Orthogonal pyramid transforms for image coding

Edward H. Adelson and Eero Simoncelli
MIT Media Laboratory, Cambridge, MA 02139

Rajesh Hingorani
SRI David Sarnoff Research Center, Princeton NJ 08540

Abstract:

We describe a set of pyramid transforms that decompose an image into a set of basis functions that are (a) spatial-frequency tuned, (b) orientation tuned, (c) spatially localized, and (d) self-similar. For computational reasons the set is also (e) orthogonal and lends itself to (f) rapid computation. The systems are derived from concepts in matrix algebra, but are closely connected to decompositions based on quadrature mirror filters. Our computations take place hierarchically, leading to a pyramid representation in which all of the basis functions have the same basic shape, and appear at many scales. By placing the high-pass and low-pass kernels on staggered grids, we can derive odd-tap QMF kernels that are quite compact. We have developed pyramids using separable, quincunx, and hexagonal kernels. Image data compression with the pyramids gives excellent results, both in terms of MSE and visual appearance. A non-orthogonal variant allows good performance with 3-tap basis kernels and the appropriate inverse sampling kernels.

Introduction:

A linear image transform expresses an image as a weighted sum of basis functions. The choice of transform (i.e. the choice of basis set) can be motivated by many factors. Much recent work in image coding has involved basis functions that are localized in both space and spatial frequency. Such transforms appear to be quite useful for various kinds of image analysis, because they give information about intensity changes over different scales, and yet retain information about where the events are occuring. There is also good evidence that the human visual system performs a similar image decomposition in its early processing.

Localization in space and spatial frequency:

The simplest way of achieving localization in both space and spatial frequency is to compute a discrete Fourier transform or a discrete cosine transform in a small block. But this introduces arbitrary block boundaries that violate the continuity of the image. Moreover the frequency localization of the basis functions is poor (when considered in the context of the image as a whole) due to the hard-edged windows defining the block.

Gabor (1946), working in the time domain, showed that functions produced as the products of complex sinusoids and Gaussian windows had optimal joint time-frequency localization, and he developed a transform using such functions as a basis set (cf. Daugman, 1985). At each position a set of basis functions analyzes a Gaussian-weighted patch into frequency components. The process is repeated for each patch. The Gabor functions do not form an orthogonal basis set, but they can form a complete representation.

Self-similarity:

A second theme in much recent work in image processing is "scaling," or "self-similarity" -- the notion that the basis functions should all have essentially the same shape but should be scaled versions of each other. The fundamental argument is that images contain information at all scales, and so it makes sense to capture this information in a uniform way. Scale invariance is violated by typical blocked or "short space" transforms. Block transforms, whether they use hard blocks (as in a blocked DCT) or soft ones (as in a Gabor transform) impose an arbitrary scale on the image analysis because the block is chosen to be some particular size. The problem of arbibary scale can be avoided in self-similar transforms, in which the basis functions come in many sizes.

Burt and Adelson (1983) coded images using the Gaussian and Laplacian pyramids. The basis functions are low-pass kernels repeated at a series of positions, and appear at scales varying by factors of two. The sampling functions are band-pass kernels, which are placed at corresponding positions and scales. The pyramid transform gives exact reconstruction, but the basis set is not orthogonal, and number of transform coeffi-cients exceeds the number of original pixels by a factor of 4/3.

Watson (1987) has developed a an oriented pyramid whose basis functions resemble Gabor functions and attain a spatial frequency and orientation tuning similar to that interred for channels in the human visual system. Again, the non-orthogonal basis set is overcomplete by 4/3. The reconstruction is nearly exact.

Our recent efforts have been to develop pyramids that use orthogonal basis functions (and therefore have the same number of transform coefficients as pixels), that incorporate orientation tuning, and that have good localization in space and spatial frequency. Such transforms should be useful for a variety of image processing tasks. The fundamental concepts can be illustrated by considering the Haar transform.

The Haar transform:

The Haar transform is the simplest self-similar orthonormal image transform. It is easy to understand and simple to compute. Its poor frequency tuning limits its utility in image processing; nonetheless the Haar transform serves as a useful tutorial device for much of what is to follow. It is special case of a QMF pyramid, as will become clear.

Figure 1 shows the construction of a Haar basis set for an image of 8 pixels. The original basis set (a) consists of 8 impulses, one at each position; these are split (b) into two classes of basis function (high-pass and low-pass), each appealing at 4 positions. The low-pass is further subdivided (c) into two more basis classes, each appearing at two positions;

and finally these are transformed into one low frequency basis function and one DC basis function. The resulting basis functions are all of the same shape (save the DC term), come in different scales, and are replicated on grids with a spacing proportional to their scale.

A two-dimensional Haar transform is readily computed by combining the 1-D basis transforms separably. The resulting 2-D basis functions are shown in figure 2(a). This is the 2-D formulation seen in textbooks unfortunately, it violates the original spirit of self-similarity. The simple application of separability leads to basis functions of widely varying shapes some some squarish and some long and thin.

A more attractive 2-D basis set is the one shown in figure 2(b) which was produced by separably combining the 1-D primitives (fig. l(b)) to form a set of four 2-D primitives. These were then applied recursively, subdividing only the low frequency basis functions from one stage to the next. Since the low-pass basis sets are always subdivided, the final pyramid basis set is composed of only three basic kernel shapes, which one may describe as taking vertical differences, horizontal differences, and diagonal differences.

Although the Haar basis functions have poor frequency selectivity, one can offer an idealized frequency-domain interpretation of the construction of the basis set shown in fig. 2(b); the interpretation is shown in figure 3. Each step of the process breaks the original band into four subbands: one tuned for low frequency, one tuned for vertical high-frequency, one tuned for horizontal high-frequency, and one tuned for both orientations of diagonal high-frequency.

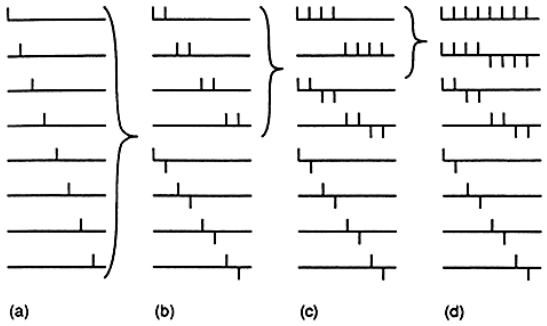

Figure 1. Haar basis functions can be constructed in a series of steps At each step the low-frequency basis set is split into two basis sets, one of high frequency and one of low frequency.

Fundamental concepts:

To derive better transforms let us review some basic ideas and notation. For simplicity consider a 1-D discrete image e[n], where n is the pixel index. An image transform expresses e[n] as a sum of basis functions $f_i[n]$ weighted by the coefficients p_i:

$$e[n] = \sum_i p_i f_i[n]$$

The coefficients p_i can be derived by taking the inner product of the image with a set of sampling functions g_i:

$$p_i = \sum_n g_i[n] e[n]$$

This may be expressed in matrix notation as follows: Let the signal be a column vector **e**. Define the basis matrix **F** to be composed of the columns f_i:

$$\mathbf{F} = \begin{bmatrix} \vdots & \vdots & \vdots & \\ f_0 & f_1 & f_2 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

Then we require that

$$\mathbf{e} = \mathbf{F}\mathbf{p}$$

where **p** is the column vector of coefficients. These coefficients are found by multiplying the image with the transpose of the sampling matrix **G**, where **G** is composed of the columns g_i. Thus,

$$\mathbf{p} = \mathbf{G}^T\mathbf{e}.$$

Then

$$\mathbf{e} = \mathbf{F}\mathbf{G}^T\mathbf{e}$$

so that G must be the transpose of the inverse of **F**, i.e.

$$\mathbf{G} = (\mathbf{F}^{-1})\mathbf{T}$$

Note that in the special case of an orthonormal transform the sampling functions are identical to the basis functions **G** = **F**. But in general this need not be the case.

We are particularly interested in the situation where the basis functions can be partitioned into a few classes, where the functions within a given class are shifted versions of each other. That is, we are interested in basis matrices of the form:

$$\mathbf{F} = \begin{bmatrix} f_a & & f_b & & \\ \vdots & f_a & \vdots & f_b & \cdots \\ & \vdots & f_a & \vdots & f_b \\ & & \vdots & & \vdots \end{bmatrix}$$

which lead to corresponding sampling matrices of the form:

$$\mathbf{G} = \begin{bmatrix} g_a & & g_b & & \\ \vdots & g_a & \vdots & g_b & \cdots \\ & \vdots & g_a & \vdots & g_b \\ & & \vdots & & \vdots \end{bmatrix}$$

An alternate description of this sort of transform is shown in the diagram of figure 4. The image is considered as a sequence e[n]; the kernels g_a, g_b,... are convolved with the image; downsampling by factors of k_a, k_b,... leads to the subimages $P_a[i]$, $P_b[i]$,... . These subimages form the transform representation. For reconstruction the subimages are upsampled and convolved with the basis kernels f_a, f_b, ... leading to the expanded subimages $q_a[n]$, $q_b[n]$,... . These are summed to retrieve the original image e[n].

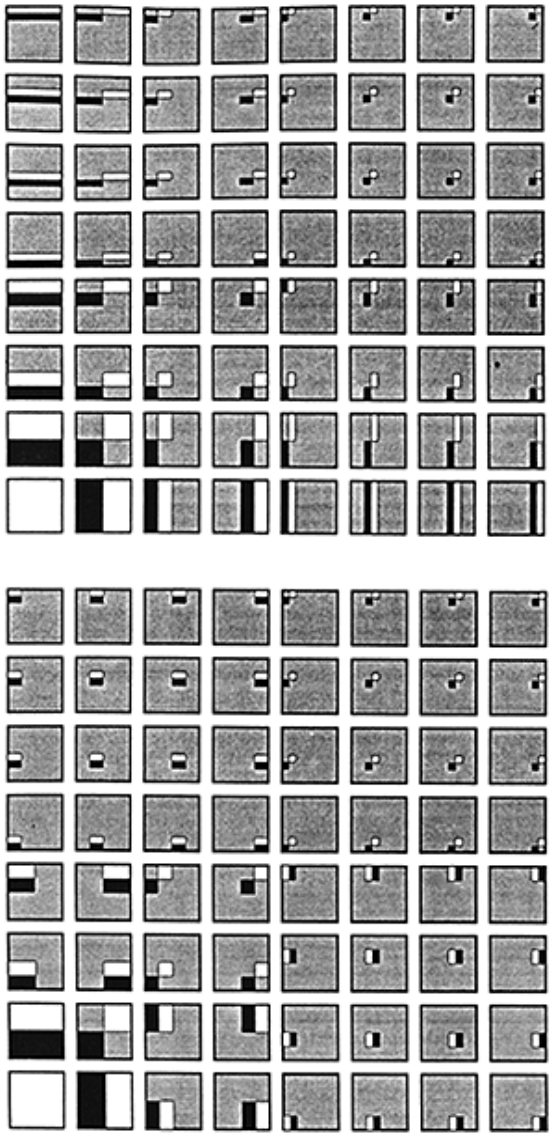

Figure 2: (a) When the 1-D Haar transform is applied separably, the resulting basis set contains kernels of many shapes, some long and thin. (b) A separable Haar pyramid can be constructed by recursively subdividing the 2-D low-frequency information at each step. In this case there are only 3 shapes of kernels, and they are repeated at various positions and various scales.

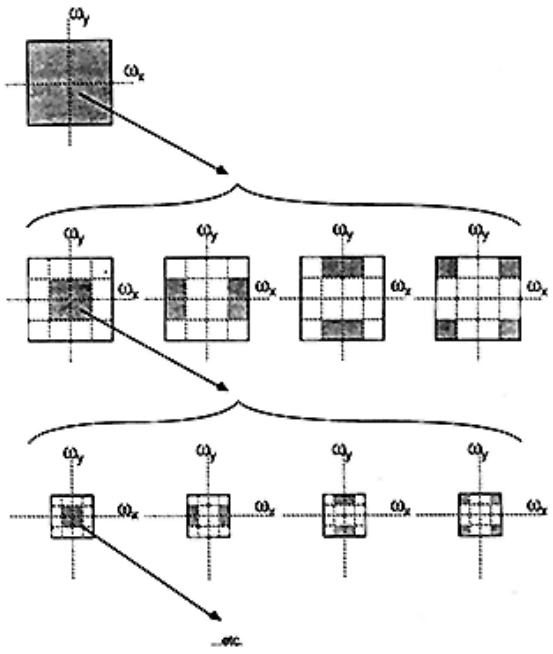

Figure 3: Frequency domain interpretation of separable pyramid construction. At each step the low-pass information is divided into 4 images: low-pass, horizontal, vertical, and diagonal.

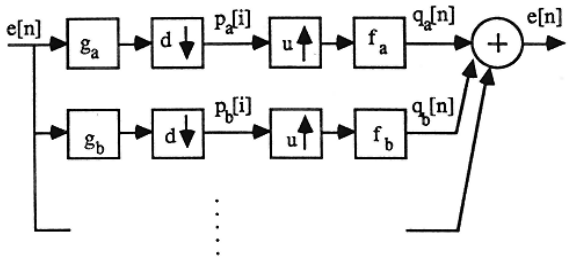

Figure 4: A signal processing diagram, corresponding to the application of a transform built of a small number of kernel types that are repeated and translated.

Band-splitting transforms:

One particular example is the case of band-splitting transforms, where there are two classes of kernel: high-pass and low-pass. If there are n pixels in the starting image there will be n/2 basis functions in each class.

The Haar transform is built recursively from a band-splitting transform. Such transforms can be applied recursively to build a pyramid. Most of the transforms we have explored are of this type.

Generally, band-splitting transforms are built from one primitive kernel. The low-pass basis set is achieved by translating this kernel by steps of two pixels. The high-pass kernel is derived from the low-pass kernel by multiplication with the modulating sequence:

$$s[n]= -1^n$$

The Haar basis functions are compact in space but do a poor job of separating the high and low frequency bands. Sinc functions ($(\sin \pi x)/\pi x$) offer the opposite extreme: they split the band sharply in frequency, but they are poorly localized in space. One can implement a self-similar sinc transform by successive subdivision of a discrete Fourier transform. Unfortunately the resulting basis functions, which are sinc functions modulated by cosinusoids, produce such poorly damped ringing that they are of little value for image analysis or image data compression.

Quadrature mirror filters.

Quadrature mirror filters were introduced by Estaban and Galand (1977) as a method of splitting a 1-D signal into sub-bands, efficiently decimating the subband signals, and achieving alias-free reconstructisn of the original signal. One can show that the QMF requirements, as normally applied, are equivalent to the requirement that a band-splitting transform have an orthogonal transformation matrix. The QMF basis functions can offer good localization in both space and spatial frequency, and may be cascaded to form a self-similar ortho-normal basis set. Moreover they can be tailored for efficient computation, and tuned for orientation as well as frequency.

Vetterli (1984), and Woods and O'Neil [1986] have extended QMP techniques to two dimensional signals. Pyramid versions of QMF decompositions have been applied to image coding by Gharavi and Tabatabai (1987) and Tran et al (1987).

Other workers outside the mainstream of digital signal processing have arrived independently at related formulations. Mallat (1987), working in the context of machine vision, has derived similar pyramids based an the wavelet theory of Meyer (1986, 1987) and other mathematicians working in France. In our own laboratory, orthogonal pyramids have been developed from a rather different point of view, originally in an effort to emulate some of the properties of the human visual system.

Because of our interests, we have approached the problem from the standpoint of orthonormal basis sets and matrix operations, rather than the standpoint of digital signal processing. This has led to a notation that is rather different from the DSP approach, but the results map readily into DSP terms.

Odd-tap kernels.

Almost all work on QMF representations to date has been based on low-pass kernels that have an even number of taps. This leads to high-pass kernels with odd symmetry. In general these kernels must be rather large (12 taps or more) in order to give satisfactory performance.

We have found that odd-tap kernels (which lead to even symmetric high-pass kernels) can be far more compact and still give good performance. These kernels have largely been overlooked, perhaps because they require a non-obvious manipulation: the high- and low-pass kernels must be staggered with respect to each other by one pixel. (In signal processing terms, the high-pass decimation must be preceded by a one-sample delay.)

Figure 5 illustrates the basis functions for a 5-tap QMF transform on an image of size 16. The low-pass functions, shown in the left column, are centered on the even-numbered pixel positions; the high-pass functions, shown in the right column, are centered on the odd-numbered pixel positions.

The illustrated case uses wraparound module 16, thus treating the image as periodic. In actual image coding it is preferable to use reflection at the edges, which leads to a somewhat more complex basis set.

Within an image of size N, odd-tap kernels cannot offer a perfectly orthogonal basis set unless they have at least (N+4)/2 taps. However they can approach orthogonality even with a rather small number of taps. For example, the mean squared reconstruction error for the Lena image is MSE = 0.103 (for a grey-level range of 0-255) when a 5-tap kernel is used in a one-level pyramid. By increasing the kernel size to 7, the MSE can be reduced to as little as .00009. However, the spatial shape and rather broad frequency tuning of the 5- and 7-tap kernels limit their value for image coding applications. For image coding we prefer to use 9-tap kernels, which are still quite compact. The 9-tap kernel with the sharpest tuning has MSE = 0.0044 for the Lena image. The error can be reduced at the expense of poorer frequency tuning; but for image processing applications this error is insignificant in any case.

The tap values for the 5-tap, 7-tap and 9-tap kernels we have used are shown in table 1. For an m-tap kernel, the values were derived by replicating the kernels within an image of size m+1 (with wraparound mod m+l), and imposing a set of constraints. For the 5- and 7-tap kernels the two constraints were: (1) orthonormality, and (2) zero DC gain for the high-pass kernel. This leads to a set of equations that can be readily solved to give unique solutions. In the case of the 9-tap kernel

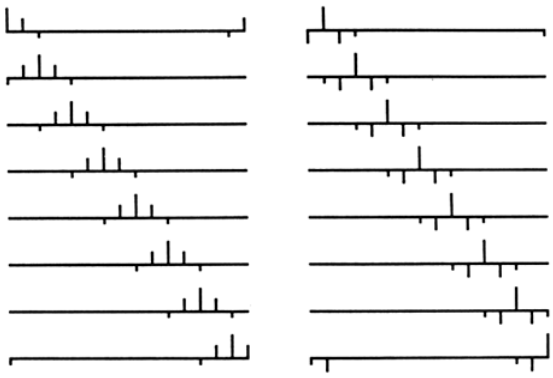

Figure 5: A 5-tap QMF basis set for a 16-pixel image. The basis set is approximately orthoganal; better orthogonaly (and thus more accurate reconstruction) is possible with more taps.

there is an extra degree of freedom; we imposed the added constraint that the Founer transform should have maximal slope through the $\pi/2$ point.

When the QMF kernels are cascaded hierarchically to create a multiscale pyramid, the equivalent kernel at each pyramid level grows in scale by factors of two. Figure 6 shows the 9-tap low-pass and high-pass kernels at various levels of the pyramid.

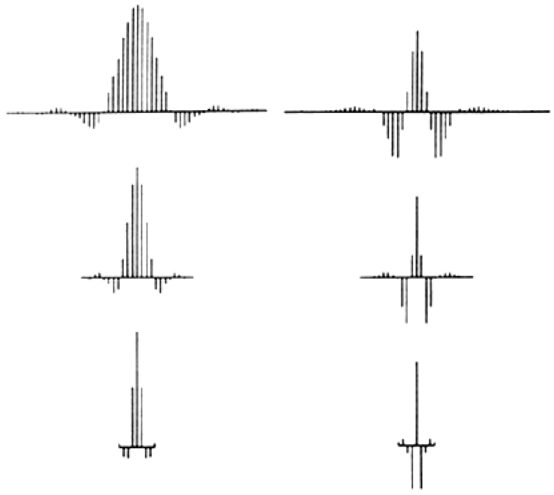

Figure 6: Left: the series of low-pass 1-D kernels derived recursively from a g-tap QMF kernel. Right: the series of high-pass kernels derived in the same way.

Two-dimensional QMF pyramids

Separable pyramids: The most straightforward way to generate 2-D QMF transforms is to apply two 1-D transforms separably (as was illustrated above with the Haar transform). This leads to a decomposition of the form shown in figure 7(a): at a given level, the image is decomposed into a low-pass subimage, a vertical subimage, a horizontal subimage, and a diagonal subimage. Also shown is the decimation pattern, which cuts the sampling density by a factor of 4. Thus one ends with 4 subimages, each sampled at l/4 the the density of the original image. The fact that the number of transform samples equals the number of image transforms follows immediately from the use of an orthogonal transform.

In the case of odd-tap kernels, there will be four such decimation grids, each offset to a different position; the interleaving of the four grids is shown in the inset of fig. 7(a) labeled "decimation pattern." For even-tap kernels, the four decimation grids coincide. Note that the diagonal subimage contains high frequencies in the corners of the frequency spectrum, corresponding to diagonal tilt in both directions.

The 2-D frequency responses of the 9-tap kernels are shown in figure 8.

Quincunx pyramids: Non-separable, non-oriented QMF pyramids can be made by successively dividing the spectrum into nested diamonds and squares, as shown in figure 9. At each stage the linear scale of the kernel increases by $\sqrt{2}$. This transform requires successive decimations on a quincunx grid (cf. Vetterli, 1984), as shown in fig. 7(b). With quincunx decimation the number of samples in 2-D is cut by l/2 at each stage.

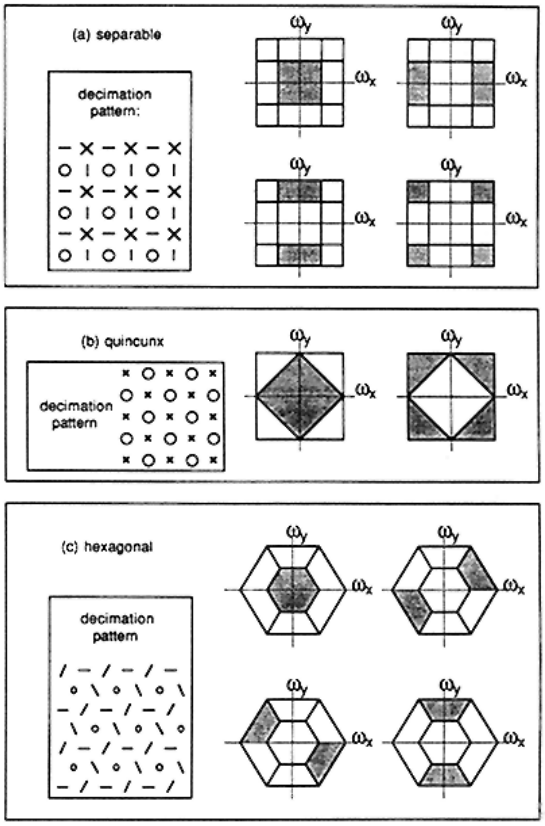

Figure 7: Three ways of building 2-D pyramids. (a) The frequency decomposition resulting from separable application of 1-D kernels. The decimation pattern is shown for the case of odd-tap kernels; in the case of even-tap kernels all four kernel types are applied on the same lattice. (b) The frequency decomposition resulting from application of quincunx kernels. (c) The frequency decomposition resulting from the application of hexagonal kernels.

A quincunx QMF kernel is given in table 2. This kernel although not optimally tuned, serves to illustrate that such kernels ean be made fairly compact. It occupies a diamond within a 7x7 square; 20 of the 49 taps are zero, and the kernel is highly symmetrical, so computations are not unreasonably expensive. The 2-D frequency response is shown in figure 10.

Hexagonal pyramids: Perhaps the most intriguing pyramids are those that can be built on a hexagonally sampled image using hexagonally symmetric kernels. As illustrated in figure 7(c), it becomes possible to divide the spectrum into three oriented subimages and one low-pass subimage. Again, this process can be applied recursively to build a pyramid. The decimation pattern reduces the sampling density by a factor of 4 at each stage.

The hex pyramid has the unique property that all of its high-pass kernels are oriented and all have the same shape. (Recall that with the separable pyramid, the horizontal and vertical kernels are the same but the diagonal one is different.). An example of a reasonably compact hex kernel is shown in table 3.

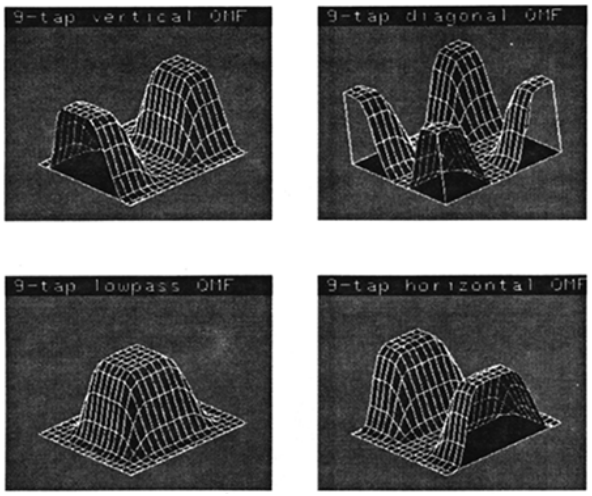

Figure 8: The two-dimensional discrete-space Fourier transforms for the 9-tap QMF kernels.

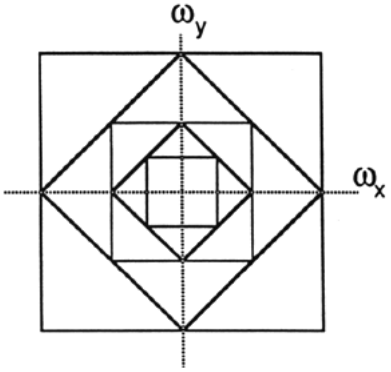

Figure 9: The nesting pattern of the frequency spectra for a quincunx pyramid.

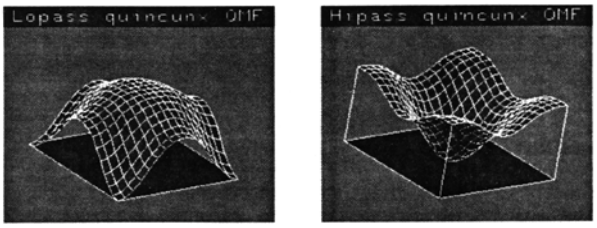

Figure 10: The Fourier transforms for the quincunx QMF kernels.

Bandsplitting inverse pairs (BIP filters):

In transforms based on QMF kernels, the basis set is orthonormal and so the sampling functions are identical to the basis functions. But orthogonality is not required for image coding: it is only necessary that the transform be invertible. In general, then, the sampling functions can be quite different from the basis functions. As noted above, the sampling matrix for a non-orthogonal basis set is readily derived as the transpose of the inverse of the basis matrix.

Suppose that one wishes to use an extremely simple decoder at the expense of a more complex encoder. A very simple band-splitting transform can be built from the 3-tap low-pass and high-pass kernels:

[.25 .5 .25] and [-.25 .5 -.25].

These kernels are not a proper QMF pair and do not lead to an orthogonal basis set. Nonetheless, when placed on a staggered grid they do form a linearly independent basis set. That is, the basis matrix **F** is non-singular, and therefore the corresponding sampling matrix, **G**, is derived as

$$\mathbf{G} = (\mathbf{F}^{-1})^{T}.$$

The resulting sampling kernels take the form shown in table 4. The true inverse sampling function has non-zero taps over the entire image; however we show a 15-tap version that has been derived by truncating it and scaling the taps to achieve orthogonality and a DC system gain of unity. We have found this 15-tap sampling function to work quite well in coding images. (A more accurate 21-tap version is also shown).

The 3-tap kernels are extremely easy to compute, using only shifts and adds, with no multiplications. They can be combined separably to produce 2-D kernels. In this case the corresponding 2-D sampling functions are just the separable products of the 1-D sampling functions.

Application to image coding:

QMF pyramids perform admirably in the task of image data compression. We report here on results with the compact odd-tap kernels of table 1. A 9-tap separable pyramid was built, each sub-image was quantized, and then the coded image was reconstructed. Entropies were computed for each level, and combined to give the net bit rate for the image. We show results for the Lena image at 256x256 pixels.

Figure 11(a) shows the Lena original; fig. 11(b) shows the image coded at 0.65 bit/pixel using the 9-tap kernel. Figure 11(c) is also coded at 0.65 bit/pixel, but using the 3-tap BIP pyramid. Both images look quite good considering the low bit rate. The 9-tap kernel performs somewhat better, both in terms of MSE and in terms of visual appearance. However, the performance of the 3-tap pyramid is impressive: since 3-tap reconstruction is so simple it may be useful in applications requiring fast and inexpensive decoding hardware.

We compared the results of this pyramid coding with 16x16 block DCT coding using the algorithm of Chen and Pratt (1984). The 9-tap QMP pyramid's performance, as measured by mean squared error, was typically equal to that of the DCT within about 5%, varying slightly with from image to image. However the QMF pyramid's visual appearance was noticeably better than that of the DCT. At low bit rates, the DCT suffers from a blocky appearance, whereas for the pyramid the noise is distributed in a less noticeable manner.

Figure 11: Data compression of the 256x256 Lena image. (a) Original. (b) 9-tap pyramid, 0.85 bit/pixel. (c) 3-tap pyramid, 0.65 bit/pixel.

Figure 12: Progressive transmission with three types of pyramid. Top row: 9-tap QMF pyramid. Bit rates for the 9-tap sequence are: 0.03, 0.09, 0.24, 0.49 bit/pixel. Rates for the other sequences are essentially the same. The final image, at 0.65 bit/pixel is not shown (see fig. 11).

Note that we are using the pyramids in a fairly "raw" state in this comparison -- we have not yet combined them with other methods such as vector quantization and adaptive techniques. Other workers (Woods and O'Neil, 1986; Tran et al, 1987) have shown that more elaborate coding can lead to further improvements in the performance of QMF coders. Because our kernels are so compact, we expect that they will combine especially well with spatially adaptive methods.

Progressive transmission:

For many applications it is desirable that a low resolution version image become available quickly, and that higher resolution becomes available as time goes on; this is known as progressive transmission. Many image coding techniques can be modified to allow progressive transmission. But in the case of pyramids, which are inherently multiscale representations, progressive transmission is achieved simply by sending information from successive pyramid levels in sequence. This adds no informational overhead.

Figure 12 shows a series of progressive transmission images for three different types of pyramid. The first row is for the 9-tap QMF pyramid; the second row is for the 3-tap BIP pyramid; and the third row is for the hexagonal QMF pyramid. The successive images show reconstructions from levels 5, 4, 3, and 2 of the pyramid (reconstruction from level 1 reproduces the original image; this is not shown.) These pyramids were also quantized for data compression, and the entropies are shown in the figure caption. A recognizable approximation to the image becomes available very quickly. The 9-tap QMF pyramid gives the best results, but the 3-tap BIP works almost as well. The hex pyamid works fairly well but introduces hexagonal artifacts, which we hope to reduce in the future with improved kernel design. Again, since the 3-tap reconstruction is so easy to implement, it may be useful in devices that must be fast and inexpensive.

Summary

Pyramids built with QMF kernels have a number of desirable properties. Their basis functions are localized in space and spatial frequency, and appear at all scales in a self-similar fashion. The hierarchical computation is highly efficient. Since the basis set is orthonormal, the number of coefficients in the transform is equal to the number of pixels in the original image.

The simplest QMF pyramid is based on the Haar transform; but its localization in frequency is so poor that it is of little use. Practical QMF pyramids require larger kernels. Well-behaved kernels with an even number of taps tend to be rather large. On the other hand, odd-tap kernels can be more compact than the commonly used even-tap kernels. To form an orthogonal basis set, these kernels must be applied on staggered grids. Kernels with widths as small as 7 can produce highly accurate reconstructions. A kernel of width 9 was found to have good performance for efficient image coding.

When 1-D QMF kernels are applied separably in two dimensions they divide an image into 4 subimages: a low-pass a vertical, a horizontal, and a diagonal subimage. The vertical and horizontal subimages are tuned for orientation as well as spatial frequency, a property that may be useful for various image processing tasks. Cascading the operations, always using the low-pass image from the prior level, leads to an efficiently computed self-similar image pyramid.

Non-separable kernels can also be used. A pyramid that scales by factors of $\sqrt{2}$ can be created with a non-separable kernel applied on a quincunx grid. A hexagonal QMF pyramid that scales by octaves can also be produced, it has three-fold symmetry and self-similarity, and is composed entirely of oriented basis functions.

QMF pyramids perform quite well for image data compression. The 9-tap separable pyramid, when used with simple entropy coding, gives performance equal to a blocked DCT in terms of mean squared error, and superior to it in terms of perceptual degradation.

Band-splitting inverse pair (BIP) kernels offer a useful alternative to standard QMF kernels. These form a basis set that is linearly independent but not orthogonal. The appropriate sampling functions, derived by inverting the basis matrix, are therefore different from the basis functions. We find that a 3-tap BIP pyramid, when used in conjunction with a 15-tap inverse-pair sampling kernel, gives data compression performance almost as good as that of a 9-tap QMF pyramid.

QMF and BIP pyramids are particularly well-suited to progressive transmission. Since the representations are inherently multi-resolution one can generate efficient progressive transmission systems by simply sending the data in the right sequence.

References:

Burt, P. J., and Adelson, E. H., "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun*., v. COM-3 1, pp. 532-540 (1983).

Chen, W., and Pratt, W., "Scene adaptive coder," *IEEE Trans. Comm*. v. COM-32 225-232 (1984).

Daugman, J. G., "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *J. Opt. Soc. Am*., A2 1160-1169 (1985).

Estaban, D., and Galand, C., "Application of quadrature minor filters to split band voice coding schemes," *Proc. ICASSP*, pp. 191-195 (1977).

Gabor, D., "Theory of Communication," *J. IEE*, no. 93, pp 429-457 (1946).

Gharavi, H, and Tabatabai, A., "Application of quadrature mirror filtering to the coding of monochrome and color images," *Proc. ICASSP*, 2384-2387 (1987).

Mallat, S. G., "A theory for multiresolution signal decomposition: the wavelet representation," GRASP Lab Technical Memo MS-CIS-87-22, University of Pennsylvania Dept. of Computer and Information Science (1987).

Meyer, Y. "Principe d'incertitude, bases hilbertiennes et algebres d'operateurs," *Bourbaki seminar*, no. 662 1985-1986.

Meyer, U., "Ondelettes, fonctions splines et analyses graduees," (unpublished manuscript, 1987).

Tran, A., Liu, K., Tzou, K, and Vogel, E., "An efficient pyramid image coding scheme," *Proc. ICASSP*, 744-747 (1987).

Vetterli, M., "Multidimensional subband coding: Some theory and algoritmns," *Signal Processing*, v. 6, pp. 97-112 (1984).

Watson, A. B., "The cortex transform" *Comp. Vis. Graph. and Im. Proc*. (in press,1987).

Woods, J. W., and O'Neil, S. D., "Subband coding of images," *IEEE Trans. Acoust. Spcech and Sig. Proc.* v. ASSP-34 1278-1288 (1986).

Acknowledegments: We thank Robert Hummel and Ronald Sverdlove for useful discussions. This work was begun when the first and second author were at the David Sarnoff Research Center, Princeton, NJ.

Tap #	5-tap	7-tap	9-tap
0	0.60762	0.60355	0.56458
1	0.25000	0.25525	0.29271
2	-0.05381	-0.05178	-0.05224
3		0.00525	-0.04271
4			0.01995

Table 1: tap values for odd tap lowpass QMF kernels. Lowpass kernel is shown, with center tap indexed to zero; corresponding high-pass kernels are derived by multiplication with -1^n. Kernels are scaled for unity DC gain.

Tap #	3-tap	Inverse: 15-tap	Inverse: 21-tap
0	0.5	1.41682	1.41415
1	0.25	0.58585	0.58569
2		-0.24274	-0.24261
3		-0.10056	-0.10049
4		0.04169	0.04163
5		0.01717	0.01724
6		-0.00736	-0.00714
7		-0.00245	-0.00296
8			0.00122
9			0.00052
10			-0.00017

Table 4: Band-splitting inverse pair (BIP) kernels: 3-tap and its inverse-pair sampling kernel, shown in 15-tap and 21-tap versions. Low-pass kernel and sampling function are shown, centered on tap 0. The high-pass kernel and the high-pass sampling function are obtained by multiplying the lowpass functions with -1^n.

```
          e                          -e                 Example
        f  c  f                    -f  c -f             tap values:
      f  d  b  d  f              -f  d -b  d -f         a =  .65549
    e  c  b  a  b  c  e        -e  c -b  a -b  c -e     b =  .12930
      f  d  b  d  f              -f  d -b  d -f         c = -.01294
        f  c  f                    -f  c -f             d = -.02594
          e                          -e                 e = -.00118
                                                        f = -.00156
```

Table 2: Quincunx 2-D QMF kernels, non-separable. High-pass kernel is obtained from low-pass by multiplication with $-1^{(n+m)}$. Tap values achieve near orthogonality but are not optimized for tuning.

```
      f   e   e   f              -f  -e  -e  -f          Example
    e   d   c   d   e            e   d   c   d   e        tap values:
  e   c   b   b   c   e        -e  -c  -b  -b  -c  -e      a  .34675
f   d   b   a   b   d   f    f   d   b   a   b   d   f    b  .14527
  e   c   b   b   c   e        -e  -c  -b  -b  -c  -e      c -.01597
    e   d   c   d   e            e   d   c   d   e        d -.01619
      f   e   e   f              -f  -e  -e  -f          e -.00143
                                                        f -.00137
```

Table 3: Hexagonal QMF kernels. Only the low-pass and the horizontal high-pass are shown; the two other high-pass kernels are the same except for axis. Tap values achieve near orthogonality but are not optimized for tuning.