A Multiresolution Spline With Application to Image Mosaics

PETER J. BURT and EDWARD H. ADELSON
RCA David Sarnoff Research Center

We define a multiresolution spline technique for combining two or more images into a larger image mosaic. In this procedure, the images to be splined are first decomposed into a set of band-pass filtered component images. Next, the component images in each spatial frequency hand are assembled into a corresponding bandpass mosaic. In this step, component images are joined using a weighted average within a transition zone which is proportional in size to the wave lengths represented in the band. Finally, these band-pass mosaic images are summed to obtain the desired image mosaic. In this way, the spline is matched to the scale of features within the images themselves. When coarse features occur near borders, these are blended gradually over a relatively large distance without blurring or otherwise degrading finer image details in the neighborhood of th e border.

Categories and Subject Descriptors: I.3.3 [**Computer Graphics**]: Picture/Image Generation; I.4.3 [**Image Processing**]: Enhancement

General Terms: Algorithms

Additional Key Words and Phrases: Image mosaics, photomosaics, splines, pyramid algorithms, multiresolution analysis, frequency analysis, fast algorithms

1. INTRODUCTION

The need to combine two or more images into a larger mosaic has arisen in a number of contexts. Panoramic views of Jupiter and Saturn have been assembled for multiple images returned to Earth from the two Voyager spacecraft. In a similar way, Landsat photographs are routinely assembled into panoramic views of Earth. Detailed images of galaxies and nebulae have been assembled from mul-

The work reported in this paper was supported by NSF grant ECS-8206321. A shorter description of this work was published in the *Proceedings of SPIE, vol. 432, Applications of Digital Image Processing VI*, The International Society for Optical Engineering, Bellingham, Washington.
Authors' address: RCA David Sarnoff Research Center, Princeton, NJ 08540.

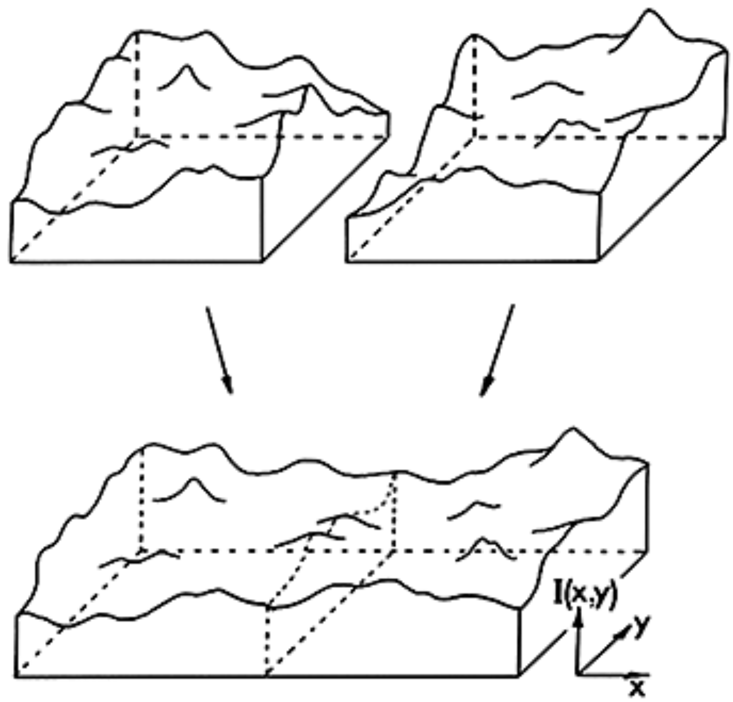


Fig. 1. A pair of images may be represented as a pair of surfaces above the (x, y) plane. The problem of image splining is to join these surfaces with a smooth seam, with as little distortion of each surface as possible.

tiple telescope photographs. In each of these cases, the mosaic technique is used to construct an image with a far larger field of view or level of detail than could be obtained with a single photograph. In advertising or computer graphics, the technique can be used to create synthetic images from possibly unrelated components.

A technical problem common to all applications of photomosaics is joining two images so that the edge between them is not visible. Even slight differences in image gray level across an extended boundary can make that boundary quite noticeable. Unfortunately, such gray level differences are frequently unavoidable; they may be due to such factors as differences in camera position or in image processing prior to assembly. Thus, a technique is required which will modify image gray levels in the vicinity of a boundary to obtain a smooth transition between images. The two images to be joined may be considered as two surfaces, where the image intensity $I(x, y)$ corresponds to the elevation above the x,y plane. The problem, as illustrated in Figure 1, may be stated as follows: How can the two surfaces be gently distorted so that they can be joined together with a smooth seam? We will use the term *image spline* to refer to digital techniques for making these adjustments. A good image spline will make the seam perfectly smooth, yet will preserve as much of the original image information as possible.

It is probably safe to say that no fully satisfactory splining technique has yet been found. Most image mosaics are still produced without any attempt at remov-

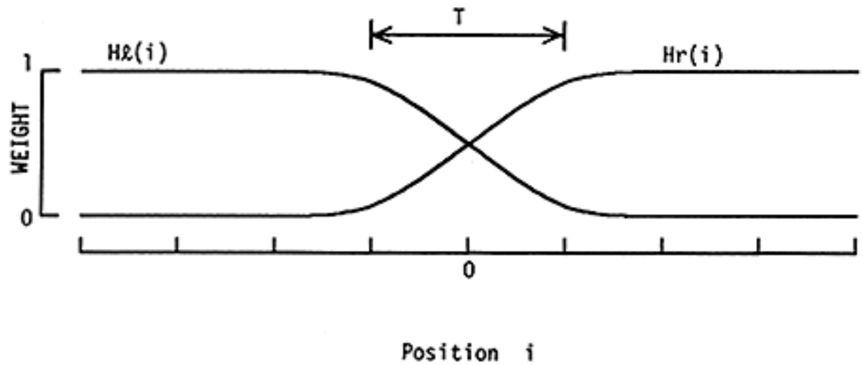


Fig. 2. The weighted average method may be used to avoid seams when mosaics are constructed from overlapped images. Each image is multiplied by a weighting function which decreases monotonically across its border; the resulting images are then summed to form the mosaic. Example weighting functions are shown here in one dimension. The width of the transition zone T is a critical parameter for this method.

ing visible boundaries (e.g., [4]). The magnitude of the gray level difference across a mosaic boundary can be reduced to some extent by a judicious choice of boundary location when splining overlapped images. The match may be improved by adding a linear ramp to pixel values on either side of the boundary to obtain equal values at the boundary itself [6, 7]. A still smoother transition can be obtained using a technique recently proposed by Peleg [9]: The "smoothest possible" correction function is constructed which can be added to each image of a mosaic to eliminate edge differences. However, this technique may not be practical for large images, since the correction functions must be computed using an iterative relaxation algorithm.

We are concerned with a weighted average splining technique. To begin, it is assumed that the images to be joined overlap so that it is possible to compute the gray level value of points within a transition zone as a weighted average of the corresponding points in each image. Suppose that one image, $Fl(i)$, is on the left and the other, $Fr(i)$, is on the right, and that the images are to be splined at a point $\hat{i}$ (expressed in one dimension to simplify notation). Let $Hl(i)$ be a weighting function which decreases monotonically from left to right and let $Hr(i) = 1 - Hl(i)$ (see Figure 2). Then, the splined image F is given by

$$F(i) = Hl(i - \hat{i})\, Fl(i) + Hr(i - \hat{i})\, Fr(i).$$

It is clear that with an appropriate choice of H, the weighted average technique will result in a transition which is smooth. However, this alone does not ensure that the location of the boundary will be invisible. Let T be the width of a transition zone over which Hl changes from 1 to 0. If T is small compared to image features, then the boundary may still appear as a step in image gray level, albeit a somewhat blurred step. If, on the other hand, T is large compared to image features, features from both images may appear superimposed within the transition zone, as in a photographic double exposure.

These extremes are illustrated in Figure 3 with several attempts to spline two synthetic images of stars. The original images, Figures 3a and 3b (257 x 257 pixels) are identical except for a slight shift in vertical position and a slight shift in mean gray level. The first of these differences can arise from optical distortions or misalignments of actual photographs, while the second can be due to differences in atmospheric conditions or in photographic development.

In this example, photomosaics are obtained by joining the left half of Figure 3a with the right half of Figure 3b. If this is done without any attempt to smooth the image transition ($T = 0$), the boundary will appear as a sharp edge (Figure 3c). If instead the images are combined by the method of weighted average within a relatively narrow transition zone ($T = 8$), the edge appears blurred but remains visible (Figure 3d). When the images are splined with a broad transition ($T = 64$), the edge is no longer visible but stars have a "double exposed" look within the transition zone (Figure 3e).

Clearly, the size of the transition zone, relative to the size of image features, plays a critical role in image splining. To eliminate a visible edge the transition width should be at least comparable in size to the largest prominent features in the image. On the other hand, to avoid a double exposure effect, the zone should not be much larger than the smallest prominent image features. There is no choice of T which satisfies both requirements in the star images of Figure 3 because these contain both a diffuse background and small bright stars.

These constraints can be stated more precisely in terms of the image spatial frequency content. In particular, a suitable T can only be selected if the images to be splined occupy a relatively narrow spatial frequency band. As a rough requirement, we may stipulate that T should be comparable in size to the wavelength of the lowest prominent frequency in the image. If T is smaller than this the spline will introduce a noticeable edge. On the other hand, to avoid a double exposure effect, T should not be much larger than two wave lengths of the highest prominent frequency component in the images. This ensures that there will not be room for multiple features within the transition zone. While it is likely that these limits can be exceeded somewhat without noticeable degradation, the general conclusion—that the band width of images to be splined should be roughly one octave—is an important one.

How can images which occupy more than an octave be splined? The approach proposed here is that such images should first be decomposed into a set of band-pass component images. A separate spline with an appropriately selected T can then be performed in each band. Finally, the splined band-pass components are recombined into the desired mosaic image. We call this approach the multi-resolution spline. It was used to obtain the image shown in Figure 3f.

In decomposing the image into frequency bands, it is important that the range of frequencies in the original be covered uniformly, although the bands themselves may overlap. As a practical matter, a set of low-pass filters are applied to generate a sequence of images in which the band limit is reduced from image to image in one-octave steps. Band-pass images can then be obtained simply by subtracting each low-pass image from the previous image in the sequence. This not only ensures complete coverage of spatial frequencies but means that the final mosaic can be obtained simply by summing the band-pass component images.

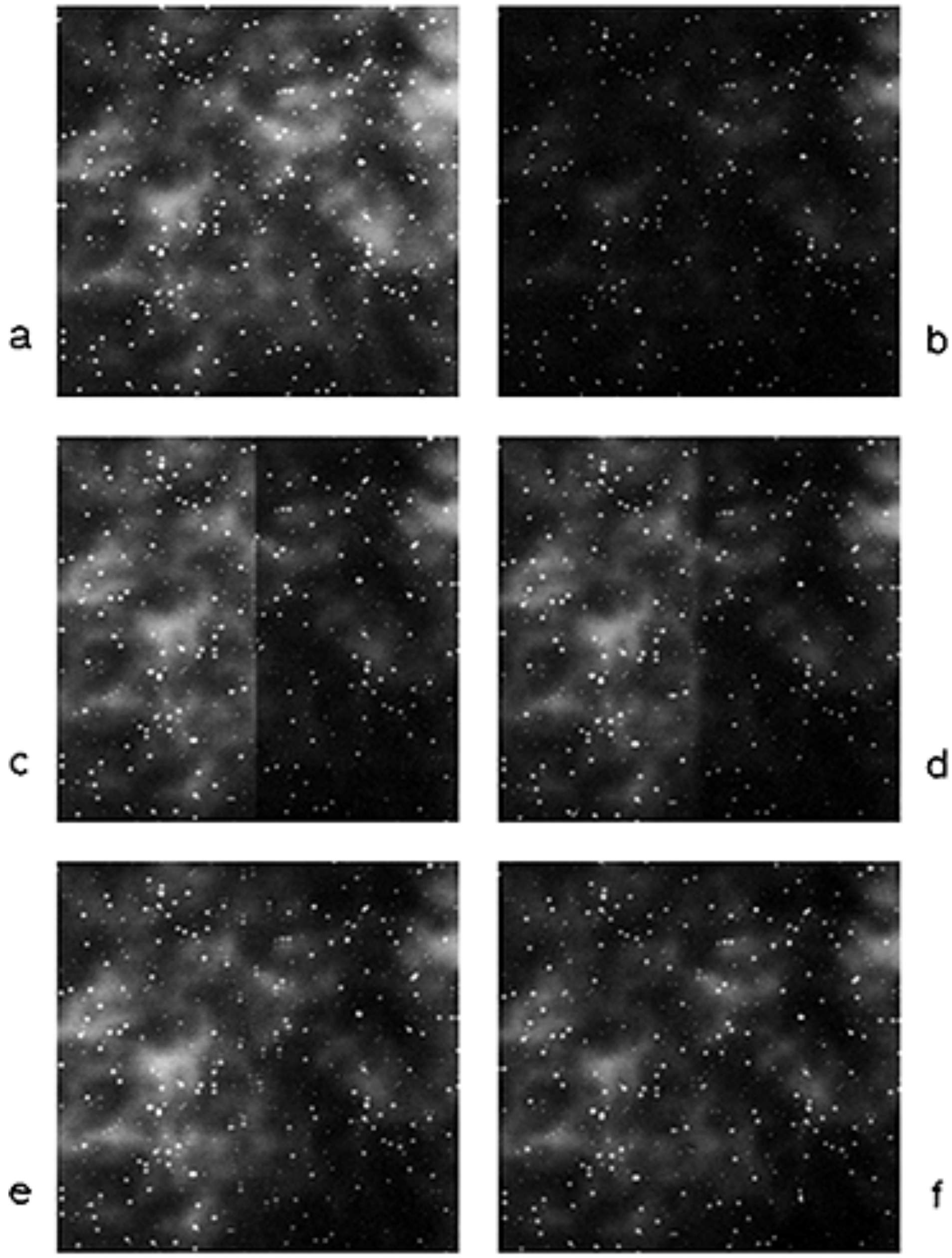


Fig. 3. Common artifacts of the weighted average techniques are demonstrated in these attempts to spline two synthetic images of stars (Figure 3a and 3b). These differ only in mean gray level and a slight vertical shift. A seam is clearly visible when the left half of figure 3a is joined with the right half of Figure 3b without any adjustment in gray level, as shown in Figure 3c. The seam is still visible when the weighted average technique is used with a narrow transition zone (Figure 3d). However, if the transition zone is wide, features within the zone appear double (Figure 3e). The first of these artifacts is due to a gray level mismatch at low spatial frequencies, while the second is due to a position mismatch at high frequencies. Both are avoided in the multiresolution method (Figure 3f).

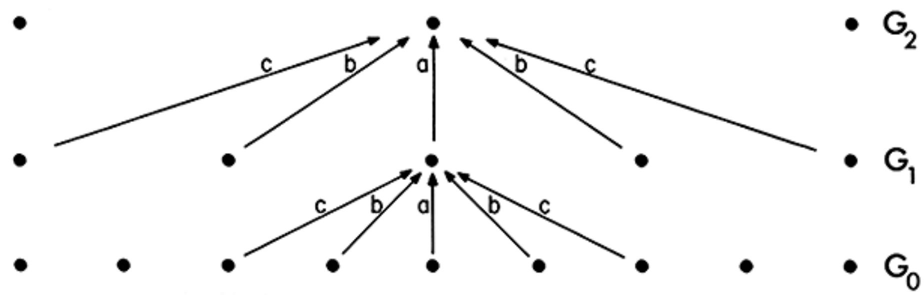


Fig. 4. A one-dimensional graphical representation of the iterative REDUCE operation used in pyramid construction.

In Section 2 we present a highly efficient "pyramid" algorithm for performing the required filtering operations and in Section 3 we show that the pyramid structure is ideally suited for performing the splining steps as well.

2. BASIC PYRAMID OPERATIONS

A sequence of low-pass filtered images G_0, G_1, . . ., G_N can be obtained by repeatedly convolving a small weighting function with an image [1, 3]. With this technique, image sample density is also decreased with each iteration so that the bandwidth is reduced in uniform one-octave steps. Sample reduction also means that the cost of computation is held to a minimum.

Figure 4 is a graphical representation of the iterative filtering procedure in one dimension. Each row of dots represents the samples, or pixels, of one of the filtered images. The lowest row, G_0, is the original image. The value of each node in the next row, G_1, is computed as a weighted average of a 5 x 5 subarray of G_0 nodes, as shown Nodes of array G_2 are then computed from G_1 using the same pattern of weights. The process is iterated to obtain G_2 from G_1, G_3 from G_2 and so on. The sample distance is doubled with each iteration so that successive arrays are half as large in each dimension as their predecessors. If we imagine these arrays stacked one above the other, the result is the tapering data structure known as a pyramid [10]. If the original image measures $2^N + 1$ by $2^N + 1$, then the pyramid will have $N + 1$ levels.[1]

Both sample density and resolution are decreased from level to level of the pyramid. For this reason, we shall call the local averaging process which generates each pyramid level from its predecessor a REDUCE operation. Again, let G_0 be the original image. Then for $0 < l < N$:

$$G_l = \text{REDUCE}\,[G_{l-1}],$$

by which we mean:

$$G_l(i,j) = \sum \sum_{m,n=1}^{5} w(m,n) G_{l-1}(2i+m, 2j+n).$$

More generally, a pyramid of $N + 1$ levels may be constructed from any image which measures $M_R 2^N + 1$ rows by $M_C 2^N + 1$ columns, where M_R and M_C are integers.

The pattern of weights $w(m, n)$ used to generate each pyramid level from its predecessor is called the generating kernel. These weights are chosen subject to four constraints: First, for computational convenience, the generating kernel is separable, $w(m, n) = \hat{w}(m) \ \hat{w}(n)$. Second, the one dimensional function $\hat{w}$ is symmetric, $\hat{w}(0) = a$, $\hat{w}(-1) = \hat{w}(1) = b$, and $\hat{w}(-2) = \hat{w}(2) = c$, as shown in Figure 4. Third, $\hat{w}$ is normalized, $a + 2b + 2c = 1$. The final constraint stipulates that each level l node must contribute the same total weight to level $l + 1$ nodes: thus, $a + 2c = 2b$. Now, combining constraints, we find that a may be considered a free variable, while $b = 1/4$ and $c = 1/4 - a/2$.

2.1 Equivalent Weighting Functions

It is clear that every level l node in the pyramid represents a weighted average of a 5×5 subarray of level $l - 1$ nodes. Each of these in turn represents an average of a subarray of level $l - 2$. In this way, we can trace the weights for a given pyramid node back to the original image G_0 to discover the "equivalent weighting function" W_l which, if convolved directly with the original image, would have given the same node values at level l. It is convenient to discuss pyramid-based computations in terms of these equivalent weighting functions, although the iterative REDUCE process is considerably more efficient and is used in all computations.

The equivalent weighting functions have several properties which will be important in filtering and splining operations. The scale of these functions doubles from level to level of the pyramid while their shape does not change [1]. Function shape does depend on the value of parameter a in the generating kernel. For example, if $a = 0.5$, the functions are all triangular in shape, while if $a = 0.4$, the functions resemble the Gaussian probability density function. Convolution with a Gaussian has the effect of low-pass filtering the image. Pyramid construction is equivalent to convolving the image with a set of Gaussian-like functions to produce a corresponding set of filtered images. Because of the importance of the multiple filter interpretation, we shall refer to this sequence of images G_0, $G_1, \ldots, G_N$ as the Gaussian pyramid.

Suppose samples in G_0 are separated by a unit distance. Then, samples at level l are separated by the distance 2^l. It can be shown that the width of the equivalent weighting function W_l is $2^{l+2} - 4$, covering $2^{l+2} - 3$ image samples, or just less than four times the sample distance (see Figure 5). Thus equivalent weighting functions centered on level l sample points will overlap in such a way that each image pixel contributes to the value of at most 16 level-l samples (4 in one dimension). If the contributions of any image pixel are summed, the result will be unity. For each i, j, and l,

$$\sum_{m,n=-2}^{2} W_l(i - m2^l, j - n2^l) = 1.$$

This result follows from the equal contribution property of the generating kernel.

The Gaussian shape and summation properties of the functions W_l mean they can be used to construct the weighting functions H needed for image splining

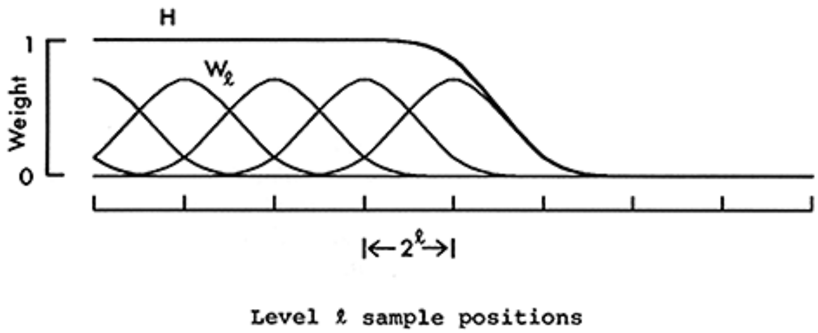


Fig. 5. Equivalent weighting functions, W_l, are shown centered at level l sample points on the left in the figure, while zero weight is given to points on the right. When these weights are summed, a uniform value of unity is obtained on the left, a value of zero on the right, and a monotonic transition in the center. The weighting functions H used in image splining can be constructed in this way (see Figure 2).

(Figure 2). Suppose W_l is associated with each node in the left half of G_l while zero weight is associated with nodes in the right half (Figure 5). Then, the sum of these functions will be a function which decreases monotonically from 1 to 0, with a transition zone width T equal to 3 times the level l sample interval. This property will be used in the pyramid-based multiresolution spline, although functions H and W will never be explicitly computed.

2.2 The Laplacian Pyramid

The Gaussian pyramid is a set of low-pass filtered images. In order to obtain the band-pass images required for the multiresolution spline we subtract each level of the pyramid from the next lowest level. Because these arrays differ in sample density, it is necessary to interpolate new samples between those of a given array before it is subtracted from the next lowest array. Interpolation can be achieved by reversing the REDUCE process. We shall call this an EXPAND operation. Let $G_{l,k}$ be the image obtained by expanding G_l k times. Then

$$G_{l,0} = G_l,$$

and for $k > 0$,

$$G_{l,k} = \text{EXPAND}[G_{l,k-1}].$$

By EXPAND, we mean

$$G_{l,k}(i,j) = 4\sum \sum_{n,n=-2}^{2} G_{l,k-1}\left(\frac{2i+m}{2},\frac{2j+n}{2}\right).$$

Here, only terms for which $(2i + m)/2$ and $(2j + n)/2$ are integers contribute to the sum. Note that $G_{l,1}$ is the same size as G_{l-1}, and that $G_{l,l}$ is the same size as the original image.

We now define a sequence of band-pass images $L_0, L_1, \ldots, L_N$. For $0 < l < N$,

$$L_l = G_l - \text{EXPAND}[G_{l+1}] = G_l - G_{l+1,1}.$$

Because there is no higher level array to subtract from G_N, we define $L_N = G_N$.

Just as the value of each node in the Gaussian pyramid could have been obtained directly by convolving the weighting function W_l with the image, each node of L_l can be obtained directly by convolving $W_l - W_{l+1}$ with the image. This difference of Gaussian-like functions resembles the Laplacian operators commonly used in the image processing [5], so we refer to the sequence $L_0, L_1, \ldots, L_N$ as the Laplacian pyramid.[2]

2.3 Summation Property

The steps used to construct the Laplacian pyramid may be reversed to recover the original image G_0 exactly. The top pyramid level, L_N, is first expanded and added to L_{N-1} to recover G_{N-1}; this array is then expanded and added to L_{N-2} to recover G_{N-2}, and so on. Alternatively, we may write

$$G_0 = \sum_{l=0}^{N} L_{l,l}.$$

The expand and sum procedure will be used to construct a mosaic image from its set of splined band-pass components.

2.4 Boundary Conditions

In both the REDUCE and EXPAND operations, special attention must be given to edge nodes. For example, when a REDUCE is performed, the generating kernel for an edge node at level $G_{l=1}$ extends beyond the edge of level G_l by two nodes. Therefore, before the REDUCE (or EXPAND) is performed, G_l is augmented by two rows of nodes on each side. Values are assigned to these nodes by reflection and inversion across the edge node. Thus, if $G_l(0, j)$ is a node on the left edge of G_l, we set

$$G_l(-1, j) = 2G_l(0, j) - G_l(1, j),$$

and

$$G_l(-2, j) = 2G_l(0, j) - G_l(2, j).$$

This treatment of boundaries has the effect of extrapolating the images in such a way that the first derivative is constant at the edge node (the second derivative is zero).

3. THE MULTIRESOLUTION SPLINE

3.1 Splining Overlapped Images

The multiresolution spline algorithm may be defined rather simply in terms of the basic pyramid operations introduced in the last section. Here variations on the method will be described for splining overlapped and nonoverlapped square im-

[2]In fact, the equivalent weighting functions for the Laplacian pyramid are slightly different from $W_l - W_{l-1}$, because of the EXPAND operation used at level $l + 1$.

ages and for splining images of arbitrary shape. Modifications for other tasks will then be apparent. To begin, suppose we wish to spline the left half of image A with the right-half of image B. Assume that these images are both square, measuring $2^N + 1$ pixels on a side, and that they overlap completely. The spline is achieved in three steps:

Step 1. Laplacian pyramids LA and LB are constructed for images A and B respectively.

Step 2. A third Laplacian pyramid LS is constructed by copying nodes from the left half of LA to the corresponding nodes of LS, and nodes in the right half of LB to the right half of LS. Nodes along the center line of LS are set equal to the average of corresponding LA and LB nodes.

The center line for level l of a Laplacian pyramid is at $i = 2^{N-1}$. Thus, for all i, j, l,

$$LS_l(i,j) = \begin{cases} LA_l(i,j) & \text{if} \quad 1 < 2^{N-1} \\ (LA_l(i,j) + LB_l(i,j))/2 & \text{if} \quad i = 2^{N-1} \\ LB_l(i,j) & \text{if} \quad i > 2^{N-1} \end{cases}$$

Step 3. The splined image S is obtained by expanding and summing the levels of LS.

The result of applying this procedure to the star example is shown in Figure 3f. Note that the transition between image halves is now smooth, without the blurred step edge of Figure 3d or the doubling of Figure 3e.

A second example is given in Figure 6. Here we wish to spline two Landsat images of San Francisco, Figures 6a and 6b. These images are identical except for diffuse background noise which has been added to simulate the effects of possible differences in atmospheric conditions or image processing. Again, we wish to construct a mosaic in which the left half of one image is joined to the right half of the other. If this is done without a spline, the boundary is easily visible (Figure 6c). If the multiresolution spline is used, however, the edge is completely removed (Figure 6d).

A third example shows the result of splining two quite different images, an apple and an orange (Figures 7a and 7b). The mosaic obtained without a spline is given in Figure 7c, while that obtained with the spline is shown in Figure 7d. In this case the transition between component images has been made slightly more gradual: in addition to averaging Laplacian nodes along the center line, nodes on either side of the center nodes have been averaged with a 3/4 to 1/4 ratio of weights. The splining process has been repeated separately for the red, green, and blue image color components. Again, a smooth transition is obtained despite the rather large step in gray level between the apple and orange halves.

In this pyramid-based splining procedure, the equivalent weighting functions W_l play a dual role. Within the domain of each image, they act as interpolation functions between level l samples. Along the boundary between the image halves, they act as the splining functions H. If the images to be splined are identical, then the mosaic obtained through the pyramid-based splining will be the same im-

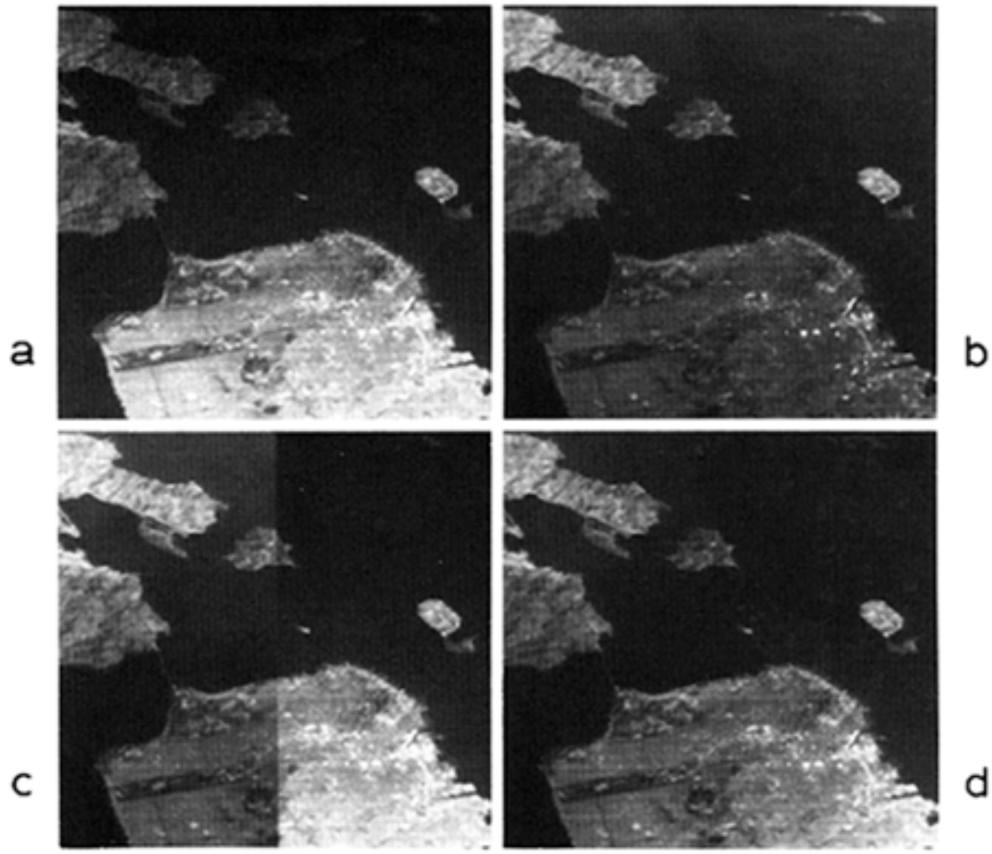


Fig. 6. The spline applied to Landsat images of San Francisco. When the left half of Figure 6a is joined to the right half of Figure 6b without a spline, the boundary is clearly visible (Figure 6c). No boundary is visible when the multiresolution spline is used (Figure 6d).

age again. In this sense, the splining procedure by itself does not introduce image distortion. As shown in Figure 5, W_l extends twice the sample distance 2^l on each side of the level l sample point. This is an appropriate transition distance for splining the frequencies represented in the lth pyramid level.

3.2 Splining Regions of Arbitrary Shape

The steps outlined above can be generalized for constructing a mosaic from image regions of arbitrary shape. Again, we assume that the regions to be splined are contained in images A and B and that these completely overlap. As before, nodes of the Laplacian pyramids LA and LB for the component images will be combined to form the Laplacian pyramid LS of the image mosaic S. We introduce an additional pyramid structure in order to determine which nodes of LS should be taken from LA, which from LB, and which should be an average of the two. Let R be a binary image of the same size as A and B, in which all pixels inside the region of A to be splined with B are 1 and all those outside the region are 0. The steps of the multiresolution spline are modified as follows (on page 230):

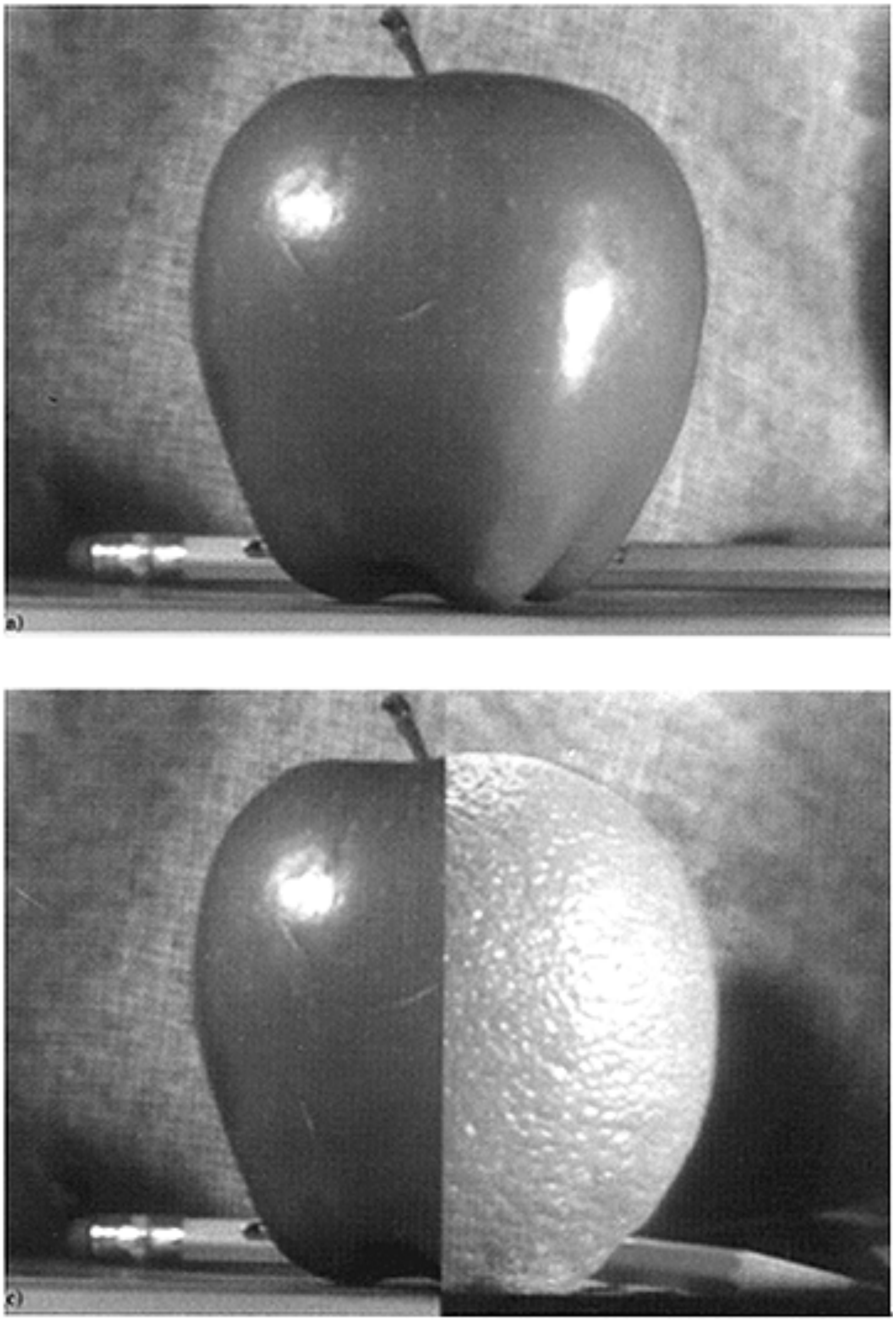
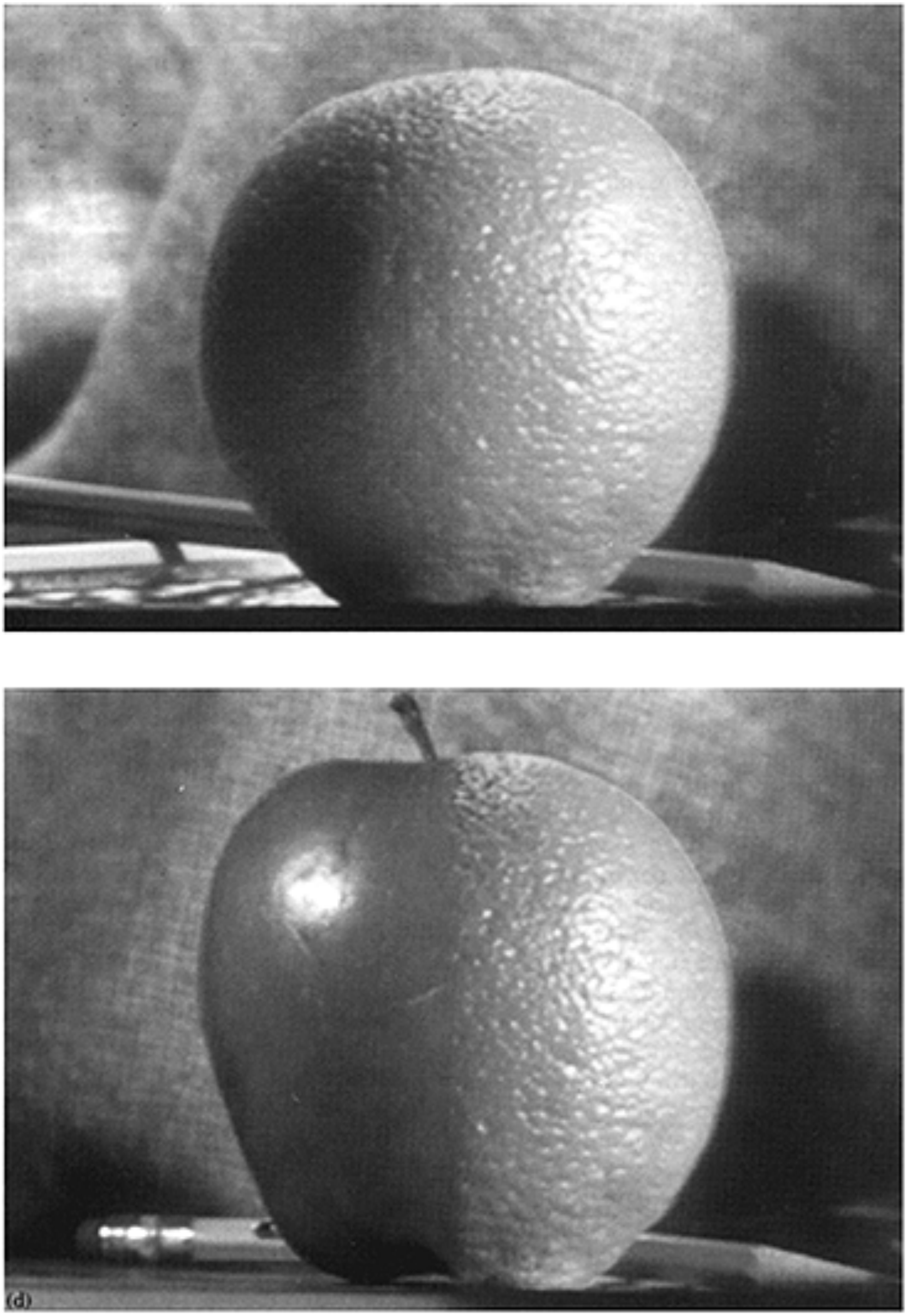


Fig. 7. The spline can be used to combine very different images. Here the left half of an apple (Figure 7a), is combined with the right half of an orange (Figure 7b). Figure 7c, obtained without a

.



spline, shows that the orange and apple differ considerably in gray level and color. Still, a smooth transition is obtained with the multiresolution spline (Figure 7d).

Step la. Build Laplacian pyramids *LA* and *LB* for images *A* and *B* respectively.
Step lb. Build a Gaussian pyramid *GR* for the region image *R*.
Step 2. Form a combined pyramid *LS* from *LA* and *LB* using nodes of *GR* as weights. That is, for each l, i and j:

$$LS_l(i, j) = GR_l(i, j)LA_l(i, j) + (1 - GR_l(i, j))LB_l(i, j).$$

Step 3. Obtain the splined image *S* by expanding and summing the levels of *LS*.

The Gaussian pyramid serves two purposes here: It is a convenient method for determining which nodes at each pyramid level lie within the mask area of image *R*, and it "softens" the edges of the mask through an effective low-pass filter. Without this the spline would be overly sensitive to the position of the mask relative to the pyramid sample points. Nodes which fall exactly on the mask edge will receive a 50 percent weight, just as in the procedure outlined in the previous section. Here nodes to a distance of two sample positions on either side of the mask edge will also be combined as a weighted average of their *LA* and *LB* values (see Figure 5).

An example using this technique is given in Figure 8. Figures 8a and 8b show the two original images, an eye and a hand. Figure 8c shows the region of the first image to be splined into the second image and Figure 8d shows the end result of the spline: a hand with an eye embedded in the palm.

3.3 Splining Nonoverlapped Images

Images must overlap if they are to be joined using any weighted average technique. Nonetheless, a satisfactory spline can be obtained with images that abut but do not overlap, if each image is first extrapolated across its boundary to form an overlapped transition zone. Since the width of the transition zone can be a significant fraction of the width of the image itself, extrapolation may at first seem to be a formidable task. However, in the multiresolution spline technique extrapolation can be performed separately in each frequency band. Furthermore, when the pyramid algorithm is used, only two samples need to be extrapolated beyond the edge of each level. In fact, it is just this type of extrapolation that is already handling boundary conditions during construction of both the Gaussian and Laplacian pyramids. No further steps need to be taken in the spline.

An application of splining to nonoverlapping images is shown in Figure 9. We begin with a single image which is itself a mosaic of 16 by 16 pixel blocks (only the central 8×8 array of blocks is shown in the figure). Each block has been reconstructed from a highly compact transform code, which, in this case, represents the image at a rate of only 0.5 bits per pixel (see for example, [8]). Block transform coding at a very low bit rate produces prominent block boundaries in addition to other severe image degradation. Our task is to remove the boundaries by means of a multiresolution spline.

The original image contains a 16×16 array of blocks. Before attempting the spline, we use extrapolation to add a row on the right and bottom sides of each block. The resulting 17×17 pixel blocks fit into the Laplacian pyramid structure and overlap in the image by one pixel on each side. The first step of the spline

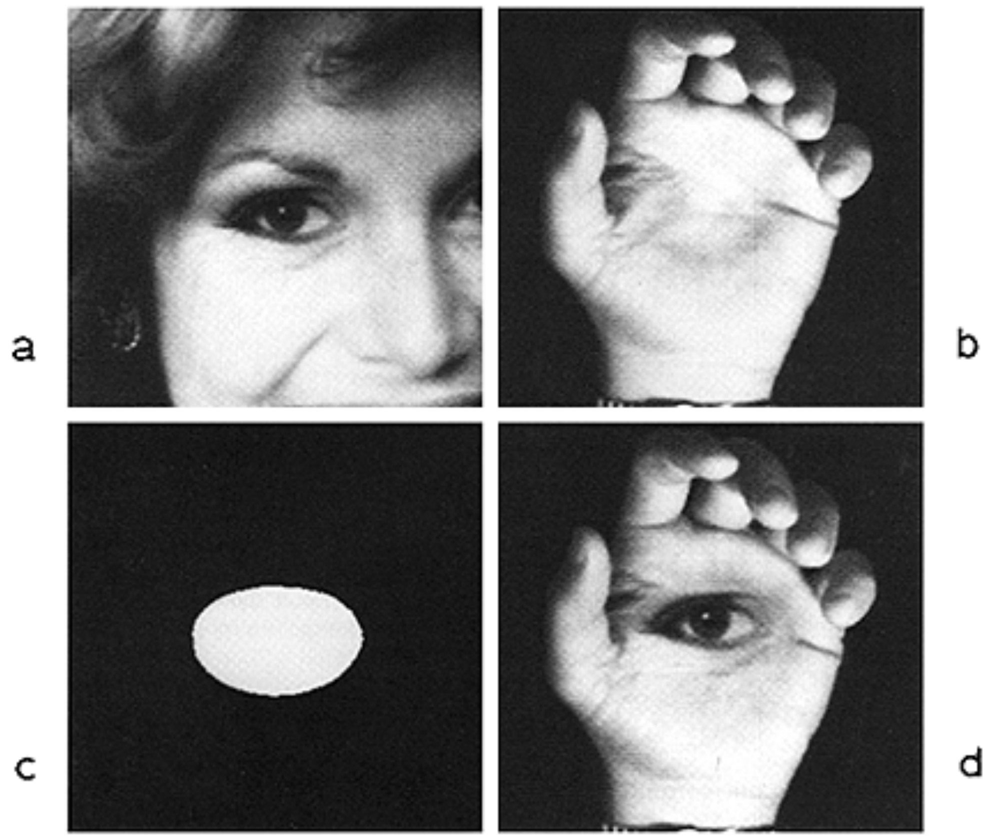


Fig. 8. The spline may be used to combine oddly shaped regions of very different images. The portion of Figure 8a within the region indicated by the mask in Figure 8c is inserted in the portion of Figure 8b which is outside this mask region (Figure 8d).

procedure is to construct a separate Laplacian pyramid for each of these 256 blocks. These pyramids are then joined into a single pyramid with nodes in the overlapped edges of each pyramid level being averaged.

Let L_{lmn} be the lth level of the Laplacian pyramid constructed for the nth block in the mth row of blocks. Then, for i and j not on block boundaries (i.e., i and j are not equal to a multiple of 2^{4-l}) and level $0 < l < 3$,

$$LS_l(i,j) = L_{lmn}(\hat{i},\hat{j}),$$

where m is the integer part of $i/2^{4-l}$, n is the integer part of $j/2^{4-l}$, $\hat{i} = i - m2^{4-l}$, and $\hat{j} = j - 2^{4-l}$.

If i or j is on a block boundary above level zero, $0 < l < 3$, then the LS node will be an average,

$$LS_l(i,j) = \frac{L_{l,m-1,n}(16,\hat{j}) + L_{l,m,n}(0,\hat{j})}{2}.$$

Fig. 9. The multiresolution spline used to remove block boundaries from a block transform encoded image. Figure 9a shows an image which has been block transform encoded at a rate of 0.5 bits per pixel. An 8 x 8 array of blocks is shown, each containing a 16 × 16 array of pixels. Blocks were treated as separate, nonoverlapped images to obtain the splined result shown in Figure 9b. Image quality remains low because of the very low bit rate of the code, but the perceptually prominent block boundaries are almost completely eliminated.

This average is not computed for boundary nodes in the bottom level because the node $L_{l,m-1,n}(16, j)$ represents an extrapolated value. Instead, we simply say

$$LS_0(i, j) = L_{0,m,n}(0, \hat{j}).$$

Reconstruction through the expand and sum process yields the image shown in Figure 9b. Note that the block boundaries have been almost completely removed. The image is still of low quality, but this is due to the very low bit rate of the original block encoded image rather than to the splining technique.

4. SUMMARY AND DISCUSSION

We have described a multiresolution spline technique for combining two or more images into a larger image mosaic. In this procedure, the images to be splined are first decomposed into a set of band-pass filtered component images Next, the component images in each spatial frequency band are assembled into a corresponding band-pass mosaic. In this step, component images are joined using a weighted average within a transition zone which is proportional in size to the wave lengths represented in the band. Finally, these band-pass mosaic images are summed to obtain the desired image mosaic. In this way, the spline is matched to the scale of features within the images themselves. When coarse features occur near borders, these are blended gradually over a relatively large distance without blurring or otherwise degrading finer image details in the neighborhood of the border.

The basic steps of the multiresolution spline are illustrated in Figure 10. In this case, the left half of an apple (Figure 7a) is splined with the right half of an orange

(Figure 7b). The first column of images in Figure 10 (10a-10d) shows high, medium and low frequency components of the half apple. Note that the high frequency components extend only slightly to the right of the midline, while the low frequencies extend considerably further. If these images are summed (along with a number of other components which are not shown), the half apple at the bottom is obtained (Figure 10d).

Figures in the center column (10e-10h) show the corresponding components for the orange. Figures in the right hand column (10i-10l) are obtained by summing the orange and apple components in each spatial frequency band. All computations are linear. Thus, the final mosaic, Figure 10l, can be obtained by summing the half apple of Figure 10d with the half orange of Figure 10h, or by summing the composite band-pass images in the right hand column (10i-10j) (along with other components not shown).

We have demonstrated the multiresolution spline with a variety of image mosaic examples. In all cases, it has eliminated visible seams between component images. This is true even when the component images are very different (e.g., the orange-apple), or of irregular shape (e.g., the hand-eye). The multiresolution approach avoids artifacts such as the blurred edge and double exposure effect obtained with a simple (single resolution) weighted average, as shown in Figures 3d and 3e.

In the implementation described here, pyramid algorithms have been used both for filtering and splining operations. The pyramid structure is uniquely suited to the present task. It is a highly efficient filter, requiring only seven arithmetic operations (adds and multiplies) per image pixel to produce a full set of low-pass images [2]. Furthermore, the weighting functions H used in each spline are implicit in the pyramid computation: they need never be specified explicitly, yet they are matched to each frequency band represented in the pyramid. In a similar way, the image extrapolation required in splining nonoverlapped images is provided as a boundary condition in the standard pyramid construction algorithm.

In sum, the multiresolution spline appears to be a practical and quite general technique for forming image mosaics. The pyramid in turn offers a unifying structure in which required filtering and splining steps may be performed both easily and efficiently.

ACKNOWLEDGMENTS

We wish to thank Professor William Pearlman for providing the image used in Figure 9a, and Joan Ogden and James Bergen for their assistance in producing the apple-orange picture (Figures 7 and 10).

REFERENCES

1. BURT, P.J. Fast filter transforms for image processing. *Comput. Gr. Image Process. 16* (1981), 20-51.
2. BURT, P.J. Fast algorithms for estimating local image properties. *Comput. Vision Gr. Image Process. 21* (1983), 368-382.
3. BURT, P.J. AND ADELSON, E.H. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun. COM-31*, (1983), 532-540.

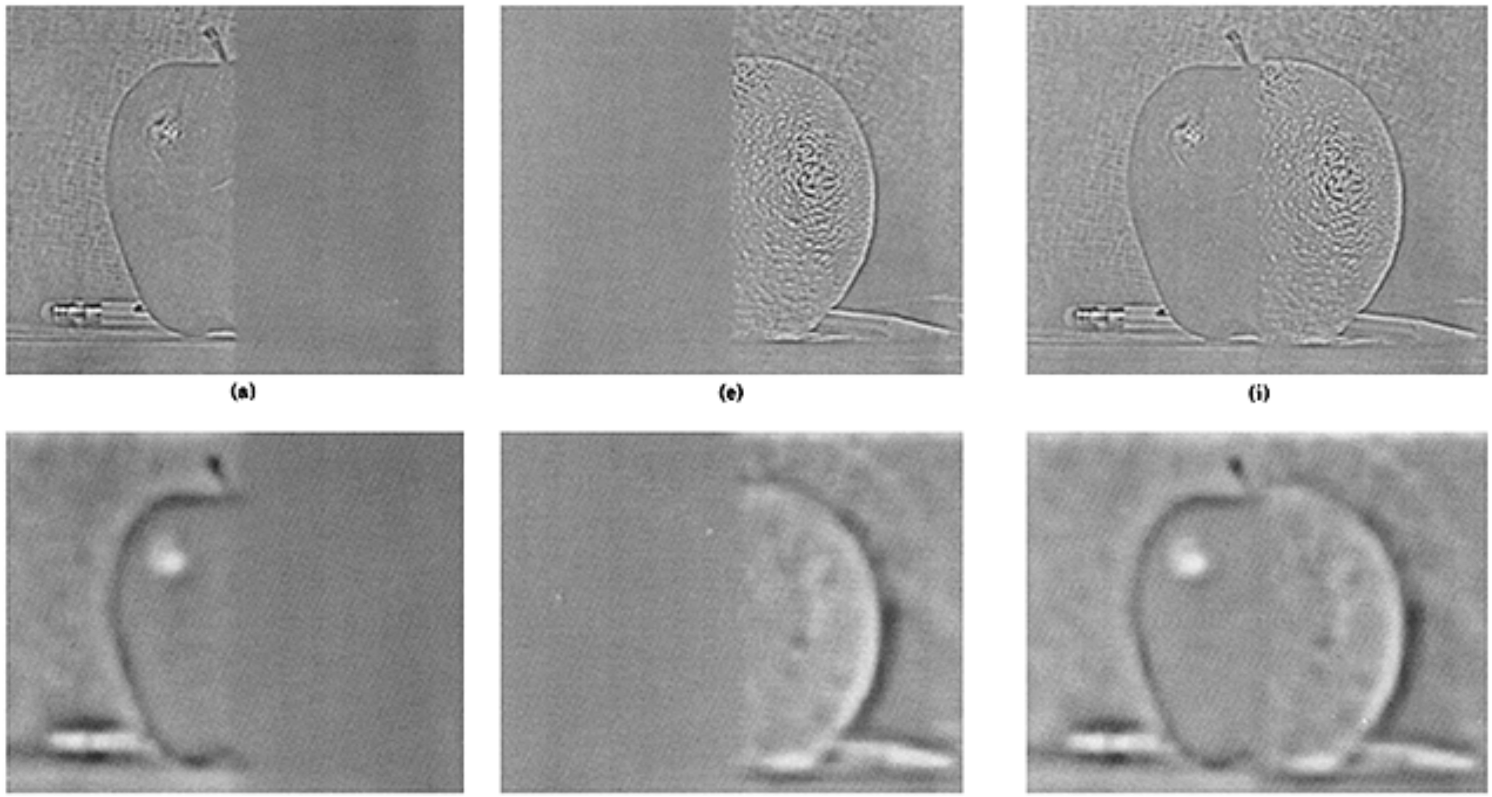
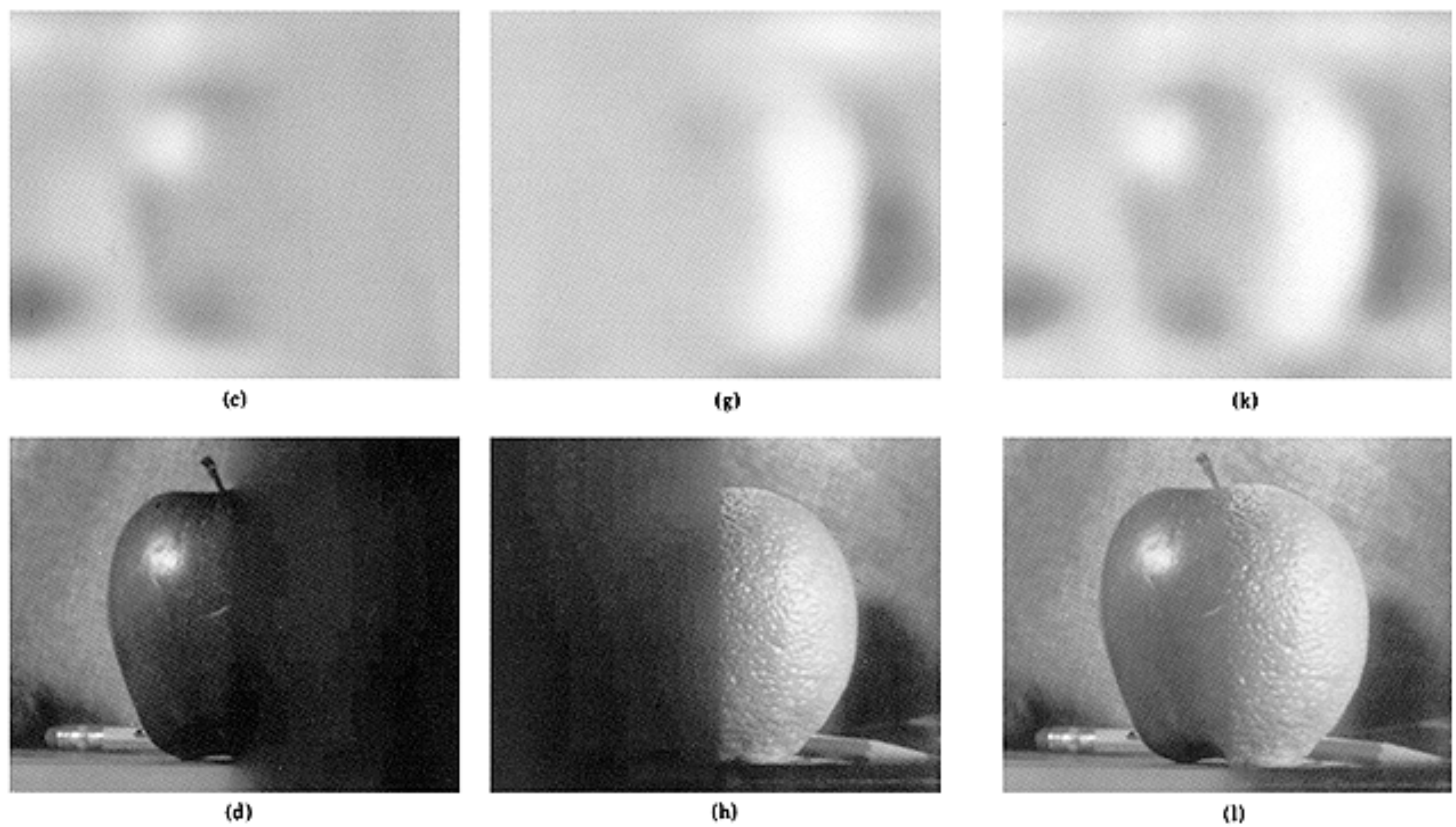


Fig. 10. Contributions of various band-pass filtered images to the apple-orange mosaic. Figures 10a-10c show high-, medium-, and low-frequency componentsof the orange obtained by expanding levels0, 2, and 4 of its Laplacian pyramid. (In these figures, a constant has been added to each band-pass imageso that zerovalues appear gray and negative values are dark. Image contrast has also been increased to improve visibility.) Note that low-frequency components"bleed" across the center line further than do high-frequency components. The half apple in Figure 10d was obtained by summing Figures 10a-10c,



along with the 5 other band-pass components in its pyramid representation. The corresponding components of the orange are shown in Figures 10e-10h. Figures 10i-10kshow the band-pass mosaics obtained by summing the apple and orange components in each frequency band. The final mosaic, Figure 10l, can be obtained eitherby summing its band pass components (Figures 10i-10k), or by summing the half orange and apple, Figures 10d and 10h.

4. DUNNE, J.A. and BURGESS, E. The voyage of Mariner 10. National Aeronautics and Space Administration SP 424 (1978).
5. MARR, D. and HILDRETH, E. Theory of edge detection. In *Proceedings of the Royal Society B-207* (London, 1980), 187-217.
6. MILGRAM, D.L. Computer methods for creating photomosaics. *IEEE Trans. Comput. C-24*, (1975), 1113-1119.
7. MOIK, J.G. Digital processing of remotely sensed images. National Aeronautics and Space Administration SP 431 (1980).
8. NETRAVALI, A.N. and LINB, J.O. Picture coding, a review. *Proc. IEEE 68*, (1980), 336-406.
9. PELEG, S. Elimination of seams from photomosaics. In *Proceedings of the Conference on Pattern Recognition and Image Processing*. (Dallas, Tex., Aug. 3-5 1981), pp. 426-429.
10. TANIMOTO, S.L. and PAVLIDIS, T. A hierachical data structure for picture processing, *Comput. Gr. Image Process. 4*, (1975), 104-119.

Received June 1983; revised January 1984; accepted January 1984