5

Fourier transforms and frequency-domain processing

5.1 Frequency space: a friendly introduction

Grasping the essentials of frequency-space methods is very important in image processing. However, if the author's experience is anything to go by, Fourier methods can prove something of a barrier to students. Many discussions of the Fourier transform in the literature simply begin by defining it and then offer a comprehensive but rather formal mathematical presentation of the related theorems. The significant risk in this approach is that all but the mathematically inclined or gifted are quickly turned off, either feeling inadequate for the task or not understanding why it is all worth the trouble of struggling with. The *real significance* of Fourier methods, the small number of really central concepts, simply gets lost in the mathematical details. We will try to avoid this overly formal approach. Of course, we cannot simply ignore the mathematics – it is central to the whole subject – but we will certainly aim to stress the key points and to underline the real significance of the concepts we discuss.

Frequency-space analysis is a widely used and powerful methodology centred around a particular mathematical tool, namely the *Fourier transform*.[1] We can begin simply by saying that the Fourier transform is a particular type of integral transform that enables us to view imaging and image processing from an alternative viewpoint by transforming the problem to another space. In image processing, we are usually concerned with 2-D spatial distributions (i.e. functions) of intensity or colour which exist in *real space* – i.e. a 2-D Cartesian space in which the axes define units of length. The Fourier transform operates on such a function to produce an entirely equivalent form which lies in an abstract space called *frequency space*. Why bother? In the simplest terms, frequency space is useful because it can make the solution of otherwise difficult problems *much easier* (Figure 5.1).

Fourier methods are sufficiently important that we are going to break the pattern and digress (for a time) from image processing to devote some time to understanding some of the key concepts and mathematics of Fourier transforms and frequency space. Once this

[1] Strictly, frequency-space analysis is not exclusively concerned with the Fourier transform. In its most general sense, it also covers the use of similar transforms such as the Laplace and Z transform. However, we will use frequency space and Fourier space synonymously.

Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab Chris Solomon and Toby Breckon
 ISBN: 978-0-470-84472-4

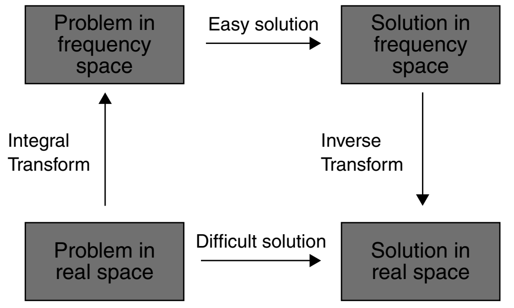

Figure 5.1 Frequency-space methods are used to make otherwise difficult problems easier to solve

foundation has been laid, we will move on to see how they can be used to excellent effect both to look at image processing from a new perspective and to carry out a variety of applications.

5.2 Frequency space: the fundamental idea

We will begin our discussion of Fourier methods by summarizing, without any attempt at rigour, some of the key concepts. To stay general, we will talk for the moment of the Fourier analysis of signals rather than images.

(1) *The harmonic content of signals.* The fundamental idea of Fourier analysis is that any signal, be it a function of time, space or any other variables, may be expressed as a weighted linear combination of *harmonic* (i.e. sine and cosine) functions having different periods or frequencies. These are called the (spatial) frequency components of the signal.

(2) *The Fourier representation is a complete alternative.* In the Fourier representation of a signal as a weighted combination of harmonic functions of different frequencies, the assigned weights constitute the *Fourier spectrum*. This spectrum extends (in principle) to infinity and any signal can be reproduced to arbitrary accuracy. Thus, the Fourier spectrum is a complete and valid, alternative representation of the signal.

(3) *Fourier processing concerns the relation between the harmonic content of the output signal to the harmonic content of the input signal.* In frequency space, signals are considered as combinations of harmonic signals. Signal processing in frequency space (analysis, synthesis and transformation of signals) is thus concerned with the constituent harmonic content and how these components are preserved, boosted or suppressed by the processing we undertake.

These first three concepts are summarized in Figure 5.2.

(4) *The space domain and the Fourier domain are reciprocal.* In the Fourier representation of a function, harmonic terms with high frequencies (short periods) are needed to construct small-scale (i.e. sharp or rapid) changes in the signal. Conversely, smooth features in the signal can be represented by harmonic terms with low frequencies (long periods). The two domains are thus *reciprocal* – small in the space domain maps to

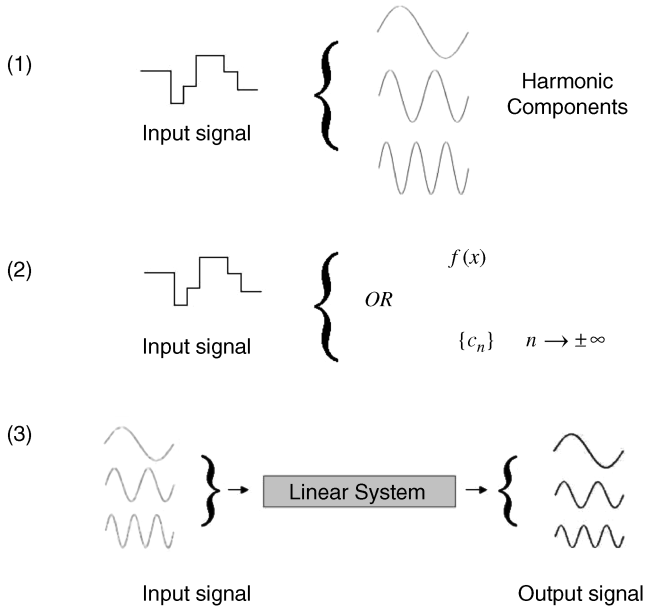

Figure 5.2 A summary of three central ideas in Fourier (frequency) domain analysis. (1) Input signals are decomposed into harmonic components. (2) The decomposition is a complete and valid representation. (3) From the frequency-domain perspective, the action of any linear system on the input signal is to modify the amplitude and phase of the input components

large in the Fourier domain and large in the space domain maps to small in the Fourier domain.

Students often cope with Fourier series quite well but struggle with the Fourier transform. Accordingly, we make one more key point.

(5) *The Fourier series expansion and the Fourier transform have the same basic goal.* Conceptually, the Fourier series expansion and the Fourier transform do *the same thing*. The difference is that the Fourier series breaks down a periodic signal into harmonic functions of *discrete* frequencies, whereas the Fourier transform breaks down a nonperiodic signal into harmonic functions of *continuously varying* frequencies. The maths is different but the idea is the same.

We will expand on these basic ideas in what follows. We begin our discussion in the next section with Fourier series.

5.2.1 The Fourier series

Key point 1

Any *periodic* signal may be expressed as a weighted combination of sine and cosine functions having different periods or frequencies.

This is Fourier's basic hypothesis. The process of breaking down a *periodic* signal as a sum of sine and cosine functions is called a *Fourier decomposition* or *Fourier expansion*. If the signal is something that varies with time, such as a voltage waveform or stock-market share price, the harmonic functions that build the signal are *1-D* and have a *temporal* frequency. In such a 1-D case, Fourier's basic theorem says that a periodic signal[2] $V(t)$ having period T can be constructed exactly as an infinite sum of harmonic functions, a Fourier series, as follows:

$$\begin{aligned} V(t) &= \sum_{n=0}^{\infty} a_n \cos\left(\frac{2\pi n t}{T}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi n t}{T}\right) \\ &= \sum_{n=0}^{\infty} a_n \cos(\omega_n t) + \sum_{n=1}^{\infty} b_n \sin(\omega_n t) \end{aligned} \tag{5.1}$$

An arbitrary periodic 1-D function of a *spatial* coordinate x $f(x)$ having spatial period λ can be represented in exactly the same way:

$$\begin{aligned} V(x) &= \sum_{n=0}^{\infty} a_n \cos\left(\frac{2\pi n x}{\lambda}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi n x}{\lambda}\right) \\ &= \sum_{n=0}^{\infty} a_n \cos(k_n x) + \sum_{n=1}^{\infty} b_n \sin(k_n x) \end{aligned} \tag{5.2}$$

Let us first make some simple observations on the Fourier expansion of the *spatial* function $f(x)$ expressed by Equation (5.2).

(1) The infinite series of harmonic functions in the expansion, namely $\cos(k_n x)$ and $\sin(k_n x)$, are called the Fourier *basis functions*.

(2) We are dealing with a function that varies in space and the (inverse) periodicity, determined by $k_n = 2\pi n/\lambda$, is thus called the *spatial frequency*.

(3) The coefficients a_n and b_n indicate how much of each basis function (i.e. harmonic wave of the given spatial frequency) is required to build $f(x)$. The complete set of coefficients $\{a_n$ and $b_n\}$ are said to constitute the *Fourier* or *frequency spectrum* of the spatial function. The function $f(x)$ itself is called the spatial domain representation.

(4) To reproduce the original function $f(x)$ *exactly*, the expansion must extend to an infinite number of terms. In this case, as $n \to \infty$, the spatial frequencies $k_n \to \infty$ and the number of coefficients $\{a_n$ and $b_n\}$ describing the Fourier spectrum also approach infinity.

[2] Strictly, the signal must satisfy certain criteria to have a valid Fourier expansion. We will not digress into these details but assume that we are dealing with such functions here.

Key point 2

The Fourier spectrum is a valid and complete alternative representation of a function.

This point is essential, namely that in knowing the coefficients a_n and b_n we have a complete representation of the function just as valid as $f(x)$ itself. This is so because we can rebuild $f(x)$ with arbitrary precision by carrying out the summation in Equation (5.2). Figure 5.3 shows how a simple 1-D periodic function (only one cycle of the function is shown) – a step function – can be increasingly well approximated by a Fourier series representation as more terms in Equation (5.2) are added. If we continue this procedure for ever, the Fourier spectrum $\{a_n, b_n\}$ can reproduce the function exactly and can thus be rightly considered an *alternative* (frequency-domain) representation.

If we examine the synthesis of the periodic square wave in Figure 5.3, we can observe our key point number 4 in the Fourier decomposition/synthesis of a function. The *low spatial frequencies* (corresponding to the lower values of n) build the 'basic' smooth shape, whereas the high spatial frequencies are required to reproduce the sharp transitions in the function (the edges). The synthesis of a step function by a Fourier series is just one simple example,

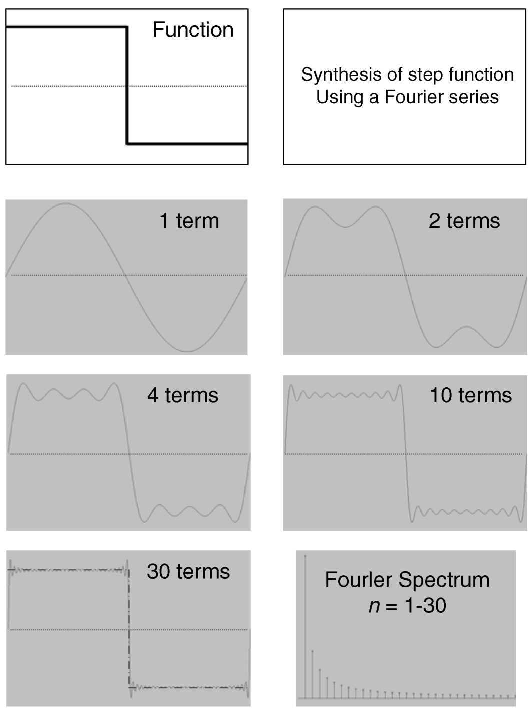

Figure 5.3 The synthesis of a step function of period λ using a Fourier series. The resulting spectrum is the frequency-domain representation of the spatial function determining the contribution of each sine wave of frequency $\sin(k_n x) = \sin(2\pi nx/\lambda)$

but this is in fact a basic 'ground rule' of the Fourier representation of *any* function (true in one or more dimensions).

It is also worth emphasizing that although the ability to synthesize a given function *exactly* requires harmonic (i.e. sine and cosine) frequencies extending to infinite spatial frequency, it is often the case that a good approximation to a function can be obtained using a *finite and relatively small number of spatial frequencies*. This is evident from the synthesis of the step function. When we approximate a spatial function by a Fourier series containing a finite number of harmonic terms N, we then effectively define a so-called spatial *frequency cut-off* $k_{\mathrm{CO}} = 2\pi N/\lambda$. The loss of the high spatial frequencies in a signal generally results in a loss of fine detail.

5.3 Calculation of the Fourier spectrum

Fourier's basic theorem states that we can synthesize periodic functions using the sinusoidal basis functions, but we have so far glossed over the question of *how we actually calculate the Fourier spectrum* (i.e. the expansion coefficients in Equation (5.2)). Fortunately, this is easy. By exploiting the *orthogonality* properties of the Fourier basis,[3] we can obtain simple formulae for the coefficients:

$$\begin{aligned} a_n &= \frac{2}{\lambda}\int_{-\lambda/2}^{\lambda/2} f(x)\cos(k_n x)\,\mathrm{d}x \\ b_n &= \frac{2}{\lambda}\int_{-\lambda/2}^{\lambda/2} f(x)\sin(k_n x)\,\mathrm{d}x \end{aligned} \tag{5.3}$$

where $k_n = 2\pi n/\lambda$. Note that we get the coefficients in each case by integrating over one full spatial period of the function.[4]

5.4 Complex Fourier series

Fourier series can actually be expressed in a more convenient and compact, complex form. Thus, a periodic, 1-D spatial function $f(x)$ is expressed as a weighted sum of *complex exponential* (harmonic) functions:

$$\begin{aligned} f(x) &= \sum_{n=-\infty}^{\infty} c_n \exp\left(\frac{\mathrm{i}2\pi n x}{\lambda}\right) \\ &= \sum_{n=-\infty}^{\infty} c_n \exp(\mathrm{i}k_n x) \end{aligned} \tag{5.4}$$

[3] A proof of these formulae is offered on the book website http://www.fundipbook.com/materials/.

[4] We can of course add *any constant value* to our chosen limits of $-\lambda/2$ and $\lambda/2$. Zero and λ are commonly quoted, but all that matters is that the periodic function is integrated over a full spatial period λ.

where n may assume all integer values from $-\infty$ to $+\infty$. In an entirely analogous way to the real form of the expansion, we may exploit the orthogonality relations between complex exponential functions to obtain the Fourier expansion coefficients:[5]

$$c_n = \frac{1}{\lambda} \int_{-\lambda/2}^{\lambda/2} f(x) \exp(\mathrm{i}k_n x) \, \mathrm{d}x \tag{5.5}$$

Note that the c_n are, in general, *complex* numbers. In using the complex representation we stress that nothing essential changes. We are still representing $f(x)$ as an expansion in terms of the real sine and cosine functions. The sines and cosines can actually be made to reappear by virtue of grouping the positive and negative exponentials with the same magnitude of the index n but of opposite sign. A strictly real function can then be constructed from the complex Fourier basis because the expansion coefficients (the c_n) are also, in general, complex. It is relatively straightforward to show that (see the exercises on the book's website[6]) that the complex coefficients c_n are related to the real coefficients a_n and b_n in the real Fourier series (Equation (5.3)) by

$$\begin{aligned} c_k &= a_k + ib_k \\ c_{-k} &= a_k - ib_k \qquad \text{for } k = 0, 1 \ldots \to \infty \end{aligned} \tag{5.6}$$

To expand the discussion of Fourier methods to deal with images, there are two main differences from the formalism we have presented so far that we must take into account:

(1) images are not, in general, periodic functions.

(2) images are typically *2-D* (and sometimes higher dimensional) spatial functions of finite support.[7]

The extension of the Fourier hypothesis from periodic functions to deal with non-periodic functions is really just the extension from a Fourier series to a Fourier transform. First, we will consider the issue of periodicity and only after extend to two dimensions. In what follows (just to keep it simple) we will consider a *1-D* spatial function $f(x)$, though an analogous argument can be made for two or more dimensions.

5.5 The 1-D Fourier transform

To move from a Fourier series to a Fourier transform, we first express our function $f(x)$ as a complex Fourier series:

[5,6] See http://www.fundipbook.com/materials/.

[7] The *support* of a function is the region over which it is nonzero. Obviously, any real digital image has finite support.

$$f(x) = \sum_{n=-\infty}^{\infty} c_n \exp(\mathrm{i}k_n x) \qquad k_n = \frac{2\pi n}{\lambda} \tag{5.7}$$

Multiplying top and bottom by $\lambda \to \infty$, this can also be written as

$$f(x) = \frac{1}{\lambda} \sum_{n=-\infty}^{\infty} (\lambda c_n) \exp(\mathrm{i}k_n x) \tag{5.8}$$

We now let the *period* of the function $\lambda \to \infty$. In other words, the trick here is to consider that *a nonperiodic function* may be considered as *a periodic function having infinite spatial period*. Note that the individual spatial frequencies in the summation expressed by Equation (5.8) are given by $k_n = 2\pi n/\lambda$ and the frequency interval between them is thus given by $\Delta k = 2\pi/\lambda$. As the spatial period $\lambda \to \infty$, the frequency interval Δk becomes infinitesimally small and λc_n tends towards a *continuous* function $F(k_x)$. It is in fact possible to show[8] that the limit of Equation (5.8) as $\lambda \to \infty$ is the *inverse Fourier transform*:

$$f(x) = \int_{-\infty}^{\infty} F(k_x) \exp(\mathrm{i}k_x x)\, \mathrm{d}k_x \tag{5.9}$$

Note that:

- the specific weights assigned to the harmonic (complex exponential) functions are given by the function $F(k_x)$;
- the frequencies k_x are now *continuous* and range over all possible values;
- The summation for *f*(x) becomes an integral and the Fourier spectrum $F(k_x)$ is now a *function* as opposed to a set of discrete values.

The orthogonality properties of the complex Fourier basis (the $\exp(\mathrm{i}k_x x)$ functions) enable us to calculate the weighting function $F(k_x)$ in Equation (5.9). This is, in fact, the *Fourier transform* of *f*(x):

$$F(k_x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x) \exp(-\mathrm{i}k_x x)\, \mathrm{d}x \tag{5.10}$$

It is vital to emphasize that the essential meaning and purpose of the Fourier transform is really no different from that of the Fourier series. The Fourier transform of a function also fundamentally expresses its decomposition into *a weighted set of harmonic functions*. Moving from the Fourier series to the Fourier transform, we move from function synthesis using weighted combinations of harmonic functions having *discrete* frequencies (a *summation*) to weighted, infinitesimal, combinations of *continuous* frequencies (an *integral*).

[8] See book website http://www.fundipbook.com/materials/ for details.

Table 5.1 Comparison of the synthesis of spatial functions using the real Fourier series, the complex Fourier series and the Fourier transform

	Real Fourier series	Complex Fourier series	Fourier transform
Spatial frequencies	$k_n = \dfrac{2\pi n}{\lambda}$ $n = 1, 2, \ldots, \infty$	$k_n = \dfrac{2\pi n}{\lambda}$ $n = \pm 1, \pm 2, \ldots, \pm\infty$	k_x $-\infty \le k_x \le \infty$
Basis functions	$\sin k_n x,\ \cos k_n x$ $n = 1, 2, \ldots, \infty$	$\exp(\mathrm{i}k_n x)$ $n = 0, \pm 1, \pm 2, \ldots, \pm\infty$	$\exp(\mathrm{i}k_x x)$ $-\infty \le k_x \le \infty$
Spectrum	Coefficients $\{a_n, b_n\}$ $n = 0, 1, 2, \ldots, \infty$	Coefficients $\{c_n\}$ $n = 0, \pm 1, \pm 2, \ldots, \pm\infty$	Function $F(k_x)$ $-\infty \le k_x \le \infty$

The synthesis of functions using the real Fourier series, the complex Fourier series and the Fourier transform are summarized in Table 5.1.

The Fourier transform $F(k_x)$ is a *complex function* and we can, therefore, also write the Fourier spectrum in polar form as the product of the Fourier modulus and (the exponential of) the Fourier phase:

$$F(k_x) = |F(k_x)| \exp \mathrm{i}\varphi(k_x)$$

where

$$|F(k_x)|^2 = [\mathrm{Re}\{F(k_x)\}]^2 + [\mathrm{Im}\{F(k_x)\}]^2 \qquad \varphi(k_x) = \tan^{-1}\left[\frac{\mathrm{Im}\{F(k_x)\}}{\mathrm{Re}\{F(k_x)\}}\right] \tag{5.11}$$

This form is useful because it helps us to see that the Fourier transform $F(k_x)$ defines both the 'amount' of each harmonic function contributing to $f(x)$ (through $|F(k_x)|$) and the relative placement/position of the harmonic along the axis through the associated complex phase term $\exp(\mathrm{i}\varphi(k_x))$. As we shall see in Figure 5.11, most of what we will loosely call the 'visual information' in a Fourier transform is actually contained in the phase part.

5.6 The inverse Fourier transform and reciprocity

Examination of Equations (5.9) and (5.10) shows that there is a close similarity between the Fourier transform and its inverse. It is in fact arbitrary whether we define the forward Fourier transform with the negative form of the exponential function or the positive form, but the convention we have chosen here is the normal one and we will use this throughout. There is also a certain freedom with regard to the 2π factor which can be placed on either the forward or reverse transforms or split between the two.[9] The factor of 2π also appears in

[9] Some authors define both forward and reverse transforms with a normalization factor of $(2\pi)^{-1/2}$ outside.

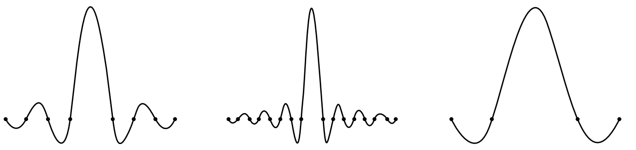

Figure 5.4 The Fourier transform of the rectangle function $F(k_x) = (a/2\pi)\{[\sin(k_x a/2)]/(k_x a/2)\}$

the 2-D Fourier transform, which is presented in the next section, but is actually *remarkably unimportant in image processing* – it enters simply as an overall scaling factor and thus has no effect on the spatial structure/content of an image. For this reason, it is quite common to neglect it entirely.

Rather like the Fourier series expansion of a square wave, the Fourier transform of the rectangle function is easy to calculate and informative. The 1-D rectangle function is defined as

$$\begin{aligned} \mathrm{rect}\left(\frac{x}{a}\right) &= 1 \qquad |x| \leq \frac{a}{2} \\ &= 0 \qquad |x| \geq \frac{a}{2} \end{aligned}$$

By substituting directly into the definition we can show that its Fourier transform is given by

$$F(k_x) = \frac{a}{2\pi}\frac{\sin(k_x a/2)}{k_x a/2} = \frac{a}{2\pi}\mathrm{sinc}\left(\frac{k_x a}{2}\right)$$

where $\mathrm{sinc}\,\theta = \sin\theta/\theta$. This function is plotted in Figure 5.4 for three values of the rectangle width a, over a spatial frequency range $-8\pi/a \leq k_x \leq 8\pi/a$. Note the reciprocity between the extent of the spatial function and the extent of its Fourier transform (the scaling factor of $2a/\pi$ has been omitted to keep the graphs the same height).

Note that a *decrease* in the width of the rectangle in the spatial domain results in the Fourier transform spreading out in the frequency domain. Similarly, an increase of the extent of the function in the spatial domain results in the frequency-domain representation shrinking. This reciprocal behaviour is a central feature of the frequency-domain representation of functions.

Important examples of 1-D Fourier transforms and Fourier transform relations are provided in Tables 5.2 and 5.3. We also offer some exercises (with worked solutions) on the book's website[10] which we strongly recommend to the reader as a means to consolidate the basic concepts discussed so far and to gaining further insight into the basic behaviour and properties of the Fourier transform.

[10] http://www.fundipbook.com/materials/.

Table 5.2 Fourier transforms of some important functions

Function name	Space domain $f(x)$ $f(x)$	Frequency domain $F(k)$ $F(k)=\int_{-\infty}^{\infty} f(x)e^{-ikx}\,dx$
Impulse (delta function)	$\delta(x)$	1
Constant	1	$\delta(k)$
Gaussian	$\exp\left(-\frac{x^2}{2\sigma^2}\right)$	$(\sigma\sqrt{2\pi})\exp\left(-\frac{\sigma^2k^2}{2}\right)$
Rectangle	$\mathrm{rect}\left(\frac{x}{L}\right)=\prod\left(\frac{x}{L}\right)\equiv\begin{cases}1 & \lvert x\rvert\le L/2\\ 0 & \text{elsewhere}\end{cases}$	$L\,\mathrm{sinc}\left(\frac{kL}{2\pi}\right)$
Triangle	$\Lambda\left(\frac{x}{W}\right)\equiv\begin{cases}1-(\lvert x\rvert/W) & \lvert x\rvert\le W\\ 0 & \text{elsewhere}\end{cases}$	$W\,\mathrm{sinc}^2\left(\frac{kW}{2\pi}\right)$
Sinc	$\mathrm{sinc}(Wx)\equiv\frac{\sin(Wx)}{Wx}$	$\frac{1}{W}\mathrm{rect}\left(\frac{k}{W}\right)$
Exponential	$e^{-a\lvert x\rvert}\quad a>0$	$\frac{2a}{a^2+k^2}$
Complex exponential	$\exp(ik_0x)$	$\delta(k-k_0)$
Decaying exponential	$\exp(-ax)u(x)\quad \mathrm{Re}\{a\}>0$	$\frac{1}{a+ik}$
Impulse train	$\sum_{n=-\infty}^{\infty}\delta(x-nx_s)$	$\frac{2\pi}{x_s}\sum_{k=-\infty}^{\infty}\delta\left[k\left(1-\frac{2\pi}{x_s}\right)\right]$
Cosine	$\cos(k_0x+\theta)$	$e^{i\theta}\delta(k-k_0)+e^{-i\theta}\delta(k+k_0)$
Sine	$\sin(k_0x+\theta)$	$-i[e^{i\theta}\delta(k-k_0)-e^{-i\theta}\delta(k+k_0)]$
Unit step	$u(x)\equiv\begin{cases}1 & x\ge 0\\ 0 & x<0\end{cases}$	$\pi\delta(k)+\frac{1}{jik}$
Signum	$\mathrm{sgn}(x)\equiv\begin{cases}1 & x\ge 0\\ -1 & x<0\end{cases}$	$\frac{2}{ik}$
Sinc^2	$\mathrm{sinc}^2(Bx)$	$\frac{1}{B}\Lambda\left(\frac{\omega}{2\pi B}\right)$
Linear decay	$1/x$	$-i\pi\,\mathrm{sgn}(k)$

5.7 The 2-D Fourier transform

In an entirely analogous way to the 1-D case, a function of two spatial variables $f(x,y)$ can be expressed as a weighted superposition of 2-D harmonic functions.[11] The basic concept is

[11] Certain functions do not, in the strict sense, possess Fourier transforms, but this is a mathematical detail which has no real practical importance in image processing and we will fairly assume that we can obtain the Fourier transform of any function of interest.

Table 5.3 Some important properties of the 2-D Fourier transform: $f(x, y)$ has a Fourier transform $F(k_x, k_y)$; $g(x, y)$ has a Fourier transform $G(k_x, k_y)$

Property	Spatial domain	Frequency domain	Comments
Addition theorem	$f(x,y)+g(x,y)$	$F(k_x,k_y)+G(k_x,k_y)$	Fourier transform of the sum equals sum of the Fourier transforms.
Similarity theorem	$f(ax,by)$	$\frac{1}{\lvert ab\rvert}F\left(\frac{k_x}{a},\frac{k_y}{b}\right)$	The frequency-domain function scales in inverse proportion to the spatial-domain function.
Shift theorem	$f(x-a,y-b)$	$\exp[-\mathrm{i}(k_x a+k_y b)]F(k_x,k_y)$	Shift a function in space and its transform is multiplied by a pure phase term.
Convolution theorem	$f(x,y)*g(x,y)$	$F(k_x,k_y)G(k_x,k_y)$	Transform of a convolution is equal to the product of the individual transforms.
Separable product	$f(x,y)=h(x)g(y)$	$F(k_x,k_y)=H(k_x)G(k_y)$	If a 2-D function separates into two 1-D functions, then so does its Fourier transform
Differentiation	$\frac{\partial}{\partial x^m}\frac{\partial}{\partial y^n}f(x,y)$	$(\mathrm{i}k_x)^m(\mathrm{i}k_y)^n F(k_x,k_y)$	Calculation of image derivatives is trivial using the image Fourier transform
Rotation	$f(x\cos\theta+y\sin\theta,$ $-x\sin\theta+y\cos\theta)$	$F(k_x\cos\theta+k_y\sin\theta,$ $-k_x\sin\theta+k_y\cos\theta)$	Rotate a function by θ in the plane and its Fourier transform rotates by θ
Parseval's theorem	$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\lvert f(x,y)\rvert^2\,\mathrm{d}x\,\mathrm{d}y$	$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\lvert F(k_x,k_y)\rvert^2\,\mathrm{d}k_x\,\mathrm{d}k_y$	Fourier transformation preserves image 'energy'
Laplacian	$-\nabla^2 f(x,y)$	$(k_x^2+k_y^2)F(k_x,k_y)$	Calculation of the image Laplacian is trivial using the image Fourier transform
Multiplication	$f(x,y)\cdot g(x,y)$	$F(k_x,k_y)**G(k_x,k_y)$	Product of functions is a convolution in the Fourier domain

graphically illustrated in Figure 5.5, in which examples of 2-D harmonic functions are combined to synthesize $f(x, y)$. Mathematically, the reverse or *inverse* 2-D Fourier transform is defined as

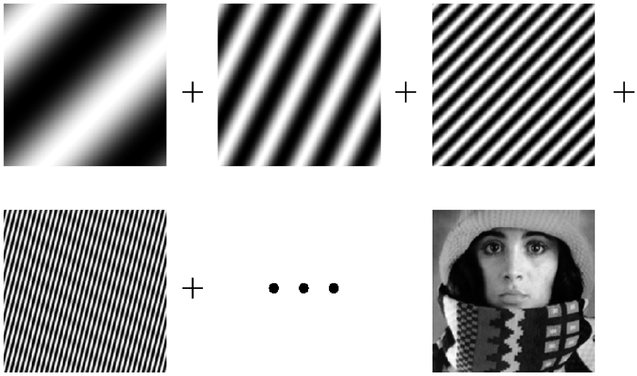

Figure 5.5 The central meaning of the 2-D Fourier transform is that some scaled and shifted combination of the 2-D harmonic basis functions (some examples are shown) can synthesize an arbitrary spatial function

$$\begin{aligned} f(x,y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(f_x,f_y)\exp[2\pi \mathrm{i}(f_x x + f_y y)]\, \mathrm{d}f_x\, \mathrm{d}f_y \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(k_x,k_y)\exp[\mathrm{i}(k_x x + k_y y)]\, \mathrm{d}k_x\, \mathrm{d}k_y \end{aligned} \tag{5.12}$$

where the weighting function $F(k_x, k_y)$ is called the *2-D Fourier transform* and is given by

$$\begin{aligned} F(f_x,f_y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)\exp[-2\pi \mathrm{i}(f_x x + f_y y)]\, \mathrm{d}x\, \mathrm{d}y \\ F(k_x,k_y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)\exp[-\mathrm{i}(k_x x + k_y y)]\, \mathrm{d}x\, \mathrm{d}y \end{aligned} \tag{5.13}$$

Considering the 2-D Fourier transform purely as an integral to be ‘worked out’, it can at first appear rather formidable.[12] Indeed, the number of functions whose Fourier transform can be *analytically* calculated is relatively small and, moreover, they tend to be rather simple functions (some are listed in Table 5.2). We stress two things, however. First, Fourier transforms of even very complicated functions *can* be calculated accurately and quickly on a computer. In fact, the development and digital implementation of the Fourier transform for the computer (known as the fast Fourier transform (FFT)) revolutionized the world of

[12] One of the authors freely confesses that he never understood it at all as an undergraduate, despite encountering it in several different courses at various times.

image processing and indeed scientific computing generally. Second, the real value of the frequency-domain representation lies not just in the ability to calculate such complex integrals numerically, but much more in the *alternative and complementary viewpoint* that the frequency domain provides on the processes of image formation and image structure.

Just as for the 1-D case, there are a relatively small number of simple 2-D functions whose Fourier transform can be analytically calculated (and a number of them are, in fact, separable into a product of two 1-D functions). It is instructive, however, to work through some examples, and these are provided with worked solutions on the book's website.[13]

There are a number of theorems relating operations on a 2-D spatial function to their effect on the corresponding Fourier transform which make working in the frequency domain and swapping back to the spatial domain much easier. Some of these are listed in Table 5.3. Again their formal proofs are provided on the book's website for the interested reader.

5.8 Understanding the Fourier transform: frequency-space filtering

In spatial-domain image processing, we are basically concerned with how imaging systems and filters of various kinds affect the individual pixels in the image. In frequency-domain image processing, however, we consider imaging systems and filtering operations from an alternative perspective – namely, *how they affect the constituent harmonic components that make up the input.* Figure 5.6 depicts the basic ideas of spatial and frequency-space

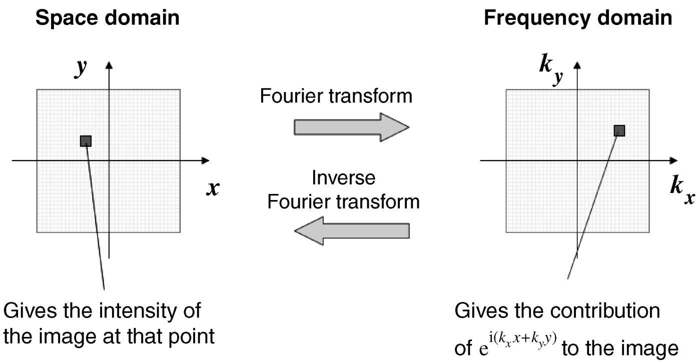

Figure 5.6 The relation between the space domain and the frequency domain. The value at a point (x, y) in the space domain specifies the intensity of the image at that point. The (complex) value at a point (k_x, k_y) in the frequency domain specifies the contribution of the harmonic function $\exp[i(kx_x + ky_y)]$ to the image

[13] See http://www.fundipbook.com/materials/.

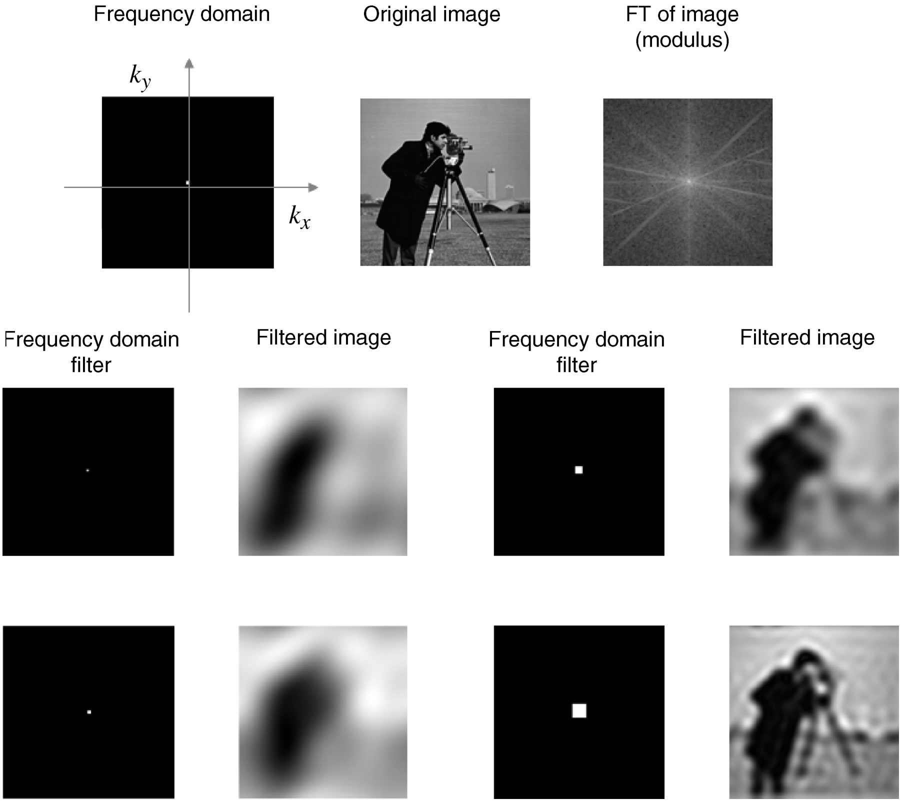

Figure 5.7 A basic illustration of frequency-domain filtering. The Fourier transform of the original image is multiplied by the filters indicated above (white indicates spatial frequency pairs which are preserved and black indicates total removal). An inverse Fourier transform then returns us to the spatial domain and the filtered image is displayed to the right. (The Matlab code for this figure can be found at http://www.fundipbook.com/materials/.)

representation. In the spatial domain, we refer to the pixel locations through a Cartesian (x, y) coordinate system. In the frequency-space representation the value at each coordinate point in the (k_x, k_y) system tells us the contribution that the harmonic frequency component $\exp[i(k_x x + k_y y)]$ makes to the image.

Figure 5.7 demonstrates the basic idea behind *frequency-domain filtering*, in which certain harmonic components are removed from the image. The basic filtering procedure involves three steps: (i) calculate the Fourier transform; (ii) suppress certain frequencies in the transform through multiplication with a *filter* function (in this case, the filter is set to zero at certain frequencies but is equal to one otherwise); and then (iii) calculate the inverse Fourier transform to return to the spatial domain. In Figure 5.7, the image and (the modulus) of its corresponding Fourier transform are displayed alongside the image which results when we remove selected groups of harmonic frequencies from the input. This removal is achieved by multiplication of the filter function image (where white is 1, black is 0) with the Fourier transform of the image. Components of increasing spatial frequency (low to high) are thus transmitted as we increase the size of the filter function centred on the Fourier image.

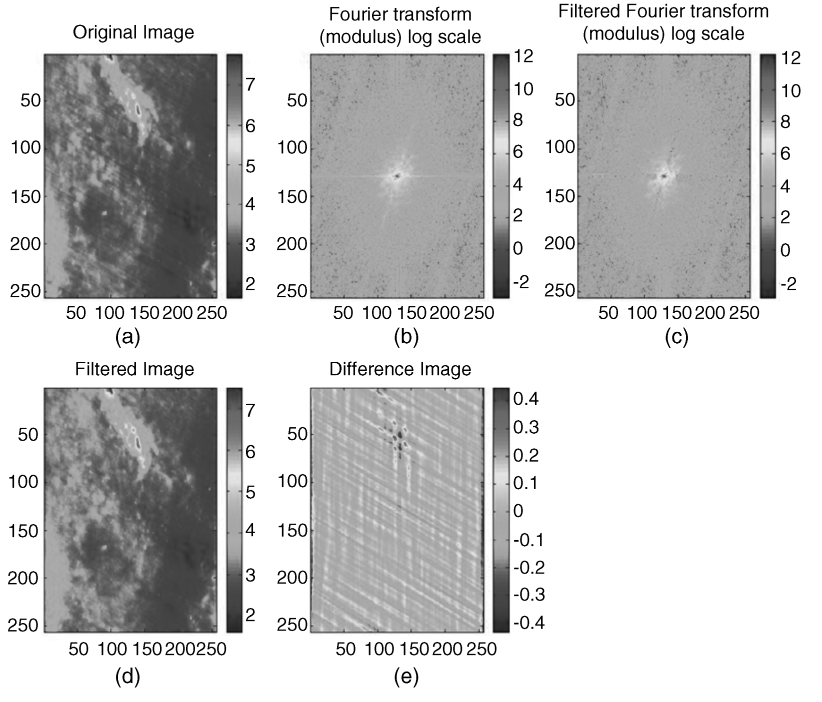

Figure 5.8 An application of frequency-domain filtering. Proceeding from left to right and top to bottom, we have: (a) the original image with striping effect apparent; (b) the Fourier modulus of the image (displayed on log scale); (c) the Fourier modulus of the image after filtering (displayed on log scale); (d) the filtered image resulting from recombination with the original phase; (e) the difference between the original and filtered images (*See colour plate section for colour version*)

Figure 5.8 shows a real-life example of frequency-domain filtering undertaken by one of the authors. The Infrared Astronomical Satellite (IRAS) was a joint project of the USA, UK and the Netherlands.[14] The IRAS mission performed a sensitive all-sky survey at wavelengths of 12, 25, 60 and 100 μm. As a result of this survey, we were presented with acquired images which suffered from striping artefacts – a result of the difficulty in properly aligning and registering the strips recorded as the detector effectively made 1-D sweeps across the sky. The stripes which are visible in the image can also be clearly seen in the Fourier transform of the image, where they show up as discrete groups of spatial frequencies which are enhanced over the background. Note that we show the absolute value of the Fourier transform on a *log* scale to render them visible. Clearly, if we can suppress these 'rogue' frequencies in frequency space, then we may expect to largely remove the stripes from the original image.

Simply setting all the identified rogue frequencies to zero is one option, but is not really satisfactory. We would expect a certain amount of the rogue frequencies to be naturally present in the scene and such a tactic will reduce the overall power spectrum of the image.

[14] See http://irsa.ipac.caltech.edu/IRASdocs/iras.html.

The basic approach taken to filtering was as follows:

(1) Calculate the Fourier transform and separate it into its modulus and phase.

(2) Identify the 'rogue frequencies'. Whilst, in principle, this could be achieved manually on a single image, we sought an automated approach that would be suitable for similarly striped images. We must gloss over the details here, but the approach relied on use of the Radon transform to identify the directionality of the rogue frequencies.

(3) Replace the rogue frequencies in the Fourier modulus with values which were statistically reasonable based on their neighbouring values.

(4) Recombine the filtered modulus with the original phase and perform an inverse Fourier transform.

5.9 Linear systems and Fourier transforms

From the frequency-domain perspective, the action of a linear imaging system on an input can be easily summarized:

> Any input image can be decomposed into a weighted sum of harmonic functions. The action of a linear system in general will be to preserve or alter the magnitude and phase of the input harmonics.

Broadly speaking, we may thus view a linear imaging system as something that operates on the constituent input harmonics of the image and can assess its quality by its ability to (more or less) faithfully transmit the input harmonics to the output. This basic action of a linear system in the frequency domain is illustrated in Figure 5.9. One measure for characterizing the performance of a linear, shift-invariant imaging system is through the *optical transfer function* (OTF). To understand this fully, we must first introduce the convolution theorem.

5.10 The convolution theorem

As we showed earlier in this chapter, the importance of the convolution integral originates from the fact that many situations involving the process of physical measurement with an imperfect instrument can be accurately described by convolution. The process of convolution (a rather messy integral in the spatial domain) has a particularly simple and convenient form in the frequency domain; this is provided by the famous *convolution theorem* – probably the single most important theorem in the field of image processing.

Consider two 2-D functions $f(x, y)$ and $h(x, y)$ having Fourier transforms respectively denoted by $F(k_x, k_y)$ and $H(k_x, k_y)$. Symbolically denoting the operation of taking a 2-D Fourier transform by $\mathbf{F}$, the *first form* of the convolution theorem states that:

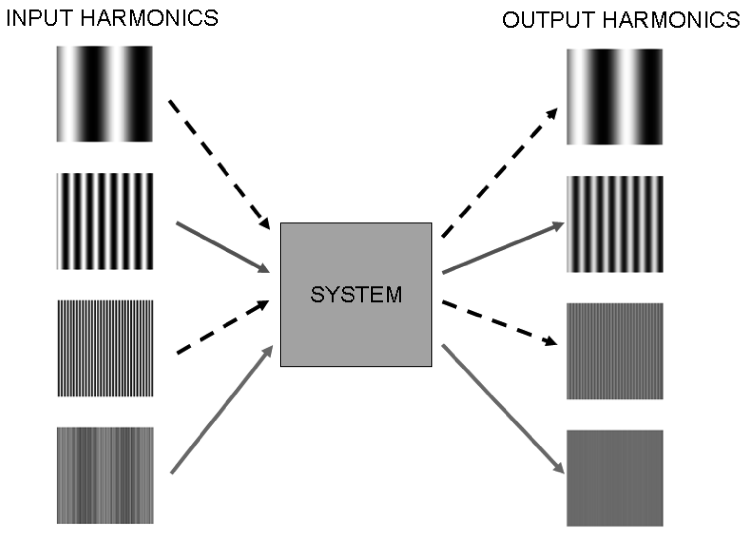

Figure 5.9 The basic action of a linear system can be understood by how well it transmits to the output each of the constituent harmonic components that make up the input. In this illustration, the lower frequencies are faithfully reproduced, but higher frequencies are suppressed.

$$\mathbf{F}\{f(x,y) ** h(x,y)\} = F(k_x,k_y)H(k_x,k_y) \tag{5.14}$$

The Fourier transform of the convolution of the two functions is equal to the product of the individual transforms.

Thus, the processing of convolving two functions in the spatial domain can be equivalently carried out by *simple multiplication of their transforms in the frequency domain*. This first form of the convolution theorem forms the essential basis for the powerful methods of *frequency-domain filtering*. Thus, rather than attempt to operate directly on the image itself with a chosen spatial-domain filter (e.g. edge detection or averaging), we alternatively approach the problem by considering *what changes we would like to effect on the spatial frequency content of the image*. Filter design is often much easier in the frequency domain, and the alternative viewpoint of considering an image in terms of its spatial frequency content often allows a more subtle and better solution to be obtained.

For completeness, we note that there is a second form of the convolution theorem, which states that

$$\mathbf{F}\{f(x,y)h(x,y)\} = F(k_x,k_y) ** H(k_x,k_y) \tag{5.15}$$

The Fourier transform of the product of the two functions is equal to the convolution of their individual transforms.

This form does not find quite such widespread use in digital image processing as it does not describe the basic image formation process. However, it finds considerable use in the fields of Fourier optics and optical image processing, where we are often interested in the effect of devices (diffraction gratings, apertures, etc.) which act multiplicatively on the incident light field.

The convolution theorem[15] lies at the heart of both frequency-domain enhancement techniques and the important subject of image restoration (a subject we will develop in more detail in Chapter 6).

5.11 The optical transfer function

Consider for a moment the specific imaging scenario in which $f(x, y)$ corresponds to the input distribution, $h(x, y)$ to the respective system PSF and $g(x, y)$ is the image given by their convolution:

$$g(x, y) = f(x, y) ** h(x, y) \tag{5.16}$$

Taking the Fourier transform of both sides, we can use the first form of the convolution theorem to write the right-hand side:

$$\begin{aligned} \mathbf{F}\{g(x, y)\} &= \mathbf{F}\{f(x, y) ** h(x, y\} \\ G(k_x, k_y) &= F(k_x, k_y)H(k_x, k_y) \end{aligned} \tag{5.17}$$

Thus, the Fourier spectrum of the output image $G(k_x, k_y)$ is given by the product of the input Fourier spectrum $F(k_x, k_y)$ with a multiplicative filter function $H(k_x, k_y)$. $H(k_x, k_y)$ is called the OTF. The OTF is the frequency-domain equivalent of the PSF. Clearly, the OTF derives its name from the fact that it determines how the individual spatial frequency pairs (k_x, k_y) are transferred from input to output. This simple interpretation makes the OTF the most widely used measure of the quality or fidelity of a linear shift-invariant imaging system.

$$\mathbf{F}\{f(x, y) * h(x, y\} = \underbrace{G(k_x, k_y)}_{\substack{\text{output} \\ \text{Fourier} \\ \text{spectrum}}} = \underbrace{F(k_x, k_y)}_{\substack{\text{input} \\ \text{Fourier} \\ \text{spectrum}}} \underbrace{H(k_x, k_y)}_{\text{OTF}} \tag{5.18}$$

This multiplicative property of the OTF on the input spectrum is particularly convenient whenever we consider complex imaging systems comprising multiple imaging elements (e.g. combinations of lenses and apertures in a camera or telescope). If the kth element is characterized by its PSF $h_k(x, y)$, then the overall image is given by a sequence of convolutions of the input with the PSFs. Taking Fourier transforms and using the convolution theorem, this can be equivalently expressed by sequential *multiplication* of the OTFs in the frequency domain- a much easier calculation:

$$\mathbf{F}\{h_1(x, y) * h_2(x, y) * \cdots h_N(x, y)\} = H_1(k_x, k_y)H_2(k_x, k_y) \cdots H_N(k_x, k_y) \tag{5.19}$$

[15] For proof, go to http://www.fundipbook.com/materials/.

A detailed discussion of OTFs and their measurement is outside our immediate scope, but we stress two key points:

(1) The OTF is normalized to have a maximum transmission of unity. It follows that an *ideal imaging system* would be characterized by an OTF given by $H(k_x, k_y) = 1$ for all spatial frequencies.

(2) As the Fourier transform of the PSF, the OTF is, in general, a *complex* function:

$$H(k_x, k_y) = |H(k_x, k_y)| \exp[i\varphi(k_x, k_y)]$$

The square modulus of the OTF is a real function known as the modulation transfer function (MTF); this gives the magnitude of transmission of the spatial frequencies. However, it is the phase term $\varphi(k_x, k_y)$ which controls the position/placement of the harmonics in the image plane. Although the MTF is a common and useful measure of image quality, the phase transmission function $\varphi(k_x, k_y)$ is also crucial for a complete picture.

Figure 5.10 (generated using the Matlab® code in Example 5.1) shows on the left an image of the old BBC test card.[16] The test card has a number of features designed to allow

Example 5.1

Matlab code	**What is happening?**
A=imread('BBC_grey_testcard.png');	%Read in test card image
FA=fft2(A); FA=fftshift(FA);	%Take FFT and centre it
PSF=fspecial('gaussian',size(A),6);	%Define PSF
OTF=fft2(PSF); OTF=fftshift(OTF);	%Calculate corresponding OTF
Afilt=ifft2(OTF.*FA); Afilt=fftshift(Afilt);	%Calculate filtered image
subplot(1,4,1);imshow(A,[]); colormap(gray);	%Display results
subplot(1,4,2); imagesc(log(1 + (PSF))); axis image; axis off;	
subplot(1,4,3); imagesc(log(1 + abs (OTF))); axis image; axis off;	
subplot(1,4,4); imagesc(abs(Afilt)); axis image; axis off;	
PSF=fspecial('gaussian',size(A),6);	%Define PSF
OTF=fft2(PSF); OTF=fftshift(OTF);	%Calculate corresponding OTF
rlow=(size(A,1)./2)-3; rhigh= (size(A,1)./2) + 3;	%Define range to be altered
clow=(size(A,2)./2)-3; chigh= (size(A,2)./2) + 3;	

[16] For those readers too young to remember this, the test card was displayed when you switched on your TV set before transmission of programmes had started. This was back in the days when television often did not start until 11 am.

```
Fphase=angle(OTF);                          %Extract Fourier phase
Fphase(rlow:rhigh,clow:chigh)=              %Add random component to selected phase
    Fphase(rlow:rhigh,clow:chigh) +
    0.*pi.*rand;

OTF=abs(OTF).*exp(i.*Fphase);               %Recombine phase and modulus
Afilt=ifft2(OTF.*FA);                       %Calculate filtered image
    Afilt=fftshift(Afilt);
psfnew=abs(fftshift((otf2psf(OTF))));       %Calculate corresponding PSF
subplot(1,4,2); imagesc(log(1 + psfnew));
    axis image; axis off; colormap(gray);
subplot(1,4,3); imagesc(log(1 +
    abs(OTF))); axis image; axis off;
subplot(1,4,4); imagesc(abs(Afilt));
    axis image; axis off;

PSF=fspecial('motion',30,30);               %Define motion PSF
OTF=psf2otf(PSF,size(A));                   %Calculate corresponding OTF
    OTF=fftshift(OTF);
Afilt=ifft2(OTF.*FA);                       %Calculate filtered image
subplot(1,4,1);imshow(A,[]);                %Display results
subplot(1,4,2); imshow(log(1 + PSF),[]);
subplot(1,4,3); imshow(log(1 + abs
    (OTF)),[])
subplot(1,4,4); imshow(abs(Afilt),[]);
```

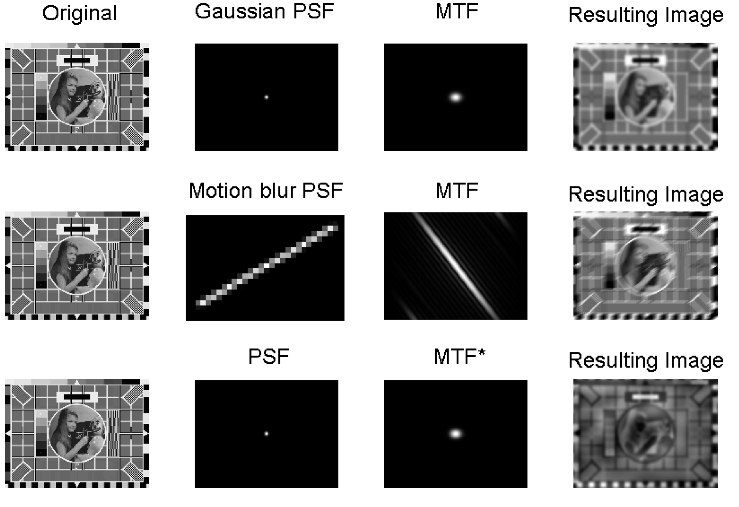

Figure 5.10 The effect of the OTF on the image quality. Column 1: the original image; column 2: the system PSF; column 3: the corresponding system MTF; column 4: the resulting images after transfer by the system OTF. Note the image at bottom right has the same MTF as that at the top right, but the phase has been shifted and this has significant consequences on the image quality

assessment of image quality. For example, hard edges, sinusoidal patterns of both low and high frequency and adjacent patches of slightly different intensities are all evident. These features allow systematic measurements to be made on the OTF and also the image contrast to be assessed. The top row shows the effect on the test card image of transmission through a system with an OTF well approximated by a 2-D Gaussian function. The PSF (also a Gaussian) corresponding to the OTF is also shown. Note that the overall effect on the image is an isotropic blurring and most of the fine structures in the original image are unresolved. The middle row shows the OTF corresponding to a *motion* of the source or detector during image acquisition. The corresponding PSF, a long thin line, is displayed (on a finer scale) indicating a motion of approximately 30 pixels at an angle of 30°. Note that the MTF extends to high spatial frequencies in the direction orthogonal to the motion but drops rapidly in the direction of the motion, and this substantiates what we have already discussed concerning the reciprocal behaviour in the space and spatial frequency domain in which short signals in the space domain become extended in the frequency domain and vice versa. The bottom row shows the effect of an OTF whose MTF is identical to that in the first row but to which *a random phase term has been added to some of the low-frequency components of the image*. The drastic effect on the image quality is apparent.

5.12 Digital Fourier transforms: the discrete fast Fourier transform

In our discussion, we have been a little disingenuous through our silence on a key computational issue. We commented earlier that Fourier transforms can be carried out quickly and effectively on a digital computer thanks to the development of a Fourier transform which works on discrete or digital data – the discrete Fast Fourier Transform (FFT) – but have offered no discussion of it. For example, we have implicitly assumed in the examples presented so far that the Fourier transform of an image (which has a finite number of pixels and thus spatial extent) *also has finite extent*. Moreover, we have implicitly made use of the fact that an $N \times N$ digital image in the spatial domain will transform to a corresponding $N \times N$ frequency-domain representation. This is true, but certainly not obvious. In fact, we have already seen that the continuous Fourier transform of a function having finite spatial extent (e.g. a delta or rectangle function) generally has *infinite* extent in the frequency domain. Clearly, then, there are issues to be resolved. In this chapter we will not attempt to offer a comprehensive treatment of the discrete FFT. This is outside our immediate concern and many excellent treatments exist already.[17] In the simplest terms, we can say that the discrete FFT is just an adaptation of the integral transform we have studied which preserves its essential properties when we deal with discrete (i.e. digital) data. The following section can be considered optional on a first reading, but it is included to ensure the reader is aware of some practical aspects of working with the discrete FFT.

[17] For example, see *The Fast Fourier Transform* by E. Oran Brigham or *Fourier Analysis and Imaging* by Ron Bracewell.

5.13 Sampled data: the discrete Fourier transform

Digital images are by definition *discrete*. A digital image which consists of a (typically) large but finite number of pixels must, by definition, have *finite support*. This immediately suggests one potential problem for using Fourier transforms in digital image processing, namely: how can we possibly represent a Fourier transform having *infinite support in a finite array of discrete pixels?* The answer is that we cannot, but there is, fortunately, an answer to this problem. The *discrete Fourier transform* (DFT) – and its inverse – calculates a frequency-domain representation of finite support from a discretized signal of finite support.

The theorems that we have presented and the concepts of frequency space that we have explored for the continuous Fourier transform are identical and carry directly over to the DFT. However, there is one important issue in using the DFT on discrete data that we need to be fully aware of.[18] This relates to the *centring of the DFT*, i.e. the centre of the Fourier array corresponding to the spatial frequency pair $(0, 0)$. Accordingly, we offer a brief description of the DFT in the following sections.

Consider a function $f(x, y)$ which we wish to represent discretely by an $M \times N$ array of sampled values. In general, the samples are taken at arbitrary but regularly spaced intervals Δx and Δy along the x and y axes. We will employ the notation $f(x, y)$ for the discrete representation on the understanding that this actually corresponds to samples at coordinates $x_0 + x\Delta x$ and $y_0 + y\Delta y$:

$$f(x, y) \triangleq f(x_0 + x\Delta x, y_0 + y\Delta y) \tag{5.20}$$

where x_0 and y_0 are the chosen starting point for the sampling and the indices x and y assume integer values $x = 0, 1, 2, \ldots, M-1$ and $y = 0, 1, 2, \ldots, N-1$ respectively.

The 2-D (forward) DFT of the $M \times N$ array $f(x, y)$ is given by the expression

$$F(u, v) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left[-2\pi i\left(\frac{ux}{M} + \frac{vy}{N}\right)\right] \tag{5.21}$$

Note that

- the DFT is also of dimension $M \times N$;
- the spatial frequency indices also assume integer values $u = 0, 1, 2, \ldots, M-1$ and $v = 0, 1, 2, \ldots, N-1$.

The discrete transform $F(u, v)$ actually corresponds to sampling spatial frequency pairs $(u\Delta u, v\Delta v)$, i.e.$\triangleq F(u, v)F(u\Delta u, v\Delta v)$, where the sampling intervals in the frequency domain $\{\Delta u, \Delta v\}$ are related to the spatial sampling interval by

$$\Delta u = \frac{1}{M\Delta x} \qquad \Delta v = \frac{1}{N\Delta v} \tag{5.22}$$

[18] It is easy to be caught out by this issue when using computer FFT routines, including the Matlab functions ***fft***, ***fft2*** and ***fftn***.

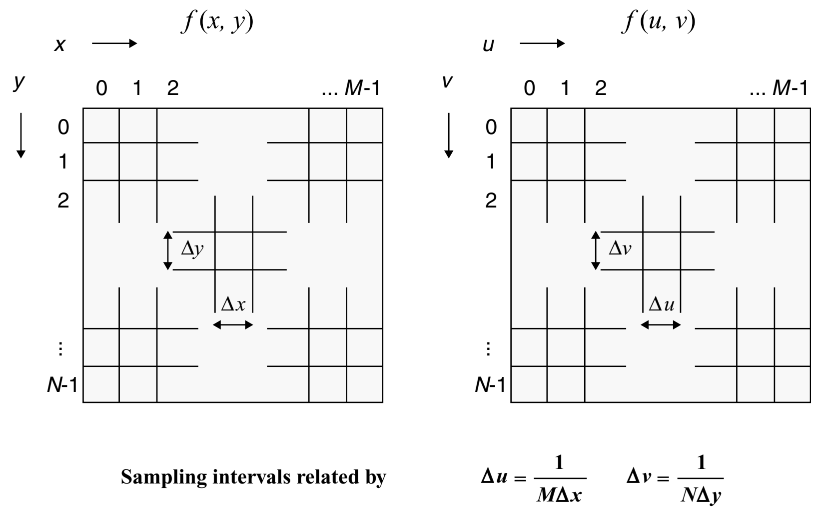

Figure 5.11 A discretely sampled image $f(x, y)$ and its DFT $F(u, v)$ have the same number of pixels. The relationship between the sampling intervals in the two domains is given in the diagram

The inverse DFT (reverse DFT) is defined in a similar manner as

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[+2\pi i\left(\frac{ux}{M} + \frac{vy}{N}\right)\right] \tag{5.23}$$

Note that the only difference in the forward and reverse transforms is the sign in the exponential. It is possible to show that the forward and reverse DFTs are *exact inverses* of each other. Equations (5.21) and (5.23) thus represent an exact transform relationship which maintains finite support in both the spatial and frequency domains as required. Figure 5.11 illustrates graphically the basic sampling relationship between a 2-D digital image represented by $M \times N$ pixels and its DFT.

5.14 The centred discrete Fourier transform

The definition of the DFT in Equation (5.21) and the diagram in Figure 5.11 indicate that the spatial frequency coordinates run from the origin at the top left corner of the array, increasing as we move across right and down. It is usual practice *to centre the DFT* by shifting its origin to the centre of the array.

For clarity, we repeat the definition for the DFT given earlier of the 2-D *discrete* array $f(x, y)$:

$$F(u,v) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp\left[-2\pi i\left(\frac{ux}{M} + \frac{vy}{N}\right)\right] \tag{5.24}$$

We now shift the frequency coordinates to new values $u' = u-(M/2)$, $v' = v-(N/2)$ so that $(u', v') = (0, 0)$ is at the centre of the array. The centred Fourier transform $F(u', v')$ is defined as

$$
\begin{aligned}
F(u', v') &= F\left(u-\frac{M}{2}, v-\frac{N}{2}\right) \\
&= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left(-2\pi i\left\{\frac{[u-(M/2)]x}{M} + \frac{[v-(N/2)]y}{N}\right\}\right)
\end{aligned}
$$

Factoring out the exponential terms which are independent of u and v and using $e^{i\pi} = -1$, we have

$$
\begin{aligned}
F(u', v') &= F\left(u-\frac{M}{2}, v-\frac{N}{2}\right) \\
&= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[(-1)^{x+y} f(x, y)\right] \exp\left[-2\pi i\left(\frac{ux}{M} + \frac{vy}{N}\right)\right]
\end{aligned}
\tag{5.25}
$$

which is, by definition, the DFT of the product $(-1)^{x+y} f(x, y)$. Thus, we have one simple way to achieve a centred Fourier transform:

Centred DFT: method 1

If $f(x, y)$ is an $M \times N$ array, then its *centred DFT* is given by the DFT of $(-1)^{x+y} f(x, y)$:

$$
F(u', v') \equiv F\left(u-\frac{M}{2}, v-\frac{N}{2}\right) = \mathbf{F}_{\mathrm{T}}\left\{(-1)^{x+y} f(x, y)\right\} \tag{5.26}
$$

where $\mathbf{F}_{\mathrm{T}}$ symbolically represents the DFT operation.

However, rather than premultiply our function by the factor $(-1)^{x+y}$, we can also achieve a centred DFT 'retrospectively'. Assume that we have calculated the DFT $F(u, v)$ of $f(x, y)$ using the standard definition. It is possible to show, that the *centred DFT* is given by a diagonal swapping of the quadrants in the DFT (Example 5.2, Figure 5.12).

Centred DFT: method 2

If $f(x, y)$ is an $M \times N$ array with DFT $F(u, v)$, its *centred DFT* is given by swapping the first quadrant of $F(u, v)$ with the third and swapping the second quadrant with the fourth.

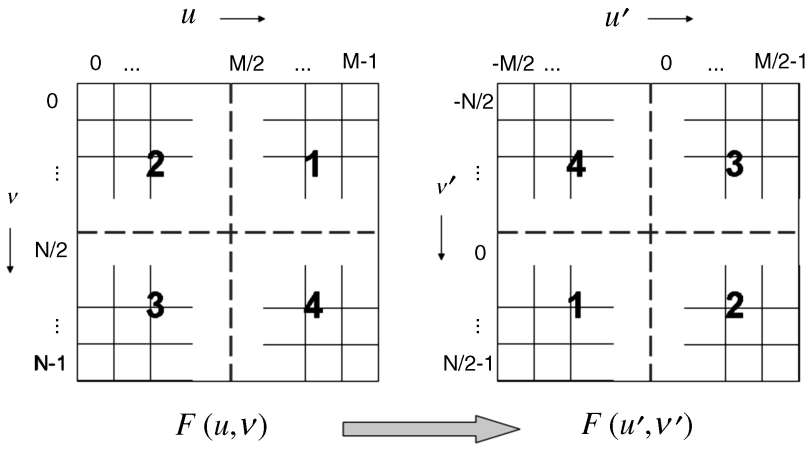

Figure 5.12 The centred DFT (right) can be calculated from the uncentred DFT (left) by dividing the array into four rectangles with two lines through the centre and diagonally swapping the quadrants. Using the labels indicated, we swap quadrant 1 with quadrant 3 and quadrant 2 with quadrant 4

Example 5.2

Matlab code	What is happening?
A=imread('cameraman.tif');	%Read in image
FT=fft2(A); FT_centred=fftshift(FT);	%Take FT, get centred version too
subplot(2,3,1), imshow(A);	%Display image
subplot(2,3,2), imshow(log(1 + abs(FT)),[]);	%Display FT modulus (log scale)
subplot(2,3,3), imshow(log(1 + abs(FT_centred)),[]);	%Display centred FT modulus (log scale)
Im1=abs(ifft2(FT)); subplot(2,3,5), imshow(Im1,[]);	%Inverse FFT and display
Im2=abs(ifft2(FT_centred)); subplot(2,3,6), imshow(Im1,[]);	%Inverse FFT and display
figure;	
[xd,yd]=size(A); x=-xd./2:xd./2-1; y=-yd./2:yd./2-1;	
[X,Y]=meshgrid(x,y); sigma=32;	
arg=(X.^2 + Y.^2)./sigma.^2; frqfilt=exp(-arg);	%Construct freq domain filter
imfilt1=abs(ifft2(frqfilt.*FT));	%Centred filter and noncentred spectrum
imfilt2=abs(ifft2(frqfilt.*FT_centred));	%image – centred filter on centred spectrum
subplot(1,3,1), imshow(frqfilt,[]);	%Display results
subplot(1,3,2), imshow(imfilt1,[]);	
subplot(1,3,3), imshow(imfilt2,[]);	

Comments

Matlab functions: ***fft2***, ***ifft2***, ***fftshift***.

This example illustrates the role played by ***fftshift***, which centres the Fourier transform such that the zeroth spatial frequency pair is at the centre of the array.

Why should we centre the DFT? The answer is that we do not *have* to do this, but there are two good reasons in its favour. First, shifting the origin to the centre makes the discrete frequency-space range $(-M/2 \rightarrow M/2-1; -N/2 \rightarrow N/2-1)$ more akin to the continuous space in which we have an equal distribution of positive and negative frequency components $(-\infty \leq k_x \leq \infty; -\infty \leq k_y \leq \infty)$. Second, the construction of frequency-domain filters for suppressing or enhancing groups of spatial frequencies is generally facilitated by using a centred coordinate system.

For further examples and exercises see http://www.fundipbook.com