J Math Imaging Vis 25: 353–364, 2006

DOI: 0.1007/s10851-006-7250-2

Edge-Forming Methods for Image Zooming

YOUNGJOON CHA
Department of Applied Mathematics, Sejong University, 98 Kunja-Dong, Seoul, 143-747, South Korea
yjcha@sejong.ac.kr

SEONGJAI KIM
Department of Mathematics and Statistics, Mississippi State University, Mississippi State, MS 39762-5921 USA
skim@math.msstate.edu

Published online: 29 August 2006

Abstract. The article is concerned with edge-forming methods to be applied as a post-process for image zooming. Image zooming via standard interpolation methods often produces the so-called *checkerboard effect*, in particular, when the magnification factor is large. In order to remove the artifact and to form reliable edges, a nonlinear semi-discrete model and its numerical algorithm are suggested along with anisotropic edge-forming numerical schemes. The algorithm is analyzed for stability and choices of parameters. For image zooming by integer factors, a few iterations of the algorithm can form clear and sharp edges for gray-scale images. Various examples are presented to show effectiveness and efficiency of the newly-suggested edge-forming strategy.

Keywords: image zooming, interpolation, checkerboard effect, edge-forming

1. Introduction

Image interpolation techniques are often required in various tasks in image processing and computer vision such as image generation, compression, and zooming. It is the first of the two basic resampling steps and transforms a discrete image into a continuous function. (The second step is to produce the resampled discrete image.) Image resampling is necessary for every geometric transform of discrete images except shifts over integer distances or rotations about multiples of 90 degrees. Thus image interpolation methods have occupied a peculiar position in image processing and computer graphics [13, 14, 18, 19, 26].

There are two kinds of interpolation methods: linear and nonlinear ones. For linear methods, diverse interpolation kernels of finite size have been introduced, in the literature, as approximations of the *ideal* interpolation kernel (sinc function) which is spatially unlimited; see Lehmann et al. [19] and references therein. Two of the simplest approximations are related to the nearest-neighbor interpolation and the bilinear interpolation. Higher-order interpolation methods involving larger number of pixel values have shown superior properties for some classes of images. However, most of the linear interpolation methods have been introduced with no count on edges. Thus they bring up the smoothing effect in resulting images. Furthermore, when the image is zoomed by a large factor, the zoomed image looks blocky; such a phenomenon is call the *checkerboard effect*.

Recently, some nonlinear interpolation methods have been suggested to overcome the artifacts of linear methods [3, 16, 20]. The major step in the nonlinear

*The work of this author is supported in part by NSF grant DMS-0312223.

methods is to either fit the edges with some templates or predict edge information for the high resolution image from the low resolution one.

In this article, we are interested in the development of edge-forming methods to be applied as a post-process of linear image zooming methods, with a hope of removing the checkerboard effect. We will introduce a nonlinear semi-discrete model that incorporates a constraint term and then suggest anisotropic edge-forming numerical schemes in order for the model to be able to form edges. For simplicity, we focus on edge-forming techniques for interpolated images by integer factors, in gray-scale in 2D. In the case, a portion of pixel values in the enlarged image can be assigned directly from the original image without approximation, while most of the values are approximated from neighboring pixel values. During the edge-forming, we try not to change the original values and try to alter the approximated values to form reliable edges, by setting the parameter of the constraint term appropriately.

For a closely related method, see a total variation based interpolation method suggested by Guichard and Malgouyres [15]. See also [21], where some linear and nonlinear interpolation methods are analyzed mathematically and experimentally, including the total variation based interpolation.

An outline of the article is as follows. In Section 2, we present a brief review on linear interpolation methods. Section 3 begins with a nonlinear semi-discrete model; the remainder of the section suggests efficient time-stepping procedures and anisotropic edge-forming schemes for the model. Stability and the choice of constraint parameter are analyzed for the discrete algorithm. In Section 4, we verify the performance of the algorithm in terms of its ability for edge-forming and efficiency, for image zooming by integer factors. Section 5 contains discussion on applications of the algorithm to 3D and/or color images, followed by its applicability to image zooming by non-integer factors. The last section summarizes our new-findings and experiments.

2. Preliminaries

In this section, we briefly review linear interpolation methods.

The interpolation step of image resampling is to construct a two-dimensional (2D) continuous function $u(x, y)$ from its discrete image samples $u(k, \ell)$, where x and y are real numbers and k and ℓ are integers. It can be formally expressed as the convolution of the discrete image samples with a continuous 2D filter H_{2D}:

$$u(x, y) = \sum_k \sum_\ell u(k, \ell)\, H_{2D}(x - k, y - \ell). \quad (1)$$

Usually, the interpolation kernel H_{2D} is selected to be symmetric and separable to reduce the computational complexity,

$$H_{2D}(x, y) = H(x) \cdot H(y), \quad (2)$$

where H is symmetric, i.e., $H(-x) = H(x)$. It is often required for the kernel H to satisfy

$$\begin{aligned} &H(0) = 1, \\ &H(x) = 0, \quad |x| = 1, 2, \ldots \end{aligned} \quad (3)$$

and the partition of unity condition

$$\sum_{k=-\infty}^{\infty} H(d + k) = 1, \quad \text{for all } 0 \le d < 1. \quad (4)$$

The conditions in (3) guarantee that the image is not modified if it is resampled on the same grid, and therefore the kernel avoids smoothing and preserves high frequencies. Kernels that fulfill (3) are called *interpolators*, while others are called *approximators*. The condition (4) implies that the energy of the resampled image remains unchanged. In other words, the mean brightness of the image is not altered when the image is interpolated or resampled. Kernels that satisfy or fail (4) are named respectively *direct current* (DC)-constant or DC-inconstant. It is known that superior kernels are interpolators and (DC)-constant. However, they are not sufficient conditions for superior kernels.

For example, the nearest-neighbor interpolation and the linear interpolation respectively correspond to the rectangular function H_1 and the triangular function H_2 defined as

$$\begin{aligned} H_1(x) &= \begin{cases} 1, & 0 \le |x| < 0.5, \\ 0, & \text{elsewhere}, \end{cases} \\ H_2(x) &= \begin{cases} 1 - |x|, & 0 \le |x| < 1, \\ 0, & \text{elsewhere}. \end{cases} \end{aligned} \quad (5)$$

It is easy to check that H_1 and H_2 are DC-constant interpolators, but they are in practice suffering from both the attenuation of high-frequency components and aliasing.

Higher-order interpolation methods may improve the image quality. For references, we present the following cubic interpolators of one-parameter family:

$$H_{4,a}(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1, & 0 \le |x| < 1, \\ a\,(|x|^3 - 5|x|^2 + 8|x| - 4), & 1 \le |x| < 2, \\ 0, & \text{elsewhere.} \end{cases} \tag{6}$$

These cubic interpolators are DC-constant and C^1-continuous; for C^2-continuity, one should set $a = -3/4$. When $a = 0$, the kernel (6) requires at most two pixel values of the image in interpolation; we will also denote it as H_2^{cubic}, indicating that the kernel is cubic and utilizes two pixel values.

Figure 1 shows zoomed images (of magnification of 4×4) by the interpolation methods mentioned in this section. As one can see from the figure, all interpolation methods have instituted checkerboard and smoothing effects to the resulting image. It should be noticed that higher-order methods improve the image quality *slightly*; however, linear image zooming methods themselves cannot guarantee a high-quality image.

We may conclude from the above example that image zooming needs to incorporate not only a decent interpolation method but also an effective post-process that is able to reduce checkerboard effect and to form reliable edges. The primary goal in this article is to

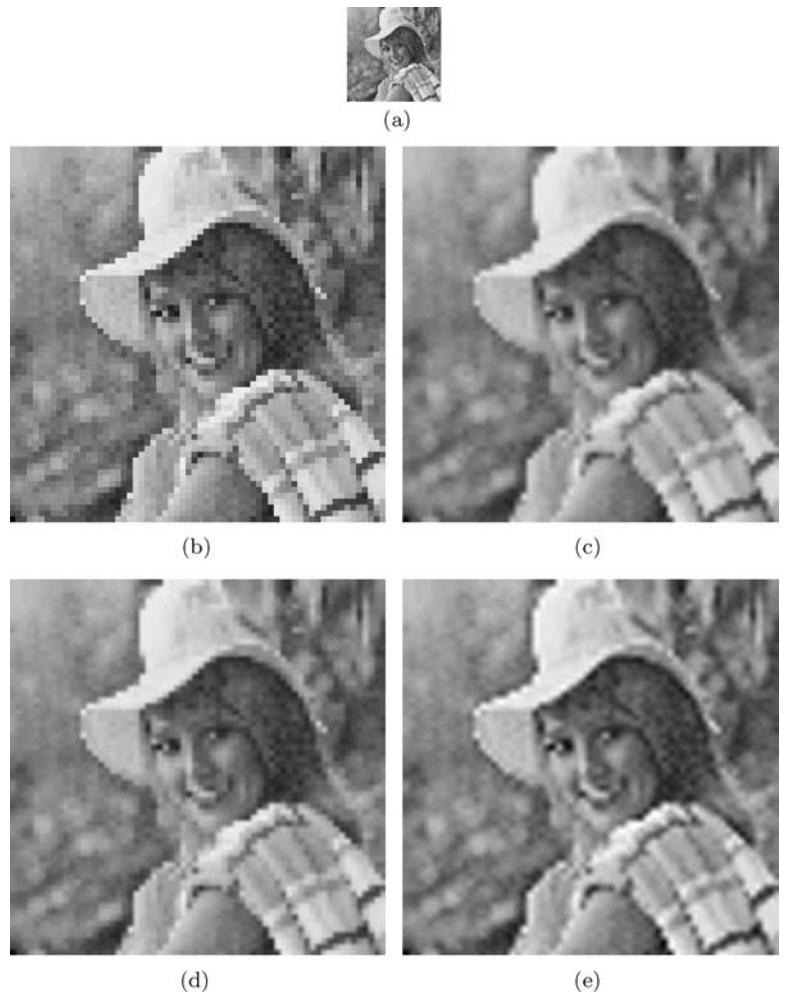


Figure 1. The Elaine images: (a) The original image in 64×64 pixels and interpolated images in 256×256 cells by (b) H_1, (c) H_2, (d) $H_2^{\text{cubic}} = H_{4,0}$, and (e) $H_{4,-3/4}$.

develop an edge-forming algorithm that can connect the edges, locally linear or smoother. Note that the linear edge-connection differs from the linear data-interpolation. Such an edge-forming algorithm must utilize surrounding values of the original image in order to satisfy conditions related to the curvature of image contents.

3. The Edge-Forming Method

This section introduces a nonlinear semi-discrete model and its linearized time-stepping procedure (a discrete algorithm) and then addresses effective anisotropic numerical schemes for edge-forming. Stability is analyzed for the discrete algorithm.

3.1. A Nonlinear Semi-Discrete Model

Let f be a given image which is zoomed by utilizing one of linear interpolation methods presented in Section 2. Then we can write

$$f = u + e,$$

where u is the desired image (hopefully, having sharp and reasonable edges) and e denotes the noise most of which is arisen during the interpolation. Consider the following nonlinear semi-discrete model of the form

$$\frac{\partial u}{\partial t} + \mathcal{A}u \;=\; \beta\,(f - u), \tag{7}$$

where β denotes a constraint parameter and $\mathcal{A} = \mathcal{A}(u)$ is a *diffusion* matrix, i.e., for $\mathcal{A} = (a_{rs})$,

$$a_{rs} \le 0, \;\; r \ne s; \quad a_{rr} = \sum_{s \ne r} |a_{rs}| > 0, \;\; \forall\, r.$$

The recovered image u becomes closer to f as β grows. Note that the matrix elements depend on the solution u. In Section 3.3, we will construct the diffusion matrix that is anisotropic and shows an ability to form edges.

3.2. A Linearized Time-Stepping Procedure

Denote the timestep size by Δt. Set $t^n = n\Delta t$ and $u^n = u(\cdot, t^n)$ for $n \ge 0$. Then, the problem can be linearized by evaluating the matrix $\mathcal{A}(u)$ from the previous time level. Consider the linearized θ-method for (7) of the form:

$$\begin{aligned}\frac{u^n - u^{n-1}}{\Delta t} + (\mathcal{A}^{n-1} + \beta I)\,[\theta u^n + (1-\theta)u^{n-1}] \\ = \beta\, f, \qquad 0 \le \theta \le 1,\end{aligned} \tag{8}$$

where $\mathcal{A}^{n-1} = \mathcal{A}(u^{n-1})$. The above time-stepping procedure is a combination of two classical numerical techniques: the incomplete iteration and the θ-method. The incomplete iteration has been studied extensively in 1970s and 1980s for the numerical solution of nonlinear parabolic partial differential equations; see [2, 8, 9, 28] and references therein. Note that the θ-method turns out to be the explicit, the implicit, and the Crank-Nicolson methods respectively for $\theta = 0$, $\theta = 1$, and $\theta = 1/2$; see e.g. [23].

For an efficient simulation of (8), we will select $\mathcal{A}^{n-1}$ separable, i.e.,

$$\mathcal{A}^{n-1} = \mathcal{A}_1^{n-1} + \mathcal{A}_2^{n-1},$$

where $\mathcal{A}_1$ and $\mathcal{A}_2$ are submatrices that represent connections of pixel values in the horizontal and vertical directions, respectively. Let

$$\mathcal{B}_\ell^{n-1} = \mathcal{A}_\ell^{n-1} + \frac{1}{2}\beta I, \quad \ell = 1, 2.$$

Then, the *alternating direction implicit* (ADI) method [7, 12, 25] is a perturbation of (8), with a splitting error of $\mathcal{O}(\Delta t^2)$:

$$\begin{aligned}&\frac{u^* - u^{n-1}}{\Delta t} + \mathcal{B}_1^{n-1}\,[\theta u^* + (1-\theta)u^{n-1}] + \mathcal{B}_2^{n-1}\,u^{n-1} \\ &\quad = \beta\, f, \\ &\frac{u^n - u^{n-1}}{\Delta t} + \mathcal{B}_1^{n-1}\,[\theta u^* + (1-\theta)u^{n-1}] \\ &\quad + \mathcal{B}_2^{n-1}\left[\theta u^n + (1-\theta)u^{n-1}\right] = \beta\, f,\end{aligned} \tag{9}$$

or equivalently [10, 11]

$$\begin{aligned}&\left[1 + \theta\,\Delta t\,\mathcal{B}_1^{n-1}\right] u^* \\ &\quad = \left[1 - (1-\theta)\,\Delta t\,\mathcal{B}_1^{n-1} - \Delta t\,\mathcal{B}_2^{n-1}\right] u^{n-1} \\ &\qquad + \Delta t\,\beta\, f,\end{aligned} \tag{10}$$

$$\left[1 + \theta\,\Delta t\,\mathcal{B}_2^{n-1}\right] u^n = u^* + \theta\,\Delta t\,\mathcal{B}_2^{n-1} u^{n-1},$$

where u^* is an intermediate solution. Note that when the matrices $\mathcal{A}_\ell^{n-1}$ are composed with a 3-point stencil, each sweep in (10) can be carried out by inverting a series of tri-diagonal matrices.

The ADI method was first introduced by Douglas, Peaceman, and Rachford [7, 12, 25], as a perturbation of the Crank-Nicolson difference equation, for solving the heat equation in 2D. Recently, Douglas and Kim [11] suggested strategies for a virtual elimination of the splitting error of the ADI method and its variants.

3.3. *Anisotropic Edge-Forming Difference Schemes*

For simplicity, we begin with edge-forming schemes for (7) in 1D. Then we will extend them for images in 2D.

3.3.1. Edge-Forming in 1D. When images are to be magnified by an integer factor k, one can manage the interpolation algorithm such that the values at each k-th pixel can be assigned directly from the original image without approximation. It is then desirable that we try not to alter those original values during the post-processing. Thus we may set β in (7) large at the pixels of original values and let $\beta = 0$ elsewhere.

There is no theoretical upper bound for β. However, it must be sufficiently large at the pixels of original values, because otherwise the values cannot be preserved as desired. See (29) below for appropriate choices of β.

Set $\beta = 0$ and consider the following explicit method ($\theta = 0$):

$$u^n = u^{n-1} - \Delta t \mathcal{A}_1^{n-1} u^{n-1}, \tag{11}$$

where Δt is small enough for the algorithm to be stable. Let x_i be the i-th pixel in the x-direction and $u_i = u(x_i)$. Then we construct the i-th row of $\mathcal{A}_1^{n-1}$ consisting of three consecutive non-zero elements which represent the connection of u_i to u_{i-1} and u_{i+1}:

$$\left[\mathcal{A}_1^{n-1}\right]_i = \left(-a_{i,W}^{n-1},\ a_{i,W}^{n-1} + a_{i,E}^{n-1},\ -a_{i,E}^{n-1}\right). \tag{12}$$

We wish to determine $a_{i,W}^{n-1}$ and $a_{i,E}^{n-1}$ in such a way that the algorithm (11) reveals an ability of edge-forming. Our candidates are as follows:

$$\begin{aligned} a_{i,W}^{n-1} &= \frac{2\, d_{i,E}^{n-1}}{d_{i,W}^{n-1} + d_{i,E}^{n-1}}, \\ a_{i,E}^{n-1} &= \frac{2\, d_{i,W}^{n-1}}{d_{i,W}^{n-1} + d_{i,E}^{n-1}}, \end{aligned} \tag{13}$$

where, for $q \geq 0$,

$$\begin{aligned} d_{i,W}^{n-1} &= \left[\left(u_i^{n-1} - u_{i-1}^{n-1}\right)^2 + \varepsilon^2\right]^{q/2}, \\ d_{i,E}^{n-1} &= \left[\left(u_{i+1}^{n-1} - u_i^{n-1}\right)^2 + \varepsilon^2\right]^{q/2} = d_{i+1,W}^{n-1}. \end{aligned} \tag{14}$$

Here the regularization parameter $\varepsilon > 0$ is introduced to prevent the denominator in (13) from approaching zero and is assumed to be small enough. Note that $a_{i,W}^{n-1} + a_{i,E}^{n-1} = 2$ and $d_{i,W}^{n-1}$ in (14) is the second-order finite difference approximation of $[(u_x^{n-1})^2 + \varepsilon^2]^{q/2}$ evaluated at $x_{i-1/2} (:= (x_{i-1} + x_i)/2)$.

At the boundary pixels, the model (7) must incorporate a certain boundary condition such as e.g. the Dirichlet or the no-flux condition. In this article, we will construct the linear system $\mathcal{A}_1^{n-1}$ as if the image is flat at the boundary (no-flux boundary condition). For instance, let x_0 be the left end pixel. Then

$$\begin{aligned} [\mathcal{A}_1^{n-1}]_0 &= \left(0,\ a_{0,W}^{n-1} + a_{0,E}^{n-1},\ -a_{0,W}^{n-1} - a_{0,E}^{n-1}\right) \\ &= (0,\ 2,\ -2). \end{aligned} \tag{15}$$

The right end pixel can be treated similarly. Other boundary conditions can be incorporated with minor modifications.

When $q = 0$, the scheme (12) turns out to be the central second-order approximation of $-\partial_{xx}$ (with the grid size equal to one). In this case, for the solution profile given as in Fig. 2, we can see $u_i^n > u_i^{n-1}$ in (11), because $[\mathcal{A}_1^{n-1} u^{n-1}]_i < 0$. Furthermore, it is not difficult to check that $u_i^n = u_i^{n-1}$ for $q = 1$ and $u_i^n < u_i^{n-1}$ for $q = 2$. In general, we can see that at x_i in Fig. 2,

$$u_i^n \begin{pmatrix} \geq \\ = \\ \leq \end{pmatrix} u_i^{n-1}, \quad \text{respectively for} \quad \begin{cases} q < 1, \\ q = 1, \\ q > 1. \end{cases}$$

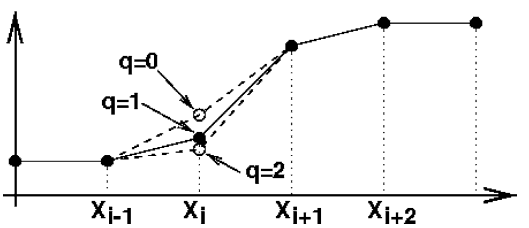


Figure 2. The edge-forming principle. The solid curve involving solid circles is the solution at the previous level u^{n-1} and the dashed curves indicate the solutions in the current level near the point x_i depending on q.

Exploiting the same arguments, we also can see that at x_{i+1} in Fig. 2,

$$u_{i+1}^n \begin{pmatrix} \le \\ = \\ \ge \end{pmatrix} u_{i+1}^{n-1}, \quad \text{respectively for} \quad \begin{cases} q < 1, \\ q = 1, \\ q > 1. \end{cases}$$

We summarize the above observation as follows: The formula in (13)–(14) can make the image sharper for $q > 1$ and therefore form edges. Such a strategy will work not only for the explicit algorithm 11 but also for other time-stepping procedures such as the implicit ($\theta = 1$) and Crank-Nicolson ($\theta = 1/2$) methods. The formula is clearly anisotropic.

Remark. The edge-forming formula in (13)–(14) can be obtained from a numerical approximation of the differential operator

$$D_q = -|u_x|_\varepsilon{}^q \left(\frac{u_x}{|u_x|_\varepsilon{}^q} \right)_x, \quad |u_x|_\varepsilon = \sqrt{u_x^2 + \varepsilon^2}. \tag{16}$$

Indeed, the formula can be obtained with the following differences:

$$\begin{aligned} \left(\frac{u_x}{|u_x|_\varepsilon{}^q} \right)_x (x_i) &\approx \frac{1}{d_{i,W}} u_{i-1} - \left(\frac{1}{d_{i,W}} + \frac{1}{d_{i,E}} \right) u_i \\ &\quad + \frac{1}{d_{i,E}} u_{i+1}, \\ |u_x|_\varepsilon{}^q (x_i) &\approx 2 \frac{d_{i,W} \cdot d_{i,E}}{d_{i,W} + d_{i,E}}, \end{aligned}$$

evaluated at proper time levels. Thus (11) can be viewed as an explicit scheme for the following problem

$$\frac{\partial u}{\partial t} + D_q = 0. \tag{17}$$

For a clearer understanding of characteristics of (16), we differentiate its right-side and simplify the result to read

$$D_q = -\frac{\varepsilon^2 - (q-1)u_x^2}{\varepsilon^2 + u_x^2} u_{xx}. \tag{18}$$

It follows from (17) and (18) that D_q is a diffusion operator when $q \le 1$. On the other hand, when $q > 1$, D_q is a diffusion operator for small gradients ($|u_x| < \varepsilon/\sqrt{q-1}$), while (17) acts as a reverse-time heat equation for larger gradients ($|u_x| > \varepsilon/\sqrt{q-1}$). In practice, q must not be too large.

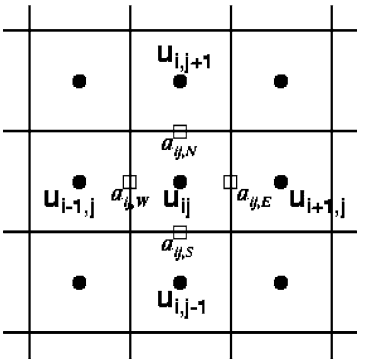


Figure 3. The stencil at the pixel $\mathbf{x}_{ij}$.

The classical reverse-time heat equation is known to be unstable. We will analyze stability for the discrete problem in 2D, the θ-method (8); see Section 3.4.

3.3.2. Edge-Forming in 2D. Now, we consider edge-forming schemes for 2D images. Recall that for the 2D problem, $\mathcal{A}_1$ and $\mathcal{A}_2$ are submatrices that represent connections of pixel values in the horizontal and vertical directions, respectively. Here we will present schemes for $\mathcal{A}_1$; the schemes can be applied correspondingly for $[\mathcal{A}_2^{n-1}]_{ij}$.

As in (12) and (13), we let the row in $\mathcal{A}_1^{n-1}$ corresponding to the pixel $\mathbf{x}_{ij}$ consist of three consecutive non-zero elements:

$$\left[\mathcal{A}_1^{n-1}\right]_{ij} = \left(-a_{ij,W}^{n-1},\ a_{ij,W}^{n-1} + a_{ij,E}^{n-1},\ -a_{ij,E}^{n-1} \right), \tag{19}$$

where, corresponding to the stencil in Fig. 3,

$$a_{ij,W}^{n-1} = \frac{2\, d_{ij,E}^{n-1}}{d_{ij,W}^{n-1} + d_{ij,E}^{n-1}}, \quad a_{ij,E}^{n-1} = \frac{2\, d_{ij,W}^{n-1}}{d_{ij,W}^{n-1} + d_{ij,E}^{n-1}}. \tag{20}$$

Here $d_{ij,W}^{n-1}$ and $d_{ij,E}^{n-1}$ can be determined as follows:

$$d_{ij,W}^{n-1} = \left[\left(\mathcal{D}\, u_{i-1/2,j}^{n-1} \right)^2 + \varepsilon^2 \right]^{q/2}, \quad d_{ij,E}^{n-1} = d_{i+1,j,W}^{n-1}, \tag{21}$$

where $\mathcal{D}\, u_{i-1/2,j}^{n-1}$ is to be defined as finite difference approximations of $|\nabla u^{n-1}|$ evaluated at $\mathbf{x}_{i-1/2,j}$, the mid point of $\mathbf{x}_{i-1,j}$ and $x_{i,j}$. In the following, we consider two cases.

- **Case 1**: We define $\mathcal{D} = \mathcal{D}_1$ as

$$\begin{aligned}\mathcal{D}_1 u^{n-1}_{i-1/2,j} &= \left(\left(u^{n-1}_{i,j} - u^{n-1}_{i-1,j}\right)^2 + \left[\frac{1}{2}\left(\frac{u^{n-1}_{i-1,j+1} + u^{n-1}_{i,j+1}}{2} - \frac{u^{n-1}_{i-1,j-1} + u^{n-1}_{i,j-1}}{2}\right)\right]^2\right)^{1/2} \\ &= \left(\left(u^{n-1}_{i,j} - u^{n-1}_{i-1,j}\right)^2 + \left(u^{n-1}_{i-1,j+1} + u^{n-1}_{i,j+1} - u^{n-1}_{i-1,j-1} - u^{n-1}_{i,j-1}\right)^2/16\right)^{1/2}. \end{aligned} \tag{22}$$

- **Case 2**: Then, we consider another approximation $\mathcal{D} = \mathcal{D}_2$ as

$$\mathcal{D}_2 u^{n-1}_{i-1/2,j} = \frac{1}{2}\left(\mathcal{D}_c u^{n-1}_{i-1,j} + \mathcal{D}_c u^{n-1}_{i,j}\right), \tag{23}$$

where $\mathcal{D}_c$ is the central second-order approximation of $|\nabla|$ defined as

$$\mathcal{D}_c u^{n-1}_{i,j} = \left(\left(u^{n-1}_{i+1,j} - u^{n-1}_{i-1,j}\right)^2/4 + \left(u^{n-1}_{i,j+1} - u^{n-1}_{i,j-1}\right)^2/4\right)^{1/2}.$$

3.4. *Stability of the θ-Model*

When the image is to be magnified by a factor of $k_1 \times k_2$, where k_1 and k_2 are positive integers, one can interpolate the image such that the values at every $(k_1 \times k_2)$-th pixel are assigned directly from the original image without approximation. Then, for the edge-forming, we set the parameter β large at those pixels not to alter the original values much, and set $\beta = 0$ elsewhere. That is,

$$\beta(\mathbf{x}) = \begin{cases} \beta_0 \text{ at each } (k_1 \times k_2)\text{-th pixel of original values,} \\ 0 \quad \text{elsewhere,} \end{cases} \tag{24}$$

where β_0 is a positive number. Being equipped with the numerical schemes in the previous subsection, the θ-method (8) satisfies the maximum principle in the following sense.

Theorem 3.1. *Let the θ-method* (8)*, $0 \le \theta \le 1$, incorporate the schemes in* (19)–(21) *and satisfy the following condition*

$$4(1-\theta)\Delta t \le 1, \tag{25}$$

with β set as in (24)*. Suppose the solution of* (8)*, u^n, have a local maximum or minimum at a point $\hat{\mathbf{x}}_{ij}$ where $\beta = 0$. Then it is constant, for all $q \ge 0$, on the block of $(k_1 \times k_2)$ pixels that contains the point $\hat{\mathbf{x}}_{ij}$.*

Proof: First note that $a^{n-1}_{ij,W} + a^{n-1}_{ij,E} + a^{n-1}_{ij,S} + a^{n-1}_{ij,N} = 4$ for all (i, j) and for all $q \ge 0$. The Eq. (8) at a point $\mathbf{x}_{ij}$ can be written as

$$\begin{aligned}&[1 + (4+\beta)\theta\Delta t]\, u^n_{ij} \\ &= \theta\Delta t\left[a^{n-1}_{ij,W}\, u^n_{i-1,j} + a^{n-1}_{ij,E}\, u^n_{i+1,j} + a^{n-1}_{ij,S}\, u^n_{i,j-1} + a^{n-1}_{ij,N}\, u^n_{i,j+1}\right] + (1-\theta)\Delta t\left[a^{n-1}_{ij,W}\, u^{n-1}_{i-1,j} + a^{n-1}_{ij,E}\, u^{n-1}_{i+1,j} + a^{n-1}_{ij,S}\, u^{n-1}_{i,j-1} + a^{n-1}_{ij,N}\, u^{n-1}_{i,j+1}\right] \\ &\quad + [1 - (4+\beta)(1-\theta)\Delta t]\, u^{n-1}_{ij} + \Delta t\beta\, f_{ij}. \end{aligned} \tag{26}$$

Let $\beta = 0$ at $\mathbf{x} = \mathbf{x}_{ij}$ and u^n_{ij} be a local maximum or minimum. Then, it follows from (25) that each of coefficients in the right side of (26) is nonnegative and their sum becomes $[1 + 4\theta\Delta t]$. Thus the values at the four adjacent pixels of $\mathbf{x}_{ij}$ in both new and previous time levels and u^{n-1}_{ij} must be the same as u^n_{ij}. The argument can be applied repeatedly to show that $u^m, m \le n$, must be constant at least on the block of $(k_1 \times k_2)$ pixels that contains the point $\mathbf{x}_{ij}$. □

Thus, when (25) is satisfied, the θ-method (8) may have local extrema only at the pixels of original values where $\beta = \beta_0 > 0$ and it cannot introduce extra local extrema inside the block of $(k_1 \times k_2)$ pixels. The maximum principle is important, because otherwise the θ-method may create unnecessary features during the edge-forming. Note that the model (7) incorporating the edge-forming schemes presented in the previous subsection, (19)–(23), acts like a reverse-time heat equation in the continuous level, as analyzed in (16)–(18) for the 1D problem.

Note that the implicit method ($\theta = 1$) is unconditionally stable. Indeed, when the θ-method satisfies $(4+\beta)(1-\theta)\Delta t \le 1$, utilizing the same argument in the proof of Theorem (3.1), one can show that for every choice of $\beta \ge 0$,

$$\min_{i,j} f_{ij} \le u^n_{ij} \le \|f\|_\infty, \quad n \ge 0, \tag{27}$$

where $\|f\|_\infty := \max_{i,j} |f_{ij}|$, the maximum norm of f.

The following theorem completes stability analysis for the θ-method (8). It also reveals a way of choosing appropriate β with which the difference $|u^n - f|$ is small enough at the pixels where $\beta > 0$.

Theorem 3.2. *Let the θ-method* (8), $0 \le \theta \le 1$, *incorporate the schemes in* (19)–(21) *and satisfy*

$$(4+\beta)(1-\theta)\Delta t \le 1. \tag{28}$$

Then,

$$\max_{\beta>0} \left|u_{ij}^n - f_{ij}\right| \le \frac{4}{4+\beta}\,\|f\|_\infty, \quad n \ge 1. \tag{29}$$

Proof: Let $\delta_{ij}^{n-1} := u_{ij}^{n-1} - f_{ij}$. Then it follows from (26) that

$$\begin{aligned}
&[1+(4+\beta)\,\theta\Delta t]\,u_{ij}^n \\
&\quad = \theta\Delta t\left[a_{ij,W}^{n-1}\,u_{i-1,j}^n + a_{ij,E}^{n-1}\,u_{i+1,j}^n + a_{ij,S}^{n-1}\,u_{i,j-1}^n\right. \\
&\qquad \left. + a_{ij,N}^{n-1}\,u_{i,j+1}^n\right] + (1-\theta)\Delta t\left[a_{ij,W}^{n-1}\,u_{i-1,j}^{n-1}\right. \\
&\qquad \left. + a_{ij,E}^{n-1}\,u_{i+1,j}^{n-1} + a_{ij,S}^{n-1}\,u_{i,j-1}^{n-1} + a_{ij,N}^{n-1}\,u_{i,j+1}^{n-1}\right] \\
&\qquad + [1-4\Delta t + (4+\beta)\theta\Delta t]\,f_{ij} \\
&\qquad + [1-(4+\beta)(1-\theta)\Delta t]\,\delta_{ij}^{n-1}
\end{aligned} \tag{30}$$

and therefore

$$\begin{aligned}
&[1+(4+\beta)\,\theta\Delta t]\left(u_{ij}^n - f_{ij}\right) \\
&\quad = \theta\Delta t\left[a_{ij,W}^{n-1}\,u_{i-1,j}^n + a_{ij,E}^{n-1}\,u_{i+1,j}^n + a_{ij,S}^{n-1}\,u_{i,j-1}^n\right. \\
&\qquad \left. + a_{ij,N}^{n-1}\,u_{i,j+1}^n\right] + (1-\theta)\Delta t\left[a_{ij,W}^{n-1}\,u_{i-1,j}^{n-1}\right. \\
&\qquad \left. + a_{ij,E}^{n-1}\,u_{i+1,j}^{n-1} + a_{ij,S}^{n-1}\,u_{i,j-1}^{n-1} + a_{ij,N}^{n-1}\,u_{i,j+1}^{n-1}\right] \\
&\qquad - 4\Delta t\,f_{ij} + [1-(4+\beta)(1-\theta)\Delta t]\,\delta_{ij}^{n-1}.
\end{aligned} \tag{31}$$

Thus it follows from (27) and the equality $a_{ij,W}^{n-1} + a_{ij,E}^{n-1} + a_{ij,S}^{n-1} + a_{ij,N}^{n-1} = 4$ that

$$\left|u_{ij}^n - f_{ij}\right| \le \frac{4\Delta t}{1+(4+\beta)\,\theta\Delta t}\,\|f\|_\infty + \gamma_0\,\|\delta^{n-1}\|_\infty, \tag{32}$$

where

$$\gamma_0 = \frac{1-(4+\beta)(1-\theta)\Delta t}{1+(4+\beta)\,\theta\Delta t} = 1 - \frac{(4+\beta)\Delta t}{1+(4+\beta)\,\theta\Delta t}.$$

The condition in (28) implies that $0 \le \gamma_0 < 1$. Note that $\|\delta^0\|_\infty = \|u^0 - f\|_\infty = 0$. Hence

$$\begin{aligned}
\left|u_{ij}^n - f_{ij}\right| &\le \frac{4\Delta t}{1+(4+\beta)\,\theta\Delta t} \cdot \sum_{k=0}^{n-1} \gamma_0^k \cdot \|f\|_\infty \\
&\le \frac{4\Delta t}{1+(4+\beta)\,\theta\Delta t} \cdot \frac{1}{1-\gamma_0} \cdot \|f\|_\infty \\
&= \frac{4}{4+\beta}\,\|f\|_\infty,
\end{aligned} \tag{33}$$

which completes the proof. □

For instance, if it is desired for $|u_{ij}^n - f_{ij}|$ to be not larger than 1% the given image f at the pixels where $\beta > 0$, one should choose $\beta \ge 396$.

4. Numerical Experiments

In this section, we provide a numerical verification for the performance of the semi-discrete model (7) incorporating schemes presented in Section 3. For all experiments, we choose $\theta = 1$, $\Delta t = 1$, $\beta = 1000$ for the θ-method (8) and $q = 1.5$ and $\varepsilon = 0.05$ in (21). Note that β is set large at the pixels of original values and zero elsewhere. For other choices which may violate the stability conditions in Theorems 3.1 and 3.2, the program works fine in practice. For example, when we select $\theta = 0.5$ with other parameters given the same, the performance shows no observable differences.

For all examples in this section, the interpolated image is scaled by 1/255 such that the maximum possible value of the image in the edge-forming computation is one. Thus every parameter choice discussed in this section is related to the scaled images. The final results are scaled back and quantized for the 8-bit display.

In Fig. 4, we compare the two schemes, $\mathcal{D}_1$ (22) and $\mathcal{D}_2$ (23), applied to edge-forming for an interpolated disk by a factor of 4×4. Figure 4(a) depicts the interpolated image by H_2^{cubic}. In Figs. 4(b) and (c), we present the edge-formed images by applying the ADI (10) along with $\mathcal{D}_1$ (22) and $\mathcal{D}_2$ (23), respectively, starting from the image in Fig. 4(a). As one can see from the figure, the interpolated image shows the checkerboard effect. In edge-forming, $\mathcal{D}_1$ performs better than $\mathcal{D}_2$, which has been observed for most examples we have tested.

For the above example, one may try to enlarge the image by two recursive applications of 2×2 magnification and edge-forming rather than once 4×4 magnification followed by edge-forming. The interpolation introduces no observable difference, while the edge-forming may improve the results a lot.

For the examples below, we utilize $\mathcal{D}_1$ only.

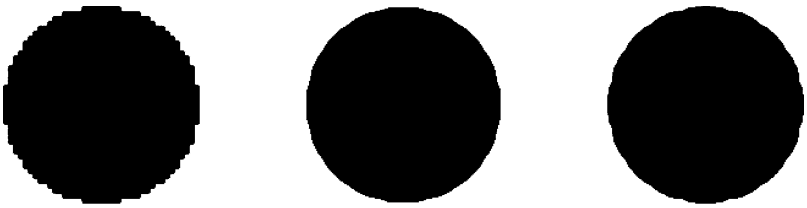


Figure 4. Synthetic binary images in 240 × 240 cells, for magnification of 4 × 4. (a) The interpolated image by H_2^{cubic} from an image in 60 × 60 cells, (b) the edge-formed image by $\mathcal{D}_1$ (22), and (c) the edge-formed image by $\mathcal{D}_2$ (23).

In Fig. 5, we continue investigating the performance of the newly suggested edge-forming schemes in Section 3. Figure 5(a) depicts the interpolated and edge-formed image by a factor of 8 × 8 once, while Fig. 5(b) contains the image magnified by three recursive applications of 2 × 2 interpolation and edge-forming. The image of large magnification, Fig. 5(a), shows portions where the edge of the disk is under-developed (concave in) from the interpolated image; the edge is developed better in Fig. 5(b).

Figure 6 shows a synthetic image in 360 × 360 cells for 9 × 9 image magnification. The original image g in Fig. 6(a) is shrunken to a small image of 40 × 40 cells and interpolated into 360 × 360 cells by H_2^{cubic}, Fig. 6(b). The small image is utilized for two recursive applications of 3 × 3 interpolation and edge-forming; the resulting image is u, as depicted in Fig. 6(c). The edges are clearly formed in the figure, although the sharp corners of the features are not well recovered due to a lack of information in the shrunken image. Note that edges near the image boundary have altered (to be normal) by the effect of boundary treatment discussed in (15). The error $|g - u|$ is shown in Fig. 6(d). The thickness of features in the error must be eight at most, except at sharp corners. The line segments in the error are verified to have thicknesses of 1–3.

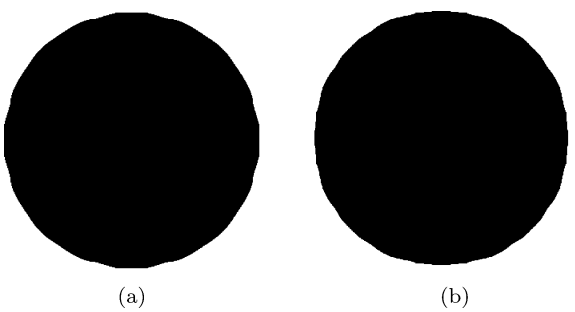


Figure 5. Synthetic binary images in 480 × 480 cells, for magnification of 8 × 8 from the same original image as in Figure 4. (a) The edge-formed image by 8 × 8 magnification once and (b) the edge-formed image by three recursive applications of 2 × 2 magnification and edge-forming. H_2^{cubic} is utilized for the interpolation.

Figure 7 presents the performance of the edge-forming algorithm applied to a real image, a Lenna Face in 50 × 50 cells. Figure 7(a) presents the 4 × 4 interpolated image by H_2^{cubic} and Fig. 7(b) contains the edge-formed image by two recursive applications of 2 × 2 interpolation and edge-forming. As one can see from the figure, the interpolated image shows the checkerboard effect, most of which has been eliminated in the edge-formed image.

For the edge-forming for Fig. 7(b), three iterations of ADI (10) are applied after each interpolation. When the image is to be zoomed 2–3 times in each direction, the edge-forming algorithm has developed reliable edges in 2–3 ADI iterations for most images; later iterations make the edge-formed image look smoother a little bit. For the image zooming by factors of $k \geq 4$, the edge-forming requires a larger number of ADI iterations. By a recursive application of smaller factors, one can speed up the simulation, because the earlier recursions are applied to smaller images and therefore much cheaper in computation.

We have tested the edge-forming algorithm with various choices of parameters: θ, Δt, β, q, and ε. It has been verified that the performance is acceptable and *barely* sensitive to the choices of parameters when we set $\theta \in [0.5, 1], \Delta t \in [0.5, 2], \beta \geq 400, q \in [1.3, 1.7]$, and $\varepsilon \in [0.01, 0.1]$.

5. Discussion

In this section, we discuss issues arising when the edge-forming algorithm is applied to 3D and/or color images, followed by its applicability to image zooming by non-integer factors. Finally we will consider deblurring.

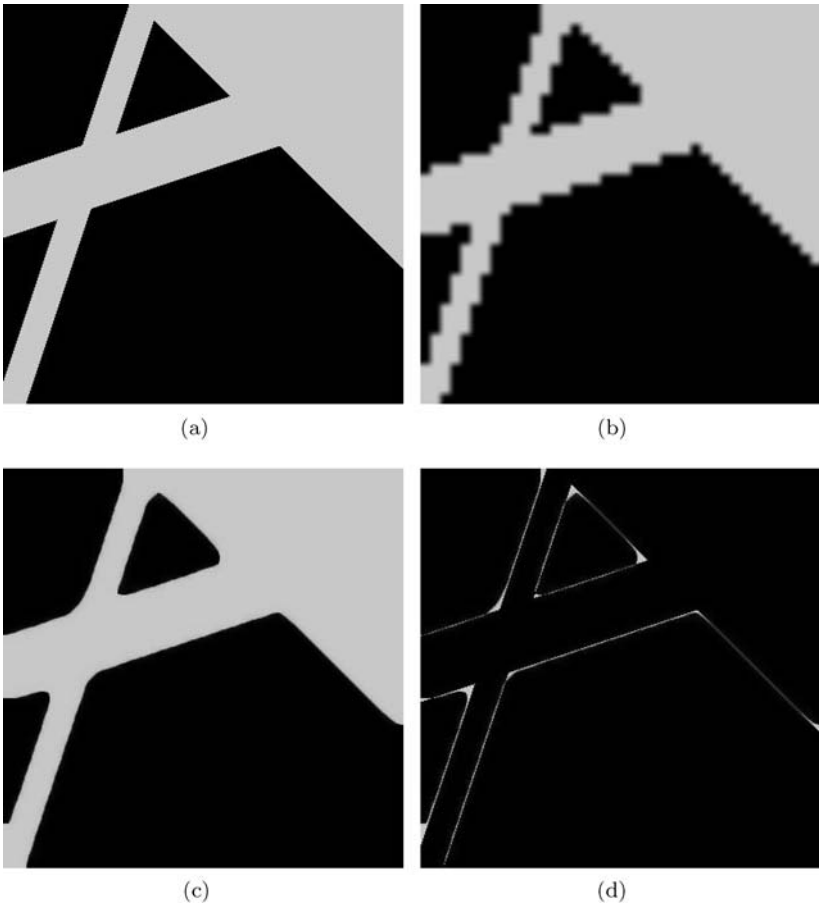


Figure 6. A synthetic image in 360 × 360 cells for 9 × 9 image magnification. (a) The original image g, (b) the H_2^{cubic}-interpolated image by a factor of 9 × 9 from a shrunken image in 40 × 40 cells, (c) the edge-formed image u, and (d) the absolute error $|g - u|$.

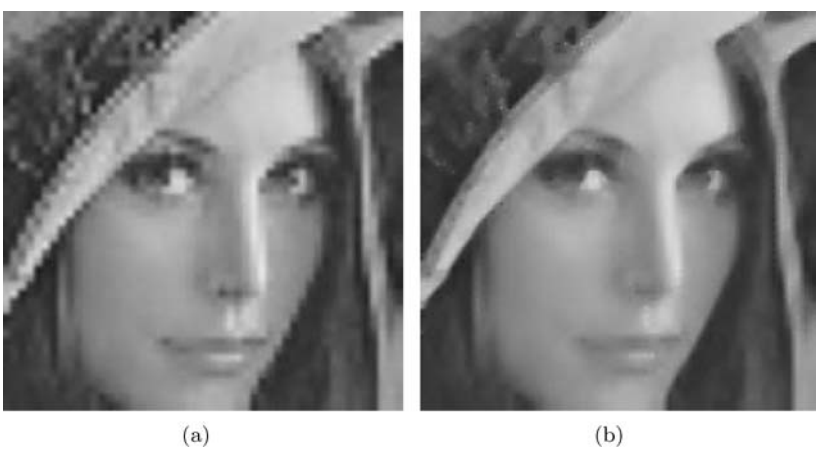


Figure 7. Lenna Face in 200 × 200 cells, for magnification of 4 × 4. (a) The interpolated image by H_2^{cubic} from an image in 50 × 50 cells and (b) the edge-formed image by two recursive applications of 2 × 2 interpolation and edge-forming.

Extension for 3D images: It is easy and almost straightforward to apply the edge-forming algorithm for 3D images. For instance, one may adopt the Douglas-Gunn's general formula [10] for the ADI (10). The numerical schemes in Section 3.3.2 can be extended for 3D images, without meeting difficulties. However, when 3D is concerned, efficiency becomes another important issue. Techniques for efficiency improvement, along with applications to 3D image construction via slice-stacking, will be presented elsewhere.

Extension for color images: Color images consist of three components, e.g., red, blue, and green. One may apply the edge-forming algorithm for color images in a component-wise fashion. However, the interpolation and the edge-formed results can involve undesirable colors, due to different developments of colors in separate treatments. Another interesting transformation overcoming such a difficulty is the brightness-chromaticity decomposition in the angle domain, which has extensively studied for color image processing [1, 6, 17, 29, 30]. Now, an efficient and reliable numerical modeling for such a system of coupled differential equations would be a challenging task.

Image zooming by non-integer factors: When image is to be magnified by a non-integer factor, it may be difficult to apply the edge-forming algorithm suggested in this article. However, the algorithm is still applicable when the factor is given in the form of $k = \ell/m$, where ℓ and m are positive integers. In the case, one can manage the interpolation algorithm such that each ℓ-th pixel value can be assigned from the original image without approximation, for which one can set β large. For example, let $k = 3/2$. Then, it is easy to see that each third value in the interpolated image can be obtained directly from the original image.

For image zooming by general factors, one has to develop an effective strategy for the choice of $\beta = \beta(\mathbf{x})$, which is very critical not to introduce an excessive blur into the resulting image. Such a subject along with color image zooming will be dealt with in detail in a forthcoming article [4].

Deblurring: When the parameter β is chosen large enough (see (29)), the values obtained from the original image can be kept with a minor change (or exactly the same in quantized integer values). Thus the edge-formed image can be viewed as a zoomed image by an *improved interpolation*. However, the result often looks blurrier than usual, which we believe is due to a corresponding expansion of light scattering inherited in the original image. Thus it is reasonable to apply a deblurring algorithm to edge-formed images [5, 22, 24, 27].

6. Conclusions

When standard interpolation methods are applied for image zooming by a large factor, the enlarged image often shows the checkerboard effect. To remove the artifact and form reliable edges, we have suggested a new edge-forming strategy. A nonlinear semi-discrete diffusion model and its discrete algorithm have been introduced along with anisotropic edge-forming numerical schemes. The algorithm has been analyzed for stability and reliable choices of parameters. Various numerical examples are presented in order to convince the effectiveness and efficiency of the algorithm. For gray-scale images in 2D, the new edge-forming strategy has shown a satisfactory performance for image zooming by integer factors; edges are formed clearly and sharply. Discussion is made for the algorithm to be applicable to image zooming for 3D and/or color images, or by non-integer factors.

Acknowledgment

The authors are grateful for the constructive and helpful suggestions from our anonymous referees.

References

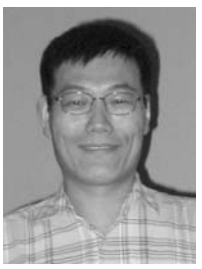
1. P.V. Blomgren and T.F. Chan, "*Color TV: Total* variation methods for restoration of vector valued images," *IEEE Trans. Image Processing*, Vol. 7, pp. 304–309, 1998.
2. J. Bramble, J.H. Pasciak, P.H. Sammon, and V. Thomée, "Incomplete iterations in multistep backward difference methods for parabolic problems with smooth and nonsmooth data," *Math. Comp.*, Vol. 52, pp. 339–367, 1989.
3. W. Carey, D. Chuang, and S. Hemami, "Regularity-preserving image interpolation," *IEEE Trans. Image Process*, Vol. 8, pp. 1293–1297, 1999.
4. Y. Cha and S. Kim, "Edge-forming methods for color image zooming," (*accepted to IEEE Trans. Image Process*).
5. T. Chan and C. Wong, "Total variation blind deconvolution," *IEEE Trans. Image Process*, Vol. 7, pp. 370–375, 1998.
6. T.F. Chan and J. Shen, "Variational restoration of non-flat image features: Models and algorithms," *SIAM Journal of Applied Mathematics*, Vol. 61, pp. 1338–1361, 2000.
7. J. Douglas, Jr., "On the numerical integration of $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{\partial u}{\partial t}$ by implicit methods," *J. Soc. Indust. Appl. Math.*, Vol. 3, pp. 42–65, 1955.

8. J. Douglas, Jr., "On incomplete iteration for implicit parabolic difference equations," *J. Soc. Indust. Appl. Math.*, Vol. 9, pp. 433–439, 1961.
9. J. Douglas, Jr., T. Dupont, and R. Ewing, "Incomplete iteration for time-stepping a Galerkin method for a quasilinear parabolic problem," *SIAM J. Numer. Anal.*, Vol. 16, pp. 503–522, 1979.
10. J. Douglas, Jr. and J. Gunn, "A general formulation of alternating direction methods Part I. Parabolic and hyperbolic problems," *Numer. Math.*, Vol. 6, pp. 428–453, 1964.
11. J. Douglas, Jr. and S. Kim, "Improved accuracy for locally one-dimensional methods for parabolic equations," *Mathematical Models and Methods in Applied Sciences*, Vol. 11, pp. 1563–1579, 2001.
12. J. Douglas, Jr. and D. Peaceman, "Numerical solution of two-dimensional heat flow problems," *American Institute of Chemical Engineering Journal*, Vol. 1, pp. 505–512, 1955.
13. G. Grevera and J. Udupa, "An objective comparision of 3-D image interpolation methods," *IEEE Trans. Medical Imaging*, Vol. 17, pp. 642–652, 1998.
14. ———, "A tack-specific evaluation of three-dimensional image interpolation techniques," *IEEE Trans. Medical Imaging*, Vol. 18, pp. 137–143, 1999.
15. F. Guichard and F. Malgouyres, "Total variation based interpolation," in *Proceedings of the Ninth European Signal Processing Conference*, Patran, Greece, 1998, pp. 1741–1744.
16. K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," *IEEE Trans. Image Process*, Vol. 4, pp. 285–295, 1995.
17. S. Kim, "PDE-based image restoration: A hybrid model and color image denoising," (accepted to *IEEE Trans. Image Processing*).
18. C. Lee, M. Eden, and M. Unser, "High-quality image resizing using oblique projection operators," *IEEE Trans. Image Process*, Vol. 7, pp. 679–692, 1998.
19. T. Lehmann, C. Gönner, and K. Spitzer, "Survey: Interpolation methods in medical image processing," *IEEE Trans. Medical Imaging*, Vol. 8, pp. 1049–1075, 1999.
20. X. Li and M. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process*, Vol. 10, pp. 1521–1527, 2001.
21. F. Malgouyres and F. Guichard, "Edge direction preserving image zooming: A mathematical and numerical analysis," *SIAM J. Numer. Anal.*, Vol. 39, pp. 1–37, 2001.
22. A. Marquina and S. Osher, "Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal," *SIAM J. Sci. Comput.*, Vol. 22, pp. 387–405, 2000.
23. K. Morton and D. Mayers, *Numerical Solution of Partial Differential Equations*, 2nd Ed.," Cambridge University Press, New York, 2005.
24. G. Panci, P. Campisi, S. Colonnese, and G. Scarano, "Multichannel blind image deconvolution using the Bussgang algorithm: Spatial and multiresolution approaches," *IEEE Trans. Image Process.*, Vol. 12, pp. 1324–1337, 2003.
25. D. Peaceman and H. Rachford, "The numerical solution of parabolic and elliptic differential equations," *J. Soc. Indust. Appl. Math*, Vol. 3, pp. 28–41, 1955.
26. G. Penney, J. Schnabel, D. Rueckert, M. Viergever, and W. Niessen, "Registration-based interpolation," *IEEE Trans. Medical Imaging*, Vol. 23, pp. 922–926, 2004.
27. M. Razaz and R. Lee, "Comparison of iterative deconvolution and wiener filtering for image restoration, in image processing: Mathematical Methods and Applications," Vol. 61 of *Inst. Math. Appl. Conf. Ser.*, Oxford Univ. Press, Oxford, 1997, pp. 145–159.
28. T. Russell, "Time stepping along characteristics with incomplete iteration for a galerkin approximation of miscible displacement in porous media," *SIAM J. Numer. Anal.*, Vol. 22, pp. 970–1013, 1885.
29. G. Sapiro and D. Ringach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Trans. Image Processing*, Vol. 5, pp. 1582–1586, 1996.
30. N. Sochen, R. Kimmel, and R. Malladi, "A general framework for low level vision," *IEEE Trans. Image Processing*, Vol. 7, pp. 310–318, 1998.

Youngjoon Cha received his B.Sc. (1988) and M.Sc. (1990) from Mathematics, Seoul National University, Seoul, South Korea; and Ph.D. (1996) from Mathematics, Purdue University, working on mathematical epidemiology, under a guidance of Prof. Fabio Milner.

He was a post-doctoral researcher at Purdue University, and Seoul National University, South Korea, from 1996 to 1997 and from 1997 to 1998, respectively.

He is currently an associate professor in the Department of Applied Mathematics, Sejong University, South Korea. His research interests include image processing, mathematical and numerical modeling for waves, and mathematical epidemiology.



Seongjai Kim received his B.Sc. (1988) and M.Sc. (1990) from Mathematics, Seoul National University, Seoul, South Korea; and Ph.D. (1995) from Mathematics, Purdue University, working on computational fluid dynamics, under a guidance of Prof. Jim Douglas, Jr..

After two years of post-doctoral research on seismic inversion at Rice University, he worked for Shell E&P Tech. Co., Houston, for a year and the Department of Mathematics, University of Kentucky, for seven years. He is currently an associate professor in the Department of Mathematics and Statistics, Mississippi State University. His research interests are in mathematical and numerical modeling for wave propagation in highly heterogeneous media, seismology, and image processing for challenging images.