ADAPTIVE, OPTIMAL-RECOVERY IMAGE INTERPOLATION

D. Darian Muresan and Thomas W. Parks

School of Electrical and Computer Engineering
Cornell University
darian, parks@ee.cornell.edu

ABSTRACT

We consider the problem of image interpolation using adaptive optimal recovery. We adaptively estimate the local quadratic signal class of our image pixels. We then use optimal recovery to estimate the missing local samples based on this quadratic signal class. This approach tends preserve edges, interpolating along edges and not across them.

1. INTRODUCTION

Image interpolation is becoming an increasingly important topic in digital image processing, especially as consumer digital photography is becoming ever more popular. From enlarging consumer images to creating large artistic prints, interpolation is at the heart of it all. It has been known for some time that classical interpolation techniques such as linear and bi-cubic interpolation are not good performers since these methods tend to blur and smooth edges.

Wavelets have been successfully used in interpolation [1, 4, 6]. These methods assume the image has been passed through a low pass filter before decimation and then try to estimate the missing details, or wavelet coefficients from the low resolution scaling coefficients. One drawback to these approaches is that they assume the knowledge of the low pass filter.

Directional interpolation algorithms try to first detect edges and then interpolate along edges, avoiding interpolation across edges [5]. In this class, there are algorithms that do not require the explicit detection of edges. Rather, the edge information is built into the algorithm itself. For example, [3] uses directional derivatives to generate weights used in estimating the missing pixels from the neighboring pixels. In [2], the local covariance matrix is used for estimating the missing pixels. This interpolation tends to adjust to an arbitrarily oriented edge.

In this paper we present a new directional interpolation technique based on optimal recovery. The results of our interpolation approach can be thought of as an extension to [2]. In regions of high frequency our approach provides slightly better results than [2] and in some cases outperforms [9].

This work was supported by NSF MIP9705349, TI and Kodak

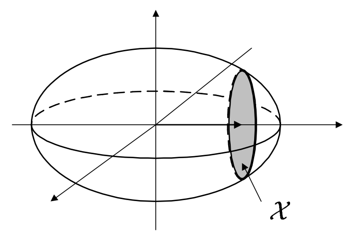

Fig. 1. Geometric Diagram of Ellipsoid Class

2. OPTIMAL-RECOVERY

In this section we briefly review the theory of optimal recovery as applied to the interpolation problem [8]. We then apply this theory to develop a new adaptive approach to image interpolation. The interpolation problem may be viewed as a problem of estimating missing samples of an image. This latter problem can be examined using the theory of optimal recovery. The theory of optimal recovery provides a broader setting, which illuminates the process of interpolation, by providing error bounds and allowing calculation of worst-case images which achieve these bounds.

Locally, at location y (Fig. 2), we model the image as belonging to a certain ellipsoidal signal class K

$$K = \{x \in R^n : x^T \mathbf{Q} x \leq \epsilon\} \tag{1}$$

where $\mathbf{Q}$ is derived from the local image pixels as shown in section 3. Vector x is any subset of the image containing the missing pixel y. Vector x is chosen such that **any** L linear functionals ($F_i, i = 1, \ldots, L$) of x are assumed known. If we note the actual values of the functionals by f_i we have $F_i(x) = f_i$. In this paper we assume that the functionals are based on derivatives and/or actual pixel values of the decimated image. The known functionals F_i, in the local image, determine a hyper-plane $\mathcal{X}$ (Fig. 1).

The intersection of the hyper-plane and ellipsoid is a hyper-circle in $\mathcal{X}$. The intersection depends upon the known functionals of the local image and we call it C_x. Formally,

$$C_x = \{x \in \mathcal{X} : F_i(x) = f_i, \|x\|_Q \leq 1\}, \qquad (2)$$

For a linear mapping $\mathbf{U}$, the image of C_x under $\mathbf{U}$ is the range of values that $\mathbf{U}f$ can take. The optimal recovery problem is to select the value in $\mathcal{X}$ which is a best approximation over all $\mathbf{U}f$ in $\mathbf{U}C_x$. We want to minimize

$$\delta = \max_{x^T Q x} |\hat{y} - y|$$

The Chebyshev center achieves this minimization. The Chebyshev center has been shown to be the minimum Q-norm signal on the hyper-plane determined by the known samples. The solution to this problem is well-known: see [8, 7].

If the collection of known functionals is f_i, the minimum norm signal is $\bar{u}$. Signal $\bar{u}$ is the unique signal in $\mathcal{X}$ with the property,

$$\|\bar{u}\|_Q = \inf_{F_i(x)=f_i} \|x\|_Q \qquad (3)$$

Our estimates signal $\bar{u}$ must satisfy $F_i(\bar{u}) = f_i$ and we are estimating $F(\bar{u}) = f$. As shown in [8] there exist vectors $\phi, \phi_1, \ldots, \phi_L$ such that $F(\hat{u}) = (\phi, \bar{u})_Q$ and

$$F_i(\hat{u}) = (\phi_i, \bar{u})_Q \qquad (4)$$

where the parentheses denote a Q dot product. Vectors ϕ_i are known as the representors. From [8] the solution is given by

$$\bar{u} = \sum_{i=1}^{L} \alpha_i \phi_i \qquad (5)$$

where the constants α_i are determined from the constraint of equation (4).

An advantage of this approach is not only that we can minimize the distance $\delta = \max_{x^T Q x} |\hat{y} - y|$, but we also obtain bounds on the maximum error δ and we can find the image which achieves this maximum error.

We now deal with the problem of determining Q adaptively from the image data. To make this explanation as simple and as straight forward as possible, we demonstrate our method with a simple toy example.

3. ADAPTIVE, OPTIMAL-RECOVERY INTERPOLATION

Our adaptively determined quadratic signal class, or Q, will be a measure of how well the local data matches the already known functionals F_i. We want to find an adaptive signal class K of the form:

$$K = \{x \in R^n : x^T \mathbf{Q} x \leq \epsilon\} \qquad (6)$$

To best understand this process, let's look at Fig. (2). In this small example, the problem is that of estimating pixel y. Our first step is to choose a signal x that contains the missing pixel y. For reasons that will be clear in a moment, let

a_1		a_2		b_1		b_2
	y_1		y_2		y_3	
a_4		a_3		b_4		b_3
	y_4		y		y_6	
d_1		d_2		c_1		c_2
	y_7		y_8		y_9	
d_4		d_3		c_4		c_3

Fig. 2. Interpolate pixel y. The only known pixels are the gray pixels.

$$x = [y_1\ y_2\ y_3\ a_3\ b_4\ y_4\ y\ y_6\ d_2\ c_1\ y_7\ y_8\ y_9]^T$$

Next, we assume that there exists weight $w_1, \ldots, w_4$ such that locally, each pixel can be estimated by the weighted sum of the four closest diagonal pixels. With

$$\|\hat{e}\|^2 = \min_{\mathbf{w}} \|e\|^2$$

our measure of how well the data matches the nearby points is

$$\hat{\mathbf{e}} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ a_3 \\ b_4 \\ y_4 \\ y \\ \vdots \end{pmatrix} - \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_2 & b_1 & b_4 & a_3 \\ b_1 & b_2 & b_3 & b_4 \\ a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_4 & a_3 & d_2 & d_1 \\ a_3 & b_4 & c_1 & d_2 \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{pmatrix} \qquad (7)$$

or equivalently

$$\hat{\mathbf{e}} = \mathbf{x} - \mathbf{\Psi}\mathbf{w}$$

The norm squared of $\hat{\mathbf{e}}$ is given by

$$\hat{\mathbf{e}}^T \hat{\mathbf{e}} = \mathbf{x}^T \left[I - \Psi \left(\Psi^T \Psi \right)^{-1} \Psi^T \right] \mathbf{x}$$

Thus,

$$\mathbf{Q} = \left[I - \Psi \left(\Psi^T \Psi \right)^{-1} \Psi^T \right] \qquad (8)$$

The problem of finding estimate $\hat{y}$ which minimizes $\delta = \max_{x^T Q x} |\hat{y} - y|$ is equivalent to finding $\mathbf{x}$ which minimizes $\mathbf{x}^T \mathbf{Q} \mathbf{x}$ given the known functionals F_i.

Assuming $\mathbf{\Psi}$ is full rank, matrix $\mathbf{Q}$ has four zero eigenvalues with the rest of them being all one. The null space of $\mathbf{Q}$ is spanned by the column vectors of $\mathbf{\Psi}$. At first glance, it seems that the solution to this problem might be any vector in the null space, since that will give zero error. That however, is not true since the solution must also satisfy the given functionals. Unless there are **only** four known functionals, our solution will not be in the null space of $\mathbf{Q}$.

Known functionals can be pixel values of the decimated image, derivative assumptions or any other linear functionals of the high resolution image. In our toy example, the linear functionals are the given pixels a_3, b_4, c_1, d_2 and assumptions about the derivatives. In particular, we look at the derivatives in the directions $d_2 - b_4$ and $a_3 - c_1$. We chose the direction with the smallest change and assume that the derivatives of the unknown pixels in that direction are equal with the derivatives of the known pixels in the same direction. For example, if $b_4 - d_2$ has the smallest difference, the derivative based functionals would be

$$y_3 - y = b_2 - b_4 \text{ and } y_7 - y = d_4 - d_2$$

When the known functionals are only the decimated pixel values, this method simplifies to the method presented in [2].

The formulation of our problem and the adaptive $\mathbf{Q}$ matrix is also quite useful when we assume that the image went through a low pass filter, before decimation. In this case, our assumption is that the pixel values of Fig. 2 are samples of the filtered image. If we let H be a filtering matrix and we assume that the image before filtering is z

$$x = Hz \tag{9}$$

then $\mathbf{Q}$ of (8) becomes $\mathbf{Q} = H^T Q H$. The new $\mathbf{Q}$ will still have four zero eigenvalues, but the other eigenvalues will no longer be one.

The approximation of the signal class, and therefore the interpolation results, can be further improved by an iterative process as follows:

1. Interpolate the missing pixels with the method described above.

2. Using the interpolated pixels, return to equation (2) and add the calculated pixels at the higher resolution as extra functionals.

We haven't proved convergence here, but from our experimental results, repeating this process three times seems to be enough.

4. RESULTS

In obtaining our results we first started with a high resolution image. We then filtered the higher resolution image by a low pass filter (Daubechies 1), to simulate camera lenses, and decimated by two. We then reconstructed the image using different interpolation approaches.

For our results section we have compared the adaptive, optimal recovery image interpolation algorithm against the algorithm presented in [2], against bi-cubic interpolation and against a commercially available algorithm [9]. When compared against [2] the algorithm outperformed slightly, especially around sharp and/or thin edges. The algorithm always outperformed bi-cubic interpolation. When compared against [9] there were places where the adaptive, optimal-recovery interpolation outperformed [9], but there were also places where it under-performed. Some sample images are included at the end of this section, but the reader is encouraged to view TIF images at
ww.ee.cornell.edu/˜splab.

Finally, we would like to thank Xin Li for providing us with his interpolation algorithm [2].

5. REFERENCES

[1] D. Darian Muresan and Thomas W. Parks, "Prediction of Image Detail," *IEEE ICIP,* Vancouver, Sept. 2000.

[2] Xin Li and Michael T. Orchard, "New Edge Directed Interpolation," *IEEE International Conference on Image Processing,* Vancouver, Sept. 2000.

[3] Ron Kimmel, "Demosaicing: Image Reconstruction from Color CCD Samples," *IEEE Transactions on Image Processing,* Vol. 8, No. 9, (September 1999), pp. 1221-1228.

[4] W. Knox Carey, Daniel B. Chuang and Sheila S. Hemami, "Regularity-Preserving Image Interpolation," *IEEE Transactions on Image Processing,* Vol. 8, No. 9 (Sept. 1999), 1293-1297.

[5] J. Allebach and P. W. Wong "Edge-directed Interpolation," *IEEE Proceedings of ICIP,* 1996, pp.707-710.

[6] S. Grace Chang, Zoran Cvetkovic, and Martin Vetterli, "Resolution Enhancement of Images Using Wavelet Transform Extrema Interpolation," *IEEE ICASSP,* (May 8-12, 1995), pp. 2379-2382.

[7] C. A. Michelli and T. J. Rivlin, " A Survey of Optimal Recovery," in *Optimal Estimation in Approximation Theory,* C. A. Michelli and T. J. Rivlin, Eds. New York: Plenum 1976, pp. 1-54.

[8] M. Golomb and H. F. Weinberger, "Optimal approximation and Error Bounds," *On Numerical Approximation,* R. E. Langer ed., The University of Wisconsin Press, Maddison, pp. 117-190, 1959.

[9] Commercial, Fractal Based Interpolation Algorithm (www.genuinefractals.com)

Fig. 3. Altamira (top), Cubic (center), Optimal-Recovery (bottom).

Fig. 4. Altamira (top), Cubic (center), Optimal-Recovery (bottom).