Image interpolation based on median-type filters

Bing Zeng

The Hong Kong University of Science and
Technology
Department of Electrical and Electronic
Engineering
Clear Water Bay, Kowloon
Hong Kong
E-mail: eezeng@ee.ust.hk

Anastasios N. Venetsanopoulos

University of Toronto
Department of Electrical and Computer
Engineering
Toronto, Ontario M5S 1A4
Canada

Abstract. Image interpolation based on median-type filters is investigated. We study this problem for two decimation schemes that are most commonly used in practice: (1) quincunx lattice and (2) rectangular lattice. For each decimation lattice, we propose a number of interpolation structures that are based on some median or finite impulse response (FIR)-median hybrid filters. Extensive simulations show that the proposed median-based interpolators outperform significantly as compared with linear schemes. To further verify the effectiveness of these median-based interpolators, we compare them with multilayer neural networks designed through a training process, and still show a preference for the median-based interpolators. As an application, they are used in an interpolative Joint Photographic Experts Group (JPEG) coding scheme to suppress blocking artifacts at very low bit rates. © 1998 Society of Photo-Optical Instrumentation Engineers. [S0091-3286(98)02209-0]

Subject terms: image decimation and interpolation; median-type filters; neural networks; interpolative Joint Photographic Experts Group coding.

Paper 31087 received Aug. 20, 1997; revised manuscript received Mar. 19, 1998; accepted for publication Mar. 23, 1998.

1 Introduction

Image decimation/interpolation has been an important technique in a number of image applications including sampling rate alteration, video scan conversion, image zooming, etc.¹⁻⁸ In addition, the combination of decimation and interpolation provides an easy way to achieve compression since the subsampled image has a reduced size, thus requiring fewer bits for transmission or storage. Furthermore, high compression ratios are achievable if a coding scheme is applied to the subsampled image since the bit rate resulting from encoding the subsampled image must be divided by the subsampling factor when one computes the overall bit rate. An image with full resolution can be reconstructed with satisfactory quality if an appropriate interpolator is used. This decimation/interpolation-based compression system is illustrated in Fig. 1, where the coding and decoding blocks are enclosed with dashed lines because they may or may not be used.

It is easy to understand that in the system shown in Fig. 1, the design of good interpolators is crucial to achieving a high-quality reconstructed image. Often, image interpolation is performed by pixel replication in a small neighborhood of each existing pixel. This is equivalent to a first-order linear filter. Obviously, such a scheme is very simple, but can hardly provide a satisfactory solution. In particular, the resulting blocking artifacts are quite annoying. The performance can be improved by employing higher order linear filters such as Gaussian filters⁹ finite impulse response (FIR) or infinite impulse response (IIR) filters.¹⁰⁻¹³ However, the use of higher order filters for interpolation leads to larger filtering masks, which require more line buffers, adders, multipliers, etc., and consequently increases the chip area in very large scale integration (VLSI) implementation. In addition, linear filters in general create artifacts along image edges, to which human eyes are quite sensi-

tive. Furthermore, good criteria do not exist for the design of "optimal" linear interpolators.

Although the characteristics of images vary substantially from one image to another, studies of human visual perception show that human eyes are mostly sensitive to some details (such as lines, edges, corners, etc.) whose spectra are primarily composed of high-frequency components.^{14,15} To better preserve these details, one should alternatively seek nonlinear interpolators. In this paper, we propose two such interpolation schemes. The first scheme uses median-type filters to do the interpolation. We choose a small window mask to achieve fast implementation. The design of such interpolators is independent of the particular image under consideration. In contrast, the second scheme proposed in this paper is image-dependent, and its structure is a multilayer perceptron network; thus, its design requires a training process. However, we see that this neural-network-based interpolator can hardly outperform the median-filter-based counterpart, unless its structure becomes complicated (i.e., the numbers of nodes and layers become larger), thus demonstrating the effectiveness of the median-based interpolators.

In our study, we concentrate on the two subsampling lattices that are most commonly used in practice: (1) the quincunx lattice and (2) the rectangular lattice. We compare the new interpolators with linear schemes in terms of their structure complexity as well as by subjectively testing both natural and synthetic images. During the testing, we consider both cases where the decimated image has or has not been compressed. The results obtained indicate a general conclusion that median-based nonlinear interpolation schemes are highly preferable.

The organization of this paper is as follows. Section 2 briefly reviews the quincunx and rectangular subsampling lattices and discusses some related issues, such as their effects on the image's frequency band and the effect of in-

Fig. 1 Decimation/interpolation-based compression system.

cluding a low-pass filter before the subsampling. In Sec. 3, we present several median-based nonlinear interpolators for each of the two subsampling lattices considered in this paper. We also discuss their computational complexity and compare them with linear filters by testing several real images. In Sec. 4, the neural-network-based interpolators are designed and compared with the median-based interpolators. In Sec. 5, we study the proposed median-based interpolation schemes for the case when a discrete cosine transform (DCT) compression scheme is applied to the decimated image. Finally, some conclusions are drawn in Sec. 6.

2 Quincunx and Rectangular Subsampling Lattices

Figure 2 shows the orthogonal sampling lattice and two basic lattices for subsampling—quincunx and rectangular—as well as their primitive bandwidths. It is clear that these two subsampling schemes provide 2:1 and 4:1 subsampling factors, respectively.

In Fig. 2(b) we see that the quincunx subsampling lattice does not reduce resolution on “pure” horizontal and vertical frequencies, but limits only high diagonal frequencies. This implies that edges along the vertical and horizontal directions will be preserved well, but those along diagonal directions will be degraded to some extent. On the other hand, as shown in Fig. 2(c), high frequencies along the horizontal, vertical, and diagonal directions have all been lost by using the rectangular subsampling lattice.

Another important issue concerning the decimation/interpolation system is whether the original image should be low-pass filtered before the subsampling takes place.

Fig. 2 (a) Orthogonal sampling lattice and its primitive bandwidth, (b) quincunx subsampling lattice and its primitive bandwidth, and (c) rectangular subsampling lattice and its primitive bandwidth.

Fig. 3 Interpolation structure for (a) a quincunx subsampling lattice and (b) a rectangular subsampling lattice.

Based on the Nyquist sampling criteria, signals must be low-pass filtered to avoid aliasing errors. Practically, the acceptability of aliasing depends on the characteristics of the image being processed. If the base image does not contain much high-frequency information, an antialiasing filter may not be required prior to subsampling. However, if the base image does contain significant high-frequency information, an antialiasing filter will be required. Of course, this antialiasing filter itself also distorts the base image so that the original quality is never obtained by any interpolation. Nevertheless, the situation is slightly different if the decimated image is to be compressed. Since a compression always introduces errors, it does no harm if certain additional errors are produced after the antialiasing filter (so that in the error of the reconstructed image we cannot differentiate the contribution of the antialiasing filter from that of the compression) as long as the total error measurement is somehow controlled under a reasonable level. On the other hand, placing an antialiasing filter in front of the subsampling can usually increase the coding efficiency (i.e., achieving higher compression ratios).

3 Median-Filter-Based Interpolators

Design of general nonlinear interpolators is usually difficult due to their nonlinear nature. In this section, we propose several nonlinear interpolation schemes that are all based on median filters, and therefore can be regarded as *ad hoc* methods. However, the results produced by these simple methods are demonstrated to be quite promising.

3.1 Quincunx Subsampling Lattice

Using Fig. 3(a), which illustrates the quincunx interpolation structure, we present in the following five median-based interpolators (denoted as QUIN1, QUIN2, QUIN3, QUIN4, and QUIN5, respectively):

$$Z^{\text{QUIN1}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4\},$$

$$Z^{\text{QUIN2}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, \bar{X}\},$$

$$Z^{\text{QUIN3}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, X_{\text{FIR}}\},$$

$$Z^{\text{QUIN4}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, \bar{X}_H, \bar{X}_V, X_{\text{FIR}}\},$$

and

$$Z^{\text{QUIN5}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, X_{\text{HFIR}}, X_{\text{VFIR}}, X_{\text{FIR}}\},$$

where the median of a set of N samples is defined to be the most-centered value in the sorted set of samples if N is odd or the average of the two most-centered values in the sorted

Fig. 4 FIR filter structures for (a) X_{HFIR} , (b) X_{VFIR} , and (c) X_{FIR} .

samples if N is even, $\bar{X}_H = (X_1 + X_2)/2$ (horizontal average), $\bar{X}_V = (X_3 + X_4)/2$ (vertical average), $\bar{X} = (X_1 + X_2 + X_3 + X_4)/4$ (overall average), and X_{HFIR} , X_{VFIR} , and X_{FIR} are obtained respectively through FIR filters, as shown in Figs. 4(a) to 4(c).

We see from the preceding equations that except for QUIN1, the other four interpolators all contain one or more linear FIR substructures, thus resulting in FIR-median hybrid filters. Actually, QUIN1 was originally introduced in Ref. 16, and a seven-point FIR-median hybrid interpolator similar to QUIN4 but using a slightly different FIR filter (which is biased toward the horizontal direction) was also proposed there. We used QUIN2 in our earlier study on interpolative BTC image coding.¹⁷ A comparative study on QUIN1 to QUIN4 was presented in Ref. 18, while their vector counterparts were studied in Ref. 19.

The window lengths of the interpolators QUIN1 to QUIN5 are also seen to be 4, 5, 5, 7, and 7, respectively, which are all small integers. This means that the procedure of interpolation is fast and simple, therefore very suitable for practical implementation. Compared to linear filters (either FIR or IIR), all the nonlinear interpolators proposed here contain no multipliers. (Note that the multipliers $1/2$, $1/4$, and $1/8$ can be implemented via bit shifts.)

To subjectively judge the proposed nonlinear interpolators, we tested them on both natural images ("Lena" and "Boat") and synthetic images (SYN1 and SYN2), all of size 512×512 , as shown in Fig. 5. The results are presented in Table 1 (the optimum in each row is in bold). For the sake of comparison, we also include two linear filters (LINE1 and LINE2) in our testing experiments, as shown

Fig. 5 Four 512×512 testing images: (a) "Lena," (b) "Boat," (c) SYN1, and (d) SYN2.

in Figs. 6(a) and 6(b). The mean square error (MSE) and mean absolute error (MAE) between the reconstructed image and the original image were measured, and the peak signal-to-noise ratio $PSNR = 10 \log(255^2/MSE)$ is computed for each reconstructed image. From the table, the following conclusions can be drawn:

1. All the five nonlinear interpolators proposed here are quite good, especially QUIN3, QUIN4, and QUIN5.
2. For the two natural images, we see that QUIN4 is slightly better than QUIN3 and QUIN5; whereas QUIN5 provides the best performance for the synthetic images.
3. QUIN1 is obviously the simplest among the five nonlinear interpolators considered here, but it still has quite promising performance.
4. For the natural images, LINE1 is inferior to all nonlinear interpolators, while LINE2 can provide only a solution similar to that of QUIN1 or QUIN2, and therefore is inferior to QUIN3 to QUIN5.

Table 1 Testing results of various median-based interpolators for quincunx subsampling.

Image		QUIN1	QUIN2	QUIN3	QUIN4	QUIN5	LINE1	LINE2
"Lena"	MSE	8.52	9.14	8.04	7.50	8.73	9.70	9.50
	MAE	1.30	1.33	1.31	1.25	1.38	1.37	1.28
	PSNR	38.83	38.52	39.08	39.38	38.72	38.26	38.35
"Boat"	MSE	15.20	16.65	12.45	11.71	13.53	18.52	14.02
	MAE	1.69	1.76	1.60	1.55	1.69	1.87	1.61
	PSNR	36.31	35.92	37.18	37.44	36.82	35.45	36.66
SYN1	MSE	62.06	62.41	56.15	62.03	49.05	77.35	754.31
	MAE	0.78	0.79	0.70	0.78	0.59	1.29	4.04
	PSNR	30.20	30.18	30.64	30.20	31.22	29.25	19.36
SYN2	MSE	70.61	70.24	65.77	69.93	60.02	141.17	367.81
	MAE	0.67	0.67	0.63	0.67	0.56	2.11	3.25
	PSNR	29.64	29.66	29.95	29.68	30.35	26.63	22.47

Fig. 6 (a) Zeroth-order maximally flat filter (LINE1) and (b) first-order maximally flat filter (LINE2).

- For the synthetic images, we see that both linear filters LINE1 and LINE2 failed totally. Since both SYN1 and SYN2 are mainly composed of sharp edges, narrow lines, etc., linear filters result in very large distortion due to their smoothing properties. This also indicates that linear interpolators under consideration are quite sensitive to the characteristics of images. On the other hand, we found that all five of the nonlinear interpolators tested are more robust.

3.2 Rectangular Subsampling Lattice

The interpolation mechanism for the rectangular subsampling lattice is significantly different from that for the quincunx subsampling lattice. For the rectangular case, the pixels that are to be reconstructed are first classified into two groups, denoted as $\{Y_i\}$ and $\{Z_i\}$, respectively, as shown in Fig. 3(b). We see that the $\{Z_i\}$ group contains all the pixels along the vertical and horizontal directions, and the $\{Y_i\}$ group contains all pixels along the diagonal direction. Based on this classification, it is quite reasonable to design different interpolators for different groups of missing pixels.

Based on the preceding discussions, the proposed interpolation for the rectangular subsampling lattice contains two stage. First, for the pixels in $\{Y_i\}$, we propose the use of the following two schemes:

$$Y^{\text{RECT1}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, X'_{\text{FIR}}\}$$

and

$$Y^{\text{RECT2}} = \text{MEDIAN}\{X_1, X_2, X_3, X_4, X_{D1}, X_{D2}, X'_{\text{FIR}}\},$$

where $X_{D1} = (X_1 + X_4)/2$, $X_{D2} = (X_2 + X_3)/2$, and X'_{FIR} is obtained by an FIR filtering, as shown in Fig. 7(a).

Fig. 7 (a) FIR filter structure for rectangular subsampling lattice and (b) 5x5 Gaussian filter.

Next, three schemes are proposed for the reconstruction of pixels in $\{Z_i\}$:

$$Z_1^{\text{RECT1}} = \frac{X_1 + X_2}{2}, \quad Z_2^{\text{RECT1}} = \frac{X_1 + X_3}{2},$$

$$Z^{\text{RECT2}} = Z^{\text{QUIN4}},$$

and

$$Z^{\text{RECT3}} = Z^{\text{QUIN5}}.$$

By combining these schemes, we can obtain a total of six interpolators, i.e., $Y^{\text{RECT1}} + Z^{\text{RECT1}}$, $Y^{\text{RECT2}} + Z^{\text{RECT1}}$, $Y^{\text{RECT1}} + Z^{\text{RECT2}}$, $Y^{\text{RECT2}} + Z^{\text{RECT2}}$, $Y^{\text{RECT1}} + Z^{\text{RECT3}}$, and $Y^{\text{RECT2}} + Z^{\text{RECT3}}$, which are all nonlinear in nature. For simplicity, we denote them as RECT1 to RECT6, respectively.* Among these interpolators, one can find that RECT3 to RECT6 possess a recursive structure since the interpolation of each $\{Z_i\}$ utilizes some $\{Y_i\}$'s that were already reconstructed earlier in the first stage. On the other hand, the other two interpolators (RECT1 and RECT2) are nonrecursive approaches. From the formulation of the various interpolators, it is quite easy to see that the computational complexity ranks from RECT1 to RECT6 (from the lowest to the highest). Meanwhile, it is also easy to justify that implementations of all these six proposed interpolators are quite simple.

The reasons for the preceding choices are based mainly on the study for the quincunx subsampling presented in the last subsection. For example, the structures similar to QUIN4 and QUIN5 were chosen because they provide the best solution. (We also tried using $\bar{X}$ to replace X'_{FIR} in Y^{RECT1} , but the performance declined to some extent for natural images and was almost the same for synthetic images. However, when the subsampling is preceded by a low-pass filter and the decimated image is to be compressed, the situation changes, as seen in Sec. 5.) To reconstruct the pixels in $\{Z_i\}$, we propose a linear scheme (Z^{RECT1}) mainly due to its simplicity and efficiency of utilizing the "strong" correlations along horizontal and vertical directions.

Table 2 presents some experimental results for the rectangular subsampling lattice using the same four testing images as we used in the last subsection for the quincunx subsampling lattice. To have a comparison with linear filters, we also include in Table 2 the results obtained using a 5x5 Gaussian (linear) filter whose coefficients are given in Fig. 7(b), as well as the replication filter.

From Table 2, one can see that all combinations of Y^{RECT} and Z^{RECT} provide nearly the same quality and their performances are extremely robust as compared to those obtained from linear approaches. Among these nonlinear interpolators, RECT3 (i.e., $Y^{\text{RECT1}} + Z^{\text{RECT2}}$) seems to be slightly better than the other choices for natural images, while RECT6 ($Y^{\text{RECT2}} + Z^{\text{RECT3}}$) seems to be the best for synthetic images. For both natural and synthetic images, the

*Note that RECT1 to RECT4 were presented in Ref. 18, and their vector versions were studied in Ref. 19.

Table 2 Testing results of various median-based interpolators for rectangular subsampling.

Image		RECT1	RECT2	RECT3	RECT4	RECT5	RECT6	GAUS	REPL
"Lena"	MSE	20.92	21.06	20.01	20.51	20.64	20.46	46.57	73.68
	MAE	2.37	2.36	2.34	2.34	2.44	2.40	3.34	4.04
	PSNR	34.92	34.90	35.12	35.01	34.98	35.02	31.45	29.46
"Boat"	MSE	50.27	51.28	48.08	50.41	49.91	50.76	95.97	144.37
	MAE	3.45	3.48	3.40	3.47	3.49	3.51	4.68	5.49
	PSNR	31.12	31.03	31.31	31.11	31.15	31.08	28.31	26.54
SYN1	MSE	141.52	138.72	129.49	126.07	126.20	124.33	393.24	301.90
	MAE	1.94	1.93	1.83	1.81	1.76	1.75	4.20	2.40
	PSNR	26.62	26.71	27.01	27.12	27.12	27.19	22.18	23.33
SYN2	MSE	372.19	368.85	371.51	367.45	369.88	365.74	1099.41	308.56
	MAE	3.64	3.63	3.62	3.61	3.59	3.57	6.19	1.61
	PSNR	22.42	22.46	22.43	22.49	22.45	22.50	17.72	23.24

Table 3 Testing results of various median-based interpolators for quincunx subsampling (with low-pass filter in front of the subsampling).

Image		QUIN1	QUIN2	QUIN3	QUIN4	QUIN5	LINE1	LINE2
"Lena"	MSE	16.85	13.48	17.87	15.71	12.41	18.36	16.68
	MAE	2/43	2.21	2.48	2.36	2.17	2.51	2.31
	PSNR	35.87	36.83	35.61	36.17	37.19	35.49	35.91
"Boat"	MSE	32.72	26.86	35.24	29.80	25.20	37.55	31.60
	MAE	3.34	3.07	3.44	3.23	3.00	3.55	3.24
	PSNR	32.98	33.84	32.66	33.39	34.12	32.36	33.13
SYN1	MSE	101.77	79.37	104.92	98.97	75.39	111.78	620.00
	MAE	2.26	1.85	2.31	2.21	1.71	2.48	4.28
	PSNR	28.05	29.13	27.92	28.18	29.36	27.65	20.21
SYN2	MSE	184.39	170.13	186.84	182.93	168.59	242.99	277.42
	MAE	3.56	3.33	3.60	3.53	3.29	4.38	4.20
	PSNR	25.47	25.82	25.41	25.51	25.86	24.27	23.70

linear interpolator performance is much worse, as compared to that obtained with nonlinear interpolators. The degradation in PSNR ranges from 3.0 to 4.8 dB. (An exception is observed with the replication filter for the image SYN2. This is simply because SYN2 is primarily composed of uniform chessboards, which thus make the replication filter very suitable. On the other hand, the replication filter results in the worst performance for the rest of three test images. Therefore, such a filter should in general be excluded from practical considerations.)

3.3 Effect of Low-Pass Filtering

In the last two subsections, we successfully searched out one or two median-based filters (for both the quincunx and rectangular subsampling schemes) that consistently work the best for the testing images. On the other hand, however, we know that in practical applications a low-pass filter is typically placed in front of the subsampler. Thus, a natural question one may ask is whether the optimal filters already derived remain unchanged when such a low-pass filter is included. To keep this study simple, we do not invoke any process of filter design in our studies; instead, we just select some simple filters that have low-pass characteristics.

Let us consider the quincunx subsampling lattice first. For this scheme, the filter we chose is a five-point averaging operator that outputs the average of the current pixel and its four most neighboring pixels (i.e., the filter mask forms a five-point plus shape). The low-pass characteristics of this filter are obvious.[†] After the low-pass filtering is done, the filtered image undergoes the quincunx subsampling, and then one of interpolation schemes already proposed is applied to reconstruct the missing pixels. The performances of individual interpolators are presented in Table 3. From these results, several conclusions can be drawn:

1. After placing a low-pass filter in front of the quincunx subsampler, it is still true that the proposed nonlinear interpolators clearly outperform their linear counterparts in terms of MSE, MAE, and robustness.

[†]According to Fig. 2(b), the five-point X-shaped average filter should be used so as to attenuate the high frequencies in the diagonal direction while passing some high frequencies in the horizontal and vertical directions. We actually tried this filter. The simulation results, however, showed that the performance deteriorated by 2 to 3 dB.

Table 4 Testing results of various median-based interpolators for rectangular subsampling (with a low-pass filter in front of the subsampling).

Image		RECT1	RECT2	RECT3	RECT4	RECT5	RECT6	GAUS	REPL
"Lena"	MSE	23.12	24.07	22.33	24.18	19.14	20.64	43.75	59.97
	MAE	2.78	2.82	2.74	2.84	2.58	2.65	3.60	4.03
	PSNR	34.49	34.32	34.64	34.30	35.31	34.98	31.72	30.35
"Boat"	MSE	50.93	54.12	48.35	54.34	43.48	48.96	85.66	113.98
	MAE	4.02	4.13	3.93	4.15	3.73	3.93	4.95	5.42
	PSNR	31.06	30.80	31.29	30.78	31.75	31.23	28.80	27.56
SYN1	MSE	129.31	133.67	125.63	134.28	102.71	111.32	726.02	222.33
	MAE	2.71	2.80	2.62	2.80	2.18	2.36	5.40	3.19
	PSNR	27.01	26.87	27.14	26.85	28.01	27.67	19.52	24.66
SYN2	MSE	305.12	333.24	276.65	332.22	262.21	318.76	317.24	311.27
	MAE	4.82	5.24	4.39	5.22	4.14	4.97	5.27	4.08
	PSNR	23.29	22.90	23.71	22.92	23.94	23.10	23.12	23.20

- The optimal choice for this case seems to be very consistent, as QUIN5 always provides the best solution.
- For all test images, QUIN2 (which uses $\bar{X}$ instead of an FIR filter) always provides the second best solution, which is actually very close to the best solution. (Notice that QUIN2 is two-points shorter than QUIN5 and requires fewer internal computations than does QUIN5.)
- Compared to the results presented in Table 1, we observe that including low-pass filtering deteriorates the PSNR by a few decibels, no matter what interpolator is used. However, we also see that QUIN5 and QUIN2 result in much smaller decreases, thus indicating their superior robustness.

For the rectangular subsampling lattice, the low-pass filter used is still the five-point plus shape averaging filter. The test results are summarized in Table 4. From this table, we can also draw some conclusions that are similar to those for the quincunx subsampling. For example, the proposed nonlinear interpolators still clearly outperform their linear counterparts in terms of MSE, MAE, and robustness after a low-pass filtering is included and the best solution (RECT5) also seems to be consistent for all the test images. On the other hand, a unique phenomenon (which did not happen in the quincunx case) is that the low-pass filtering placed in front of the rectangular subsampling does not always deteriorate the PSNR of the reconstructed image. To the contrary, it always improves the PSNR values slightly in the best case. The reason lies in the fact that the rectangular subsampling is quite substantial as compared to the quincunx subsampling and thus correlations between pixels are not fully utilized in the interpolation if no low-pass filtering is done before the subsampling. On the other hand, by doing a low-pass filtering, we can preserve a relatively higher degree of correlations in the subsampled image, thus implying that it is possible to achieve an increased PSNR.

Note also concerning the results presented in Table 4 is that as compared to the results shown in Table 2, the MAE measures are now deteriorated to some extent. This is be-

cause the MAE measure is highly dependent on small details such as narrow lines, steep edges, sharp corners, etc. Obviously, low-pass filtering is not suitable to preserve these high-frequency components.

4 Neural-Network-Based Interpolators

The extensive comparisons presented in the last section support a quite general conclusion that the proposed median-based nonlinear interpolators clearly outperform linear filters. This seems to be a natural result: as images exhibit highly nonlinear characteristics in general, nonlinear filters should work better if they are designed properly. On the other hand, a second question is whether we can find other operators that work better than the median-based interpolators among the class of nonlinear operators. The answer seems to be affirmative; nevertheless, developing a systematic design method to find such a solution is at present quite impossible if no additional constraints on the nonlinear filter class are imposed, simply because that nonlinear filters, i.e., any filter that is not linear, constitute a huge class. Put in a more clear manner, we do believe that median filters may not be the best choice for the particular task considered in this paper (in terms of the reconstructed image quality). However, we are not very optimistic about the possibility of finding other nonlinear interpolators that work considerably better than median filters while possessing computational simplicity and implementation efficiency comparable to median filters. To have some solid support for this claim, in this section, we introduce another nonlinear interpolation based on multilayer perceptron networks.^{20,21}

We select to do this neural-based interpolation not only due to the tractability of multilayer neural networks but, more importantly, because neural networks are usually believed to be able to accommodate nonlinearity quite well, especially when the number of neurons and/or the number of layers is large.

Figure 8 presents a fully connected three-layer neural network. The first and last layers are termed the input and output layers, respectively, while the layer in between is called the hidden layer. There are no activities in each neuron of the input layer except for passing through the input

Fig. 8 Fully connected three-layer neural network.

value it receives. On the other hand, each neuron in the hidden layer or output layer acts as a nonlinear function. Usually, we assume the same nonlinear function for all these neurons, and the most commonly used one is the sigmoid function (which is used in this paper):

$$\text{out} = f(\text{sum}) = \frac{1}{1 + \exp(-\alpha \text{sum})},$$

where $\alpha > 0$ is a constant (usually set to 1) and $\text{sum} = \sum_{i=0}^N w_i \text{in}_i$. Here, $\text{in}_0 = 1$ and w_0 works as a threshold level; in_i and w_i ($i > 0$) represent the i 'th input variable and its corresponding weight, respectively.

As the sigmoid function can produce only values between 0 and 1, input variables must be scaled down into the range [0,1]. For images with 8-bit representation, the scaling can be done by simply dividing each pixel value by $2^8 = 256$. At the output layer, the results are multiplied by this scaling factor and quantized to integers so as to recover the original 8-bit representation.

To perform the neural-based interpolation, we just need to feed the available pixels [i.e., X 's in Figs. 3(a) and 3(b)] into the network. The output from the network is the interpolated value of the missing pixel. Naturally, we can also take the outputs of some FIR substructures (such as $\bar{X}$,

X_{FIR} , X'_{FIR} , etc.) as additional inputs and feed them into the network. By doing so, it is generally expected that the performance improves. In analogy to QUIN1, RECT1, etc. used in Sec. 3, we denote the various neural interpolators that have the same inputs as N-QUIN1 to N-QUIN5 (for the quincunx lattice) and N-RECT1 to N-RECT6 (for the rectangular lattice), respectively.

For the interpolation problem considered here, the desired signal is composed of all missing pixels that are available at the encoder. Therefore, we can train the neural interpolator using the back-propagation algorithm before subsampling. Once the training is accomplished, all resulting weights must be transmitted (or stored), since they are used later to carry out the interpolation. Obviously, this introduces some overheads that are proportional to the number of weights used in the neural network and the number of bits used to represent the weights. Usually, these numbers are small as compared to the image size so that they can be ignored when computing the coding bit rate.

Using the same test images, we performed extensive simulations for the various neural-based interpolators. However, we do not present all the results here simply because, for each case (i.e., each particular neural interpolator for either the quincunx or rectangular subsampling scheme), there are several parameters (such as number of hidden layers, number of each hidden layer, step size, and etc.) that can vary. Instead, we show only some examples that we believe are very representative. As such, Fig. 9 shows the typical converging behaviors of N-QUIN4 and N-QUIN5 (for "Lena" and "Boat"), where the perceptron structure (thus, no hidden layers) is used. We selected to present these results because QUIN4 and QUIN5 are, respectively, the best interpolators for their corresponding cases (see Tables 1 and 3). We excluded the two synthetic images from our considerations because both images are quite uniform (except for a few details). Thus, using a (non-

Fig. 9 Typical converging behaviors of perceptron interpolators for quincunx subsampling lattice: (a) N-QUIN4 for "Lena" (without prefiltering), (b) N-QUIN4 for "Boat" (without prefiltering), (c) N-QUIN4 for "Lena" (with prefiltering), and (d) N-QUIN4 for "Boat" (with prefiltering).

Fig. 10 Typical converging behavior of a three-layer neural interpolator N-QUIN4 (without prefiltering) (a) for "Lena" and (b) for "Boat."

linear) neural interpolator would not be very appropriate. (Actually, this was confirmed during our simulations: not only does the training take a large number of iterations, but the reconstruction quality is also quite poor.) On the other hand, for the rectangular subsampling scheme, one can design neural interpolators with three outputs (for Y , Z_1 , and Z_2 , respectively), or follow the two-stage idea, as outlined earlier, to first design a neural interpolator for Y 's and then the second interpolator for Z 's. The latter approach seems to be more appropriate from the viewpoint of comparisons and actually was demonstrated to work better than the first approach.

It is clear from Fig. 9 that the neural interpolators having perceptron structure produce worse performance as compared to their median-based counterparts if no pre-low-pass filtering is involved, whereas the prefiltering causes the neural interpolators to be tied (in reconstruction quality) with the corresponding median-based interpolators. However, one must remember that, for a neural interpolator, the implementation is much more complicated, some overhead information results due to the representation of all weights, and a process of training exists (thus causing a longer delay). Of course, the reconstruction quality can be enhanced using the multilayer structure. However, we observed that such an improvement is only marginal, as is shown by an example in Fig. 10 where a three-layer network with 10 neurons in the hidden layer is trained to do the interpolation. The improvement after training for 1000 iterations is 0.52 dB for "Lena" and 0.75 dB for "Boat," respectively (as compared to QUIN4).

5 Image Compression Application

In this section, we consider the case where a compression is performed for the decimated image. The purpose is two-

fold: Our primary goal here is to study whether or not our previous conclusions obtained for the nonlinear interpolators still stand. In the meanwhile, however, this also provides a nice application to do image compression based on the decimation/interpolation approach (as depicted earlier in Fig. 1). By this application, we demonstrate that such an approach is actually quite suitable to alleviate the blocking effects that are regarded as a main drawback with block-based coding schemes (such as DCT) at low bit rates.

To facilitate the study, we assume that the compression for the decimated image is done using Joint Photographic Experts Group^{22,23} (JPEG) (which is basically a DCT+ZIGZAG scanning+run-length Huffman coding). JPEG is recognized as an international standard for still-frame, continuous-tone images (either monochrome or color) and can provide a quite satisfactory solution (judged either by measuring the PSNR or by visually observing the decoded image) for many image coding applications at a moderate bit rate. However, if a higher compression ratio is required, JPEG will suffer severely from the blocking effect, because some (or, sometimes, even most) of the AC components of a block will be quantized to zero. To demonstrate this result, Fig. 11 provides the JPEG-coded "Lena" (at 0.25 bpp) and "Boat" (at 0.30 bpp), both of size 512×512 (the original images were shown earlier in Fig. 5). It is easy to see that for such a case the coded images are rather disturbing to our eyes due to the existence of many blocking artifacts, although the measured PSNR is relatively good (31.86 dB for "Lena" and 30.09 dB for "Boat").

In the simulations that follows, we use the decimation/interpolation coding scheme whose block diagram is shown in Fig. 12. The original image undergoes a low-pass filter operation before the subsampling, where the low-pass filter is chosen to be the same as before (i.e., five-point plus shaped for both quincunx and rectangular). Besides reducing aliasing, the low-pass filter also contributes to the increase of the coding efficiency (i.e., the compression ratio) as the subsampled image to be compressed is now smoother. Here, we consider only two natural images ("Lena" and "Boat"). The simulation results are given in Table 5 (for the quincunx subsampling) and Table 6 (for the rectangular subsampling), respectively. Notice that in RECT1, RECT3, and RECT5, we have used $\bar{X}$ to replace X'_{FIR} , because we found that this replacement brings a substantial improvement in PSNR (over 1 dB).

Fig. 11 JPEG-coded images (only part of the image is shown): (a) "Lena" at 0.25 bpp (PSNR=31.86 dB) and (b) "Boat" at 0.30 bpp (PSNR=30.09 dB).

Fig. 12 Block diagram of the interpolative JPEG scheme.

Table 5 Testing results of various median-based interpolators for quincunx subsampling (with a low-pass filter in front of the subsampling and JPEG).

Image		QUIN1	QUIN2	QUIN3	QUIN4	QUIN5	LINE1	LINE2
"Lena"	MSE	32.22	29.62	32.41	30.70	28.82	32.64	32.02
	MAE	3.57	3.47	3.56	3.51	3.45	3.56	3.47
	PSNR	33.05	33.42	33.02	33.26	33.53	32.99	33.08
"Boat"	MSE	54.50	49.97	55.61	51.84	48.80	57.51	53.06
	MAE	4.48	4.34	4.52	4.41	4.32	4.59	4.41
	PSNR	30.77	31.14	30.68	30.98	31.25	30.53	30.88

Table 6 Testing results of various median-based interpolators for rectangular subsampling (with a low-pass filter in front of the subsampling and JPEG).

Image		RECT1	RECT2	RECT3	RECT4	RECT5	RECT6	GAUS	REPL
"Lena"	MSE	35.75	35.85	36.53	36.67	34.06	34.52	52.93	69.07
	MAE	3.65	3.66	3.67	3.68	3.59	3.62	4.27	4.70
	PSNR	32.60	32.59	32.51	32.49	32.81	32.75	30.89	29.74
"Boat"	MSE	70.11	70.72	71.62	72.70	67.68	69.03	97.39	126.23
	MAE	5.00	5.01	5.03	5.05	4.93	4.96	5.68	6.21
	PSNR	29.67	29.64	29.58	29.52	29.83	29.74	28.25	27.12

During the compression of the decimated image using JPEG, the standard JPEG 8×8 quantization matrix is selected.²³ For the "LENA" image, the compression ratios achieved using the proposed interpolative JPEG scheme are 16.48 for quincunx and 29.44 for rectangular, respectively; while the corresponding ratios for the "BOAT" image are 15.15 for quincunx and 26.24 for rectangular. From Tables 5 and 6, we found that it is still true that the proposed nonlinear interpolators work in general better than the linear counterparts, no matter what subsampling scheme is employed. In particular, the performances provided by the best nonlinear interpolators (i.e., QUIN5 for quincunx and

RECT5 for rectangular) are improved substantially as compared to those achieved with linear approaches.

Finally, to observe visually how the blocking effect is reduced by using the proposed decimation/interpolation scheme as well as to compare visually the linear and nonlinear interpolators, Figs. 13 and 14 show two coded "Lena" images and two coded "Boat" images (all obtained in the rectangular case), respectively, where different interpolators are used. Comparing these results with Fig. 10, it is obvious that the visual perception now becomes much more pleasant, although new PSNRs measured for "Boat" have even deteriorated slightly.

Fig. 13 Interpolative JPEG coded "LENA" image at 0.27 bpp [the same part is shown as in Fig. 11(a)]: (a) using the interpolator RECT5 (PSNR=32.81 dB) and (b) using the Gaussian interpolator GAUS (PSNR=30.89 dB).

Fig. 14 Interpolative JPEG coded "BOAT" image at 0.30 bpp [the same part is shown as in Fig. 11(b)]: (a) using the interpolator RECT5 (PSNR=29.83 dB) and (b) using the Gaussian interpolator GAUS (PSNR=28.25 dB).

6 Concluding Remarks

In this paper, we developed several median-filter-based image interpolation schemes for the quincunx subsampling and rectangular subsampling. It was shown that such approaches are computationally efficient and provide quite promising reconstruction quality. Furthermore, the extensive comparisons with some linear and neural-network-based schemes showed that the median-type filters presented in this paper provide nearly the best compromise between quality (PSNR, MAE, robustness, visual effect, etc.) and cost (software or hardware implementation).

A few additional remarks related to this work are worth presenting here. First, the neural interpolators designed in this paper were demonstrated to provide reconstruction quality comparable only to their median-based counterparts. This result essentially remains unchanged even when we increase the number of hidden layers and the number of neurons in each hidden layer. However, to some extent, we believe that this is because the training is trapped at a local minimum. Effort is now devoted to finding ways to search for the global minimum (or better local minima). Nevertheless, we strongly believe that even the global minimum will be unable to significantly outperform the proposed median interpolators. Furthermore, considering the long training process, overheads, and complicated implementation of a neural interpolator, the quality-cost trade-off it can provide may be ranked lower than what can be achieved using simple median-based schemes.

Second, the linear filters used in the paper for comparisons were not optimized in any sense. Obviously, we can design optimal linear filters according to a certain criterion. For example, we can design Wiener filters that are optimal in the MSE sense. Actually, we did design several Wiener filters and found that they slightly outperformed the corresponding median-based interpolators. On the other hand, since all coefficients of a Wiener filter are floating-point numbers, real multipliers are involved and therefore its implementation (in VLSI) is much more complicated. Moreover, Wiener filters are very sensitive to an image's characteristics (thus implying that a substantial degradation in performance might result if somehow certain errors happened in the filter's coefficients, e.g., due to transmission channel), and a design procedure (involving calculation of the correlation matrix and its inverse which is computationally intensive) must be invoked for each image under consideration.

Finally, the comparative study presented in this paper is not complete in many aspects, and more filter structures (in particular, nonlinear ones) must be considered. For example, we can design an optimal stack or weighted-order-statistic (WOS) filter. However, since such design procedures are quite complex and time-consuming, they are not the primary considerations in our future research. Alternatively, we will study certain filters that are easy to design and provide nearly the same performance as these optimal ones, such as weighted median filters,²⁴ feature-preserving filters,²⁵ and some adaptive filters.

Acknowledgments

This work was supported in part by the RGC Earmarked Research Grant 1995/96 (HKUST696/95E) and a research grant from Hongkong Telecom Institute of Information Technology (HKTIIT93/94.EG02).

References

1. R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ (1983).
2. A. J. Jerri, "The Shannon sampling theorem—its various extensions and applications: a tutorial review," *Proc. IEEE* **65**(11), 1565–1596 (1977).
3. G. J. Tonge, "The sampling of television images," Independent Broadcasting Authority, Report 112/81 (May 1981).
4. E. Dubois, "The sampling and reconstruction of time-varying imagery with applications in video systems," *Proc. IEEE* **73**(4), 502–522 (April 1985).
5. R. A. F. Belfor, R. L. Lagendijk, and J. Biemond, "Subsampling of digital image sequences using motion information," in *Motion Analysis and Image Sequence Processing*, M. I. Sezan and R. L. Lagendijk, Eds., Kluwer Academic, (1993).
6. G. Schamel, "Pre- and post-filtering of HDTV signals for sampling rate reduction and display up-conversion," *IEEE Trans. Circ. Syst.* **34**(11), 1432–1439 (1987).
7. B. Girod and W. Geuen, "Vertical sampling rate decimation and line-offset decimation of colour difference signals," *Signal Process.* **16**(2), 109–127 (1989).
8. P. Haavisto, Y. Neuvo, and J. Juhola, "Motion adaptive scan rate up-conversion," *Multidim. Syst. Signal Process.* **3**(5), 113–130 (1992).
9. L. Wang and M. Goldberg, "Comparative performance of pyramid data structures for progressive image transmission," *IEEE Trans. Commun.* **39**(4), 540–548 (1991).
10. G. C. Gurski, M. T. Orchard, and A. W. Hull, "Optimal linear filters for pyramidal decomposition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 633–636, San Francisco (March 1992).
11. M. Ernst and T. Reuter, "Adaptive filtering for improved standard conversion," in *Proc. 2nd Int. Workshop on HDTV*, pp. 449–458 (Feb./Mar. 1988).
12. C. Gumacos, "Weighting coefficients for certain maximally flat non-recursive digital filters," *IEEE Trans. Circ. Syst.* **25**(4), 234–235 (1978).
13. T. Ramstad, Y. Wang, and S. K. Mitra, "Efficient image interpolation scheme using hybrid IIR Nyquist filters," *Opt. Eng.* **31**(6), 1277–1283 (1992).
14. D. Chen and A. C. Bovik, "Visual pattern image coding," *IEEE Trans. Commun.* **38**(12), 2137–2146 (1990).
15. D. Chen and A. C. Bovik, "Hierarchical visual pattern image coding," *IEEE Trans. Commun.* **41**(4), 671–675 (1993).
16. A. Lehtonen and M. Renfors, "Nonlinear quincunx interpolation filtering," *Visual Communications and Image Processing, Proc. SPIE* **1360**, 135–142 (Oct. 1990).
17. B. Zeng, Y. Neuvo, and A. N. Venetsanopoulos, "Interpolative BTC image coding," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 493–496, San Francisco (Mar. 1992).
18. B. Zeng and A. N. Venetsanopoulos, "A comparative study of several nonlinear image interpolation schemes," in *Proc. Visual Communications and Image Processing '92*, pp. 21–29, Boston (Nov. 1992).
19. N. Herodotou and A. N. Venetsanopoulos, "Colour image interpolation for high resolution acquisition and display devices," *IEEE Trans. Consum. Electron.* **1118**–1126 (Nov. 1995).
20. R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Mag.* **4**–22 (Apr. 1987).
21. J. A. Greeman, D. M. Skapura, *Neural Networks: Algorithms, Applications, and Programming Techniques*, Addison-Wesley, Reading, MA (1991).
22. G. K. Wallace, "The JPEG still-picture compression standard," *Commun. ACM* **30**–44 (April 1991).
23. "JPEG digital compression and coding of continuous-tone still images," Draft ISO 10918 (1991).
24. R. Yang, L. Yin, M. Gabbouj, and Y. Neuvo, "2-D optimal weighted median filters under structural constraints," in *Proc. IEEE Workshop on Visual Signal Processing and Communications*, pp. 31–35 (Sep. 1992).
25. B. Zeng, "Optimal median-type filters under structural constraints," *IEEE Trans. Image Process.* **4**(7), 921–932 (1995).

Bing Zeng received his BEng and MEng degrees from the University of Electronic Science and Technology of China, Chengdu, in 1983 and 1986, respectively, and his PhD degree from Tampere University of Technology, Tampere, Finland, in 1991, all in electrical engineering. He was a postdoctoral fellow with the University of Toronto and Concordia University in 1991 and 1992. In January 1993, he became an assistant professor with the Department of

Electrical and Electronic Engineering, the Hong Kong University of Science and Technology. Dr. Zeng is a member of IEEE and has been an associate editor for the *IEEE Transactions on Circuits and Systems for Video Technology* since 1995. His research interests include digital signal and image processing, linear and nonlinear digital filter design, image/video coding and transmission, and neural networks. He has published over 60 referred papers in various journals and conference proceedings.

Anastasios N. Venetsanopoulos received his BEng degree from the National Technical University of Athens (NTU), Greece, in 1965, and his MS, MPhil, and PhD degrees in electrical engineering from Yale University, New Haven, Connecticut, in 1966, 1968, and 1969, respectively. He joined the University of Toronto in September 1968, where he has been a professor in the Department of Electrical and Computer Engineering since 1981. He

was chair of the Communications Group and associate chair of the

Department of Electrical and Computer Engineering. He has been on research leave at the Imperial College of Science and Technology, the National Technical University of Athens, the Swiss Federal Institute of Technology, the University of Florence, and the Federal University of Rio de Janeiro, and was also an adjunct professor at Concordia University. He has been a lecturer in over 140 short courses to industry and continuing education programs and a consultant to numerous organizations. He is a contributor to 24 books, a coauthor of 2 books, and has published more than 580 papers on digital signal and image processing and digital communications. Dr. Venetsanopoulos has chaired numerous boards, councils, and technical conference committees of the IEEE, was president of the Canadian Society for Electrical Engineering and vice president of the Engineering Institute of Canada (EIC) from 1983 to 1986, was a guest editor or associate editor for several IEEE journals, and was the editor of the *Canadian Electrical Engineering Journal* from 1981 to 1983. He is a member of the IEEE Communication, Circuits and Systems, Computer, and Signal Processing Societies and of Sigma Xi, the Technical Chamber of Greece, the European Association of Signal Processing, and the Association of Professional Engineers of Ontario (APEO) and Greece. He is a fellow of the IEEE and the EIC and received an honorary doctorate from the National Technical University of Athens in 1994. In 1996, he was awarded the Excellence in Innovation Award of the Information Technology Research Centre of Ontario.