Locally Adaptive Wavelet-Based Image Interpolation

S. Grace Chang, Zoran Cvetković, *Senior Member, IEEE*, and Martin Vetterli, *Fellow, IEEE*

***Abstract*—We describe a spatially adaptive algorithm for image interpolation. The algorithm uses a wavelet transform to extract information about sharp variations in the low-resolution image and then implicitly applies interpolation which adapts to the image local smoothness/singularity characteristics. The proposed algorithm yields images that are sharper compared to several other methods that we have considered in this paper. Better performance comes at the expense of higher complexity.**

***Index Terms*—Image interpolation, locally adaptive, nonlinear interpolation, wavelet.**

I. Introduction

The CLASSIC problem of image interpolation refers to extracting information from the given image to fill in the unknown pixels values. It is used for magnification and zooming purposes, which are the applications we have in mind. The challenge is to process the image in such a way as to keep the magnified image sharp. Traditional methods, such as bilinear interpolation or spline approximation, inherently assume smoothness constraints on the signal and, as a result, they typically generate blurred images. The subjective quality of the interpolated images can be improved by employing more sophisticated image models as has been explored in many different directions in the signal processing literature, including edge-directed interpolation, various methods that use projections onto convex constraints, morphological filtering, etc [1]–[14]. The problem of image interpolation has been also studied within the computer vision framework where it is normally approached using probabilistic modeling that relies on extensive training [15], [16]. In this paper, we study a wavelet-based method which attempts to extract information about local regularity and sharp variations in a low-resolution image and use that information to apply different interpolation functions that adapt locally to image smoothness/singularity characteristics.

Points of sharp variations, or singularities, are among the most meaningful features of a signal. For images, these points typically correspond to edges, or boundaries between regions. Information about singularities can be obtained by multiscale edge detection methods developed in the computer vision community [17]–[20]. The multiscale edge detection can be formulated in the wavelet framework (e.g., as the Canny edge detector [20] is equivalent to finding the local maxima in the wavelet transform). This multiscale edge characterization framework will be used here, as it allows both a convenient analysis of edges and a model for the interpolation algorithm introduced in this paper.

For a family of wavelets, the wavelet-transform modulus maxima capture the sharp variation points of a signal, and their evolution across scales characterizes the local Lipschitz regularity of the signal [21]–[23]. For example, Fig. 1 shows a one-dimensional (1-D) signal and its wavelet transform at several scales. This signal includes singularities, such as a step and an impulse, and other sharply varying regions. Each of these sharp variations induces peaks in the wavelet transform across scales, and the values of the peaks corresponding to the same singularity change across the scales according to an exponential function, the exponent of which depends on the singularity.

The proposed interpolation algorithm will first capture and characterize sharp variation points based on the multiscale wavelet analysis. This characterization is then used to estimate the high-frequency information necessary to preserve sharpness of the edges. From the model of the problem, one can identify constraints on the estimate and, thus, refine the estimate iteratively. The major difference between the proposed method and other nonlinear or locally adaptive schemes is that by means of wavelet analysis, it adapts to singularities of different kinds (e.g., step-like edges, relatively smooth edges, Dirac-like behavior, variations which are continuous but not continuously differentiable, etc.).

The outline of the proposed method has been previously described in a conference paper [24] where many important practical issues and investigations could not be addressed either because of the required brevity or the lack of their complete understanding at that time. However, the ideas presented in [24] have led several interesting developments on the topic of wavelet-based image enhancement and interpolation [25]–[30]; hence, we felt that this more in-depth treatment of the proposed algorithm would be in place.

The paper is organized as follows. Section II introduces the wavelet-transform framework, and relates the multiscale edge detection to the wavelet analysis. The discussion starts in continuous time, followed by issues due to discretization. In Section III-A, details of the interpolation problem model and algorithm are discussed in the 1-D case for clarity. This algorithm is extended to reconstruct two-dimensional (2-D) images in Section III-C. Results and comparisons with traditional interpolation methods are presented in Section IV.

Manuscript received November 5, 2004; revised February 8, 2005. This work was supported in part by the National Science Foundation under the Alan T. Waterman Award Grant CCR-0139398. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yucel Altunbasak.

S. G. Chang is with Hewlett-Packard Taiwan Ltd., Taipei 110, Taiwan, R.O.C.

Z. Cvetković was with the Division of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA. He is now with the Department of Electronic Engineering, King's College London, London WC2R 2LS, U.K. (e-mail: zoran.cvetkovic@kcl.ac.uk).

M. Vetterli is with the School of Computer and Communications Sciences, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne 1015, Switzerland, and also with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720 USA.

Digital Object Identifier 10.1109/TIP.2006.871162

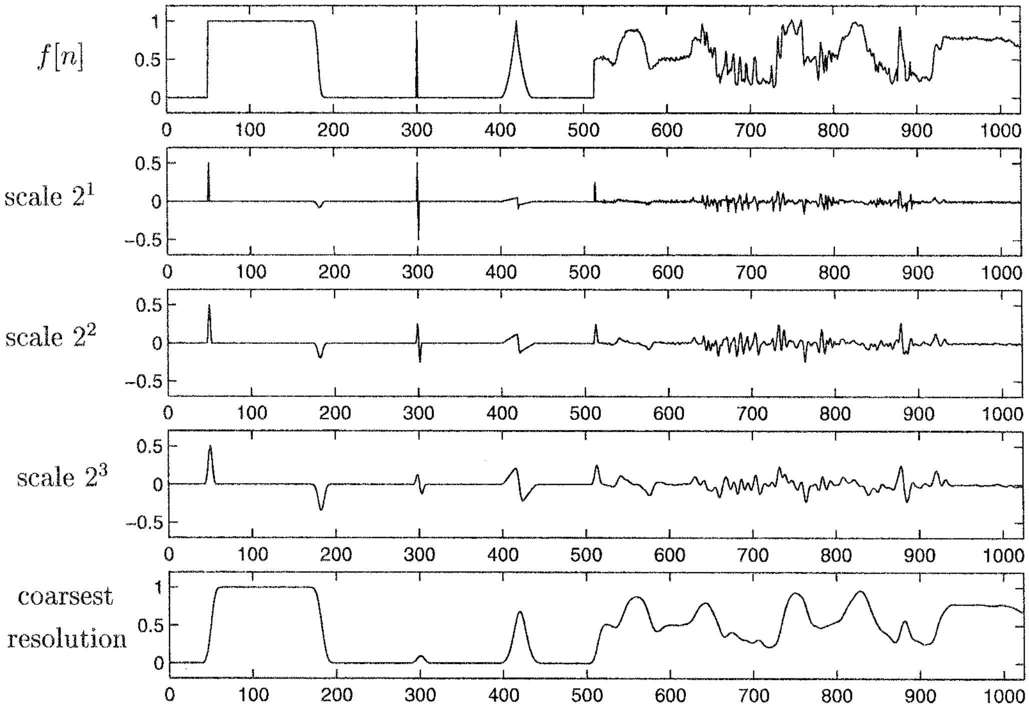


Fig. 1. A 1-D waveform and its wavelet transform for three scales, showing the propagation of extrema points across the scales.

II. Multiscale Edges and the Wavelet Transform

In this section, we review work by Mallat [22], [23] and Meyer [21] which relates edge detection algorithms to the wavelet transform.

Most traditional edge detectors extract sharp variation points by examining the first derivatives of the signal or its smoothed version. This is because an inflection point indicates a neighborhood of signal variation, and an inflection point in the signal domain corresponds to the local extremum of its first derivative. Furthermore, a local extremum of the first derivative with large magnitude corresponds to a region of sharp variation in the signal domain, while one with small magnitude corresponds to a region of slow transition. This edge detection strategy can be formulated in the wavelet framework as follows.

Let $\theta(x)$ be a smoothing function which satisfies $\lim_{x\to\pm\infty}\theta(x) = 0$ and $\int_{-\infty}^{\infty}\theta(x)dx = 1$. Assume that $\theta(x)$ is differentiable and define a function $\psi(x)$: $\psi(x) = d\theta(x)/dx$. A wavelet is defined to be any function which integrates to zero. Hence, $\psi(x)$ can be considered as a wavelet. Now let $\psi_s(x)$ denote the dilated version of the wavelet function

$$\psi_s(x) = \frac{1}{s}\psi\left(\frac{x}{s}\right)$$

where s is the scale. The wavelet transform of f at scale s and position x, denoted by $W_s f(x)$, is given by the convolution $W_s f(x) = f(x) * \psi_s(x)$. From the linearity of convolution and differentiation, it is easy to verify that

$$W_s f(x) = f(x) * \left(s\frac{d\theta_s(x)}{dx}\right) = s\frac{d}{dx}(f * \theta_s)(x) \qquad (1)$$

where

$$\theta_s(x) = \frac{1}{s}\theta\left(\frac{x}{s}\right).$$

Hence, the wavelet transform of the signal at scale s is equivalent (up to a constant) to taking the first derivative of $f * \theta_s$, the signal smoothed at scale s.

As elucidated in [31], the notion of viewing an image at different scales is very natural for its understanding and analysis. The role of the scale s determines how global or local the signal features are that we want to capture. At a given scale, an extremum point in $W_s f(x)$ of large magnitude has the physical meaning of locating a sharp transition region in $f * \theta_s$, while an extremum of small magnitude indicates a region of relatively slow variation. In the case that $\theta(x)$ is Gaussian, the detection of extrema points corresponds to the Canny edge detector [20]. Note that a Gaussian $\theta(x)$ is the unique function with the property of not creating additional spurious extrema points at larger scales [31]. Therefore, for edge characterization, it is important to choose a filter which is Gaussian or approximately Gaussian.

The extension of the multiscale edge detection to two dimensions is straightforward. Let $\theta(x, y)$ be a smoothing function which integrates to 1 and converges to zero at infinity, and let $\theta_s(x, y)$ denote the dilation of $\theta(x, y)$

$$\theta_s(x, y) = \frac{1}{s^2}\theta\left(\frac{x}{s}, \frac{y}{s}.\right).$$

The image $f(x, y)$ is smoothed by $\theta_s(x, y)$, and its gradient $\vec{\nabla}(f * \theta_s)(x, y)$ is computed. The direction of the gradient vector at (x, y) is the direction at which $f(x, y)$ has the sharpest variation. An edge point is defined to be a point (x_0, y_0) at which $|\vec{\nabla}(f * \theta_s)(x, y)|$ is the maximum along the direction of the gradient vector, and it is an inflection point of $f * \theta_s$.

To relate multiscale edges to the two-dimensional (2-D) wavelet transform, first define $\psi_s^1(x, y) = \partial\theta_s(x, y)/\partial x$ and $\psi_s^2(x, y) = \partial\theta_s(x, y)/\partial y$, respectively. The wavelet transform of $f(x, y)$ consists of two components $W_s^1 f(x, y) =$

$f * \psi_s^1(x,y)$ and $W_s^2 f(x,y) = f * \psi_s^2(x,y)$, and it is related to the gradient vector by

$$\begin{bmatrix} W_s^1 f(x,y) \\ W_s^2 f(x,y) \end{bmatrix} = s \begin{bmatrix} \frac{\partial}{\partial x}(f * \theta_s)(x,y) \\ \frac{\partial}{\partial y}(f * \theta_s)(x,y) \end{bmatrix} = s\vec{\nabla}(f * \theta_s)(x,y).$$

The singularities in images occur at points where the modulus

$$M_s f(x,y) = \sqrt{|W_s^1 f(x,y)|^2 + |W_s^2 f(x,y)|^2}$$

is maximum in the direction of the gradient vector. Modulus maxima form contours in the two-dimensional wavelet transform domain and an extension of the 1-D edge detector would require tracking the propagation of those contours across the scales. That process would add considerable complexity to the proposed interpolation algorithm; hence, we will consider a separable 2-D algorithm. That is, $W_s^1 f(x,y)$ and $W_s^2 f(x,y)$ will be treated separately, row by row and column by column, as in the 1-D case.

From the previous discussion, it is clear that the value of the wavelet transform at scale s measures the smoothness of the signal smoothed at scale s. Furthermore, a sharp variation induces a local extremum in the wavelet transform which propagates across scales. To illustrate, we return to Fig. 1 which shows a waveform and its wavelet transform at the dyadic scales $s = 2^j$, for $j = 1, 2, 3$. Each isolated singularity produces extrema points which propagate across scales, and the power law according to which the values these extrema evolve across the scales characterizes the local Lipschitz regularity of the function. In particular, a function $f(x)$ is uniformly Lipschitz α over an interval (a,b) if and only if there exists a constant $K > 0$ such that for all $x \in (a,b)$, the wavelet transform of $f(x)$ satisfies [21]

$$|W_s f(x)| \leq K s^{\alpha}. \tag{2}$$

If $f(x)$ is differentiable but not continuously differentiable at x_0, then it is Lipschitz 1 at x_0 and the corresponding wavelet transform maxima behave as $O(s)$ around x_0. The larger the α, the more regular or smooth the function is. If $f(x)$ is discontinuous but bounded in the neighborhood of x_0, then $\alpha = 0$ at x_0, and the corresponding maxima remain constant across the scales. On the other hand, the Dirac function has a negative Lipschitz exponent $\alpha = -1$, and it produces wavelet transform modulus maxima which evolve as $O(1/s)$.

A. Discretization Issues

For discrete processing, any continuous-time signal must also be sampled before being processed. Thus, a signal is measured at a finite resolution. Its wavelet transform can only be computed over a countable and finite range of scales. In many applications, it suffices to compute the wavelet transform at the dyadic scale $s = 2^j$ with $j = 1, 2, \ldots$, which also allows a fast discrete computation. The fast computation algorithm, the design of the discrete filters, and their relations with underlying continuous filters are well explained in [23], to which the readers are referred for more details. Here, only the necessary results and notations will be introduced.

Let the finest scale be $s = 1$, and the coarsest scale computed to be $s = 2^J$. Define a smoothing operator at scale $s = 2^j$ to be $S_{2^j} f(x) = f * \phi_{2^j}(x)$, $j = 0, 1, \ldots, J$, where

$$\phi_s(x) = \frac{1}{s}\phi\left(\frac{x}{s}\right)$$

and $\phi(x)$ is a function which satisfies certain properties such that the difference, or details, between $S_{2^j} f$ and $S_{2^{j+1}} f$ is $W_{2^j} f$, as defined in (1). Now let $D = \{d_n\}_{n \in \mathbb{Z}}$ be a discrete sequence such that there exists a (non-unique) continuous function $f(x) \in L_2(\mathbb{R})$ satisfying $S_1 f(n) = d_n$ for all $n \in \mathbb{Z}$. Hence, we assume that the underlying signal is the continuous function $f(x)$, but only the discretized version $S_1 f(n)$ is available for processing. For a particular class of wavelets, one can compute from the discrete sequence $D = \{S_1 f(n)\}_{n \in \mathbb{Z}}$ the uniform sampling of the wavelet transform of $f(x)$ at dyadic scales $s \geq 1$. Let the following notations denote these discrete samples $W_{2^j}^d f = \{W_{2^j} f(n+\epsilon)\}_{n \in \mathbb{Z}}$ and $S_{2^j}^d f = \{S_{2^j} f(n+\epsilon)\}_{n \in \mathbb{Z}}$ where ϵ is the shift due to convolution with ϕ_{2^j} and ψ_{2^j}. The set of signals $\left\{(W_{2^j}^d f)_{1 \leq j \leq J}, S_{2^J}^d f\right\}$ forms the discrete dyadic wavelet transform of $D = \{S_1 f(n)\}_{n \in \mathbb{Z}}$. Henceforth, the discussion will concern discrete sequences; thus, to simplify notation, the discrete sequence $f[n]$ will denote the samples $S_1 f[n]$, and $W_{2^j} f[n]$ will denote the discrete dyadic transform of $f[n]$ (note the omittance of the superscript d).

The forward discrete dyadic wavelet transform is characterized by two filters: a lowpass filter $h_0[n]$ and a highpass filter $h_1[n]$. Let $h_0^{(j)}[n]$ and $h_1^{(j)}[n]$ be the filters obtained by upsampling $h_0[n]$ and $h_1[n]$, respectively, by a factor of 2^j (i.e., inserting $2^j - 1$ zeros between the coefficients). The wavelet transform of a signal $f \in l_2(\mathbb{Z})$ can be computed through the convolution with $h_0^{(j)}[n]$ and $h_1^{(j)}[n]$ in a recursive manner

$$\begin{array}{rcl} W_{2^j} f & = & \frac{1}{\lambda_{j-1}} S_{2^{j-1}} f * h_1^{(j-1)} \\ S_{2^j} f & = & S_{2^{j-1}} f * h_0^{(j-1)} \end{array}, \qquad j = 1, 2, \ldots, J \tag{3}$$

where $S_1 f = f$, $h_0^{(0)} = h_0$, and $h_1^{(0)} = h_1$. Let the wavelet transform operator $\mathcal{W}$ denote the linear operator mapping f to $\{S_{2^J} f, W_{2^j} f, j = 1, \ldots, J\}$. The operator $\mathcal{W}$ can be implemented by the octave-band nonsubsampled filter bank, provided the multiplication with λ_j coefficients is incorporated appropriately. The multiplicative coefficients λ_j are needed to compensate for the deviation in the estimation of the Lipschitz regularity introduced by discretization. More specifically, the constants λ_j are found empirically so as to make the discrete time step function have Lipschitz regularity $\alpha = 0$. Obviously, the values of λ_j are dependent on the chosen wavelet. The quadratic spline filters are used for our work because they approximate coarsely the Gaussian function and its first derivative and they also can be used in fast implementation of the discrete dyadic wavelet transform [23]. These filters are shown in Fig. 2. Their coefficients and the associated constants λ_j are in Table I. For perfect reconstruction to be possible, it is necessary and sufficient that there exist a synthesis pair $\tilde{h}_0[n]$ and $\tilde{h}_1[n]$ which satisfy the perfect reconstruction condition

$$H_0(z)\tilde{H}_0(z) + H_1(z)\tilde{H}_1(z) = 1 \tag{4}$$

where $H_0(z)$, $H_1(z)$, $\tilde{H}_0(z)$, and $\tilde{H}_1(z)$ are the z-transform of the filters $h_0[n]$, $h_1[n]$, $\tilde{h}_0[n]$, and $\tilde{h}_1[n]$, respectively. The in-

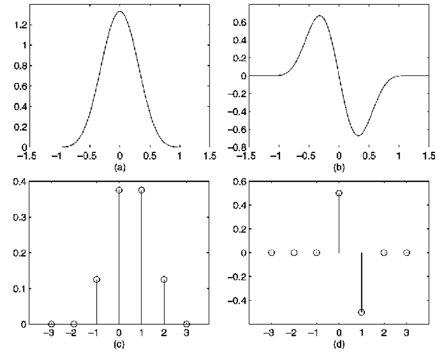


Fig. 2. Quadratic spline wavelet and smoothing function used in this work. The continuous-time smoothing function $\phi(x)$ in (a), and wavelet $\psi(x)$ in (b). The corresponding FIR coefficients of the smoothing function (lowpass filter $h_0[n]$) in (c), and of the wavelet (highpass filter $h_1[n]$) in (d).

TABLE I
FILTER COEFFICIENTS OF THE QUADRATIC SPLINE WAVELETS AND ASSOCIATED λ FACTORS

n	H_0	H_1	$\tilde{H}_0$	$\tilde{H}_1$	L	j	λ_j
-3				-0.001953125	0.0078125	0	1.0
-2			0.125	-0.01367125	0.046875	1	0.75
-1	0.125		0.375	-0.04296875	0.1171875	2	0.6875
0	0.375	0.5	0.375	0.04296875	0.65625	3	0.6719
1	0.375	-0.5	0.125	0.01367125	0.1171875	4	0.6680
2	0.125			0.001953125	0.046875	5	0.6670
3					0.0078125	6	0.6668

verse wavelet transform reconstructs the original signal by progressively adding finer and finer details onto the coarse residual signal $S_{2^J}f$. It can be calculated recursively as

$$S_{2^{j-1}}f = \lambda_j W_{2^j} f * \tilde{h}_1^{(j-1)} + S_{2^j} f * \tilde{h}_0^{(j-1)}, \qquad j = J, J-1, \ldots, 1 \tag{5}$$

where $\tilde{h}_0^{(0)} = \tilde{h}_0$ and $\tilde{h}_1^{(0)} = \tilde{h}_1$. The inverse wavelet transform operator $\mathcal{W}^{-1}$ can be implemented as a nonsubsampled synthesis octave band filter bank. Note again that the λ_j constants are needed to offset the scaling in the wavelet transform (3).

The discrete dyadic wavelet transform is a redundant representation of a function. An arbitrary set of sequences $\{g_j\}_{j=1,\ldots,J+1}$ is not necessarily the wavelet transform of some function f in $l_2(\mathbb{Z})$. It is the wavelet transform of some function $f \in l_2(\mathbb{Z})$ if and only if

$$\mathcal{W}(\mathcal{W}^{-1}(\{g_j\}_{j=1,\ldots,J+1})) = \{g_j\}_{j=1,\ldots,J+1}. \tag{6}$$

If the set of sequences $\{g_j\}_{j=1,\ldots,J+1}$ satisfies (6), then we say that it belongs to the range of the wavelet transform operator $\mathcal{W}$. The operator $\mathcal{W}\mathcal{W}^{-1}$ is thus the projection operator onto the range of the wavelet transform.

In practice, there are only finitely many, N, available samples of $f[n]$, which creates a problem at the boundary in the computation of the wavelet transform. To mitigate this problem, the signal is extended with mirror symmetry. This periodization avoids creating a spurious first-order discontinuity at the boundaries.

The 2-D forward and inverse wavelet transforms can be computed in a recursive manner similar to the 1-D case, implemented with the nonsubsampled filter banks shown in Fig. 3. For the 2-D wavelet transform, we choose separable filters, as shown in Fig. 3, where the 1-D filters H_0, H_1, $\tilde{H}_0$, and $\tilde{H}_1$ are the same as in the 1-D wavelet transform. For perfect reconstruction in the 2-D case, an additional filter L is needed, which satisfies $L(z) = (1 + H_0(z)\tilde{H}_0(z))/2$. Readers interested in more detail about this synthesis 2-D filter bank are referred to [23] and [32].

III. Interpolation Algorithm

The interpolation algorithm is first explained in one dimension for clarity. The discussion concentrates on magnification by a factor of two. Magnifications for other factors which are powers of two can be achieved by iteratively performing this algorithm. First, the main concepts will be introduced in Section III-A, and the details will be given in Section III-B. The 2-D algorithm will be developed in Section III-C.

A. Main Concepts of the Algorithm

The model of the interpolation problem is shown in Fig. 4. The available signal $\{f[n], n = 0, \ldots, N-1\}$ is modeled to be obtained from the high resolution signal $\{f_0[n], n = 0, \ldots, 2N-1\}$, which we wish to recover, by lowpass filtering using the filter $H_0(z)$ of our wavelet filter bank followed by downsampling by a factor of two. Naturally, one does not assume the exact knowledge of the lowpass filter used in the sampling process. We conjecture that as long as it is reasonable, the result of our algorithm will not depend strongly on the choice of filters, and that has been confirmed by the experimental results reported in Section IV. Further, we have at our disposal the highpass filter $H_1(z)$ such that $H_0(z)$ and $H_1(z)$ together with a synthesis pair $\tilde{H}_0(z)$ and $\tilde{H}_1(z)$, constitute a perfect reconstruction nonsubsampled filter bank. With this model, the goal of the interpolation algorithm is to estimate the signals f_u and g_u at the output of $H_0(z)$ and $H_1(z)$, and then reconstruct an estimate of f_0 via the synthesis filters. The algorithm consists of two stages: initial estimation and refinement.

1) Initial Estimation: An initial estimate $\hat{f}_u$ of the low-frequency component f_u can be obtained by simply interpolating f using, for instance, linear or spline interpolation. To find an initial estimate of the high-frequency component g_u, first notice that it contains information that would add sharpness to f. That is, if there were a sharp edge in the length $2N$ signal f_0, then the length N component f would contain a smoothed edge in this region. The reconstruction based solely on f_u would not be as sharp as the original edge in f_0. The information about the additional sharpness resides in g_u, whose essence is well captured by local extrema points, assuming that the filters used are appropriate for multiscale edge characterization. Thus, the central part of the initial estimation is to find the values and positions of the local extrema in g_u. The detailed procedures are illustrated in Fig. 5.

The first step in estimating g_u is to identify the edge regions via analysis of the available signal f. This identification is based on extracting local extrema of the wavelet transform of f which propagate across scales, and estimating the parameters in (2) which characterize this propagation. The knowledge of an edge location in f conveys knowledge about the edge location in g_u as well, up to a possible ambiguity of ±1 in location, since the

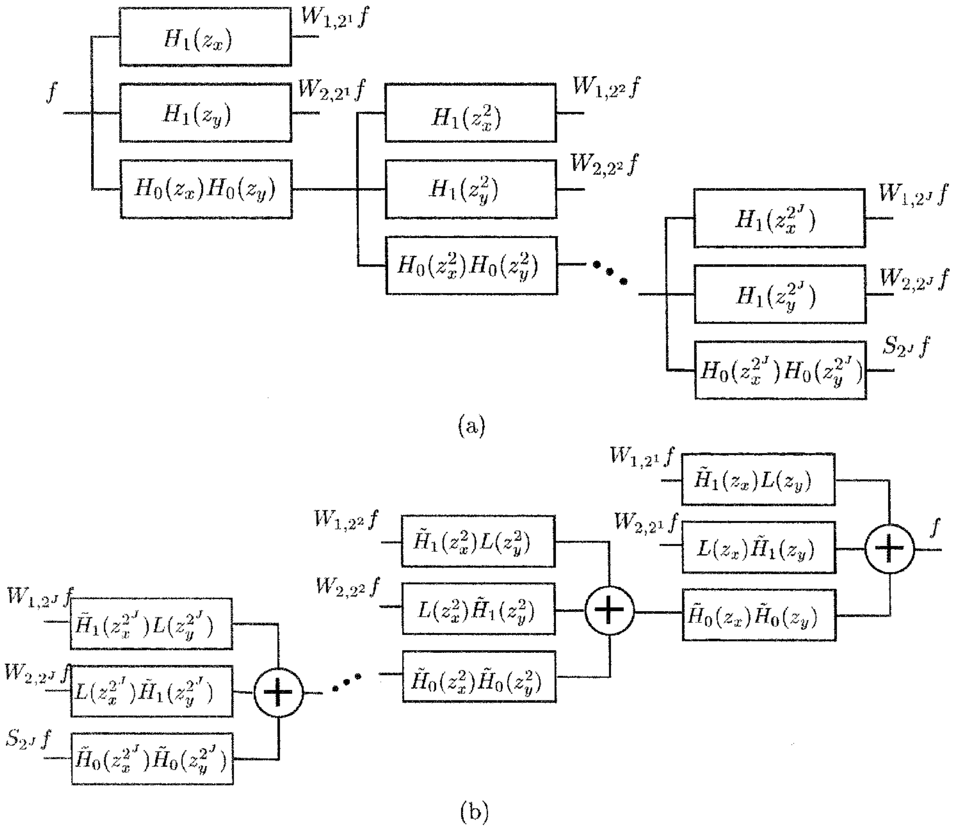

f
$H_1(z_x)$
$W_{1,2^1}f$
$H_1(z_y)$
$W_{2,2^1}f$
$H_0(z_x)H_0(z_y)$
$H_1(z_x^2)$
$W_{1,2^2}f$
$H_1(z_y^2)$
$W_{2,2^2}f$
$H_0(z_x^2)H_0(z_y^2)$
$H_1(z_x^{2^J})$
$W_{1,2^J}f$
$H_1(z_y^{2^J})$
$W_{2,2^J}f$
$H_0(z_x^{2^J})H_0(z_y^{2^J})$
$S_{2^J}f$
(a)
$\tilde{H}_1(z_x)L(z_y)$
$L(z_x)\tilde{H}_1(z_y)$
$\tilde{H}_0(z_x)\tilde{H}_0(z_y)$
$\tilde{H}_1(z_x^2)L(z_y^2)$
$L(z_x^2)\tilde{H}_1(z_y^2)$
$\tilde{H}_0(z_x^2)\tilde{H}_0(z_y^2)$
$\tilde{H}_1(z_x^{2^J})L(z_y^{2^J})$
$L(z_x^{2^J})\tilde{H}_1(z_y^{2^J})$
$\tilde{H}_0(z_x^{2^J})\tilde{H}_0(z_y^{2^J})$
(b)

Fig. 3. The 2-D discrete dyadic wavelet transform. (a) Forward transform. (b) Inverse transform.

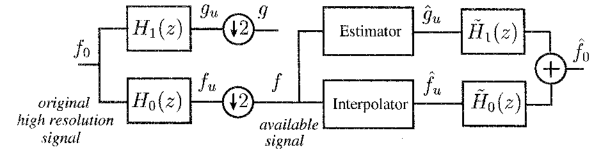


Fig. 4. Interpolation problem model for 1-D. The available signal f is modeled as the subsampled lowpass component of a higher resolution signal f_0, which is the desired signal.

wavelet transform of f is the decimated version, by a factor of two, of the wavelet transform of f_0 starting from the scale $s = 2^2$ (Fig. 6)

$$W_{2^j} f_0[2n] = W_{2^{j-1}} f[n], \qquad j = 2,3,\ldots. \qquad (7)$$

An edge information at $f[x_0]$ extracted from the analysis of $\{W_{2^j} f\}_{j=1,2,\ldots,J}$ and characterized in the parameters K and α of (2) translates to an edge at $f_0[2x_0]$. That is, an extremum in $W_{2^1} f_0$ can be estimated to be $W_{2^1} f_0[2x_0] = W_{2^0} f[x_0] = K$. Naturally, the downsampling operation in (7) introduces some ambiguity which needs to be addressed in the estimation process. More specifically, the true extrema points of $\{W_{2^j} f_0\}_{j=1,\ldots,J+1}$ may not have been sampled in the downsampling process. Thus, the edge identified at $f[x_0]$ may actually be at one of $\{f_0[2x_0 - 1], f_0[2x_0], f_0[2x_0 + 1]\}$. In Section III-B, we will discuss constraints which allow possible corrections of this ambiguity.

The edge characterization allows the estimation of significant extrema points of g_u. To obtain an initial estimate of g_u that may be closer to the real g_u, the points in between are then filled in by linearly interpolating between the extrema points.

2) Refinement by Alternating Projection: The initial estimates of f_u and g_u can be further refined by identifying constraints which they should obey. These constraints define convex sets and one can use the projection onto convex sets (POCS) method to find a solution existing in the intersection of these sets, called the reconstruction set. The POCS method alternately projects the signal onto the various convex sets until it converges to a solution in the reconstruction set (provided that it is nonempty). We identify three convex sets of constraints $\mathcal{V}$, $\mathcal{S}$, and $\mathcal{E}$, with the following meaning.

1) $\mathcal{V}$: The waveforms $\{\hat{f}_u, \hat{g}_u\}$ must belong to the range of the wavelet transform; we denote this subspace by $\mathcal{V}$.
2) $\mathcal{S}$: $\hat{f}_u$ must belong to a set $\mathcal{S}$, which comprises of length $2N$ signals whose downsampled version is consistent with f, the available signal.
3) $\mathcal{E}$: The edge points of f_0 (estimated from the analysis of f) should be reflected in local extrema of $\hat{g}_u$. $\mathcal{E}$ is comprised of signals whose structure is consistent with the edge information, and $\hat{g}_u$ should reside in $\mathcal{E}$.

The first two items are hard constraints in that they follow from the consistency of the problem model in Fig. 4. The third constraint is based on the estimation of how the signal should be at finer scales, and its purpose is to enhance the resolution of the reconstructed signal beyond that achieved by the first two constraints.

The projection operator $P_{\mathcal{V}}$ onto the subspace $\mathcal{V}$ is the operator in (6) and is pictorially illustrated in Fig. 7: it puts the pair

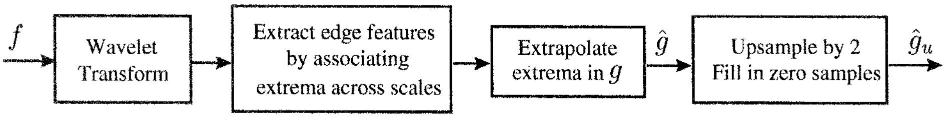

f
Wavelet Transform
Extract edge features by associating extrema across scales
Extrapolate extrema in g
$\hat{g}$
Upsample by 2 Fill in zero samples
$\hat{g}_u$

Fig. 5. Estimation of g_u based on f.

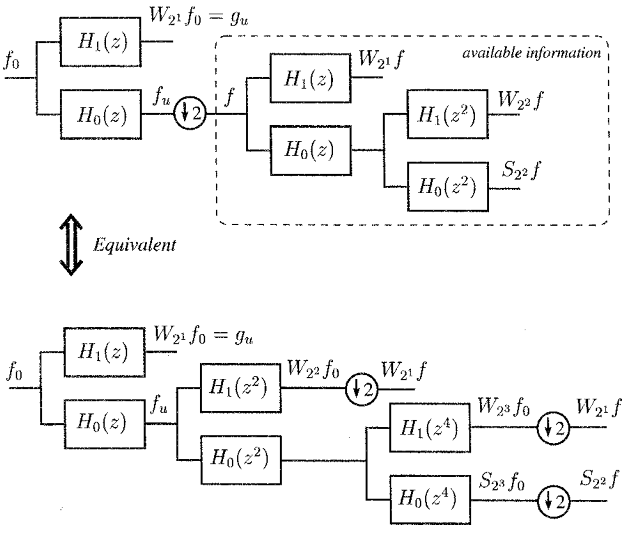


Fig. 6. Illustrating the equivalence between the wavelet transform of f and the decimated version of the wavelet transform of f_0 starting from scale $s = 2^2$.

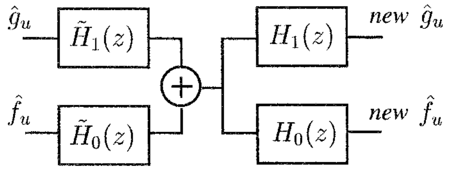


Fig. 7. Projection operator $P_{\mathcal{V}}$ onto the subspace $\mathcal{V}$, the range of the wavelet transform.

$(\hat{f}_u, \hat{g}_u)$ through the synthesis filter bank, followed by the analysis filter bank, where the filters obey the perfect reconstruction property in (4).

The projection operator $P_{\mathcal{S}}$ for the convex set $\mathcal{S}$ needs to ensure that $\hat{f}_u$ is consistent with the available signal f. At the very least, $\hat{f}_u[2n] = f[n]$ must hold. In practice, better performance could be achieved by placing restrictions, such as smoothness constraints, on the odd samples $\hat{f}_u[2n+1]$ as well, especially in regions of sharp variation. The details of this operator will be discussed in the implementation section.

The high-frequency component $\hat{g}_u$ must reside in the set $\mathcal{E}$, which consists of signals that are consistent with the estimated edge information. However, only the estimated edge information is available and, thus, one must allow some error tolerances. In Section III-B, we discuss the structure of the set $\mathcal{E}$ which allows varying degrees of leniency on the values and locations of the wavelet transform extrema, and finding a corresponding operator $P_{\mathcal{E}}$ which projects $\hat{g}_u$ onto $\mathcal{E}$.

The enhancement algorithm iteratively improves the estimates with the three projection operators $P_{\mathcal{V}}$, $P_{\mathcal{S}}$, and $P_{\mathcal{E}}$. Let $\{\hat{f}_u^{(0)}, \hat{g}_u^{(0)}\}$ denote the initial estimates of $\hat{f}_u$ and $\hat{g}_u$. At the end of the kth iteration, the estimates of $\hat{f}_u$ and $\hat{g}_u$ are $\{\hat{f}_u^{(k)}, \hat{g}_u^{(k)}\} = P_{\mathcal{E}}(P_{\mathcal{S}}(P_{\mathcal{V}}(\{\hat{f}_u^{(k-1)}, \hat{g}_u^{(k-1)}\})))$.

B. Implementation Details

The association of extrema points across scales and the characterization of Lipschitz regularities are not so simple when we deal with real data. Wavelet transform extrema points due to closely spaced sharp variations may interfere with each other and make association difficult. This interference also complicates the estimation of the parameters in (2), and these complications will be discussed below. The estimation of g_u will be elaborated, as well as the the exact structure of the sets $\mathcal{S}$ and $\mathcal{E}$ and their respective projection operators.

1) Associating Extrema Across Scales: To extrapolate the extrema points, we need to first select important singularities and associate the corresponding extrema points across scales. Since $W_{2^1} f$ contains an abundance of extrema which are not necessarily due to global structures, the extrema selection is, instead, done at a coarser scale $s = 2^2$. For each extremum at scale $s = 2^2$, the algorithm searches in the other scales for extrema associated with it.

Due to various reasons, not all extrema are observed to propagate from scale 2^j to 2^{j+1}. Extrema points at fine scales induced by closely spaced singularities may merge into one extremum point at coarse scales. Also, because the wavelet transform is discretized in both scale and space, one may not always observe the extrema points evolving across scales. For these reasons, it is sometimes difficult to associate the extrema points and, thus, some empirical rules are used. Suppose we are analyzing the mth singularity which induces extrema points at a location $x_m^{(j)}$ in scale $s = 2^j$. The values of $x_m^{(j)}$ are unknown except for $x_m^{(2)}$, since the association starts from $x_m^{(2)}$ in scale $s = 2^2$. We search in other scales in a small neighborhood around $x_m^{(2)}$ to find extrema points which obey the following rules. These extrema must be of the same sign and must all be maxima (or minima). Furthermore, it is reasonable to assume that the extrema values should not differ too much from scale to scale and that not all types of singularities are important for image quality; thus, we restrict the ratio between two extrema points of consecutive scales to be within a range $1/2^{1.5} \leq |W_{2^j} f[x_m^{(j)}]|/|W_{2^{j+1}} f[x_m^{(j+1)}]| \leq 2^{1.5}$. In this way we focus on singularities which correspond to Lipschitz regularity α [as defined in (2)]) in the range $\alpha \in (-1-\epsilon, 1+\epsilon)$ and allow some margin to account for possible merging of local extrema as well as for possibly missing an actual extremum due to downsampling. The choice of the range of α is not critical as long as it is not too large. In [25], the authors report results obtained by focusing only on step edges, that is, singularities which correspond to $\alpha = 0$, which simplifies the described algorithm and still exhibits good performance.

2) Estimating High-Frequency Component g_u: Let us first rewrite the relationship in (2) in discrete-time and explicitly show the dependence of the local Lipschitz parameters on the different singularities. This results in

$$W_{2^j} f[x_m^{(j)}] = K_m (2^j)^{\alpha_m}, \qquad j = 1, \ldots, J \tag{8}$$

where $x_m^{(j)}$ is the location of the local extremum at scale 2^j corresponding to the mth singularity, α_m is the Lipschitz regularity of f at the singular point, and K_m is a nonzero constant. The objective is to estimate K_m and α_m, and then extrapolate to an extremum point at scale $s = 2^0$ through estimating its location $x_m^{(0)}$ and value $W_{2^0} f[x_m^{(0)}]$. Recall that the relation between g_u, $W_{2^j} f_0$, and $W_{2^j} f$ is $W_{2^{j+1}} f_0[2n] = W_{2^j} f[n]$ and $g_u[2n] \overset{\text{def}}{=} W_{2^1} f_0[2n] = W_{2^0} f[n]$. Thus, this extrapolation provides the first step in obtaining an estimate of the high-frequency component $W_{2^1} f_0$ (or g_u) by first estimating $W_{2^0} f$.

For those singularities whose sequence of extrema, $W_{2^j} f[x_m^{(j)}]$ $j = 1, \ldots, J$ is available, the parameters α_m and K_m in (8) can be estimated via linear regression on

$$\log_2(W_{2^j} f[x_m^{(j)}]) = \log_2 K_m + j\alpha_m, \qquad j = 1, \ldots, J.$$

An initial estimate of the extremum point of the wavelet transform of f at scale 2^0 is then given by $W_{2^0} f[x_m^{(0)}] = \hat{K}_m = g_u[2x_m^{(0)}]$. The extrema location in scales $s = 2^0$ and $s = 2^1$ is assumed to be the same, that is, we let $x_m^{(0)} = x_m^{(1)}$.

The extrema extrapolation yields an estimate of the extrema positions and values in $g_u[2x_m^{(0)}]$. An initial estimate of the remaining points are obtained by linearly interpolating between consecutive extrema points.

3) Projection Operator $P_{\mathcal{S}}$ for $\mathcal{S}$: From the problem model in Fig. 4, it follows that $P_{\mathcal{S}}$ must, at the very least, assign $\hat{f}_u[2n] = f[n]$. In practice, this constraint alone does not prevent the spurious oscillations which often occur in sharp variation regions. To avoid this artifact, each odd sample $\hat{f}_u[2n+1]$ is bounded within an interval determined by the smoothness of $\hat{f}_u[2n]$ in that vicinity.

Let $\tilde{f}_u[n]$ be a length $2N$ bicubic spline interpolated version of $f[n]$. Also, let the discrete Laplacian gradient of $f[n]$ be defined as $\nabla f[n] = f[n] - (1/2)(f[n-1] + f[n+1])$. The upper bound on the odd samples of $\hat{f}_u$ is made to be $\mathrm{HI}_{f_u}[2n+1] = \tilde{f}_u[2n+1] + \varepsilon * (|\nabla f[n]| + |\nabla f[n+1]|)$. The value of $\varepsilon = .5$ was used. Similarly, the lower bound $\mathrm{LO}_{f_u}[2n+1]$ is calculated as $\mathrm{LO}_{f_u}[2n+1] = \tilde{f}_u[2n+1] - \varepsilon * (|\nabla f[n]| + |\nabla f[n+1]|)$.

To summarize, the operator $P_{\mathcal{S}}$ modifies $\hat{f}_u$ by assigning f to the even samples and bounding the odd samples to within the interval $[\mathrm{LO}_{f_u}[2n+1], \mathrm{HI}_{f_u}[2n+1]]$.

4) Projection Operator $P_{\mathcal{E}}$ for $\mathcal{E}$: Being the highpass component, the waveform $\hat{g}_u$ should reflect sharp variations in f_0. From the analysis of the wavelet transform of f, we have some knowledge of the extrema values and positions in $\hat{g}_u$. Hence, the set $\mathcal{E}$ can be thought of as the set of waveforms minimizing a specified cost function which penalizes when the extrema values do not conform to this knowledge. The operator $P_{\mathcal{E}}$ modifies $\hat{g}_u$ in a way such that the result has a lower cost.

This edge information, however, is estimated and, thus, prone to inaccuracy especially when using data containing more than just isolated singularities. The downsampling process introduces errors as well. Knowing that a certain set of points are edge points implies that the other points are not. Thus, one needs to prevent additional spurious edges being created during the reconstruction. With this in mind, there are various degrees of leniency that can be employed when constructing the cost function. We can either 1) constrain $\hat{g}_u$ to retain the initial estimates throughout the reconstruction; 2) allow the values to be within an allowable range; or 3) have no constraints at all on the values. Approaches 1) and 3) are extreme cases, assigning either infinite cost for wrong values or no cost at all. The allowed interval of approach 2) serves as a moderation, and yields better results. In the following, we will not construct explicitly an analytical cost function, but rather describe how $P_{\mathcal{E}}$ modifies the input to conform to the edge information.

Extrema Location: Because the initial estimate of $\hat{g}_u$ is obtained by interpolating from the estimate of the subsampled waveform g, the sampling may be such that we miss the true extrema and obtain instead the adjacent points. Thus, for each extremum of $\hat{g}_u$, the points immediately next to it are also allowed to be extrema points to account for this ambiguity. More specifically, if we initially determine $x_m^{(0)}$ to be an extremum point in the length-N signal $\hat{g}$ (which translates to location $2x_m^{(0)}$ in $\hat{g}_u$), then after the projection $P_{\mathcal{S}} P_{\mathcal{V}}$, $2x_m^{(0)}$ may not be an extremum point of $\hat{g}_u$ any longer. If the point of interest is a maximum (minimum) point, then the abscissa corresponding to the greatest (smallest) of $\{\hat{g}_u[2x_m^{(0)} - 1], \hat{g}_u[2x_m^{(0)}], \hat{g}_u[2x_m^{(0)} + 1]\}$ is assigned as the new local maximum (minimum).

Between Extrema Points: The points between adjacent extrema need also to be constrained to prevent spurious "edges." For example, by definition, the points between a pair of adjacent maximum and minimum points should have values bounded by these extrema values and, furthermore, the slopes of these in-between points should be monotonic so that there is no other extrema among them. Such a consistent reconstruction can be achieved by a simple algorithm proposed in [32] which reconstructs a signal from only its wavelet extrema points. For the interpolation problem, it has been found experimentally that these constraints are too restrictive for reconstructing g_u, since the extrema information is estimated and more leniency should be allowed. Therefore, "softer" constraints will be described.

In predicting the extrema points of g_u, only a subset of them could be extrapolated from the coarser scales due to the fact that coarser scales typically have fewer extrema than finer scales. Thus, for each extremum predicted in $\hat{g}_u$, we only assume that it is valid locally. For each maximum (minimum) examined, the points in a small neighborhood around it (a seven-point-centered window is used) are clipped to be less (greater) than or equal to this maximum (minimum) point. Since we are working with gray-scale images, another optimization is to clip all of the pixel values to be within [0,255]. These constraints are very lenient, and we prefer them over the more restrictive ones when analyzing real data, where it is difficult to ensure the robustness of capturing all of the extrema points. In our previous work in [24], we used strict constraints, such as bounding extrema values, to be within an estimated range, and enforcing monotonicity between consecutive extrema

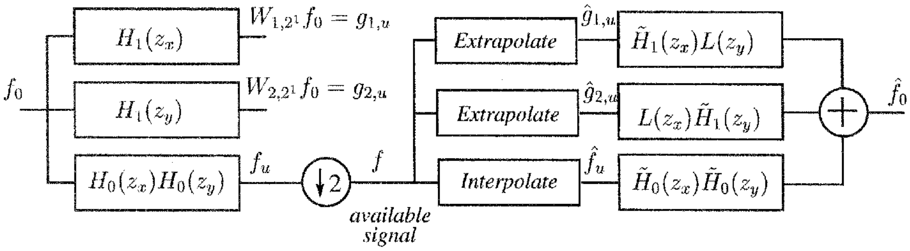


Fig. 8. Interpolation problem model for 2-D.

points. This sometimes resulted in images with some unpleasant artifacts, such as overly pronounced edges or small streaks. Here, we find that the softer constraints yield much more pleasant-looking results.

C. Enhancement Algorithm for 2-D Images

In general, analyzing a 2-D problem by treating the two coordinates independently is not an optimal approach. However, for computational reasons, we propose here to treat the two coordinates separately. The problem model for the 2-D case is analogous to the 1-D case, and is illustrated in Fig. 8 for clarity. To reiterate, the goal is to extrapolate from f information about f_u, $g_{1,u} \overset{\text{def}}{=} W_{1,2^1}f_0$, and $g_{2,u} \overset{\text{def}}{=} W_{2,2^1}f_0$, which are the necessary components of f_0.

1) Initial Estimates: In the wavelet transform, the data are filtered by the separable 2-D filter bank as discussed earlier. The wavelet transform generates the row components $\{W_{1,2^j}f\}_{j=1,\ldots,J}$, the column components $\{W_{2,2^j}f\}_{j=1,\ldots,J}$, and the low resolution component $S_J f$, all of which are $N \times N$. Bicubic spline interpolation is used to obtain the initial estimate of the size $2N \times 2N$ signal f_u. We observed that a better performance is achieved if bicubic spline interpolation is used for the initial estimate of f_u compared to pixel replication, bilinear interpolation, or interpolation by zeros. That is because the intersection of the convex constraints imposed by our algorithm consists of more than one image, so the alternating projections converge to different images depending on the starting point. Therefore, the initial estimate should be a more natural image. The ith row of $\{W_{1,2^j}f\}_{j=1,\ldots,J}$ is used to estimate the $2i$th row of the scale $s = 2^0$ row component as in the 1-D case. After interpolating this row to length $2N$, we have an initial estimate of the $2i$th row of $W_{1,2^1}f_0$. Having only extrema constraints on the even lines may result in jagged edges during the reconstruction process. To mitigate this artifact, we estimate the extrema of an odd row based on its two neighboring even rows. Typical images have smooth contours which traverse numerous rows or columns. Thus, for a given extremum on the $2i$th row, if there is an extremum on the $(2i+2)$th row which is of the same type (i.e., both maxima or both minima) and same sign, and is in a close proximity (within ± 4 pixels), then we assume there is an extremum of the same type and sign on the $(2i+1)$th row. The location and value are taken to be the average of the corresponding extrema on the neighboring rows. For simplicity, averaging is used rather than fitting a smoothed curve across these lines since the considered neighborhood is small, and the difference in location is not significant. A similar analysis is also done on the columns of $\{W_{2,2^j}f\}_{j=1,\ldots,J}$ to obtain an estimate of $W_{2,2^1}f_0$.

2) Alternating Projections: The estimates $\hat{f}_u$, $\hat{g}_{1,u}$, and $\hat{g}_{2,u}$ are iteratively refined using constraints analogous to those proposed in the 1-D case. The 2-D version of $P_{\mathcal{V}}$, $P_{\mathcal{S}}$, and $P_{\mathcal{E}}$ will be described.

The projection operator $P_{\mathcal{V}}$ is simply a one-level 2-D inverse wavelet transform followed by a one-level 2-D forward wavelet transform. The operator $P_{\mathcal{S}}$ first makes the assignment $\hat{f}_u[2n_1, 2n_2] = f[n_1, n_2]$ for the even samples. To constrain the odd samples, we define $\tilde{f}_u[n_1, n_2]$ to be a $2N \times 2N$ bicubic spline interpolated version of $f[n_1, n_2]$, and the discrete Laplacian gradient of f to be

$$\begin{aligned}\vec{\nabla} f[n_1, n_2] =& f[n_1, n_2] - \frac{1}{4}(f[n_1-1, n_2]+ \\ &+ f[n_1+1, n_2] + f[n_1, n_2-1] + f[n_1, n_2+1]).\end{aligned} \tag{9}$$

The upperbound on the samples of $f_u[n_1, n_2]$ is taken to be

$$\begin{aligned}\mathrm{HI}_{f_u}[n_1, n_2] =& \tilde{f}_u[n_1, n_2]+ \\ &+ w[n_1, n_2] * \text{Upsample}(|\vec{\nabla} f[n_1, n_2]|)\end{aligned} \tag{10}$$

where the second term is the convolution between a weighting function $w[n_1, n_2]$ depicted by the matrix

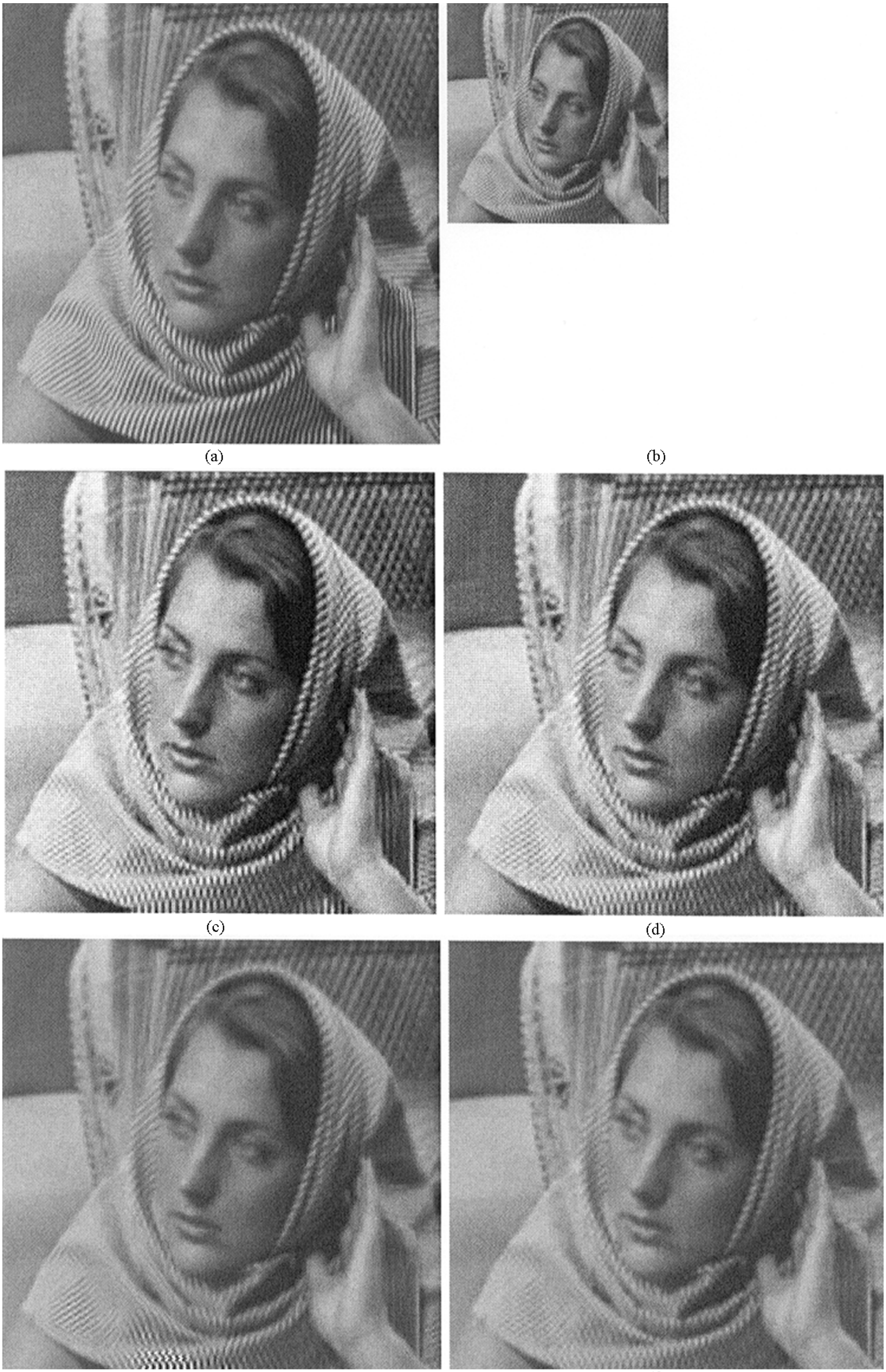
$$\begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}$$

and the upsampled version of $|\vec{\nabla} f[n_1, n_2]|$ (upsampled by a factor of 2 in each direction). The lower bound $\mathrm{LO}_{f_u}[n_1, n_2]$ is defined similarly, but with a subtraction substituting the addition in (10). The operator $P_{\mathcal{S}}$ then bounds $\hat{f}_u[n_1, n_2]$ to be within $[\mathrm{LO}_{f_u}[n_1, n_2], \mathrm{HI}_{f_u}[n_1, n_2]]$.

Each of the $2N$ available rows of $\hat{g}_{1,u}$ and the $2N$ available columns of $\hat{g}_{2,u}$ are treated as a separate 1-D problem, and are projected onto $\mathcal{E}$ using the 1-D operator $P_{\mathcal{E}}$ described in Section III-A.

IV. Experimental Results

The performance of the algorithm will be compared with several standard methods, such as bilinear interpolation, bicubic-spline interpolation, and bicubic-spline interpolation followed by unsharp masking and also with the edge-directed interpola-



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 9. Interpolation of the Barbara image, with the even-length low-pass filter $\varphi_1[n]$. (a) Original 256 × 256 image. (b) Lowpass, available image 128 × 128. (c) Wavelet-based interpolation. (d) Bicubic spline interpolation with unsharp masking and prefiltering. (e) Edge-directed interpolation. (f) Bicubic spline interpolation.

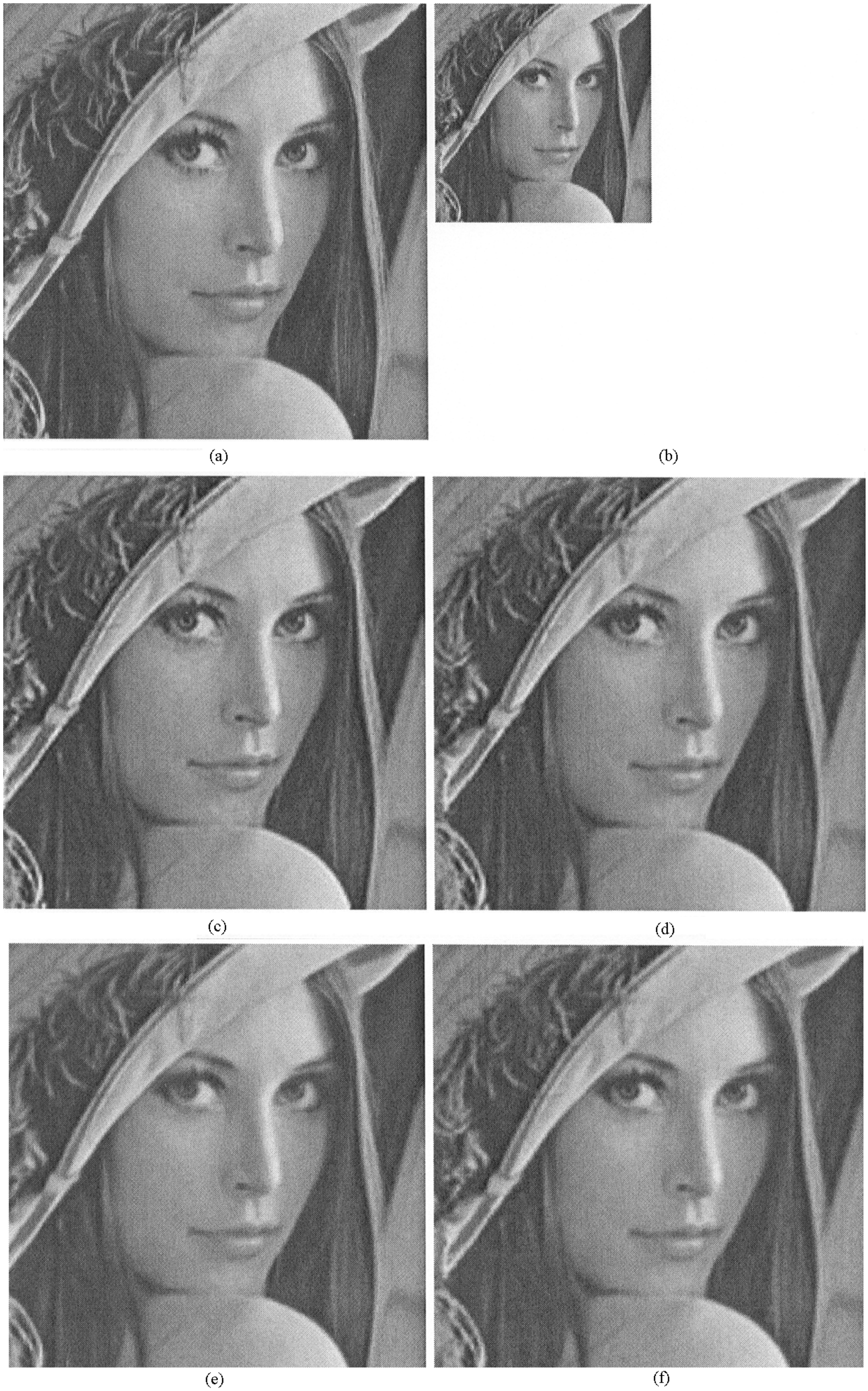


Fig. 10. Interpolation of the Lena image with the odd-length low-pass filter $\varphi_2[n]$. (a) Original 256 × 256 image. (b) Lowpass, available image 128 × 128. (c) Wavelet-based interpolation. (d) Bicubic spline interpolation with unsharp masking and prefiltering. (e) Edge-directed interpolation. (f) Bicubic spline interpolation.

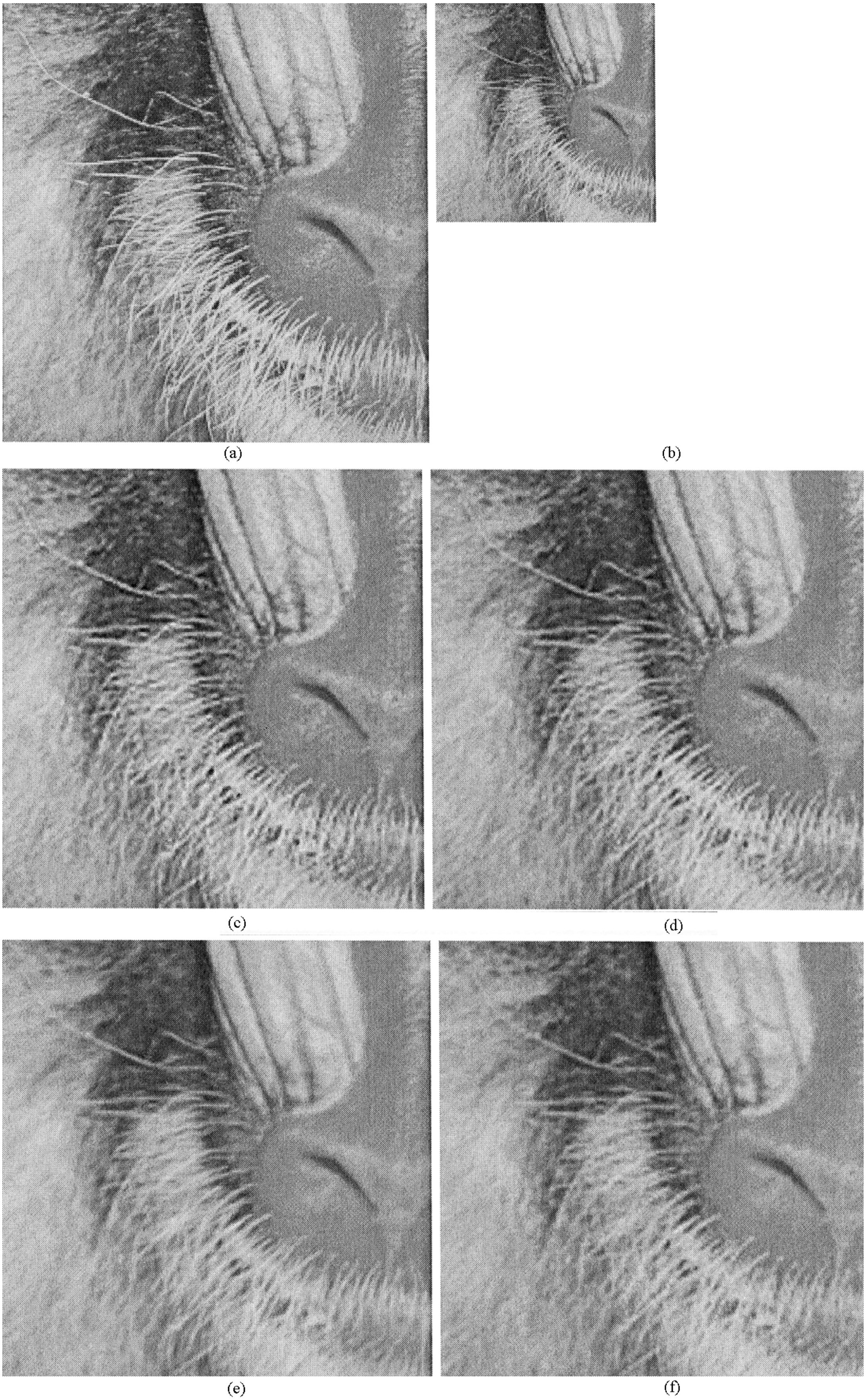
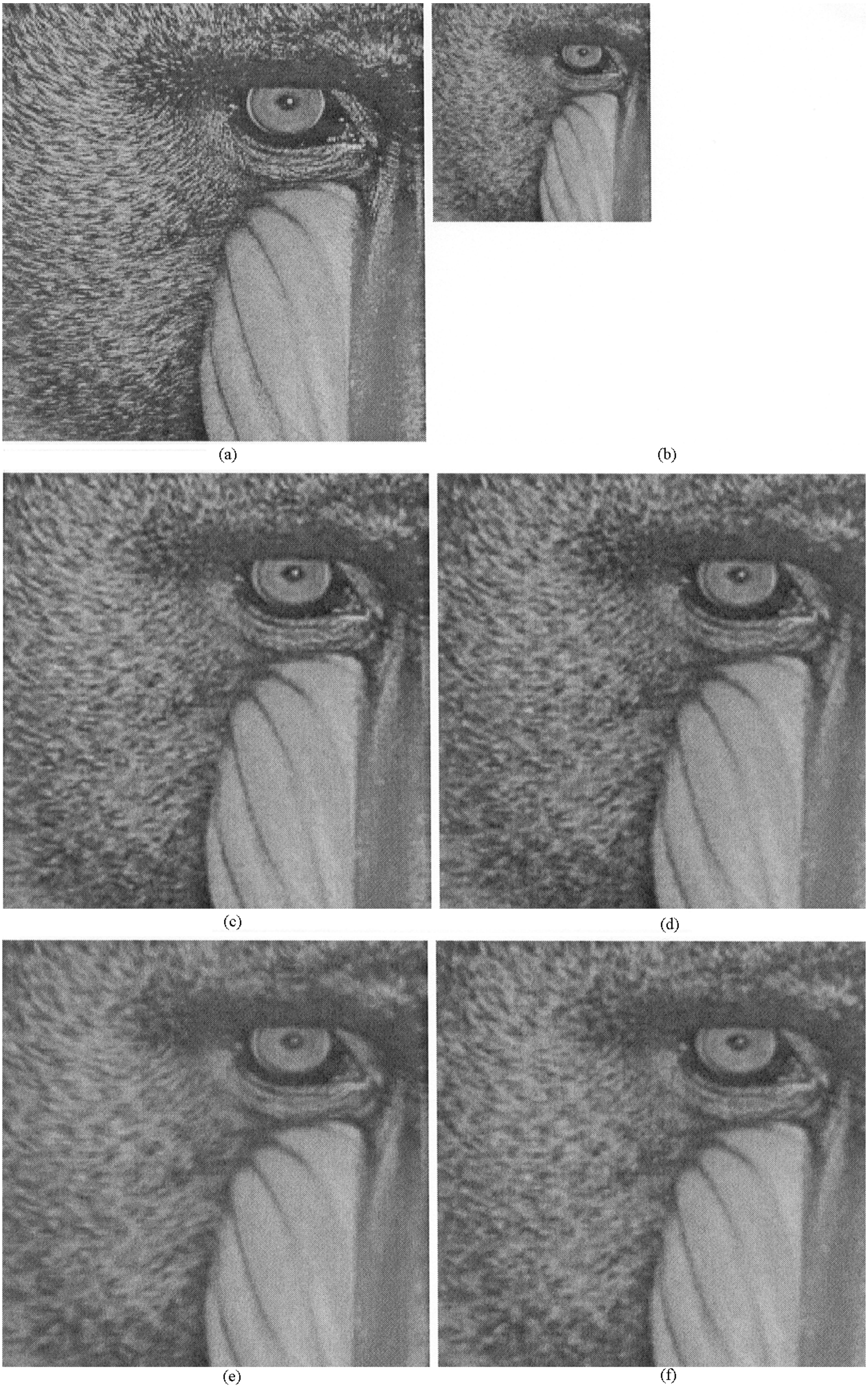
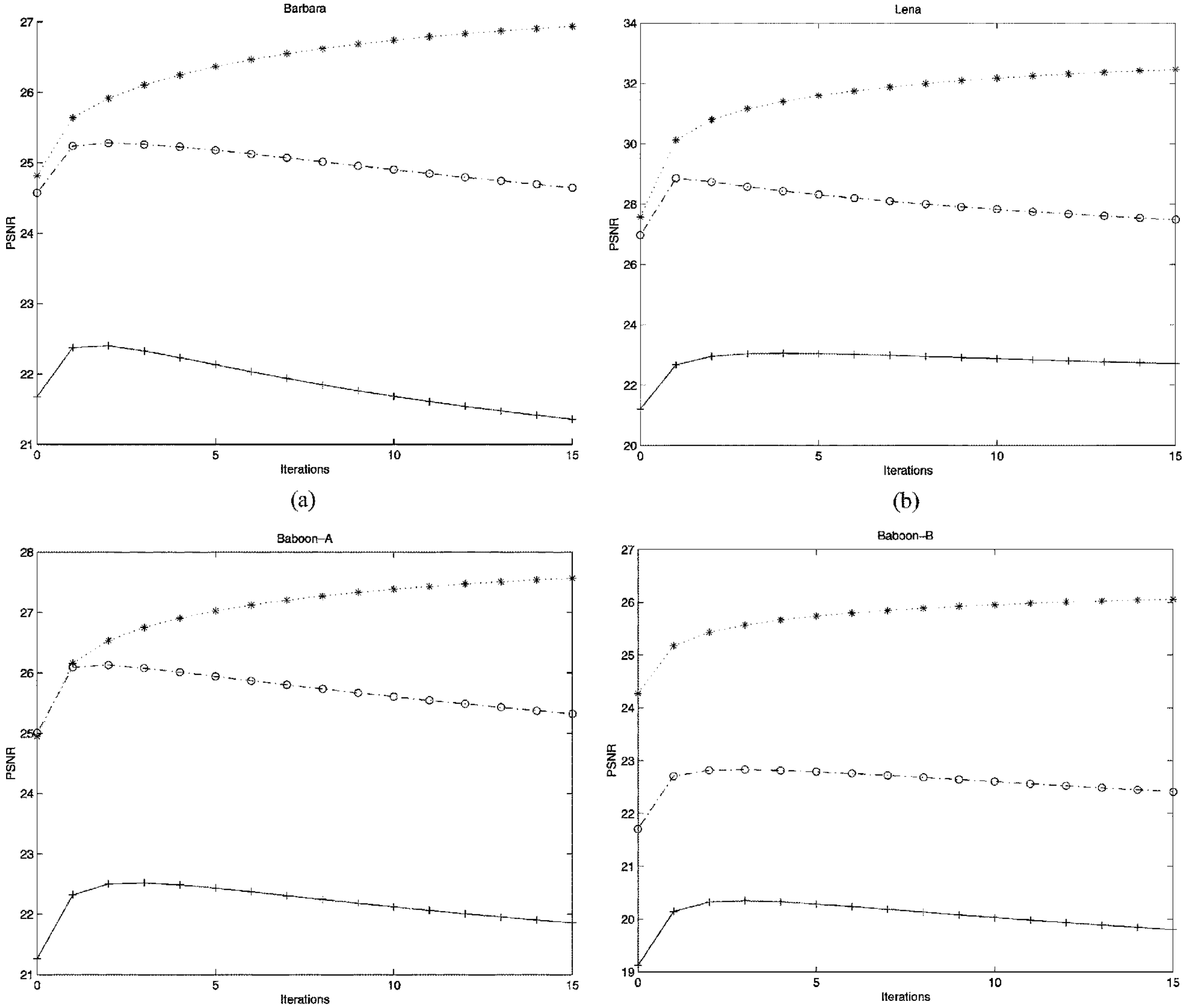


Fig. 11. Interpolation of the Baboon-A image, with the odd-length lowpass filter $\varphi_2[n]$. (a) Original 256 × 256 image. (b) Lowpass, available image 128 × 128. (c) Wavelet-based interpolation. (d) Bicubic spline interpolation with unsharp masking and prefiltering. (e) Edge-directed interpolation. (f) Bicubic spline interpolation.



(a) (b) (c) (d) (e) (f)

Fig. 12. Interpolation of the Baboon-B image, with the lowpass filter $\varphi_3[n] = h_0[n]$. (a) Original 256 × 256 image. (b) Lowpass, available image 128 × 128. (c) Wavelet-based interpolation. (d) Bicubic spline interpolation with unsharp masking and prefiltering. (e) Edge-directed interpolation. (f) Bicubic spline interpolation.



(a) (b) (c) (d)

Fig. 13. PSNR as a function of iterations. The curves are for images with φ_1 (+——+), φ_2 (o — · — o), and φ_3 ($*\cdots*$). (a) Barbara. (b) Lena. (c) Baboon-A. (d) Baboon-B.

tion proposed in [14]. Unsharp masking is a commonly used method for boosting the high-frequency portion of a signal. The general operation is to take the input $f[n_1, n_2]$ and yield $v[n_1, n_2] = f[n_1, n_2] + \lambda u[n_1, n_2]$ where $\lambda > 0$, and $u[n_1, n_2]$ is a defined gradient at location $[n_1, n_2]$. A commonly used gradient is the discrete Laplacian defined in (9), and a commonly used value for λ is 1. We consider two types of bicubic-spline interpolation: the method which involves a prefiltering step, as described in [33], so that the known pixels are reproduced exactly, and the traditional spline "interpolation" which consists of applying just the bicubic-spline filter. The filters used in the wavelet decomposition are given in Table I, and three levels of decomposition are computed.

In order to obtain peak signal-to-noise ratio (PSNR) measurements in addition to the visual judgement, we take a $2N \times 2N$ image, $f_0[n_1, n_2]$, filter it with some lowpass filter $\varphi[n_1, n_2]$, and downsample it to obtain the available $N \times N$ image $f[n_1, n_2]$. The choice of the filter $\varphi[n_1, n_2]$ is a parameter which we wish to test to see how sensitive the algorithm is with this choice. Each 2-D lowpass filter $\varphi[n_1, n_2]$ is a separable filter $\varphi[n_1, n_2] = \varphi[n_1]\varphi[n_2]$. The three choices of $\varphi[n]$ are $\varphi_1[n]$ – 12-tap symmetric lowpass filter generated by the MATLAB fir1(11, 0.5) function, $\varphi_2[n]$ – 11-tap symmetric lowpass filter generated by the MATLAB fir1(10, 0.5) function, and $\varphi_3[n]$ – the same filter $h_0[n]$ used in the wavelet analysis. The even-length filter has a delay of 1/2, while the odd-length filter has a delay of zero. The reason for choosing $\varphi_3[n]$ is to obtain a benchmark to see how well the algorithm can perform when we "cheat" by knowing the nature of degradation from f_0 to f.

The reconstructed images generally attain most of their quality in few iterations and practically do not change after 7–8 iterations, both in visual quality and in PSNR measurements. The measurements listed and images displayed are obtained after 15 iterations. The results obtained using only the initial estimate of the proposed algorithm and focusing only on step-edge singularities, as reported in [25], also demonstrate good performance compared to bilinear interpolation and bicubic spline interpolation without prefiltering and unsharp masking.

TABLE II
COMPARING PSNR OF DIFFERENT METHODS WHEN THE GIVEN IMAGE IS DOWNSAMPLED AFTER LOW-PASS FILTERING USING: (a) THE EVEN-LENGTH FILTER $\varphi_1[n]$, (b) THE ODD-LENGTH FILTER $\varphi_2[n]$, (c) THE FILTER $\varphi_3[n] = h_0[n]$

Image	Wavelet	Bicub.	Linear	Bicub. + UnsharpMask
Barbara	**21.36**	21.20	**21.36**	20.50
Lena	**22.71**	21.44	21.69	20.86
Baboon-A	21.86	21.66	**21.90**	20.91
Baboon-B	**19.80**	19.56	19.71	18.96

(a)

Image	Wavelet	Bicub.	Linear	Bicub. + UnsharpMask
Barbara	24.64	26.79	26.55	**27.34**
Lena	27.48	32.16	31.07	**32.69**
Baboon-A	25.32	27.37	26.98	**27.79**
Baboon-B	22.41	23.59	23.31	**23.92**

(b)

Image	Wavelet	Bicub.	Linear	Bicub. + UnsharpMask
Barbara	**26.94**	24.99	24.86	25.14
Lena	**32.46**	27.86	27.49	28.13
Baboon-A	**27.57**	25.53	25.35	25.72
Baboon-B	**26.06**	24.62	24.45	24.76

(c)

Each set of experiments consists of taking one of the four test images and one of the three $\varphi_i[n]$ lowpass filters, and interpolating the images using four different interpolation methods. Here, only one set of experiments for each test image will be shown for wavelet interpolation, edge-directed interpolation, traditional bicubic-spline interpolation, and bicubic-spline interpolation preceded by the prefiltering step and followed by unsharp masking. The Barbara experiment with filter $\varphi_1[n]$ is shown in Fig. 9, Lena with $\varphi_2[n]$ in Fig. 10, Baboon-A with $\varphi_2[n]$ in Fig. 11, and Baboon-B with $\varphi_3[n]$ in Fig. 12. The images are also posted at www.kcl.ac.uk/cdspr/zc/images/*name*.tiff, where *name* is Barbara, Lena, BaboonA, or BaboonB. Note that in the Barbara experiments, the interpolated images show aliasing on the scarfs. This is through no fault of the interpolation algorithms, but rather that the downsampling operation, used to obtain the test image f, already introduced aliasing. In all of the experiments, the wavelet interpolation approach yields images considerably sharper than those from linear, bicubic spline, or edge-directed interpolation, and also improves, but less markedly, over the bicubic-spline method used in combination with unsharp masking. We observed a barely noticeable effect of the prefiltering step on the visual quality of the interpolated images in the case of bicubic-spline interpolation. Unsharp masking, on the other hand, did make a difference.

Visually, experiments from the three different filters $\varphi_i[n]$ yield very similar results and conclusions, but the PSNR tells quite a different story. Although the PSNR is not a good indication of image quality, it is nevertheless frequently used, and the results are tabulated in Table II for the wavelet method, bilinear interpolation, bicubic spline interpolation, and bicubic spline interpolation with unsharp masking. The best numbers are highlighted in bold. The PSNR results depend on the choice of lowpass filter $\varphi_i[n]$. For the even-length filter $\varphi_1[n]$,the methods with the highest PSNR are either the wavelet or the linear method. When the odd-length filter $\varphi_2[n]$ is used, bicubic spline interpolation with unsharp masking yields the highest PSNR. With $\varphi_3[n] = h_0[n]$, not surprisingly, the wavelet approach yields the highest PSNR.

Fig. 13 shows the PSNR as a function of the iteration number for the images Barbara, Lena, Baboon-A, and Baboon-B. Each plot shows three curves, for the three choices of lowpass filter $\varphi_i[n]$. As mentioned previously, the reconstructed image remains visually indistinguishable after 7–8 iterations. The PSNR also shows quick convergence, though it is not always monotonically increasing. For filters $\varphi_1[n]$ and $\varphi_2[n]$, the PSNR actually decreases after the third or fourth iteration, but for $\varphi_3[n]$, it is monotonically increasing. The reason for the monotonic increase of PSNR when $\varphi_3[n] = h_0[n]$ is because, in that case, our model for the process in which the low-resolution image is obtained is correct, so the iterative procedure in every step decreases the distance between the interpolated image and the original, or an image which is very close to the original. On the other hand, when filters other than $h_0[n]$ are used to generate the low-resolution image, there is a discrepancy between the assumed model and the actual process, so the alternating projections may converge to images which are not as close to the original as they would have been had the model been accurate. Nevertheless, our experimental results demonstrate that the visual quality of the images which the algorithm converges to is not sensitive to the accuracy of the model and, therefore, the perceptual quality of interpolated images steadily improves toward a saturation level, despite the fact that the PSNR may decrease.

V. SUMMARY

We have proposed a wavelet-based method for image interpolation which attempts to capture and preserve sharp variations of different kinds. By characterizing edge points via the wavelet transform, we extrapolate the extrema needed at a finer scale for reconstruction of a higher-resolution image. The results show that the enhanced image is significantly sharper than simple schemes, such as linear and bicubic-spline interpolation, or edge-directed interpolation, and sharper, but less markedly, than bicubic-spline interpolation used in conjunction with unsharp masking. The better performance comes at the expense of significantly higher complexity. We believe, however, that the method leaves space for many modifications that can further improve its performance and reduce computational complexity, and that the ideas and concepts presented in this paper indicate a promising path for further research of image interpolation in the direction of wavelet-based locally adaptive schemes.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for their constructive criticism which improved the presentation of the paper.

REFERENCES

[1] V. R. Algazi, G. E. Ford, and R. Potharlanka, "Directional interpolation of images based on visual properties and rank order filterink," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 4, May 1991, pp. 3005–3008.

[2] B. Ayazifar and J. S. Lim, "Pel-adaptive model-based interpolation of spatially subsampled images," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, Mar. 1992, pp. 181–184.

[3] S. W. Lee and J. K. Paik, "Image interpolation using adaptive fast B-spline filtering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 5, Apr. 1993, pp. 177–180.

[4] K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," *IEEE Trans. Image Process.*, vol. 4, no. 3, pp. 285–295, Mar. 1995.

[5] J. E. Adams, Jr., "Interactions between color plane interpolation and other image processing functions in electronic photography," *Proc. SPIE*, vol. 2416, pp. 144–151, Feb. 1995.

[6] D. A. Florencio and R. W. Schafer, "Post-sampling aliasing control for natural images," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 2, May 1995, pp. 893–896.

[7] S. Carrato, G. Ramponi, and S. Marsi, "A simple edge-sensitive image interpolation filter," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sep. 1996, pp. 711–714.

[8] J. Allebach and P. W. Wong, "Edge-directed interpolation," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sep. 1996, pp. 707–710.

[9] K. Ratakonda and N. Ahuja, "POCS based adaptive image magnification," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Oct. 1998, pp. 203–207.

[10] B. S. Morse and D. Scwartzwald, "Isophote-based interpolation," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Oct. 1998, pp. 227–231.

[11] D. Calle and A. Montanvert, "Superresolution inducing of an image," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Oct. 1998, pp. 232–235.

[12] R. Fekri, R. M. Mersereau, and R. W. Schafer, "A general interpolative VQ method for jointly optimal quantization and interpolation of images," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 5, May 1998, pp. 2657–2660.

[13] N. V. Boulgouris and M. G. Strinzis, "Orientation-sensitive interpolative pyramids for lossless and progressive image coding," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 710–715, Apr. 2000.

[14] X. Li and M. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[15] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1167–1183, Sep. 2002.

[16] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graphics Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.

[17] A. Rosenfel and M. Thurston, "Edge and curve detection for visual scene analysis," *IEEE Trans. Comput.*, vol. C-20, no. 5, pp. 562–569, May 1971.

[18] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Royal Soc. London*, vol. 207, pp. 187–217, 1980.

[19] A. Witkin, "Scale-space filtering: a new approach to multi-scale description," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 9, Mar. 1984, pp. 150–153.

[20] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Nov. 1986.

[21] Y. Meyer, *Ondelettes et Operateurs*. New York: Hermann, 1990.

[22] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 617–643, Mar. 1992.

[23] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, no. 7, pp. 2207–2232, Jul. 1992.

[24] S. G. Chang, Z. Cvetković, and M. Vetterli, "Resolution enhancement of images using wavelet transform extrema extrapolation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 4, May 1995, pp. 2379–2382.

[25] W. K. Carey, D. B. Chuang, and S. S. Hemami, "Regularity-preserving image interpolation," *IEEE Trans. Image Process.*, vol. 8, no. 9, pp. 1293–1297, Sep. 1999.

[26] Y. Itoh, Y. Izumi, and Y. Tanaka, "Image enhancement based on estimation of high resolution component using wavelet transform," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Oct. 1999, pp. 489–493.

[27] F. Nicolier and F. Truchetet, "Image magnification using decimated orthogonal wavelet transform," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sep. 2000, pp. 355–358.

[28] D. D. Muresan and T. W. Parks, "Prediction of image detail," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sep. 2000, pp. 323–326.

[29] T. H. Reeves and N. G. Kingsbury, "Prediction of coefficients from coarse to fine scale in the complex wavelet domain," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, Jun. 2000, pp. 508–511.

[30] K. Kinebuchi, D. D. Muresan, and T. W. Parks, "Image interpolation using wavelet-based hidden Markov trees," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, May 2001, pp. 1957–1960.

[31] J. J. Koenderink, "The structure of images," *Biol. Cybern.*, vol. 50, pp. 363–370, 1984.

[32] Z. Cvetković and M. Vetterli, "Discrete-time wavelet extrema representation. Design and consistent reconstruction," *IEEE Trans. Signal Process.*, vol. 43, no. 3, pp. 681–693, Mar. 1995.

[33] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Trans. Med. Imag.*, vol. 19, no. 7, pp. 739–758, Jul. 2000.

S. Grace Chang received the B.S. degree from the Massachusetts Institute of Technology (MIT), Cambridge, in 1993, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of California, Berkeley, in 1995 and 1998, respectively,

She has been with Hewlett-Packard Co. since 1998, first at Hewlett-Packard Laboratories, Palo Alto, CA, and now as a Business Program Manager in telecommunication software in Asia Pacific, Hewlett-Packard Taiwan Ltd., Taipei, Taiwan, R.O.C.

Zoran Cvetković (SM'04) received the Ph.D. degree in electrical engineering from the University of California, Berkeley, the M.Phil. degree from Columbia University, New York, NY, and the Mag. and Dipl.Ing. degrees from the University of Belgrade, Belgrade, Yugoslavia.

Currently, he is a Reader in Signal Processing with King's College London, London, U.K. Before that, he was with the Division of Engineering and Applied Sciences, Harvard University, and from 1997 to 2002, he was with AT&T Shannon Laboratory, Florham Park, NJ. He was also involved with the research activities of the Keck Center for Integrative Neuroscience, University of California, San Francisco, and the Department of Electrical Engineering, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. He also conducted research and taught at the Department of Electrical Engineering, University of Belgrade. His research interests include time-frequency and time-scale signal analysis, audio and speech technology, image processing, source coding, and signal processing for telecommunications.

Martin Vetterli (F'95) received the M.S. degree from Stanford University, Stanford, CA, and the Ph.D. degree from École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

In 1986, he joined Columbia University, New York, NY, first with the Center for Telecommunications Research and then with the Department of Electrical Engineering, where he was an Associate Professor of Electrical Engineering. In 1993, he joined the University of California, Berkeley, where he was Full Professor until 1997. Currently, he is a Professor with EPFL, where he headed the Communication Systems Division from 1996 to 1997 and heads the Audiovisual Communications Laboratory. From 2001 to 2004, he directed the National Competence Center in Research on mobile information and communication systems. He is also a Vice-President for International Affairs at EPFL since 2004. He has held visiting positions at Eidgenössische Technische Hochschule Zürich (ETHZ), Zürich, Switzerland, in 1990 and Stanford University in 1998. His research interests include sampling, wavelets, multirate signal processing, computational complexity, signal processing for communications, digital video processing, and joint source/channel coding.