5

Processing Images in Frequency Space

Some necessary mathematical preliminaries

What frequency space is all about

It is unusual to pick up a book on image analysis without finding at least a portion of it devoted to a discussion of Fourier transforms (see especially Pratt, 1991; Gonzalez and Wintz, 1987; Jain, 1989). In part, this is due to the utility of working in frequency space to perform certain image measurement and processing operations. Many of these same operations can be performed in the original (spatial or pixel domain) image only with significantly greater computational effort. Another reason for the lengthy sections on these methods is that the authors frequently come from a background in electrical engineering and signal processing and so are familiar with the mathematics and the use of these methods for other types of signals, particularly the one-dimensional (time varying) electrical signals that make up much of our modern electronics.

The typical image analyst interested in applying computer methods to images for purposes of enhancement or measurement, however, may not be comfortable with the pages of mathematics (and intimidating notation) used in these discussions. Furthermore, he or she may not have the fortitude to relate these concepts to the operation of a dedicated image analysis computer. Unable to see the connection between the topics discussed and the typical image problems encountered in real life, the potential user might therefore find it easier to skip the subject. This is a loss, because the use of frequency space methods can offer benefits in many real-life applications, and it is not essential to deal deeply with the mathematics to arrive at a practical working knowledge of these techniques.

The Fourier transform and other frequency space transforms are applied to two-dimensional images for many different reasons. Some of these have little to do with the purposes of enhancing visibility and selection of features or structures of interest for measurement. For instance, some of these transform methods are used as a means of image compression, in which less data than the original image may be transmitted or stored. In this type of application, it is necessary to reconstruct the image (bring it back from the frequency to the spatial domain) for viewing. It is desirable to be able to accomplish both the forward and reverse transform rapidly and with a minimum loss of image quality. Image quality is a somewhat elusive concept that certainly includes the alteration of grey levels and color values, definition and location of feature boundaries, and introduction or removal of fine-scale texture in the image. Usually, the greater the degree of compression, the greater the loss of image fidelity, as shown in Chapter 2.

Speed is usually a less important concern to image measurement applications, since the acquisition and subsequent analysis of the images are likely to require some time anyway, but the computational advances (both in hardware and software or algorithms) made to accommodate the requirements of the data compression application help to shorten the time for some other processing operations, as well. On the other hand, the amount of image degradation that can be tolerated by most visual uses of the compressed and restored images is far greater than is usually acceptable for image analysis purposes. Consequently, the amount of image compression that can be achieved with minimal loss of fidelity is rather small, as discussed in Chapter 2.

In most cases, the transmission of an image from the point of acquisition to the computer used for analysis is not a major concern; therefore, we will ignore this entire subject here and assume that the transform retains all of the data, even if this means that there is no compression at all. Indeed, most of these methods are free from any data loss. The transform encodes the image information completely and it can be exactly reconstructed, at least to within the arithmetic precision of the computer being used (which is generally better than the precision of the original image sensor or analog-to-digital converter).

Although many different types of image transforms can be used, the best known (at least, the one with the most recognizable name) is the Fourier transform. This is due in part to the availability of a powerful and very efficient algorithm for computing it, known as the fast Fourier transform (FFT) (Cooley and Tukey, 1965; Bracewell, 1989), which we will encounter in due course. Although some programs actually perform the computation using the Fast Hartley Transform or FHT (Hartley, 1942; Bracewell, 1984, 1986; Reeves, 1990), the frequency space images are usually presented in the same form that the Fourier method would yield. For the sake of explanation, it is easiest to describe the better-known method.

The usual approach to developing the mathematical background of the Fourier transform begins with a one-dimensional waveform and then expands to two dimensions (an image). In principle, this can also be extended to three dimensions, although it becomes much more difficult to visualize or display. Three-dimensional transforms between the spatial domain (now a volume image constructed of voxels instead of pixels) and the three-dimensional frequency space are used, for example, in some tomographic reconstructions.

The mathematical development that follows has been kept as brief as possible, but if you suffer from "integral-o-phobia" then it is permitted to skip this section and go on to the examples and discussion, returning here only when (and if) a deeper understanding is desired.

The Fourier transform

Using a fairly standard nomenclature and symbology, begin with a function $f(x)$, where x is a real variable representing either time or distance in one direction across an image. It is very common to refer to this function as the spatial or time domain function and the transform F introduced below as the frequency space function. The function f is a continuous and well-behaved function. Do not be disturbed by the fact that in a digitized image, the values of x are not continuous but discrete (based on pixel spacing), and the possible brightness values are quantized as well. These values are considered to sample the real or analog image that exists outside the computer.

Fourier's theorem states that it is possible to form any one-dimensional function $f(x)$ as a summation of a series of sine and cosine terms of increasing frequency. The Fourier transform of the function $f(x)$ is written $F(u)$ and describes the amount of each frequency term that must be added together to make $f(x)$. It can be written as

$$F(u) = \int_{-\infty}^{+\infty} f(x) e^{-2\pi i u x} dx \tag{1}$$

where i is (as usual) $\sqrt{-1}$. The use of the exponential notation relies on the mathematical identity (Euler's formula)

$$e^{-2\pi i u x} = \cos(2\pi u x) - i \sin(2\pi u x) \tag{2}$$

One of the very important characteristics of this transform is that given $F(u)$, it is possible to recover the spatial domain function $f(x)$ in the same way.

$$f(x) = \int_{-\infty}^{+\infty} F(u) e^{2\pi i u x} du \tag{3}$$

These two equations together comprise the forward and reverse Fourier transform. The function $f(x)$ is generally a real function, such as a time-varying voltage or a spatially-varying image brightness; however, the transform function $F(u)$ is generally complex, the sum of a real part R and an imaginary part I.

$$F(u) = R(u) + iI(u) \tag{4}$$

It is usually more convenient to express this in polar rather than Cartesian form

$$F(u) = |F(u)| \cdot e^{i\Phi(u)} \tag{5}$$

where $|F|$ is called the magnitude and ϕ is called the phase. The square of the magnitude $|F(u)|^2$ is commonly referred to as the power spectrum, or spectral density of $f(x)$.

The integrals from minus to plus infinity will in practice be reduced to a summation of terms of increasing frequency, limited by the finite spacing of the sampled points in the image. The discrete Fourier transform is written as

$$F(u) = \tfrac{1}{N} \sum_{x=0}^{N-1} f(x) \cdot e^{-i 2\pi u x / N} \tag{6}$$

where N depends on the number of sampled points along the function $f(x)$, which are assumed to be uniformly spaced. Again, the reverse transform is similar (but not identical; note the absence of the $1/N$ term and the change in sign for the exponent).

$$f(x) = \sum_{u=0}^{N-1} F(u) \cdot e^{i 2\pi u x / N} \tag{7}$$

The values of u from 0 to N–1 represent the discrete frequency components added together to construct the function $f(x)$. As in the continuous case, $F(u)$ is complex and may be written as real and imaginary or as magnitude and phase components.

The summation is normally performed over terms up to one-half the dimension of the image (in pixels), since it requires a minimum of two pixel brightness values to define the highest frequency present. This limitation is described as the Nyquist frequency. Because the summation has half as many terms as the width of the original image, but each term has a real and imaginary part, the

total number of values produced by the Fourier transform is the same as the number of pixels in the original image width, or the number of samples of a time-varying function. Since the original pixel values are usually small integers (0..255 for an 8-bit image), while the values produced by the Fourier transform are floating point numbers (and double precision ones in the best implementations), this actually represents an expansion in the storage requirements for the data.

In both the continuous and the discrete cases, a direct extension from one-dimensional functions to two- (or three-) dimensional ones can be made by substituting $f(x,y)$ for $f(x)$ and $F(u,v)$ for $F(u)$, and performing the summation or integration over two (or three) variables instead of one. Because the dimensions x,y,z are orthogonal, so are the u,v,w dimensions. This means that the transformation can be performed separately in each direction. For a two-dimensional image, for example, it would be possible to perform a one-dimensional transform on each horizontal line of the image, producing an intermediate result with complex values for each point. Then a second series of one-dimensional transforms can be performed on each vertical column, finally producing the desired two-dimensional transform.

The program fragment listed below shows how to compute the FFT of a function. It is written in Fortran, but can be translated into any other language (you may have to define a type to hold the complex numbers). On input to the subroutine, *F* is the array of values to be transformed (usually the imaginary part of these complex numbers will be 0) and *LN* is the power of 2 (up to 10 for the maximum 1024 in this implementation). The transform is returned in the same array *F*. The first loop reorders the input data, the second performs the successive doubling that is the heart of the FFT method, and the final loop normalizes the results.

```
      SUBROUTINE FFT(F,LN)
      COMPLEX F(1024),U,W,T,CMPLX
      PI=3.14159265
      N=2**LN
      NV2=N/2
      NM1=N-1
      J=1
      DO 3 I=1,NM1
            IF (I.GE.J) GOTO 1
            T=F(J)
            F(J)=F(I)
            F(I)=T
1           K=NV2
2           IF (K.GE.J) GOTO 3
            J=J-K
            K=K/2
            GOTO 2
3           J=J+K
      DO 5 L=1,LN
            LE=2**L
            LE1=LE/2
            U=(1.0,0.0)
            W=CMPLX(COS(PI/LE1),-SIN(PI/LE1))
            DO 5 J=1,LE1
                  DO 4 I=J,N,LE
                        IP=I+LE1
                        T=F(IP)*U
                        F(IP)=F(I)-T
```

```
4                          F(I)=F(I)+T
5                    U=U*W
      DO 6 I=1MN
6              F(I)=F(I)/FLOAT(N)
      RETURN
      END
```

Applying this one-dimensional transform to each row and then each column of a two-dimensional image is not the fastest way to perform the calculation, but it is by far the simplest and is actually used in many programs. A somewhat faster approach, known as a butterfly because it uses various sets of pairs of pixel values throughout the two-dimensional image, produces identical results. Storing the array of W values can also provide a slight increase in speed. Many software math packages include highly optimized FFT routines. Some of these allow for array sizes that are not an exact power of two.

The resulting transform of the original image into frequency space has complex values at each pixel. This is difficult to display in any readily interpretable way. In most cases, the display is based on only the magnitude of the value, ignoring the phase. If the square of the magnitude is used, this may be referred to as the image's power spectrum, because different frequencies are represented at different distances from the origin, different directions represent different orientations in the original image, and the power at each location shows how much of that frequency and orientation is present in the image. This display is particularly useful for isolating periodic structures or noise, which is discussed below; however, the power spectrum by itself cannot be used to restore the original image. The phase information is also needed, although it is rarely displayed and is usually difficult or impossible to interpret visually.

Fourier transforms of real functions

A common illustration in introductory-level math textbooks on the Fourier transform (which usually deal only with the one-dimensional case) is the quality of the fit to an arbitrary, but simple, function by the sum of a finite series of terms in the Fourier expansion. **Figure 1** shows the familiar case of a step function, illustrating the ability to add up a series of sine waves to produce the desired step. The coefficients in the Fourier series are the magnitudes of each increasing frequency needed to produce the fit. **Figure 2a** shows the result of adding together the first 4, 10, and 25 terms. Obviously, the greater the number of terms included, the better the fit (especially at the sharp edge). **Figure 2b** shows the same comparison for a ramp function. One of the important characteristics of the Fourier transform is that the first few terms include much of the information and adding more terms progressively improves the quality of the fit.

Notice in both of these cases that the function is actually assumed to be repetitive or cyclical. The fit goes on past the right and left ends of the interval as though the function were endlessly repeated in both directions. This is also the case in two dimensions; the image in the spatial domain is effectively one tile in an endlessly repeating pattern. If the right and left edges or the top and bottom edges of the image are different, this can produce very noticeable effects in the resulting transform. One solution is to embed the image in a larger one consisting of either zeroes or the average brightness value of the pixels. This kind of padding makes the image twice as large in each direction, requiring four times as much storage and calculation. It is needed particularly when correlation is performed, as discussed below. Padding out to a larger size is also the simplest way to deal with an image that does not have dimensions that are an exact power of two.

The magnitude of the Fourier coefficients from the transforms shown in **Figure 2** is plotted as amplitude vs. frequency (**Figure 3**). Notice that the step function consists only of odd terms, while the magnitudes for the ramp function transform decrease smoothly. Rather than the magnitudes, it

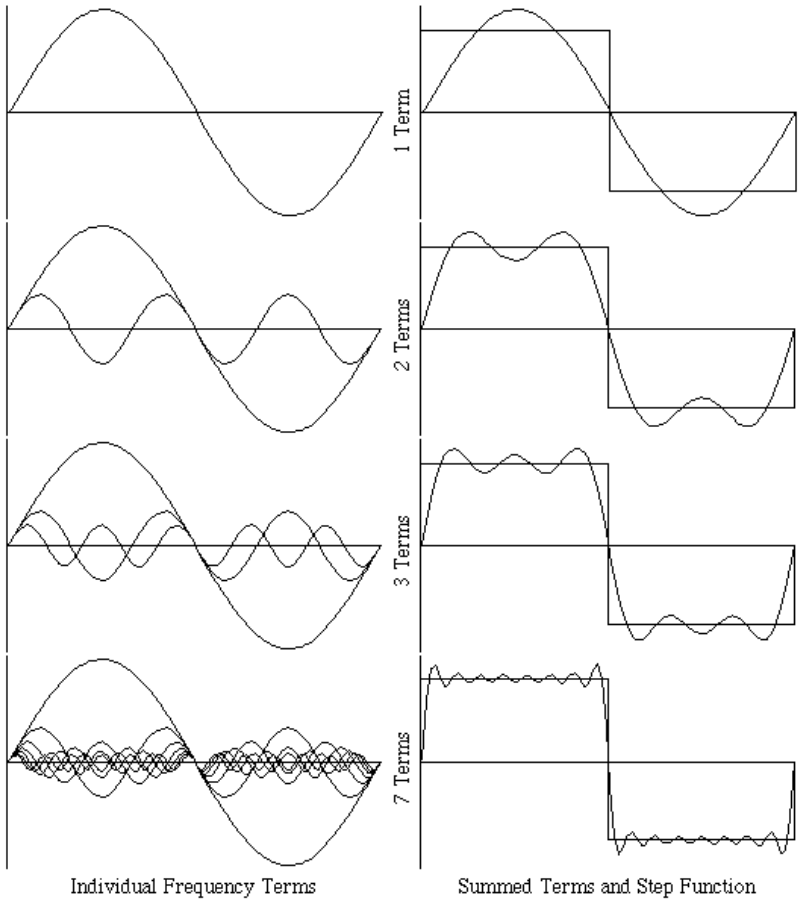
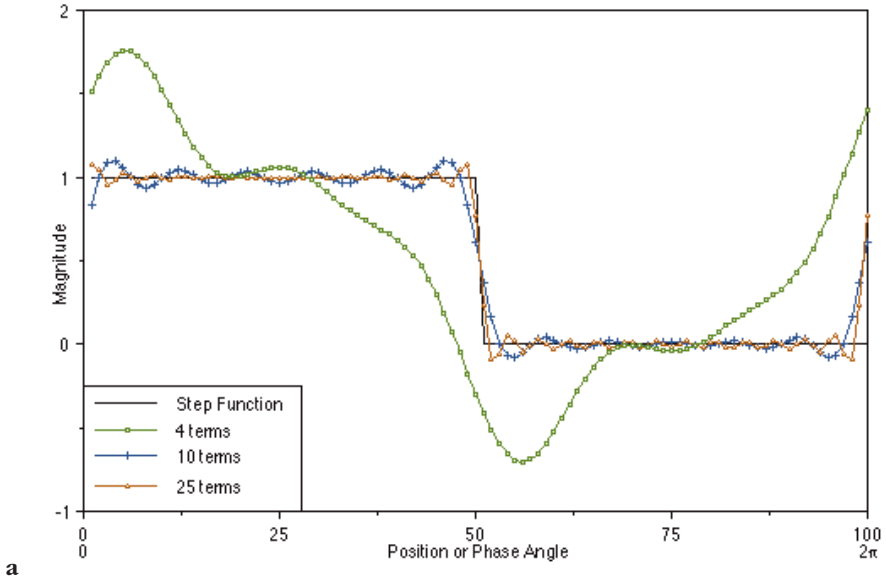


Figure 1. *Summation of Fourier frequency terms to fit a simple step function.*

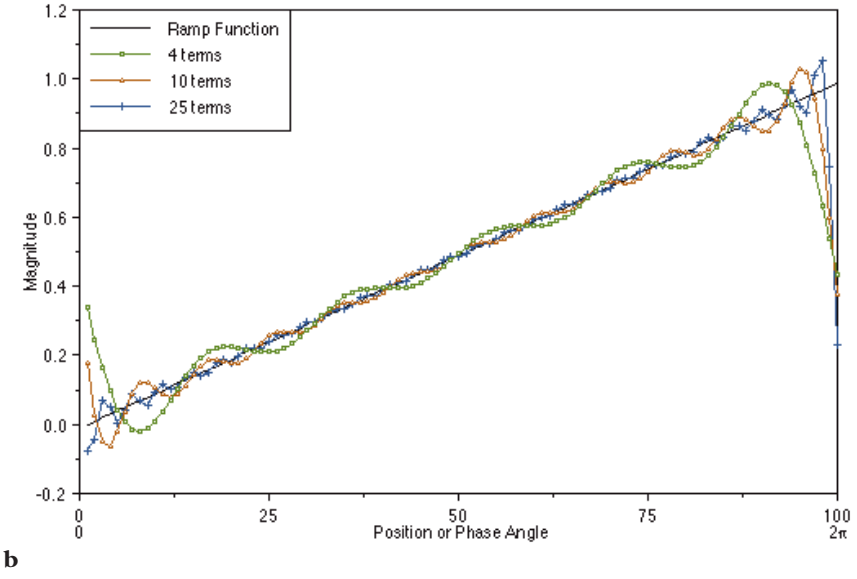
is somewhat more common to plot the power spectrum of the transform, and to plot it as a symmetric function extending to both sides of the origin (zero frequency, or the DC level). As noted above, the power is simply the square of the magnitude. Because the range of values can be very large, the power spectrum is often plotted with a logarithmic or other compressed vertical scale to show the smaller terms usually present at high frequencies, along with the lower frequency terms.

Figure 4 reiterates the duality of the Fourier transform process. The spatial and frequency domains show the information in very different ways, but the information is the same. Of course, the plot of amplitude or power in the frequency transform does not show the important phase information, but we understand that the values are actually complex. Shifting the spatial domain image does not alter the amplitude values, but does change the phase values for each sinusoidal component.

It is important to recall, in examining these transforms, that the axes represent frequency. The low-frequency terms provide the overall shape of the function, while the high-frequency terms are needed to sharpen edges and provide fine detail. The second point to be kept in mind is that these terms are independent of each other (this is equivalent to the statement that the basis functions — the sinusoidal waves — are orthogonal). Performing the transform to determine coefficients to higher and higher frequencies does not change the previous ones, and selecting any particular range of terms to reconstruct the function will do so to the greatest accuracy possible with those frequencies.



a



b

Figure 2. *Match between a step function* ***(a)*** *and a ramp function* ***(b)*** *and the first 4, 10, and 25 Fourier terms.*

Proceeding to two dimensions, **Figure 5** shows four images of perfectly sinusoidal variations in brightness. The first three vary in spacing (frequency) and orientation; the fourth is the superposition of all three. For each, the two-dimensional frequency transform is particularly simple. Each of the pure tones has a transform consisting of a single point (identifying the frequency and orientation). Because of the redundancy of the plotting coordinates, the point is shown in two symmetrical locations around the origin, which by convention lies at the center of the power spectrum plot.

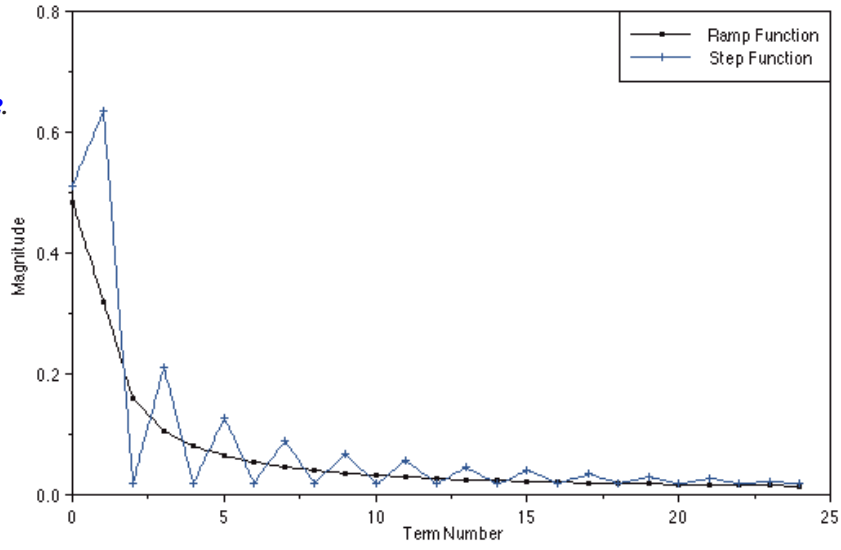


Figure 3. *Magnitude of the first 25 Fourier terms fit to the step and ramp in* ***Figure 2****.*

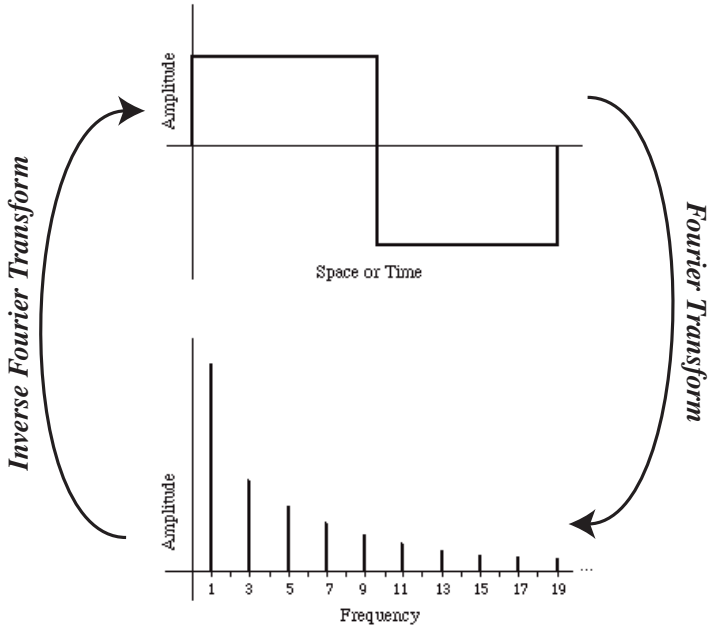


Figure 4. *Role of the forward and inverse transform and the spatial and frequency domain representations of a step function.*

Two-dimensional power spectra are easiest to describe using polar coordinates. The frequency increases with radius ρ, and the orientation depends on the angle θ. It is common to display the two-dimensional transform with the frequencies plotted from the center of the image, which is consequently redundant (the top and bottom or left and right halves are simply duplicates, with symmetry about the origin). In some cases, this image is shifted so that the origin is at the corners of the image and the highest frequencies are in the center. One format can be converted to the other by swapping quadrants of the display. For the purposes of image processing (removing or selecting specific frequencies, etc.) the display with the origin centered is simplest to use and has been adopted here.

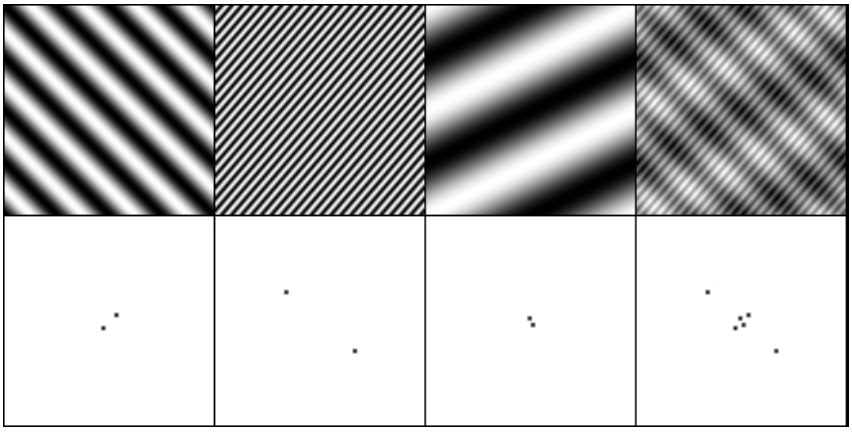
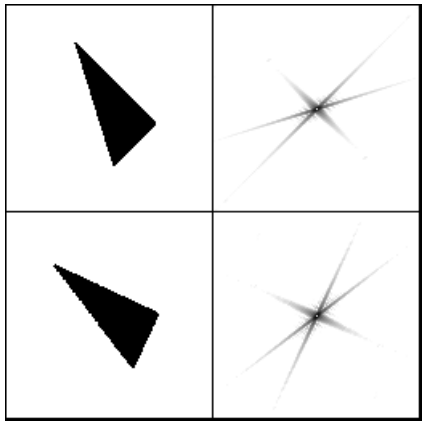


Figure 5. *Three sinusoidal patterns, their frequency transforms, and their sum.*

The power spectrum of the low frequency (greatest spacing) sinusoid has a point close to the origin, and the higher frequency sinusoids have points farther away, and in directions that identify the orientation of the lines. The superposition of the three sinusoids produces an image whose frequency transform is simply the sum of the three individual transforms. This principle of additivity will be important for much of the discussion in the following paragraphs. Subtracting the information from a location in the frequency transform is equivalent to removing the corresponding information from every part of the spatial-domain image. **Figure 6** shows two images with the same shape in different orientations. The frequency transforms rotate with the feature.

Figure 7 shows a two-dimensional step consisting of a rectangle. The two-dimensional frequency transform of this image produces the same series of diminishing peaks in the x and y axis directions as the one-dimensional step function. The darkness of each point in the power spectrum display represents the log of the square of the amplitude of the corresponding frequency. Limiting the reconstruction to only the central (low-frequency) terms produces the reconstructions shown. Just as for the one-dimensional case, this limits the sharpness of the edge of the step and produces some ringing (oscillations near the edge) in the overall shape. The line profiles through the image show the same shape as previously discussed for the one-dimensional case.

Figure 6. *Rotation of a spatial domain image (left), and the corresponding rotation of the frequency transform (right).*



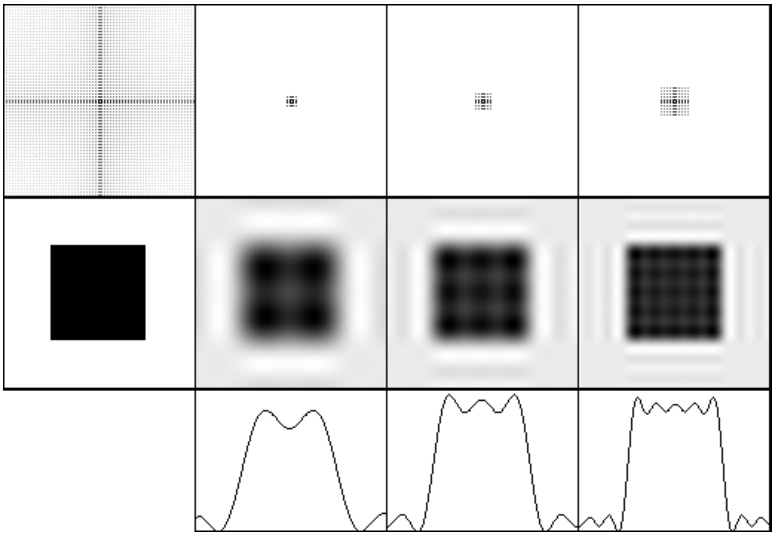


Figure 7. *A two-dimensional step function and its frequency transform (left), and reconstructions with different numbers of terms (shown as a portion of the frequency transform). Bottom row shows horizontal line profiles through the center of the reconstructed spatial image.*

Frequencies and orientations

It is helpful to develop a little familiarity with the power spectrum display of the frequency-space transform of the image using simple images. In **Figure 8a**, the lines can be represented by a single peak because their brightness profile is perfectly sinusoidal; thus only a single frequency is present. If the line profile is different, more terms are needed to represent the shape and consequently more peaks appear in the power spectrum. **Figure 8b** shows an example in which the frequency transform consists of a series of peaks at multiples of the lowest frequency, in the same orientation (perpendicular to the line angle).

In **Figure 8c,** the lines have the aliasing common in computer displays (and in halftone printing technology), in which the lines at a shallow angle on the display are constructed from a series of steps corresponding to the rows of display pixels. This further complicates the frequency transform, which now has additional peaks representing the horizontal and vertical steps in the image that correspond to the aliasing, in addition to the main line of peaks seen in the figure.

It is possible to select only the peaks along the main row and eliminate the others with a mask or filter, as will be discussed in the next section. After all, the frequency-domain image can be modified just like any other image. If this is done and only the peaks in the main row are used for the inverse transformation (back to the spatial domain), the aliasing of the lines is removed. In fact, that is how the images in **Figures 8a** and **8b** were produced. This will lead naturally to the subject of filtering (discussed in a later section): removing unwanted information from spatial-domain images by operating on the frequency transform. For example, it offers one practical technique to remove aliasing from lines and make them visually smooth.

Measuring images in the frequency domain

Orientation and spacing

The idealized examples shown in the preceding tutorial show that any periodic structure in the original spatial-domain image will be represented by a peak in the power spectrum image at a radius corresponding to the spacing and a direction corresponding to the orientation. In a real image, which typically consists of mostly non-periodic information, any such peaks will be superimposed on a broad and sometimes noisy background, although, finding the peaks is generally much

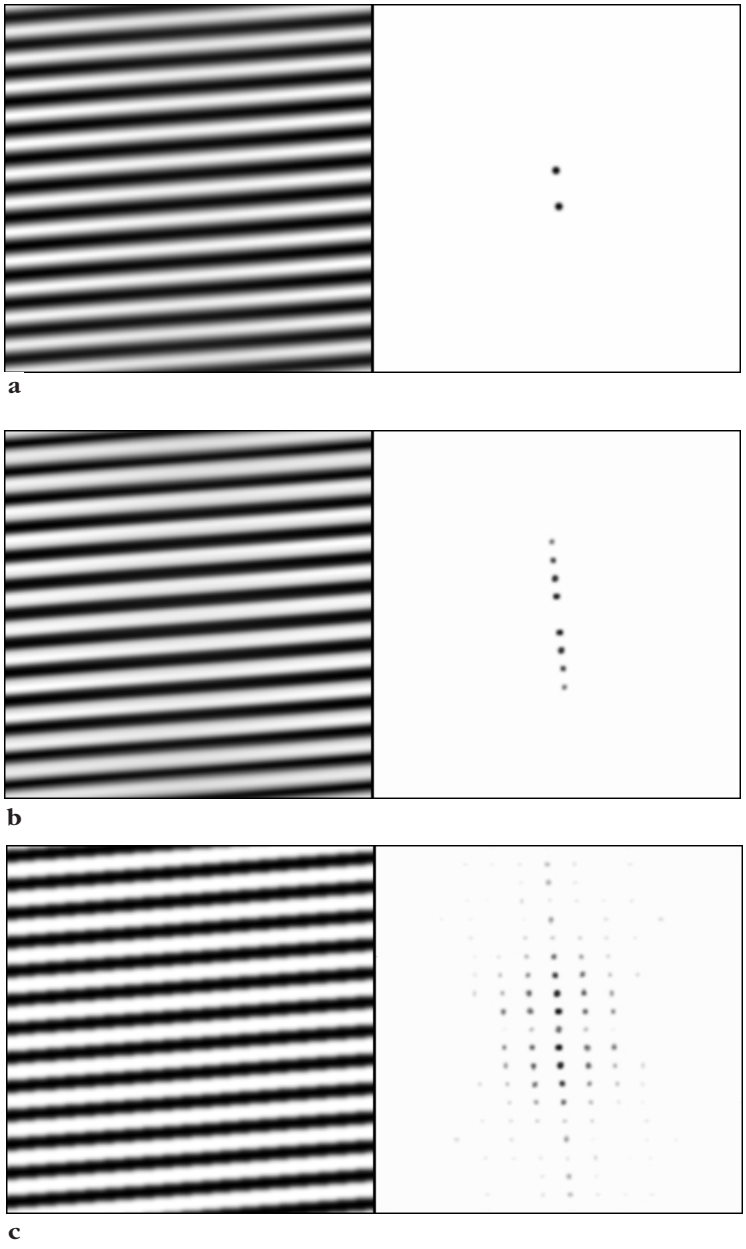


Figure 8. A set of lines (left) and their frequency transform (right): ***(a)*** *sinusoidal lines;* ***(b)*** *with a non-sinusoidal brightness profile;* ***(c)*** *the lines from* ***b*** *with aliasing.*

easier than finding the original periodic structure. Also, measuring the peak locations accurately is much easier and more accurate than trying to extract the same information from the original image, because all of the occurrences are effectively averaged together in the frequency domain.

Figure 9 shows an example of this kind of peak location measurement. The spatial-domain image is a high-resolution transmission electron microscope (TEM) image of the lattice structure in pure silicon. The regular spacing of the bright spots represents the atomic structure of the lattice. Measuring all of the individual spacings of the spots would be very time-consuming and not particularly accurate. The frequency-domain representation of this image shows the periodicity clearly. The series of peaks indicates that the variation of brightness is not a simple sine wave, but contains many higher harmonics. The first-order peak gives the basic atomic spacing (and orientation),

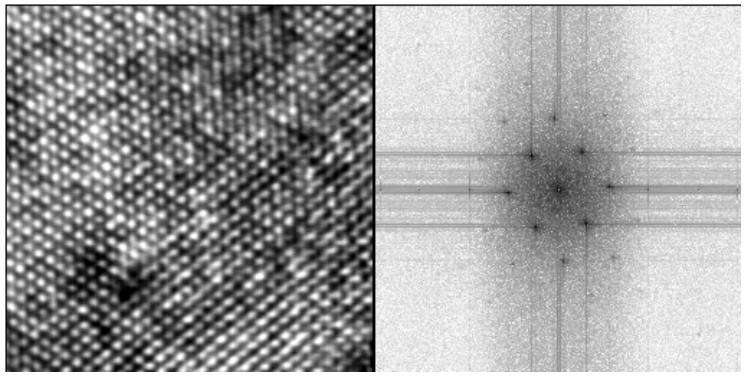


Figure 9. *High resolution TEM image of atomic lattice in silicon (left), with the frequency transform (right) (image courtesy Sopa Chevacharoenkul, Microelectronics Center of North Carolina).*

which can be measured by interpolating the peak position to a fraction of a pixel width, corresponding to an accuracy for the atom spacing of a few parts in 10,000. The spacing of the features that produce the point in the power spectrum is simply the width of the image (e.g., 256 pixels in the example, times whatever calibration applies) divided by the distance from the origin to the center of the peak in the power spectrum.

To the electron microscopist, the power spectrum image of the frequency-domain transform looks just like an electron diffraction pattern, which in fact it is. The use of microscope optics to form the diffraction pattern is an analog method of computing the frequency-domain representation. This can be done with any image by setting up suitable optics. While it is a fast way to obtain the frequency-domain representation, this method has two serious drawbacks for use in image processing.

First, the phase information is lost when the diffraction pattern is recorded, so it is not possible to reconstruct the spatial-domain image from a photograph of the diffraction pattern. (Recording the entire diffraction pattern including the phase information results in a hologram, from which the original image *can* be reconstructed.) It is possible to perform the reconstruction from the diffraction pattern in the microscope by using suitable lenses (indeed, that is how the microscope functions), so in principle it is possible to insert the various masks and filters discussed below. Making these masks and filters is difficult and exacting work, however, which must usually be performed individually for each image to be enhanced. Consequently, it is much easier (and more controllable) to use a computer to perform the transform and to apply any desired masks.

It is also easier to perform measurements on the frequency-domain representation using the computer. Locating the centers of peaks by curve fitting would require recording the diffraction pattern (typically with film, which may introduce nonlinearities or saturation over the extremely wide dynamic range of many patterns), followed by digitization to obtain numerical values. Considering the speed with which a spatial-domain image can be recorded, the frequency transform calculated, and interactive or automatic measurement performed, the computer is generally the tool of choice. This analysis is made easier by the ability to manipulate the display contrast so that both brighter and dimmer spots can be seen, and to use image processing tools such as background leveling and a top hat filter to isolate the peaks of interest.

When spots from a periodic structure are superimposed on a general background, the total power in the spots expressed as a fraction of the total power in the entire frequency transform gives a useful quantitative measure of the degree of periodicity in the structure. This may also be used to compare different periodicities (different spacings or orientations) by comparing summations of values in the power spectrum. For electron diffraction patterns, this is a function of the atomic density of various planes and the atomic scattering cross sections.

Although the display of the power spectrum corresponds to a diffraction pattern and is the most familiar presentation of frequency-space information, it must not be forgotten that the phase information is also needed to reconstruct the original image. **Figure 10** shows a test image, consisting of a regular pattern of spots, and its corresponding power spectrum and phase values. If the phase information is erased (all phases set to zero), the reconstruction (**Figure 11**) shows some of the same periodicity, but the objects are no longer recognizable. The various sine waves have been shifted in phase, so that the feature boundaries are not reconstructed.

The assumption that the image is one repetition of an endless sequence is also important. Most real images do not have perfectly matching left and right or top and bottom edges. This produces a large step function at the edge, which is more apparent if the image is shifted by an arbitrary offset (**Figure 12**). As noted previously, this does not alter the power spectrum image, although the phase image is shifted. The discontinuity requires high-frequency terms to fit, and since the edges are precisely horizontal and vertical, the power spectrum display shows a central cross superimposed on the rest of the data, which is visible in **Figure 10**. For the test pattern, the result of eliminating these lines from the original frequency transform and then retransforming is shown in **Figure 13**. The central portion of the image is unaffected, but at the edges the discontinuity is no longer sharp because the many frequencies needed to create the step functions at the edges are missing. The pattern from each side has been reproduced on the other side of the boundary, superimposed on the correct data.

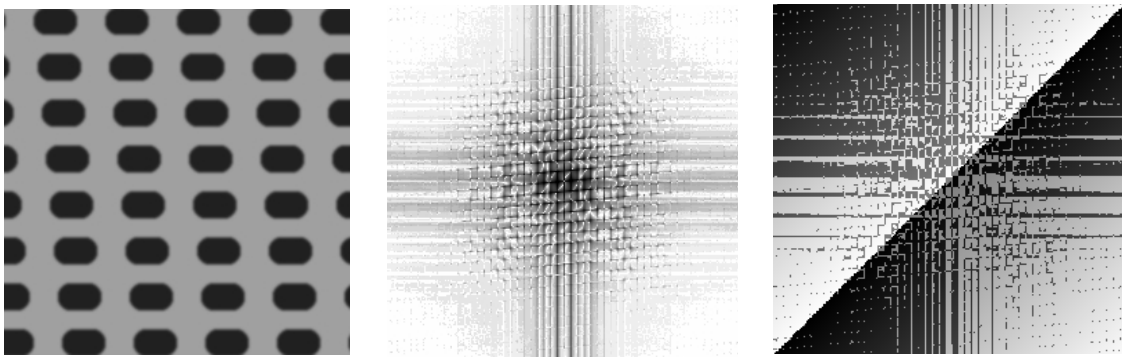


Figure 10. *Test image consisting of a regular pattern* ***(a)****, with its frequency transform power spectrum* ***(b)*** *and phase values* ***(c)****.*

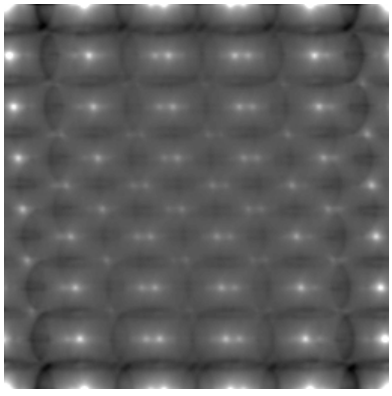


Figure 11. *Retransformation of* ***Figure 10*** *with all phase information set to zero.*

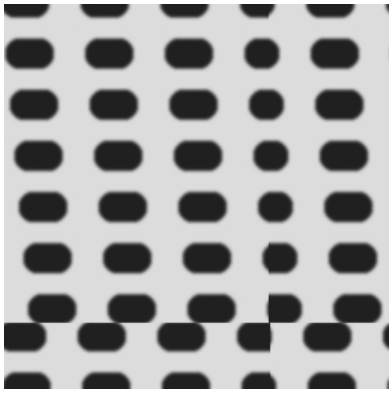


Figure 12. *The test image of* ***Figure 10*** *with an arbitrary spatial shift, showing the discontinuities at the image boundaries*

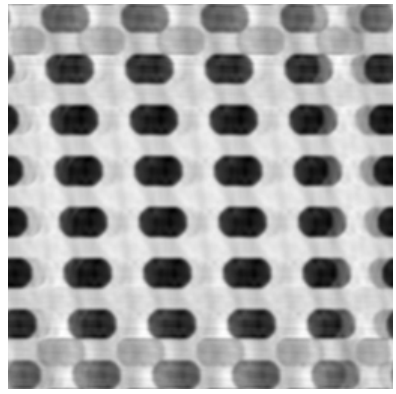


Figure 13. *Retransformation of* ***Figure 10*** *with the central cross (horizontal and vertical lines) reduced to zero magnitude, so that the left and right edges of the image, and the top and bottom edges, are forced to match.*

Preferred orientation

Figure 14 shows another example of a periodic structure, only much less perfect and larger in scale than the lattice images above. The specimen is a thin film of magnetic material viewed in polarized light. The stripes are oppositely oriented magnetic domains in the material that are used to store information in the film. The frequency transform of this image clearly shows the width and spacing of the domains. Instead of a single peak, some arcs show the variation in orientation of the stripes, which is evident in the original image but difficult to quantify.

The length of the arcs and the variation of brightness (power) with angle along them can be easily measured to characterize the preferred orientation in the structure. Even for structures that are not perfectly periodic, the integrated power as a function of angle can be used to measure the preferred orientation. This is identical to the results of autocorrelation operations carried out in the spatial domain, in which a binary image is shifted and combined with itself in all possible displacements to obtain a matrix of fractional values, but it is much faster to perform with the frequency-domain representation. Also, this makes it easier to deal with grey scale values.

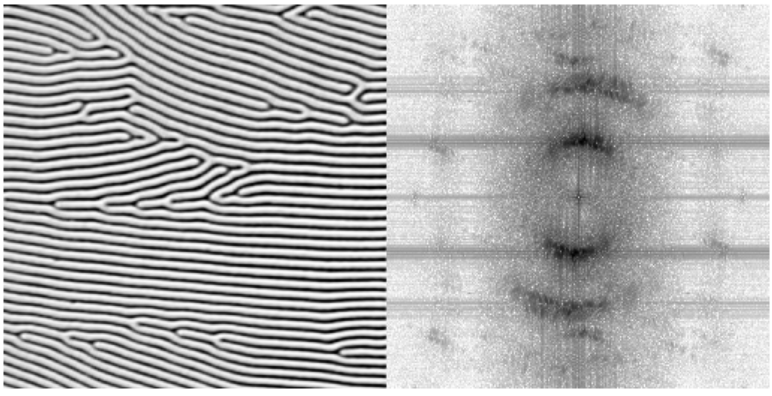


Figure 14. *Polarized light image of magnetic domains in thin film material (left), with the frequency transform (right).*

Reconstructing periodic structures that are not perfectly aligned can be performed by selecting the entire arc in the frequency transform. **Figure 15** illustrates this with a virus particle. The TEM image hints at the internal helical structure, but does not show it clearly. In the frequency transform, the periodic spacing and the variation in direction is evident. The spacing can be measured (2.41 nm) and the helix angle determined from the length of the arc. Retransforming only these arcs shows the periodicity, but this is not limited spatially to the virus particle. Using the spatial-domain image as a mask (as discussed in Chapter 7) makes the internal helical pattern evident.

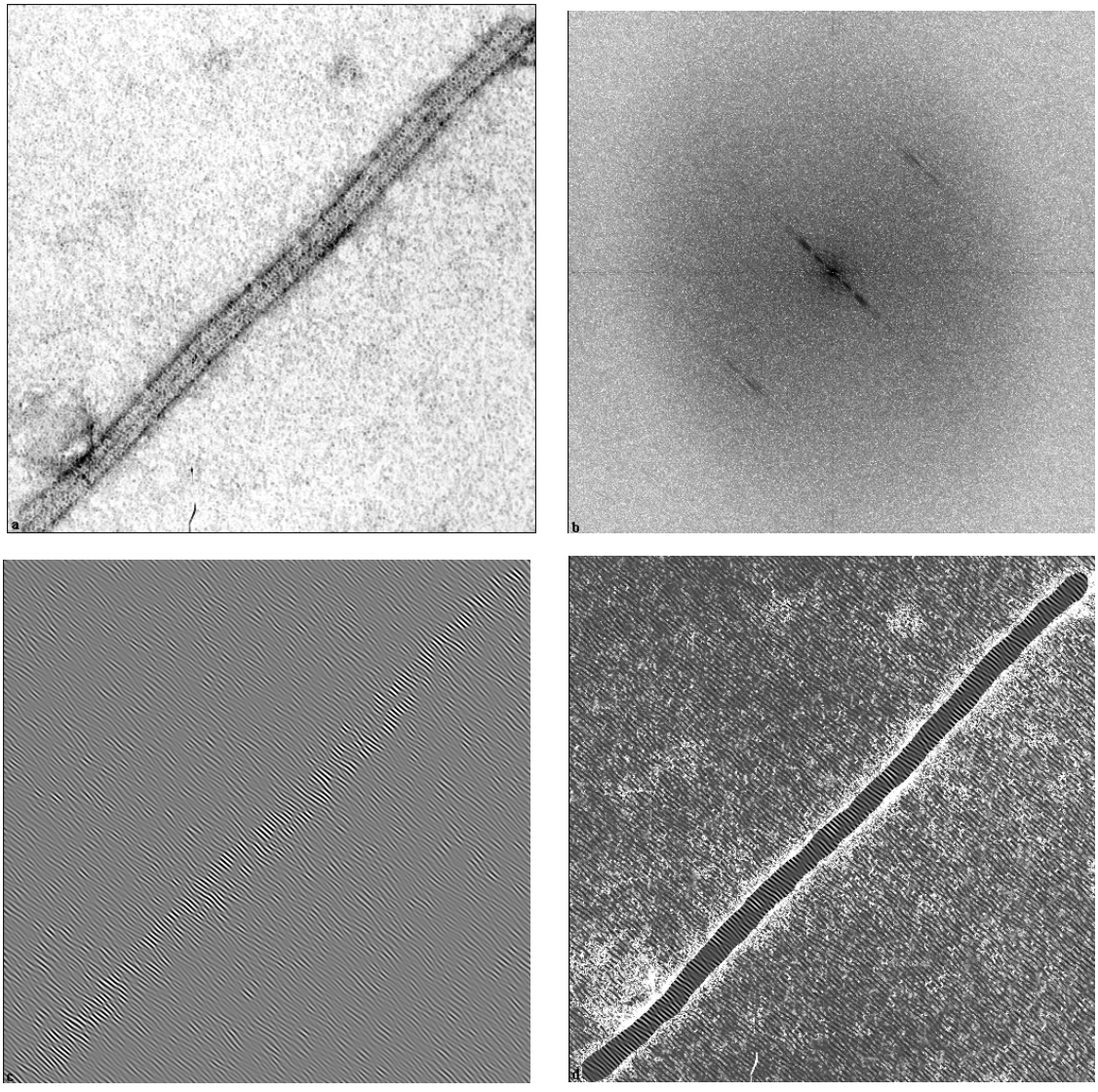


Figure 15. TEM image of a virus *(courtesy Dr. R. L. Grayson, Virginia Polytechnic Institute, Blacksburg, VA):*
(a) *original image, in which the internal helical structure is difficult to discern;*
(b) *frequency transform of image* ***a,*** *in which the regular repeating structure of the virus and its angular variation in orientation is evident;*
(c) *retransformation of just the peaks in the frequency transform, in which the periodic lines are not limited to the virus;*
(d) *using the virus particle as a mask, the helical pattern becomes evident.*

One particular type of preferred orientation in images, which arises not from the specimen but rather from the imaging system itself, is astigmatism. This is a particular problem with electron microscopes because of the operating principles of electromagnetic lenses. Even skilled operators devote considerable time to making adjustments to minimize astigmatism, and it is often very difficult to recognize it in images in order to correct it. Astigmatism results in the defocusing of the image and a consequent loss of sharpness in one direction, sometimes with an improvement in the perpendicular direction. This becomes immediately evident in the frequency transform, since the decrease in brightness or power falls off radially (at higher frequencies) and the asymmetry can be noted.

Figure 16 shows an example. The specimen is a cross section with three layers. The bottom is crystalline silicon, above which is a layer of amorphous (non-crystalline) silicon, followed by a layer of glue used to mount the sample for thinning and microscopy. The glue is difficult to distinguish by eye from the amorphous silicon. Frequency transforms for the three regions are shown. The regular structure in the pattern from the crystalline silicon gives the expected diffraction pattern. While the two regions above do not show individual peaks from periodic structures, they are not the same. The amorphous silicon has short-range order in the atomic spacings based on strong covalent bonding that is not visible to the human observer because of its chaotic overall pattern. This shows up in the frequency transform as a white cross in the dark ring, indicating that in the 45-degree directions there is a characteristic distance and direction to the next atom. This pattern is absent in the glue region, where there is no such structure.

In both regions, the dark circular pattern from the amorphous structure is not a perfect circle, but an ellipse. This indicates astigmatism. Adjusting the microscope optics to produce a uniform circle will correct the astigmatism and provide uniform resolution in all directions in the original image. It is much easier to observe the effects of small changes in the frequency-space display than in the spatial-domain image.

The frequency transform of an image can be used to optimize focus and astigmatism. When an image is in focus, the high-frequency information is maximized in order to sharply define the edges.

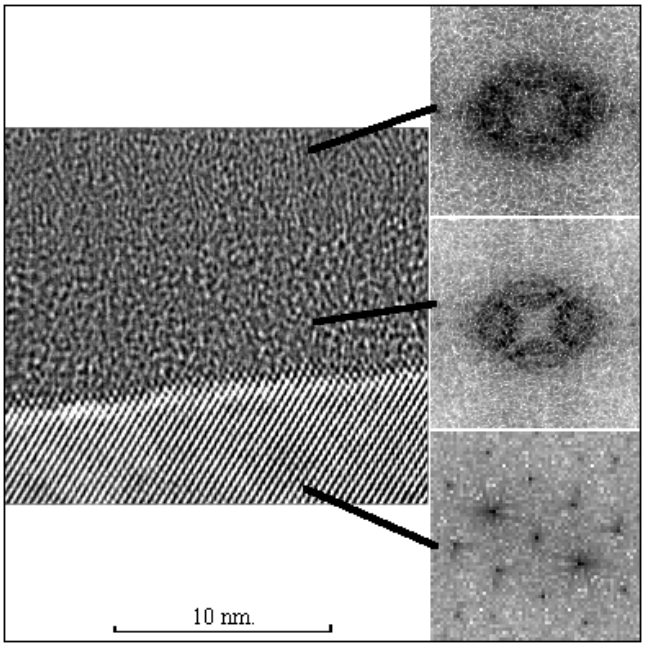


Figure 16. *TEM image of cross section of crystalline and amorphous silicon, and glue, with frequency transforms of each region shown at right (image courtesy Sopa Chevacharoenkul, Microelectronics Center of North Carolina).*

This provides a convenient test for sharpest focus. **Figure 17** shows a light microscope image that is in focus. The line profiles of the power spectrum in both the vertical and horizontal directions show a more gradual drop-off at high frequencies than **Figure 18**, which is the same image out of focus. When astigmatism is present (**Figure 19**), the power spectrum is asymmetric, as shown by the profiles.

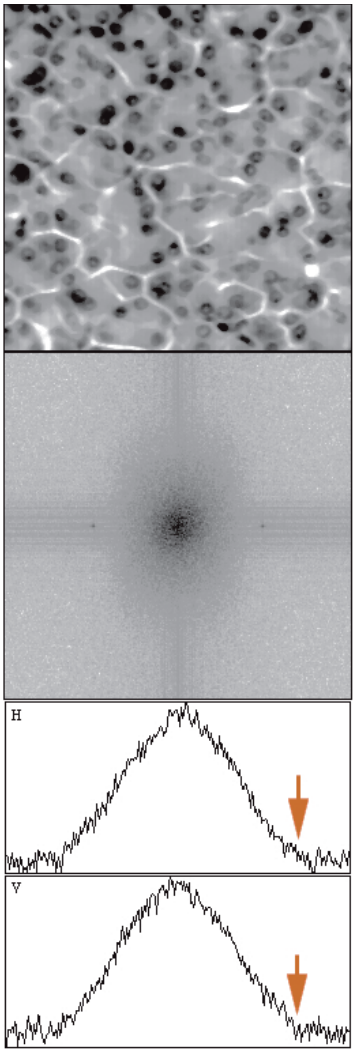


Figure 17. *Out-of-focus image with its power spectrum and horizontal and vertical line profiles, showing presence of high-frequency information as compared with* ***Figure 18****. Red arrows indicate actual resolution limit as discussed in text.*

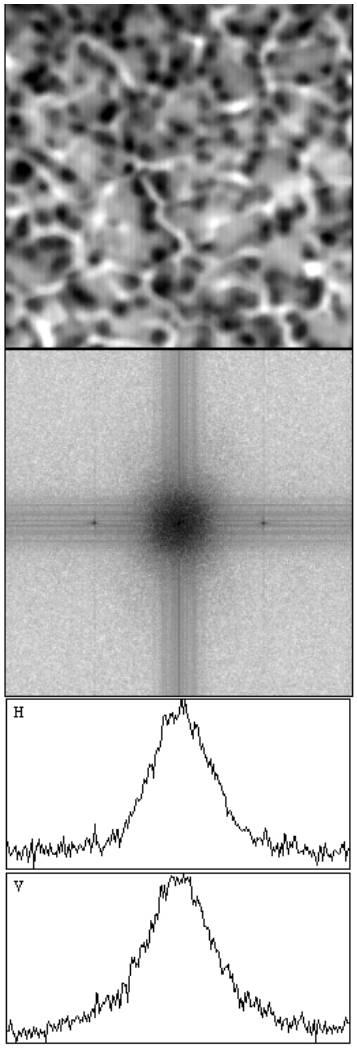


Figure 18. *Out-of-focus image with its power spectrum and horizontal and vertical line profiles, showing loss of high frequency information as compared to* ***Figure 17.***

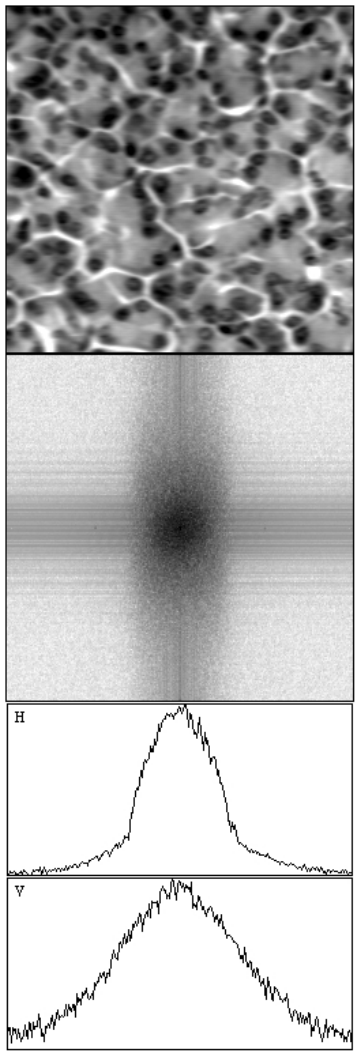


Figure 19. *Astigmatic image produced by misaligning lens, with its power spectrum and horizontal and vertical line profiles showing different high-frequency components.*

Profiles of the power spectrum also provide a convenient tool for measuring the actual resolution of images, to characterize various types of cameras and acquisition devices as discussed in Chapter 1. Most image acquisition procedures store an image that has somewhat more pixels than the actual resolution. As shown by the red arrows in **Figure 17**, the power spectrum shows a definite break in slope. This indicates the frequency at which the real information in the image ends; only noise is present at higher frequencies (smaller spacings).

Figure 20 shows a test pattern of radial lines. Due to the finite spacing of detectors in the video camera used, as well as limitations in electronics bandwidth that eliminate the very high frequencies required to resolve small details, these lines are incompletely resolved where they are close together. **Figure 21** shows the two-dimensional Fourier transform power spectrum of this image. The power spectrum is plotted isometrically in **Figure 22** to emphasize the drop-off in magnitude, which is different in the horizontal and vertical directions. As is common in video, the resolution along each scan line is poorer than the vertical resolution. **Figure 23** shows the complete set of data using color. One image shows the real and imaginary components of the transform

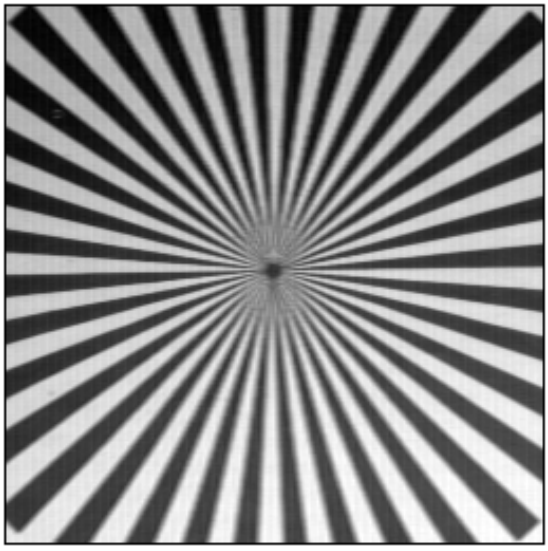
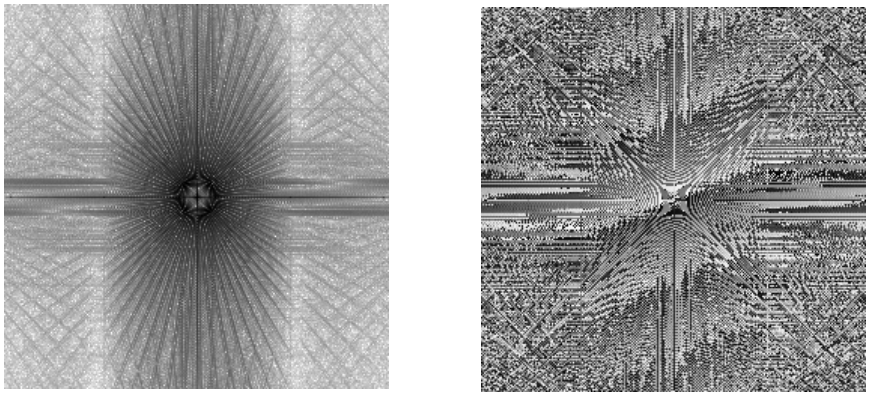


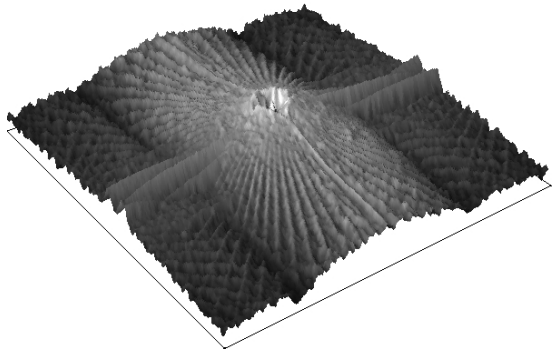
Figure 20. *Test pattern image.*

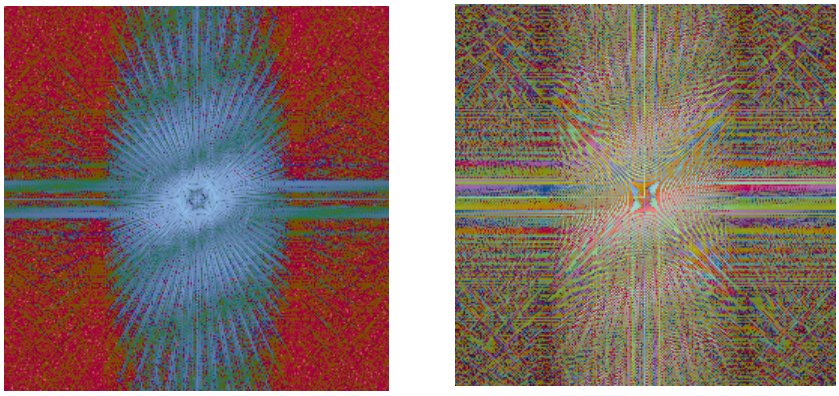


a **b**

Figure 21. Fourier transform of the test image in Figure 20:
(a) *power spectrum magnitude;*
(b) *phase (angles from 0 to 180° displayed as grey-scale values from black to white).*

Figure 22. *The power spectrum from* ***Figure 21*** *presented as an isometric view.*





a **b**

Figure 23. Color representations of the Fourier transform from Figure 21:
(a) *YUV image in which U = Real, V = Imaginary;*
(b) *RGB image in which Intensity = Magnitude, Hue = Phase.*

and the other the magnitude and phase, encoded in different color channels. Although technically complete, these displays are rarely used because the phase information is confusing to the viewer.

In the power spectrum, it is evident that there is a well-defined boundary, different in the x and y directions, beyond which the magnitude drops abruptly. This corresponds to the image resolution, which is different in the horizontal and vertical directions. In many cases, it is not so obvious where the physical source of resolution limitation lies. It can arise from the finite spacing of detectors in the camera, or various electronic effects in the amplification and digitization process; however, the Fourier transform power spectrum will still show the limit, permitting the resolution of any imaging system to be ascertained.

Texture and fractals

Besides the peaks in the power spectrum resulting from periodic structures that may be present and the ultimate limitation at high frequencies due to finite image resolution, it may seem as though there is only a noisy background containing little useful information. This is far from true. Many images represent the brightness of light scattered from surfaces, or other data such as surface elevation. In these images, the roughness or texture of the surface is revealed and may be measured from the power spectrum.

The concept of a fractal surface dimension will not be explained here in detail, but is discussed in Chapter 13. Surfaces that are fractal have an area that is mathematically undefined. It is greater than the projected area covered by the irregular surface and increases as the measurement scale becomes finer. The fractal dimension may be determined the slope of a line on a log-log plot of measured area vs. the size of the measuring tool. Many naturally occurring surfaces resulting from wear, erosion, agglomeration of particles, or fracture are observed to have this character. It has also been shown that images of these surfaces, whether produced by the scattering of diffuse light or the production of secondary electrons in an scanning electron microscope (SEM), are also fractal. That is, the variation of brightness with position obeys the same mathematical relationship. The fractal dimension is an extremely powerful and compact representation of the surface roughness, which can often be related to the history of the surface and the properties that result.

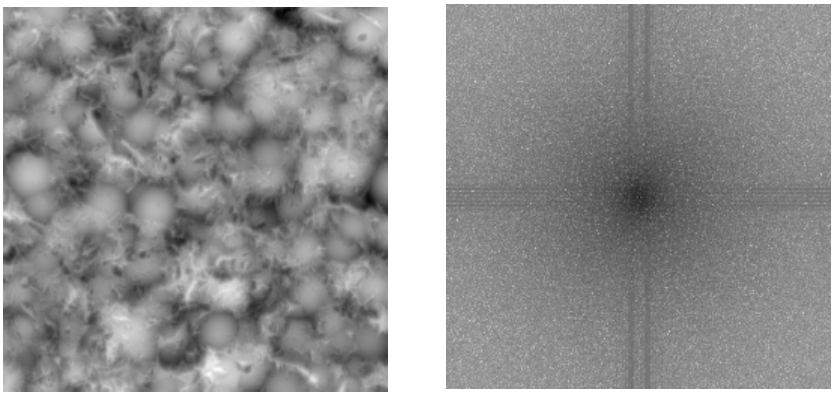
Measuring surface fractals directly is rarely practical, although it can be done physically by determining the number of molecules of various gases that can adhere to the surface as a function of their size. One common imaging approach is to reduce the dimensionality and measure the fractal dimension of a boundary line produced by intersecting the surface with a sampling plane. This may either be produced by cross sectioning or by polishing down into the surface to produce islands. In either case, the rougher the surface, the more irregular the line. In the case of polishing parallel to the surface to produce islands, the perimeter line also has a fractal dimension (the slope of a log-log plot relating the measured line length and the length of the measurement tool), which is just 1.0 less than that of the surface.

For a fractal surface, the power spectrum shows the superposition of sinusoids of all frequencies and orientations to have a specific shape: the magnitudes of the coefficients in a Fourier transform of a fractal curve decrease exponentially with the log of frequency, while the phases of the terms are randomized. This can be understood qualitatively, since by definition a fractal curve is self-similar and has detail extending to ever-finer scales (or higher frequencies). This also implies that the proportion of amplitudes of higher frequency terms must be self-similar. An exponential curve satisfies this criterion.

Plotting the log of the amplitude (or the power spectrum, which is the square of the amplitude) versus the logarithm of frequency produces a straight line plot, which is easily analyzed. There is a simple relationship between the fractal dimension of a profile and the exponential decrease in magnitude of the terms in a Fourier expansion, as had been predicted by Feder (1989). This correlation makes it practical to use the radial decrease of magnitude in a two-dimensional Fourier-transform image as a measure of roughness and the directional variation of that decrease as a measure of orientation (Mitchell and Bonnell, 1990; Russ, 1990; Russ, 1994).

Figure 24 shows a range image (brightness represents elevation) for a fractal surface, with its Fourier transform power spectrum. Plotting log (amplitude) vs. log (frequency) as shown in **Figure 25** produces a straight line, and a plot of the histogram of the phase values shows a uniform random distribution, which confirms the fractal geometry of the surface. The slope of the plot gives the dimension of the surface (which must lie between 2.0 for a Euclidean surface and 2.999, for one so irregular that it effectively fills three-dimensional space) as Fractal Dimension = (6 + Slope)/2, or about 2.2 for the example shown. This is an isotropic surface produced by shotblasting a metal, so the slope is the same in all directions.

Figure 26 shows another surface, this produced by machining, with tool marks oriented in a specific direction. Similar analysis of this surface shows (**Figure 27**) that while it is still fractal (with average dimension about 2.25), it is not isotropic. Actually, two methods are used to depart from isotropy: one is changing the slope of the Log (amplitude) vs. Log (frequency) plot, and the other its intercept, as a function of direction. Chapter 13 discusses fractal surface geometry in more detail (see also Russ, 1994 and Russ, 2001b).



a b

Figure 24. *Range image of a shotblasted metal surface, with its Fourier transform power spectrum.*

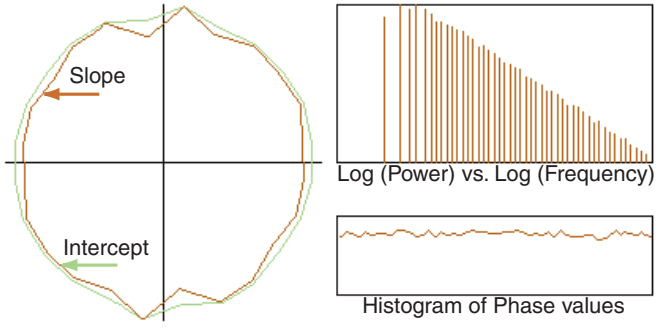


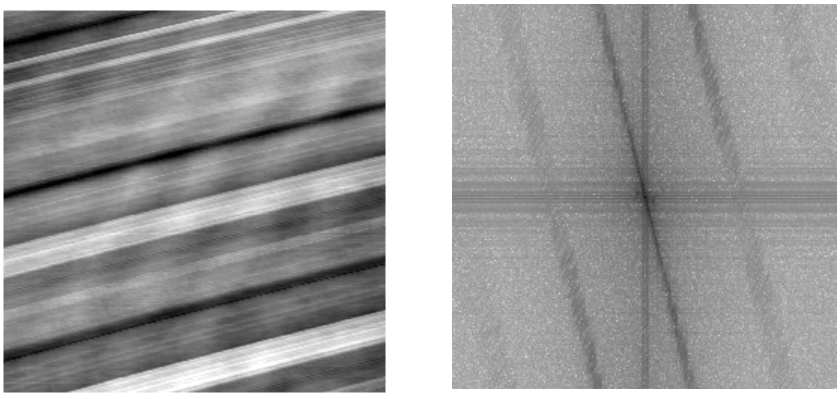
Figure 25. *Analysis of the Fourier transform data showing the isotropic fractal nature of the surface in* ***Figure 24.***

Measuring the two-dimensional Fourier transform is more efficient than measuring many individual brightness profiles in the original image. It also allows any periodic structures that may be present to be ignored, since these show up as discrete points in the frequency-transform image and can be skipped in determining the overall exponential decrease in the magnitude values. In other words, it is possible to look beyond the periodic structures or noise (e.g., arising from electronic components) in the images and still characterize the underlying chaotic but self-similar nature of the surface.

Filtering images

Isolating periodic noise

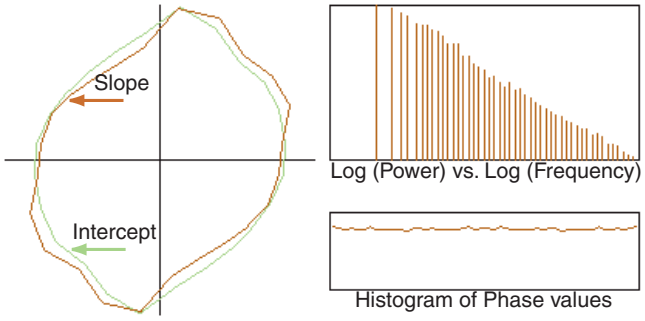
It was noted earlier (and shown in **Figure 5** for a simple case) that the frequency transform has a property of separability and additivity. Adding together the transforms of two original images or functions produces the same result as the transform of the sum of the originals. This idea opens the way to using subtraction to remove unwanted parts of images. It is most commonly used to remove periodic noise, which can be introduced by the devices used to record or transmit images. We will see some examples below. If the frequencies associated with the noise can be determined (often possible directly from the Fourier transform), then setting the amplitude of those terms to zero will leave only the desired part of the information. This can then be inverse transformed back to the spatial domain to produce a noise-free image.



a b

Figure 26. *Range image of a machined metal surface, with its Fourier transform power spectrum.*

Figure 27. *Analysis of the Fourier transform data showing the anisotropic fractal nature of the surface in* ***Figure 26.***



The same method of removing selected frequencies can be used for a more basic type of filtering as well. **Figure 28** shows an image with its Fourier transform power spectrum. There are no evident "spikes" or noise peaks, just the usual gradual reduction in the amplitude of higher frequencies. Keeping the low frequencies and removing the high frequencies can be accomplished by zeroing the amplitude of all sinusoids above a selected frequency. The red circle in **Figure 28b** was used to perform this "low-pass" filtering operation (i.e., passing or keeping the low frequencies), to produce the result in **Figure 28c**. Conversely, keeping the high frequencies and removing the low frequencies (a "high-pass" filter) produces the result in **Figure 28d**.

Except for the ringing around boundaries (discussed below), the results are exactly the same as those for Gaussian smoothing and Laplacian sharpening, shown in the two preceding chapters. In fact, it can be shown mathematically that the operations are exactly the same whether performed in the frequency domain or in the pixel domain.

In this illustration of filtering, portions of the Fourier-transform image were selected based on frequency, which is why these filters are generally called low-pass and high-pass filters. Usually, selecting arbitrary regions of the frequency domain for reconstruction produces artefacts, unless some care is taken to shape the edges of the filter region to attenuate the data smoothly. This can be seen in the one-dimensional example of the step function in **Figure 1** and the corresponding two-dimensional example of **Figure 7**. If only the first few terms are used then, in addition to not

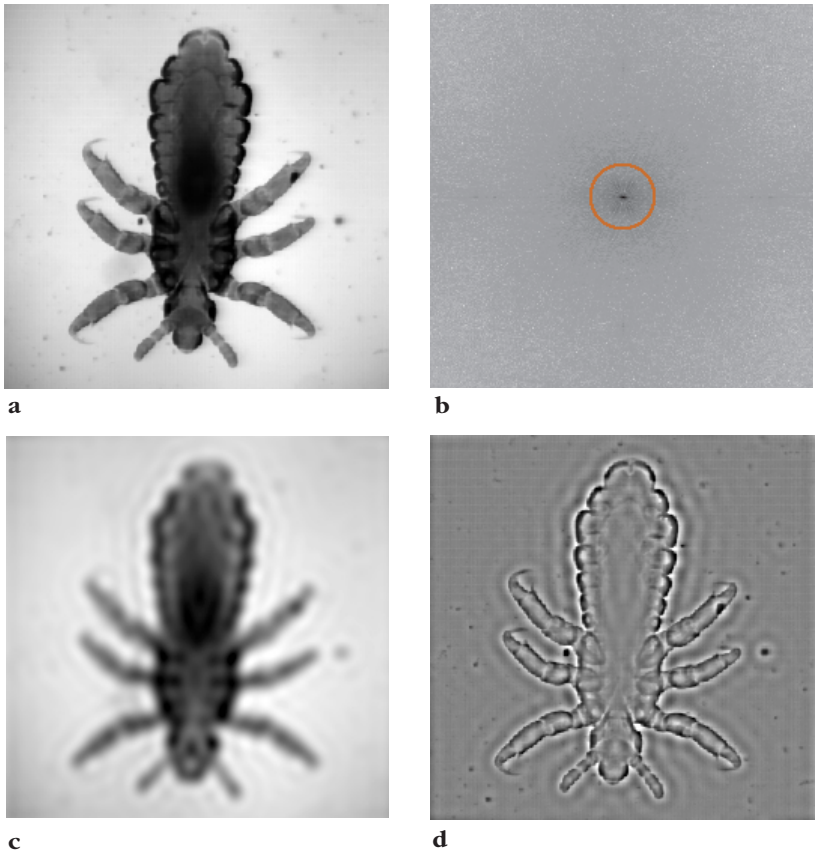


Figure 28. Frequency filtering using the Fourier transform: ***(a)*** *original image;* ***(b)*** *power spectrum with a circular frequency cutoff;* ***(c)*** *retransforming just the low frequencies inside the circle (a low-pass filter);* ***(d)*** *retransforming just the high frequencies outside the circle (a high-pass filter).*

modeling the steepness of the step, the reconstruction has oscillations near the edge, which are generally described as ringing.

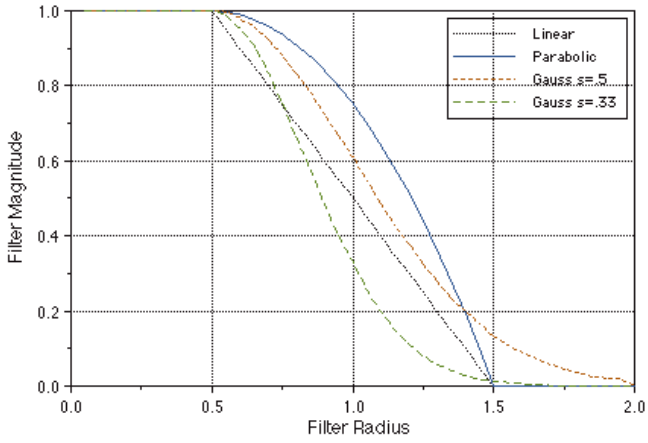
It is necessary to shape the edge of the filter to prevent ringing at sharp discontinuities. This behavior is well-known in one-dimensional filtering (used in digital signal processing, for example). Several different shapes are commonly used. Over a specified width (usually given in pixels, but of course ultimately specified in terms of frequency or direction), the filter magnitude can be reduced from maximum to minimum using a weighting function. The simplest function is linear interpolation (also called a Parzen window function). Better results can be obtained using a parabolic or cosine function (also called Welch and Hanning window functions, respectively, in this context). The most elaborate filter shapes do not drop to the zero or minimum value, but extend a very long tail beyond the cutoff point. One such shape is a Gaussian. **Figure 29** shows several of these shapes.

Another filter shape often used in these applications is a Butterworth filter, with a magnitude that can be written as

$$H = 1 / \left[1 + C\left(\frac{R}{R_0}\right)^{2n} \right] \tag{8}$$

where R is the distance from the center of the filter (usually the center of the power spectrum image, or zero-frequency point), and R_O is the nominal filter cutoff value. The constant C is often set equal to 1.0 or to 0.414; the value defines the magnitude of the filter at the point where $R = R_0$ as either 50% or $1/\sqrt{2}$. The integer n is the order of the filter; its most common values are 1 or 2.

Figure 29. *Some common filter edge profiles.*



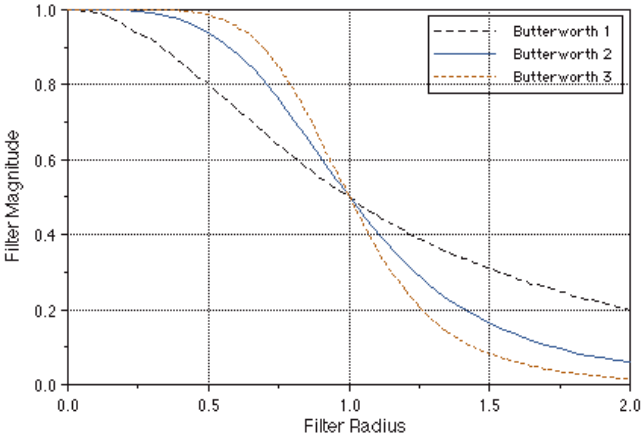


Figure 30. *Shapes for Butterworth filter profiles of order 1, 2, and 3.*

Figure 30 shows comparison profiles of several Butterworth low-pass filters (ones that attenuate high frequencies). The converse shape having negative values of *n* passes high frequencies and attenuates low ones. **Figure 31** shows the effect that shaping the frequency cutoff has on the quality of the retransformed images from **Figure 28**.

To illustrate the effects of these filters on ringing at edges, **Figure 32** shows a simple test shape and its two-dimensional FFT power spectrum image. The orientation of principal terms perpendicular to the major edges in the spatial-domain image is evident. Performing a reconstruction using a simple aperture with a radius equal to 25 pixels (called an ideal filter) produces the result shown in **Figure 33a**. The oscillations in brightness near the edges are quite visible.

Ringing can be reduced by shaping the edge of the filter, as discussed above. The magnitudes of frequency terms near the cutoff value are multiplied by factors less than one, whose values are based on a simple function. If a cosine function is used, which varies from 1 to 0 over a total width of 6 pixels, the result is improved (**Figure 33b**). In this example, the original 25-pixel radius used for the ideal filter (the sharp cutoff) is the point at which the magnitude of the weighting factor drops to 50%. The weights drop smoothly from 1.0 at a radius of 22 pixels to 0.0 at a radius of 28 pixels.

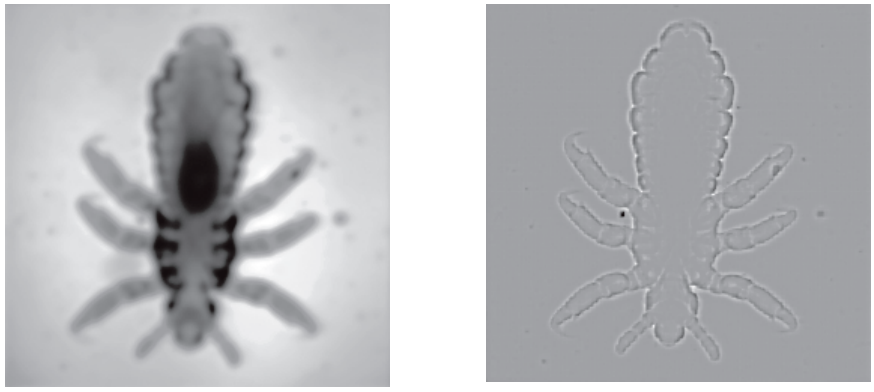


Figure 31. Shaping the frequency cutoff with a Butterworth filter:
(a) *low-pass filter (compare with* ***Figure 28c****);*
(b) *high-pass filter (compare with* ***Figure 28d****).*

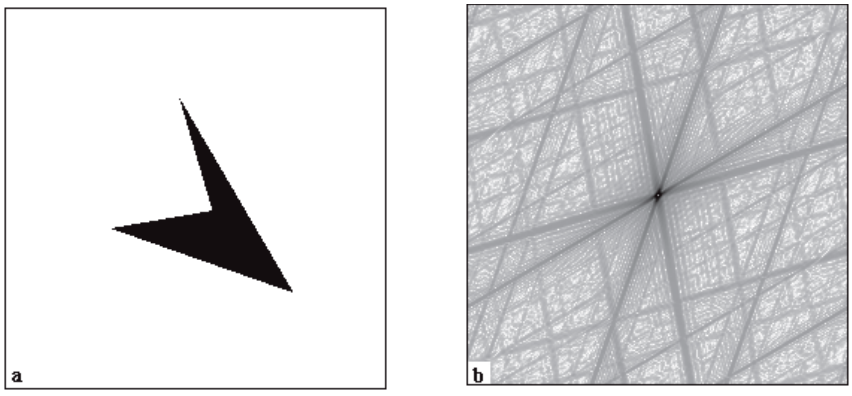


Figure 32. *Test shape with its frequency transform power spectrum.*

Increasing the distance over which the transition takes place further reduces the ringing, as shown in **Figure 33c**. Here, the 50% point is still at 25 pixels but the range is from 15 to 35 pixels. Note that the improvement is not achieved simply by increasing the high-frequency limit, which would improve the sharpness of the feature edges but would not by itself reduce the ringing. **Figure 33d** shows the same reconstruction using a second-degree Butterworth filter shape whose 50% point is set at 25 pixels.

Figure 34 shows an image with both fine detail and some noise along with its frequency transform. Applying Butterworth low-pass filters with radii of 10 and 25 pixels in the frequency-domain image smooths the noise with some blurring of the high-frequency detail (**Figure 35**), while the application of Butterworth high-pass filters with the same radii emphasizes the edges and reduces the contrast in the large (low-frequency) regions (**Figure 36**). All of these filters were applied to the amplitude values as multiplicative masks. Adjusting the radius of the filter cutoff controls the range of frequencies in the inverse transformed image, just as changing the size of the convolution kernel alters the degree of smoothing or sharpening when the equivalent procedure is carried out in the spatial or pixel domain.

Of course, it is not necessary for the variation in magnitude to be from 1 to 0. Sometimes the lower limit is set to a fraction, so that the high (or low) frequencies are not completely attenuated.

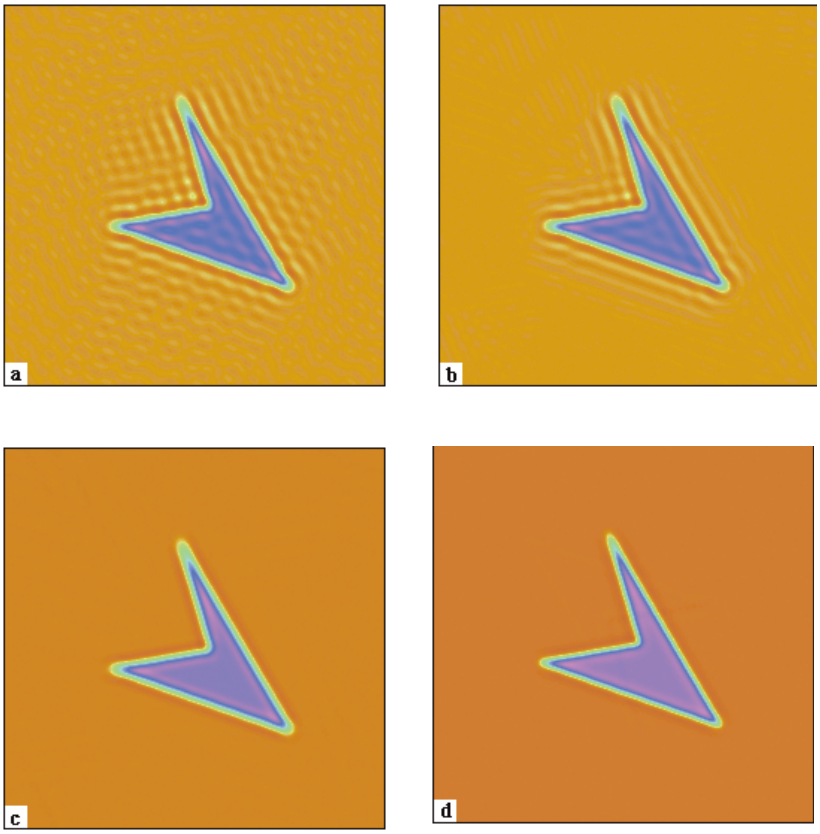


Figure 33. ***Reconstruction of shape from its frequency transform in Figure 46, using a 25-pixel aperture (mask or filter) diameter:*** ***(a)*** *ideal filter in which the cutoff is exact and abrupt;* ***(b)*** *cosine-weighted edge shape with a half-width of 3 pixels;* ***(c)*** *cosine-weighted edge shape with a half-width of 10 pixels;* ***(d)*** *Butterworth second-degree shape. False color has been added to increase the visibility of small variations.*

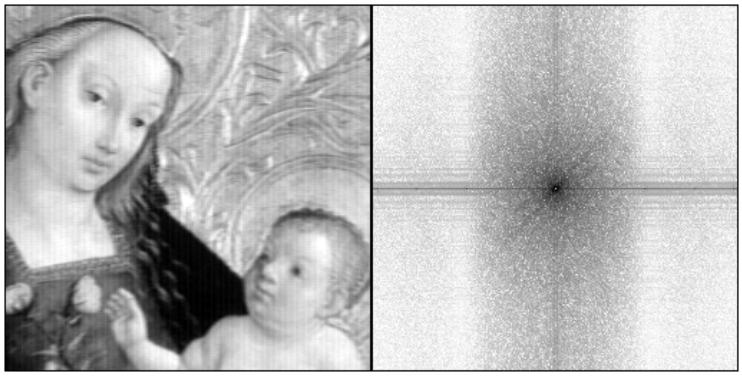


Figure 34. *Image and its transform used for filtering in* ***Figures 35*** *and* ***36****.*

It is also possible to use values greater than 1. A high-frequency emphasis filter with the low-frequency value set to a reduced value, such as 0.5, and a high-frequency value greater than 1, such as 2, is called a homomorphic filter. It is usually applied to an image whose brightness values have previously been converted to their logarithms (using a LUT). This filtering operation will simultaneously increase the high-frequency information (sharpening edges) while reducing the overall brightness range to allow edge brightness values to show. The physical reasoning behind the homomorphic filter is a separation of illumination and reflectance components in the image. As with most of these filters, though, the real justification is that it improves the appearance of many images of practical interest.

Figure 35. *Filtering of* ***Figure 34*** *with low-pass Butterworth Filters having 50% cutoff diameters of 10 (left) and 25 pixels (right).*

Figure 36. *Filtering of* ***Figure 34*** *with high-pass Butterworth Filters having 50% cutoff diameters of 10 (left) and 25 pixels (right).*

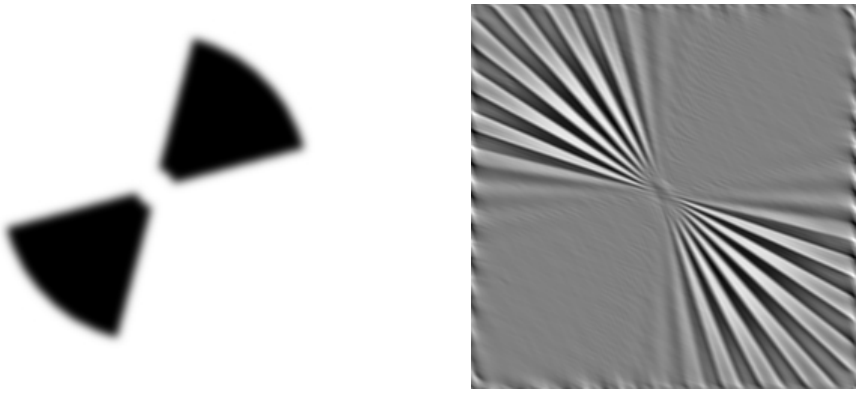
It is also possible to select a region of the Fourier transform image that is not centered on the origin. **Figure 37** shows a selection of intermediate frequency values lying in a particular direction on the transform in **Figure 20**, along with the resulting reconstruction. This kind of filtering may be useful to select directional information from images. It also demonstrates the basic characteristic of Fourier transform images: locations in the Fourier transform image identify periodicity and orientation information from any or all parts of the spatial-domain image. Note that the correct shape for off-centered regions is a combination of annuli and wedges that have cutoffs corresponding to frequencies and angles, rather than the more easily constructed circles or rectangles, and that the edges require the same type of smoothing shown previously.

Masks and filters

Once the location of periodic noise in an original image is isolated into a single point or a few points in the Fourier-transform image, it becomes possible to remove the noise by removing those sinusoidal terms. A filter removes selected frequencies and orientations by reducing the magnitude values for those terms, either partially or to zero, while leaving the phase information alone (which is important in determining where in the image that information appears).

Many different ways are used to specify and to apply this reduction. Sometimes it is practical to specify a range of frequencies and orientations numerically, but most often it will be convenient to do so using the magnitude or power spectrum display of the Fourier transform image. Manually or automatically selecting regions on this display allows specific peaks in the power spectrum, corresponding to the periodic information, to be selected for elimination.

Rather than the use of combinations of arcs and radial lines, for isolated noise peaks (often referred to as noise “spikes”) it is convenient and usually acceptable to use small circles to define the



a b

Figure 37. *Inverse transform of the Fourier transform of* ***Figure 20 (b)*** *through a filter selecting a range of frequencies and angles* ***(a)****.*

regions. It is important to use a smoothing function to modify the edges of the regions. When the transition takes place over a distance of only a few pixels, the differences between the various transition curves described previously are of little importance, and a simple Gaussian smooth is most commonly applied.

Usually, it is only by first examining the transform-image power spectrum that the presence and exact location of these points can be determined. **Figure 38** shows an image of a halftone print from a magazine. The pattern results from the halftone screen used in the printing process. In the frequency transform of the image, this regular pattern shows up as well-defined narrow peaks or spikes. Filtering removes the peaks by setting the magnitude at those locations to zero (but not altering any of the phase information). This allows the image to be retransformed without the noise. The filtered Fourier transform image power spectrum shown in **Figure 39** shows the circular white spots where filtering was performed.

It is interesting to note that image compression does not necessarily remove these noise spikes in the Fourier transform. Compression is generally based on discarding the terms in the transform (Fourier, Cosine, etc.) whose magnitudes are small. The halftone pattern or other periodic noise is interpreted as an important feature of the image by these compression methods. The need to preserve the periodic noise spikes actually reduces the amount of other useful detail that can be retained.

The effects of compression are best examined in the frequency transform. **Figure 40** shows a portion of the image of Saturn used in Chapter 4, before and after compression by the JPEG algorithm. Some differences, particularly in the rings, can be seen. The frequency transforms of the two images show that this is due to the loss of lines of high frequency information that are needed to define the profiles of the rings and the detail within them.

The example in **Figure 39** relied on human observation of the peaks in the Fourier transform image, recognition that the peaks were responsible for the periodic noise, and selecting them to produce the filter. In some cases, it is possible to construct an appropriate image filter automatically from the Fourier transform power spectrum. The guiding principle is that peaks in the power spectrum that are narrow and rise significantly above the local background should be removed. If the Fourier transform magnitude image is treated like an ordinary spatial-domain grey scale image, this peak removal can often be accomplished automatically using a rank-based filter like the top hat.

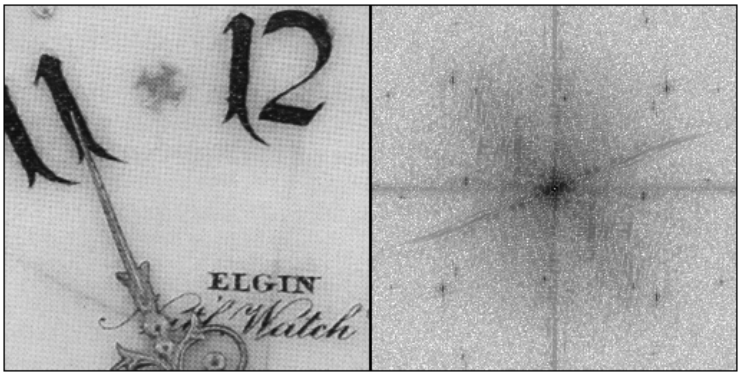


Figure 38. *A halftoned printed image (left) and its frequency transform (right).*

Figure 39. *Removal of periodic information from* ***Figure 38*** *by reducing the magnitude (left) and retransforming (right).*

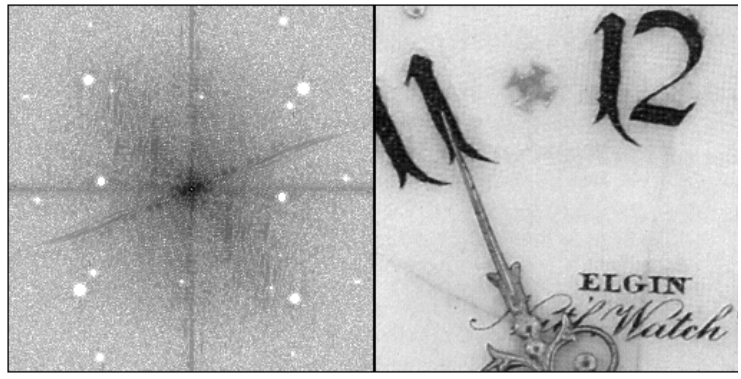


Figure 41 shows the power spectrum from the image in **Figure 38** and the result of applying a top hat filter with an inner radius of 3 pixels, an outer radius of 5 pixels, and a height of 8 grey-scale values. The brim of the hat rests on the background that drops off gradually and smoothly as frequency (radius) increases. The located features correspond to the noise spikes. Enlarging the spots by a few pixels and applying a smoothing to the edges, and inverting this result creates a mask that will remove the spikes. The figure shows the resulting modified power spectrum and the retransformed result.

Sometimes, the process of determining where the noise peaks are located can be simplified by selecting a region of the image that exhibits the noise pattern in an otherwise uniform area. This sequence is demonstrated in **Figure 42** for the same image as in **Figures 38** and **41**. A region of halftone periodic noise is selected and the rest of the image cleared. The transform of this image is then processed by smoothing and leveling, and then thresholded to locate the peaks. The inverse of this mask is then smoothed and multiplied by the frequency transform of the original image to produce the filtered result.

Removal of noise spikes is not always enough to restore an image completely. The example in **Figure 43** shows an example of an image scanned in from a newspaper; the halftone dots are very evident. The dark spots ("spikes") in the power spectrum (**Figure 43b**) correspond to the periodic structure in the image. They align with the repetitive pattern of dots, and their darkness indicates how much of various frequencies are present. Removing them is equivalent to removing the periodic component. In this case a top hat filter was used to locate the spikes and create a mask (**Figure 43c**). The result of applying this mask as a filter to remove the spikes shows that the periodic noise has been removed without affecting any of the other information present (**Figure 43d**). There is still some pixel-to-pixel noise because the image has been scanned at a higher magnification than it was printed, and the halftone cells in the original image are separated.

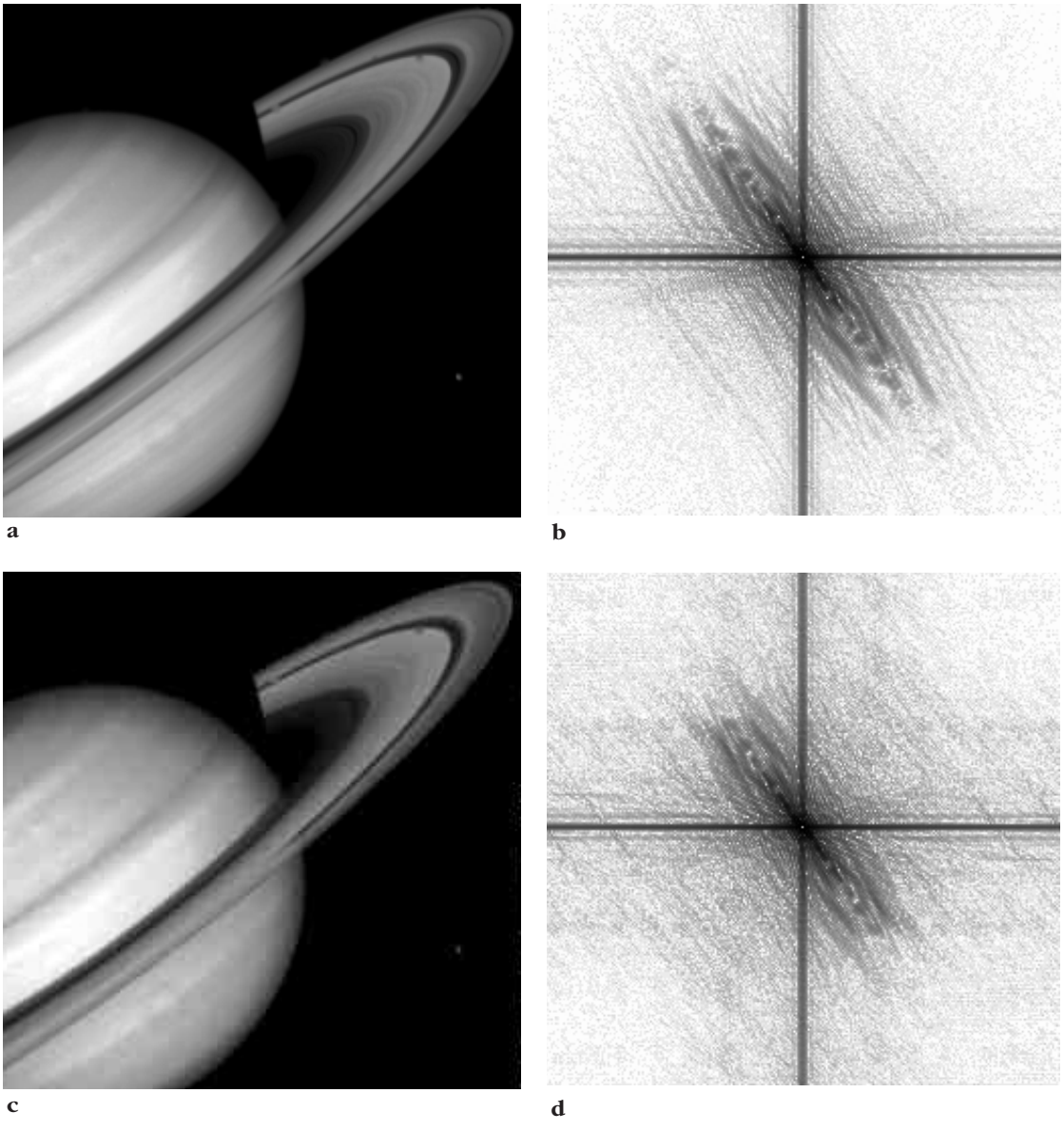
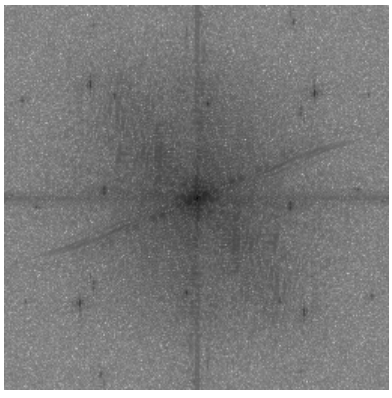
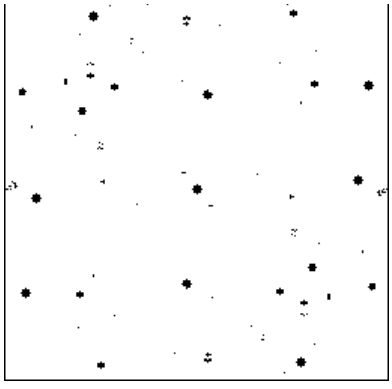


Figure 40. Portion of the image of Saturn from Chapter 4:
(a) *original;*
(b) *FT power spectrum showing high frequencies needed for detail in rings;*
(c) *image after JPEG compression (factor of 12);*
(d) *FT power spectrum of* ***c*** *showing loss of high-frequency information.*

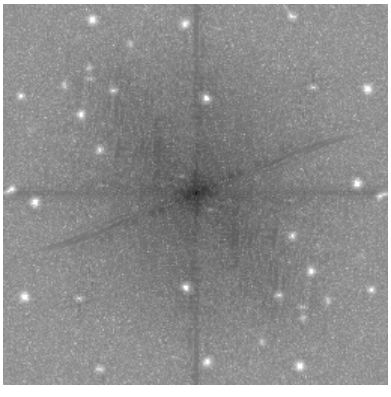
An additional filter can be employed to fill in the space between cells. A Butterworth second-order, high-frequency cutoff filter keeps low frequencies (gradual variations in grey scale) while progressively cutting off higher ones. In this case the midpoint of the cutoff was set to the spacing of the halftone dots in the original image. The final version of the power spectrum (**Figure 43e**) shows the periodic spots removed and the high frequencies attenuated. An inverse transform produces the final image (**Figure 43f**). Note that even the cat's whiskers which are barely discernible in the original image can be clearly seen.



a



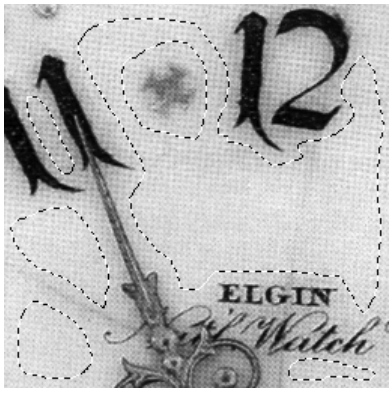
b



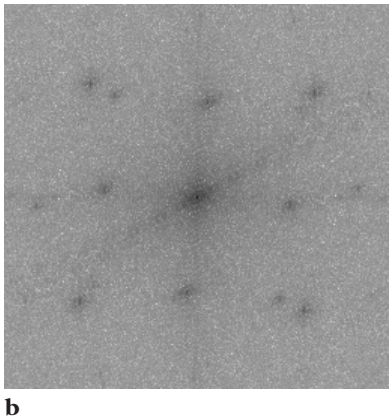
c

d

Figure 41. Automatic location and removal of noise spikes: ***(a)*** *original power spectrum from* ***Figure 38****;* ***(b)*** *top hat filter applied to* ***a****;* ***(c)*** *modified power spectrum after removal of spikes found in* ***b****;* ***(d)*** *retransformed image.*

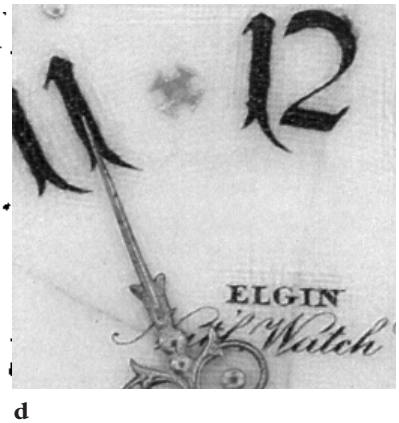


a



b

c



d

Figure 42. Identification of noise spikes by selecting image regions that exhibit the noise pattern (a): ***(b)*** *Fourier transform power spectrum from the selected areas;* ***(c)*** *mask created by processing* ***b****;* ***(d)*** *result of applying the filter from* ***c*** *to the Fourier transform of the entire original image.*

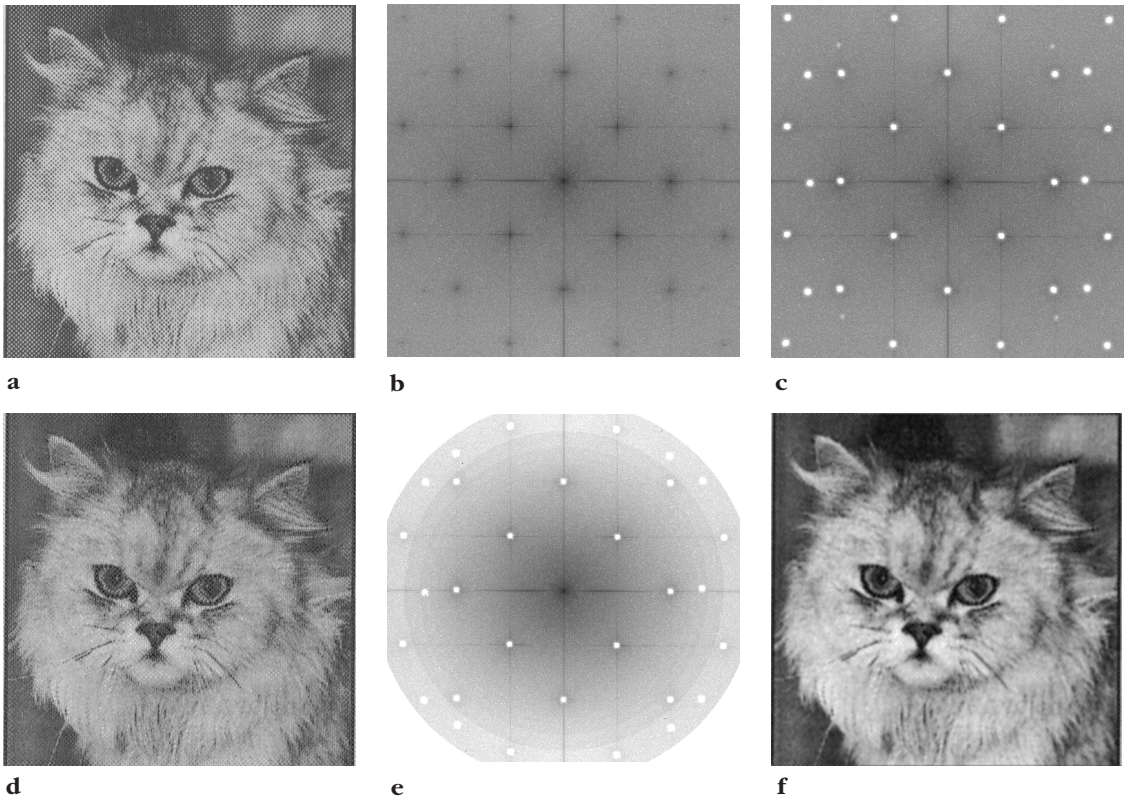
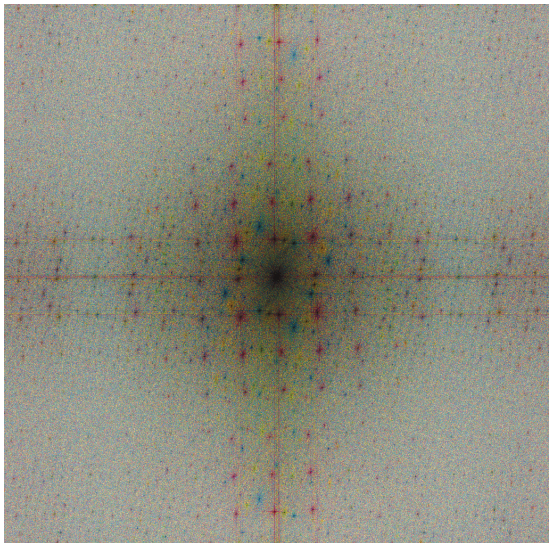


Figure 43. Removal of halftone pattern:
(a) *original scanned image;*
(b) *power spectrum;*
(c) *mask applied to remove spikes;*
(d) *inverse transform with spikes removed;*
(e) *Butterworth low-pass filter applied to power spectrum;*
(f) *final result.*

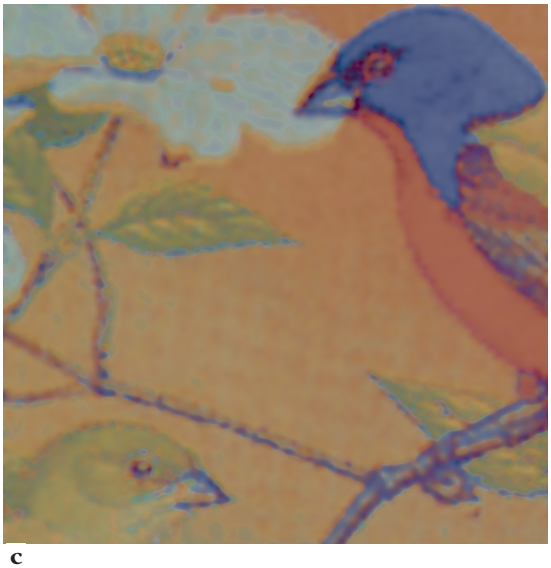
A common recommendation (found in internet newsgroups, for instance) for dealing with scanned halftone images or ones with moiré patterns from the scanner is to use smoothing to eliminate the pattern (sometimes by scanning the image at an angle and rotating it in software, using the interpolation as a low-pass filter, sometimes by scanning at a larger size and again depending on the filtering effects of interpolation as the image size is reduced, and sometimes by the overt application of a smoothing algorithm). This is a flawed strategy. The use of a smoothing or blur function in this image would have erased these fine lines long before the halftone dots were smoothed out.

When the original image has been scanned in from a color print, the situation is slightly more complicated. As mentioned in Chapter 2 and shown in **Figure 44**, colored halftone prints are typically made using cyan, magenta, yellow, and black inks, with each color having a halftone screen oriented at a different angle. The procedure required is to separate the color channels, process each one, and then recombine them for the final result. In the figure, the power spectra for each of the channels is shown superimposed, in the corresponding color, to show the different screen orientations. A top hat filter was applied to the Fourier transform power spectrum from each channel to select the noise peaks, which were then removed. Recombining the resultant color channel images produces the result shown.

a



b



c

Figure 44. Halftone removal in a color image: ***(a)*** *original image (a portion of a postage stamp, printed in CMYK);* ***(b)*** *Fourier transform power spectra for each color channel, superimposed in color;* ***(c)*** *removal of noise from each channel and recombining the color channels.*

Selection of periodic information

In some types of images, it is the periodic information that is useful and the non-periodic noise that must be suppressed. The methods for locating the periodic peaks, constructing filters, smoothing the filter edges, and so forth are unchanged. The only difference is that the filter sense is changed and in the case of a multiplicative mask, the values are inverted.

Figure 45 shows a one-dimensional example. The image is a light micrograph of stained skeletal muscle, in which there is a just-visible band spacing that is difficult to measure because of the spotty staining contrast (even with contrast expansion). The FT image shows the spots that identify the band structure. Reconstruction with only the first harmonic shows the basic band structure, and adding the second and third harmonics defines the band shape fairly well.

Figure 46 shows a high-resolution TEM lattice image from a crystalline ceramic (mullite). The two-dimensional periodicity of the lattice can be seen, but it is superimposed on a variable and

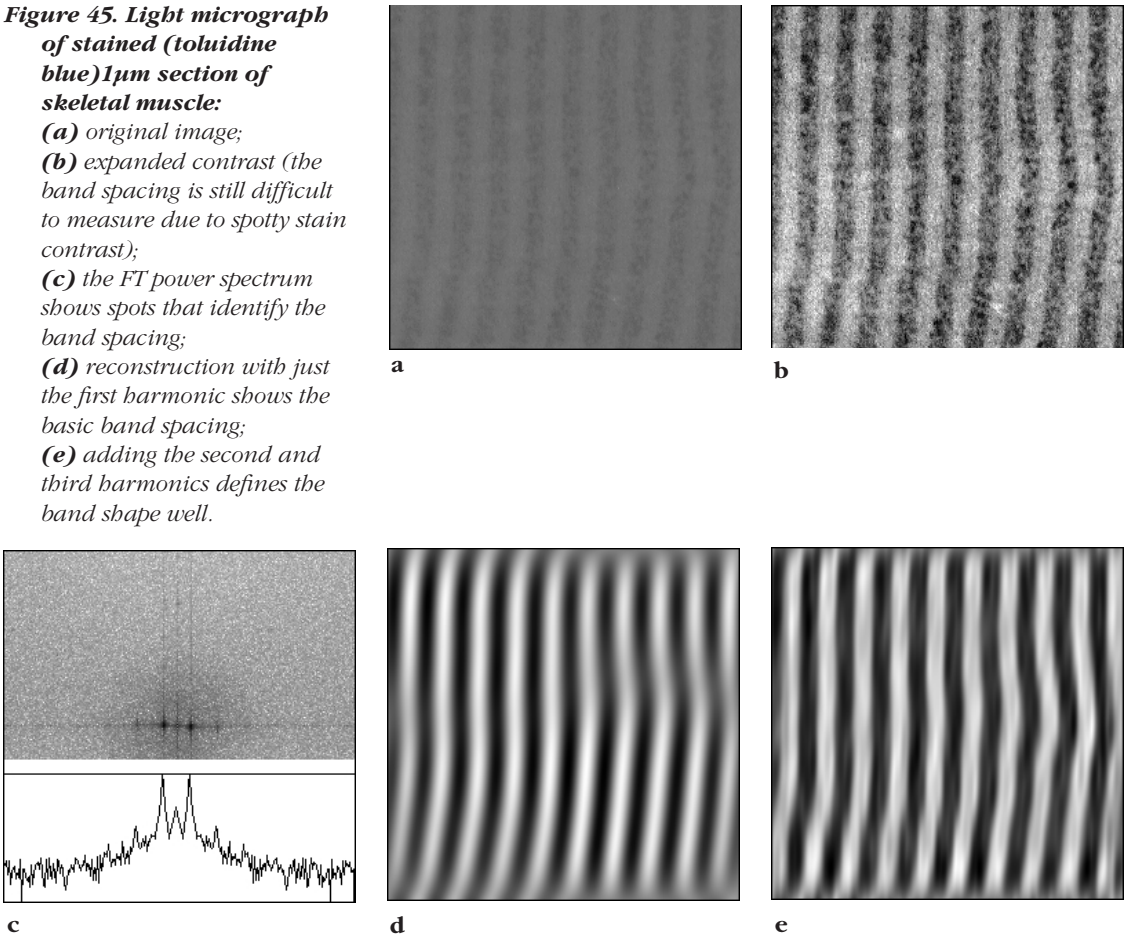


Figure 45. Light micrograph of stained (toluidine blue)1µm section of skeletal muscle: ***(a)*** *original image;* ***(b)*** *expanded contrast (the band spacing is still difficult to measure due to spotty stain contrast);* ***(c)*** *the FT power spectrum shows spots that identify the band spacing;* ***(d)*** *reconstruction with just the first harmonic shows the basic band spacing;* ***(e)*** *adding the second and third harmonics defines the band shape well.*

noisy background, which alters the local contrast making it more difficult to observe the details in the rather complex unit cell of this material. The Fourier transform image has peaks that correspond to the periodic structure. As noted before, this image is essentially the same as would be recorded photographically using the TEM to project the diffraction pattern of the specimen to the camera plane. Of course, retransforming the spatial-domain image from the photographed diffraction pattern is not possible because the phase information has been lost. In addition, more control over the Fourier transform display is possible because a log scale or other rule for converting magnitude to screen brightness can be selected.

A filter mask is constructed to select a small circular region around each of the periodic spots. This was done, as before, by using a top hat filter with size just large enough to cover the peaks. In this case the filter is used to keep the amplitude of terms in the peaks but reduce all of the other terms to zero, which removes the random or non-periodic noise, both the short-range (high-frequency) graininess and the gradual (low-frequency) variation in overall brightness. Retransforming this image produces a spatial-domain image which shows the lattice structure clearly.

Figure 47 shows an even more dramatic example. In the original image (a cross section of muscle myofibrils) it is practically impossible to discern the periodic structure due to the presence of noise. In isolated locations, a few of the fibrils can be seen to have a regular spacing and arrangement, but human observers do not easily see through the noise and variability to find this regularity. The Fourier transform image shows the peaks from the underlying regularity,

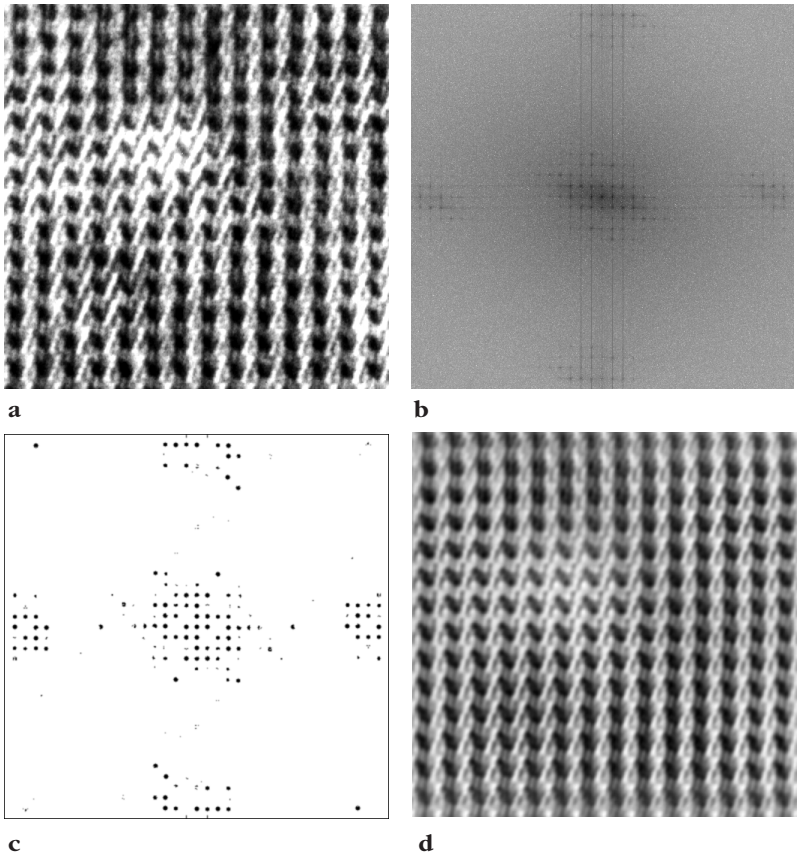


Figure 46. Transmission electron microscope lattice image of the mullite (ceramic): ***(a)*** *original;* ***(b)*** *Fourier transform power spectrum;* ***(c)*** *result of applying top hat filter to* ***b*** *to construct a filter;* ***(d)*** *retransformation of just the selected periodic information.*

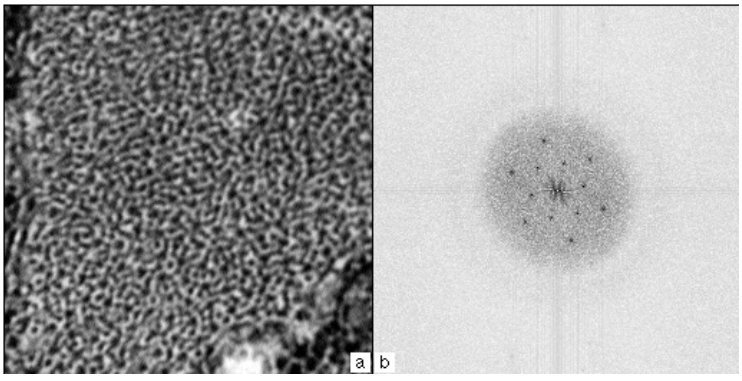


Figure 47. *Transmission electron microscope image of cross section of muscle myofibrils (left) and the frequency transform (right) (image courtesy Arlo Reeves, Dartmouth College).*

however. Selecting only these peak points in the magnitude image (with their original phase information) and reducing all other magnitude values to zero produces the result shown in **Figure 48**. The retransformed image clearly shows the sixfold symmetry expected for the myofibril structure. The inset shows an enlargement of this structure in even finer detail, with both the thick and thin filaments shown. The thin filaments, especially, cannot be seen clearly in the original image.

A caution is needed in using this type of filtering to extract periodic structures. It is possible to construct a mask that will eliminate real information from the image while keeping artefacts and noise.

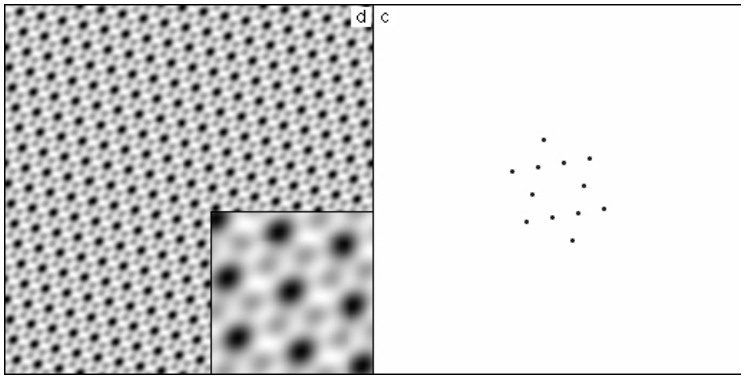


Figure 48. *Retransformation of* ***Figure 46*** *(left) with only the principal periodic peaks in the frequency transform (right). The points in the filter mask have been enlarged for visibility.*

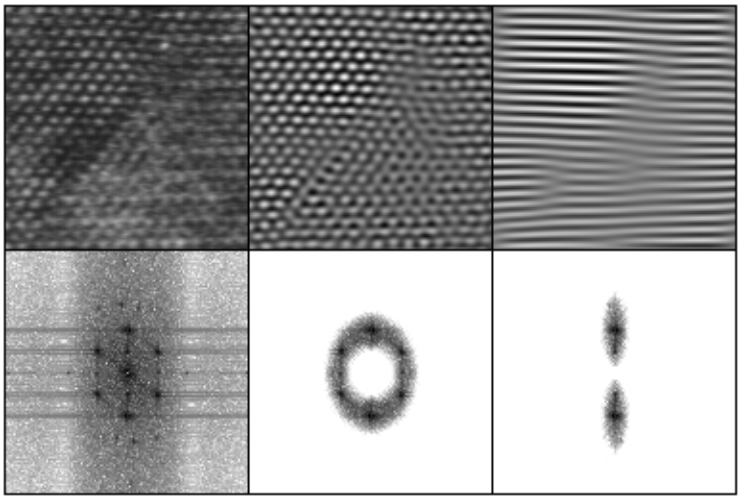


Figure 49. *Noisy high-resolution TEM image (left), and the results of applying an annular filter to select atomic spacings (center), and a slit filter to select vertical spacings only (right). Top images show the spatial domain, and bottom row shows frequency domain.*

Selecting points in the power spectrum with sixfold symmetry ensured that the filtered and retransformed spatial image would show that type of structure. This means that the critical step is the recognition and selection of the peaks in the Fourier transform image. Fortunately, many suitable tools are available for finding and isolating such points, because they are narrow peaks that rise above a gradually varying local background. The top hat filter illustrated previously is an example of this approach.

It is also possible to construct a filter to select a narrow range of spacings, such as the interatomic spacing in a high-resolution image. This annular filter makes it possible to enhance a selected periodic structure. **Figure 49** shows a high-resolution TEM image of an atomic lattice. Applying an annular filter that blocks both the low and high frequencies produces the result shown, in which the atom positions are more clearly defined. If the filter also selects a particular orientation (a slit or wedge filter), however, then the dislocation that is difficult to discern in the original image becomes clearly evident.

As for the case of removing periodic noise, a filter that selects periodic information and reveals periodic structure can often be designed by examining the Fourier transform power spectrum image itself to locate peaks. The mask or filter can be constructed either manually or automatically. In some cases, there is *a priori* information available (such as lattice spacings of crystalline specimens).

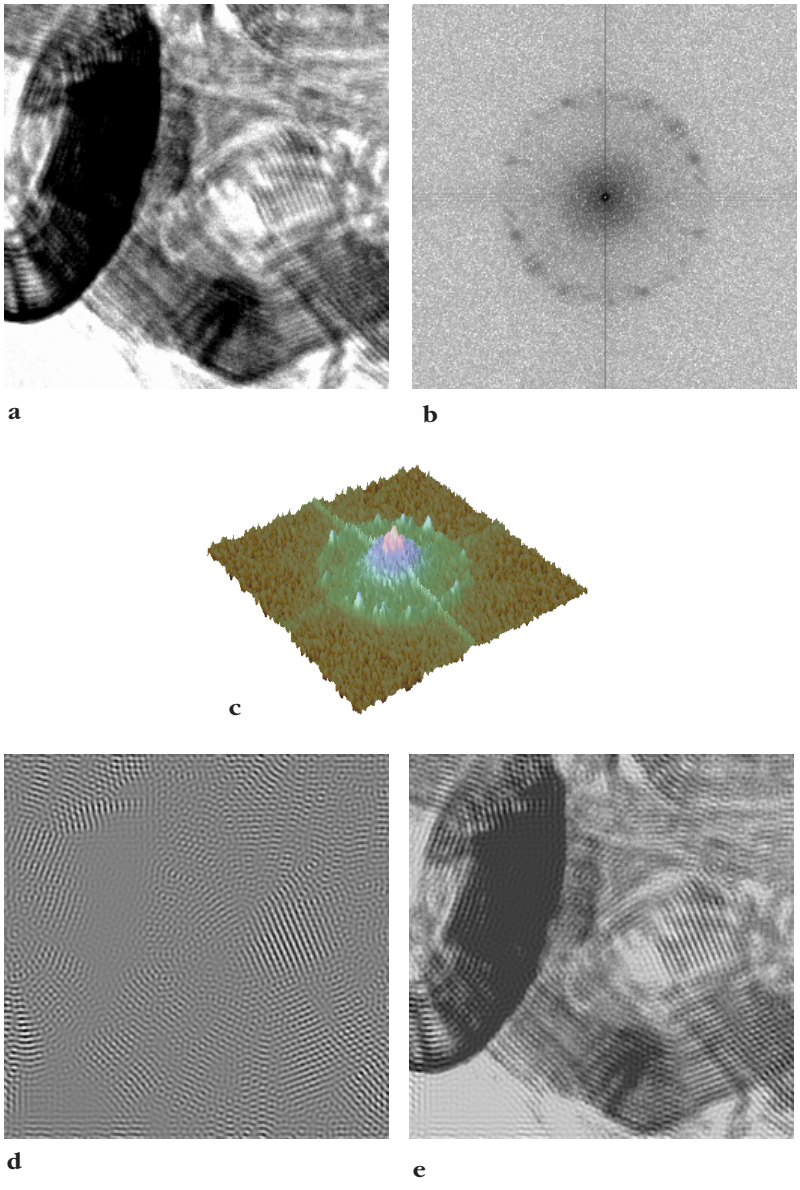


Figure 50. Transmission electron microscope (TEM) image of graphitized carbon ***(a): (b)*** *the FT power spectrum from image a showing the broken ring of spots corresponding to the 3.5 Angstrom plane spacing in the graphite;* ***(c)*** *the power spectrum plotted as a perspective drawing to emphasize the ring of peaks;* ***(d)*** *inverse transform of just the ring of spots;* ***(e)*** *image* ***d*** *added to the original image* ***a*** *to enhance the visibility of the lattice.*

In **Figure 50**, the structure (of graphitized carbon in particles used for tire manufacture) shows many atomic lattices in different orientations. In this case, the diffraction pattern shows the predominant atom plane spacing of 3.5 Ångstroms, in the form of a ring of spots. Applying an annular filter to select just that spacing, retransforming and adding the atom positions to the original image, enhances the visibility of the lattice structure.

The use of filtering in FT space is also useful for isolating structural information when the original image is not perfectly regular. **Figure 51** shows the same skeletal muscle tissue shown in **Figure 45**, but cross-sectioned and stained with uranyl acetate/lead citrate. The black dots are immunogold labeling for fast myosin, involved in muscle contraction. The spots are more or less regularly spaced because of the uniform diameters of the muscle fibers, but have no regular order to their arrangement. Sharpening the image using a top hat filter, as discussed above and in Chapter 4,

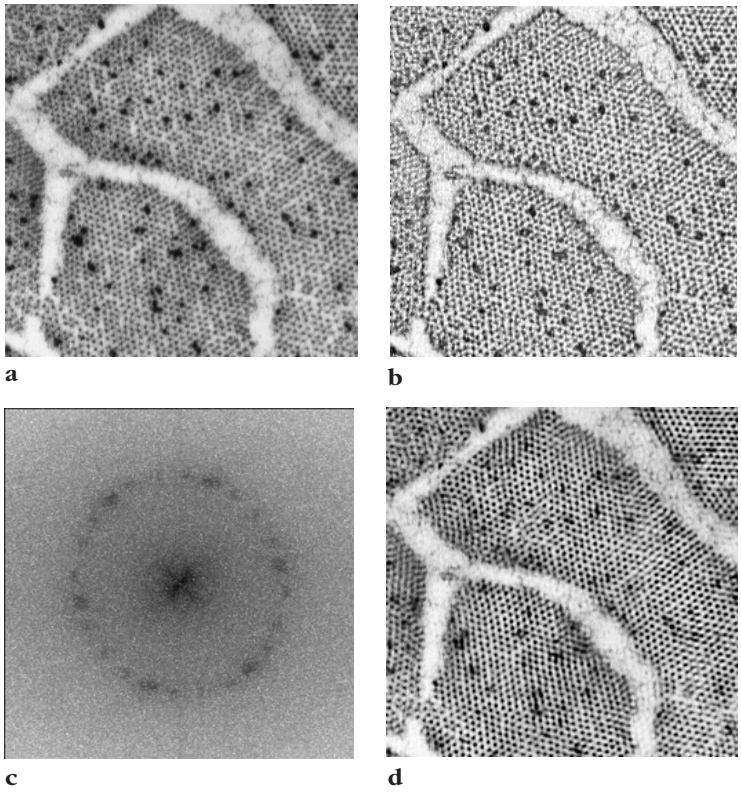


Figure 51. Light micrograph of cross section of the same skeletal muscle as in Figure 62, cross-sectioned and uranyl acetate/lead citrate stained. Black dots are immunogold labeling for fast myosin:
(a) *original;*
(b) *top hat filter applied to the image, as shown in Chapter 4;*
(c) *the power spectrum from the Fourier transform, showing a broken ring of spots corresponding to the average diameter of the fibers;*
(d) *retransforming the ring of spots with an annular filter that selects just the spacings of the gold particles and adding this back to the original image increases the contrast of the fibers.*

improves the visibility of the spots as compared to the original image. So does the use of an annular Fourier filter that selects just the ring of spots corresponding to the fiber diameter.

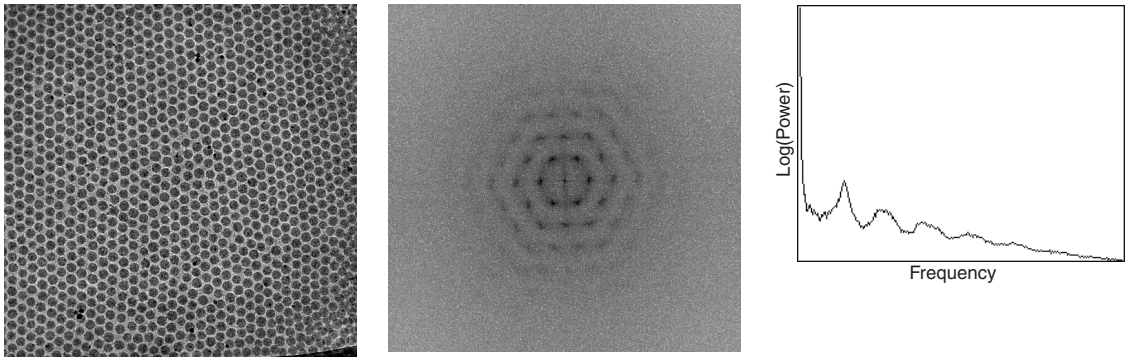
Similarly, measurement of "quasi-periodic" structures can also be performed in Fourier space. In **Figure 52**, the packed arrays of latex spheres are not perfectly regular, but the Fourier transform power spectrum shows the rings of nearest neighbors, second-nearest, third-nearest, and so forth. Measuring the spacing is simplified by using a plot of the circularly averaged power spectrum values, as shown.

Convolution and correlation

Fundamentals of convolution

One of the very common operations on images performed in the spatial domain is convolution, in which a kernel of numbers is multiplied by each pixel and its neighbors in a small region, the results summed, and the result placed in the original pixel location. This is applied to all of the pixels in the image. In all cases, the original pixel values are used in the multiplication and addition, and the new derived values are used to produce a new image, although as a practical matter of implementation the operation may be performed a few lines at a time, so that the new image ultimately replaces the old one.

This type of convolution is particularly common for the smoothing and derivative operations illustrated in Chapter 3 and 4. For instance, a simple smoothing kernel might contain the following values:



a **b** **c**

Figure 52. Quasi-periodic arrays of latex spheres:
(a) *original image;*
(b) *Fourier transform power spectrum;*
(c) *circularly averaged plot of the power spectrum, in which peak locations identify the distances of first, second-, etc., neighbors.*

1/16	2/16	1/16
2/16	4/16	2/16
1/16	2/16	1/16

In practice, the fastest implementation would be to multiply the pixel and its 8 immediate neighbors by the integers 1, 2, or 4, sum the products, then divide the total by 16. In this case, using integers that are powers of 2 allows the math to be extremely fast (involving only bit shifting), and the small size of the kernel (3×3) makes the application of the smoothing algorithm on the spatial-domain image very fast, as well.

Many spatial-domain kernels are used, including kernels that apply Gaussian smoothing (to reduce noise), take first derivatives (for instance, to locate edges), and take second derivatives (for instance, the Laplacian, which is a non-directional operator that acts as a high-pass filter to sharpen points and lines). They are usually presented as a set of integers, although greater accuracy is attainable with floating point numbers, with it understood that there is a divisor (usually equal to the sum of all the positive values) that normalizes the result. Some of these operators may be significantly larger than the 3×3 example shown previously, involving the adding together of the weighted sum of neighbors in a much larger region that is usually, but not necessarily, square.

Applying a large kernel takes time. **Figure 53** illustrates the process graphically for a single placement of the kernel. Even with very fast computers and with careful coding of the process to carry out additions and multiplications in the most efficient order, performing the operation with a 25×25 kernel on a 1024×1024 image would require a significant amount of time (and even larger kernels and images are often encountered). Though it can be speeded up somewhat by the use of special hardware, such as a pipelined array processor, a special-purpose investment is required. A similar hardware investment can be used to speed up the Fourier transform. Our interest here is in the algorithms, instead of their implementation.

For any computer-based system, increasing the kernel size eventually reaches a point at which it is more efficient to perform the operation in the Fourier domain. The time needed to perform the FFT transformation from the spatial domain to the frequency domain and back is more than balanced by the speed with which the convolution can be carried out. If any other reasons are needed to perform

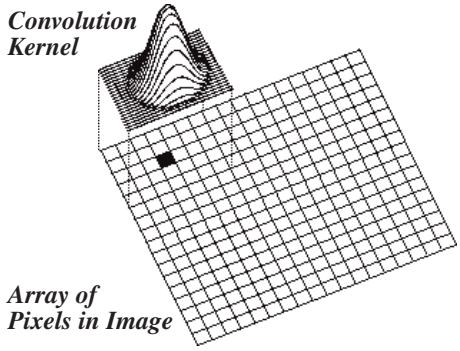


Figure 53. *Illustration of applying a convolution kernel to an image in the spatial domain (illustration courtesy Arlo Reeves, Dartmouth College).*

the transformation to the frequency-domain representation of the image, then even small kernels can be most efficiently applied there.

This is because the equivalent operation to spatial-domain convolution is a single multiplication of each pixel in the magnitude image by the corresponding pixel in a transform of the kernel. The transform of the kernel can be obtained and stored beforehand just as the kernel is stored. If the kernel is smaller than the image, it is padded with zeroes to the full image size. Convolution in the spatial domain is exactly equivalent to multiplication in the frequency domain. Using the notation presented before, in which the image is a function *f(x,y)* and the kernel is *g(x,y),* we describe the convolution operation in which the kernel is positioned everywhere on the image and multiplied by it as

$$g(x,y) * f(x,y) = \iint (f(\alpha,\beta) \cdot g(x-\alpha, y-\beta) d\alpha \, d\beta \tag{9}$$

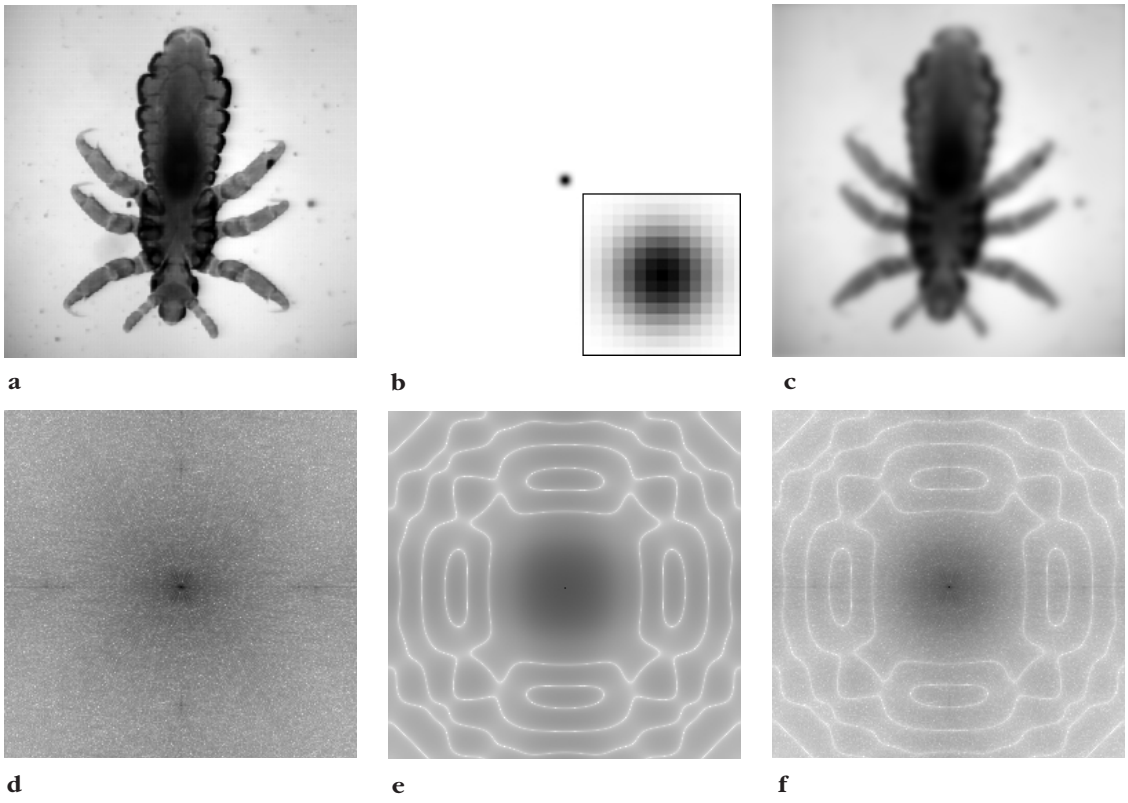
where α and β are dummy variables for the integration, the range of which is across the entire image, and the symbol * indicates convolution. If the Fourier transforms of $f(x,y)$ and $g(x,y)$ are $F(u,v)$ and $G(u,v)$, respectively, then the convolution operation in the Fourier domain is simple point-by-point multiplication, or

$$g(x,y) * f(x,y) \Leftrightarrow G(u,v)F(u,v) \tag{10}$$

A few practical differences exist between the two operations. The usual application of a kernel in the spatial domain avoids the edge pixels (those nearer to the edge than the half-width of the kernel), since their neighbors do not exist. As a practical alternative, a different kernel that is one-sided and has different weights can be applied near edges, or the edge can be considered to be a mirror. In transforming the image to the frequency domain, however, the assumption is made that the image wraps around at edges, so that the left edge is contiguous with the right and the top edge is contiguous with the bottom. Applying a convolution by multiplying in the frequency domain is equivalent to addressing pixels in this same wraparound manner when applying the kernel to the spatial image. It will usually produce some artefacts at the edges. The most common solution for this problem is to embed the image of interest in a larger one in which the borders are either filled with the mean brightness value of the image, or smoothly interpolated from the edge values.

Figure 54 shows the equivalence of convolution in the spatial domain and multiplication in the frequency domain, for the case of a smoothing kernel. The kernel, a Gaussian filter with standard deviation of 2.0 pixels, is shown as an array of grey-scale values, along with its transform. Applying the kernel to the image in the spatial domain produces the result shown in the example.

Figure 54. Smoothing by applying a large kernel in the spatial domain and convolution in the frequency domain:
(a) *original image;*
(b) *smoothing kernel (Gaussian, standard deviation = 2.0 pixels), with enlargement to show pixel detail;*
(c) *smoothed image produced by spatial convolution with kernel or inverse Fourier transform of* ***e****;*
(d) *Fourier transform of* ***a****;*
(e) *Fourier transform of* ***b****;*
(f) *product of* ***d*** *and* ***e****.*



Multiplying the kernel transform by the image transform produces the frequency-domain image with the power spectrum shown. Retransforming this image produces the identical result to the spatial-domain operation.

Notice that the equivalence of frequency-domain multiplication to spatial-domain convolution is restricted to multiplicative filters, which are also known as "linear" filters. Other neighborhood operations, such as rank filtering (saving the brightest, darkest, or median brightness value in a neighborhood) and histogram modification (e.g., local adaptive equalization), are nonlinear and have no frequency-domain equivalent.

Imaging system characteristics

Convolution can also be used as a tool to understand how imaging systems alter or degrade images. For example, the blurring introduced by imperfect lenses can be described by a function $H(u,v)$ which is multiplied by the frequency transform of the image (**Figure 55**). The operation of physical optics is readily modeled in the frequency domain. Sometimes it is possible to determine the separate characteristics of each component of the system; often it is not. In some cases,

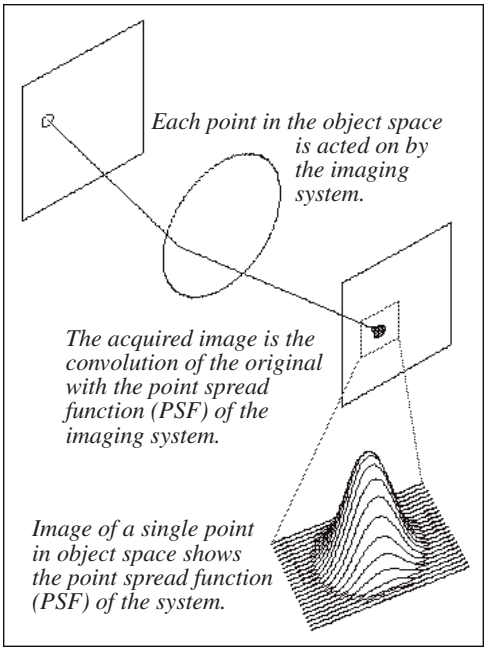


Figure 55. *System characteristics introduce a point spread function into the acquired image (illustration courtesy Arlo Reeves, Dartmouth College).*

determining the point-spread function of the system (the degree to which a perfect point in the object plane is blurred in the image plane) may make it possible to sharpen the image by removing some of the blur. This is done by dividing the transform of the point-spread image by $H(u,v)$.

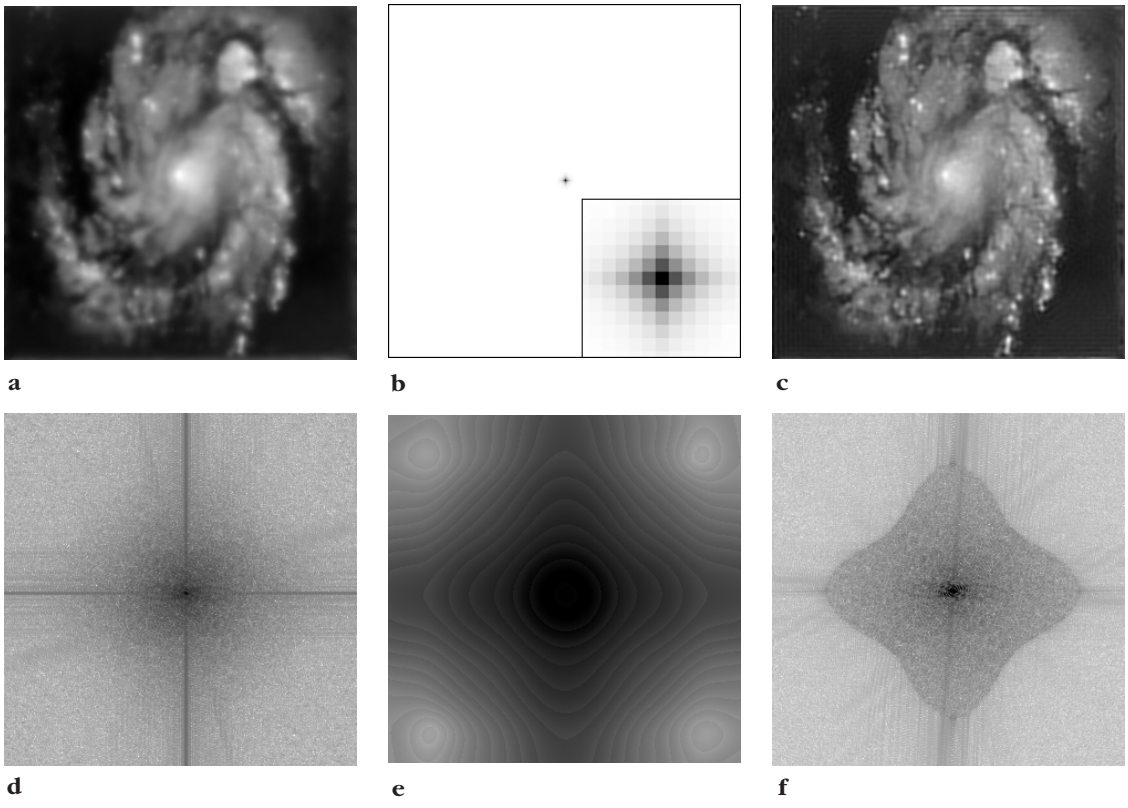
To illustrate this sharpening we can use the one of the best-known examples of correction of an out-of-focus condition due to an imperfect imaging system. As originally deployed, the Hubble telescope had an incorrectly figured main mirror that produced poorly focused images. Eventually a compensating optical device was installed in the imaging system to correct most of the defect, but even with the original telescope it was possible to obtain high resolution results by using deconvolution with the known point spread function (PSF). The corrected optical package, however, restored much of the telescope's light-gathering power, which was severely reduced due to the incorrect mirror curvature.

In this particular instance, it was possible to calculate the PSF from available measurement data on the mirror, but in many astronomical situations, the problem is simplified because the point spread function can be measured by examining the image of a single star, which is effectively a point as seen from earth. This requires a considerable grey-scale depth to the image, but in astronomy, cooled cameras often deliver 12 or 14 bits of information, which is also important for obtaining enough precision to perform the deconvolution.

Figure 56 shows the result of this method. The original, blurred image is sharpened by dividing its Fourier transform by that of the measured PSF, and the resulting inverse transform shows a considerably sharpened image.

For deconvolution, we divide the complex frequency-domain image from the out-of-focus test pattern by the image for the point spread function. This is complex division, performed by dividing the magnitude values and subtracting the phase values. One of the problems with division is that division by very small values can cause numeric overflow problems, and the Fourier transform of a symmetrical and well behaved point spread function often contains zero values. The usual

Figure 56. Hubble telescope image sharpening:
(a) *original;*
(b) *measured point spread function, with enlargement to show pixel detail;*
(c) *deconvolution result obtained as inverse transform of* ***f****;*
(d) *Fourier transform of* ***a****;*
(e) *Fourier transform of* ***b****;*
(f) ***d*** *divided by* ***e****.*



solution to this problem is apodization to restrict the division operation to those pixels in the complex transform images that will not cause overflow.

Deconvolution of image blur, which may arise from out-of-focus optics, motion, the size and arrangement of transistors in the camera chip, insufficient bandwidth in the electronics, or other causes, is an imperfect science. A mathematically optimum procedure neither exists, nor is there a proof that one exists in all cases. The practical techniques that are used have been developed under a variety of assumptions and apply to many real situations, producing impressive improvements in image quality. The results, although good, represent trade-offs between different limitations, one of which is the time required for the computations.

Deconvolution is discussed here in terms of grey scale images. In most cases, color images need to be separated into discrete color channels that correspond to the physics of the acquisition device, usually red, green and blue, and each one deconvolved separately. It is usual to find that the point spread function is different for each channel, especially if a single-chip camera with a Bayer pattern color filter has been used for acquisition (see Chapter 2).

It is important to always start with the best image possible. This means obtaining the best focus, least motion blur, etc., that you can achieve. Deconvolution is never as good a solution as

correcting the source of the problem beforehand. Next, capture the image with the best possible range of contrast, from nearly full white to nearly full black without clipping. The tonal or gray scale range should have good precision and a wide dynamic range. Eight-bit images from uncooled digital cameras are marginal, and poorer images from video cameras usually unacceptable unless special procedures such as averaging multiple frames are used. It is particularly important to have high precision and bit depth for the point spread function, whether it is obtained by measurement or by calculation.

Finally, random pixel noise (speckle) must be minimized. Noise in either the acquired image or (especially) in the point spread function is significantly amplified in the deconvolution process and will dominate the result if it is too great. Long exposures and image averaging may be useful in some cases. The usual description of the magnitude of image noise is a signal-to-noise ratio expressed in decibels. This is defined in terms of the standard deviation of values in the blurred image and in the noise (which of course may not be known).

$$SNR[\text{dB}] = 10 \cdot \log_{10}\left(\frac{\sigma_{image}}{\sigma_{noise}}\right) \tag{11}$$

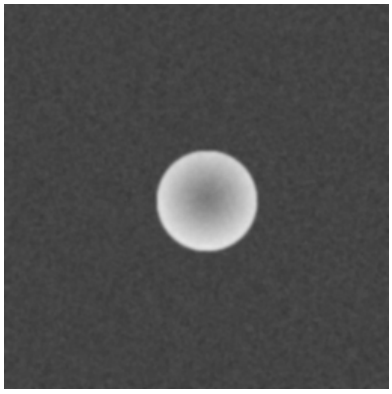
When the signal-to-noise ratio is in the range of 40–50 dB, the noise is, practically speaking, invisible in the image and has a minimal effect on deconvolution. On the other hand, a low signal-to-noise ratio of 10–20 dB makes the noise so prominent that deconvolution becomes quite impractical.

The ideal and simplest form of deconvolution is to measure (or in a few cases calculate from known optical parameters) the point spread function of the system. Computing the Fourier transform of the blurred image and that of the PSF, dividing the second into the first, and performing an inverse transform, produces the deconvolved result as shown previously. The key requirement is that the blur due to the imaging system is assumed to be the same everywhere in the image, which is a good assumption for telescope images (**Figure 56**). When some portion of the blur is due to the passage of light through the sample itself, as occurs in thick sections examined in the light microscope, the blur can vary from point to point and this method becomes less useful.

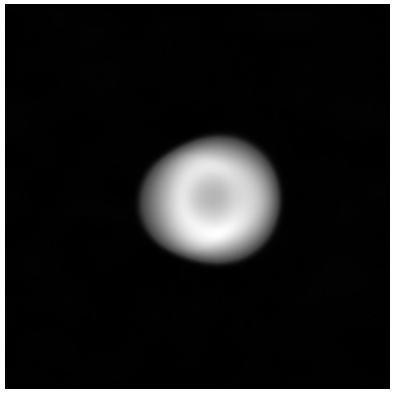
An indirect measurement of the PSF can be accomplished by capturing an image of a precisely known object using the same optical setup as that used for the real image. Dividing the Fourier transform of the image of the object by that of the ideal shape, and inverse transforming the result, produces the system PSF, which can then be used to deconvolve images obtained with the optical system.

In some microscope situations, the insertion of fluorescing microbeads, or even an image of a small dust particle on the slide, may be useful as an estimate of the PSF. In the atomic force microscope, a direct measurement of the PSF may be accomplished by scanning an image of a known shape, usually a circular test pattern produced expressly for this purpose by the same methods used to etch integrated circuits. As shown in **Figure 57**, if the resulting image is deconvolved by dividing its Fourier transform by that for the ideal shape, the result is an image of the point spread function, which in this case corresponds to the shape of the scanning tip. This shape can then be used to deconvolve other images, at least until the tip is further damaged or replaced.

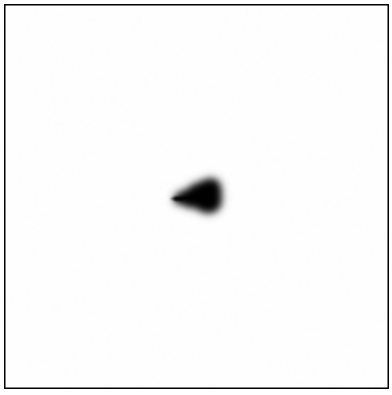
If an image contains many edges oriented in different directions, and the actual edge shape is known (ideally a perfectly sharp knife-edge transition in grey level), then the PSF can be determined at least in principle by measuring the actual transition across each edge, and combining the various profiles to form a PSF image. In practice this usually requires assuming that the function is symmetrical and of known shape (usually a Gaussian as a convenient approximation to the central portion of the Airy disk produced by real optics), so that measurements on a few edges can be generalized to an entire two-dimensional PSF image.



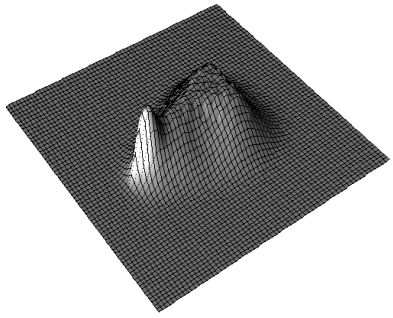
a



b



c



d

Figure 57. Measuring the PSF of an atomic force microscope: ***(a)*** *ideal image of a circular test pattern;* ***(b)*** *measured image using an imperfect tip;* ***(c)*** *deconvolution of* ***b*** *by* ***a****, producing an image of the tip;* ***(d)*** *perspective view of the shape in* ***c****.*

If it is not possible to obtain a PSF by measurement, it may be possible to calculate a useful approximate one based on assumptions such as a Gaussian blur function or straight line motion. Real optical defocusing does not produce an ideal Gaussian blur, but it is not necessary in many cases to have the exact PSF, just a sufficiently useful approximation to remove most of the blur by deconvolution. Sometimes important insight into the nature of the PSF can be obtained by examining the Fourier transform of the blurred image. The presence of zeroes (or near-zero values) in lines, arcs, etc., is a powerful clue to the nature of the blur. Straight lines indicate motion blur, in a direction orthogonal to the lines. Arcs indicate non-Gaussian blurring (e.g., by a uniform disk). These can be used to estimate a PSF which can be used to perform a deconvolution, perhaps iterating the estimated PSF either automatically using one of the convergence methods discussed below, or by interactive selection, to reach a satisfactory result.

The blur produced by purely optical effects is frequently uniform in all directions, although astigmatism can modify this; however, the tip shape in scanned probe microscopes can produce arbitrarily shaped point spread functions. In many cameras the shape and spacing of the transistors produces different blur magnitudes in the horizontal and vertical directions, and the effects of the color filter pattern used can also alter the shape of the PSF. In any scanned image acquisition, the electronic parameters of the amplifiers used can produce different amounts of blur in the fast scan direction (generally horizontal) as compared to the slow scan direction (vertical). Time constants in phosphors, amplifiers or other components can also produce asymmetrical blurs ("comet tails") in the output signal.

Noise and Wiener deconvolution

If there is significant noise content in the image to be sharpened or, worse yet, in the measured PSF, it can exacerbate the numerical precision and overflow problems and greatly degrades the resulting inverse transform. Removal of more than a portion of the blurring in a real image is almost never possible, but of course, some situations exist in which even a small improvement may be of considerable practical importance.

Division by the frequency transform of the blur is referred to as an inverse filter. Using the notation introduced previously, it can be written as

$$F(u,v) \approx \left[\frac{1}{H(u,v)}\right] G(u,v) \tag{12}$$

If the presence of noise in the blurred image prevents satisfactory deconvolution by simply dividing the Fourier transforms, then it may be practical to perform a Wiener deconvolution. Instead of calculating the deblurred image by dividing the Fourier transform of the original image by that of the blur function, a scalar value is used to increase the denominator. Theoretically, the additive factor K is dependent on the statistical properties of the images and their relative noise contents, but in practice these are not usually known and so the additive constant is typically treated as an adjustable parameter that controls the tradeoff between sharpening and noise.

$$F(u,v) \approx \left[\frac{1}{H(u,v)}\right] \cdot \left[\frac{|H(u,v)|^2}{|H(u,v)|^2 + K}\right] \cdot G(u,v) \tag{13}$$

Figure 58 shows an example of the effect of additive random noise on a image with out-of-focus blur. In the absence of any noise, a direct deconvolution using a Gaussian model for the PSF produces useful results. The addition of random Gaussian noise (SNR = 15 dB) to the image results in a deconvolution using the same PSF that is dominated by noise and ringing that obscures the features. Wiener filtering (**Figure 59**) reduces this problem and allows adjustment which trades off the sharpness of the restoration (higher K values leave more blurring) against the amount of noise (higher K values reduce the noise). Because of the low signal-to-noise ratio, it is not possible in this case to recover all the internal details of the original image.

a **b** **c**

Figure 58. Effect of noise on deconvolution: (a) *severely blurred image of the bug in* ***Figure 54*** *with added random noise;* ***(b)*** *deconvolution that can be achieved in the absence of noise;* ***(c)*** *deconvolution with noise present.*

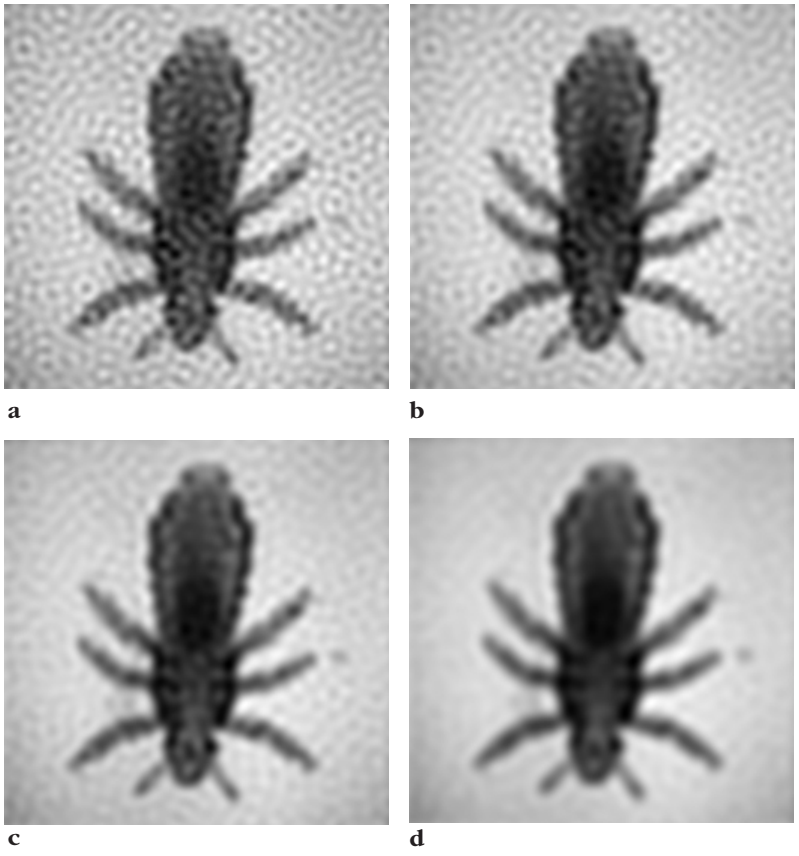


Figure 59. Wiener filtering of the image in Figure 58a: (a–d) *increasing the empirical K value reduces noise at the expense of sharpness.*

The Wiener deconvolution has a long history as a practical method for achieving useful, if not perfect blur removal. Another approach of long standing is the Van Cittert iterative technique. This method has apparently been discovered many times and is also known as a Bially or Landweber iteration. Instead of trying to remove the entire blur in one step, a series of iterative removal steps are performed. In the limit, this iteration would reach the same result as an ideal deconvolution, but instead it is terminated before convergence, resulting in a (partially) deblurred image that does not exhibit an unacceptable noise level. The quality of the results is generally similar to that of the Wiener method.

Many other methods are used, and most of them are much more complicated mathematically (and require much more computation and time). Many of them are iterative, trying to find the best form of the blur function and the deblurred image. Typically this involves trying to solve a very large number of simultaneous equations, which may be over- or under-determined. The techniques for such solutions occupy a significant mathematical literature, and are discussed further in Chapter 11. One of the issues is how to efficiently guide the iteration and how to determine when to terminate it. Another related question is the best measure of image quality, which in different implementations may be based on Bayesian statistics, maximization of entropy, etc. The calculations may be performed in either the frequency or the spatial domain. Even a small PSF in the spatial domain becomes as large as the entire image in the frequency domain, and the number of simultaneous equations is the number of pixels in the image.

Most of these methods are highly specific to particular types of applications, and depend on the *a priori* information that can be supplied, usually in the form of constraints on the solution of the equations. For example, a commonly used constraint is that negative values for pixels have no

physical meaning and so values are not permitted to become negative. If more information on the image, such as the presence of substantial background (as in astronomical or fluorescence images), or known specific pixel values that are permitted, can be incorporated it improves the quality of the result and the efficiency of reaching it. The details of these methods are far beyond the scope of this text, but a very clear and comprehensive revue and comparison can be found in Lagendijk and Biemond (1991).

Another area of current interest is the deconvolution of multiple images that are related to each other. This includes multiple channel images (e.g., different wavelengths or colors) of the same scene, or a series of images from parallel closely spaced planes in semi-transparent specimens (e.g., a series of focal planes in the light microscope). Information from one channel or plane can be used to guide the deconvolution of another.

If the blur is not known *a priori*, it can often be estimated from the power spectrum or by trial and error to find an optimum (or at least useful) result. A recent development in deconvolution that has great efficiency because it is not an iterative method has been published by Carasso (2001). Analysis of the shape of radial profile of the values in the Fourier transform of the blurred image (avoiding the regions dominated by noise) allows constructing an approximate PSF that can be used for deconvolution. This method cannot be used for all types of blur functions (particularly motion blur) but produces very good results for those situations where it is applicable.

An important area for these image restoration techniques is deblurring the images formed by optical sectioning. This is the technique in which a series of images are recorded from different depths of focus in a semitransparent specimen using a light microscope. The passage of light through the overlying layers of the specimen cause a blurring that adversely affects the sharpness and contrast of the images, preventing their being assembled into a three-dimensional stack for visualization and measurement of the three-dimensional structures present.

The confocal light microscope overcomes some of these problems by rejecting light from points away from the point of focus, which improves the contrast of the images (it also reduces the depth of field of the optics, producing higher resolution in the depth axis, but this is important only at the highest magnifications); but the scattering and diffraction of light by the upper layers of the specimen still degrade the image resolution.

In principle, the images of the upper layers contain information that can be used to deconvolute those from below. This would allow sharpening of those images. The entire process is iterative and highly computer-intensive, more so because the blurring of each point on each images may be different from other points. In practice, it is usual to make some assumptions about the blurring and noise content of the images which are used as global averages for a given specimen or for a given optical setup.

Even with these assumptions, the computations are still intensive and iterative. A considerable theoretical and limited practical literature has been published in this field (Carrington, 1990; Holmes et al., 1991; Monck et al., 1992; Joshi and Miller, 1993; Richardson, 1972; Snyder et al., 1992) A review of several of the leading methods can be found in (Van Kempen et al., 1997). The examples shown deal only with idealized structures and averaging assumptions about the noise characteristics of the images and the point spread function of the microscope, which suggests that restoration of real images will not be as good as the examples shown. Similar concerns and methods can in principle be applied to other *in situ*, three-dimensional imaging techniques such as tomography and seismic imagery.

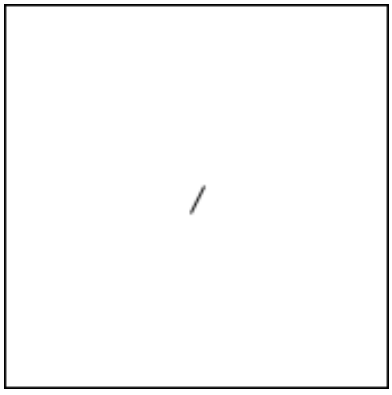
Motion blur

Additional defects besides out-of-focus optics can be corrected by deconvolution as well. These operations are not always performed in the frequency domain, but the basic understanding of the process of removing the blur convolution imposed by the system is most clearly illustrated there. One of the most common defects is blur caused by motion. This is rarely a problem in microscopy applications, but can be very important in remote sensing, in which light levels are low and the exposure time must be long enough for significant camera motion to occur with respect to the scene. Fortunately, in most of these circumstances the amount and direction of motion is known. That makes it possible to draw a line in the spatial domain that defines the blur. The frequency transform of this line is then divided into the transform of the blurred image. Retransforming the resulting image restores the sharp result. **Figure 60** illustrates the method.

It is important to note the similarity and the difference between this example and the removal of out-of-focus blur. Both involve dividing the transform of the blurred image by that of the defect. This follows directly from the equation presented for convolution, in which the transform of the convolved image is the product of those from the original image and the defect. The major difference is that in the motion blur case we can sometimes calculate the exact blurring vector to be removed, while in the out-of-focus blur case we must estimate the blur from an actual image. This estimation introduces unavoidable noise in the image, which is greatly magnified by the division.

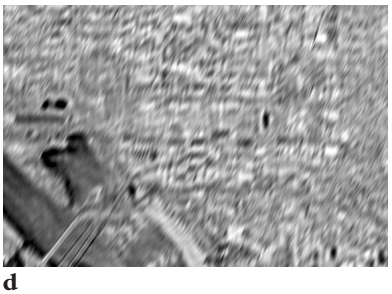
As in the example shown previously for out-of-focus blur, the use of a Wiener deconvolution can partially alleviate the effects of noise. In the example of **Figure 61**, the motion blur can be estimated from the orientation and spacing of the lines of zero (or near zero) values in the power spectrum of the Fourier transform of the blurred image. Using this to determine the motion vector as the point spread function allows performing a deconvolution. Increasing the Wiener K factor controls the tradeoff between image sharpness and noise as shown in **Figure 62**. This factor is usually treated as an empirical adjustment. Human observers generally prefer an image that is noisier but less blurred than a theoretical K value based on the noise content delivers.

a



b

c



d

Figure 60. Removal of motion blur: ***(a)*** *original image (the Mall and tidal basin in Washington, D.C.);* ***(b)*** *motion vector based on airplane speed, direction, and exposure time;* ***(c)*** *deconvolution result;* ***(d)*** *deconvolution result using 8-bit images and single-precision arithmetic.*

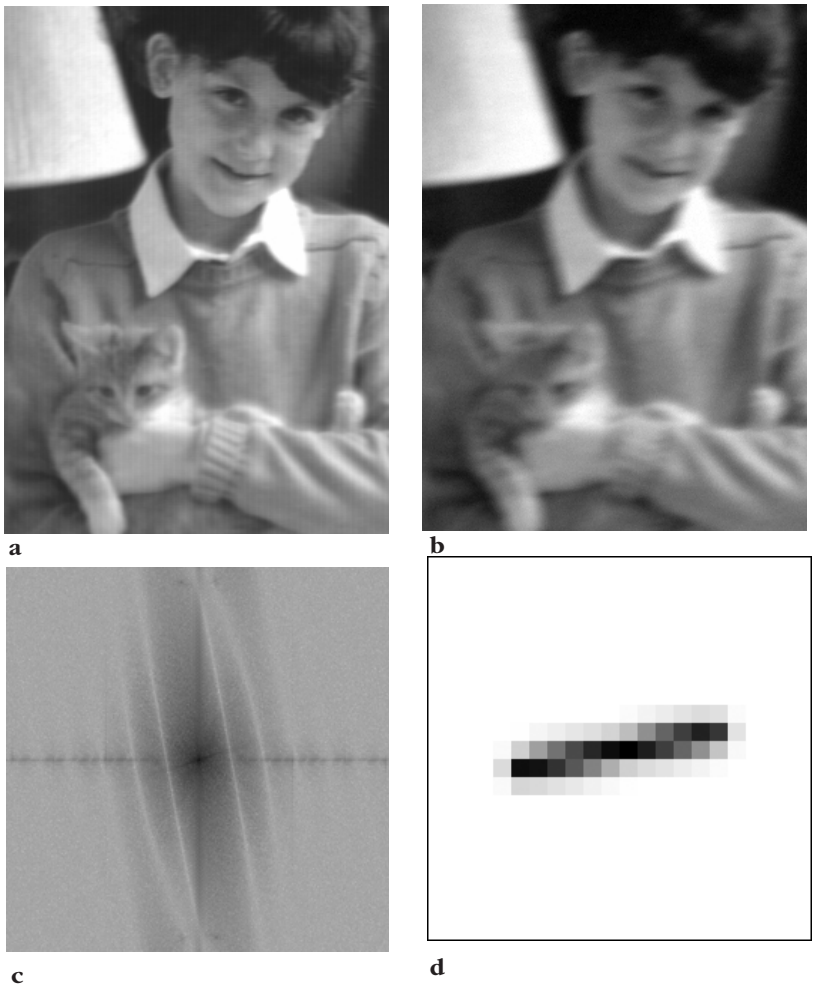


Figure 61. Motion blur: ***(a)*** *unblurred image;* ***(b)*** *uniform motion blur;* ***(c)*** *power spectrum of the Fourier transform of* ***b****;* ***(d)*** *motion vector shown as a point spread function (enlarged to show pixels).*

Increasing the value of the constant K in the denominator of equation 13 reduces the magnitude of all terms in the complex Fourier transform that is inverted to obtain the restored image, but it reduces terms that were originally small more than it does large ones. These terms are the ones most likely to be dominated by random noise. One effect of this reduction is the elimination of more of the small terms by apodization. **Figure 63** shows the terms eliminated in the Wiener filtered restoration of the image (compare it to the FT power spectrum of the original image in **Figure 61c**).

Additional limitations arise from the finite internal precision in computer implementation of the Fourier transform and the storage of the complex frequency-transform image. Programs may use either floating-point or fixed-point notation, the latter requiring less storage space and offering somewhat greater speed; double precision internal values give the best results because the minimum magnitude values that can be accurately recorded generally occur at higher frequencies, controlling the appearance of sharp edges in the spatial-domain image. Too small a precision in either the image or the calculation limits the ability to sharpen images by removing degrading blur due to the imaging system. To illustrate the problems introduced by images with insufficient precision, in **Figure 60d**, the original image was reduced from a bit depth of 12 (more than 4000 grey scale values obtained by scanning a 35-mm film negative) to 6 bits (the performance of a typical video camera). Note the vastly inferior (even useless) quality of the results.

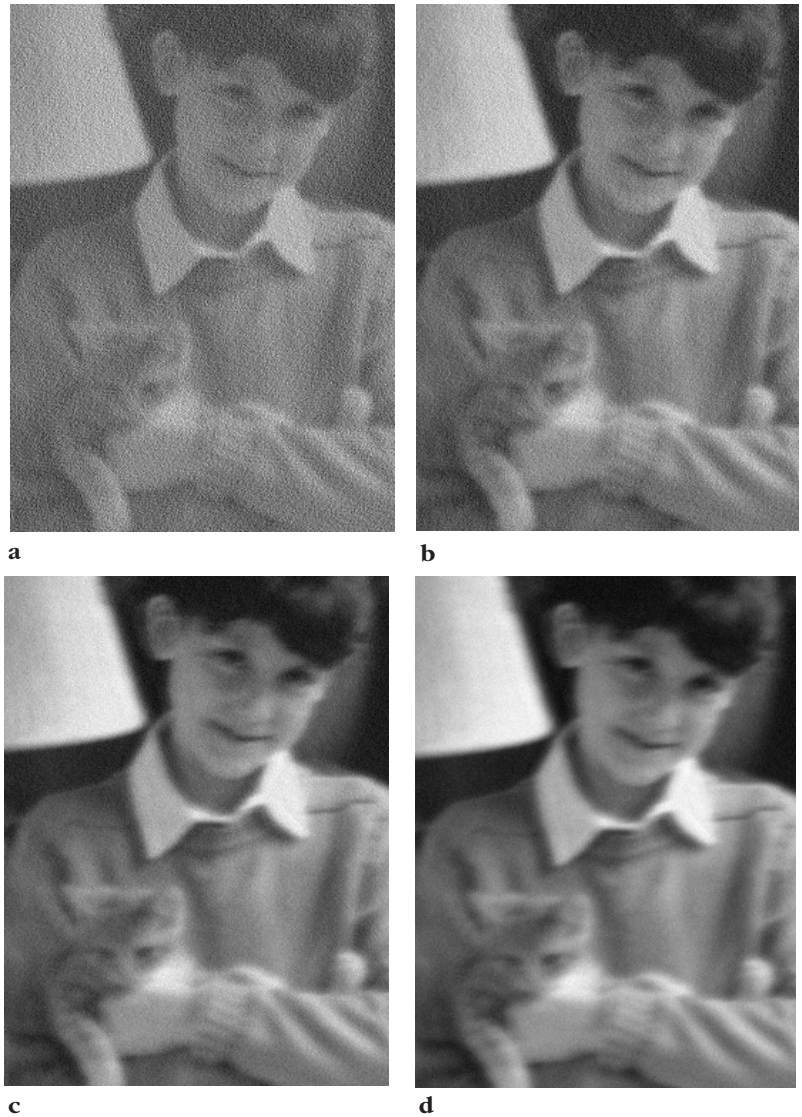
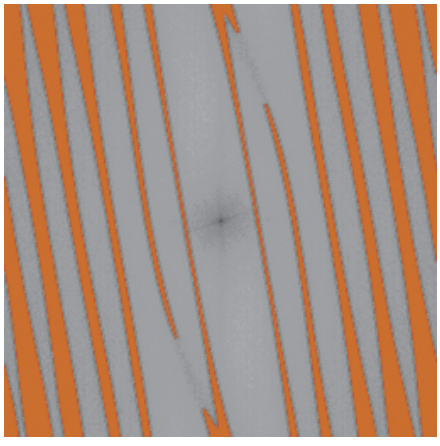


Figure 62. ***Deconvolution of the image in Figure 61b:*** ***(a)*** *Wiener constant K set to zero;* ***(b–d)*** *increasing K values covering the range from too noisy to too blurred.*

Figure 63. *Power spectrum of the image in* ***Figure 62*** *showing in red the values eliminated from the restored image by apodization (compare with* ***Figure 61c****).*



Template matching and correlation

Closely related to the spatial-domain application of a kernel for smoothing, derivatives, etc. is the idea of template matching or cross-correlation. In this case, a target pattern is shifted to every location in the image, the values are multiplied by the pixels that are overlaid, and the total is stored at that position to form an image showing where regions identical or similar to the target are located. **Figure 64** illustrates this process. The multiplication and summation process is identical to convolution, except that the target is rotated 180° first, so that the upper left corner value in the target pattern is multiplied by the lower right value in the neighborhood on the image and so forth. When the process is performed in frequency space, this amounts to a phase shift of the Fourier transform values.

This method is used in many contexts to locate features within images. One is searching reconnaissance images for particular objects such as vehicles. Tracking the motion of hurricanes in a series of weather satellite images or cells moving on a microscope slide can also use this approach. Modified to deal optimally with binary images, it can be used to find letters in text. When the target is a pattern of pixel brightness values from one image in a stereo pair and the searched image is the second image from the pair, the method can be used to perform fusion (locating matching points in the two images) to measure parallax and calculate range, as shown in Chapter 13.

For continuous two-dimensional functions, the cross-correlation image is calculated as

$$c(i,j) = \iint f(x,y)g(x-i,y-j)dx\,dy \tag{14}$$

Replacing the integrals by finite sums over the dimensions of the image gives **Equation 15**. In order to normalize the result of this template matching or correlation without the absolute brightness value of the region of the image biasing the results, the operation in the spatial domain is usually calculated as the sum of the products of the pixel brightnesses divided by their geometric mean.

$$\frac{\sum_{i,j} f_{x+i,y+j} \cdot g_{i,j}}{\sqrt{\sum_{i,j} f^2_{x+i,y+j} \cdot \sum_{i,j} g^2_{i,j}}} \tag{15}$$

When the dimensions of the summation are large, this is a slow and inefficient process compared to the equivalent operation in frequency space. The frequency-space operation is simply

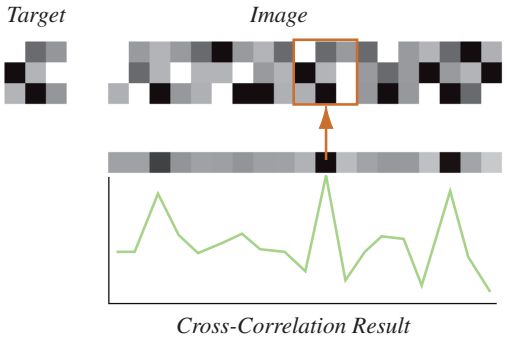


Figure 64. *Cross-correlation matches a pattern of grey scale values to many points in the target image to find the location with the best match. In the tiny example shown here, the 3 × 3 target is applied to every possible position in the image to produce the line of values shown. The marked location is most similar (but not identical) to the target.*

$$C(u,v) = F(u,v)G^*(u,v) \tag{16}$$

where * indicates the complex conjugate of the function values. The complex conjugate of the pixel values affects only the phase of the complex values. The operation is thus seen to be very similar to convolution, and indeed it is often performed using many of the same program subroutines. Operations that involve two images (division for deconvolution, multiplication for convolution, and multiplication by the conjugate for correlation) are sometimes called dyadic operations, to distinguish them from filtering and masking operations (monadic operations) in which a single frequency transform image is operated on (the filter or mask image is not a frequency-domain image nor does it contain complex values).

Usually, when correlation is performed, the wraparound assumption joining the left and right edges and the top and bottom of the image is not acceptable. In these cases, the image should be padded by surrounding it with zeroes to bring it to the next larger size for transformation (the next exact power of two, required by many FFT routines). Because the correlation operation also requires that the actual magnitude values of the transforms be used, slightly better mathematical precision can be achieved by padding with the average values of the original image brightnesses rather than with zeroes. It may also be useful to subtract the average brightness value from each pixel, which removes the zeroth (DC) term from the transformation. This value is usually the largest in the transform (it is the value at the central pixel), so its elimination allows the transform data more dynamic range.

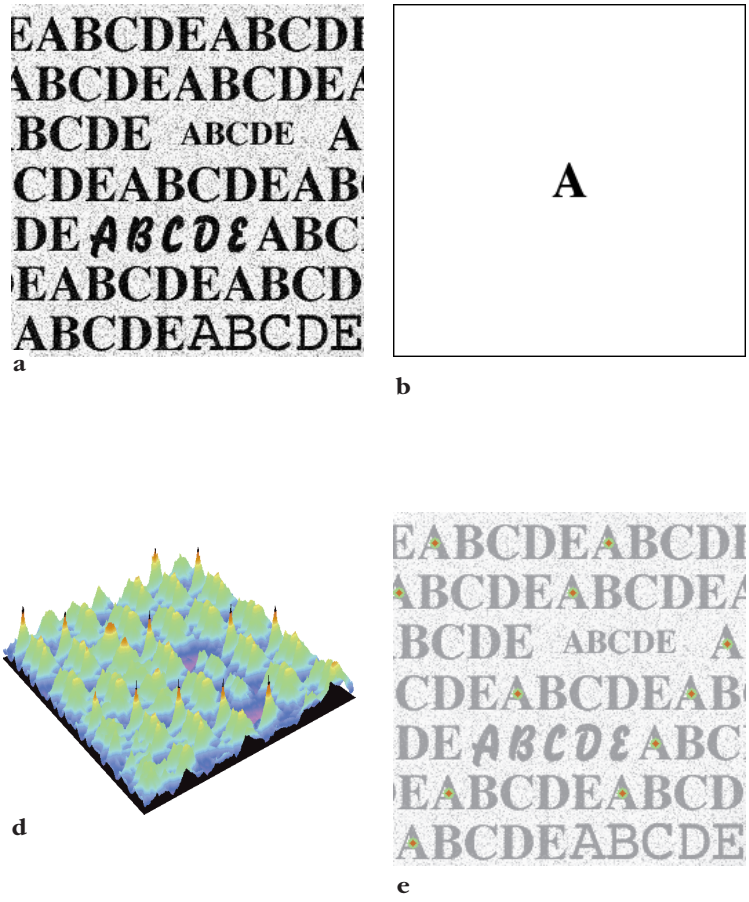


Figure 65. Cross-correlation example:
(a) *image containing many letters with superimposed random noise;*
(b) *target letter;*
(c) *cross-correlation result (grey-scale representation of goodness of match);*
(d) *isometric view of* ***c*** *showing peaks on the A's in the original;*
(e) *thresholded peaks in* ***c*** *superimposed on* ***a*** *— note that only the letters in the same size and font have been located.*

Correlation is primarily used for locating features in one image that appear in another. **Figure 65** shows an example. The image contains text, while the target contains the letter "A" by itself. The result of the cross-correlation after retransforming the image to the spatial domain shows peaks where the target letter is found, which may be more apparent when the same image presented as an isometric display. The brightest points in the correlation image correspond to the occurrences of the letter "A" in the same size and font as the target. Lower but still significant peaks exist, corresponding to two of the other letter As, in different fonts. In general, however, cross-correlation is quite size- and shape-specific as discussed in Chapter 10.

When combined with other image processing tools to analyze the cross-correlation image, this is an extremely fast and efficient tool for locating known features in images. In the example of **Figure 66**, a very large number of SEM images of Nuclepore filters with latex spheres presented a problem for automatic counting because of the variation in contrast and noise, the presence of dirt, and the texture presented by the filters. Also, the contrast of an isolated sphere is consistently different than that of one surrounded by other spheres.

A target image (**Figure 66a**) was created by averaging together ten representative latex spheres. This was then cross-correlated with each image, a top-hat filter (inner radius = 3 pixels; outer radius = 5 pixels; height = 8 grey scale values) applied to isolate each peak in the image, and the resulting spots counted automatically. The figure shows a few example images from the set, with marks showing the particles that were found and counted. After the top hat has been applied to select the peaks in the cross-correlation image, the resulting spots can be convolved with the stored target image to produce the result shown in **Figure 67**, which is just the latex spheres without the other features present in the original image.

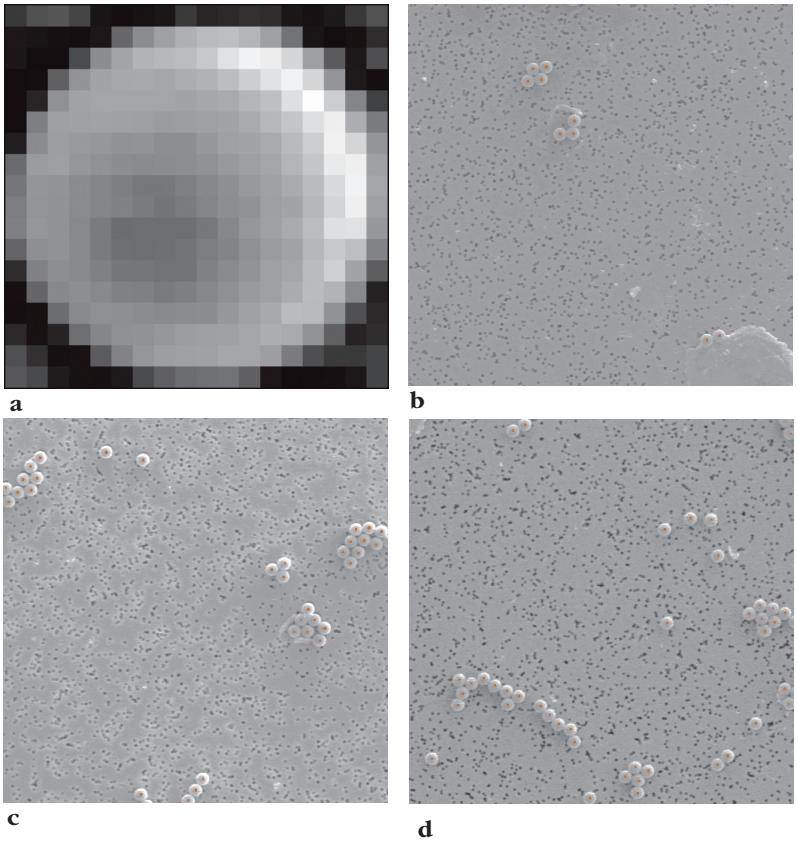


Figure 66. Cross-correlation to detect and count latex spheres on Nuclepore filters: (a) *average of ten representative latex spheres, used as a target;* ***(b–d)*** *representative example images, with red spots marking the result of the procedure described in the text.*

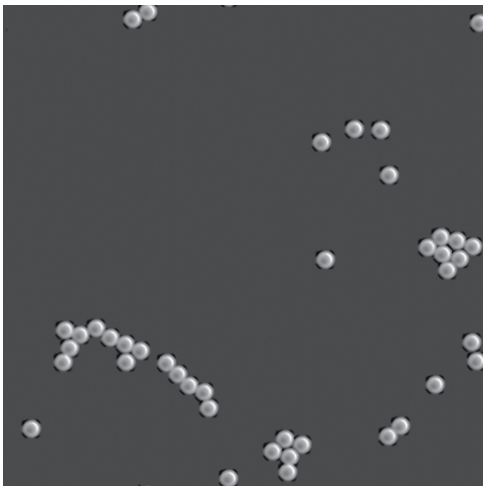


Figure 67. *Convolution of the target image from* ***Figure 66a*** *with the binary image of peaks found by the top hat filter applied to the cross-correlation image (the red marks superimposed on* ***Figure 66d****), showing just the latex spheres.*

Cross-correlation is also used as a tool for aligning serial section images. Even if the images are not identical (as in general they are not), enough common features exist so that a sharp cross-correlation peak occurs when two successive images are in best *x,y* alignment. The location of the peak can be determined to sub-pixel accuracy and used to shift the images into optimum alignment. The method does not deal with rotation, but it is possible to iteratively rotate the images and use the peak cross-correlation value as a measure of quality to determine the best value. Serial section alignment is discussed further in Chapter 12.

Autocorrelation

In the special case when the image functions *f* and *g* (and their transforms *F* and *G*) are the same, the correlation operation is called autocorrelation. This is used to combine together all parts of the image, in order to find repetitive structures. In Chapter 13, this method is used to measure the texture of surface roughness in terms of the characteristic distance over which amplitude of the peak drops. **Figure 68** shows another use of the autocorrelation function, to measure preferred orientation. The image is a felted textile material. Due to the fabrication process the fibers have neither a randomized nor uniform orientation; neither are they regularly arranged. The frequency transform shown indicates a slightly asymmetric shape. Performing the cross-correlation and retransforming to the spatial domain gives the image shown in **Figure 69**. The brightness profiles of the central spot along its major and minor axes show the preferred orientation quantitatively. This is the same result achieved by cross-correlation in the spatial domain (rotating the image by 180° and sliding the image across itself while recording the area of feature overlap as a function of relative offset) to determine preferred orientation.

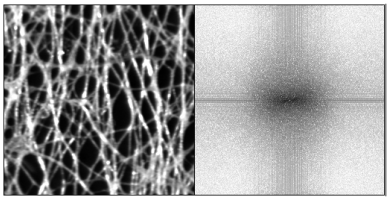


Figure 68. *Image of felted textile fibers (left) and the frequency transform (right).*

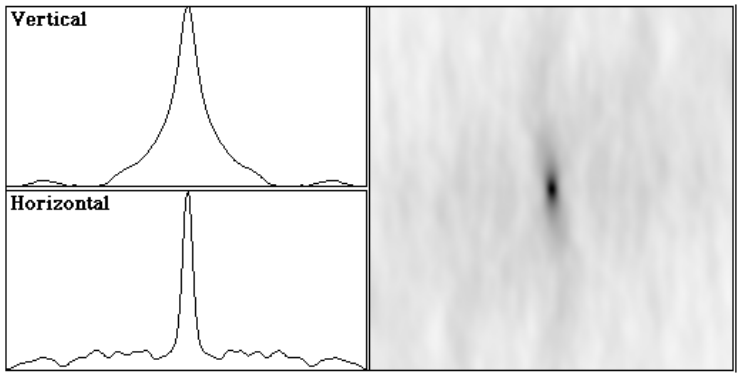


Figure 69. *Autocorrelation image from* ***Figure 68*** *(right), and its horizontal and vertical brightness profiles (left) along the major and minor axes of the peak.*

Conclusion

Processing images in the frequency domain is useful for removing certain types of noise and image blur, applying large convolution kernels, enhancing periodic structures, and locating defined structures in images. It can also be used to measure images to determine periodicity or preferred orientation. All these operations can be carried out in the spatial domain, but are often much more efficient in the frequency domain. The FFT operations can be efficiently computed in small (desktop) systems. This makes frequency-domain operations useful and important tools for image analysis.