 2, \ldots, k, \tag{16b}$$

where $R_{\mathbf{f}} = \dfrac{\partial R(\mathbf{f})}{\partial \mathbf{f}}$, and again we emphasize that $\boldsymbol{G}_j \equiv \boldsymbol{G}(u_j) = \boldsymbol{G}(\boldsymbol{P}\boldsymbol{\gamma}_j)$ and $\boldsymbol{S}_j \equiv \boldsymbol{S}(u_j) = \boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}_j)$. The equations given in (16) represent a nonlinear system in $\mathbf{f}$ and $\boldsymbol{\gamma}_j$. There are many options to solve such a system; in the following subsections we discuss three possible approaches.

3.1 The fully coupled approach

First, we consider using the Gauss-Newton method to solve for $\mathbf{f}$ and $\boldsymbol{\gamma}$ simultaneously. First we must linearize the system (16) to obtain an approximation of the Hessian

$$\boldsymbol{H} := \begin{pmatrix} \sum_{j=1}^{k} \boldsymbol{S}_j^\top \boldsymbol{K}^\top \boldsymbol{K} \boldsymbol{S}_j + \alpha R_{\mathbf{ff}} & \boldsymbol{B}_1 & \cdots & \boldsymbol{B}_k \\ \boldsymbol{B}_1^\top & \boldsymbol{P}^\top \boldsymbol{G}_1^\top \boldsymbol{K}^\top \boldsymbol{K} \boldsymbol{G}_1 \boldsymbol{P} & & \\ \vdots & & \ddots & \\ \boldsymbol{B}_k^\top & & & \boldsymbol{P}^\top \boldsymbol{G}_k^\top \boldsymbol{K}^\top \boldsymbol{K} \boldsymbol{G}_k \boldsymbol{P} \end{pmatrix} \tag{17}$$

where $\boldsymbol{S}_j = \boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}_j)$, $R_{\mathbf{ff}} = \dfrac{\partial^2 R(\mathbf{f})}{\partial \mathbf{f}^2}$, and

$$\boldsymbol{B}_j = \frac{\partial}{\partial \boldsymbol{\gamma}} \left(\boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}^{(j)})^\top \boldsymbol{K}^\top (\boldsymbol{K} \boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}^{(j)})\mathbf{f}) \right)$$

which is computed similar to the other derivatives.

In the Gauss-Newton iteration we solve the system

$$\boldsymbol{H} \begin{pmatrix} \delta \mathbf{f} \\ \delta \boldsymbol{\gamma} \end{pmatrix} = - \begin{pmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \end{pmatrix} \tag{18}$$

for the perturbations $\delta\boldsymbol{\gamma}$ and $\delta\mathbf{f}$, where $\mathbf{g}_2^\top = \begin{bmatrix} \mathbf{g}_{2,1}^\top & \cdots & \mathbf{g}_{2,k}^\top \end{bmatrix}$, and $\mathbf{g}_1$ and $\mathbf{g}_{2,j}$ are given, respectively, in equations (16a) and (16b).

The matrix $\boldsymbol{H}$ is typically very large, but it is also sparse, so we can use a preconditioned conjugate gradient method to solve (18). To further reduce the load at each iteration we can also use an inexact Gauss-Newton method [15] where the system (18) is solved only to a very rough tolerance. The approach is summarized in Algorithm 1.

There are a few difficulties when using the Gauss-Newton method on this fully coupled problem. First, we do not take algorithmic advantage of the fact that the problem is strongly convex in $\mathbf{f}$. Thus, we may use very small steps due to the nonlinearity induced by $\boldsymbol{\gamma}$. Moreover, the solution of the linear system can be rather expensive since it is difficult to find effective preconditioners for the coupled linear system. We therefore look at two alternative approaches.

Algorithm 1 Inexact Gauss-Newton method for Super Resolution

initialize $\{\boldsymbol{\gamma}, \mathbf{f}\}$ and the regularization parameter α
while true **do**
- compute $\mathbf{g}_1$ and $\mathbf{g}_2$ (equation (16)) and the Hessian $\boldsymbol{H}$ (equation (17))
- approximately solve the linear system (18)
- use a weak line search to accept/reject step
- test for termination

end while

3.2 The partially coupled approach

For quadratic regularization operators one can obtain a highly efficient algorithm by noting that equation (16a) is linear with respect to $\mathbf{f}$. In this case it is possible to eliminate $\mathbf{f}$ from the first equation by solving the system

$$\left(\sum_{j=1}^{k} \boldsymbol{S}_j^\top \boldsymbol{K}^\top \boldsymbol{K} \boldsymbol{S}_j \;+\; \alpha \boldsymbol{L}^\top \boldsymbol{L} \right) \mathbf{f} = \sum_j \boldsymbol{S}_j^\top \boldsymbol{K}^\top \mathbf{d}_j. \tag{19}$$

This yields a high resolution image $\mathbf{f}(\boldsymbol{\gamma})$. Substituting $\mathbf{f}(\boldsymbol{\gamma})$ into (16b) we obtain a set of equations that depends only on $\boldsymbol{\gamma}$,

$$\boldsymbol{P}^\top \boldsymbol{G}_j^\top \boldsymbol{K}^\top (\boldsymbol{K} \boldsymbol{S}_j \mathbf{f}(\boldsymbol{\gamma}) - \mathbf{d}_j) = 0, \quad j = 1, 2, \ldots, k. \tag{20}$$

It is easy to verify that this approach is a slight modification of the method proposed by Golub and Pereyra [7], and is similar to the scheme used by Vogel, Chan and Plemmons [18] for a problem arising in phase diversity blind deconvolution. In fact, it is easy to verify that equation (20) is the gradient of the reduced cost functional

$$\phi_{\text{red}}(\boldsymbol{\gamma}) = \frac{1}{2} \| \sum_{j=1}^{k} \boldsymbol{K} \boldsymbol{S}_j \mathbf{f}(\boldsymbol{\gamma}) - \mathbf{d}_j \|^2, \tag{21}$$

where $\mathbf{f}(\boldsymbol{\gamma})$ is the solution of the linear problem given in equation (19). Furthermore, it is easily verified that the Jacobian is given by

$$\boldsymbol{J}(\boldsymbol{\gamma}) = \begin{bmatrix} \boldsymbol{J}_1(\boldsymbol{\gamma}) & & \\ & \ddots & \\ & & \boldsymbol{J}_k(\boldsymbol{\gamma}) \end{bmatrix}$$

where

$$\boldsymbol{J}_j(\boldsymbol{\gamma}) = \frac{\partial(\boldsymbol{K}\boldsymbol{S}_j\mathbf{f}(\boldsymbol{\gamma}))}{\partial\boldsymbol{\gamma}_j} = \boldsymbol{K}\boldsymbol{G}_j\boldsymbol{P}. \tag{22}$$

Using the above observation we are able to use the Gauss-Newton method to minimize (21). The evaluation of the function ϕ_{red} and its derivatives involves the solution of the linear system (19). Note that this approach decouples the problem into solving a large sparse linear system for $\mathbf{f}$ and $\boldsymbol{J}$, and solving a small nonlinear optimization problem with respect to $\boldsymbol{\gamma}$. Since $\boldsymbol{J}$ can be easily evaluated given $\mathbf{f}$ the cost of every Gauss-Newton step for $\boldsymbol{\gamma}$ is negligible compared with the computation of $\mathbf{f}$. Furthermore, as we demonstrate in our numerical experiments, given a reasonable initial guess, the method converges in very few iterations.

3.3 The decoupled approach

In the case of nonlinear regularization the above decoupling may not be practical. In this case, we consider the simple idea of alternating between the minimization with respect to $\mathbf{f}$ and with respect to γ. This approach, which is referred to as coordinate descent in the optimization literature [15], has been used for super-resolution problems by Hardie, Barnard and Armstrong [9] within a MAP framework.

For our problem, we decouple the nonlinear system (16) in the obvious way. That is, we first assume that at iteration ℓ we have a guess for the fine scale, high resolution image, $\mathbf{f}^{(\ell)}$. We can then solve the system $\mathbf{g}_2(\mathbf{f}^{(\ell)}, \boldsymbol{\gamma}) = 0$ which yields k separate registration problems, which we solve to obtain $\boldsymbol{\gamma}_j^{(\ell)} \quad j = 1, \ldots, k$. Next, given the updated $\boldsymbol{\gamma}_j^{(\ell)}$, we can solve the system $\mathbf{g}_1(\mathbf{f}, \boldsymbol{\gamma}^{(\ell)}) = 0$ and proceed in an alternating fashion. Thus we can write the solution process as

$$(\boldsymbol{S}_j^{(\ell-1)})^\top \boldsymbol{K}^\top (\boldsymbol{K}\boldsymbol{S}_j^{(\ell-1)}\mathbf{f}^{(\ell)} - \mathbf{d}_j) \;+\; \alpha R_{\mathbf{f}}(\mathbf{f}^{(\ell)}) = 0 \tag{23a}$$

$$\boldsymbol{P}^\top \boldsymbol{G}_j^\top \boldsymbol{K}^\top (\boldsymbol{K}\boldsymbol{S}(\boldsymbol{P}\boldsymbol{\gamma}_j^{(\ell)})\mathbf{f}^{(\ell)} - \mathbf{d}_j) = 0 \quad j = 1, \ldots, k \tag{23b}$$

The advantage of this process is that we can use standard algorithms for image deblurring as well as standard algorithms for image registration. The difficulty, as is well known with coordinate descent type methods, is that it is not clear what are the practical convergence properties of the method. Moreover, if the method does converge, it will typically be very slow, especially for tightly coupled variables [15].

4 Numerical examples

In this section we demonstrate that the coupled algorithms can be superior to uncoupled approaches for the super-resolution problem. For the numerical tests reported in this section, we use a magnetic resonance (MR) image, which is available in MATLAB. The original high resolution image with 128^2 pixels, together with three low resolution images of 32^2 pixels, is shown in Figure 2.

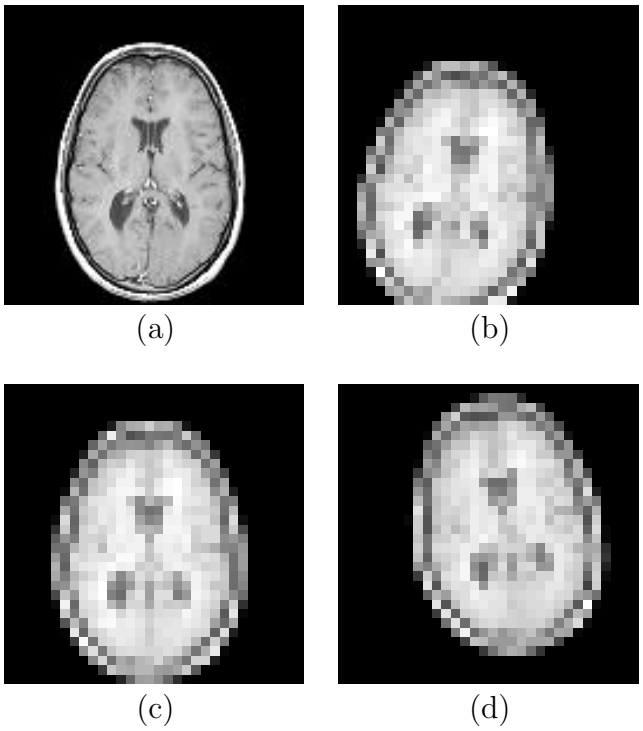

Figure 2: The high resolution image is shown in (a), and three selected low resolution images are shown in (b-d).

We assume that we have 32 low resolution images which are generated by a sequence of rotations and translations of the original image. For the reconstruction we choose quadratic regularization with $\boldsymbol{L}$ a discretization of the gradient operator. Since the regularization is quadratic we have used only the partially coupled approach and the decoupled approach described in Section 3. For each algorithm we compare the reconstructed image with

the result obtained by the standard approach of doing just one registration and one image reconstruction step. All computations were done in MATLAB, using IEEE double precision arithmetic.

4.1 Partially coupled approach

Here we consider the method described in subsection 3.2. We use two values of the regularization parameter. In order to obtain an initial guess for $\boldsymbol{\gamma}_j$ we solve the registration problem using the coarse images. The optimization problem is considered to converge when the relative gradient is smaller than 10^{-1} or when the displacement is smaller than half a pixel of the fine grid.

The images of the partially coupled approach and the standard uncoupled approach are presented in Figure 3. There is a noticeable difference between the computed reconstructions. In particular, many of the artifacts which can be seen in the standard approach do not appear when using the partially coupled method, even though we have used a very simple regularization functional.

A remarkable improvement can be observed when comparing the true displacements with the recovered ones. We define the relative error as

$$\Delta\boldsymbol{\gamma} = \frac{\|\boldsymbol{\gamma}^{\text{est}} - \boldsymbol{\gamma}^{\text{true}}\|}{\|\boldsymbol{\gamma}^{\text{true}}\|}$$

The initial relative difference between the true displacement and the coarse grid displacement is 0.5561. After our code completes its run, the difference is 0.0856 for $\alpha = 10^{-1}$ and 0.0855 for $\alpha = 10^{-2}$. This implies that our algorithm manages to obtain registration parameters which are roughly 4 times more accurate compared with the standard approach. Although we demonstrate marked improvement in the images, any reconstruction technique could benefit from such improvement. Additional information summarizing the convergence behavior of the algorithm is given in Table 1.

4.2 Decoupled approach

In order to understand how well the partially coupled approach performs for super-resolution type problems, we want to compare its performance to that of the decoupled approach described in subsection 3.3. Here we employ a coordinate descent type optimization method for the same problem as described above, with the regularization parameter set to 10^{-1}. Notice that

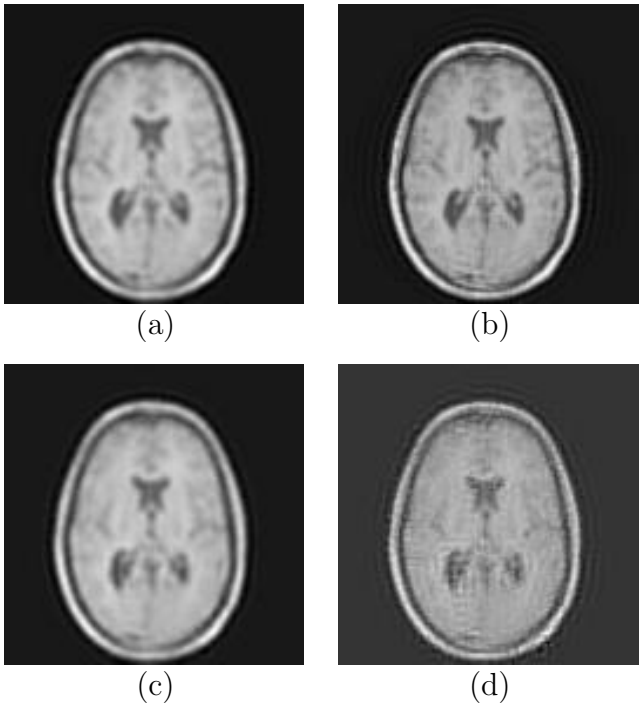

Figure 3: Comparison between the partially coupled and standard approach. The image shown in (a) used the partially coupled approach with $\alpha = 10^{-1}$, the image in (b) used the partially coupled approach with $\alpha = 10^{-2}$. In (c) we used the standard approach $\alpha = 10^{-1}$, and in (d) we used the standard approach $\alpha = 10^{-2}$.

iteration	rel objective	rel gradient	$\Delta\gamma$
$\alpha = 10^{-1}$			
0	1.0000	1.0000	0.5561
1	0.7931	0.5645	0.3231
2	0.7160	0.3578	0.1884
3	0.6854	0.2464	0.1185
4	0.6749	0.1630	0.0926
5	0.6712	0.1031	0.0856
$\alpha = 10^{-2}$			
0	1.0000	1.0000	0.5561
1	0.5689	0.5997	0.3531
2	0.3917	0.3748	0.2269
3	0.3112	0.3004	0.1480
4	0.2746	0.2210	0.1076
5	0.2598	0.1579	0.0911
6	0.2535	0.1058	0.0855

Table 1: Convergence of iterations for the partially coupled approach.

to ensure minimization of the objective function, we must implement a basic line search method after solving each of the systems, equation (23a) and equation (23b).

Convergence information of this algorithm is given in Table 2. Our numerical experiments show that this method is successful in decreasing the relative value of the objective function with each successive iteration; however, as expected, the convergence is much slower than that of the partially coupled approach. More specifically, the accuracy of the registration parameters at 30 iterations of the decoupled approach (further iterations produce essentially no additional accuracy) compares to that attained with only 1 iteration of the partially coupled approach.

Recall that the difficulty of these coordinate descent type algorithms is that practical convergence properties are uncertain. We found that with smaller regularization parameters, such as $\alpha = 10^{-2}$, the decoupled approach was ineffective. In particular, the iterations show very little improvement to the image and to the registration parameters.

iteration	rel objective	$\Delta\gamma$
$\alpha = 10^{-1}$		
0	1.000	0.5561
1	0.9834	0.5414
2	0.9741	0.5294
3	0.9578	0.5186
4	0.9247	0.4984
5	0.8833	0.4324
30	0.7465	0.3612

Table 2: Convergence of iterations for the decoupled approach.

5 Concluding Remarks

In this paper we have explored a new class of coupled optimization algorithms for super-resolution, which jointly treats the registration and deblurring parts of the problem. We have used Newton type methods and demonstrated that our approach is able to better estimate the registration parameters which in turn yields improved reconstructed images.

There are numerous points to explore in future work. First, we would like to extend our approach to different deformation models. Second, we would like to explore various techniques that can potentially produce better reconstructed images. For example, we could use different semi-norms such as Huber style regularizers and add nonnegativity constraints.

References

[1] H. C. Andrews. *N* Topics in search of an editorial: Heuristics, superresolution, and bibliography. *Proc. IEEE*, 60(7):340–343, 1972.

[2] U. Ascher, E. Haber, and H. Haung. On effective methods for implicit piecewise smooth surface recovery. *SIAM J. Sci. Comput.*, to appear, 2005.

[3] N. K. Bose and K. J. Boo. High-resolution image reconstruction with multisensors. *Int. J. Imaging Syst. Technol.*, 9:294–304, 1998.

[4] P. Cheeseman, B. Kanefsky, R. Kraft, J. Stutz, and R. Hanson. Super-resolved surface reconstruction from multiple images. Technical Report Tech. Rep. FIA-94-12, NASA Ames Research Center, Moffett Field, CA, Dec., 1994.

[5] M. Elad and A. Feuer. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. *IEEE Trans. Image Proc.*, 6(12):1646–1658, 1997.

[6] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar. Advances and challenges in super-resolution. *Int. J. Imaging Syst. Technol.*, 14(2):47–57, 2004.

[7] G. Golub and V. Pereyra. Separable nonlinear least squares: the variable projection method and its applications. *Inverse Problems*, 19:R1–R26, 2003.

[8] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems*. SIAM, Philadelphia, 1997.

[9] R. Hardie, K. J. Barnard, and E. E. Armstrong. Joint MAP registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Trans. Image Proc.*, 6:1621–1633, 1997.

[10] S. Horn. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[11] M. G. Kang and S. Chaudhuri. Super-resolution image reconstruction. *IEEE Signal Processing Magazine*, 20(3):19–20, 2003.

[12] J. Modersitzki. *Numerical Methods for Image Registration*. Oxford, 2004.

[13] M. Ng, R. Chan, T. Chan, and A. Yip. Cosine transform preconditioners for high resolution image reconstruction. *Linear Algebra Appl.*, 316:89–104, 2000.

[14] N. Nguyen, P. Milanfar, and G. Golub. Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement. *IEEE Trans. Image Proc.*, 10(9):1299–1308, 2001.

[15] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, New York, 1999.

[16] S. C. Park, M. K. Park, and M. G. Kang. Super-resolution image reconstruction: A technical overview. *IEEE Signal Processing Magazine*, 20(3):21–36, 2003.

[17] B. C. Tom and A. K. Katsaggelos. Reconstruction of a high-resolution image by simultaneous registration, restoration, and interpolation of low-resolution images. In *Proc. IEEE Int. Conf. Image Processing*, Washington, DC, 1995.

[18] C. R. Vogel, T. Chan, and R. J. Plemmons. Fast algorithms for phase diversity-based blind deconvolution. In *Adaptive Optical System Technologies*, volume 3353. SPIE, 1998.