

2.3

Basic Tools for Image Fourier Analysis

Alan C. Bovik
The University of Texas
at Austin

| | | |
|---|--|----|
| 1 | Introduction..... | 57 |
| 2 | Discrete-Space Sinusoids | 57 |
| 3 | Discrete-Space Fourier Transform | 58 |
| | Linearity of Discrete-Space Fourier Transform • Inversion of Discrete-Space Fourier Transform • Magnitude and Phase of Discrete-Space Fourier Transform • Symmetry of Discrete-Space Fourier Transform • Translation of Discrete-Space Fourier Transform • Convolution and the Discrete-Space Fourier Transform | |
| 4 | Two-Dimensional Discrete Fourier Transform | 61 |
| | Linearity and Invertibility of Discrete Fourier Transform • Symmetry of Discrete Fourier Transform • Periodicity of Discrete Fourier Transform • Image Periodicity Implied by Discrete Fourier Transform • Cyclic Convolution Property of the Discrete Fourier Transform • Linear Convolution Using the Discrete Fourier Transform • Computation of the Discrete Fourier Transform • Displaying the Discrete Fourier Transform | |
| 5 | Understanding Image Frequencies and the Discrete Fourier Transform..... | 67 |
| | Frequency Granularity • Frequency Orientation | |
| 6 | Related Topics in this <i>Handbook</i> | 72 |
| | Acknowledgment | 72 |

1 Introduction

In this third chapter on basic methods, the basic mathematic and algorithmic tools for the frequency-domain analysis of digital images are explained. Also, introduced is the two-dimensional discrete-space convolution. Convolution is the basis for linear filtering, which plays a central role in many places in this *Handbook*. An understanding of frequency-domain and linear filtering concepts is essential to be able to comprehend such significant topics as image and video enhancement, restoration, compression, segmentation, and wavelet-based methods. Exploring these ideas in a two-dimensional setting has the advantage that frequency-domain concepts and transforms can be visualized as images, often enhancing the accessibility of ideas.

2 Discrete-Space Sinusoids

Before defining any frequency-based transforms, first we shall explore the concept of *image frequency*, or more generally, of *two-dimensional frequency*. Many readers may have a basic background in the frequency-domain analysis of one-dimensional signals and systems. The basic theories in two dimensions are founded on the same principles. However, there are some extensions. For example, a two-dimensional frequency component, or sinusoidal function, is characterized not only by its location (phase shift) and frequency of oscillation, but also by its direction of oscillation.

Sinusoidal functions will play an essential role in all of the developments in this chapter. A *two-dimensional discrete-space*

sinusoid is a function of the form

$$\sin[2\pi(Um + Vn)] \quad (1)$$

Unlike a one-dimensional sinusoid, the function (1) has two frequencies, U and V (with units of cycles/pixel), which represent the frequency of oscillation along the vertical (m) and horizontal (n) spatial image dimensions. Generally, a two-dimensional sinusoid oscillates (is nonconstant) along every direction except for the direction orthogonal to the direction of fastest oscillation. The frequency of this fastest oscillation is the *radial frequency*:

$$\Omega = \sqrt{U^2 + V^2} \quad (2)$$

that has the same units as U and V , and the direction of this fastest oscillation is the *angle*:

$$\theta = \tan^{-1}\left(\frac{V}{U}\right) \quad (3)$$

with units of radians. Associated with (1) is the complex exponential function

$$\exp[j2\pi(Um + Vn)] = \cos[2\pi(Um + Vn)] + j\sin[2\pi(Um + Vn)], \quad (4)$$

where $j = \sqrt{-1}$ is the pure imaginary number.

In general, sinusoidal functions can be defined on discrete integer grids, hence (1) and (4) hold for all integers $-\infty < m, n < \infty$. However, sinusoidal functions of infinite duration are not encountered in practice, although they are useful for image modeling and in certain image decompositions that we will explore.

In practice, discrete-space images are confined to finite $M \times N$ sampling grids, and we will also find it convenient to use *finite-extent* ($M \times N$) *two-dimensional discrete-space sinusoids*, which are defined only for integers

$$0 \leq m \leq M - 1, 0 \leq n \leq N - 1, \quad (5)$$

and undefined elsewhere. A sinusoidal function that is confined to the domain (5) can be contained within an image matrix of dimensions $M \times N$ and is thus easily manipulated digitally.

In the case of finite sinusoids defined on finite grids (5), it will often be convenient to use the scaled frequencies,

$$(u, v) = (MU, NV) \quad (6)$$

which have the visually intuitive units of cycles/image. With this, the two-dimensional sinusoid (1) defined on finite grid (5) can be reexpressed as

$$\sin\left[2\pi\left(\frac{u}{M}m + \frac{v}{N}n\right)\right] \quad (7)$$

with similar redefinition of the complex exponential (4).

Figure 1 depicts several discrete-space sinusoids of dimensions 256×256 displayed as intensity images after linear mapping the gray scale of each to the range of 0 to 255. Because of the nonlinear response of the eye, the functions in Fig. 1 look somewhat more like square waves than smoothly varying sinusoids, particularly at higher frequencies. However, if any of the images in Fig. 1 is sampled along a straight line of arbitrary orientation, the result is an ideal (sampled) sinusoid.

A peculiarity of discrete-space (or discrete-time) sinusoids is that they have a maximum possible *physical* frequency at which they can oscillate. Although the frequency variables (u, v) or (U, V) may be taken arbitrarily large, these large values do not correspond to arbitrarily large physical oscillation frequencies. The ramifications of this are quite deep and significant and relate to the restrictions placed on sampling of continuous-space images (the Sampling Theorem) and the Nyquist frequency. Sampling of images and video is covered in Chapters 7.1 and 7.2.

As an example of this principle, we will study a one-dimensional example of discrete sinusoid. Consider the finite cosine function $\cos[2\pi(\frac{u}{M}m + \frac{v}{N}n)] = \cos(2\pi\frac{u}{16}m)$, which results by taking $M=N=16$, and $v=0$. This is a cosine wave propagating in the m direction only (all columns are the same) at frequency u (cycles/image).

Figure 2 depicts the one-dimensional cosine for various values of u . As can be seen, the physical oscillation frequency increases until $u=8$; for incrementally larger values of u , however, the physical frequency diminishes. In fact, the function is period-16 in the frequency index u :

$$\cos\left(2\pi\frac{u}{16}m\right) = \cos\left[2\pi\frac{(u+16k)}{16}m\right] \quad (8)$$

for all integers k . Indeed, the highest physical frequency of $\cos(2\pi\frac{u}{M}m)$ occurs at $u=M/2+kM$, (for M even) for all integers k . At these periodically placed frequencies, (8) is equal to $(-1)^m$; the fastest discrete-index oscillation is the alternating signal. This observation will be important next as we define the various frequency-domain image transforms.

3 Discrete-Space Fourier Transform

The *discrete-space Fourier transform*, or DSFT, of a given discrete-space image f is given by

$$F(U, V) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m, n) e^{-j2\pi(Um + Vn)} \quad (9)$$

with *inverse discrete-space Fourier transform* (IDSFT):

$$f(m, n) = \int_{-0.5}^{0.5} \int_{-0.5}^{0.5} F(U, V) e^{j2\pi(Um + Vn)} dU dV. \quad (10)$$

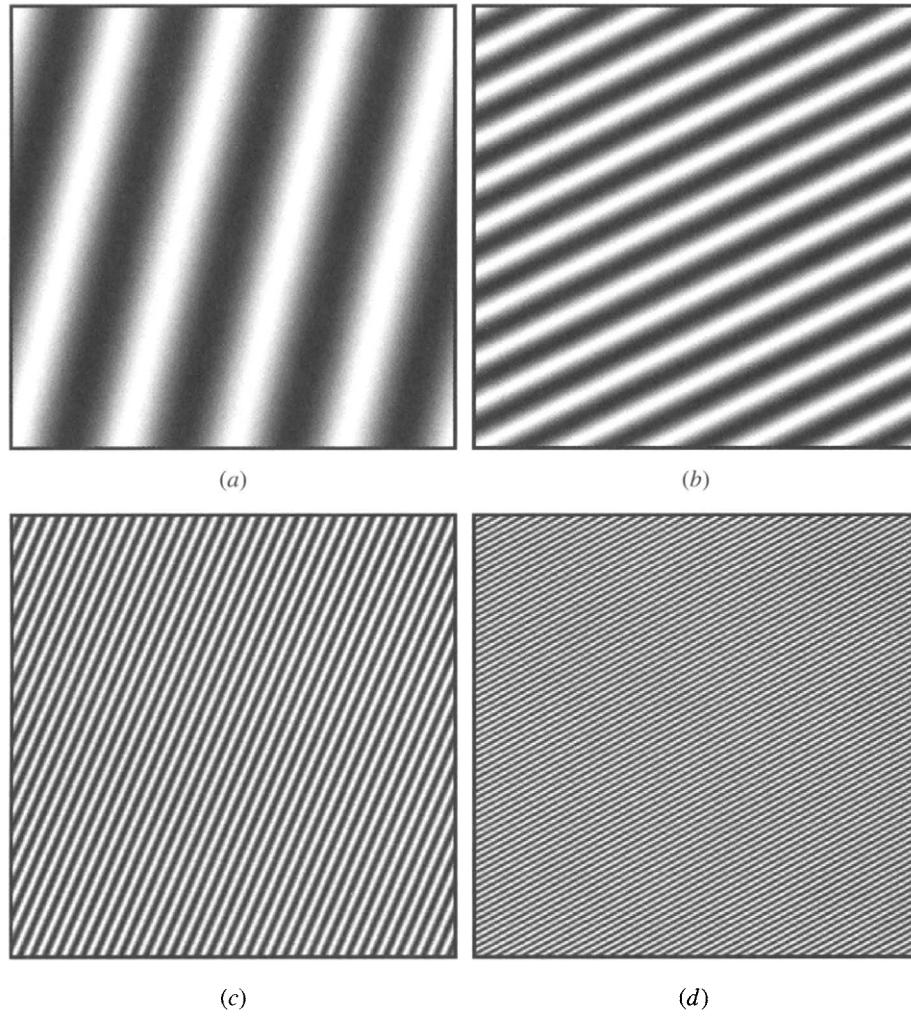


FIGURE 1 Examples of finite two-dimensional discrete-space sinusoidal functions. The scaled frequencies (6) measured in cycles/image are (a) $u = 1, v = 4$; (b) $u = 10, v = 5$; (c) $u = 15, v = 35$; and (d) $u = 65, v = 35$.

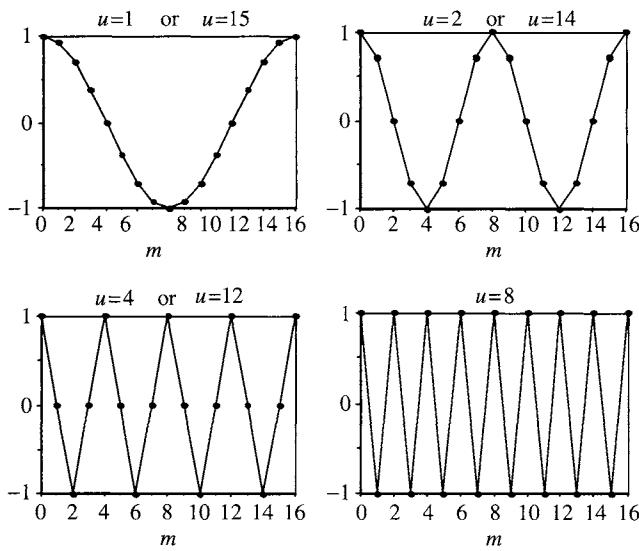


FIGURE 2 Illustration of physical versus numeric frequencies of discrete-space sinusoids.

When (9), (10) hold we will often make the notation $f \leftrightarrow F$ and say that f, F form a *DSFT pair*. The units of the frequencies (U, V) in (9), (10) are cycles/pixel. It should be noted that, unlike continuous Fourier transforms, the DSFT is asymmetric in that the forward transform F is continuous in the frequency variables (U, V) , whereas the image or inverse transform is discrete. Thus, the DSFT is defined as a summation, whereas the IDSFT is defined as an integral.

There are several ways of interpreting the DSFT (9), (10). The most usual mathematical interpretation of (10) is as a decomposition of $f(m, n)$ into orthonormal complex exponential basis functions $e^{j2\pi(Um+Vn)}$ that satisfy

$$\int_{-0.5}^{0.5} \int_{-0.5}^{0.5} e^{j2\pi(Um+Vn)} e^{-j2\pi(Up+Vq)} dU dV \\ = \begin{cases} 1; & m = p \text{ and } n = q \\ 0; & \text{else} \end{cases}. \quad (11)$$

Another (somewhat less precise) interpretation is the engineering concept of the transformation, without loss, of space-domain image information into frequency-domain image information. Representing the image information in the frequency domain has significant conceptual and algorithmic advantages, as will be seen. A third interpretation is a physical one, where the image is viewed as the result of a sophisticated constructive-destructive interference wave pattern. By assigning each of the infinite number of complex exponential wave functions $e^{j2\pi(Um+Vn)}$ the appropriate complex weights $F(U, V)$, the intricate structure of any discrete-space image can be recreated exactly as an interference-sum.

The DSFT possesses a number of important properties that will be useful in defining applications. In the following, assume that $f \xleftrightarrow{3} F$, $g \xleftrightarrow{3} G$, and $h \xleftrightarrow{3} H$.

Linearity of Discrete-Space Fourier Transform

Given images f , g and arbitrary complex constants a , b , the following holds:

$$af + bg \xleftrightarrow{3} aF + bG. \quad (12)$$

This property of linearity follows directly from (9) and can be extended to a weighted sum of any countable number of images. It is fundamental to many of the properties of, and operations involving, the DSFT.

Inversion of Discrete-Space Fourier Transform

The two-dimensional function $F(U, V)$ uniquely satisfies the relationships (9), (10). That the inversion holds can be easily shown by substituting (9) into (10), reversing the order of sum and integral, and then applying (11).

Magnitude and Phase of Discrete-Space Fourier Transform

The DSFT F of an image f is generally complex-valued. As such, it can be written in the form

$$F(U, V) = R(U, V) + jI(U, V) \quad (13)$$

where

$$R(U, V) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m, n) \cos[2\pi(Um + Vn)] \quad (14)$$

and

$$I(U, V) = - \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m, n) \sin[2\pi(Um + Vn)] \quad (15)$$

are the real and imaginary parts of $F(U, V)$, respectively.

The DSFT can also be written in the often-convenient phasor form

$$F(U, V) = |F(U, V)| e^{j\angle F(U, V)} \quad (16)$$

where the *magnitude spectrum* of image f is

$$|F(U, V)| = \sqrt{R^2(U, V) + I^2(U, V)} \quad (17)$$

$$= \sqrt{F(U, V)F^*(U, V)} \quad (18)$$

where “*” denotes the complex conjugation. The *phase spectrum* of image f is

$$\angle F(U, V) = \tan^{-1} \left[\frac{I(U, V)}{R(U, V)} \right]. \quad (19)$$

Symmetry of Discrete-Space Fourier Transform

If the image f is real, which is usually the case, the DSFT is *conjugate symmetric*:

$$F(U, V) = F*(-U, -V), \quad (20)$$

which means that the DSFT is completely specified by its values over any half-plane. Hence, if f is real, the DSFT is redundant. From (20), it follows that the magnitude spectrum is *even symmetric*:

$$|F(U, V)| = |F(-U, -V)|, \quad (21)$$

whereas the phase spectrum is *odd symmetric*:

$$\angle F(U, V) = -\angle F(-U, -V). \quad (22)$$

Translation of Discrete-Space Fourier Transform

Multiplying (or modulating) the discrete-space image $f(m, n)$ by a two-dimensional complex exponential wave function $\exp[j2\pi(U_0m + V_0n)]$ results in a translation of the DSFT:

$$f(m, n) \exp[j2\pi(U_0m + V_0n)] \xleftrightarrow{3} F(U - U_0, V - V_0). \quad (23)$$

Likewise, translating the image f by amounts m_0 , n_0 produces a modulated DSFT:

$$f(m - m_0, n - n_0) \xleftrightarrow{3} F(U, V) \exp[-j2\pi(Um_0 + Vn_0)]. \quad (24)$$

Convolution and the Discrete-Space Fourier Transform

Given two images or two-dimensional functions, f and h , their *two-dimensional discrete-space linear convolution* is given by

$$\begin{aligned} g(m, n) &= f(m, n) * h(m, n) = h(m, n) * f(m, n) \\ &= \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} f(p, q)h(m-p, n-q). \end{aligned} \quad (25)$$

The linear convolution expresses the result of passing an image signal f through a two-dimensional linear convolution system h (or vice versa). The commutativity of the convolution is easily seen by making a substitution of variables in the double sum in (25).

If g , f , and h satisfy the spatial convolution relationship (25), their DSFTs satisfy

$$G(U, V) = F(U, V)H(U, V), \quad (26)$$

hence, convolution in the space domain corresponds directly to multiplication in the spatial frequency domain. This important property is significant conceptually, as a simple and direct means for effecting the frequency content of an image, and computationally, because the linear convolution has such a simple expression in the frequency domain.

The two-dimensional DSFT is the basic mathematic tool for analyzing the frequency-domain content of two-dimensional discrete-space images. However, it has a major drawback for digital image processing applications: The DSFT $F(U, V)$ of a discrete-space image $f(m, n)$ is continuous in the frequency coordinates (U, V) ; there are an uncountably infinite number of values to compute. As such, discrete (digital) processing or display in the frequency domain is not possible using the DSFT unless it is modified in some way. Fortunately, this is possible when the image f is of finite dimensions. In fact, by sampling the DSFT in the frequency domain we are able to create a computable Fourier-domain transform.

4 Two-Dimensional Discrete Fourier Transform

Now we restrict our attention to the practical case of discrete-space images that are of finite extent. Hence, assume that image $f(m, n)$ can be expressed as a matrix $\mathbf{f} = [f(m, n) \mid 0 \leq m \leq M-1, 0 \leq n \leq N-1]$. As we will show, a finite-extent image matrix \mathbf{f} can be represented exactly as a *finite* weighted sum of two-dimensional frequency components, instead of an infinite number. This leads to computable and numerically manipulable frequency-domain representations. Before showing how this is done, we will introduce a special notation for the complex exponential that will simplify much of the ensuing development.

We will use

$$W_K = \exp\left[-j\frac{2\pi}{K}\right] \quad (27)$$

as a shorthand for the basic complex exponential, where K is the dimension along one of the image axes ($K=N$ or $K=M$). The notation (27) makes it possible to index the various elementary frequency components at arbitrary spatial and frequency coordinates by simple exponentiation:

$$W_M^{um} W_N^{vn} = \cos\left[2\pi\left(\frac{u}{M}m + \frac{v}{N}n\right)\right] - j\sin\left[2\pi\left(\frac{u}{M}m + \frac{v}{N}n\right)\right]. \quad (28)$$

This process of space and frequency indexing by exponentiation greatly simplifies the manipulation of frequency components and the definition of the discrete Fourier transform (DFT). Indeed, it is possible to develop concepts and frequency transforms without the use of complex numbers (and in fact some of these, such as the discrete cosine transform, or DCT, are widely used, especially in image/video compression—see Chapters 5.5, 5.6, 6.4, and 6.5 of this *Handbook*).

For the purpose of analysis and basic theory, it is much simpler to use W_M^{um} and W_N^{vn} to represent finite-extent (of dimensions M and N) frequency components oscillating at u (cycles/image) and v (cycles/image) in the m - and n -directions, respectively. Clearly,

$$|W_M^{um} W_N^{vn}| = 1 \quad (29)$$

and

$$\angle W_M^{um} W_N^{vn} = -2\pi\left(\frac{u}{M}m + \frac{v}{N}n\right). \quad (30)$$

Observe that the minimum physical frequency of W_M^{um} periodically occurs at the indices $u=kM$ for all integers k :

$$W_M^{kMm} = 1 \quad (31)$$

for any integer m ; the minimum oscillation is no oscillation. If M is even, the maximum physical frequency periodically occurs at the indices $u=kM+M/2$,

$$W_M^{(kM+M/2)m} = 1 \cdot e^{-j\pi m} = (-1)^m \quad (32)$$

which is the discrete period-2 (alternating) function, the highest possible discrete oscillation frequency.

The *two-dimensional discrete Fourier transform*, or *DFT* of the finite-extent ($M \times N$) image \mathbf{f} is given by

$$\tilde{F}(u, v) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) W_M^{um} W_N^{vn} \quad (33)$$

for integer frequencies $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. Hence, the DFT is also of finite extent $M \times N$, and can be expressed as a (generally complex-valued) matrix $\tilde{\mathbf{F}} = [\tilde{F}(u, v); 0 \leq u \leq M - 1, 0 \leq v \leq N - 1]$. It has a unique *inverse discrete Fourier transform*, or *IDFT*:

$$f(m, n) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \tilde{F}(u, v) W_M^{-um} W_N^{-vn} \quad (34)$$

for $0 \leq m \leq M - 1, 0 \leq n \leq N - 1$. When (33) and (34) hold it is often denoted $\mathbf{f} \xleftrightarrow{\text{DFT}} \tilde{\mathbf{F}}$ and we say that \mathbf{f} , $\tilde{\mathbf{F}}$ form a *DFT pair*.

A number of observations regarding the DFT and its relationship to the DSFT are necessary. First, the DFT and IDFT are symmetric, because both forward and inverse transforms are defined as sums. In fact, they have the same form, except for the polarity of the exponents and a scaling factor. Secondly, both forward and inverse transforms are finite sums; both $\tilde{\mathbf{F}}$ and \mathbf{f} can be represented uniquely as *finite* weighted sums of *finite-extent* complex exponentials with integer-indexed frequencies. Thus, for example, any 256×256 digital image can be expressed as the weighted sum of $256^2 = 65,536$ complex exponential (sinusoid) functions including those with real parts shown in Fig. 1. Note that the frequencies (u, v) are scaled so that their units are in cycles/image, as in (6) and Fig. 1.

Most importantly, the DFT has a direct relationship to the DSFT. In fact, the DFT of an $M \times N$ image \mathbf{f} is a uniformly *sampled* version of the DSFT of \mathbf{f} :

$$\tilde{F}(u, v) = F(U, V) \Big|_{U=\frac{u}{M}, V=\frac{v}{N}} \quad (35)$$

for integer frequency indices $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. Because \mathbf{f} is of finite extent, and contains MN elements, the DFT $\tilde{\mathbf{F}}$ is conservative in that it also requires only MN elements to contain complete information about \mathbf{f} (to be exactly invertible). Also, since $\tilde{\mathbf{F}}$ is simply evenly spaced samples of F , many of the properties of the DSFT translate directly with little or no modification to the DFT.

Linearity and Invertibility of Discrete Fourier Transform

The DFT is linear in the sense of (12). It is uniquely invertible, as can be established by substituting (33) into (34), reversing the order of summation, and using the fact that the discrete complex exponentials are also orthogonal

$$\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} (W_M^{um} W_M^{-vp})(W_N^{vn} W_N^{-vq}) = \begin{cases} MN; & m=p \text{ and } n=q \\ 0; & \text{else} \end{cases} \quad (36)$$

The DFT matrix $\tilde{\mathbf{F}}$ is generally complex; hence, it has an associated magnitude spectrum matrix, denoted

$$|\tilde{\mathbf{F}}| = [|\tilde{F}(u, v)|; 0 \leq u \leq M - 1, 0 \leq v \leq N - 1] \quad (37)$$

and phase spectrum matrix denoted

$$\angle \tilde{\mathbf{F}} = [\angle \tilde{F}(u, v); 0 \leq u \leq M - 1, 0 \leq v \leq N - 1]. \quad (38)$$

The elements of $|\tilde{\mathbf{F}}|$ and $\angle \tilde{\mathbf{F}}$ are computed in the same way as the DSFT magnitude and phase (16)–(19).

Symmetry of Discrete Fourier Transform

Like the DSFT, if \mathbf{f} is real-valued, the DFT matrix is conjugate symmetric, but in the matrix sense:

$$\tilde{F}(u, v) = \tilde{F} * (M - u, N - v) \quad (39)$$

for $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. This follows easily by substitution of the reversed and translated frequency indices $(M - u, N - v)$ into the forward DFT equation (33). An apparent repercussion of (39) is that the DFT $\tilde{\mathbf{F}}$ matrix is redundant, and hence can represent the $M \times N$ image with only about $MN/2$ DFT coefficients. This mystery is resolved by realizing that $\tilde{\mathbf{F}}$ is complex-valued; hence, it requires twice the storage for real and imaginary components. If \mathbf{f} is not real-valued, (39) does not hold.

Of course, (39) implies symmetries of the magnitude and phase spectra:

$$|\tilde{F}(u, v)| = |\tilde{F}(M - u, N - v)| \quad (40)$$

and

$$\angle \tilde{F}(u, v) = -\angle \tilde{F}(M - u, N - v) \quad (41)$$

for $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$.

Periodicity of Discrete Fourier Transform

Another property of the DSFT that carries over to the DFT is frequency periodicity. Recall that the DSFT $F(U, V)$ has unit period in U and V . The DFT matrix $\tilde{\mathbf{F}}$ was defined to be of finite extent $M \times N$. However, the forward DFT equation (33) admits the possibility of evaluating $\tilde{F}(u, v)$ outside of the range $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. It turns out that $\tilde{F}(u, v)$ is period- M and period- N along the u and v dimensions, respectively. For any integers k, l :

$$\tilde{F}(u + kM, v + lN) = \tilde{F}(u, v) \quad (42)$$

for every $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. This follows easily by substitution of the periodically extended frequency indices $(u + kM, v + lN)$ into the forward DFT equation (33). The interpretation (42) of the DFT is called the *periodic extension* of the DFT. It is defined for all integer frequencies u, v .

Although many properties of the DFT are the same, or similar to those of the DSFT, certain important properties are different. These effects arise from sampling the DSFT to create the DFT.

Image Periodicity Implied by Discrete Fourier Transform

A seemingly innocuous, yet extremely important consequence of sampling the DSFT is that the resulting DFT equations imply that the image \mathbf{f} is itself periodic. In fact, the IDFT equation (34) implies that for any integers k, l :

$$f(m + kM, n + lN) = f(m, n) \quad (43)$$

for every $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. This follows easily by substitution of the periodically extended space indices $(m + kM, n + lN)$ into the inverse DFT equation (34).

Clearly, finite-extent digital images arise from imaging the real world through finite field of view (FOV) devices, such as cameras, and outside that FOV, the world does not repeat itself periodically, *ad infinitum*. The implied periodicity of \mathbf{f} is purely a synthetic effect that derives from sampling the DSFT. Nevertheless, it is of paramount importance, because any algorithm that is developed, and that uses the DFT, will operate as though the DFT-transformed image were spatially periodic in the sense (43). One important property and application of the DFT that is affected by this spatial periodicity is the frequency-domain convolution property.

Cyclic Convolution Property of the Discrete Fourier Transform

One of the most significant properties of the DSFT is the linear convolution property (25), (26), which says that space-domain convolution corresponds to frequency-domain multiplication:

$$\mathbf{f} * \mathbf{h} \xleftrightarrow{\text{FH.}} \mathbf{F}\mathbf{H}. \quad (44)$$

This useful property makes it possible to analyze and design linear convolution-based systems in the frequency domain. Unfortunately, property (44) does not hold for the DFT; a product of DFTs does not correspond (inverse transform) to the linear convolution of the original DFT-transformed functions or images. However, it does correspond to another type of convolution, variously known as *cyclic convolution*, *circular convolution*, or *wraparound convolution*.

We will demonstrate the form of the cyclic convolution by deriving it. Consider the two $M \times N$ image functions $\mathbf{f} \xleftrightarrow{\text{DFT}} \widetilde{\mathbf{F}}$ and $\mathbf{h} \xleftrightarrow{\text{DFT}} \widetilde{\mathbf{H}}$. Define the *point wise* matrix product¹

$$\widetilde{\mathbf{G}} = \widetilde{\mathbf{F}} \otimes \widetilde{\mathbf{H}} \quad (45)$$

according to

$$\widetilde{\mathbf{G}}(u, v) = \widetilde{\mathbf{F}}(u, v)\widetilde{\mathbf{H}}(u, v) \quad (46)$$

for $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. Thus we are interested in the form of \mathbf{g} . For each $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$ we have that

$$\begin{aligned} g(m, n) &= \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \widetilde{\mathbf{G}}(u, v) W_M^{-um} W_N^{-vn} \\ &= \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \widetilde{\mathbf{F}}(u, v) \widetilde{\mathbf{H}}(u, v) W_M^{-um} W_N^{-vn} \\ &= \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \left\{ \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} f(p, q) W_M^{up} W_N^{vq} \right\} \\ &\quad \times \left\{ \sum_{r=0}^{M-1} \sum_{s=0}^{N-1} h(r, s) W_M^{ur} W_N^{vs} \right\} W_M^{-um} W_N^{-vn} \end{aligned} \quad (47)$$

by substitution of the definitions of $\widetilde{\mathbf{F}}(u, v)$ and $\widetilde{\mathbf{H}}(u, v)$. Rearranging the order of the summations to collect all of the complex exponentials inside the innermost summation reveals that

$$\begin{aligned} g(m, n) &= \frac{1}{MN} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} f(p, q) \sum_{r=0}^{M-1} \sum_{s=0}^{N-1} h(r, s) \\ &\quad \times \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} W_M^{u(p+r-m)} W_N^{v(q+s-n)} \end{aligned} \quad (48)$$

Now, from (36), the innermost summation

$$\sum_{u=0}^{M-1} \sum_{v=0}^{N-1} W_M^{u(p+r-m)} W_N^{v(q+s-n)} = \begin{cases} MN; & r = m - p \text{ and } s = n - q \\ 0; & \text{else} \end{cases} \quad (49)$$

hence

$$g(m, n) = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} f(p, q) h[(m - p)_M, (n - q)_N] \quad (50)$$

$$= f(m, n) \oplus h(m, n) = h(m, n) \oplus f(m, n) \quad (51)$$

¹As opposed to the standard matrix product.

where $(x)_N = x \bmod N$ and the symbol “ \oplus ” denotes the two-dimensional cyclic convolution.² The final step of obtaining (50) from (49) follows since the argument of the shifted and twice-reversed (along each axis) function $h(m-p, n-q)$ finds no meaning whenever $(m-p) \notin \{0, \dots, M-1\}$ or $(n-q) \notin \{0, \dots, N-1\}$, since h is undefined outside of those coordinates. However, because the DFT was used to compute $g(m, n)$, then the periodic extension of $h[(m-p), (n-q)]$ is implied, which can be expressed as $h[(m-p)_M, (n-q)_N]$. Hence (50) follows. That “ \oplus ” is commutative is easily established by a substitution of variables in (50). It can also be seen that cyclic convolution is a form of linear convolution, but with one (either, but not both) of the two functions being periodically extended. Hence

$$\begin{aligned} f(m, n) \oplus h(m, n) &= f(m, n) * h[(m)_M, (n)_N] \\ &= f[(m)_M, (n)_N] * h(m, n). \end{aligned} \quad (52)$$

This *cyclic convolution property* of the DFT is unfortunate, because in most applications it is not desired to compute the cyclic convolution of two image functions. Instead, what is frequently desired is the linear convolution of two functions, as in the case of linear filtering. In both linear and cyclic convolutions, the two functions are superimposed, with one function reversed along both axes and shifted to the point at which the convolution is being computed. The product of the functions is computed at every point of overlap, with the sum of products being the convolution. In the case of the cyclic convolution, one (not both) of the functions is periodically extended; thus, the overlap is much larger and wraps around the image boundaries. This produces a significant error with respect to the correct linear convolution result. This error is called *spatial aliasing* because the wraparound error contributes false information to the convolution sum.

Figure 3 depicts the linear and cyclic convolutions of two hypothetical $M \times N$ images f and h at a point (m_0, n_0) . From Fig. 3, it can be seen that the wraparound error can overwhelm the linear convolution contribution. Note in Fig. 3(b) that although the linear convolution sum (25) extends over the indices $0 \leq m \leq M-1$ and $0 \leq n \leq N-1$, the overlap is restricted to the indices.

Linear Convolution Using the Discrete Fourier Transform

Fortunately, it turns out that it is possible to compute the linear convolution of two arbitrary finite-extent two-dimensional discrete-space functions or images using the DFT. The process

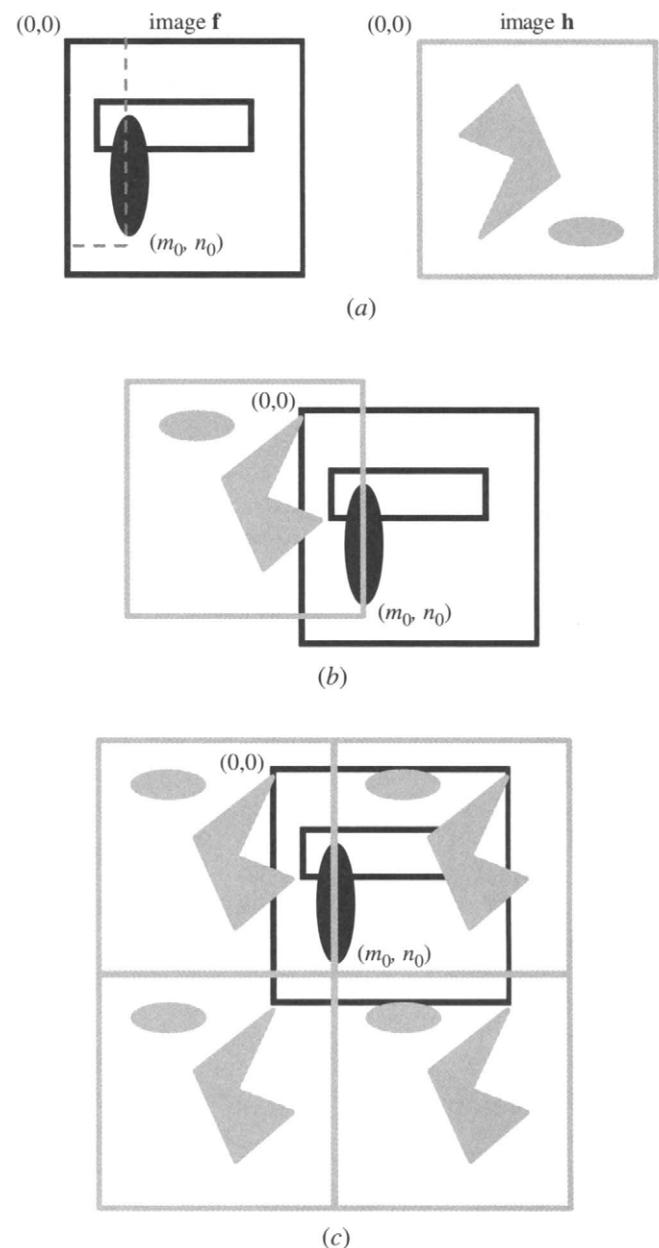


FIGURE 3 Convolution of two images. (a) Images f and h . (b) Linear convolution result at (m_0, n_0) is computed as the sum of products where f and h overlap. (c) Cyclic convolution result at (m_0, n_0) is computed as the sum of products where f and the periodically extended h overlap.

requires modifying the functions to be convolved prior to taking the product of their DFTs. The modification acts to cancel the effects of spatial aliasing.

Suppose more generally that f and h are two arbitrary finite-extent images of dimensions $M \times N$ and $P \times Q$, respectively. We are interested in computing the linear convolution $g = f * h$ using the DFT. We assume the general case where the images f, h do not have the same dimensions, because in most applications an image is convolved with a filter function of different (usually much smaller) extent.

²Modular arithmetic is remaindering. Hence $(x)_N$ is the integer remainder of (x/N) .

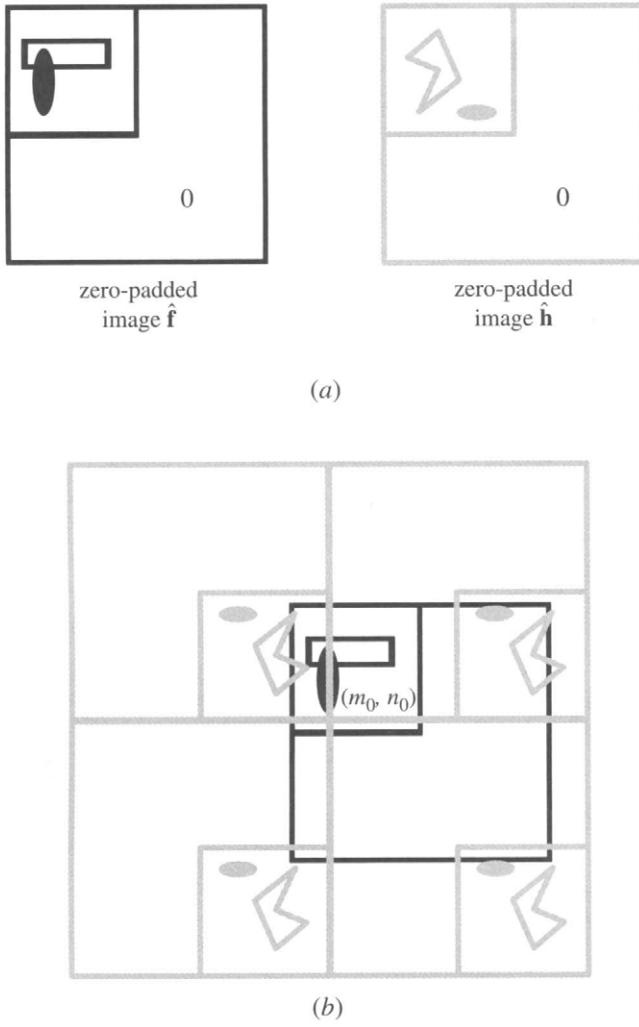


FIGURE 4 Linear convolution of the same two images as Figure 2 by zero-padding and cyclic convolution (via the DFT). (a) Zero-padded images \hat{f} and \hat{h} . (b) Cyclic convolution at (m_0, n_0) computed as the sum of products where \hat{f} and the periodically extended \hat{h} overlap. These products are zero except over the range $0 \leq p \leq m_0$ and $0 \leq q \leq n_0$.

Clearly,

$$g(m, n) = f(m, n) * h(m, n) = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} f(p, q)h(m - p, n - q). \quad (53)$$

Inverting the pointwise products of the DFTs $\tilde{\mathbf{F}} \otimes \tilde{\mathbf{H}}$ will not lead to (53), because wraparound error will occur. To cancel the wraparound error, the functions f and h are modified by increasing their size by *zero-padding* them. Zero-padding means that the arrays f and h are expanded into larger arrays, denoted \hat{f} and \hat{h} , by filling the empty spaces with zeroes. To compute the linear convolution, the pointwise product $\tilde{\mathbf{G}} = \tilde{\mathbf{F}} \otimes \tilde{\mathbf{H}}$ of the DFTs of the zero-padded functions \hat{f} and

\hat{h} is computed. The inverse DFT \hat{g} of $\tilde{\mathbf{G}}$ then contains the correct linear convolution result.

The question remains as to how many zeroes are used to pad the functions f and h . The answer to this lies in understanding how zero-padding works and how large the linear convolution result should be. Zero-padding acts to cancel the spatial aliasing error (wraparound) of the DFT by supplying zeroes where the wraparound products occur. Hence, the wraparound products are all zero and contribute nothing to the convolution sum. This leaves only the linear convolution contribution to the result. To understand how many zeroes are needed, it must be realized that the resulting product DFT $\tilde{\mathbf{G}}$ corresponds to a periodic function \hat{g} . If the horizontal/vertical periods are too small (not enough zero-padding), the periodic replicas will overlap (spatial aliasing). If the periods are just large enough, the periodic replicas will be contiguous instead of overlapping; hence, spatial aliasing will be canceled. Padding with more zeroes than this results in excess computation. Figure 4 depicts the successful result of zero-padding to eliminate wraparound error.

The correct period lengths are equal to the lengths of the correct linear convolution result. The linear convolution result of two arbitrary $M \times N$ and $P \times Q$ image functions will generally be $(M + P - 1) \times (N + Q - 1)$, hence we would like the DFT $\tilde{\mathbf{G}}$ to have these dimensions. Therefore, the $M \times N$ function f and the $P \times Q$ function h must both be zero-padded to size $(M + P - 1) \times (N + Q - 1)$. This yields the correct linear convolution result:

$$\hat{g} = \hat{f} \oplus \hat{h} = f * h. \quad (54)$$

In most cases, linear convolution is performed between an image and a filter function much smaller than the image: $M \gg P$ and $N \gg Q$. In such cases, the result is not much larger than the image, and often only the $M \times N$ portion indexed $0 \leq m \leq M - 1, 0 \leq n \leq N - 1$ is retained. The reasoning behind this is first, that it may be desirable to retain images of size MN only, and second, that the linear convolution result beyond the borders of the original image may be of little interest because the original image was zero there anyway.

Computation of the Discrete Fourier Transform

Inspection of the DFT relation (33) reveals that computation of each of the MN DFT coefficients requires on the order of MN complex multiplies/additions. Hence, on the order of M^2N^2 complex multiplies and additions are needed to compute the overall DFT of an $M \times N$ image f . For example, if $M = N = 512$, then on the order of $2^{36} = 6.9 \times 10^{10}$ complex multiplies/additions are needed, which is a very large number. Of course, these numbers assume a naive implementation

without any optimization. Fortunately, fast algorithms for DFT computation, collectively referred to as *fast Fourier transform (FFT)* algorithms, have been intensively studied for many years. We will not delve into the design of these, because it goes beyond what we want to accomplish in a handbook and they are available in any image processing programming library or development environment (Chapter 4.13 reviews these) and most math library programs.

The FFT offers a computational complexity of order not exceeding $MN \log_2(MN)$, which represents a considerable speedup. For example, if $M=N=512$, the complexity is on the order of $9 \times 2^{19} = 4.7 \times 10^6$. This represents a very common speedup of more than 14,500:1!

Analysis of the complexity of cyclic convolution is similar. If two images of the same size $M \times N$ are convolved, then again, the naive complexity is on the order of M^2N^2 complex multiplies and additions. If the DFT of each image is computed, the resulting DFTs point wise multiplied, and the inverse DFT of this product calculated, the overall complexity is on the order of $MN \log_2(2M^3N^3)$. For the common case $M=N=512$, the speedup still exceeds 4,700:1.

If linear convolution is computed via the DFT, the computation is increased somewhat because the images are increased in size by zero-padding. Thus, the speedup of DFT-based linear convolution is somewhat reduced (although in a fixed-hardware realization, the known existence of these zeroes can be used to effect a speedup). However, if the functions being linearly convolved are both not small, the DFT approach will always be faster. If one of the functions is very small, say covering fewer than 32 samples (such as a small linear filter template), it is possible that direct space-domain computation of the linear convolution may be faster than DFT-based computation. However, there is no strict rule of thumb to determine this lower cutoff size, because it depends on the filter shape, the algorithms used to compute DFTs and convolutions, any special-purpose hardware, and so on.

Displaying the Discrete Fourier Transform

It is often of interest to visualize the DFT of an image. This is possible because the DFT is a sampled function of finite (periodic) extent. Displaying one period of the DFT of image f reveals a picture of the frequency content of the image. Because the DFT is complex, one can display either the magnitude spectrum $|\tilde{F}|$ or the phase spectrum $\angle \tilde{F}$ as a single two-dimensional intensity image.

However, the phase spectrum $\angle \tilde{F}$ is usually not visually revealing when displayed. Generally it appears quite random, and so usually the magnitude spectrum $|\tilde{F}|$ only is absorbed visually. This is not intended to imply that image phase information is not important; in fact, it is exquisitely important, because it determines the relative shifts of the component complex exponential functions that make up the DFT decomposition. Modifying or ignoring image phase will

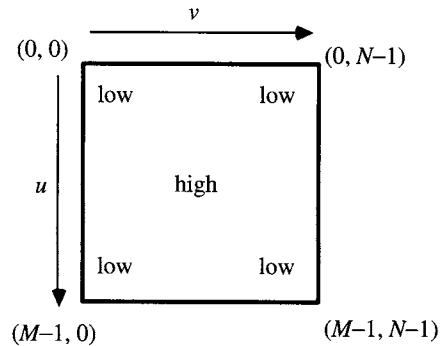


FIGURE 5 Distribution of high and low frequency DFT coefficients.

destroy the delicate constructive — destructive interference pattern of the sinusoids that make up the image.

As briefly noted in Chapter 2.1, displays of the Fourier transform magnitude will tend to be visually dominated by the low-frequency and zero-frequency coefficients, often to such an extent that the DFT magnitude appears as a single spot. This is highly undesirable, because most of the interesting information usually occurs at frequencies away from the lowest frequencies. An effective way to bring out the higher-frequency coefficients for visual display is via a point logarithmic operation: instead of displaying $|\tilde{F}|$, display

$$\log_2[1 + |\tilde{F}(u, v)|] \quad (55)$$

for $0 \leq u \leq M - 1, 0 \leq v \leq N - 1$. This has the effect of compressing all of the DFT magnitudes, but larger magnitudes much more so. Of course, because all of the logarithmic magnitudes will be quite small, a full-scale histogram stretch should then be applied to fill the gray-scale range.

Another consideration when displaying the DFT of a discrete-space image is illustrated in Fig. 5. In the DFT formulation, a single $M \times N$ period of the DFT is sufficient to represent the image information and for display. However, the DFT matrix is even symmetric across both diagonals. More important, the center of symmetry occurs in the image center, where the high-frequency coefficients are clustered near $(u, v) = (M/2, N/2)$. This is contrary to conventional intuition, because in most engineering applications, Fourier transform magnitudes are displayed with zero and low-frequency coefficients at the center. This is particularly true of one-dimensional continuous Fourier transform magnitudes, which are plotted as graphs with the zero frequency at the origin. This is also visually convenient, because the dominant lower frequency coefficients then are clustered together at the center, instead of being scattered about the display.

A natural way of remedying this is to instead display the shifted DFT magnitude

$$|\tilde{F}(u - M/2, v - N/2)| \quad (56)$$

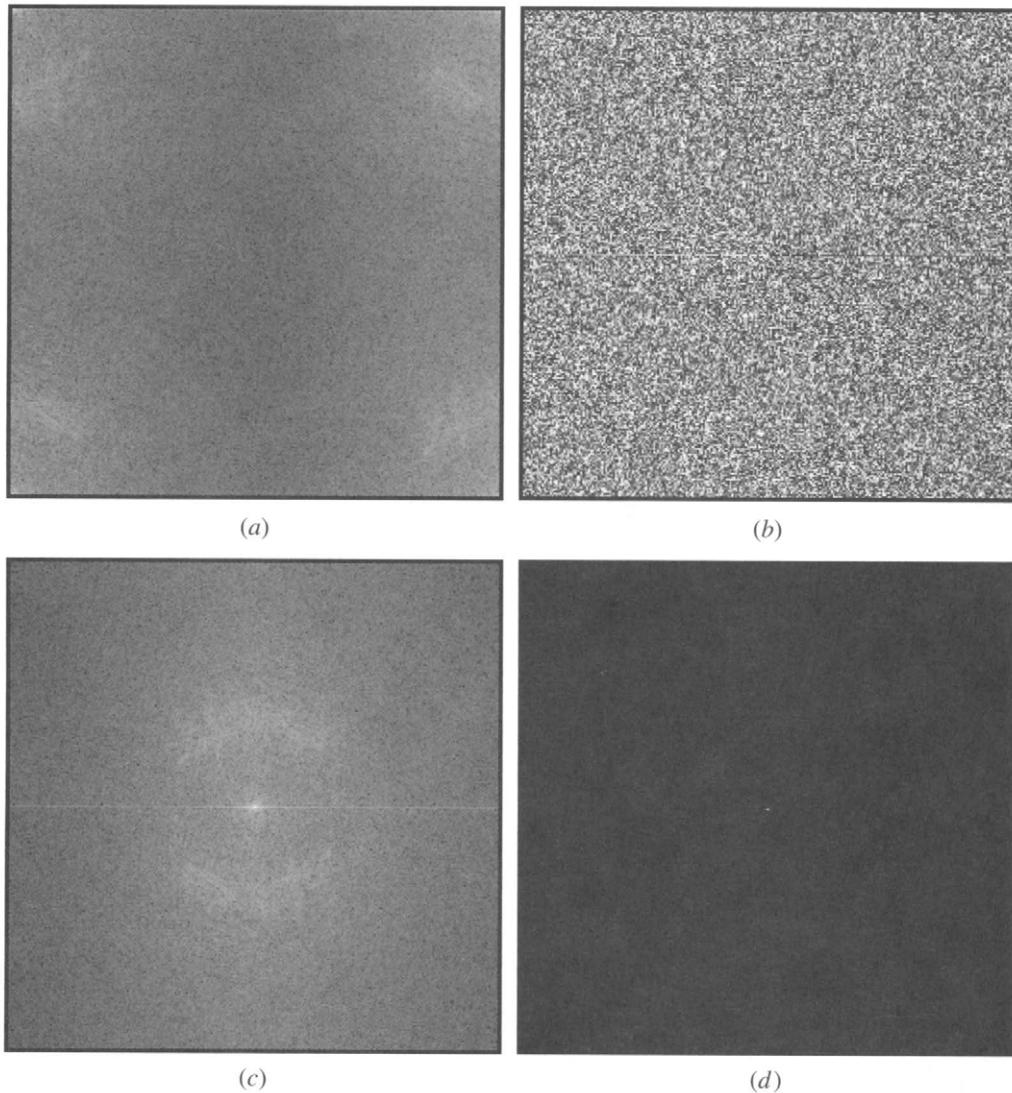


FIGURE 6 Display of DFT of image “fingerprint” from Chapter 1.1. (a) DFT magnitude (logarithmically compressed and histogram stretched); (b) DFT phase; (c) centered DFT (logarithmically compressed and histogram stretched); (d) centered DFT (without logarithmic compression).

for $0 \leq u \leq M - 1$, $0 \leq v \leq N - 1$. This can be accomplished in a simple way by taking the DFT of

$$(-1)^{m+n} f(m, n) \xrightarrow{\text{DFT}} \tilde{F}(u - M/2, v - N/2) \quad (57)$$

Relation (57) follows since $(-1)^{m+n} = e^{j\pi(m+n)}$. Hence, from (23) the DSFT is shifted by amount 1/2 cycles/pixel along both dimensions; since the DFT uses the scaled frequencies (6), the DFT is shifted by $M/2$ and $N/2$ cycles/image in the u - and v - directions, respectively.

Figure 6 illustrates the display of the DFT of the image “fingerprint” image which is Fig. 8 of Chapter 1.1. As can be seen, the DFT phase is visually unrevealing, while the DFT magnitude is most visually revealing when it is centered and logarithmically compressed.

5 Understanding Image Frequencies and the Discrete Fourier Transform

It is sometimes easy to lose track of the meaning of the DFT and of the frequency content of an image in all of the (necessary!) mathematics. When using the DFT, it is important to remember that the DFT is a detailed map of the frequency content of the image, which can be visually digested and digitally processed. It is a useful exercise to examine the DFT of images, particularly the DFT magnitudes, because it reveals much about the distribution and meaning of image frequencies. It is also useful to consider what happens when the image frequencies are modified in certain simple ways, since this reveals further insights into spatial frequencies

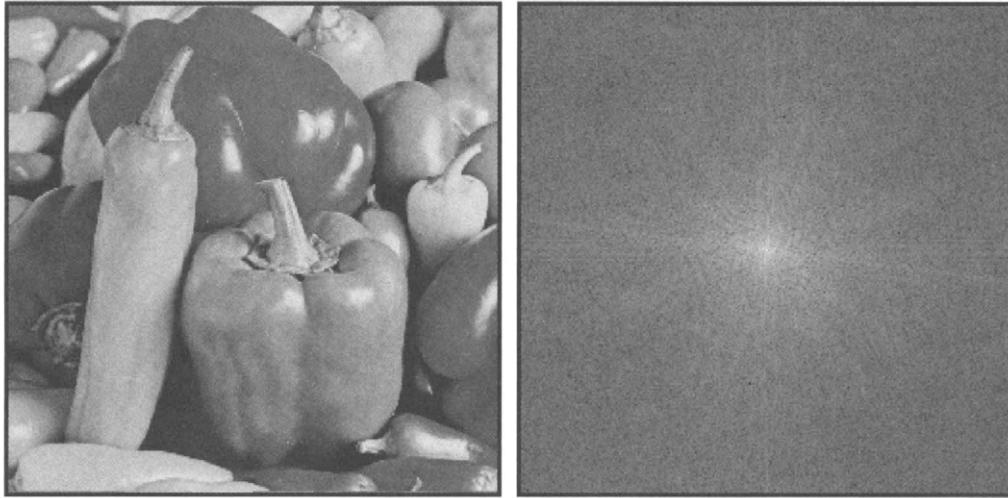


FIGURE 7 Image “peppers” (left) and DFT magnitude (right).

and it moves toward understanding how image frequencies can be systematically modified to produce useful results.

In the following we will present and discuss a number of interesting digital images along with their DFT magnitudes represented as intensity images. When examining these, recall that bright regions in the DFT magnitude “image” correspond to frequencies that have large magnitudes in the real image. Also, in some cases, the DFT magnitudes have been logarithmically compressed and centered via (55) and (57) for improved visual interpretation.

Most engineers and scientists are introduced to Fourier-domain concepts in a one-dimensional setting. One-dimensional signal frequencies have a single attribute — that of being either “high” or “low” frequency. Two-dimensional (and higher-dimensional) signal frequencies have richer descriptions characterized by both magnitude and direction,³ which lend themselves well to visualization. We will seek intuition into these attributes as we separately consider the *granularity* of image frequencies, corresponding to radial frequency (2), and the *orientation* of image frequencies, corresponding to frequency angle (3).

Frequency Granularity

The granularity of an image frequency refers to its radial frequency. “Granularity” describes the appearance of an image that is strongly characterized by the radial frequency portrait of the DFT. An abundance of large coefficients near the DFT origin corresponds to the existence of large, smooth, image components, often of smooth image surfaces or background. Note that nearly every image will have a significant peak at the

DFT origin (unless it is very dark), because from (33) it is the summed intensity of the image (integrated optical density):

$$\tilde{F}(0,0) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n). \quad (58)$$

The image “fingerprint” (Fig. 8 of Chapter 1.1) with DFT magnitude shown in Fig. 5(c) just above is an excellent example of image granularity. The image contains relatively little low-frequency or very high — frequency energy, but does contain an abundance of mid-frequency energy as can be seen in the symmetrically placed half arcs above and below the frequency origin. The “fingerprint” image is a good example of an image that is primarily bandpass.

Figure 7 depicts image “peppers” and its DFT magnitude. The image contains primarily smooth intensity surfaces separated by abrupt intensity changes. The smooth surfaces contribute to the heavy distribution of low-frequency DFT coefficients, while the intensity transitions (“edges”) contribute a noticeable amount of mid-to-higher frequencies over a broad range of orientations.

Finally, Fig. 8, “cane” depicts an image of a repetitive weave pattern that exhibits a number of repetitive peaks in the DFT magnitude image. These are *harmonics* that naturally appear in signals (such as music signals) or images that contain periodic or nearly periodic structures.

As an experiment toward understanding frequency content, suppose that we define several zero-one image frequency masks, as depicted in Fig. 9.

By masking (multiplying) the DFT $\tilde{\mathbf{F}}$ of an image \mathbf{f} with each of these will produce, following an inverse DFT, a resulting image containing only low, mid, or high frequencies. In the following, we show examples of this operation. The astute reader may have observed that the zero-one frequency masks, which are defined in the DFT domain, may be regarded

³Strictly speaking, one-dimensional frequencies can be positive or negative. This polarity may be regarded as a directional attribute, although without much meaning for real-valued, one-dimensional signals.

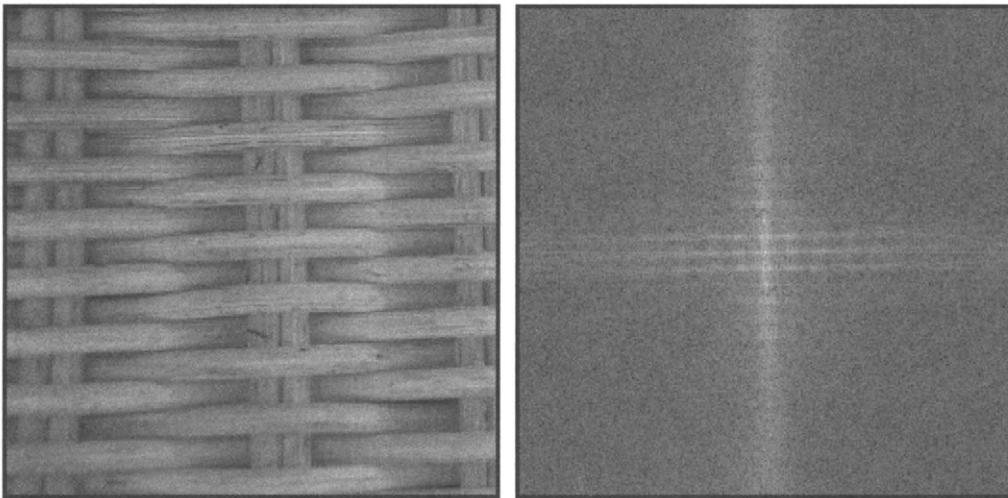


FIGURE 8 Image “cane” (left) and DFT magnitude (right).

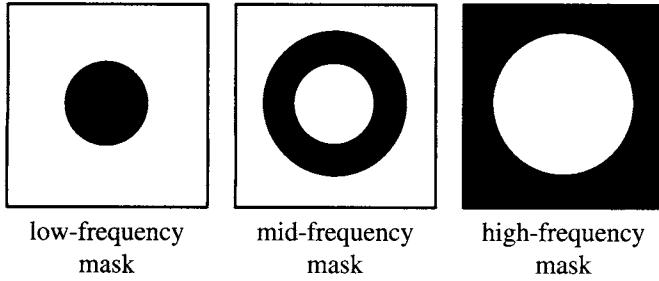


FIGURE 9 Image radial frequency masks. Black pixels take value “1”, white pixels take value “0.”

as DFTs with IDFTs defined in the space domain. Because we are taking the products of functions in the DFT domain, it has the interpretation of cyclic convolution (46–51) in the space domain. Therefore, the following examples should not be

thought of as low-pass, bandpass, or high-pass linear filtering operations in the proper sense. Instead, these are instructive examples where image frequencies are being directly removed. The approach is not a substitute for a proper linear filtering of the image using a space-domain filter that has been DFT-transformed with proper zero-padding. In particular, the naive demonstration here does dictate how the frequencies between the DFT frequencies (frequency samples) are effected, as a properly designed linear filter does.

In all of the examples, the image DFT was computed, multiplied by a zero-one frequency mask, and inverse DFT-ed. Finally, a full-scale histogram stretch was applied to map the result to the gray-level range (0, 255), because otherwise, the resulting image is not guaranteed to be positive.

In the first example, shown in Fig. 10, the image “fingerprint” is shown following treatment with the low-frequency

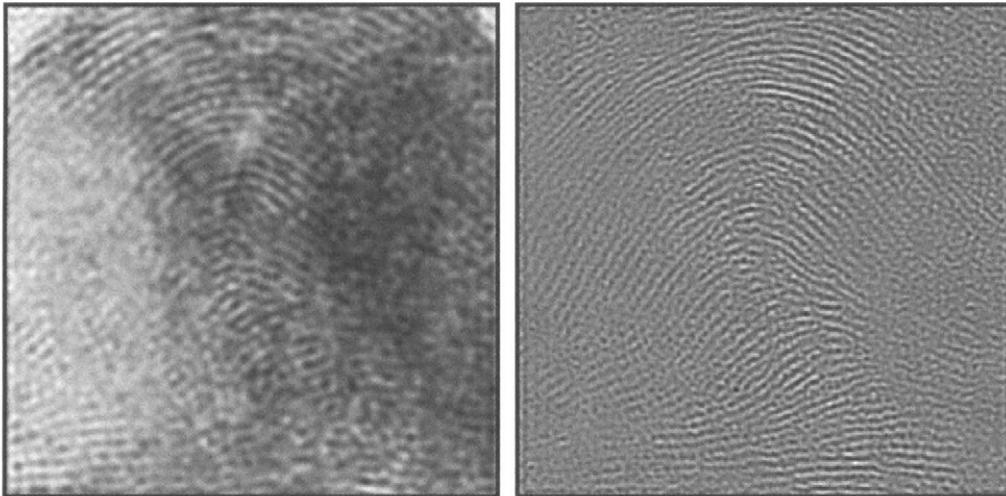


FIGURE 10 Image “fingerprint” processed with the (left) low-frequency DFT mask, and the (right) mid-frequency DFT mask.

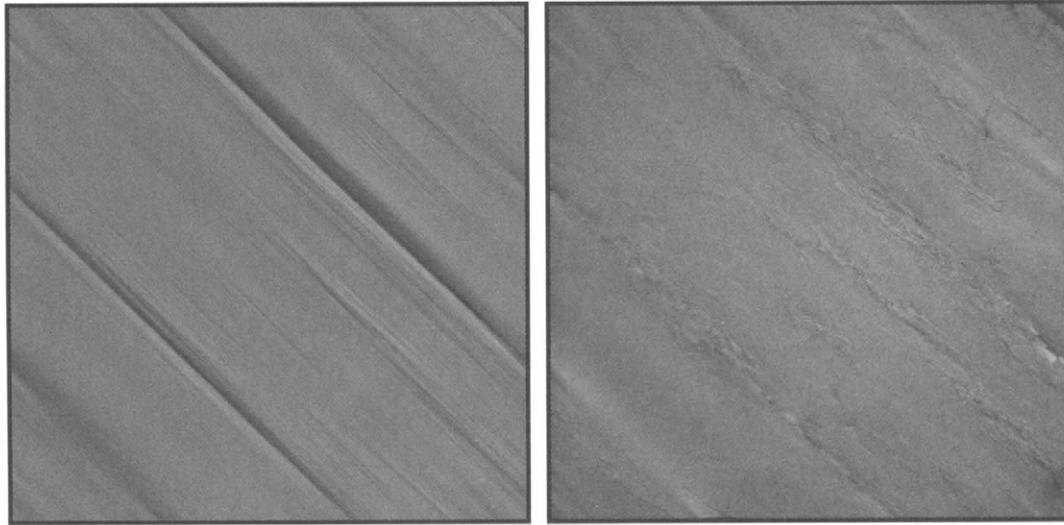


FIGURE 11 Image “peppers” processed with the (left) mid-frequency DFT mask, and the (right) high-frequency DFT mask.

mask and the mid-frequency mask. The low-frequency result looks blurred, and there is an apparent loss of information. However, the mid-frequency result seems to enhance and isolate much of the interesting ridge information about the fingerprint.

In the second example (Fig. 11), image “peppers” were treated with the mid-frequency DFT mask and the high-frequency DFT mask. The mid-frequency image is visually quite interesting since it is apparent that the sharp-intensity changes were significantly enhanced. A similar effect was produced with the higher frequency mask, but with greater emphasis on sharp details.

Frequency Orientation

The orientation of an image frequency refers to its angle. The term “orientation” applied to an image or image component

describes those aspects of the image that contribute to an appearance that is strongly characterized by the frequency orientation portrait of the DFT. If the DFT is brighter along a specific orientation, the image contains highly oriented components along that direction.

The image “fingerprint” (with DFT magnitude in Fig. 6(c)) is also an excellent example of image orientation. The DFT contains significant mid-frequency energy between the approximate orientations 45 to 135 degrees from the horizontal axis. This corresponds perfectly to the orientations of the ridge patterns in the fingerprint image.

Figure 12 shows the image “planks,” which contain a strong directional component. This manifests as a very strong extended peak extending from lower left to upper right in the DFT magnitude. Figure 13 (“escher”) exhibits several such extended peaks, corresponding to strongly oriented structures in the horizontal and slightly off-diagonal directions.

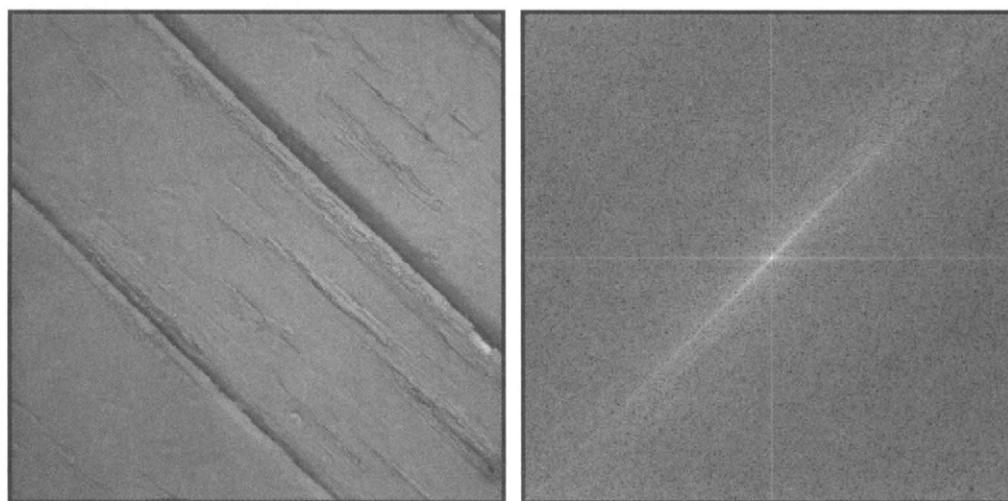


FIGURE 12 Image “planks” (left) and DFT magnitude (right).

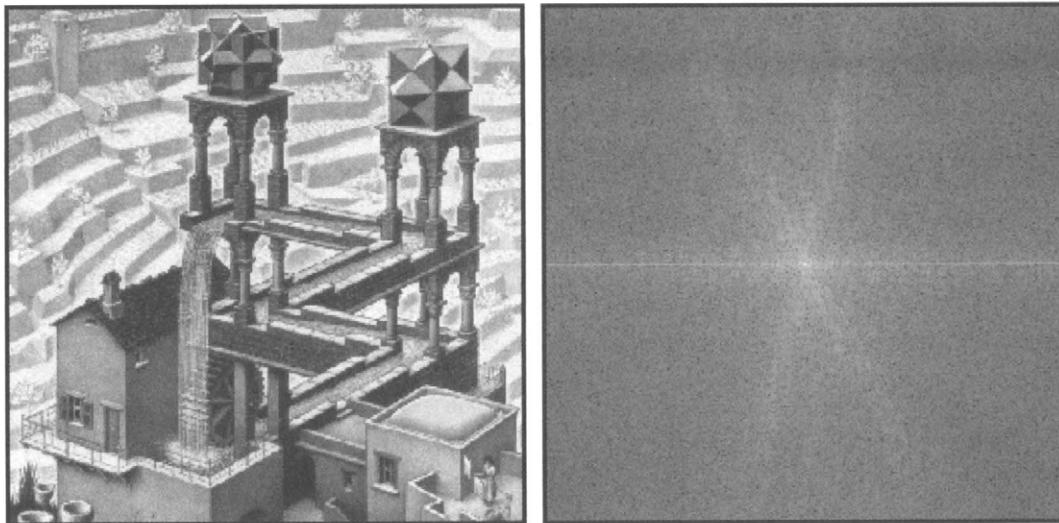


FIGURE 13 Image "escher" (left) and DFT magnitude (right).

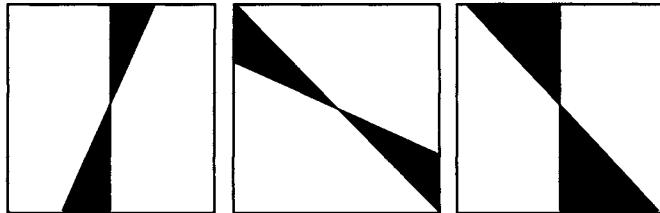
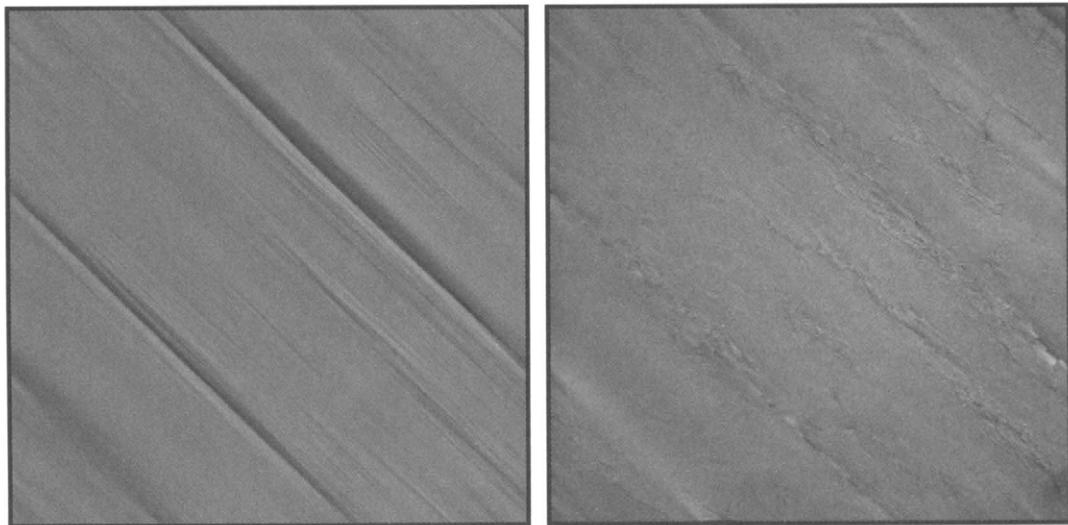


FIGURE 14 Examples of image frequency orientation masks.

Again, an instructive experiment can be developed by defining zero-one image frequency masks, this time tuned to different orientation frequency bands instead of radial frequency bands. Several such oriented frequency masks are depicted in Fig. 14.

As a first example, the DFT of the image "planks" was modified by two orientation masks. In Fig. 15 (left), an orientation mask that allows the frequencies in the range of 40 to 50 degrees only (as well as the symmetrically placed frequencies 220 to 230 degrees) was applied. This was designed to capture the bright ridge of DFT coefficients easily seen in Fig. 12. As can be seen, the strong oriented information describing the cracks in the planks and some of the oriented grain is all that remains. Possibly, this information could be used by some automated process. Then, in Fig. 15 (right), the frequencies in the much larger ranges of 50 to 220 degrees (and -130 to 40 degrees) were admitted. These are the complementary frequencies to the first range chosen, and they contain all the other information other than the strongly

FIGURE 15 Image "planks" processed with oriented DFT masks that allow frequencies in the range (measured from the horizontal axis): (left) 40° to 50° (and 220° to 230°), and (right) 50° to 220° (and -130° to 40°).

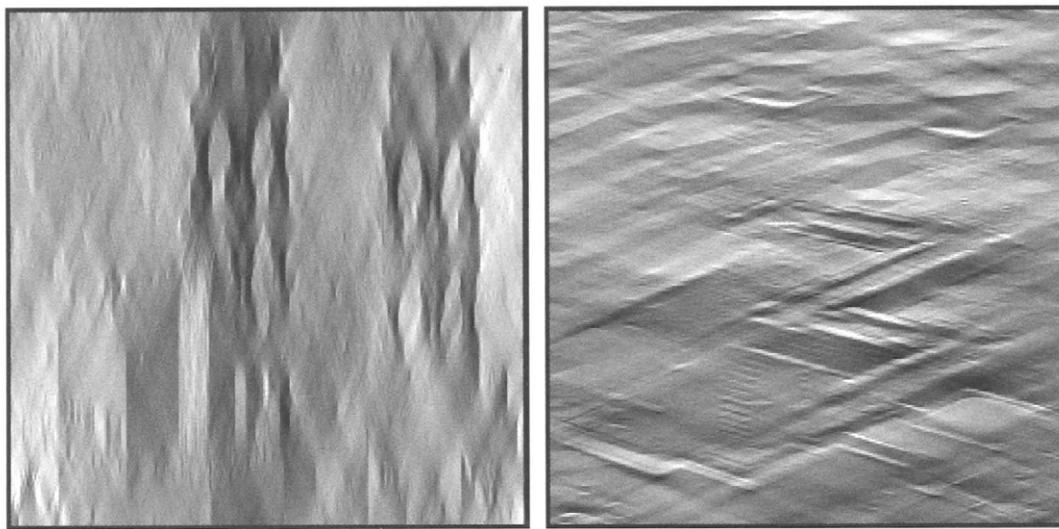


FIGURE 16 Image “escher” processed with oriented DFT masks that allow frequencies in the range (measured from the horizontal axis): (left) -25° to 25° (and 155° to 205°), and (right) 45° to 135° (and 225° to 315°).

oriented component. As can be seen, this residual image contains little oriented structure.

As a first example, the DFT of the image “escher” was also modified by two orientation masks. In Fig. 16 (left), an orientation mask that allows the frequencies in the range -25 to 25 degrees (and 155 to 205 degrees) only was applied. This captured the strong horizontal frequency ridge in the image, corresponding primarily to the strong vertical (building) structures. Then, in Fig. 16 (right), frequencies in the vertically oriented ranges 45 to 135 degrees (and 225 to 315 degrees) were admitted. This time, completely different structures were highlighted, including the diagonal waterways, the background steps, and the paddlewheel.

6 Related Topics in this Handbook

The Fourier transform is one of the most basic tools for image processing, or for that matter, the processing of any kind of signal. It appears throughout this *Handbook* in various contexts.

One topic that was not touched on in this basic chapter is the frequency-domain analysis of sampling continuous images/video to create discrete-space images/video. Understanding the relationship between the DSFT and the DFT (spectrum of digital image signals) and the continuous Fourier

transform of the original, unsampled image is basic to understanding the information content, and possible losses of information, in digital images. These topics are ably handled in Chapters 7.1 and 7.2 of this *Handbook*. Sampling issues were not covered in this chapter, because it was believed that most users deal with digital images that have been already created. Hence, the emphasis is on the immediate processing, and sampling issues are offered as a background understanding.

Fourier domain concepts and linear convolution pervade most of the chapters in Section 3 of the *Handbook*, because linear filtering, restoration, enhancement, and reconstruction all depend on these concepts. Most of the mathematic models for images and video in Section 4 have strong connections to Fourier analysis, especially the wavelet models, which extend the ideas of Fourier techniques in very powerful ways. Extended frequency-domain concepts are also heavily used in Sections 5 and 6 (image and video compression) of the *Handbook*, although the transforms used differ somewhat from the DFT.

Acknowledgment

Thanks to Dr. Hung-Ta Pai for carefully reading and commenting on this chapter.