Optimal Prefiltering for Improved Image Interpolation*

Jeffery R. Price† Monson H. Hayes III
Center for Signal and Image Processing
School of Electrical and Computer Engineering
Georgia Institute of Technology
price@ece.gatech.edu

Abstract

In this paper we derive an optimal (MMSE) prefilter for image interpolation. This derivation is based upon a model of the sensor used to capture the image. To employ this model, we restate the interpolation problem in an intuitive, reconstruction-like fashion. Using a simple CCD sensor model, an example prefilter is derived. Simulations with this prefilter are performed using linear and cubic interpolation as well as an ad hoc, directional interpolation scheme. Quantitative and subjective results indicate that prefiltering generally improves the quality of the interpolated images.

1. Introduction

Digital image interpolation is a fundamental component in a variety of technologies today. Some common examples include digital zoom in CCD based video cameras, color plane interpolation for single CCD digital cameras, printing (reasonably sized) low-resolution images on high-resolution printers, and conversion between various television formats. For ease of implementation and speed of computation, conventional techniques such as nearest neighbor, linear, cubic and spline interpolation have been widely used in the past [1]-[4]. With the explosive growth in computational power, however, more advanced techniques have received significant attention. A few examples can be found in [5]-[14]. Many of these techniques use adaptive processing. Some information about the local image content is gathered and used to drive the interpolation process. Few methods, however, have proposed to exploit any knowledge of the image capture device. If there exists some knowledge of the capture sensor, we would expect that such information could be used to improve interpolation. In this paper, we consider the application of a sensor model to conventional (e.g., linear and cubic) interpolation and to a simple directional interpolation scheme.

We first mention some previous interpolation algorithms that have employed sensor models and describe how our method differs from these. In [15], an optimal interpolation kernel is proposed that minimizes a deterministic, sum-of-squares quantity. This filter attempts to reconstruct points of the continuous image that result from the convolution of the original (continuous) scene with the sensor PSF. In other words, the given low-resolution pixels are blurred samples of some continuous scene, and the interpolation filter seeks to produce additional *blurred* samples of this scene. In [14], the authors employ a sensor model (in a deterministic fashion) to iteratively improve their edge-directed interpolation. Our approach differs from each of these. Unlike [15], we seek *deblurred* samples of a higher-resolution (but not continuous) image that is related to the observed image by a sensor model. Contrary to both [14] and [15], we adopt a stochastic approach to account for sensor effects. Additionally, our approach differs from [14] in that it is not iterative. (It is certainly possible, however, that a prefilter might serve as an initial step in such an iterative approach.)

The remainder of this paper is organized as follows. In Section 2, we restate the interpolation problem to include a sensor model. We then derive an expression for an optimal prefilter based upon this model and present an example. We describe a simple, ad hoc directional interpolation scheme in Section 3. In Section 4, experiments are performed using the example prefilter with standard and directional interpolation. Quantitative and subjective results are given and some observations obtained during the experiments are mentioned. Finally, some closing comments are made in Section 5.

 In *Proceedings of the 32nd Asilomar Conference on Signals, Systems, and Computers*, vol. 2, pp. 959-963, November, 1998, Pacific Grove, CA, USA.

† J. Price is a Kodak Fellow. Funding for this research has been graciously provided by the Eastman Kodak Company.

2. Sensor model optimal prefilter

2.1. Interpolation restated

To incorporate knowledge of the sensor, we restate the interpolation problem in a reconstruction-like manner:

> The given image is the observation of a scene captured by a low-resolution sensor. An interpolated image is sought that is an observation of the same scene, captured by a higher-resolution sensor.

(This interpretation is mentioned in [14] and [16] as well.) We intend to employ a model that relates these high-resolution and low-resolution images. As a simple example, we consider an image captured by a CCD array with element size $(d \times d)$. We seek an interpolated image with twice the number of pixels in each direction, corresponding to a CCD array with element size $(d/2 \times d/2)$. This idea is illustrated in Fig. 1.

The response of a CCD element is directly proportional to its area. With this in mind, we employ the layout of Fig. 1 to construct a simple, discrete model relating the desired high-resolution image to the observed low-resolution image. This model, shown in Fig. 2, convolves the high-resolution image with an FIR filter $h(\mathbf{n})$, and then downsamples by a factor of M in each direction. With the CCD arrangement of Fig. 1, we have $M = 2$ and

$$h = \begin{pmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{pmatrix}. \tag{1}$$

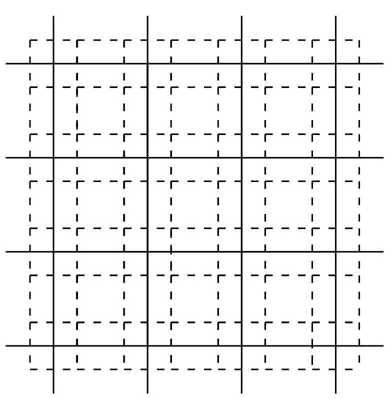


Figure 1. Example CCD layout for interpolation by factor of 2 (in each direction). Solid lines indicate the low-resolution sensor and dashed lines the high-resolution sensor.

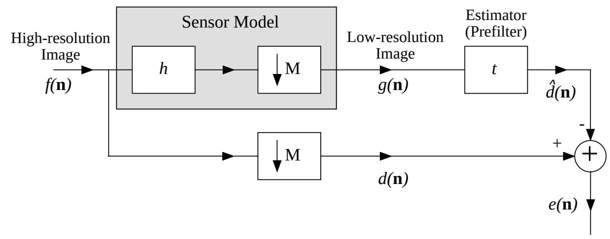


Figure 2. Simple model relating desired high-resolution image to observed low-resolution image and layout for prefilter design.

To apply this model, we first regard interpolation from a slightly different perspective. We consider interpolation as a technique to fill in or "predict" missing pixels from known pixels. In the context of the model above, however, the pixels of the observed image are *not* pixels of the desired image – they are distorted by the FIR filter h in Fig. 2. We therefore seek to estimate actual pixels of the desired image and interpolate (predict from) these rather than interpolating the distorted pixels of the observed image. Such an estimator will serve as an interpolation prefilter. In other words, the prefilter will be applied to the low-resolution image and then the image will be interpolated to the higher-resolution. To derive this estimator, we employ the layout illustrated in Fig. 2.

2.2. Optimal prefilter

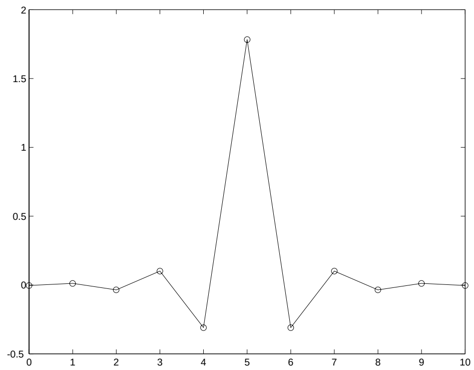
We propose a minimum MSE estimator and seek filter coefficients $t(\mathbf{n})$ to minimize $\xi = \mathcal{E}\{e(\mathbf{n})^2\}$, with $e(\mathbf{n})$ defined as shown in Fig. 2. Note that all indices represent two-element vectors – i.e., $\mathbf{n} = (n_1, n_2)$. For simplicity, we begin with the (not necessarily accurate) assumption that the original image $f(\mathbf{n})$ can be modeled as a wide-sense stationary (WSS), random process with covariance function $r(\mathbf{k})$. Differentiating ξ with respect to $t(\mathbf{n})$ and setting the result to zero will, after some manipulation, yield

$$\sum_{\mathbf{p}} t(\mathbf{p}) \sum_{\mathbf{j}} c_h(\mathbf{j}) \Big[r(M\mathbf{n} - M\mathbf{p} - \mathbf{j}) + r(M\mathbf{p} - M\mathbf{n} - \mathbf{j}) \Big] = 2 \sum_{\mathbf{i}} h(\mathbf{i}) r(M\mathbf{n} + \mathbf{i}), \tag{2}$$

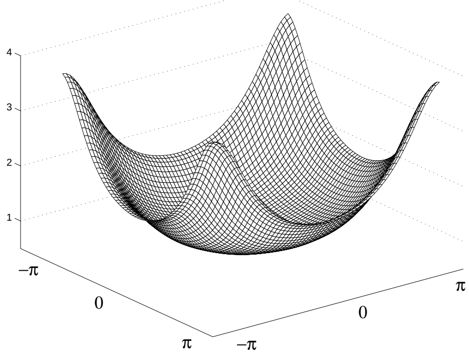
where $c_h(\mathbf{j})$ represents the deterministic autocorrelation of $h(\mathbf{n})$ given by

$$c_h(\mathbf{j}) = \sum_{\mathbf{l}} h(\mathbf{l}) h(\mathbf{l} + \mathbf{j}). \tag{3}$$

For $t(\mathbf{n})$ an $(L \times L)$ FIR filter, Equation (2) defines a system of $(L \times L)$ linear equations that can be solved to find the



(a) Center column of $t(\mathbf{n})$.



(b) Magnitude of the DFT of $t(\mathbf{n})$.

Figure 3. Optimal (11×11) prefilter $t(\mathbf{n})$ corresponding to sensor model of Equation (1).

coefficients $t(\mathbf{n})$. Note that $t(\mathbf{n})$ is not separable unless both the sensor model $h(\mathbf{n})$ and the covariance function $r(\mathbf{k})$ are separable.

To find an example (11×11) filter, we use the nonseparable, exponential, image covariance model suggested in [4] (with $\rho_1 = \rho_2 = 0.95$) and the sensor model of Equation (1). In this case, the coefficients of $t(\mathbf{n})$ tend rapidly to zero. The center column of $t(\mathbf{n})$ is displayed in Fig. 3(a), and the magnitude of its DFT in Fig. 3(b). It is evident from the DFT that the prefilter accents the higher frequencies. We note that the frequency response of this filter is similar to that of the direct spline transform prefilter used in the interpolation scheme of [12].

3. Directional interpolation

After some preliminary experimentation with the example optimal prefilter, observations indicated that its performance might be improved by incorporation into a directional interpolation scheme. The intuitive reasoning behind this hypothesis stems from the global, WSS covariance model used in the prefilter derivation of the previous section. The WSS model implies that all pixels in an image are equally correlated with their nearby neighbors. This is true for many pixels of most natural images, but there are, however, cases where this implication is incorrect. Edges, for example, are one such case – pixels on one side of an edge might be correlated with one another, but not with those on the opposite side of the edge. Similar arguments can be made for other image structures. In such scenarios, we might expect different correlations in different directions. Therefore, by using directional interpolation, we hope to compensate for some of the inaccuracies introduced by the global nature of the prefilter. (Oriented Gaussian post-filters are used in [12] for this function.) For this reason, we have implemented an ad hoc directional scheme for interpolation by a factor of $M = 2$ (in each direction).

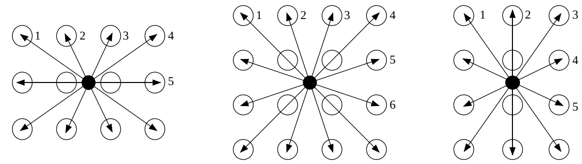


Figure 4. The three possible pixel arrangements and associated directions for directional interpolation by factor of 2. The filled circles indicate the pixel to be interpolated and the empty circles indicate original pixels.

The ideas behind our scheme are similar to those of [11] and [17]. For factor of 2 interpolation, there are three distinct arrangements for the interpolated pixels as illustrated in Fig. 4. For each new pixel, either five or six directions are considered. The four low-resolution pixels nearest the directional lines of Fig. 4 are used to calculate the direction of lowest variation. Once this direction has been found, the interpolated pixel value p is calculated by

$$p = \frac{1}{32}\left(5d_1 + 11d_2 + 11d_3 + 5d_4,\right) \qquad (4)$$

where d_k for $k = 1, \ldots, 4$ indicate the four pixels along the minimum variation direction, with d_2 and d_3 the two nearest the new pixel location. The coefficients of Equation (4) were chosen by experimentation. This scheme is very basic – every pixel is interpolated in exactly the same manner. This can produce excessive smoothing in areas, but is nonetheless sufficient for our purposes. Examples of more advanced directional interpolation approaches can be found in the references.

Interpolation Type	Avg	Std Dev	Min	Max
Linear (22/28)	1.08	0.35	0.01	1.87
Cubic (21/28)	1.12	0.33	0.72	2.09
Directional (21/28)	0.87	0.26	0.58	1.42

Table 1. Summary of PSNR gain (in dB) for prefiltering prior to interpolation.

Interpolation Type	Avg	Std Dev	Min	Max
Linear (6/28)	0.21	0.16	0.02	0.49
Cubic (7/28)	0.32	0.17	0.11	0.63
Directional (7/28)	0.24	0.17	0.03	0.57

Table 2. Summary of PSNR loss (in dB) for prefiltering prior to interpolation.

4. Simulations and results

To test the performance of the prefilter, experiments were performed using 28 monochrome images. Each original, high-resolution image was filtered and downsampled using the sensor model from Fig. 2, with $M = 2$ and h defined by Equation (1). The resulting low-resolution image was interpolated to the original size, with and without the prefilter, using linear, cubic, and the directional interpolation scheme of Section 3.

For an objective measure, the PSNRs between the interpolated and original images were calculated. The differences between the PSNRs (in dB) for images interpolated with and without prefiltering are summarized in Tables 1 and 2. The numbers in parentheses in the first columns indicate the number of images that were improved (Table 1) or degraded (Table 2) when using the prefilter. Only those images were used in calculating the values in each table. It is evident from Table 1 that in the majority of cases (approximately 75%), the prefilter provides a modest improvement in PSNR – about 1.1dB on average for linear and cubic interpolation, and about 0.9dB for directional interpolation. In the few cases shown in Table 2, where the prefilter worsened the PSNR, the loss was only about 0.3dB or less on average. We note that the seven cubic interpolated images that suffered PSNR loss from prefiltering were the same seven images that experienced loss for directional interpolation. (These include the six images that experienced loss with linear interpolation.)

More important than such quantitative measures, however, is the perceived quality of the interpolated images. For this reason, subjective tests were conducted. Subjects were asked to view 12 pairs of (unlabeled) images and to score each image in the pair on a relative scale of 1 to 10. The 12 pairs were broken into three sets of four image pairs each to perform the following three comparisons:

I. Cubic interpolation with prefiltering (A) vs. cubic interpolation only (B),

II. Directional interpolation with prefiltering (A) vs. directional interpolation only (B), and

III. Directional interpolation with prefiltering (A) vs. cubic interpolation with prefiltering (B).

Comparison	Avg	Std Dev	Min	Max
I	+2.82	1.45	-2.00	+6.00
II	+2.88	1.47	-2.00	+6.00
III	+1.41	1.91	-3.00	+6.00

Table 3. Summary of subjective quality comparisons. I – Cubic, prefilter vs. no prefilter; II – directional, prefilter vs. no prefilter; III – prefiltered directional vs. prefiltered cubic.

The sets for comparisons I and II were comprised only of the seven images for which there was a loss in PSNR in the quantitative test (i.e., those of Table 2). The set for III of was chosen arbitrarily. A total of 14 subjects were administered the test resulting in $14 \cdot 4 = 56$ data points for each of the three comparisons. For every image pair, the score difference $(A-B)$, with A and B as indicated in the list above, was calculated and the results are summarized in Table 3.

From the results in Table 3 for I and II, we see that the prefiltered images scored significantly higher. Recall that these images were those where prefiltering produced a PSNR loss. Although subjective tests were not performed on those images with PSNR improvements, our observations indicate similar results would be expected. We note that only four of the 112 total data points for sets I and II indicated that the prefiltered image was worse (resulting in a negative score difference). The results for set III, though not as definitive, show that prefiltering with directional interpolation might be generally better than prefiltering with cubic interpolation. Of the 56 data points for set III, 10 indicated that cubic interpolation was preferred. Comments from the test subjects mentioned that the overall sharpness of the prefiltered, cubic interpolated images was better, but that these images tended to have “jagged” edges and/or noticeable ringing artifacts. Directional interpolation produced better edges, but tended to cause excessive smoothing in other regions. This smoothing, of course, can be attributed to the simplistic nature of our directional interpolation scheme. We would expect a more sophisticated approach to provide better and more definitive results.

5. Summary

In this paper we have presented a novel and simple approach for incorporating sensor knowledge into the image interpolation problem. Using an example sensor model, an optimal prefilter was derived and simulations using this prefilter were performed. Both quantitative and subjective results from these simulations indicate that prefiltering tends to improve the quality of the interpolated images for both standard and directional interpolation. Additionally, comments and results from the subjective experiments indicate that prefiltering with a more sophisticated directional interpolation scheme might provide more visually pleasing results than any of the other techniques explored.

References

[1] H. Hou and H. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 6, pp. 508–517, 1978.

[2] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, 1981.

[3] A. Parker, R. Kenyon, and D. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Transactions on Medical Imaging*, vol. 2, no. 1, pp. 31–39, 1983.

[4] A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc., 1989.

[5] V. R. Algazi and R. Ford, Gary E. amd Potharlanka, "Directional interpolation of images based on visual properties and rank order filtering," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 3005–3008, 1991.

[6] K. Xue, A. Winans, and E. Walowit, "An edge-restricted spatial interpolation algorithm," *Journal of Electronic Imaging*, vol. 1, no. 2, pp. 152–161, 1992.

[7] R. H. Bamberger, "A method for image interpolation based on a novel multirate filter bank structure and properties of the human visual system," in *Image Processing Algorithms and Techniques III*, vol. 1657, pp. 351–362, 1992.

[8] A. B. Mahmoodi, "Adaptive image interpolation algorithm," in *Medical Imaging: Image Processing*, pp. 311–319, 1993.

[9] H. Chen and G. E. Ford, "An FIR image interpolation filter design method based on properties of human vision," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 581–585, 1994.

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

[11] S. D. Bayrakeri and R. M. Mersereau, "A new method for directional image interpolation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2383–2386, 1995.

[12] K. Hong, J. Paik, H. Kim, and C. Lee, "An edge-preserving image interpolation system for digital camcorder," *IEEE Transactions on Consumer Electronics*, vol. 42, no. 3, pp. 279–284, 1996.

[13] A. M. Darwish and M. S. Bedair, "An adaptive resampling algorithm for image zooming," in *Image and Video Processing IV*, pp. 131–144, 1996.

[14] J. Allebach and P. W. Wong, "Edge-directed interpolation," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 707–710, 1996.

[15] J. P. Oakley and M. J. Cunningham, "A formula for least-squares projection and its applications in image reconstruction," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1602–1605, 1989.

[16] S. Carrato, G. Ramponi, and S. Marsi, "A simple edge-sensitive image interpolation filter," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 711–714, 1996.

[17] K. Kido, J. Taguchi, and K. Sano, "Improvement of MRI image quality by a directional adaptive filter," *Systems and Computers in Japan*, vol. 28, no. 10, pp. 69–76, 1997.